Chapter 7
Ordinary Least Squares Estimation

7.1 Introduction

Relationships. We have relationships with family, friends, romantic partners, pets, maybe even with a favorite book. And yes, data may have relationships as well.

Take the team-level Major League Baseball statistics for 2025.[1]

If you're a baseball fan, it probably wouldn't surprise you to hear that Earned Run Average (ERA), the average opponent score per nine-innings, may affect wins.[2] If you need any convincing, check out Figure 7.1. As we might expect, there seems to be a *negative* relationship: as ERA increases (so the opponents are scoring more), the less wins you are expected to have.

Along with the scatter plot, you may have noticed the red line that appears to move in the general direction of the points. You may have heard this referred to as a "line of best fit." But this line is more than a visual aid. Once determined, it gives us a rule. It allows us to predict y for any given value of x, and its slope summarizes how y tends to change as x changes. In other words, the line transforms a pattern into a measurable relationship.

7.1.1 A Brief History of Least Squares

The idea of fitting a line through data is so natural that you might assume it has always been around. It hasn't. The method of least squares was invented independently by **Adrien-Marie Legendre** in 1805 and **Carl Friedrich Gauss** in 1809, and both were trying to solve the same problem: tracking celestial bodies. In 1801, the asteroid Ceres was observed for just 41 days before it disappeared behind the Sun. Using the few avail-

[1] Baseball-Reference: 2025 MLB Team Statistics.

[2] If you aren't a baseball fan, then just trust me.

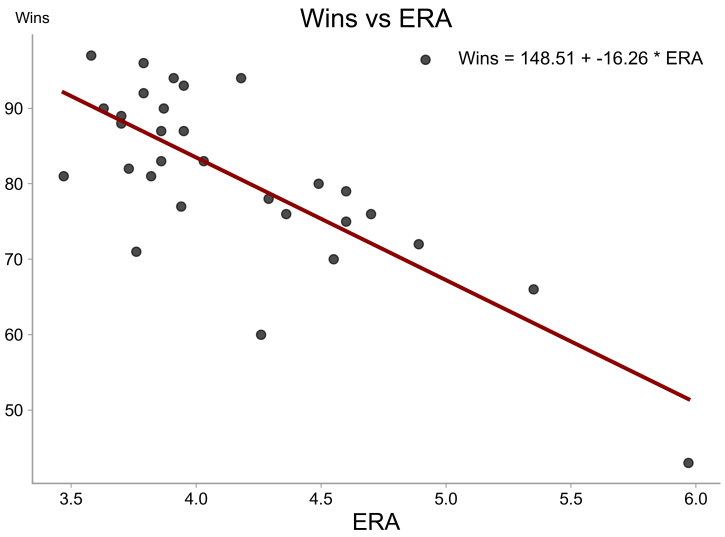


Fig. 7.1: Team wins versus ERA for the 2025 MLB season. Each point is one team. The red line is the OLS fitted line, showing a negative relationship: higher ERA (more runs allowed) is associated with fewer wins.

able observations, Gauss fit a curve by minimizing the sum of squared deviations and predicted where Ceres would reappear months later. He was right.[3]

For nearly a century, least squares was a technique for fitting curves. You could draw a line through your data, but you had no formal way to ask: *how precise is this line?* Is the slope meaningfully different from zero, or could we have gotten a similar slope by chance?

In 1886, **Francis Galton** gave us the word "regression" while studying hereditary height. He noticed that exceptionally tall parents tended to have children who were tall but closer to the population average. Heights "regressed toward mediocrity," as he put it. The name stuck, even though modern regression has nothing to do with reverting to the mean.

With Legendre and Gauss we got the method, and with Galton we got the name. In this chapter, we focus on the foundational mechanics: how to derive the line and what assumptions make it work. Later chapters will build on this foundation, extending the model to handle multiple variables (Chapter 8), measuring how well the line fits (Chapter 9.1), and testing whether the slope is meaningfully different from zero (Chapter 10).

[3] Gauss claimed to have discovered the method before Legendre, which led to one of mathematics' most entertaining priority disputes. Both contributed enormously to the theory.

7.1.2 The Equation of a Line

In grade-school notation, the equation of a line is

$$y = mx + b.$$

where m is the slope and b is the y-intercept. We can rewrite the same line using different letters:

$$y = \beta_0 + \beta_1 x,$$

where β_0 is the **intercept** (where the line crosses the vertical axis) and β_1 is the **slope** (how much y changes when x increases by 1).

Notice that there are no points that fall exactly on the line, though some are close. Real data are noisy. Some y_i are above the line, some are below it. So we need a way to write that idea formally: a line plus some deviation from the line.

7.1.3 Introduction to a Simple Linear Model

To allow for noise, we write

$$y_i = \beta_0 + \beta_1 x_i + u_i. \tag{7.1}$$

Here, x_i is the variable we use to explain variation in y_i. For that reason, it is often called the **explanatory variable**. You may also see it called an **independent variable**.[4] The variable y_i is the **outcome**.[5]

The term u_i measures everything that pushes y_i away from the straight-line relationship with x_i. Equivalently, u_i is how **far off** the line we are for observation i:

$$u_i = y_i - (\beta_0 + \beta_1 x_i).$$

[4] Unfortunately, there are a number of names that it can go by. We could also call it a **predictor**, a **covariate**, or a **regressor**.

[5] Similar to x_i, we have many names for y_i. It is sometimes called the **dependent variable**,**regressand**, or the **response**.

We have just written down a linear model that describes a relationship between two variables. Before we move on, let's discuss some assumptions about it.

Classical Linear Regression Assumption 1: Linearity in Parameters

$y_i = \beta_0 + \beta_1 x_i + u_i$.
The model is linear in the parameters. Each coefficient multiplies a variable (or a constant) and the resulting terms are added together. The parameters are not squared, multiplied by each other, or placed inside nonlinear functions.

This does **not** restrict x_i *at all*. We are free to include x_i^2, $\log x_i$, $1/x_i$, interaction terms, or any other transformation of the data. As long as the unknown parameters enter additively, the model is "linear" in the sense that matters here. (We will explore nonlinear transformations of x extensively in Chapter 15.) There are certainly other ways that we could model a relationship between variables, but we will see that this way has a lot of nice properties.

So far we have talked as if our data are simply "there," like a natural sample from the world. But real datasets do not fall from the sky. Someone decided what to measure, when to measure it, and which observations to include.

Sometimes that selection is harmless. Other times it creates a trap. If the way observations enter the sample is related to the variables we care about, the pattern we see in the data may reflect the sampling rule rather than the true relationship in the population. When we compute $\bar{y}$ or $\frac{1}{n}\sum x_i u_i$, we are relying on the fact that each observation is an independent draw from the same population. Without random sampling, patterns in the data might reflect how the sample was collected rather than the true relationship between x and y.

For example, suppose you only collected MLB data on teams that made the playoffs. It sounds silly, but this kind of selection happens all the time in practice. The relationship between ERA and wins in that restricted sample could look very different from the relationship in the full league. In the playoffs-only dataset, the "bad" teams have already been filtered out.

To rule out this kind of selection, we impose a sampling assumption.

Classical Linear Regression Assumption 2: Random Sampling

The observations $\{(x_i, y_i)\}_{i=1}^n$ are drawn randomly from the population. Each observation follows the same relationship $y_i = \beta_0 + \beta_1 x_i + u_i$, and the draws are independent of one another.

At this point, we should pause and ask an important question: where do the numbers β_0 and β_1 come from?

The line we drew in Figure 7.1 is based on a particular dataset, the 2025 MLB season. But imagine that we could observe every possible season of Major League Baseball: past seasons, future seasons, seasons that never quite happened because of injuries or trades.

Across all of those possible realizations, there would be some underlying relationship between ERA and wins.

That underlying relationship,the one that governs the entire population of possible observations, is what we mean by the **population line**. Its slope and intercept, β_1 and β_0, are fixed but *unknown* numbers. They describe how wins and ERA are related in the broader process that generates the data.

The problem is that we don't get to see the entire population. We only observe a sample. In this case, one season. So we do not know the true β_0 and β_1. Instead, we must estimate them from the data we have. If we knew the *true* β_0 and β_1, the line $\beta_0 + \beta_1 x_i$ would be the **population line**, and u_i would be the true (unobserved) error.

Since we don't know the true values, we have to estimate them using a process known as "regression", which we will go over later in this chapter. In order to distinguish these estimates from the real deal, we put a "hat" on them: $(\hat{\beta}_0, \hat{\beta}_1)$. We refer to these verbally as "Beta Zero Hat" and "Beta One Hat". Now, we can compute the **estimate** of u_i, $\hat{u}_i$ which we call the **residuals**. They are the *actual* vertical distances between each point in our sample and the fitted line.

$$\hat{u}_i = y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i. \tag{7.2}$$

7.2 The Model and Exogeneity

7.2.1 The Simple Linear Regression Model

Let's start by restating our simple model.

$$y_i = \beta_0 + \beta_1 x_i + u_i, \quad i = 1, \ldots, n. \tag{7.3}$$

Where:

- y_i is the outcome (or dependent variable) for observation i,
- x_i is the explanatory (or independent) variable for observation i,
- β_0 is the intercept parameter,
- β_1 is the slope parameter,
- u_i is the unobserved error term for observation i, and
- n is the number of observations in the sample.

7.2.2 The Exogeneity Assumption

Let's think about something that could go wrong. Suppose you collect data on ice cream sales and drowning deaths across several dozen time periods. You plot the points and draw the best-fitting line through them. The data show a strong positive relationship: more ice cream sold, more drownings. Should cities ban ice cream to save lives?

Obviously not.

The real driver is temperature. When it is hot outside, people buy more ice cream *and* more people go swimming, which increases the risk of drowning. Temperature affects both x (ice cream sales) and y (drowning deaths). But if temperature is not included in the model, its effect does not disappear. It has to go somewhere. In our model,

$$y_i = \beta_0 + \beta_1 x_i + u_i,$$

anything that affects y_i but is not explicitly included is absorbed into the error term u_i.

Figure 7.2 illustrates this idea. The line shows a strong positive slope. But that slope does not reflect a direct relationship between ice cream and drowning. Instead, it reflects the influence of a third factor that has been left out.

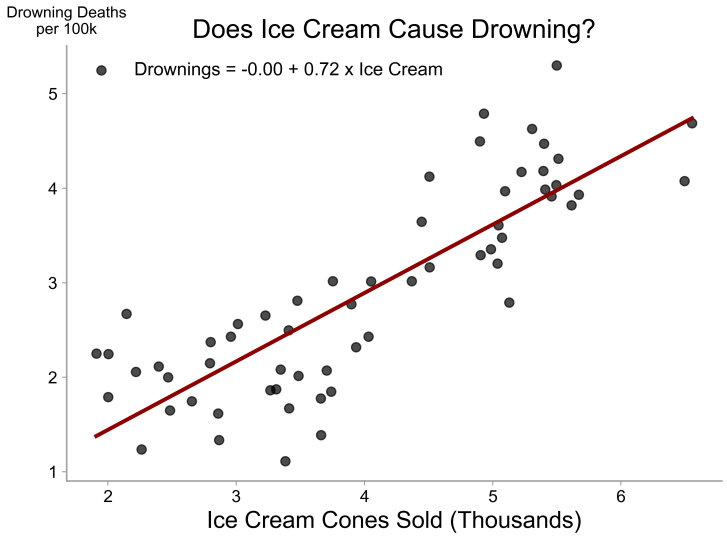


Fig. 7.2: A fictional dataset of ice cream sales and drowning deaths. The line shows a strong positive relationship. However, both variables are driven by temperature, which is not included in the model. The error term u_i absorbs the effect of temperature.

Although we cannot observe the true error term u_i, we can reason about what it must contain. In hot months, ice cream sales are high. But hot weather also increases

drownings directly, and that influence is not included in the model. It therefore becomes part of u_i.

Now comes the key point. Temperature and ice cream sales tend to move together. That means the temperature component inside u_i is systematically larger when x_i is larger. The error term is no longer pure random variation. It contains a hidden force that is itself correlated with x_i.

This creates a serious problem. The line cannot distinguish between the effect of ice cream and the effect of temperature. When ice cream sales are high, temperature is high. When temperature is high, drownings are high. The line therefore attributes part of temperature's effect to ice cream. The slope we estimate mixes the true effect of x with the effect of the omitted variable.

We must rule out this situation. To do so, we impose the following assumption:

Classical Linear Regression Assumption 3: Exogeneity

$\mathbb{E}[u_i \mid x_i] = 0$.
This condition states that, in the population, the expected value of the error term is zero at every value of x_i.[a]

[a] It is important to emphasize that this condition refers to the unobserved population error u_i, not the residuals $\hat{u}_i$. The exogeneity assumption is a statement about the underlying data-generating process, not about the fitted residuals. The residuals are constructed from the fitted line and, as we will see later, have an average value of zero by design.

Assumption 3 states that once we account for x_i, there is no systematic tendency for the error term to be positive or negative. The error may vary across observations, but its conditional expectation does not drift upward or downward as x_i changes. If this condition holds, the slope reflects the true relationship between x and y within the model. If it fails, the slope can be misleading. If we assume it holds, we get two important implications.

Implication 1: Mean of the Error Is Zero

By the Law of Iterated Expectations (Appendix A):

$$\begin{aligned} \mathbb{E}[u_i] &= \mathbb{E}\big[\mathbb{E}[u_i \mid x_i]\big] && \text{Law of Iterated Expectations} \\ &= \mathbb{E}[0] && \text{Exogeneity: } \mathbb{E}[u_i \mid x_i] = 0 \\ &= 0 && \text{Expectation of a constant} \end{aligned}$$

Implication 2: No Systematic Relationship Between x and Error ($\mathbb{E}[x_i u_i] = 0$)

Again use iterated expectations:

$$\begin{aligned}
\mathbb{E}[x_i u_i] &= \mathbb{E}\big[\mathbb{E}[x_i u_i \mid x_i]\big] && \text{Law of Iterated Expectations} \\
&= \mathbb{E}\big[x_i \, \mathbb{E}[u_i \mid x_i]\big] && \text{Pull out } x_i \text{ (fixed under conditioning)} \\
&= \mathbb{E}[x_i \cdot 0] && \text{Exogeneity: } \mathbb{E}[u_i \mid x_i] = 0 \\
&= 0 && \text{Anything times zero is zero}
\end{aligned}$$

From Population Moments to Sample Equations

As we see above, the exogeneity assumption implies two population conditions:

$$\mathbb{E}[u_i] = 0 \quad \text{and} \quad \mathbb{E}[x_i u_i] = 0.$$

These are statements about the true, unobservable error u_i in the population. We cannot check them directly. But they suggest a strategy: *find the line whose residuals satisfy the sample versions of these conditions*. That is, choose $\hat{\beta}_0$ and $\hat{\beta}_1$ so that:[6]

$$\sum_{i=1}^{n} \hat{u}_i = 0 \quad \text{and} \quad \sum_{i=1}^{n} x_i \hat{u}_i = 0. \tag{7.4}$$

These two equations are called the **normal equations**.[7] We will refer back to them frequently.

What did we just do? The logic we followed has a name: the **method of moments** (also called the **analog principle**).[8] We started with population moment conditions implied by exogeneity, replaced population expectations with sample averages, and obtained equations that pin down $\hat{\beta}_0$ and $\hat{\beta}_1$. The resulting equations happen to be called the "normal equations" because of their geometric meaning (perpendicularity), but the *derivation path* we took to get here is the method of moments.

---

[6] Notice the shift: the population conditions involve u_i (which we never observe), while the sample conditions involve $\hat{u}_i = y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i$ (which depend on our choice of coefficients). The population conditions are assumptions about the world. The sample conditions are equations we *impose* to pin down our estimates.

[7] The word "normal" here does not mean "ordinary." It comes from the Latin *normalis*, meaning "perpendicular." The normal equations require the residual to be perpendicular to the regressors. We will see this geometric meaning vividly in Chapter 8.

[8] The idea generalizes far beyond OLS. Any time you have population conditions of the form $\mathbb{E}[g(y_i, x_i, \theta)] = 0$ and you replace the expectation with a sample average, you are doing method of moments estimation. Lars Peter Hansen's **Generalized Method of Moments** (GMM) builds an entire estimation framework around this principle. OLS is the simplest special case: the moment conditions are $\mathbb{E}[u_i] = 0$ and $\mathbb{E}[x_i u_i] = 0$, and the sample analogs are the two equations above.

Notice the logic carefully. We are not claiming that the sample analogs *follow from* exogeneity. The population conditions motivated us to look for an estimator with these properties, but the sample conditions are *defining equations*: they are the system we solve to obtain $\hat{\beta}_0$ and $\hat{\beta}_1$. Later, in Section 7.4, we will see that minimizing the sum of squared residuals forces these same two conditions to hold mechanically. So two seemingly different strategies ("make the residuals uncorrelated with x" and "make the squared residuals as small as possible") lead to exactly the same estimator.

