Applied Regression

Francisco Villamil

Applied Quantitative Methods II

MA in Social Sciences, Spring 2026

Welcome back. Today we go deeper into regression – this is the core tool you will use throughout the course and in your final projects. We start with a review, add some formal properties, then move to multiple regression, interactions, and presenting results.

Today's goals

- Review regression as modeling conditional expectations
- Understand OLS properties: assumptions, bias, standard errors
- Understand multiple regression and control variables
- Learn how to model conditional relationships (interactions)
- Present results effectively with `modelsummary`

Give a quick roadmap of the session. Emphasize that we are building on what they learned in AQMSS-I but making it more applied. The new OLS properties section is a bridge between theory and practice – just enough to understand what can go wrong and why robust SEs matter.

Roadmap

What question does regression answer?

- "What is the average value of Y for different values of X?"
- This is the **conditional expectation function** (CEF)
- Written as: $E[Y|X]$
- Regression approximates this function

Start from the intuition: we want to know how the average of one variable changes as we move along another. The CEF is the object of interest. Regression gives us a tractable way to approximate it. Ask students: what does the CEF look like for income given education? It's probably increasing but not linear – regression still gives the best linear approximation.

What does $E[\text{Income} \mid \text{Education}]$ look like?

Is it linear? Why or why not?

Pause for 1–2 minutes. Let students think and discuss briefly with a neighbor. The point is that the CEF for income given education is probably increasing but curved – marginal returns to education may diminish at higher levels, or there may be jumps at degree completion. This motivates why regression is an approximation, not the truth.

The regression model

The most common tool in social science:

$$Y = \beta_0 + \beta_1 X + \varepsilon$$

- Y: outcome we want to explain
- X: explanatory variable(s)
- β: coefficients (what we estimate)
- ε: error term (what we can't explain)

Remind them of the basic setup. The key point is that β captures the systematic relationship and ε captures everything else. This is a model – it's an approximation of reality, not reality itself. The goal of OLS is to find the β values that make the errors as small as possible (in the squared sense).

The regression model in matrix form

With n observations and k variables:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

The OLS estimator:

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

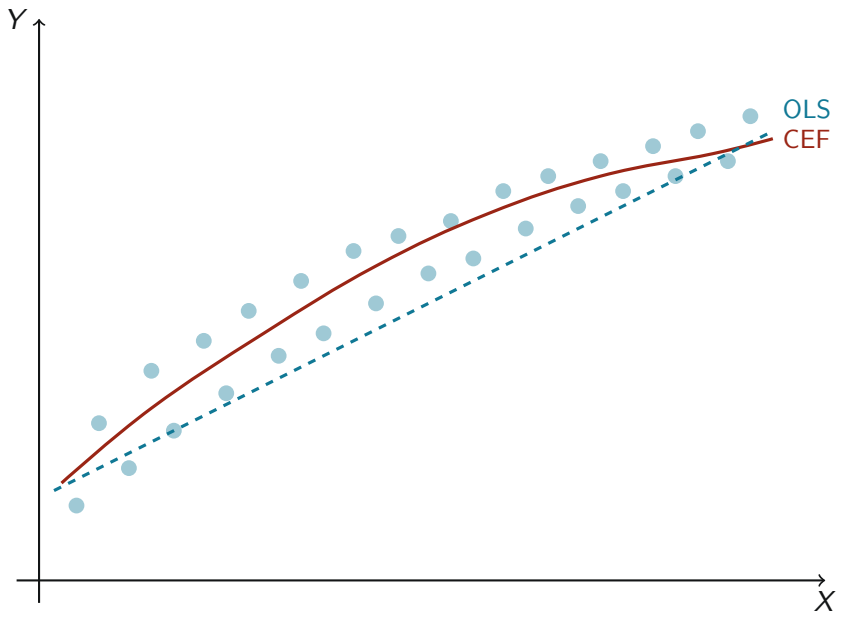
Don't memorize – just know this is what `lm()` computes for you

Keep this light. The point is not to derive the formula but to show that regression has a closed-form solution. Mention that $\mathbf{X}$ is a matrix where each row is an observation and each column is a variable. The formula tells us OLS is just matrix algebra – nothing mysterious. You can mention that this requires $\mathbf{X}'\mathbf{X}$ to be invertible (no perfect multicollinearity). The last line is key: R does this for you.

