Causal Machine Learning (I): Regression

Prof. Tzu-Ting Yang
楊子霆

Institute of Economics, Academia Sinica
中央研究院經濟研究所

March 26, 2026

Main Idea

Causal Machine Learning: Overview

- **Machine learning (ML)** methods use data-driven algorithms to model the relationship between an outcome $\mathbf{Y}$ and covariates $\mathbf{X}$
  - There are many ML methods; **regression** is a fundamental one
  - The best method depends on the application
- The primary goal of **machine learning** is to **predict** an outcome Y given covariates X
  - Forecast economic growth rate using many factors
  - Predict user-rating of products
  - Classify the types of individuals given many socio-economic measures and predict their loan repayment probability

From Prediction to Counterfactual Prediction

- **Causal inference** requires answering a fundamentally different question:
  - Not just "what is Y likely to be given X?"
  - But "what would Y have been **under a different treatment status**?" $\longrightarrow$ **counterfactual**
- **ML methods help us predict the missing counterfactual:**
  - Use X_i to find units with similar characteristics but different treatment status
  - Under the CIA, the predicted counterfactual is valid
  - **Regression** is the simplest ML tool for predicting counterfactual outcomes

Main Idea of Regression

- A multivariate regression can help us study the relationship between treatment D_i and outcome Y_i

$$\mathrm{Y}_i = \delta + \alpha D_i + X_i\beta + \epsilon_i$$

- Here, X is a vector of covariates and β is a vector of coefficients

$$X = (x_1', x_2', \ldots, x_k')$$

$$\beta = (\beta_1, \beta_2, \ldots, \beta_k)$$

$$X\beta = \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k$$

Main Idea of Regression

- We can interpret α as the causal effect of treatment when we include all relevant confounding factors X_i in the regression
- The inclusion of X allows for an "apples-to-apples" comparison
  - We compare units with the same values of X but different values of treatment D

Identification

Identification Assumption

Conditional Independence Assumption

$$(Y_i^1, Y_i^0) \perp\!\!\!\perp D_i | X_i$$

- Both matching and regression require CIA (selection on observable) to get causal affects
    - But regression implicitly assume a specific functional form of **potential outcomes**
    - Matching additionally requires **common support**: treated and control units must overlap in their covariate distributions
    - Regression does **not** require common support — it extrapolates outside the region of overlap using the assumed functional form

Regression and Potential Outcome

- Regression estimates the causal effect by **predicting both potential outcomes** and taking the difference
- Under the CIA, we can estimate the following regression to get causal effect of D by including all possible confounding factors X

$$\mathrm{Y}_i = \delta + \alpha D_i + X_i \beta + \epsilon_i$$

- The fitted model predicts **both** potential outcomes for any unit with covariates X_i:
  - Predicted Y^1 (set $D = 1$): $\mathrm{E}[\mathrm{Y}_i^1 | X_i] = \delta + \alpha + X_i \beta$
  - Predicted Y^0 (set $D = 0$): $\mathrm{E}[\mathrm{Y}_i^0 | X_i] = \delta + X_i \beta$
  - CIA implies $\mathrm{E}[\epsilon_i | D_i, X_i] = 0$, so these predictions are valid
- The **causal effect** is the difference between the two predicted outcomes:

$$\mathrm{E}[\mathrm{Y}_i^1 - \mathrm{Y}_i^0 | X_i] = \alpha \quad \text{(constant across all } X_i\text{)}$$

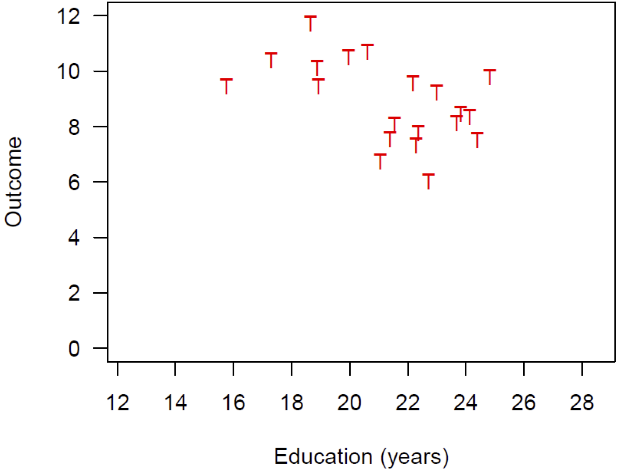
Regression and Matching

A Graphical Example

- Suppose we want to examine the effect of a treatment D on an outcome Y
  - Education is a observed confounding factor X
- Matching:
  - Require sufficient overlap in covariate distributions (X) between treated and control groups
  - This is known as the common support assumption
  - Ensures valid counterfactual comparisons