7.3 Derivation of the SLR Model Using Exogeneity Assumption As Motivation

7.3.1 Solving for $\hat{\beta}_0$

There are a few ways to derive the estimates for the parameters of our model. For a simple version like what we have been looking at, we can do so with just algebra. We only need the two normal equations above. Since we are solving for the *estimates* $\hat{\beta}_0$ and $\hat{\beta}_1$ (not the true population parameters), we will use hats throughout. Bars over a variable indicate the mean of that variable, such as $\bar{y}$.

To get started, we solve the first normal equation $\sum_{i=1}^{n} \hat{u}_i = 0$ step by step.

$$\sum_{i=1}^{n} \left(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i\right) = 0 \qquad \text{First normal equation: } \textstyle\sum \hat{u}_i = \sum_{i=1}^{n} \left(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i\right) = 0$$

$$\sum_{i=1}^{n} y_i - \sum_{i=1}^{n} \hat{\beta}_0 - \sum_{i=1}^{n} \hat{\beta}_1 x_i = 0 \qquad \text{Distribute } \textstyle\sum \text{ over the three terms}$$

$$\sum_{i=1}^{n} y_i - n\hat{\beta}_0 - \hat{\beta}_1 \sum_{i=1}^{n} x_i = 0 \qquad \text{Pull constants out: } \textstyle\sum \hat{\beta}_0 = n\hat{\beta}_0$$

$$n\hat{\beta}_0 = \sum_{i=1}^{n} y_i - \hat{\beta}_1 \sum_{i=1}^{n} x_i \qquad \text{Isolate } n\hat{\beta}_0 \text{ on the left}$$

$$\hat{\beta}_0 = \frac{1}{n} \sum_{i=1}^{n} y_i - \hat{\beta}_1 \frac{1}{n} \sum_{i=1}^{n} x_i \qquad \text{Divide both sides by } n$$

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x} \qquad \bar{y} = \tfrac{1}{n} \textstyle\sum y_i,\ \bar{x} = \tfrac{1}{n} \sum x_i \qquad (7.5)$$

7.3.2 Solving for $\hat{\beta}_1$

We now move to the second normal equation, which comes from setting the average product of x_i and the residuals to zero.

$$
\begin{aligned}
\sum_{i=1}^{n} x_i\left(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i\right) = 0 \qquad & \sum x_i \hat{u}_i = \textstyle\sum_{i=1}^{n} x_i\left(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i\right) = 0 \\
\sum_{i=1}^{n} x_i\Big(y_i - \underbrace{(\bar{y} - \hat{\beta}_1 \bar{x})}_{\hat{\beta}_0} - \hat{\beta}_1 x_i\Big) = 0 \qquad & \text{Substitute } \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x} \\
\sum_{i=1}^{n} x_i\Big((y_i - \bar{y}) + \hat{\beta}_1 \bar{x} - \hat{\beta}_1 x_i\Big) = 0 \qquad & \text{Expand parentheses; separate components} \\
\sum_{i=1}^{n} \Big[x_i(y_i - \bar{y}) + \hat{\beta}_1 x_i(\bar{x} - x_i)\Big] = 0 \qquad & \text{Group: terms with } \hat{\beta}_1 \text{ vs. without; distribute } x_i \\
\sum_{i=1}^{n} x_i(y_i - \bar{y}) + \hat{\beta}_1 \sum_{i=1}^{n} x_i(\bar{x} - x_i) = 0 \qquad & \text{Split sum; factor out } \hat{\beta}_1 \\
\sum_{i=1}^{n} x_i(y_i - \bar{y}) - \hat{\beta}_1 \sum_{i=1}^{n} x_i(x_i - \bar{x}) = 0 \qquad & \text{Flip sign: } \bar{x} - x_i = -(x_i - \bar{x}) \\
\sum_{i=1}^{n} x_i(y_i - \bar{y}) = \hat{\beta}_1 \sum_{i=1}^{n} x_i(x_i - \bar{x}) \qquad & \text{Move } \hat{\beta}_1 \text{ term to the right}
\end{aligned}
\tag{7.6}
$$

To solve this, we can make use of some useful identities.

7.3.3 Key Identity: $\sum x_i(y_i - \bar{y}) = \sum (x_i - \bar{x})(y_i - \bar{y})$

We start with $\sum_{i=1}^{n} x_i(y_i - \bar{y})$ and rewrite $x_i = (x_i - \bar{x}) + \bar{x}$ (adding and subtracting $\bar{x}$ nets to zero, so this changes nothing):

$$\begin{aligned}
\sum_{i=1}^{n} x_i(y_i - \bar{y}) &= \sum_{i=1}^{n} \Big[\underbrace{(x_i - \bar{x}) + \bar{x}}_{=x_i} \Big](y_i - \bar{y}) && \text{Rewrite } x_i \text{ as deviation + mean} \\
&= \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y}) \; + \; \bar{x} \sum_{i=1}^{n} (y_i - \bar{y}) && \text{Distribute and factor out constant } \bar{x} \\
&= \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y}) \; + \; \bar{x} \cdot 0 && \text{Deviations from mean sum to zero} \\
&= \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y}) && (7.7)
\end{aligned}$$

I love this result. The two terms only differ in the fact that one side subtracts $\bar{x_i}$ and the other does not, yet they are equal. Math truly is amazing. So we have it that once y_i is written with a deviation from its mean, you can subtract $\bar{x}$ from x_i "for free" without changing the total.

For the purposes of this formula, x and y can be any two variables, even the same one. Setting $y = x$ gives us a second identity:

$$\sum_{i=1}^{n} x_i(x_i - \bar{x}) = \sum_{i=1}^{n} (x_i - \bar{x})^2. \tag{7.8}$$

7.3.4 Plug the Identities In and Solve

Apply Identity (7.7) to the left of (7.6) and Identity (7.8) to the right:

$$\begin{aligned}
\sum_{i=1}^{n} x_i(y_i - \bar{y}) &= \hat{\beta}_1 \sum_{i=1}^{n} x_i(x_i - \bar{x}) && \text{Equation (7.6) from above} \\
\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y}) &= \hat{\beta}_1 \sum_{i=1}^{n} (x_i - \bar{x})^2 && \text{Apply identities (7.7) and (7.8)} \\
\hat{\beta}_1 &= \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2} && \text{Divide both sides by } \sum (x_i - \bar{x})^2 \quad (7.9)
\end{aligned}$$

Note that we could multiply the numerator and denominator by $\frac{1}{n-1}$ without changing the value, which gives us an interesting way to think about it:

$$\hat{\beta}_1 = \frac{\frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})^2} = \frac{S_{xy}}{S_{xx}}. \tag{7.10}$$

where S_{xy} is the sample covariance and S_{xx} is the sample variance. In other words, $\hat{\beta}_1$ is *how much x and y vary together relative to how much x varies on its own*.

Remember that we cannot divide by zero, so we must have it that $\sum(x_i - \bar{x})^2 \neq 0$. In fact, this is another assumption, but since there is actually a bit more to it than just this, we will hold off on declaring it for the moment.

Now that we have solved for $\hat{\beta}_1$, we have an actual numerical value to fill in our result for $\hat{\beta}_0$ from earlier:

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}. \tag{7.11}$$

Note that if we solve (7.11) for $\bar{y}$, we get $\bar{y} = \hat{\beta}_0 + \hat{\beta}_1 \bar{x}$. Since when the sample mean for x is the input, the sample mean of y is the output, this means the regression line always passes through the sample mean $(\bar{x}, \bar{y})$.

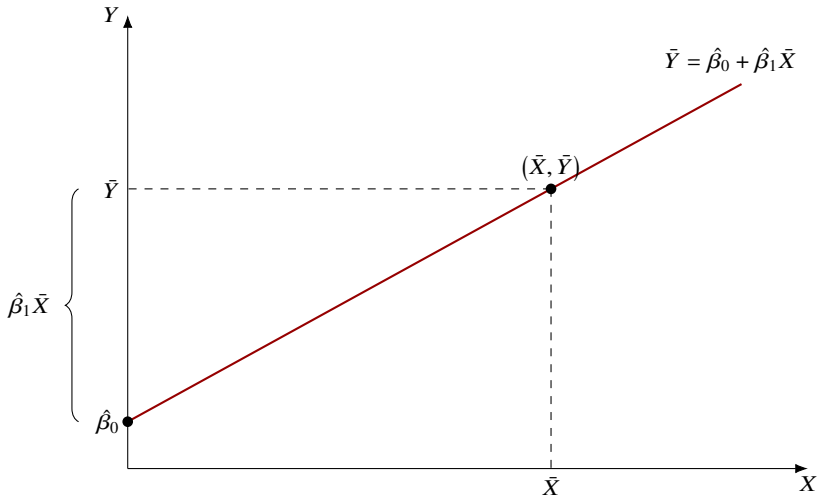


Fig. 7.3: Because $\bar{Y} = \hat{\beta}_0 + \hat{\beta}_1 \bar{X}$, the fitted line passes through $(\bar{X}, \bar{Y})$.

The formulas in action

We derived two formulas: $\hat{\beta}_1 = S_{xy}/S_{xx}$ and $\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$. Let's see what they produce with real data. We will use MLB team batting statistics to predict runs per game from on base percentage. The code computes the sample means, the cross products S_{xy} and S_{xx}, and then plugs them into the formulas.

For the Stata and Python code, see Appendices Appendix G and Appendix H.

Output:

```
Constant = -5.9317
On Base Pct = 32.9565
```

The coefficient on On Base Percentage is $\hat{\beta}_1 \approx 32.96$. Since OBP is measured as a proportion (for example, 0.330), this says: a team whose OBP is higher by 0.010 is predicted to score about $32.96 \times 0.010 \approx 0.33$ more runs per game. Over a 162 game season that is roughly 53 extra runs, which is a big deal. The constant $\hat{\beta}_0 \approx -5.93$ is the predicted runs per game for a team with an OBP of zero. That is not a meaningful number (no team has an OBP near zero), but the formula needs it to anchor the line.

Every regression coefficient has the same interpretation: it is the predicted change in y for a one unit change in x, holding everything else constant. We will return to this idea repeatedly, especially in Chapter 8 where "holding everything else constant" becomes more subtle with multiple regressors.

7.4 Derivation of SLR Coefficients via Calculus (Ordinary Least Squares)

If you think back to your calculus courses, you may recall finding a minimum (or maximum) by taking a derivative and setting it equal to zero. At a minimum, the slope of the function is flat, so the derivative equals zero. By imposing that condition, we can solve for the values that make it true.

For our model in particular, we want to choose the parameters so that the fitted line matches the observed data as closely as possible. That raises a natural question: what exactly should we minimize?

The procedure of choosing coefficients to *best* fit the data is called **regression**. What is "best" though? In this chapter, we adopt a particular criterion called **ordinary least squares** (OLS), which chooses the coefficients to minimize the sum of the squared residuals. The term *least squares* refers to this minimization of squared deviations, and *ordinary* simply means that each observation is weighted equally—no observation is given special treatment.

We square the residuals so that negative and positive deviations do not cancel each other out, and so that larger mistakes are penalized more heavily than smaller ones. We define the sum of squared residuals (SSR) as:

$$S(\beta_0, \beta_1) = \sum_{i=1}^{n} \left(y_i - \beta_0 - \beta_1 x_i\right)^2.$$

To find the minimizing values of β_0 and β_1, we take partial derivatives of $S(\beta_0, \beta_1)$ with respect to each parameter and set them equal to zero. These derivative conditions are called the **first-order conditions (FOCs)**.[9]

7.4.1 First-Order Conditions for Simple Regression

Taking partial derivatives and setting them to zero:

$$\frac{\partial S}{\partial \beta_0} = \sum_{i=1}^{n} 2\big(y_i - \beta_0 - \beta_1 x_i\big)(-1) = 0 \quad \text{Chain rule on } (y_i - \beta_0 - \beta_1 x_i)^2$$

$$\implies \quad \sum_{i=1}^{n} \big(y_i - \beta_0 - \beta_1 x_i\big) = 0 \quad \text{Divide by } -2;\ \text{same as } 1^{st} \text{ normal equation}$$

$$\frac{\partial S}{\partial \beta_1} = \sum_{i=1}^{n} 2\big(y_i - \beta_0 - \beta_1 x_i\big)(-x_i) = 0 \quad \text{Chain rule; inner derivative is } -x_i$$

$$\implies \quad \sum_{i=1}^{n} x_i\big(y_i - \beta_0 - \beta_1 x_i\big) = 0 \quad \text{Divide by } -2;\ \text{same as } 2^{nd} \text{ normal equation}$$

These are *exactly* the same normal equations we wrote down earlier. In Section 7.3, exogeneity *motivated* us to impose

$$\sum \hat{u}_i = 0 \quad \text{and} \quad \sum x_i \hat{u}_i = 0.$$

Now we see that minimizing the sum of squared residuals *forces* those same two conditions to hold, regardless of whether exogeneity is true. The calculus does not know or care about our assumptions. It simply finds the coefficients that make the squared mistakes as small as possible, and the first order conditions that result are the normal equations.

So the two derivations are not just two ways to get the same answer by coincidence. They express the same underlying principle: the best fitting line is the one whose residuals have no systematic pattern with respect to x. Once those conditions hold, the same algebra as in Section 7.3 gives

$$\hat{\beta}_1 = \frac{S_{xy}}{S_{xx}} \quad \text{and} \quad \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}.$$

[9] They are called *first-order* conditions because they come from setting the first derivatives equal to zero. A natural follow up question: how do we know the FOCs give us a minimum and not a maximum? See Appendix Appendix B.

7.5 Summary

Let's take stock of what we have established. Starting from a simple linear model

$$y_i = \beta_0 + \beta_1 x_i + u_i,$$

and three assumptions (linearity, random sampling, and exogeneity), we derived the OLS estimators using two different approaches.

- **Via moment conditions** (Section 7.3): The exogeneity assumption implies population conditions $\mathbb{E}[u_i] = 0$ and $\mathbb{E}[x_i u_i] = 0$. We *imposed* the sample analogs $\sum \hat{u}_i = 0$ and $\sum x_i \hat{u}_i = 0$ as defining equations. Solving them algebraically gives us closed form expressions for the slope and intercept.
- **Via calculus** (Section 7.4): Minimizing the sum of squared residuals $S(\beta_0, \beta_1) = \sum (y_i - \beta_0 - \beta_1 x_i)^2$ forces exactly the same normal equations to hold, and therefore gives the same solutions.

Both roads lead to the same destination:

$$\hat{\beta}_1 = \frac{S_{xy}}{S_{xx}} = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2}, \qquad \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}.$$

We also noted two useful properties of the fitted line:

- The regression line always passes through the point $(\bar{x}, \bar{y})$.
- The slope $\hat{\beta}_1$ equals the sample covariance of x and y divided by the sample variance of x. It measures how much x and y move together, scaled by how much x varies on its own.

These formulas are elegant, but they only handle one explanatory variable at a time. In practice, outcomes are influenced by many variables simultaneously. In Chapter 8, we will rewrite the model in matrix form and derive the general OLS estimator $\hat{\boldsymbol{\beta}} = (X'X)^{-1}X'\mathbf{y}$, which handles any number of regressors at once. We will also establish the key statistical properties of that estimator: unbiasedness, its variance, and efficiency.

Chapter 8
Multiple Regression and Properties of OLS

8.1 Introduction

In 1925, **R. A. Fisher** used thirty years of wheat yield data from the Broadbalk experiment at Rothamsted, England, the world's longest running agricultural trial (since 1843), to publish the first regression accompanied by a quantitative measure of its own sampling uncertainty, what we call the *standard error*. His 1925 textbook, *Statistical Methods for Research Workers*, is the first place where standard errors appear alongside regression coefficients. By the end of this chapter, we will reproduce Fisher's exact numbers from his original data.

To get there, we need to generalize the simple regression formulas to handle any number of regressors at once. The language that makes this reasonable is matrix algebra (Chapter 2). In this chapter, we rewrite the model in matrix form, derive the OLS estimator $\hat{\boldsymbol{\beta}} = (X'X)^{-1}X'\mathbf{y}$, and then establish its three most important properties: unbiasedness, a formula for its variance, and optimality among all linear unbiased estimators (the Gauss–Markov theorem).

8.2 From Simple to Multiple Regression

In Chapter 7, we worked with a single explanatory variable. That is called **simple** regression. But in most real applications, we want to allow more than one variable to influence the outcome at the same time.

Think back to our baseball example. Wins probably depend on more than just ERA. What about batting average? Home runs? Stolen bases? If we want to include all of these at once, our model gets longer:

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \cdots + \beta_{k-1} X_{(k-1)i} + u_i.$$

Writing out all those subscripts gets tedious quickly and the calculations done in the same way get much more challenging. Using the matrix tools from Chapter 2, we can pack the entire model for all n observations into one compact expression:

$$\mathbf{y} = X\boldsymbol{\beta} + \mathbf{u}, \tag{8.1}$$

where $\mathbf{y}$ is the $n \times 1$ vector of outcomes, X is the $n \times k$ matrix containing the explanatory variables (with the first column typically all ones for the intercept), $\boldsymbol{\beta}$ is the $k \times 1$ vector of coefficients, and $\mathbf{u}$ is the $n \times 1$ vector of error terms.[1] Similar to the original formula, $\mathbf{y} - X\boldsymbol{\beta}$ is the vector of residuals.