Linear regression as approximation

- The true CEF might be complicated
- Linear regression fits the **best linear approximation**
- Even if the true relationship is non-linear
- The linear fit is still the best predictor among linear functions
- Why linear? Simple, interpretable, often good enough

This is an important conceptual point. Even if the world is non-linear, the linear regression is still doing something useful: it gives the best linear predictor. Angrist and Pischke call this the "regression as approximation" view. In practice, many relationships are approximately linear over the observed range, and we can always add non-linear terms (polynomials, logs) if needed.



Point to the two lines. The curved red line is the true CEF – the actual average of Y at each value of X. The dashed blue line is the OLS regression line – the best linear approximation. They are close over most of the range, but the CEF curves while OLS cannot. This is why we say regression approximates the CEF. Where would the approximation be worst?

Interpreting the slope coefficient

$$Y = \beta_0 + \beta_1 X + \varepsilon$$

- β_1 represents:
  - → The difference in average Y
  - → Between groups that differ by 1 unit in X
- This is a **comparison**, not necessarily a causal effect

This is the predictive or descriptive interpretation: we are comparing groups. People with one more year of education earn, on average, \$X more. This does not tell us what would happen if we gave someone an extra year of education. That's the causal interpretation, which requires additional assumptions. The coefficient is the same number either way – the interpretation is what changes.

Two framings of β_1

- **Predictive framing**:
  - → "Groups that differ by 1 in X differ by β_1 in Y, on average"
  - → A comparison across units
- **Counterfactual framing**:
  - → "If we changed X by 1, Y would change by β_1"
  - → A statement about what would happen
- Same number, very different claims
- The counterfactual framing requires **causal assumptions**

This distinction is crucial and often confused. When we say "a year of education is worth \$5,000," we can mean two very different things. The predictive framing is always valid as a description of the data. The counterfactual framing is only valid if we can argue that all confounders are accounted for. Ask students: which framing does a randomized experiment give you? Both – because randomization eliminates confounding.

Which framing — predictive or counterfactual —

does a randomized experiment give you?

Pause and let students discuss for a minute. The answer: both! In an experiment, random assignment ensures that the groups being compared differ only in the treatment, so the predictive comparison IS the causal effect. This is the key insight – randomization makes the descriptive and causal interpretations equivalent. No other research design does this automatically.

Descriptive vs. Causal interpretation

- **Descriptive**: How do units with different X values compare?
  - → "People with more education earn more, on average"
- **Causal**: What happens if we change X for a given unit?
  - → "If we give someone more education, they will earn more"
- Same coefficient, very different claims!

Reinforce the previous slide with concrete language. The descriptive statement is about patterns in the world as it is. The causal statement is about what would happen under an intervention. Most applied research wants the causal interpretation, but the regression alone cannot give us that – we need a research design (experiment, natural experiment, etc.) to close the gap.

Running a regression in R

- The basic function: `lm(y ~ x, data = df)`
- Getting tidy output:
  - → `broom::tidy(model)` — coefficients as a data frame
  - → `broom::glance(model)` — model-level statistics (R^2, etc.)
- These are much easier to work with than `summary()`

Quick practical slide. Remind them of the syntax from AQMSS-I. The broom package is essential: tidy() gives a clean data frame of coefficients with standard errors and p-values, glance() gives model-level stats. Show that these outputs can be piped and filtered just like any other data frame. We will use them extensively in the lab later.