Regression and Matching

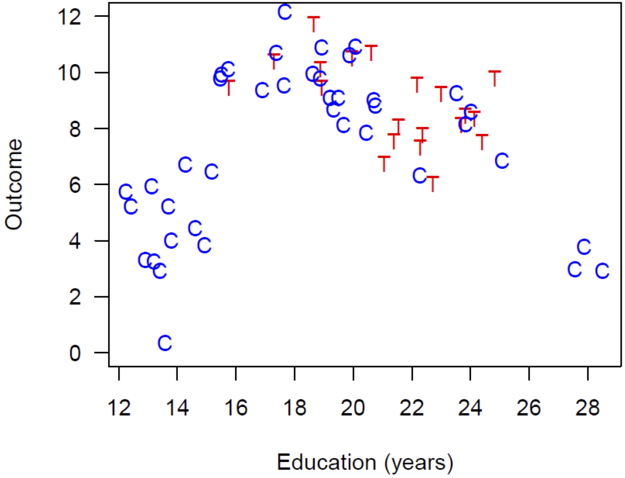
A Graphical Example



Source: Ben Elsner's slides

Regression and Matching

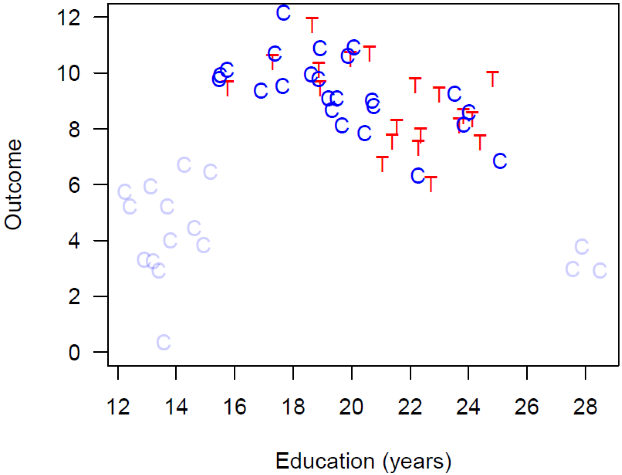
A Graphical Example



Source: Ben Elsner's slides

Regression and Matching

A Graphical Example



Source: Ben Elsner's slides

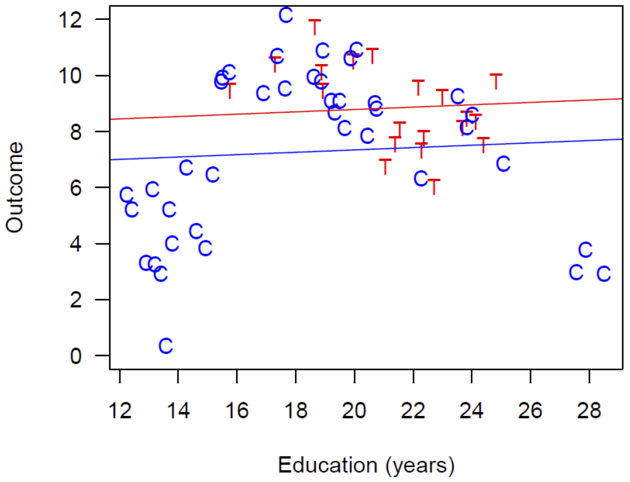
Regression and Matching

A Graphical Example

- Regression:
  - Can potentially extrapolate beyond the common support region
  - By relying on the specified regression model to predict counterfactual outcomes
    - Linear term for education: $\mathrm{Y}_i = \delta + \alpha D_i + \beta_1 X_i + \epsilon_i$
    - Quadratic term for education: $\mathrm{Y}_i = \delta + \alpha D_i + \beta_1 X_i + \beta_2 X_i^2 + \epsilon_i$
    - Estimated effect of treatment D can be different for these two models
  - The extrapolation may be unreliable if:
    - Model is misspecified
    - Extrapolation region is too far from data