Our goal is the same as before: choose the coefficients that make the sum of squared residuals as small as possible. In Section 7.4, we called this quantity $S(\beta_0, \beta_1)$. Now that we are working with vectors and matrices, the same quantity can be written as:

$$S(\boldsymbol{\beta}) = (\mathbf{y} - X\boldsymbol{\beta})'(\mathbf{y} - X\boldsymbol{\beta}).$$

Multiplying a vector by its own transpose gives the sum of its squared entries (see Chapter 2), so this is the sum of squared residuals. We can expand this using the identity[2] $(a - b)'(a - b) = a'a - 2a'b + b'b$:

$$S(\boldsymbol{\beta}) = \mathbf{y}'\mathbf{y} - 2\mathbf{y}'X\boldsymbol{\beta} + \boldsymbol{\beta}'X'X\boldsymbol{\beta} \qquad \text{Expand the product}^3$$

Now we minimize, just as we did before: take the derivative and set it equal to zero. The only difference is that now we are differentiating with respect to an entire vector $\boldsymbol{\beta}$ instead of a single variable. The matrix calculus rules work out as follows.[4]

$$\frac{\partial S}{\partial \boldsymbol{\beta}} = -2X'\mathbf{y} + 2X'X\boldsymbol{\beta} = \mathbf{0} \qquad \text{Set the derivative equal to zero}^5$$

$$X'X\,\hat{\boldsymbol{\beta}} = X'\mathbf{y} \qquad \text{Divide by 2 and rearrange} \tag{8.2}$$

$$\hat{\boldsymbol{\beta}} = (X'X)^{-1}X'\mathbf{y} \qquad \text{Pre-multiply both sides by } (X'X)^{-1} \tag{8.3}$$

[1] See Section 8.4 for a full explanation on why the intercept column is a vector of ones.

[2] This is the vector version of the familiar scalar identity $(a - b)^2 = a^2 - 2ab + b^2$. The transpose allows for multiplication when working with vectors, so $(a - b)'(a - b)$ expands in the same way.

[3] Matrix multiplication is not commutative, so the order matters. The term comes from $(X\boldsymbol{\beta})'(X\boldsymbol{\beta})$. Using $(AB)' = B'A'$, we get $(X\boldsymbol{\beta})' = \boldsymbol{\beta}'X'$, which gives $\boldsymbol{\beta}'X'X\boldsymbol{\beta}$. The dimensions also force this order: $\boldsymbol{\beta}$ is $k \times 1$, so $\boldsymbol{\beta}'X'X\boldsymbol{\beta}$ is 1×1, a scalar.

[4] The derivative of $\mathbf{a}'\boldsymbol{\beta}$ with respect to $\boldsymbol{\beta}$ is $\mathbf{a}$, and the derivative of $\boldsymbol{\beta}'A\boldsymbol{\beta}$ is $2A\boldsymbol{\beta}$ when A is symmetric. The matrix $X'X$ is always symmetric.

β in action

Equation (8.3) gives us the OLS estimates for all k coefficients, including the constant, at once. Just plug in X and $\mathbf{y}$, do the matrix arithmetic, and out come the answers. Let's see what it produces.[6] We will stack a column of ones for the intercept with On-base Percentage (OBP) and Slugging Percentage (SLG) to form the $n \times 3$ matrix X, build $(X'X)^{-1}$, and multiply through.

Matrix Output:

```
Constant: -4.9937
On Base Pct: 19.1979
Slugging Pct: 8.4121
```

The coefficient on On Base Percentage is about 19.20 and the coefficient on Slugging Percentage is about 8.41. In simple regression (Chapter 7), On Base Percentage alone had a coefficient of about 32.96. The coefficient shrank because some of what OBP was "explaining" about run scoring was really attributable to Slugging Percentage. Once we control for both, the credit is split more honestly between them. A team whose OBP is higher by 0.010 (holding SLG constant) is predicted to score about $19.20 \times 0.010 \approx 0.19$ more runs per game, and a team whose SLG is higher by 0.010 (holding OBP constant) gains about $8.41 \times 0.010 \approx 0.08$ runs per game. The "holding constant" part is a key part of multiple regression. Each coefficient measures the effect of one variable after accounting for the others.

Now let's do this with Fisher's data. In Fisher's 1925 book, he used wheat yield data from the Broadbalk experiment at Rothamsted. Table 29 in his textbook lists the dressed grain yields (in bushels per acre) for two adjacent plots over thirty years, 1855 to 1884. One plot received nitrate of soda (plot 9*a*), the other sulphate of ammonia (plot 7*b*). Fisher asked whether one type of nitrogen was gaining an advantage over time. His dependent variable was the difference $y_i = \text{yield}_{9a} - \text{yield}_{7b}$, and his independent variable was the year, coded as a simple trend. Let's build the matrices by hand and reproduce his numbers.

For the Stata and Python code, see Appendices Appendix G and Appendix H.

Output:

```
X'X =
  [30 465]
  [465 9455]

(X'X)^{-1} =
  [0.140230 -0.006897]
```

[5] The factor of 2 appears for the same reason it does when differentiating b^2 in ordinary calculus. The quadratic term $\boldsymbol{\beta}' X' X \boldsymbol{\beta}$ plays the role of a squared term. Just as $\frac{d}{db}(b^2) = 2b$, differentiating a quadratic form in $\boldsymbol{\beta}$ produces a factor of 2 and reduces the power by one. The matrix expression is the vector analog of the power rule.

[6] The Stata and Python code for these calculations appears in Appendices Appendix G and Appendix H.

```
  [-0.006897 0.000445]

Constant = 0.3317
Trend = 0.2668
Fisher reported: 0.2668
```

The slope is 0.2668: each passing year, the yield advantage of nitrate of soda over sulphate of ammonia grew by about 0.27 bushels per acre. Over the thirty year span, that accumulates to roughly eight bushels. Fisher's question was whether one nitrogen source was gaining an advantage, and this trend suggests it was.

8.2.1 Multicollinearity

You may recall one of our most basic requirements in the simple linear regression formula for $\hat{\beta}_1$ was *sample variation in X*: we needed $\sum_{i=1}^{n}(X_i - \bar{X})^2 \neq 0$, because that sum sits in the denominator of $\hat{\beta}_1$. We cannot divide by zero. (And practically speaking, if X never varies, why would we be looking for a relationship between X and Y in the first place?)

In multiple regression with regressors $X = x_1, \ldots, x_k$, the same idea generalizes:we need variation not just in each X_j individually, but also in their linear combinations. The last step in the derivation of Equation 8.3 requires $(X'X)$ to be invertible, which means X must have full column rank. This is the generalization of "variation in X" to the case of many regressors.

Classical Linear Regression Assumption 4: No Perfect Multicollinearity

The $n \times k$ regressor matrix X has full column rank: $\text{rank}(X) = k$: No column of X can be written as a linear combination of the other columns.

To see why, consider a simple example. Suppose your model has an intercept and one X variable, but that variable never changes:

$$X = \begin{pmatrix} 1 & 5 \\ 1 & 5 \\ 1 & 5 \end{pmatrix}.$$

Column 2 is just 5 times Column 1. The columns do not contain independent information: X does not have full column rank. As a result, $X'X$ is singular and cannot be inverted, so OLS cannot produce estimates. This is **perfect multicollinearity**: one column is an exact linear combination of another.

In ordinary algebra, you can divide by a number only if it is not zero. If you try to divide by zero, the operation is undefined. In matrix algebra, "dividing by" $X'X$ means multiplying by $(X'X)^{-1}$. That operation is only possible if the inverse exists. When

the columns of X are linearly dependent, $X'X$ has no inverse.[7] In practice, statistical software detects this problem and drops one of the perfectly collinear columns so that the remaining variables contain independent information and the model can be estimated.

8.2.2 Unbiasedness of OLS

In assessing the quality of our model, we should ask a simple question: is it biased? In other words, if we could repeat this estimation process over and over again on new samples drawn from the same population, would our estimates tend to land on the true parameter values, or would they systematically miss? *Systematically* is the key word. An estimate can be far from the truth in a particular sample and still be unbiased. Bias is not about being wrong once. It is about being wrong in a predictable direction on average. So when we ask whether OLS is unbiased, we are asking whether its statistical *expectation* equals the true parameter value.

We can make a quick check that OLS, if our assumptions hold, is unbiased. Given $\hat{\boldsymbol{\beta}} = (X'X)^{-1}X'\mathbf{y}$ and $\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{u}$, we can substitute $X\boldsymbol{\beta} + \boldsymbol{u}$ in for $\mathbf{y}$. Following some quick algebra, we can see that OLS is indeed unbiased:

$$
\begin{aligned}
\hat{\boldsymbol{\beta}} &= (X'X)^{-1}X'\mathbf{y} && \text{OLS formula} \\
&= (X'X)^{-1}X'(X\boldsymbol{\beta} + \boldsymbol{u}) && \text{Substitute } \mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{u} \\
&= (X'X)^{-1}X'X\boldsymbol{\beta} + (X'X)^{-1}X'\boldsymbol{u} && \text{Distribute} \\
&= \boldsymbol{\beta} + (X'X)^{-1}X'\boldsymbol{u} && \text{Since } (X'X)^{-1}X'X = I_k
\end{aligned} \tag{8.4}
$$

We see that our estimate, $\hat{\boldsymbol{\beta}}$, equals the true parameter $\boldsymbol{\beta}$ plus the term $(X'X)^{-1}X'\boldsymbol{u}$. So if there is bias, it must come from that extra piece. Bias, by definition, is a systematic error. To determine whether OLS is biased, we take expectations:

[7] In the simple regression case, this reduces to the requirement that x must vary. If every x_i is the same, then the second column of X is just a constant multiple of the intercept column. There is no independent movement in x, so we cannot separate its effect from the intercept. Algebraically, $\sum(x_i - \bar{x})^2 = 0$, which makes $S_{xx} = 0$ and prevents us from dividing by it. The matrix condition that $X'X$ be invertible is the general version of the requirement that we not divide by zero.

$$
\begin{aligned}
\mathbb{E}[\hat{\boldsymbol{\beta}}] &= \mathbb{E}\left[\boldsymbol{\beta} + (X'X)^{-1}X'\boldsymbol{u}\right] && \text{Take expectations} \\
&= \boldsymbol{\beta} + (X'X)^{-1}\underbrace{\mathbb{E}[X'\boldsymbol{u}]}_{=\mathbf{0}} && \boldsymbol{\beta} \text{ and } (X'X)^{-1} \text{ are constants} \\
&= \boldsymbol{\beta} && \text{Exogeneity: } \mathbb{E}[X'\boldsymbol{u}] = \mathbf{0}
\end{aligned} \tag{8.5}
$$

So while any single estimate may miss the true parameter, there is no systematic pull in one direction. In expectation, the OLS estimate of β, $\hat{\beta}$, equals β. That is unbiasedness.

What does unbiasedness look like? Let's return to the ice cream example from Section 3. Suppose we simulate the experiment 50,000 times: in each sample, temperature drives both ice cream sales and drowning deaths, but ice cream has *zero* causal effect on drowning. Figure 8.1 shows what happens.

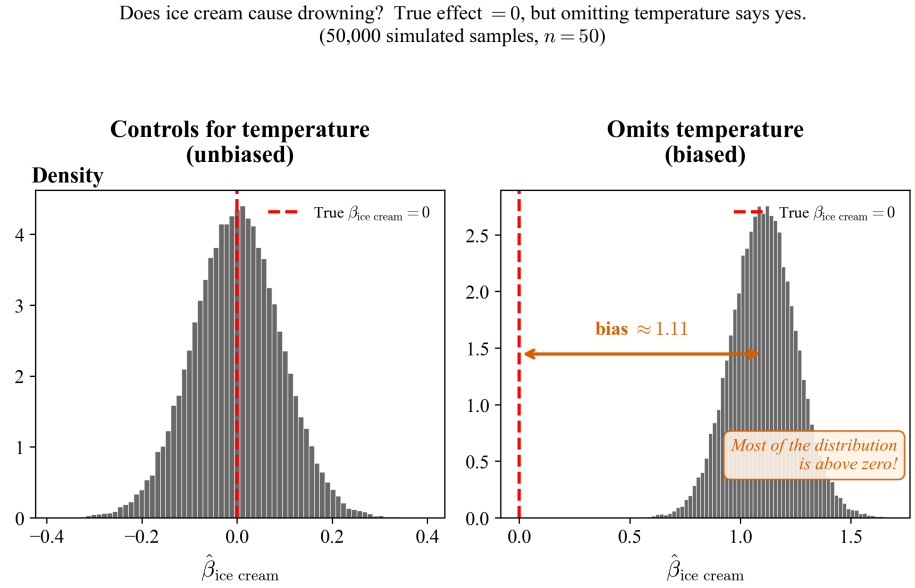


Fig. 8.1: The ice cream problem, simulated 50,000 times ($n = 50$). The true causal effect of ice cream on drowning is zero. **Left**: with temperature included, the estimated coefficient is centered on zero. **Right**: omitting temperature shifts the entire sampling distribution to the right. Almost all estimates are positive. Bias does not simply add noise; it systematically displaces the estimator from the true value.

The left panel is what OLS delivers when exogeneity holds: the distribution is centered on the truth. Individual estimates scatter around zero, but the misses cancel out on average. The right panel shows what happens when we violate exogeneity by omitting a

confound. The entire distribution shifts away from the truth. It is not that some samples are misleading. *Every* sample is misleading, systematically, in the same direction. A researcher looking at any one of those samples would find a "significant" positive effect. Collecting more data will not help: the bias does not shrink with n. Only including temperature in the model fixes the problem. OLS, under our assumptions, would give us the left panel, but it is up to us to ensure that we build our model thoughtfully. In truth, this assumption is the most likely to fail. See Chapter 12 for further discussion on what to do when assumptions fail.[8]

8.2.3 Variance of the OLS Estimator

Unbiasedness told us where OLS lands *on average*. Now we want to know how much it moves around from sample to sample. That is what the variance measures. A small variance means the estimator is stable across samples. A large variance means it is jumpy.

Figure 8.2 illustrates the idea with another Monte Carlo experiment. Both estimators are unbiased (centered on the true value), but they differ dramatically in how spread out they are.

The left panel is what we want: estimates tightly concentrated around the truth. The right panel is unbiased but unreliable. Any single estimate could be far off, even though the average is correct. This is why unbiasedness alone is not enough. We also need the variance to be small. The formula we are about to derive tells us exactly what determines how spread out $\hat{\boldsymbol{\beta}}$ is: it depends on σ^2 (the population noise level), $(X'X)^{-1}$ (the information in the regressors), and n (the sample size). The left panel has more data and less noise. The right panel has less data and more noise. That is the variance formula at work.

To compute this, we subtract $\boldsymbol{\beta}$ from both sides of Formula 8.4, which gives us an exact expression for the estimation error:

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

If the errors were zero, the estimator would equal the true parameter exactly. At this point we make an important clarification. In studying the sampling variability of OLS, we treat the observed matrix X as fixed. Once the data are collected, the X values are just numbers. The only randomness comes from the error vector $\mathbf{u}$. Under this view, $(X'X)^{-1}X'$ is a fixed matrix, and all variation in $\hat{\boldsymbol{\beta}}$ comes from $\mathbf{u}$.

[8] Once the fundamentals in this text are mastered, this is the natural next step. Entire books are devoted to these topics, largely because addressing violations of the classical assumptions can become technically and conceptually complex. Excellent starting points include *Causal Inference: The Mixtape* by Scott Cunningham, *Causal Inference in Python* by Matheus Facure, and *The Effect* by Nick Huntington-Klein.