Roadmap

OLS assumptions

For OLS to work well, we need:

1. **Linearity**: $Y = \mathbf{X}\beta + \varepsilon$
2. **Random sampling**: observations are i.i.d.
3. **No perfect multicollinearity**: $\mathbf{X}'\mathbf{X}$ is invertible
4. **Zero conditional mean**: $E[\varepsilon|\mathbf{X}] = 0$
5. **Homoskedasticity**: $\text{Var}(\varepsilon|\mathbf{X}) = \sigma^2$

A1–A4 are needed for unbiasedness; A5 for efficient SEs

Go through each assumption briefly:

- Linearity: the model is correctly specified. This is about the functional form, not about the variables themselves being linear (we can include polynomials, logs, etc.).
- Random sampling: each observation is drawn independently. Violated with clustered or panel data.
- No perfect multicollinearity: we can't have one variable that is an exact linear function of others. R will drop the variable automatically if this happens.
- Zero conditional mean: this is the big one. It means the error is unrelated to X. Violated when we have omitted variables. This is what makes OVB a problem.
- Homoskedasticity: the error variance is constant. Almost always violated in practice – this is why we use robust SEs.

~~Emphasize the last line: A1–A4 give unbiasedness, A5 gives correct SEs.~~

OLS is unbiased (under A1–A4)

If assumptions A1–A4 hold:

$$E[\hat{\beta}] = \beta$$

- On average, OLS gives us the right answer
- Any single estimate might be off, but there is no systematic error

- *Think of it like an unbiased dart thrower:*
  centered on the bullseye, but with some scatter

The target/bullseye analogy works well here. An unbiased estimator is like a dart thrower whose throws are centered on the target. Individual throws might miss, but on average they hit the center. Contrast with a biased estimator, where the throws cluster away from the center. The key assumption doing the work here is A4 (zero conditional mean) – if the errors are correlated with X, we get bias. Note: if you have the bullseye image, replace the text reference with the actual image.

Unbiased
centered on target

Biased
systematically off

Point to the two targets. On the left, the darts cluster around the bullseye – this is an unbiased estimator. Each individual estimate may miss, but on average they hit the center. On the right, the darts cluster tightly but away from the center – this is a biased estimator (e.g., when we have omitted variable bias). Ask students: which would you prefer? What if the biased one had less scatter? This connects to the bias-variance tradeoff.

Standard errors and uncertainty

OLS gives us $\hat{\beta}$, but how precise is it?

$$SE(\hat{\beta}_j) = \sqrt{\frac{\hat{\sigma}^2}{SST_j(1 - R_j^2)}}$$

- $\hat{\sigma}^2$: error variance (more noise $\rightarrow$ larger SE)
- SST_j: variation in X_j (more variation $\rightarrow$ smaller SE)
- R_j^2: correlation of X_j with other predictors (multicollinearity $\rightarrow$ larger SE)

- The SE tells us how much $\hat{\beta}$ would vary across samples

Walk through each component:

1. More noise in the data (larger $\hat{\sigma}^2$) makes estimates less precise.
2. More variation in X (larger sample, more spread) gives more information and smaller SEs.
3. When predictors are correlated with each other (R_j^2 is high), it's hard to separate their effects, so SEs blow up.

This is the formula under homoskedasticity (A5). In practice, A5 is usually violated, which is why we need robust SEs. Key takeaway: SEs quantify our uncertainty about the estimate, and we should always report them.