Regression and Matching

A Graphical Example

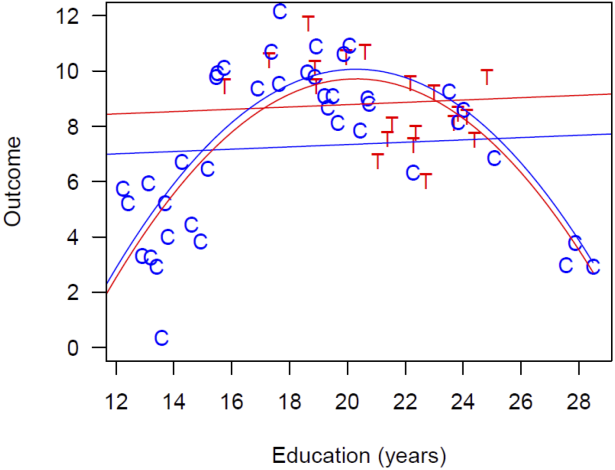


Source: Ben Elsner's slides

- The **red line** is the fitted regression for the treated group ($D = 1$); the **blue line** for the control group ($D = 0$)
  - Under the linear model, both lines have the **same slope** (β_1) but different intercepts
  - The **vertical gap** between the two lines is the estimated treatment effect $\hat{\alpha}$ — constant across all values of X

Regression and Matching

A Graphical Example

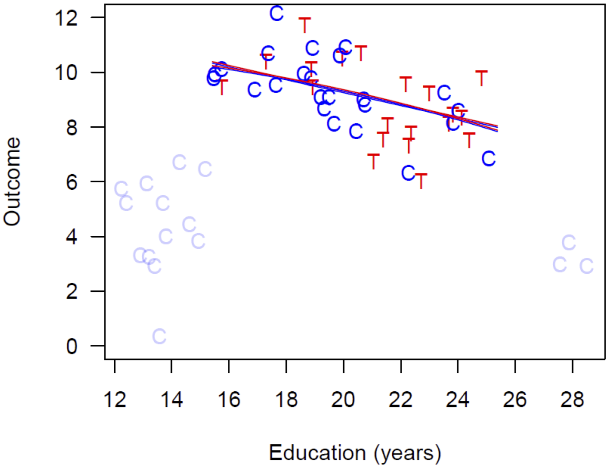


Source: Ben Elsner's slides

- With a **quadratic** functional form, the two curves have the same shape but are shifted vertically by $\hat{\alpha}$
  - The vertical gap between the red and blue curves is still the constant treatment effect $\hat{\alpha}$
  - However, the **curvature** of the fitted lines changes how each line extrapolates outside the common support region
  - This illustrates why the choice of functional form matters: different models can yield different treatment effect estimates

Regression and Matching

A Graphical Example



Source: Ben Elsner's slides

- Among these units within common support region, there is no difference in outcomes between treatment and control groups

- When covariate distributions do not overlap, regression must **extrapolate** into regions where one group is not observed
  - The estimated treatment effect relies entirely on the assumed functional form in those regions
  - Results can be sensitive to model misspecification
- Matching avoids extrapolation by restricting comparisons to the **common support region**
- **Trade-off between the two approaches:**
  - Matching: no functional form assumptions, but discards observations outside common support
  - Regression: uses all data and is more efficient, but risks bias if the functional form is misspecified

Identification Results for Regression

- We estimate the following regression:

$$\mathrm{Y}_i = \delta + \alpha D_i + X_i\beta + \epsilon_i$$

- The estimated coefficient of treatment D is the following:

$$\alpha = \underbrace{\mathrm{E}[\mathrm{Y}_i|X_i, D_i = 1] - \mathrm{E}[\mathrm{Y}_i|X_i, D_i = 0]}_{\text{ODO at given } X_i}$$