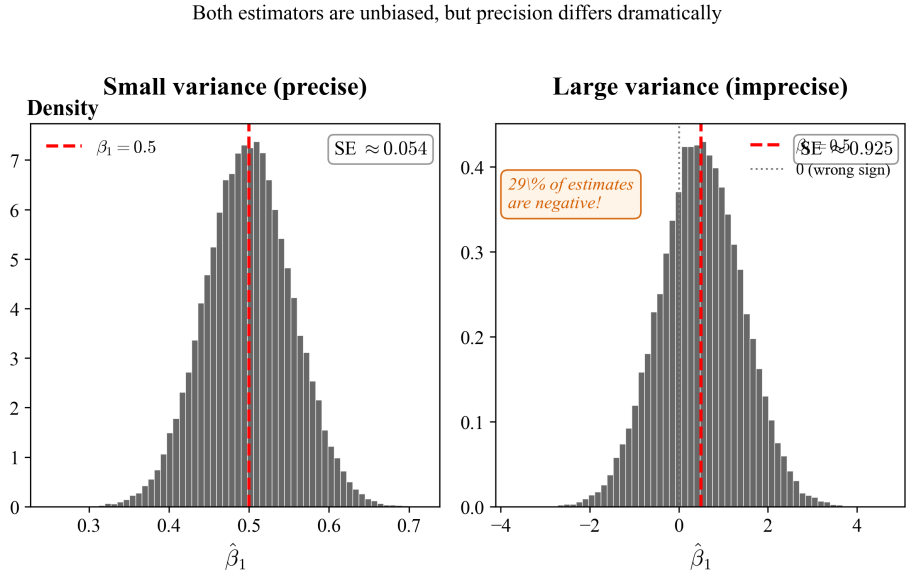


Fig. 8.2: Two unbiased estimators with different variances, each simulated 50,000 times. **Left**: $n = 200$ and $\sigma = 2$ produces a tightly concentrated distribution. Most samples produce estimates very close to $\beta_1 = 0.5$. **Right**: $n = 15$ and $\sigma = 10$ produces a widely spread distribution. Both are centered on the truth, but the right panel is far less useful in practice: any single estimate could be far off. Notice that a substantial fraction of the imprecise estimates are *negative*, giving the wrong sign entirely. The estimator is unbiased, so on average it is correct, but any individual sample could mislead you. This is exactly the situation where inference (hypothesis testing, confidence intervals) becomes essential: it tells you how seriously to take any single estimate.

Since $\boldsymbol{\beta}$ is a constant, it contributes no randomness. So the variance of $\hat{\boldsymbol{\beta}}$ is the variance of that right-hand side:

$$\begin{aligned}
\operatorname{Var}(\hat{\boldsymbol{\beta}}) &= \operatorname{Var}\Big(\boldsymbol{\beta} + (X'X)^{-1}X'\mathbf{u}\Big) && \text{Take variance of Equation 8.4} \\
&= \operatorname{Var}\Big((X'X)^{-1}X'\mathbf{u}\Big) && \boldsymbol{\beta} \text{ is constant, so it contributes no variance} \\
&= (X'X)^{-1}X' \operatorname{Var}(\mathbf{u})\, X(X'X)^{-1} && \text{Rule: } \operatorname{Var}(A\mathbf{z}) = A \operatorname{Var}(\mathbf{z})\, A' \text{ with } A = (X'X)^{-1}X'
\end{aligned} \tag{8.6}$$

Up to this point, no additional assumptions have been imposed. Everything follows directly from the algebra of the model. The expression in (8.6) is often called the *sand-*

wich form: the matrices $(X'X)^{-1}X'$ and its transpose form the "bread," and $\mathrm{Var}(\mathbf{u})$ is the "filling."

To simplify this expression further, we must say something about $\mathrm{Var}(\mathbf{u})$.

Classical Linear Regression Assumption 5: Homoskedasticity, No Serial Correlation

$$\text{Var}(\mathbf{u}) = \sigma^2 I_n.$$

This compact statement does two things at once.
Same spread everywhere (homoskedasticity).

$$\text{Var}(u_i) = \sigma^2 \quad \text{for all } i.$$

The size of the error does not systematically grow or shrink across observations.
No linkage across observations (no correlation).

$$\text{Cov}(u_i, u_j) = 0 \quad \text{for } i \neq j.$$

This means that once we account for X, the error for one observation tells us nothing about the error for another.
In matrix form:

$$\text{Var}(\mathbf{u}) = \sigma^2 \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}.$$

Here, σ is the population standard deviation of the error term, and σ^2 is its variance.

[9]

What does "correlation across observations" actually mean?

Suppose we are studying wages across workers in the same firm. Even after controlling for education and experience, there may be unobserved firm-wide shocks. For example, a particularly strong manager, a new bonus policy, or a bad year for the company. Those unobserved influences enter the error term. If one worker's wage error is high because of a firm-wide bonus, other workers' errors are likely to be high as well. In that case,

$$\text{Cov}(u_i, u_j) \neq 0.$$

In time series data, the idea is similar. If this year's error is positive because of an unobserved economic boom, next year's error may also be positive. That is serial correlation.

[9] You may see this called the "spherical errors" assumption (Greene, *Econometric Analysis*, Section 2.3.4; Hayashi, *Econometrics*, Assumption 1.4). The name refers to the fact that I_n weights every observation identically, giving the error vector no preferred direction. This is also the assumption behind the Gauss-Markov theorem: when it fails, OLS loses its optimality to Generalized Least Squares (Chapter 20).

What does that do to the variance matrix?

Under our assumption,

$$\operatorname{Var}(\mathbf{u}) = \sigma^2 I_n = \sigma^2 \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}.$$

That diagonal structure says two things: each observation has the same variance, and no two observations are correlated.

If errors are correlated across observations, the matrix instead looks like

$$\operatorname{Var}(\mathbf{u}) = \begin{bmatrix} \operatorname{Var}(u_1) & \operatorname{Cov}(u_1, u_2) & \cdots \\ \operatorname{Cov}(u_2, u_1) & \operatorname{Var}(u_2) & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix},$$

with nonzero entries off the diagonal. The simple formula

$$\operatorname{Var}(\hat{\boldsymbol{\beta}}) = \sigma^2 (X'X)^{-1}$$

relied on the identity matrix sitting in the middle of the sandwich expression. When $\operatorname{Var}(\mathbf{u})$ is no longer $\sigma^2 I_n$, that cancellation no longer happens. The variance becomes

$$\operatorname{Var}(\hat{\boldsymbol{\beta}}) = (X'X)^{-1} X' \operatorname{Var}(\mathbf{u})\, X (X'X)^{-1},$$

which generally does not simplify. The OLS coefficient estimates remain unbiased under exogeneity, but the variance formula $\sigma^2 (X'X)^{-1}$ is no longer correct when errors are correlated. In particular, with positive correlation across observations, the true sampling variability is typically larger than this formula suggests. As a result, the uncertainty surrounding the estimates is understated.

If our assumption holds, we can now plug the assumption into the sandwich:

$$\begin{aligned} \operatorname{Var}(\hat{\boldsymbol{\beta}}) &= (X'X)^{-1} X' (\sigma^2 I_n) X (X'X)^{-1} && \text{Substitute } \operatorname{Var}(\mathbf{u}) = \sigma^2 I_n \\ &= \sigma^2 (X'X)^{-1} X' I_n X (X'X)^{-1} && \text{Factor out the scalar } \sigma^2 \\ &= \sigma^2 (X'X)^{-1} X' X (X'X)^{-1} && I_n \text{ is the identity, so } X' I_n X = X'X \\ &= \sigma^2 (X'X)^{-1} && \text{Cancel } X'X \text{ with } (X'X)^{-1} \end{aligned} \tag{8.7}$$

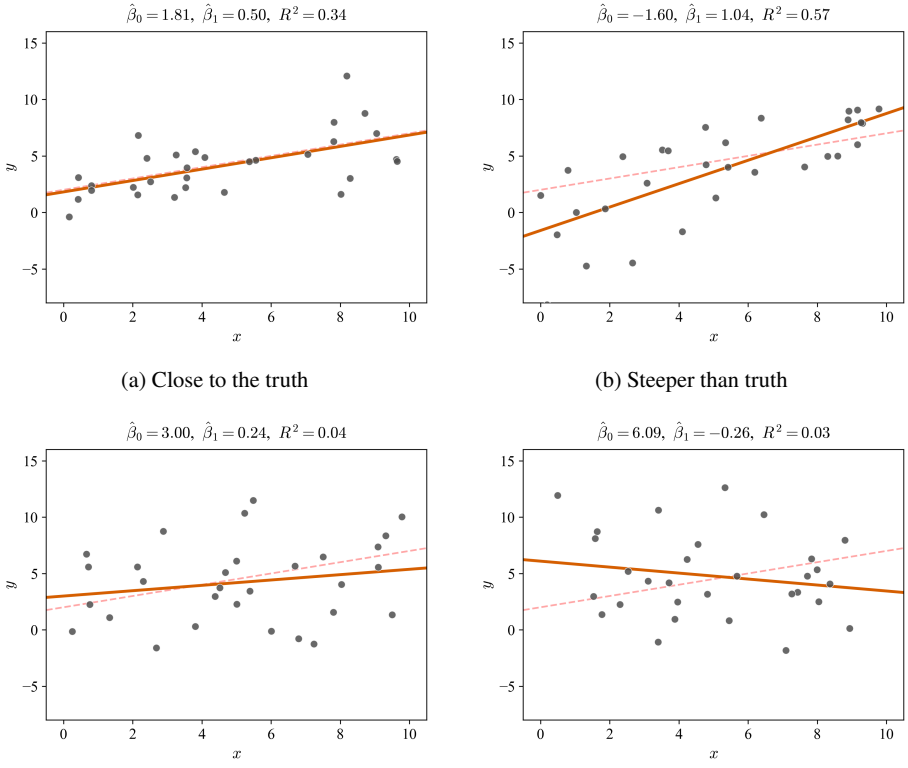
Under homoskedasticity and no correlation, the variance collapses to this clean expression. If the assumption fails, we are left with the full sandwich form instead and

must use a more general calculation (see Chapter 14 for the derivation). But even in the homoskedastic case, we are not finished. The formula

$$\text{Var}(\hat{\boldsymbol{\beta}}) = \sigma^2 (X'X)^{-1}$$

still contains σ^2, an unknown population quantity. We cannot plug in what we do not know. So we need to continue working on this to figure out how to estimate σ^2 from the data. To do this, the next step in our journey will be to chat about a matrix called the **hat matrix**.

Before we move on, though, let's pause and look at what $\text{Var}(\hat{\boldsymbol{\beta}})$ actually means in practice. Figure 8.3 draws four samples from the exact same population. Same true slope, same true intercept, same σ. Yet every sample gives a wildly different OLS line. One sample (panel a) lands almost on top of the truth. Another (panel b) is twice as steep. Panel c finds a slope that is barely positive, and panel d gets the sign completely wrong. All four came from the *same* data generating process. That is sampling variation, and $\text{Var}(\hat{\boldsymbol{\beta}}) = \sigma^2 (X'X)^{-1}$ is the formula that tells us how much we should expect those lines to wiggle from sample to sample.



(a) Close to the truth

(b) Steeper than truth

(c) Shallow positive

(d) Wrong sign!

Fig. 8.3: Four independent random samples drawn from the *same* data generating process ($y = 2 + 0.5\,x + u$, $\sigma = 3$, $n = 30$). The faint dashed line is the true population regression line; the solid line is the OLS fit for that particular sample. Every sample gives a different intercept, slope, and standard error. This is sampling variation in action.

8.3 Introduction to the Hat Matrix

The fitted values are obtained by replacing the unknown coefficients $\boldsymbol{\beta}$ with their estimates. Starting from $\hat{\mathbf{y}} = X\hat{\boldsymbol{\beta}}$ and substituting the OLS estimator $\hat{\boldsymbol{\beta}} = (X'X)^{-1}X'\mathbf{y}$ gives

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

That big matrix sitting in front of $\mathbf{y}$ is so important that it gets its own name. We call it the hat matrix:

$$H = X(X'X)^{-1}X'. \tag{8.8}$$

Why "hat matrix"? Because H is the matrix that "puts a hat on $\mathbf{y}$" (it turns $\mathbf{y}$ into $\hat{\mathbf{y}}$). In other words, it takes the observed outcomes and produces the fitted values:

$$\hat{\mathbf{y}} = H\mathbf{y} \tag{8.9}$$

The residual vector $\hat{\mathbf{u}} = \mathbf{y} - \hat{\mathbf{y}} = \mathbf{y} - H\mathbf{y}$ represents the portion of $\mathbf{y}$ not explained by the model: the component that remains after differencing the real values from the fitted ones. We will examine the hat matrix more carefully in the next section and return to it again in Chapter 9.1, where it will help us measure observation influence and diagnose multivariate outliers.

8.3.1 The Hat Matrix as an Orthogonal Projection

So far, we have derived the Hat Matrix and seen what it does: when we multiply H by $\mathbf{y}$, we obtain the fitted values $\hat{\mathbf{y}}$. We also know that its individual elements will later help us study influence and leverage. But What IS the hat matrix and why does it work?

H is an **orthogonal projection**. If that term feels unfamiliar, do not worry…we will go on the journey of discovering what it means together below. The easiest way to see what is happening is to work through a small example. Oh, and be ready to put on your geometry hats.[10] [11]

Suppose we observe three data points and fit the model $y_i = \beta_0 + \beta_1 x_i + u_i$:

$$\mathbf{y} = \begin{pmatrix} 4 \\ -2 \\ 7 \end{pmatrix}, \qquad X = \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{pmatrix}.$$

[10] No, not $\widehat{geometry}$…

[11] Let's answer a question your future self may be forming: "We already derived OLS and got the right formula. Why are we doing it again with geometry?" Because geometry explains *why* the formula works.

The first column of X is the constant vector $\mathbf{1} = (1, 1, 1)'$ representing the intercept, and the second column is the regressor $\mathbf{x} = (0, 1, 2)'$. Because we have three observations, each column of X is a vector in $\mathbb{R}^3$.

This is key to understanding working this through with geometry: the axes of our space do not represent variables like x and y. Instead, they represent *observations*. The first coordinate corresponds to observation 1, the second to observation 2, and the third to observation 3.

Since $n = 3$, every vector in this problem lives in $\mathbb{R}^3$. That means we can visualize the regression as taking place inside a three-dimensional space whose axes correspond to observations 1, 2, and 3.[12]

So what does it actually look like to place a column of numbers into this space?

Columns of X As Vectors

Start with something familiar: a number line. Think about the number +4 sitting on that line. Yes, +4 is a location (a point on the line), but it also tells you something richer: it says "start at zero and go 4 units to the right." The sign tells you which direction to go, and the size tells you how far. You could draw this as a little arrow from 0 to 4. Now think about −2. That says "start at zero and go 2 units to the left." Again, a direction and a distance from zero. Every number on the number line is secretly an instruction: from zero, go this far, in this direction.

Now do the same thing in two dimensions. The point $(4, -2)$ on a flat graph says "go 4 to the right along the first axis, then 2 down along the second axis." You could just plot a dot where you end up. But you could also draw the path from the origin $(0, 0)$ to the destination $(4, -2)$ as an arrow. That arrow carries more information than the dot: it shows not just *where you are*, but *how you got there from zero*, both the distance and the direction.

Three dimensions is no different. Take our intercept column $\mathbf{1} = (1, 1, 1)'$. That says "from the origin, go 1 unit along observation axis 1, then 1 unit along observation axis 2, then 1 unit along observation axis 3." We can draw this as an arrow from $(0, 0, 0)$ to $(1, 1, 1)$. The arrow points in a specific direction and has a specific length.

In mathematics, an object that carries both a **magnitude** (how far) and a **direction** (which way) is called a **vector**. That is what the arrow represents. The arrow is not some extra decoration we are pasting onto the numbers. It is what the numbers *look like* once

[12] I try not to repeat myself, but…just this time. We have translated our three variables (our outcome, the **1** vector which corresponds to our intercept, and an independent variable) into three points: (4, -2, 7), (1,1,1), and (0,1,2) in a 3D graph. It is a 3D graph, not because we have three variables, but because we have three observations. Each observation has its own axis.

you recognize that each coordinate is an instruction: how far to move from zero along that axis.[13]

The same logic applies to every column of data we have. The regressor column $\mathbf{x} = (0, 1, 2)'$ is also a vector: an arrow from the origin to the point $(0, 1, 2)$. It says "observation 1 is at zero, observation 2 is 1 unit out, observation 3 is 2 units out." A different direction from $\mathbf{1}$, because the pattern of values across observations is different.

The origin, $(0, 0, 0)$, is not just a reference point, it is the outcome where every observation is zero. Every vector is a departure from that silence: a specific direction and distance away from nothing. The arrow to $\mathbf{1}$ says "every observation is equally above zero." The arrow to $\mathbf{x}$ says "the observations fan out: 0, 1, 2."

The regression has two knobs, β_0 and β_1. Each one controls how far we travel along one of these arrows. To explore this, let us start by writing the fitted values observation by observation:

$$\begin{aligned}\hat{y}_1 &= \beta_0 + \beta_1(0),\\ \hat{y}_2 &= \beta_0 + \beta_1(1),\\ \hat{y}_3 &= \beta_0 + \beta_1(2).\end{aligned}$$

Stack them into a single vector:

$$\hat{\mathbf{y}} = \beta_0 \begin{pmatrix}1\\1\\1\end{pmatrix} + \beta_1 \begin{pmatrix}0\\1\\2\end{pmatrix} = \begin{pmatrix}\beta_0\\ \beta_0 + \beta_1\\ \beta_0 + 2\beta_1\end{pmatrix}$$

Look at what happened. The fitted value vector is β_0 copies of $\mathbf{1}$ plus β_1 copies of $\mathbf{x}$. Each coefficient is a dial that controls how far along its column's direction we travel. Crank β_0 up and you move farther along the $\mathbf{1}$ arrow. Crank β_1 up and you move farther along the $\mathbf{x}$ arrow. *The columns supply the directions; the coefficients supply the distances.* Figure 8.4 shows these two directions in $\mathbb{R}^3$.