Heteroskedasticity

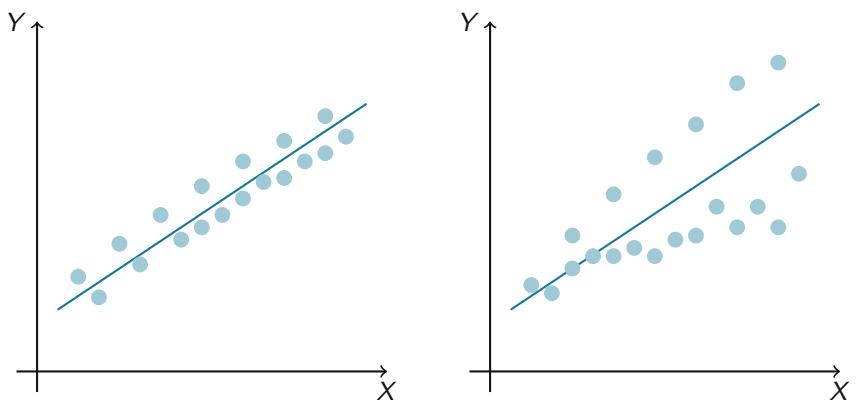
- Assumption A5 says the error variance is constant
- In practice, it almost never is

- **Heteroskedasticity**: $\text{Var}(\varepsilon|X)$ changes with X

- Example: income variation is larger for people with more education

- The estimates $\hat{\beta}$ are still unbiased!
- But the standard errors are wrong

The crucial point is that heteroskedasticity does NOT bias the coefficients – it biases the standard errors. So your estimates are still centered on the right answer, but the confidence intervals are wrong. This means hypothesis tests and p-values are unreliable.



Homoskedastic
constant spread

Heteroskedastic
spread increases with X

Compare the two panels. On the left, the dots have roughly constant spread around the regression line at every value of X – this is homoskedasticity (assumption A5). On the right, the spread fans out as X increases – this is heteroskedasticity. The regression line is the same in both (estimates are unbiased either way), but the standard errors from the classical formula would be wrong for the right panel. This is why we need robust SEs.

Solution: robust standard errors

- Robust ("sandwich") SEs are valid even with heteroskedasticity
- Also called HC (heteroskedasticity-consistent) standard errors
- In R with `modelsummary`:
  → `modelsummary(model, vcov = "robust")`
- Or using `lmtest` and `sandwich`:
  → `coeftest(model, vcov = vcovHC(model))`
- **Practical rule**: always use robust SEs
- There is no real cost when errors are homoskedastic

This is the key practical takeaway from this section. Robust SEs fix the problem of heteroskedasticity without any downside. If the errors happen to be homoskedastic, robust SEs are almost identical to classical SEs. If they are heteroskedastic, robust SEs are correct while classical SEs are wrong. So there is no reason not to use them. Show the modelsummary syntax: vcov = "robust" is all you need. We will practice this in the lab.

Roadmap

Regression Review

OLS Properties

Multiple Regression

Interaction Effects

Presenting Results

Adding predictors

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon$$

- β_1 now represents:
  - → The difference in average Y
  - → Between groups that differ by 1 in X_1
  - → **Holding X_2 constant**
- This is the "controlled" effect of X_1

Transition to the core of applied regression: adding control variables. The key idea is "holding constant" – we compare units that have the same value of X_2 but differ in X_1. This is what makes multiple regression so useful: it lets us approximate an experiment by statistically adjusting for confounders. But it only works if we control for the right things, which is the topic of the next few slides.

How controlling works

- OLS with multiple variables "partials out" the controls
- Technically: we look at variation in X_1 that is unrelated to X_2
- This isolates the unique contribution of X_1

This connects to the Frisch-Waugh-Lovell theorem, though you don't need to name it. The intuition: first regress X_1 on X_2 and get the residuals – these residuals represent the part of X_1 that is "left over" after accounting for X_2. Then regress Y on those residuals. The slope is the same as β_1 in the multiple regression. This is why we say OLS "partials out" the controls.