The Column Space

So we have two directions, and two dials. But in reality, β_0 and β_1 can both vary at the same time. We are not limited to sliding along just one direction or the other. We can move some amount in the $\mathbf{1}$ direction and some amount in the $\mathbf{x}$ direction simultaneously.

If we allow both coefficients to vary freely, the fitted vector

[13] If you have taken a physics class, you have already encountered vectors: velocity is a vector (speed and direction), force is a vector (strength and direction). In our setting the "direction" is not north or east. It is a direction in the abstract space of observations. But the idea is the same: magnitude plus direction.

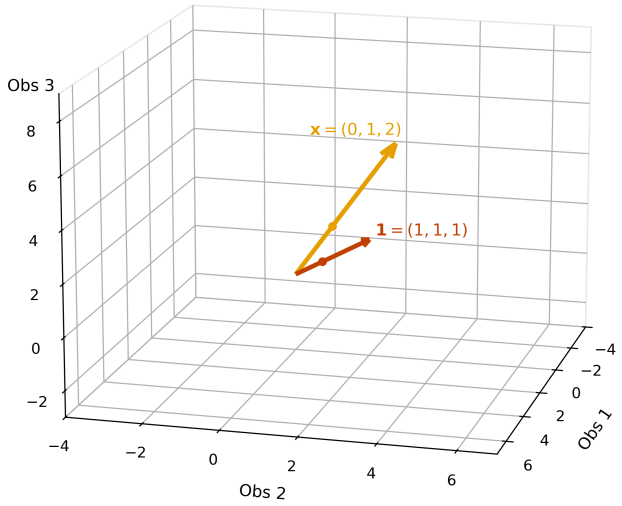


Fig. 8.4: The two column vectors of X in $\mathbb{R}^3$: the constant column $\mathbf{1} = (1, 1, 1)'$ in dark red and the regressor column $\mathbf{x} = (0, 1, 2)'$ in orange. Each arrow starts at the origin and points in the direction of its column. The arrow length is for visibility only; the coefficient controls how far you actually travel along that direction.

$$\hat{\mathbf{y}} = \beta_0 \mathbf{1} + \beta_1 \mathbf{x}$$

can land anywhere on the flat surface generated by those two axes.

That flat surface is a plane. We call that plane the **column space**, written $\text{Col}(X)$. We can see our vectors and the column space in Figure 8.5.

Every point on the teal surface is a linear combination of the two columns. The arrows show you the two directions; the plane shows you the result of mixing them. Notice that the plane extends past the arrowheads and back through the origin in every direction. That is important: the coefficients are not forced to be positive. A negative β_0 moves you in the *opposite* direction of the dark red arrow: instead of heading toward $(1, 1, 1)$, you slide toward $(-1, -1, -1)$. Same for β_1. The plane captures all of this. No matter how you set the two knobs, positive or negative, large or small, you stay on this flat surface. Adding a third regressor would expand the menu to a three dimensional subspace, but the idea is the same: the column space is everything the model can produce.

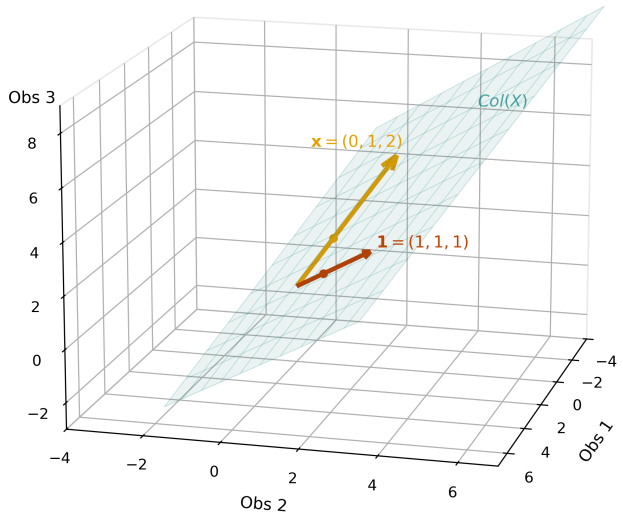


Fig. 8.5: The column space of X, shown as a teal plane in $\mathbb{R}^3$. The dark red arrow is the constant column $\mathbf{1} = (1, 1, 1)'$; the orange arrow is the regressor column $\mathbf{x} = (0, 1, 2)'$. Every point on the plane is some linear combination $\beta_0\mathbf{1} + \beta_1\mathbf{x}$. This is the "menu" of all possible fitted value vectors.

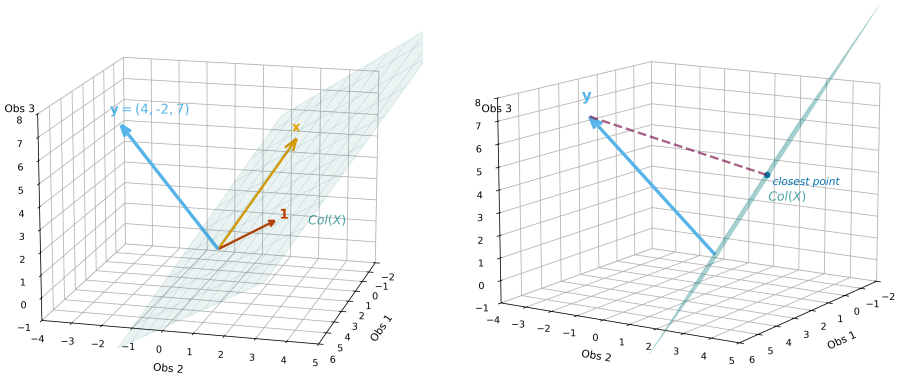
Where Is y?

We have our column space: a plane in $\mathbb{R}^3$ containing every fitted value vector the model can produce. What about $\mathbf{y}$? Let's plot $\mathbf{y} = (4, -2, 7)'$ in the same space as the column space.

Just like the columns of X, the outcome $\mathbf{y}$ is a vector: an arrow from the origin to the point $(4, -2, 7)$. It says "from silence, observation 1 goes to 4, observation 2 drops to -2, observation 3 jumps to 7." That is the data's departure from zero, its direction and its distance. Does $\mathbf{y}$ land on the plane? Think about what that would mean. If $\mathbf{y}$ were on the plane, there would exist some β_0 and β_1 such that $\beta_0\mathbf{1} + \beta_1\mathbf{x} = \mathbf{y}$ exactly. Written out observation by observation, we would need

$$\begin{aligned} \beta_0 &= 4, \\ \beta_0 + \beta_1 &= -2, \\ \beta_0 + 2\beta_1 &= 7. \end{aligned}$$

The first equation tells us $\beta_0 = 4$. Plugging into the second, $4+\beta_1 = -2$, so $\beta_1 = -6$. But then the third equation would need $4 + 2(-6) = -8$ to equal 7. It clearly does not, so the vector floats off the plane.



(a) The outcome vector $\mathbf{y} = (4, -2, 7)'$ does not lie on $\text{Col}(X)$. No choice of β_0 and β_1 can reproduce $\mathbf{y}$ exactly.

(b) The same scene rotated so the plane appears nearly edge on.

Fig. 8.6: $\mathbf{y}$ floats off the column space: no linear combination of the columns can reach $\mathbf{y}$.

Figure 8.6a and Figure 8.6 show that $\mathbf{y}$ is not on the surface. This is not really surprising. We have two knobs (β_0 and β_1) but three observations to match. Our plane is a quite thin slice of $\mathbb{R}^3$. The chance that $\mathbf{y}$ just happens to land exactly on that slice is zero unless the model is perfect with no error at all.

And this generalizes: whenever $n > k$, the column space is a k dimensional subspace sitting inside $\mathbb{R}^n$. It is a tiny flat sliver of the full space. Essentially every $\mathbf{y}$ we will ever encounter in practice will float off the plane.

The Shadow on the Wall

Now, imagine we shine a flashlight at $\mathbf{y}$, angled so that the beam hits the plane straight on. The shadow that $\mathbf{y}$ casts onto the plane is the closest point on the surface to $\mathbf{y}$ itself. That shadow is $\hat{\mathbf{y}}$.

Think about why the angle of the light matters. If we tilted the flashlight, the shadow would slide across the plane and land somewhere else, farther from $\mathbf{y}$. The only way to get the shadow as close as possible to the original is to shine the light *straight down*:

perpendicular to the plane. When the light rays hit at right angles, the shadow cannot be improved. It is the nearest point. [14]

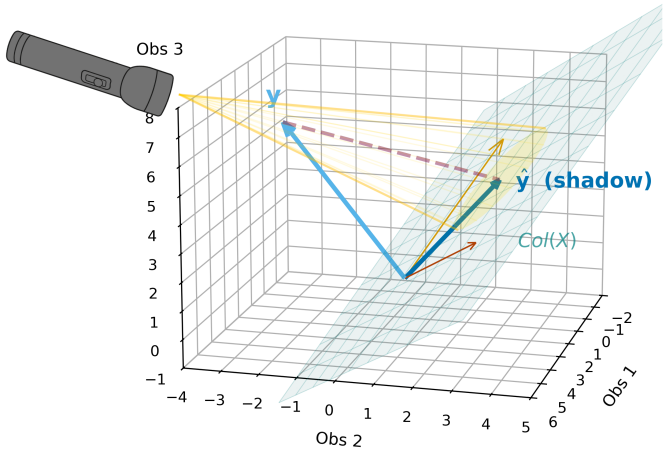


Fig. 8.7: The flashlight shines perpendicular to Col(X). The shadow of $\mathbf{y}$ on the plane is the fitted value vector $\hat{\mathbf{y}}$, shown as the blue arrow. The dashed wine gap between $\mathbf{y}$ and its shadow is the residual $\hat{\mathbf{u}}$: the difference between the vector, $\mathbf{y}$, and its projection, $\hat{\mathbf{y}}$.

Figure 8.7 demonstrates this. The yellow cone is the light, $\mathbf{y}$ is the object floating above the plane, and the blue arrow is its shadow. The dashed wine line connecting $\mathbf{y}$ to its shadow is the residual. When the light rays hit the plane at right angles, the shadow is called an **orthogonal projection**. "Orthogonal" just means "at right angles." And the matrix that performs this projection is the hat matrix $H = X(X'X)^{-1}X'$ from Equation 8.8. Multiplying H by $\mathbf{y}$ shines the flashlight and reads off where the shadow lands:

$$\hat{\mathbf{y}} = H\mathbf{y}.$$

Now recall what OLS does. It minimizes the sum of squared residuals. In the picture, those residuals are the dashed wine line connecting $\mathbf{y}$ to the plane. They measure how far $\mathbf{y}$ lies above the plane of possible fitted values. Squaring them turns that distance into a positive number and gives greater weight to larger misses. Minimizing the sum of squared residuals therefore amounts to choosing the point on the plane that is closest to $\mathbf{y}$.

[14] Another interesting thing with this analogy is this: imagine you have a light shining across the room in an otherwise dark room. You are standing in between the light and the wall, thus you have a shadow. The light turned you, a 3D object, into a 2D object to match the plane it is being projected on, the wall.

Computing the Shadow

Let us find out where the shadow lands. We already know the OLS formula: $\hat{\boldsymbol{\beta}} = (X'X)^{-1}X'\mathbf{y}$. Plugging in our numbers gives $\hat{\boldsymbol{\beta}} = (3/2,\ 3/2)'$. So the fitted value vector is

$$\hat{\mathbf{y}} = \frac{3}{2}\begin{pmatrix}1\\1\\1\end{pmatrix} + \frac{3}{2}\begin{pmatrix}0\\1\\2\end{pmatrix} = \begin{pmatrix}3/2\\3\\9/2\end{pmatrix} = \begin{pmatrix}1.5\\3\\4.5\end{pmatrix}.$$

That is a point on the plane, built from $3/2$ parts of the $\mathbf{1}$ direction and $3/2$ parts of the $\mathbf{x}$ direction. Look at the equation above: $\hat{\mathbf{y}}$ is the *sum* of two vectors, $\hat{\beta}_0\mathbf{1}$ and $\hat{\beta}_1\mathbf{x}$. When you add two vectors geometrically, the result is the diagonal of a parallelogram. The coefficients $\hat{\beta}_0 = 3/2$ and $\hat{\beta}_1 = 3/2$ tell us how far to stretch each column before adding them together, and the sum is the point on the plane closest to $\mathbf{y}$.

Figure 8.8 shows this from directly above. The two solid component arrows from the origin are $\hat{\beta}_0\mathbf{1} = \frac{3}{2}(1,1,1)'$ in dark red and $\hat{\beta}_1\mathbf{x} = \frac{3}{2}(0,1,2)'$ in orange. Those are the two vectors being added. From the tip of each, a dashed line shows the other component shifted over: the dashed orange line from $\hat{\beta}_0\mathbf{1}$ to $\hat{\mathbf{y}}$ is a copy of $\hat{\beta}_1\mathbf{x}$, and the dashed dark red line from $\hat{\beta}_1\mathbf{x}$ to $\hat{\mathbf{y}}$ is a copy of $\hat{\beta}_0\mathbf{1}$. The four sides form a parallelogram, and $\hat{\mathbf{y}}$ sits at the far corner where both paths meet.

The Residual Is Perpendicular

Let's calculate our residuals:

$$\hat{\mathbf{u}} = \mathbf{y} - \hat{\mathbf{y}} = \begin{pmatrix}4\\-2\\7\end{pmatrix} - \begin{pmatrix}3/2\\3\\9/2\end{pmatrix} = \begin{pmatrix}5/2\\-5\\5/2\end{pmatrix}.$$

Remember the flashlight? The light rays hit the plane at right angles. We can see this visually in Figures 8.9a and 8.9b.

Suppose the residual vector was not perpendicular. Suppose it tilted slightly to the left. Then you could slide $\hat{\mathbf{y}}$ a little to the left along the plane and get closer to $\mathbf{y}$. As long as the gap has any component along the plane, there is room for improvement. Only when the gap points *straight off* the plane is there nothing left to gain, which is what we achieve through OLS.

Like H, the matrix $I_n - H$ is an orthogonal projection.[15] While H projects $\mathbf{y}$ onto $\text{Col}(X)$, the matrix $I_n - H$ projects onto the perpendicular complement, keeping only the perpendicular leftover.[16] Figure 8.10 shows both projections: $\hat{\mathbf{y}} = H\mathbf{y}$ on the plane and $\hat{\mathbf{u}} = (I_n - H)\mathbf{y}$ perpendicular to it.

[15] More on this matrix soon!

[16] The perpendicular complement of $\text{Col}(X)$ is written $\text{Col}(X)^{\perp}$.

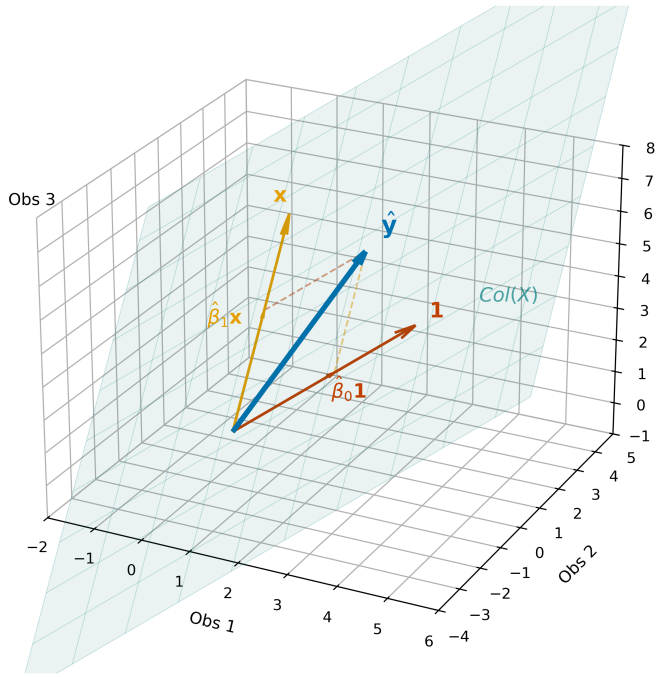


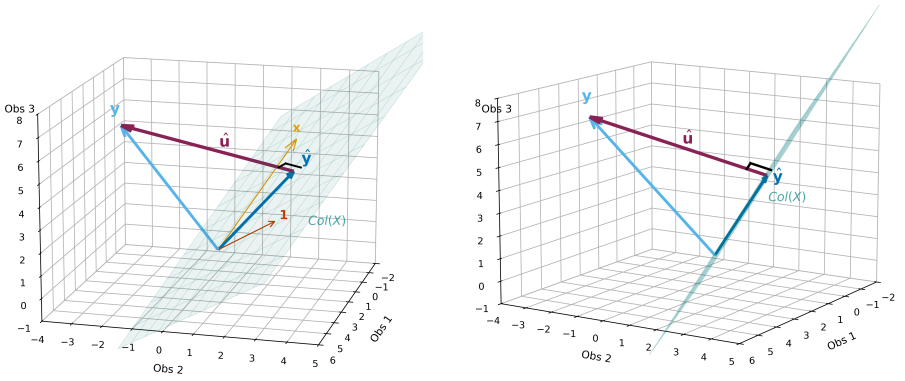
Fig. 8.8: The projection viewed from directly above. Adding two vectors produces a parallelogram: the solid arrows are $\hat{\beta}_0\mathbf{1}$ (dark red) and $\hat{\beta}_1\mathbf{x}$ (orange); the dashed lines are shifted copies of each component completing the shape. The diagonal is $\hat{\mathbf{y}}$.