Omitted variable bias

If we omit a relevant variable X_2, the short regression gives:

$$\tilde{\beta}_1 = \hat{\beta}_1 + \hat{\beta}_2 \cdot \tilde{\delta}$$

- $\hat{\beta}_1$: the "true" coefficient from the long regression
- $\hat{\beta}_2$: the effect of the omitted variable on Y
- $\tilde{\delta}$: the relationship between X_2 and X_1
  $\rightarrow$ (coefficient from regressing X_2 on X_1)
- Bias $= \hat{\beta}_2 \cdot \tilde{\delta}$
- Zero only if $\hat{\beta}_2 = 0$ *or* $\tilde{\delta} = 0$

This is the formal OVB formula. Walk through it carefully:

- The short regression coefficient ($\tilde{\beta}_1$) equals the long regression coefficient ($\hat{\beta}_1$) plus a bias term.
- The bias depends on two things: how much the omitted variable affects Y ($\hat{\beta}_2$) and how related the omitted variable is to X_1 ($\tilde{\delta}$).
- If either is zero, there is no bias.

Give the education-income example: if we omit "ability," and ability both increases education ($\tilde{\delta} > 0$) and increases income ($\hat{\beta}_2 > 0$), the bias is positive – we overestimate the effect of education. We will verify this formula numerically in the lab.

OVB in practice: education and income

	Short regression (omits ability)	Long regression (includes ability)
Education (β_1)	\$5,000	\$3,000
Ability (β_2)	—	\$5,000

- Auxiliary regression: $\tilde{\delta} = 0.4$ (ability on education)
- Check: $\underbrace{\$3{,}000}_{\hat{\beta}_1} + \underbrace{\$5{,}000}_{\hat{\beta}_2} \times \underbrace{0.4}_{\tilde{\delta}} = \underbrace{\$5{,}000}_{\tilde{\beta}_1}$ ✓
- Bias = \$2,000 — we overestimate by 67%!
- Because ability ↑ education *and* ability ↑ income

Walk through this carefully. The short regression (without ability) gives \$5,000 per year of education. The long regression (with ability) gives \$3,000. The difference of \$2,000 is the bias. The formula works: $3{,}000 + 5{,}000 \times 0.4 = 5{,}000$. The bias is positive because ability is positively related to both education and income. This is the "ability bias" in returns to education – one of the most studied problems in labor economics. We will verify this same decomposition with real data in the lab.

What makes a good control?

Good controls are variables that:

- Affect both the treatment and the outcome
- Are determined **before** the treatment
- Are not affected by the treatment

Pre-treatment confounders are the key!

Simple rule of thumb: control for things that came before and could affect both X and Y. Pre-treatment means determined before the treatment occurred – this rules out mediators and post-treatment variables. A confounder is something that opens a "back door" between X and Y. This is the language of DAGs (directed acyclic graphs), which we won't formalize here but is useful to know.

You study the effect of job training on wages.

Is *current job type* a good or bad control?

Why?

Give students a minute to think. Then reveal: it's a bad control – job type is affected by the training (post-treatment), so controlling for it blocks part of the causal pathway. This sets up the next slide on post-treatment variables perfectly. If students say "good control," ask: could training change the type of job someone gets? If yes, then job type is a consequence of the treatment, not a confounder.

Bad controls: Post-treatment variables

- Never control for variables caused by the treatment
- Example: Studying effect of job training on wages
  - → Don't control for job type (affected by training)
  - → Do control for education (determined before training)
- Controlling for post-treatment variables can *introduce* bias

This is one of the most common mistakes in applied work. If training changes the type of job someone gets, and job type affects wages, then controlling for job type blocks part of the causal pathway. You are effectively asking: "what is the effect of training among people who ended up in the same job?" That's a different question, and it underestimates the total effect. The general rule: only control for pre-treatment variables.

Bad controls: Colliders

- A **collider** is caused by both X and Y
- Controlling for it creates a spurious association

- Example: NBA players
  - → Height and skill both affect being in NBA
  - → Among NBA players, height and skill are negatively correlated
  - → But not in the general population!

The collider is a subtler problem. In the general population, height and basketball skill are unrelated (or weakly positive). But if you condition on being in the NBA – which requires either height or skill or both – you create a negative association. Short NBA players must be very skilled; tall ones can get by with less skill. Conditioning on the collider "opens" a path that creates bias. This is sometimes called Berkson's paradox.

Categorical predictors

- What if X is a category (region, party, gender)?

- R automatically creates **dummy variables**
  - → One indicator (0/1) for each category
  - → One category is the **reference** (omitted)

- Coefficients represent the difference from the reference

- Example: `lm(income ~ factor(region), data = df)`
  - → If reference is "North", the "South" coefficient means: average income in South minus average income in North

Practical point about how R handles categorical variables. When you include a factor variable, R creates $k - 1$ dummy variables (where k is the number of categories). The omitted category becomes the reference group, and all coefficients are differences relative to that group. The choice of reference category does not affect the model fit – it just changes what the coefficients mean. You can change the reference with relevel(). Remind students to always use factor() for categorical variables.

Roadmap

Regression Review

OLS Properties

Multiple Regression

Interaction Effects

Presenting Results

When effects depend on context

- Sometimes, the effect of X on Y depends on another variable Z
- Examples:
  - → Effect of education on income may differ by gender
  - → Effect of campaign spending may differ by incumbency status
  - → Effect of democracy on growth may depend on economic development
- We model this with **interaction terms**

Interactions are one of the most powerful and most misunderstood tools in regression. The core idea is simple: the slope changes depending on context. This is very common in social science – effects are rarely the same for everyone. Give a concrete example: the return to education might be higher for women than men, or the effect of campaign spending might matter more for challengers than incumbents.

The interaction model

$$Y = \beta_0 + \beta_1 X + \beta_2 Z + \beta_3 (X \times Z) + \varepsilon$$

- β_1: effect of X when $Z = 0$
- β_2: effect of Z when $X = 0$
- β_3: how the effect of X changes as Z increases

Write the model on the board. Emphasize that β_1 and β_2 are only meaningful at specific values of the other variable. If Z is gender coded 0/1, then β_1 is the effect of X for the reference group (gender $= 0$). If Z is a continuous variable like GDP per capita, then β_1 is the effect of X when GDP is zero – which may not be meaningful. This is why centering variables is sometimes useful.

The marginal effect of X

$$\frac{\partial Y}{\partial X} = \beta_1 + \beta_3 Z$$

- The effect of X is no longer a single number
- It's a **function** of Z
- Need to report effects at meaningful values of Z

This is the key equation. The marginal effect of X depends on Z. So you can't just look at the coefficient table and say "the effect of X is β_1" – that's only true when $Z = 0$. You need to evaluate the marginal effect at different values of Z that are substantively interesting (e.g., the mean, or one SD above and below). This is why plotting is essential for interaction models.