Let us verify the perpendicularity with numbers. Two vectors are perpendicular when their dot product is zero. The residual needs to be perpendicular to every direction on the plane, but we do not need to check every direction individually: if $\hat{\mathbf{u}}$ is perpendicular to $\mathbf{1}$ and perpendicular to $\mathbf{x}$, it is perpendicular to every combination of them, and therefore to the entire plane.[18] So we check both:

$$X'\hat{\mathbf{u}} = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 2 \end{pmatrix} \begin{pmatrix} 5/2 \\ -5 \\ 5/2 \end{pmatrix} = \begin{pmatrix} 5/2 - 5 + 5/2 \\ 0 - 5 + 5 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

[17] The subspace $\mathrm{Col}(X)^{\perp}$ is a line through the origin perpendicular to the plane. We do not draw it because it invites confusion: $\hat{\mathbf{u}}$ points in the direction of that line, but we draw it starting at $\hat{\mathbf{y}}$ (not at the origin) so that the tip-to-tail addition $\hat{\mathbf{y}} + \hat{\mathbf{u}} = \mathbf{y}$ is visible. If you slid the line over to start at $\hat{\mathbf{y}}$, the residual would lie exactly along it.

[18] Why is this sufficient? Any vector on the plane has the form $c_1\mathbf{1} + c_2\mathbf{x}$. Its dot product with $\hat{\mathbf{u}}$ is $c_1(\mathbf{1}'\hat{\mathbf{u}}) + c_2(\mathbf{x}'\hat{\mathbf{u}})$. If both of those individual dot products are zero, the whole thing is zero regardless of c_1 and c_2.



(a) The full decomposition $\mathbf{y} = \hat{\mathbf{y}} + \hat{\mathbf{u}}$. The right angle marker at $\hat{\mathbf{y}}$ indicates that $\hat{\mathbf{u}}$ is perpendicular to the column space.

(b) The decomposition viewed edge on. The column space collapses to a line, and the residual rises straight off it at a right angle.

Fig. 8.9: The decomposition $\mathbf{y} = \hat{\mathbf{y}} + \hat{\mathbf{u}}$. The residual $\hat{\mathbf{u}}$ points straight off the column space.

Both dot products are zero. The residual is orthogonal to the constant column and to the regressor column. Since these two columns span the entire plane, the residual is orthogonal to *every* vector on the plane. That includes $\hat{\mathbf{y}}$ itself, $\hat{\mathbf{y}} = \hat{\beta}_0\mathbf{1} + \hat{\beta}_1\mathbf{x}$ is a linear combination of the columns and therefore lives on the plane. So

$$\hat{\mathbf{y}}'\hat{\mathbf{u}} = 0.$$

The fitted values and the residuals are perpendicular. We can verify: $\hat{\mathbf{y}}'\hat{\mathbf{u}} = (3/2)(5/2) + (3)(-5) + (9/2)(5/2) = 15/4 - 15 + 45/4 = 60/4 - 60/4 = 0$.

Why residuals sum to zero (and when they don't). You learned in Chapter 7 that OLS residuals always sum to zero: $\sum_{i=1}^{n} \hat{u}_i = 0$. We proved it algebraically from the first order conditions. Now the geometry tells us *why*.

When the model includes an intercept, the constant vector $\mathbf{1} = (1, 1, \ldots, 1)'$ is a column of X, so $\mathbf{1} \in \text{Col}(X)$. We just showed that $\hat{\mathbf{u}} \in \text{Col}(X)^{\perp}$, meaning the residuals are orthogonal to every vector in $\text{Col}(X)$. In particular, $\hat{\mathbf{u}}$ is orthogonal to $\mathbf{1}$:

$$\mathbf{1}'\hat{\mathbf{u}} = \sum_{i=1}^{n} \hat{u}_i = 0.$$

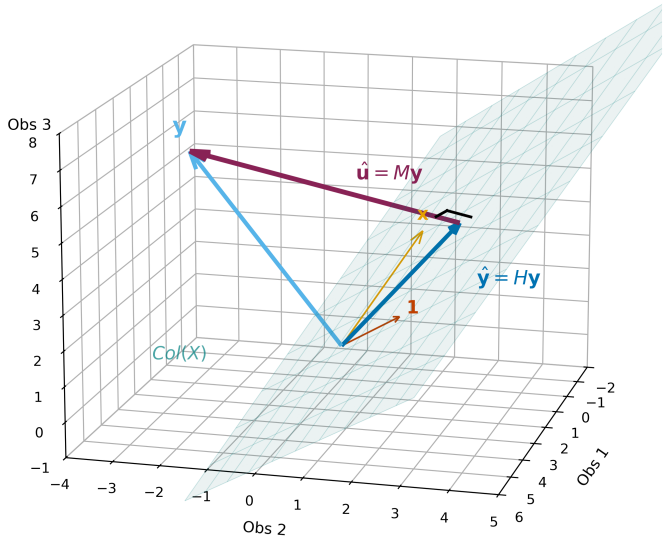


Fig. 8.10: The two complementary projections. H projects $\mathbf{y}$ onto $\text{Col}(X)$ (the teal plane, dimension $k = 2$), producing $\hat{\mathbf{y}}$. The residual maker $I_n - H$ projects $\mathbf{y}$ onto $\text{Col}(X)^\perp$ (dimension $n - k = 1$), producing $\hat{\mathbf{u}}$, the perpendicular gap from $\hat{\mathbf{y}}$ to $\mathbf{y}$. Together, $\hat{\mathbf{y}} + \hat{\mathbf{u}} = \mathbf{y}$: the two projections exhaust all of $\mathbb{R}^n$.[17]

If we drop the intercept, $\mathbf{1}$ is no longer a column of X, so $\mathbf{1}$ may not be in $\text{Col}(X)$, and the orthogonality $\mathbf{1}'\hat{\mathbf{u}} = 0$ is no longer guaranteed.[19]

Properties of H and $I_n - H$

The hat matrix has some properties that will do some heavy lifting in later chapters. The first is that it is Idempotent, meaning $H^2 = H$. Projecting twice is the same as projecting once. This occurs because $\hat{\mathbf{y}}$ already lives on the plane. If you take a point that is already on the plane and project it onto the same plane, you get the same point back. There is nowhere closer to go. So $H\hat{\mathbf{y}} = H(H\mathbf{y}) = H^2\mathbf{y} = H\mathbf{y} = \hat{\mathbf{y}}$. The second projection does nothing. It is like casting a shadow of a shadow: an object lying flat on the ground casts a shadow that is itself.

[19] More precisely, the residuals sum to zero if and only if $\mathbf{1} \in \text{Col}(X)$. The condition is membership in $\text{Col}(X)$, not the literal presence of a column of ones. If some linear combination of the columns of X equals $\mathbf{1}$, that is enough. We will see an example of this in Chapter 16.

The second is that it is symmetric: $H' = H$. The hat matrix treats any two vectors in $\mathbb{R}^n$ "the same way from both sides." Concretely, symmetry means that for any vectors $\mathbf{a}$ and $\mathbf{b}$,

$$\mathbf{a}'H\mathbf{b} = \mathbf{b}'H\mathbf{a}.$$

The projection of $\mathbf{b}$ onto the column space, as seen from $\mathbf{a}$'s direction, equals the projection of $\mathbf{a}$ onto the column space, as seen from $\mathbf{b}$'s direction. There is no asymmetry: the flashlight does not favor one side of the room over the other. Algebraically, this falls out of the formula. Recall $H = X(X'X)^{-1}X'$. Transposing:

$$\begin{aligned}
H' &= \left[X(X'X)^{-1}X'\right]' && \text{Take the transpose} \\
&= X\left[(X'X)^{-1}\right]'X' && \text{Reverse the order: } (ABC)' = C'B'A' \\
&= X\,(X'X)^{-1}\,X' && (X'X)^{-1} \text{ is symmetric, so its transpose is itself}^{20} \\
&= H.
\end{aligned}$$

Finally, we have one more useful fact. The trace of the hat matrix equals the number of columns in X:

$$\operatorname{tr}(H) = k.$$

The trace of a matrix is simply the sum of its diagonal entries. So this statement says that if we add up the diagonal elements of H, we obtain k, the number of estimated parameters, including the intercept.

Let's prove it! Suppose X is $n \times k$, with $k \leq n$, and has full column rank. Then $\operatorname{rank}(X) = k$, so $X'X$ is invertible and $(X'X)^{-1}$ exists.

Recall that the hat matrix is

$$H = X(X'X)^{-1}X'.$$

We want to compute its trace:

$$\operatorname{tr}(H) = \operatorname{tr}\big(X(X'X)^{-1}X'\big).$$

At first glance, this looks messy. But recall the cyclic property of the trace from Section 2.5.2: $\operatorname{tr}(ABC) = \operatorname{tr}(CAB)$. We can rotate the order of multiplication inside the trace without changing its value. So we rotate the factors:

$$\operatorname{tr}\big(X(X'X)^{-1}X'\big) = \operatorname{tr}\big(X'X(X'X)^{-1}\big).$$

Now look carefully at what happened. Inside the trace, we now have $X'X$ multiplied by its inverse. But a matrix times its inverse is the identity:

[20] Why is $(X'X)^{-1}$ symmetric? Because $X'X$ is symmetric ($[X'X]' = X'X$), and the inverse of a symmetric matrix is symmetric: if $A = A'$, then $(A^{-1})' = (A')^{-1} = A^{-1}$.

$$X'X(X'X)^{-1} = I_k.$$

So the expression simplifies to

$$\operatorname{tr}(I_k).$$

The trace of the $k \times k$ identity matrix is simply the sum of its diagonal entries. The identity matrix has ones on the diagonal and zeros everywhere else, so its trace is k. Therefore,

$$\operatorname{tr}(H) = k.$$

Now let us establish the corresponding facts for the residual maker matrix, $I_n - H$. The OLS residual vector is $\hat{\mathbf{u}} = \mathbf{y} - H\mathbf{y} = (I_n - H)\mathbf{y}$.[21]

Like H itself, the residual maker is a projection matrix: it is both symmetric and idempotent. To see why, recall that H is symmetric ($H' = H$) and idempotent ($H^2 = H$). Then:

$(I_n - H)' = I_n' - H' = I_n - H$ Symmetric: transpose distributes, $H' = H$

$(I_n - H)^2 = I_n - 2H + H^2 = I_n - 2H + H = I_n - H$ Idempotent: $H^2 = H$, so $-2H + H^2 = -H$

These two facts imply $(I_n - H)'(I_n - H) = I_n - H$, which will drastically simplify quadratic forms in the derivations to follow.

Since $\operatorname{tr}(I_n) = n$ and $\operatorname{tr}(H) = k$, linearity of the trace gives $\operatorname{tr}(I_n - H) = n - k$. Each estimated parameter uses up one direction; the residuals have $n - k$ directions left.

Finally, multiply $H = X(X'X)^{-1}X'$ by X:

$$HX = X \underbrace{(X'X)^{-1}X'X}_{=I_k} = X. \tag{8.10}$$

The inverse eats its original, so H reproduces anything already in $\operatorname{Col}(X)$. Therefore $(I_n - H)X = X - X = \mathbf{0}$: the residual maker annihilates the regressors completely. The same goes for $\hat{\mathbf{y}}$, since $\hat{\mathbf{y}} = X\hat{\boldsymbol{\beta}}$ is just a linear combination of the columns: $(I_n - H)\hat{\mathbf{y}} = \mathbf{0}$. This is why some textbooks call $I_n - H$ by its much cooler name, the **annihilator matrix**: the annihilator wipes out everything the model can explain, and all that survives is the residual.

We will need all of these properties of H and $I_n - H$ to continue in our mission to calculate the standard errors.

[21] Many textbooks define $M \equiv I_n - H$ and write the residual maker as M throughout. We will spell it out as $I_n - H$ to keep its connection to the hat matrix visible.

8.3.2 Estimating σ^2 and Calculating Standard Errors

With the properties of H and $I_n - H$ in hand, we can derive the formula for estimating σ^2 and, from there, the standard errors.

The full chain from residuals to standard errors is built in seven steps. We lay them out one at a time so the logic is clear.

Step 1: Put $\hat{u}$ in terms of $I_n - H$ and u)

$$\begin{aligned}
\hat{\mathbf{u}} &= (I_n - H)\mathbf{y} && \text{Definition of residuals} \\
&= (I_n - H)(X\boldsymbol{\beta} + \mathbf{u}) && \text{Substitute } \mathbf{y} = X\boldsymbol{\beta} + \mathbf{u} \\
&= \underbrace{(I_n - H)X}_{=\mathbf{0}}\boldsymbol{\beta} + (I_n - H)\mathbf{u} && \text{Distribute; from (8.10) the first term vanishes} \\
&= (I_n - H)\mathbf{u} && \text{Signal vanishes, only noise remains}
\end{aligned}$$

Step 2: Sum of squared residuals. Now take the inner product $\hat{\mathbf{u}}'\hat{\mathbf{u}}$ and simplify:

$$\begin{aligned}
\hat{\mathbf{u}}'\hat{\mathbf{u}} &= \mathbf{u}'(I_n - H)'(I_n - H)\mathbf{u} && \text{Substitute } \hat{\mathbf{u}} = (I_n - H)\mathbf{u} \\
&= \mathbf{u}'(I_n - H)^2\mathbf{u} && \text{Symmetric (Section 8.3.1): } (I_n - H)' = I_n - H \\
&= \mathbf{u}'(I_n - H)\mathbf{u} && \text{Idempotent (Section 8.3.1): } (I_n - H)^2 = I_n - H
\end{aligned} \tag{8.11}$$

Step 3: Connecting $\mathbb{E}[\mathbf{u}\mathbf{u}' \mid X]$ to $\sigma^2 I_n$. The next step will produce the expression $\mathbb{E}[\mathbf{u}\mathbf{u}' \mid X]$, and we are going to replace it with $\sigma^2 I_n$. That replacement is not obvious, so let's prove it.

First, what is $\mathbf{u}\mathbf{u}'$? It is the **outer product** of the error vector with itself: an $n \times n$ matrix whose (i, j) entry is $u_i u_j$. Taking its expectation gives us a matrix whose (i, j) entry is $\mathbb{E}[u_i u_j \mid X]$. We need to connect this matrix to the variance-covariance matrix $\mathrm{Var}(\mathbf{u} \mid X)$ that we already know from Assumption 5.