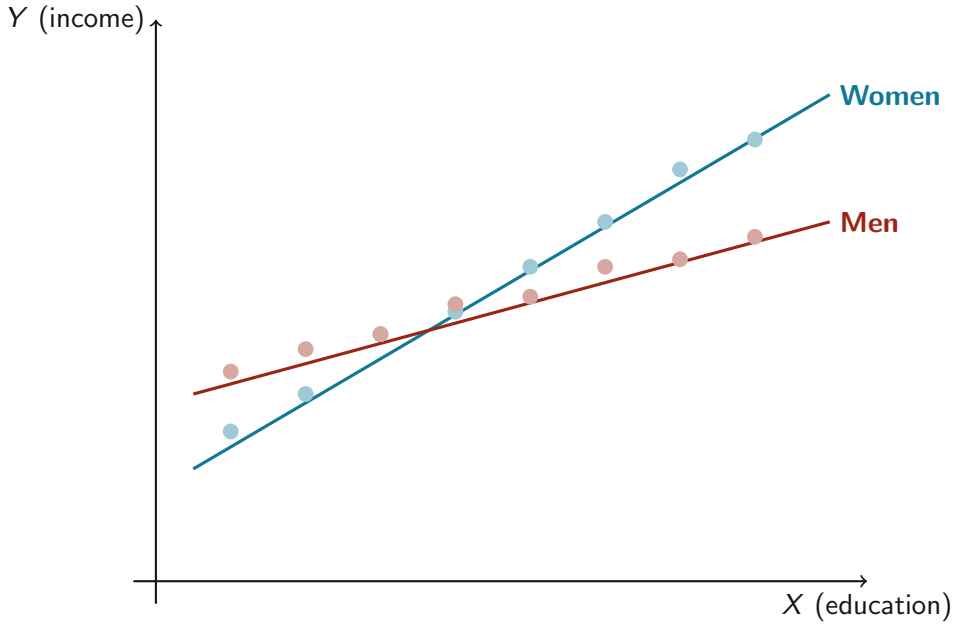
Continuous × categorical interactions

- When Z is categorical (e.g., gender, regime type)
- The interaction gives a **different slope** for each group

- Example: `lm(income ~ education * gender, data = df)`
  - → One slope for men, a different slope for women

- Equivalent to fitting separate regressions by group
- But estimated jointly (shares the error variance)

This is the most intuitive case. With a binary moderator, the interaction just gives you two different slopes. The advantage of estimating them jointly (instead of running separate regressions for each group) is that you pool information about the error variance, which gives more precise estimates. You can also test whether the difference between slopes is statistically significant, which is what β_3 tells you.



This picture shows what a continuous-by-categorical interaction looks like. The two lines represent the relationship between education (X) and income (Y) for men and women separately. The slopes differ: the return to education is steeper for women than for men. The interaction coefficient β_3 captures the difference between these two slopes. If the lines were parallel, the interaction would be zero and the effect of education would be the same for both groups.

Continuous $\times$ continuous interactions

- When both X and Z are continuous
- The slope of X varies smoothly with Z (and vice versa)
- Harder to interpret from coefficients alone
- Best communicated through plots:
  - → Predicted values at different combinations of X and Z
  - → Marginal effect of X across values of Z

Continuous-by-continuous interactions are harder because the slope is changing continuously, not jumping between two values. There's no single "effect of X" – it's a line (or surface). The best approach is visualization: plot predicted Y at different values of Z (holding X constant), or plot the marginal effect of X as a function of Z. The marginaleffects package makes this easy. Mention that continuous interactions often have low power, so large samples help.

Common mistakes with interactions

- **Mistake 1**: Interpreting β_1 as "the effect of X"
  - → It's only the effect when $Z = 0$
  - → May not even be meaningful!
- **Mistake 2**: Omitting constitutive terms
  - → Always include X and Z separately, not just $X \times Z$
- **Mistake 3**: Not showing how the effect varies
  - → Plot the marginal effect across values of Z

These are the three most common mistakes in published research using interactions:

- Mistake 1: The coefficient on X in an interaction model is conditional on $Z = 0$. If Z has no natural zero (like GDP per capita), this is meaningless. Center Z if needed.
- Mistake 2: Brambor, Clark, and Golder (2006) showed that many published papers omit the constitutive terms, which biases all the estimates. R includes them automatically with the * operator, so this is less of a problem in practice.
- Mistake 3: A regression table for an interaction model is essentially uninterpretable without a plot showing how the effect varies. Always plot.

Visualizing interactions

- Tables of coefficients are hard to interpret
- Better approach:
  - → Plot predicted values of Y for different combinations of X and Z
  - → Plot the marginal effect of X across values of Z
  - → Include confidence intervals
- In R: `marginaleffects::plot_predictions()`

Show a quick example of how to use plot_predictions(). The basic call is plot_predictions(model, condition = c("x", "z")). This will plot predicted Y on the y-axis, X on the x-axis, with separate lines for different values of Z. You can customize the values of Z with the condition list. For marginal effects plots, use plot_slopes() from the same package. Emphasize that the plot IS the result for interaction models – the table is just a summary.