In Chapter 2, we defined the variance-covariance matrix of a random vector $\mathbf{z}$ as:

$$\mathrm{Var}(\mathbf{z}) \;=\; \mathbb{E}\Big[\big(\mathbf{z} - \mathbb{E}[\mathbf{z}]\big)\big(\mathbf{z} - \mathbb{E}[\mathbf{z}]\big)'\Big].$$

This is the matrix version of the scalar formula $\mathrm{Var}(X) = \mathbb{E}[(X - \mathbb{E}[X])^2]$. Let's expand it for our error vector $\mathbf{u}$ (conditioning on X throughout). Write $\boldsymbol{\mu} = \mathbb{E}[\mathbf{u} \mid X]$ to keep things compact:

$$
\begin{aligned}
\mathrm{Var}(\mathbf{u} \mid X) &= \mathbb{E}\Big[(\mathbf{u}-\boldsymbol{\mu})(\mathbf{u}-\boldsymbol{\mu})' \;\Big|\; X\Big] && \text{Definition from Chapter 2}\\
&= \mathbb{E}\Big[\mathbf{u}\mathbf{u}' - \mathbf{u}\boldsymbol{\mu}' - \boldsymbol{\mu}\mathbf{u}' + \boldsymbol{\mu}\boldsymbol{\mu}' \;\Big|\; X\Big] && \text{FOIL the product}\\
&= \mathbb{E}[\mathbf{u}\mathbf{u}' \mid X] - \boldsymbol{\mu}\boldsymbol{\mu}' - \boldsymbol{\mu}\boldsymbol{\mu}' + \boldsymbol{\mu}\boldsymbol{\mu}' && \text{Take } \mathbb{E} \text{ of each term; } \boldsymbol{\mu} \text{ is constant given } X\\
&= \mathbb{E}[\mathbf{u}\mathbf{u}' \mid X] \;-\; \boldsymbol{\mu}\boldsymbol{\mu}' && \text{Last two terms cancel}
\end{aligned}
$$

Rearranging:

$$\mathbb{E}[\mathbf{u}\mathbf{u}' \mid X] \;=\; \mathrm{Var}(\mathbf{u} \mid X) \;+\; \boldsymbol{\mu}\boldsymbol{\mu}'.$$

The zero conditional mean assumption (Assumption 3) says $\boldsymbol{\mu} = \mathbb{E}[\mathbf{u} \mid X] = \mathbf{0}$. So $\boldsymbol{\mu}\boldsymbol{\mu}' = \mathbf{0} \cdot \mathbf{0}' = \mathbf{0}$, and we are left with:

$$\mathbb{E}[\mathbf{u}\mathbf{u}' \mid X] \;=\; \mathrm{Var}(\mathbf{u} \mid X).$$

When the mean is zero, the average of the square *is* the variance. Finally, homoskedasticity (Assumption 5) tells us $\mathrm{Var}(\mathbf{u} \mid X) = \sigma^2 I_n$, so:

$$\mathbb{E}[\mathbf{u}\mathbf{u}' \mid X] \;=\; \sigma^2 I_n.$$

Step 4: Expected value. Taking the conditional expectation of both sides of Equation (8.11):[22]

$$
\begin{aligned}
\mathbb{E}[\hat{\mathbf{u}}'\hat{\mathbf{u}} \mid X] &= \mathbb{E}[\mathbf{u}'(I_n - H)\mathbf{u} \mid X] && \text{Equation (8.11)}\\
&= \mathbb{E}[\mathrm{tr}(\mathbf{u}'(I_n - H)\mathbf{u}) \mid X] && \text{A scalar equals its trace}^{23}\\
&= \mathbb{E}[\mathrm{tr}((I_n - H)\mathbf{u}\mathbf{u}') \mid X] && \text{Cyclic property of trace}\\
&= \mathrm{tr}(\mathbb{E}[(I_n - H)\mathbf{u}\mathbf{u}' \mid X]) && \mathbb{E}[\mathrm{tr}(A)] = \mathrm{tr}(\mathbb{E}[A])\\
&= \mathrm{tr}((I_n - H)\,\mathbb{E}[\mathbf{u}\mathbf{u}' \mid X]) && (I_n - H) \text{ is constant given } X\\
&= \mathrm{tr}\Big((I_n - H)\sigma^2 I_n\Big) && \text{From Step 3, } \mathrm{Var}(\mathbf{u} \mid X) = \sigma^2 I_n\\
&= \sigma^2\,\mathrm{tr}(I_n - H) && \text{Pull out the scalar } \sigma^2\\
&= \sigma^2(n - \mathrm{tr}(H)) && \mathrm{tr}(I_n) = n\\
&= (n-k)\sigma^2 && \mathrm{tr}(H) = k \text{ (Section 8.3.1)}
\end{aligned}
$$

[22] The hat matrix H is built entirely from X, so once we know X we know H. Conditioning on X lets us treat H as a known constant, and the only randomness left comes from the errors $\mathbf{u}$.

[23] Any scalar equals its trace. Using the cyclic property of the trace (Section 2.5.2), $\mathrm{tr}(ABC) = \mathrm{tr}(BCA)$, we obtain

Step 5: The variance estimator. Rearranging gives us an unbiased estimator of σ^2:[24]

$$\hat{\sigma}^2 = \frac{\hat{\mathbf{u}}'\hat{\mathbf{u}}}{n-k} \qquad \mathbb{E}[\hat{\mathbf{u}}'\hat{\mathbf{u}} \mid X] = (n-k)\sigma^2;\ \text{divide both sides by } (n-k) \tag{8.12}$$

The denominator $n-k$ is the **degrees of freedom** of the regression (see Section 3.6 for a more complete explanation). We start with n observations, but estimating k coefficients forces k linear restrictions on the data. After fitting the model, only $n-k$ independent pieces of variation remain in the residuals. Those remaining pieces are what we use to estimate σ^2. Notice that we did not choose to divide by $n-k$ simply out of convention. The $n-k$ came of the derivation: Step 4 showed that $\mathbb{E}[\hat{\mathbf{u}}'\hat{\mathbf{u}} \mid X] = (n-k)\sigma^2$, so dividing by $n-k$ is the only choice that makes the estimator unbiased. We started with n observations and used up k of them estimating the coefficients, leaving $n-k$ independent pieces of information in the residuals.

Step 6: Estimated variance of $\hat{\boldsymbol{\beta}}$. Replace the unknown σ^2 with our estimate:

$$\widehat{\mathrm{Var}(\hat{\boldsymbol{\beta}})} = \hat{\sigma}^2(X'X)^{-1} = \frac{\hat{\mathbf{u}}'\hat{\mathbf{u}}}{n-k}\,(X'X)^{-1} \quad \text{Replace unknown } \sigma^2 \text{ with } \hat{\sigma}^2 \text{ in (8.7)}$$

Step 7: Standard errors. Finally, take the square root of each diagonal element:

$$\mathrm{SE}(\hat{\beta}_j) = \sqrt{\hat{\sigma}^2 \cdot \left[(X'X)^{-1}\right]_{jj}} \qquad \text{Square root of } j\text{th diagonal element} \tag{8.13}$$

Notice that nothing in $I_n - H$ requires the vector being projected to be $\mathbf{y}$. If we apply $(I_n - H)$ to *any* $n \times 1$ vector $\mathbf{v}$, the result $(I_n - H)\mathbf{v}$ is the part of $\mathbf{v}$ orthogonal to $\mathrm{Col}(X)$: the residuals from regressing $\mathbf{v}$ on X. We can build a residual maker for any set of regressors. If Z is some subset of our control variables, then $I_n - Z(Z'Z)^{-1}Z'$ removes the influence of Z from whatever we apply it to. Want to "clean" both X and Y of the effect of Z? Just apply the Z residual maker to each. This idea is the engine behind the **Frisch Waugh Lovell theorem** (Chapter 11), which shows that the coefficient on X in a multiple regression equals the slope from a simple regression after removing the influence of Z from both sides. The residual maker turns "controlling for other variables" into a concrete, mechanical operation.

For simple linear regression, the $(1,1)$ element of $(X'X)^{-1}$ is $1/\sum(X_i - \bar{X})^2$, so:

$$\mathrm{SE}(\hat{\beta}_1) = \sqrt{\frac{\hat{\sigma}^2}{\sum_{i=1}^{n}(X_i - \bar{X})^2}} \qquad \text{SLR special case} \tag{8.14}$$

$$\mathbf{x}'A\mathbf{x} = \mathrm{tr}(\mathbf{x}'A\mathbf{x}) = \mathrm{tr}(A\mathbf{x}\mathbf{x}').$$

This identity is frequently used when working with quadratic forms.

[24] The quantity $\hat{\sigma}^2$ is often called the **mean squared error** (MSE) of the regression.

We would clearly like our standard errors to be as small as possible, but what could affect this? Look at formula (8.13) and think about what each piece does:

- **Small residuals** ($\hat{\mathbf{u}}'\hat{\mathbf{u}}$ is small): The model fits well. Little noise means less uncertainty about the slope.
- **Lots of variation in** X ($[(X'X)^{-1}]_{jj}$ is small): The regressors spread out over a wide range, giving more information about the slope.
- **More degrees of freedom** ($n-k$ is large): More observations relative to parameters means more information for estimating σ^2.

In short: **OLS loves spread in** X because it gives more information about the slope, and it **dislikes noisy residuals** because they obscure the signal.

The standard errors in practice

We just derived $\widehat{\text{Var}}(\hat{\boldsymbol{\beta}}) = \hat{\sigma}^2(X'X)^{-1}$ and showed that the standard error of the jth coefficient is $\text{SE}(\hat{\beta}_j) = \sqrt{\hat{\sigma}^2[(X'X)^{-1}]_{jj}}$. Let's see what that formula actually produces. We will take the baseball data, compute the OLS residuals, estimate $\hat{\sigma}^2 = \hat{\mathbf{u}}'\hat{\mathbf{u}}/(n-k)$, form the full variance covariance matrix, and extract the standard errors from its diagonal. Every step follows directly from the derivation above.For the Stata and Python code, see Appendices Appendix G and Appendix H.

Output:

```
beta_hat =
  Constant = -4.9937
  On Base Pct = 19.1979
  Slugging Pct = 8.4121

s^2 = 0.0173

Var(beta_hat) =
  [0.519287 -2.005796 0.279868]
  [-2.005796 11.629468 -4.105017]
  [0.279868 -4.105017 2.509839]

SE =
  Constant = 0.7206
  On Base Pct = 3.4102
  Slugging Pct = 1.5842
```

Look at the magnitudes. The coefficient on On Base Percentage is 19.20 with a standard error of 3.41, so the ratio is about $19.20/3.41 \approx 5.6$. The coefficient on Slugging Percentage is 8.41 with a standard error of 1.58, giving a ratio of about $8.41/1.58 \approx 5.3$. Both coefficients are several times their standard errors, which is a sign that we have measured something real. We are not ready to conclude anything yet (that comes in

Chapter 10), but the intuition is already here: a coefficient that is large relative to its standard error is one we can be confident about, while a coefficient close to its standard error might just be noise.

Let's return to Fisher's 1925 data. We computed $\hat{\boldsymbol{\beta}}$ earlier. Now we compute the standard errors and the t-statistic and compare against the exact numbers Fisher published in *Statistical Methods for Research Workers*, Table 29.For the Stata and Python code, see Appendices Appendix G and Appendix H.

Output:

```
beta_hat =
  Constant = 0.3317
  Trend = 0.2668

s^2 = 30.7354

Var(beta_hat) =
  [4.310022 -0.211968]
  [-0.211968 0.013675]

SE =
  Constant = 2.0761
  Trend = 0.1169
```

8.3.3 The Gauss–Markov Theorem: Why OLS Is "Best"

We have shown that OLS is unbiased: on average, we should get the correct estimate of $\boldsymbol{\beta}$. But unbiasedness alone is not enough to recommend an estimator. There are *many* unbiased estimators. For instance, you could pick any two observations from your sample and draw a line through them. That estimator would be unbiased too. But it would be wildly imprecise, because it throws away most of the data.

So here is the natural next question: among all the unbiased estimators we could come up with, is OLS the most precise one? Does it have the smallest variance?

The answer is yes, as long as we restrict attention to estimators that are **linear** and **unbiased**. This result is known as the **Gauss–Markov theorem**, and it is one of the most important results in econometrics. It is the reason OLS earns the "B" in **BLUE** (**B**est **L**inear **U**nbiased **E**stimator). Let's prove it carefully, showing every step.

What we already know about OLS

Before we start the proof, let's collect the facts we have established so far. The OLS estimator $\hat{\boldsymbol{\beta}} = (X'X)^{-1}X'\mathbf{y}$ has the following properties:

1. It is a **linear** function of the observable random vector $\mathbf{y}$.
2. It is a **random vector** with a sampling distribution.
3. It is **unbiased**: $\mathbb{E}[\hat{\boldsymbol{\beta}} \mid X] = \boldsymbol{\beta}$ (Section 8.2.2).
4. It has a **sampling precision** (covariance matrix) equal to $\sigma^2(X'X)^{-1}$ (from (8.7)).

Our task now is to compare OLS with *all other* competing estimators from the class of linear unbiased estimators.

The competitors: the class of linear estimators

What do we mean by a "linear" estimator? Any estimator that can be written as

$$\tilde{\boldsymbol{\beta}} = A\mathbf{y},$$

where A is some $k \times n$ matrix that may depend on X but not on $\mathbf{y}$. A particular member of this class is defined by specifying the matrix A. The OLS estimator is one such linear estimator: it uses $A = (X'X)^{-1}X'$. But there are infinitely many other choices of A we could try instead. The question is whether any of them have lower variance.

What does "better" mean for a vector?

When we are comparing two unbiased estimators for a single parameter, "better" simply means "lower variance." But here we are estimating an entire vector $\boldsymbol{\beta}$ with k elements. How do we compare two covariance matrices?

Let $\Sigma_{\hat{\beta}}$ denote the covariance matrix of OLS and $\Sigma_{\tilde{\beta}}$ the covariance matrix of any other linear unbiased estimator. We say $\hat{\boldsymbol{\beta}}$ is **better** than $\tilde{\boldsymbol{\beta}}$ if

$$\mathrm{Var}(\mathbf{a}'\hat{\boldsymbol{\beta}}) \le \mathrm{Var}(\mathbf{a}'\tilde{\boldsymbol{\beta}}) \quad \text{for all vectors } \mathbf{a}.$$

Why does this definition work? Any scalar quantity we might care about is a linear combination of the elements of $\boldsymbol{\beta}$. If we are interested in just β_1, we set $\mathbf{a} = (1, 0, \ldots, 0)$. If we are interested in $\beta_1 + \beta_2$, we set $\mathbf{a} = (1, 1, 0, \ldots, 0)$. The condition above says that for *every* such linear combination, the OLS version has variance no larger than the alternative.

Since $\mathrm{Var}(\mathbf{a}'\hat{\boldsymbol{\beta}}) = \mathbf{a}'\Sigma_{\hat{\beta}}\mathbf{a}$ and $\mathrm{Var}(\mathbf{a}'\tilde{\boldsymbol{\beta}}) = \mathbf{a}'\Sigma_{\tilde{\beta}}\mathbf{a}$, the condition becomes:

$$\mathbf{a}'\left(\Sigma_{\tilde{\beta}} - \Sigma_{\hat{\beta}}\right)\mathbf{a} \ge 0 \quad \text{for all } \mathbf{a}. \tag{8.15}$$

This is describing **positive semidefiniteness** (Chapter 2, Section 2.10). So our goal reduces to showing that the matrix $\Sigma_{\tilde{\beta}} - \Sigma_{\hat{\beta}}$ is positive semidefinite.

What does unbiasedness require?

For $\tilde{\boldsymbol{\beta}} = A\mathbf{y}$ to be unbiased, we need $\mathbb{E}[\tilde{\boldsymbol{\beta}} \mid X] = \boldsymbol{\beta}$ no matter what the true $\boldsymbol{\beta}$ is. Let's substitute $\mathbf{y} = X\boldsymbol{\beta} + \mathbf{u}$ and work through it step by step:

$$\begin{aligned}
\tilde{\boldsymbol{\beta}} &= A\mathbf{y} && \text{Definition of the linear estimator} \\
&= A(X\boldsymbol{\beta} + \mathbf{u}) && \text{Substitute the model } \mathbf{y} = X\boldsymbol{\beta} + \mathbf{u} \\
&= AX\boldsymbol{\beta} + A\mathbf{u} && \text{Distribute } A
\end{aligned}$$

Now take the conditional expectation:

$$\begin{aligned}
\mathbb{E}[\tilde{\boldsymbol{\beta}} \mid X] &= \mathbb{E}[AX\boldsymbol{\beta} + A\mathbf{u} \mid X] && \text{Substitute from above} \\
&= AX\boldsymbol{\beta} + A\,\mathbb{E}[\mathbf{u} \mid X] && AX\boldsymbol{\beta} \text{ is non-random given } X;\ \text{factor } A \text{ out} \\
&= AX\boldsymbol{\beta} + A \cdot \mathbf{0} && \text{Exogeneity (Assumption 3): } \mathbb{E}[\mathbf{u} \mid X] = \mathbf{0} \\
&= AX\boldsymbol{\beta}
\end{aligned}$$

For this to equal $\boldsymbol{\beta}$ regardless of what $\boldsymbol{\beta}$ is, we need

$$AX = I_k. \tag{8.16}$$

Any matrix A satisfying $AX = I_k$ gives an unbiased linear estimator of $\boldsymbol{\beta}$. The OLS matrix $(X'X)^{-1}X'$ satisfies this (check: $(X'X)^{-1}X' \cdot X = I_k$), but so do many others.

The key trick: decompose the alternative

Since every qualifying A must satisfy $AX = I_k$, and the OLS matrix $(X'X)^{-1}X'$ already satisfies this, we can always write

$$A = (X'X)^{-1}X' + D, \tag{8.17}$$

where $D = A - (X'X)^{-1}X'$ is the difference between our alternative matrix and the OLS matrix. If $D = \mathbf{0}$, we are doing OLS. If $D \neq \mathbf{0}$, we are doing something else.

With this decomposition, the alternative estimator becomes:

$$
\begin{aligned}
\tilde{\boldsymbol{\beta}} &= A\mathbf{y} && \text{Definition} \\
&= \left[(X'X)^{-1}X' + D\right]\mathbf{y} && \text{Substitute } A = (X'X)^{-1}X' + D \\
&= \left[(X'X)^{-1}X' + D\right](X\boldsymbol{\beta} + \mathbf{u}) && \text{Substitute } \mathbf{y} = X\boldsymbol{\beta} + \mathbf{u} \\
&= (X'X)^{-1}X'X\boldsymbol{\beta} + DX\boldsymbol{\beta} + (X'X)^{-1}X'\mathbf{u} + D\mathbf{u} && \text{FOIL: distribute both terms} \\
&= \boldsymbol{\beta} + DX\boldsymbol{\beta} + (X'X)^{-1}X'\mathbf{u} + D\mathbf{u} && (X'X)^{-1}X'X = I_k
\end{aligned} \tag{8.18}
$$

Now, what does the unbiasedness condition $AX = I_k$ tell us about D? Substituting our decomposition:

$$
\begin{aligned}
AX &= I_k && \text{Unbiasedness requirement (8.16)} \\
\left[(X'X)^{-1}X' + D\right]X &= I_k && \text{Substitute } A = (X'X)^{-1}X' + D \\
\underbrace{(X'X)^{-1}X'X}_{I_k} + DX &= I_k && \text{Distribute; the first term simplifies} \\
I_k + DX &= I_k \\
DX &= \mathbf{0} && \text{Subtract } I_k \text{ from both sides}
\end{aligned} \tag{8.19}
$$

So D must satisfy $DX = \mathbf{0}$. In other words, the rows of D must be orthogonal to every column of X. The deviation from OLS cannot "see" the regressors at all.

With this result, the $DX\boldsymbol{\beta}$ term in (8.18) vanishes, and we have:

$$
\tilde{\boldsymbol{\beta}} = \boldsymbol{\beta} + (X'X)^{-1}X'\mathbf{u} + D\mathbf{u}. \tag{8.20}
$$

Compared to OLS, where $\hat{\boldsymbol{\beta}} = \boldsymbol{\beta} + (X'X)^{-1}X'\mathbf{u}$, the alternative estimator picks up an *extra* noise term $D\mathbf{u}$. This is the source of the additional variance, as we will now show.

The variance of the alternative estimator

From (8.20), the estimation error is:

$$
\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta} = (X'X)^{-1}X'\mathbf{u} + D\mathbf{u}.
$$

The covariance matrix of $\tilde{\boldsymbol{\beta}}$ is $\mathbb{E}\left[(\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta})(\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta})' \mid X\right]$. Let's expand the outer product and then take expectations term by term:

$$
\begin{aligned}
(\tilde{\boldsymbol{\beta}}-\boldsymbol{\beta})(\tilde{\boldsymbol{\beta}}-\boldsymbol{\beta})' &= \left[(X'X)^{-1}X'\mathbf{u}+D\mathbf{u}\right]\left[(X'X)^{-1}X'\mathbf{u}+D\mathbf{u}\right]' && \text{Substitute the estimation error}\\
&= \left[(X'X)^{-1}X'\mathbf{u}+D\mathbf{u}\right]\left[\mathbf{u}'X(X'X)^{-1}+\mathbf{u}'D'\right] && \text{Transpose: } (AB)'=B'A'
\end{aligned}
$$

Now FOIL (four terms):

$$
\begin{aligned}
&= (X'X)^{-1}X'\mathbf{u}\mathbf{u}'X(X'X)^{-1} && \text{Term 1: OLS } \times \text{ OLS}\\
&\quad + (X'X)^{-1}X'\mathbf{u}\mathbf{u}'D' && \text{Term 2: OLS } \times D\\
&\quad + D\mathbf{u}\mathbf{u}'X(X'X)^{-1} && \text{Term 3: } D \times \text{ OLS}\\
&\quad + D\mathbf{u}\mathbf{u}'D' && \text{Term 4: } D \times D
\end{aligned}
\tag{8.21}
$$

Now take the conditional expectation. The key ingredient is $\mathbb{E}[\mathbf{u}\mathbf{u}' \mid X] = \sigma^2 I_n$ (homoskedasticity, Assumption 5). The matrices X, D, and $(X'X)^{-1}$ are all non-random given X, so they pass through the expectation. Applying $\mathbb{E}[\mathbf{u}\mathbf{u}' \mid X] = \sigma^2 I_n$ to each term:

$$
\begin{aligned}
\Sigma_{\tilde{\boldsymbol{\beta}}} &= \mathbb{E}\left[(\tilde{\boldsymbol{\beta}}-\boldsymbol{\beta})(\tilde{\boldsymbol{\beta}}-\boldsymbol{\beta})' \mid X\right]\\
&= (X'X)^{-1}X'\underbrace{\mathbb{E}[\mathbf{u}\mathbf{u}' \mid X]}_{\sigma^2 I_n}X(X'X)^{-1} && \text{Term 1}\\
&\quad + (X'X)^{-1}X'\underbrace{\mathbb{E}[\mathbf{u}\mathbf{u}' \mid X]}_{\sigma^2 I_n}D' && \text{Term 2}\\
&\quad + D\underbrace{\mathbb{E}[\mathbf{u}\mathbf{u}' \mid X]}_{\sigma^2 I_n}X(X'X)^{-1} && \text{Term 3}\\
&\quad + D\underbrace{\mathbb{E}[\mathbf{u}\mathbf{u}' \mid X]}_{\sigma^2 I_n}D' && \text{Term 4}\\
&= \sigma^2\underbrace{(X'X)^{-1}X'X}_{=I_k}(X'X)^{-1} && \text{Term 1: } I_n \text{ disappears; collect}\\
&\quad + \sigma^2(X'X)^{-1}\underbrace{X'D'}_{=\mathbf{0}} && \text{Term 2}\\
&\quad + \sigma^2\underbrace{DX}_{=\mathbf{0}}(X'X)^{-1} && \text{Term 3}\\
&\quad + \sigma^2 DD' && \text{Term 4}
\end{aligned}
\tag{8.22}
$$

Now we simplify each term. For Term 1: $(X'X)^{-1}X'X(X'X)^{-1} = I_k(X'X)^{-1} = (X'X)^{-1}$. For Terms 2 and 3, we use $DX = \mathbf{0}$ from (8.19). Since $DX = \mathbf{0}$, taking the transpose gives $X'D' = \mathbf{0}$ as well:

$$
\begin{aligned}
\Sigma_{\tilde{\beta}} &= \sigma^2(X'X)^{-1} && \text{Term 1: } (X'X)^{-1}X'X(X'X)^{-1} = (X'X)^{-1} \\
&\quad + \sigma^2(X'X)^{-1}\cdot\mathbf{0} && \text{Term 2: } X'D' = (DX)' = \mathbf{0}' = \mathbf{0} \\
&\quad + \sigma^2\cdot\mathbf{0}\cdot(X'X)^{-1} && \text{Term 3: } DX = \mathbf{0} \text{ from (8.19)} \\
&\quad + \sigma^2 DD' && \text{Term 4: nothing simplifies} \\
&= \sigma^2(X'X)^{-1} + \sigma^2 DD' && \text{Cross terms are gone; only Terms 1 and 4 survive}
\end{aligned}
\tag{8.23}
$$

Recognizing that $\sigma^2(X'X)^{-1} = \Sigma_{\hat{\beta}}$ is the OLS covariance matrix (from (8.7)), we can write the result as:

$$
\Sigma_{\tilde{\beta}} - \Sigma_{\hat{\beta}} = \sigma^2 DD' \tag{8.24}
$$

Is this difference positive semidefinite? Yes. For any vector $\mathbf{a}$:

$$
\begin{aligned}
\mathbf{a}'(\sigma^2 DD')\mathbf{a} &= \sigma^2\mathbf{a}'DD'\mathbf{a} && \text{Factor out the scalar } \sigma^2 > 0 \\
&= \sigma^2(D'\mathbf{a})'(D'\mathbf{a}) && \text{Rewrite: } \mathbf{a}'DD'\mathbf{a} = (D'\mathbf{a})'(D'\mathbf{a}) \\
&= \sigma^2\|D'\mathbf{a}\|^2 && \text{This is a squared length} \\
&\geq 0 && \text{A squared length is always } \geq 0
\end{aligned}
$$

A matrix multiplied by its own transpose is always positive semidefinite (see Section 2.10 in Chapter 2). Since $\sigma^2 > 0$, the product $\sigma^2 DD'$ is positive semidefinite.

Therefore, the condition (8.15) is satisfied: for every linear combination $\mathbf{a}'\boldsymbol{\beta}$, the variance under OLS is no larger than under any other linear unbiased estimator. The **only** way to match the OLS variance is to set $D = \mathbf{0}$, which forces $A = (X'X)^{-1}X'$, which *is* OLS.

Implications of the Gauss–Markov Theorem

Let's step back and read the result. The Gauss–Markov theorem says: if you insist on an estimator that is (1) linear in $\mathbf{y}$ and (2) unbiased, then you cannot do better than OLS. It does not matter how creative or clever the alternative is. As long as it plays by those two

rules, its variance will exceed that of OLS by the positive semidefinite amount $\sigma^2 DD'$. Any departure from the OLS formula adds extra noise through the $D\mathbf{u}$ term.

This is why OLS is called **BLUE**: the **B**est **L**inear **U**nbiased **E**stimator.

Let's also note the assumptions that did the heavy lifting:

- **Exogeneity** (Assumption 3): needed to establish that both $\hat{\boldsymbol{\beta}}$ and $\tilde{\boldsymbol{\beta}}$ are unbiased.
- **Homoskedasticity** (Assumption 5): needed so that $\mathbb{E}[\mathbf{u}\mathbf{u}' \mid X] = \sigma^2 I_n$, which is what allowed the I_n to disappear and the cross terms to collapse cleanly.

If either assumption fails, the theorem does not hold.

It does **not** say OLS is the best estimator. There may exist *nonlinear* or *biased* estimators with lower mean squared error. For instance, ridge regression deliberately introduces a small bias in exchange for a potentially large reduction in variance. In some settings that tradeoff is worth it. The Gauss–Markov theorem is silent about such tradeoffs. It only speaks about the linear unbiased class.

It also requires homoskedasticity (Assumption 5). If the errors do not have constant variance, the $\sigma^2 I_n$ simplification breaks down, and OLS is no longer best even within the linear unbiased class. That is exactly the situation that motivates Generalized Least Squares (Chapter 20).

8.4 A Bit More on $X'X$ and the Intercept

You might compare the SLR and MLR estimators for the β terms and notice that the MLR equivalent of the sample variance and covariance are $X'X$ and $X'\mathbf{y}$, but elementwise they are simply dot products. For example, the (j, ℓ) element of $X'X$ is

$$(X'X)_{j\ell} = \sum_i x_{ij}\, x_{i\ell}.$$

So how do we justify the variance–covariance similarity? This interpretation relies on including a column in the design matrix corresponding to a constant term. We implement this by adding a column of ones (why this works will follow). Let

$$\mathbf{1} = (1, 1, \ldots, 1)'.$$

Write the design matrix as

$$X = [\mathbf{1},\ Z],$$

where Z contains the remaining regressors.

OLS solves

$$\min_{\alpha,\boldsymbol{\beta}} \|\mathbf{y} - \alpha\,\mathbf{1} - Z\boldsymbol{\beta}\|^2,$$

which yields the normal equations

$$X'\hat{\mathbf{u}} = \mathbf{0}, \quad \text{where } \hat{\mathbf{u}} = \mathbf{y} - X\hat{\boldsymbol{\beta}}.$$

Recall from earlier in this chapter that $X'\hat{\mathbf{u}} = \mathbf{0}$: the residuals are orthogonal to every column of X. Since $\mathbf{1}$ is a column of X, we get $\mathbf{1}'\hat{\mathbf{u}} = \sum_i \hat{u}_i = 0$. Including an intercept forces the residuals to have mean zero.

The intercept also absorbs all constant components of the regressors. There can be only one independent constant in the design matrix; any additional constant component would be a linear combination of the intercept column and would violate full rank. As a result, any constant (mean) component of a regressor is perfectly collinear with the intercept and is therefore irrelevant for identifying slope coefficients.

Consequently, slope coefficients are identified only from variation in the regressors that are orthogonal to the intercept, that is, from deviations from their means. Let's investigate this algebraically. Start from the first OLS normal equation:

$$X'\hat{\mathbf{u}} = \mathbf{0},$$

where $\hat{\mathbf{u}} = \mathbf{y} - X\hat{\boldsymbol{\beta}}$. Substituting for $\hat{\mathbf{u}}$ gives

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

which can be rearranged as

$$X'X\hat{\boldsymbol{\beta}} = X'\mathbf{y}.$$

Now include an intercept by writing the design matrix as $X = [\mathbf{1}, Z]$ as above, and partition the coefficient vector as $\hat{\boldsymbol{\beta}} = (\hat{\alpha},\ \hat{\boldsymbol{\beta}}_Z)'$.

Using this partition, the normal equations become

$$\begin{pmatrix} \mathbf{1}'\mathbf{1} & \mathbf{1}'Z \\ Z'\mathbf{1} & Z'Z \end{pmatrix} \begin{pmatrix} \hat{\alpha} \\ \hat{\boldsymbol{\beta}}_Z \end{pmatrix} = \begin{pmatrix} \mathbf{1}'\mathbf{y} \\ Z'\mathbf{y} \end{pmatrix}.$$

With this partition, the normal equations $X'X\hat{\boldsymbol{\beta}} = X'\mathbf{y}$ generate two sets of equations: one associated with the intercept and one associated with the slope coefficients.

The intercept equation is

$$\mathbf{1}'\mathbf{y} = \hat{\alpha}\,\mathbf{1}'\mathbf{1} + \mathbf{1}'Z\,\hat{\boldsymbol{\beta}}_Z.$$

Since $\mathbf{1}'\mathbf{1} = n$, this implies

$$\hat{\alpha} = \bar{y} - \bar{Z}'\hat{\boldsymbol{\beta}}_Z,$$

where $\bar{Z}$ is the vector of column means of Z.

The slope equations are

$$Z'\mathbf{y} = \hat{\alpha}\, Z'\mathbf{1} + Z'Z\,\hat{\boldsymbol{\beta}}_Z.$$

Substituting the expression for $\hat{\alpha}$ into the slope equations yields

$$Z'(\mathbf{y} - \bar{y}\,\mathbf{1}) = (Z'Z - n\bar{Z}\bar{Z}')\,\hat{\boldsymbol{\beta}}_Z.$$

This expression shows explicitly that the slope coefficients are determined by deviations of the regressors and the outcome from their means.

Using the identity

$$Z'Z - n\bar{Z}\bar{Z}' = (Z - \mathbf{1}\bar{Z}')'(Z - \mathbf{1}\bar{Z}'),$$

we obtain

$$\hat{\boldsymbol{\beta}}_Z = \left[(Z - \mathbf{1}\bar{Z}')'(Z - \mathbf{1}\bar{Z}')\right]^{-1}(Z - \mathbf{1}\bar{Z}')'\,(\mathbf{y} - \bar{y}\,\mathbf{1}).$$

Thus, although the raw data are not explicitly centered, the normal equations show that slope coefficients are estimated using demeaned regressors and a demeaned outcome.

Now consider two regressors x_j and x_ℓ. The (j, ℓ) element of $X'X$ is

$$\sum_i x_{ij}\, x_{i\ell},$$

while the numerator of the sample covariance is

$$\sum_i (x_{ij} - \bar{x}_j)(x_{i\ell} - \bar{x}_\ell).$$

These expressions coincide when variables are mean zero. Since including the intercept causes OLS to operate on demeaned regressors for the purpose of estimating slope coefficients, the relevant cross product matrix for slope estimation is exactly the covariance matrix of the regressors, up to a scaling factor.

This also explains the equivalence between the following two procedures:

- Running OLS on centered regressors with no intercept.
- Running OLS on raw regressors with an intercept.

Both procedures yield identical slope estimates because, in the presence of an intercept, constant components of the regressors are absorbed and only deviations from the mean affect the estimation of slopes.

Thus, when an intercept is included, $X'X$ and $X'\mathbf{y}$ can be interpreted as variance–covariance objects for slope estimation, even though elementwise they are simply dot products. The variance–covariance interpretation arises because the intercept forces the normal equations to depend only on demeaned variables.

8.4.1 Summary Table: Classical Linear Model Assumptions

#	Assumption	Purpose
1	Linearity in parameters	Model form
2	Random sampling	Representative sample
3	No perfect multicollinearity	$(X'X)$ invertible
4	Zero conditional mean	Ensures unbiasedness
5	Homoskedasticity	Needed for efficiency (BLUE)

In summary, we have derived the OLS estimator using matrix algebra and established:

- The normal equations $X'\hat{\mathbf{u}} = 0$ lead to $\hat{\boldsymbol{\beta}} = (X'X)^{-1}X'\mathbf{y}$.
- $\hat{\boldsymbol{\beta}}$ is unbiased under $\mathbb{E}[X'\mathbf{u}] = 0$.
- $\text{Var}(\hat{\boldsymbol{\beta}} \mid X) = \sigma^2(X'X)^{-1}$ under homoskedastic errors.
- An unbiased estimator of σ^2 is $\hat{\sigma}^2 = \hat{\mathbf{u}}'\hat{\mathbf{u}}/(n-k)$.
- The standard error of $\hat{\beta}_j$ is $\sqrt{\hat{\sigma}^2\,[(X'X)^{-1}]_{jj}}$.
- OLS is the Best Linear Unbiased Estimator (Gauss–Markov theorem).

These results form the foundation for inference in linear regression. In the next chapters, we delve deeper into the geometry of OLS (the hat matrix and projection in Chapter 9.1), hypothesis testing and confidence intervals (Chapter 10), and the Frisch–Waugh–Lovell theorem for partial regressions (Chapter 11). For the complete derivation chain from model to standard errors in one unbroken sequence, see Appendix Appendix C.