Roadmap

Why presentation matters

- A regression table is not the end of the analysis
- Readers need to understand the **substance** of your findings
- Good presentation:
  - → Shows what the results **mean**, not just what they are
  - → Communicates **uncertainty** honestly
  - → Helps readers evaluate the **size** of effects

This section is about the last mile: you've run the regression, now you need to communicate what you found. Many students stop at summary() output or a screenshot. We want publication-quality tables and plots that a reader can understand without seeing the code. Emphasize that a number in a table only matters if the reader can assess whether it's big or small, and whether it's precise or noisy.

The `modelsummary` package

- Creates publication-quality tables from model objects
- Basic usage:
  - → `modelsummary(model)`
  - → `modelsummary(list(m1, m2, m3))`
- Output formats: LaTeX, HTML, Word, markdown
- Highly customizable: statistics, labels, notes

modelsummary is the go-to package for regression tables in R. It supports dozens of model types and output formats. Key features to mention: you can rename coefficients with coef_map, add goodness-of-fit statistics with gof_map, and include robust SEs with the vcov argument. For the assignments, they will use modelsummary to create tables comparing multiple models. Show a quick example if time permits.

Coefficient plots

- A visual alternative to tables
- `modelsummary::modelplot(model)`
  - → Each coefficient as a point with confidence interval
  - → Easy to compare multiple models
- Often more effective than tables for communicating results
- Readers immediately see which effects are large vs. small

Coefficient plots (sometimes called "dot-and-whisker" plots) show each coefficient as a dot with a confidence interval bar. They are especially useful when you have many predictors or want to compare coefficients across models. The visual makes it immediately clear which effects are significant (intervals not crossing zero) and how they compare in magnitude. Use modelplot() from modelsummary – it returns a ggplot object you can customize further.

Building sequential models

- Common strategy: show how results change as you add variables
- Step 1: Bivariate model (just X and Y)
- Step 2: Add control variables
- Step 3: Add interactions
- Present all three in one table:
  - → `modelsummary(list(m1, m2, m3))`
- Shows robustness and what adding controls does to the estimate

This is standard practice in quantitative social science. By showing a sequence of models, you let the reader see how the key coefficient changes as you add controls. If β_1 barely changes, that's reassuring – it suggests the relationship is robust to confounding. If it changes a lot, that's informative too – it tells you which variables were confounding the relationship. The sequence also connects directly to the OVB formula we discussed earlier.

Example workflow in R

```
m1 <- lm(y ~ x, data = df)
m2 <- lm(y ~ x + z1 + z2, data = df)
m3 <- lm(y ~ x * z1 + z2, data = df)
modelsummary(list(m1, m2, m3), vcov = "robust")
modelplot(list(m1, m2, m3))
plot_predictions(m3, condition = c("x", "z1"))
```


Walk through this as a template workflow. Note the addition of vcov = "robust" in modelsummary – this is the practical advice from the OLS properties section in action. The three-step workflow (bivariate, controls, interaction) is very common and they should follow it in their assignments. Each step adds a layer of complexity and tells a richer story about the data.

Summary: Key takeaways

- Regression estimates conditional expectations
- OLS is unbiased under standard assumptions
- Always use robust standard errors
- Multiple regression: "holding constant" interpretation
- Control variables help only if chosen correctly
- Interactions model conditional relationships
- Present results clearly: tables, coefficient plots, marginal effects

Recap the session. Highlight the three most important practical takeaways: (1) always use robust SEs, (2) think carefully about which controls to include and which to avoid, and (3) when using interactions, always plot the marginal effects. These three lessons will come up again and again throughout the course and in their research projects.

For next week

- Read Urdinez & Cruz (2020), chapter 8
- Read Gelman et al., chapters 13–14
- Complete Assignment 2

- Next session: Binary outcomes
  - → Linear probability model vs. logistic regression
  - → Interpreting logit results
  - → Predicted probabilities and marginal effects

Mention that the readings focus on regression with binary outcomes (logit/probit), which we cover next week. Assignment 2 has two parts: the in-class lab they started today and the take-home exercises due before next session. Encourage them to start the take-home early and come to office hours if stuck.

Questions?