- Based on CIA, including all relevant covariates X_i into regression can help us eliminate selection bias
- Note: the linear functional form assumption implies α is **constant** across all values of X

Identification Results for Regression

$$\alpha = \underbrace{\mathrm{E}[\mathrm{Y}_i|X_i, D_i = 1] - \mathrm{E}[\mathrm{Y}_i|X_i, D_i = 0]}_{\text{ODO at given } X_i}$$

$$= \underbrace{\mathrm{E}[\mathrm{Y}_i^1 - \mathrm{Y}_i^0|X_i, D_i = 1]}_{\text{CATT}} + \underbrace{\mathrm{E}[\mathrm{Y}_i^0|X_i, D_i = 1] - \mathrm{E}[\mathrm{Y}_i^0|X_i, D_i = 0]}_{\text{Selection Bias}}$$

$$= \underbrace{\mathrm{E}[\mathrm{Y}_i^1 - \mathrm{Y}_i^0|X_i, D_i = 1]}_{\text{CATT}} + \underbrace{0}_{\text{Selection Bias} = 0 \text{ by CIA}}$$

$$= \underbrace{\mathrm{E}[\mathrm{Y}_i^1 - \mathrm{Y}_i^0|X_i, D_i = 0]}_{\text{CATU}} = \underbrace{\mathrm{E}[\mathrm{Y}_i^1 - \mathrm{Y}_i^0|X_i]}_{\text{CATE}}$$

Identification Results for Regression

- With continuous or multi-valued covariates X_i, we obtain a CATE for each unique value of X_i
- Applying the **law of iterated expectations (LIE)**, we can aggregate CATE into ATE:

$$\mathrm{ATE} = \mathrm{E}_X \left[\mathrm{E}[\mathrm{Y}_i^1 - \mathrm{Y}_i^0 | X_i] \right] = \mathrm{E}[\mathrm{Y}_i^1 - \mathrm{Y}_i^0]$$

- Under the baseline regression without interaction terms, CATE $= \alpha$ for all X_i, so LIE immediately gives ATE $=$ ATT $=$ ATU $= \alpha$

Estimation

Estimation Methods

- So far, we've discussed **identification**: under what conditions can regression coefficients be interpreted as causal effects
- Now we turn to **estimation**: how do we obtain numerical values for these causal parameters?
- Multiple estimation methods exist:
  - Ordinary Least Squares (OLS)
  - Maximum Likelihood Estimation (MLE)

Review: Ordinary Least Squares Estimation

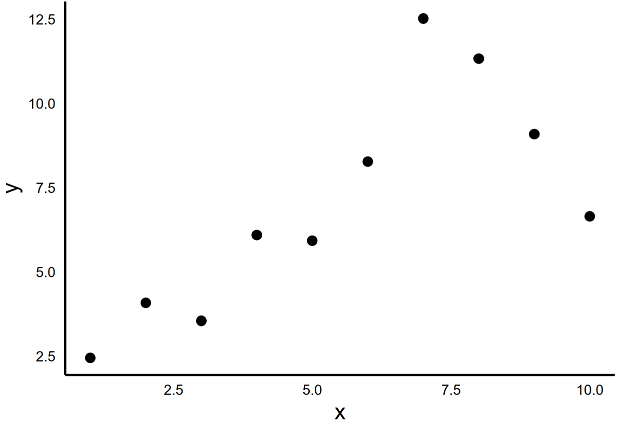
- Regression analysis assigns values to model parameters (δ, α, and β) to make predicted values $\hat{\mathrm{Y}}_i$ as close as possible to observed values Y_i
- OLS estimation accomplishes this by choosing values that **minimize the sum of squared errors (SSE)**

$$(\hat{\delta}, \hat{\alpha}, \hat{\beta}) = \arg\min_{\delta,\alpha,\beta} \frac{1}{N} \sum_{i=1}^{N} (\mathrm{Y}_i - \delta - \alpha D_i - X_i'\beta)^2$$

- OLS provides consistent estimates of the causal parameters we identified earlier

Review: Ordinary Least Squares Estimation

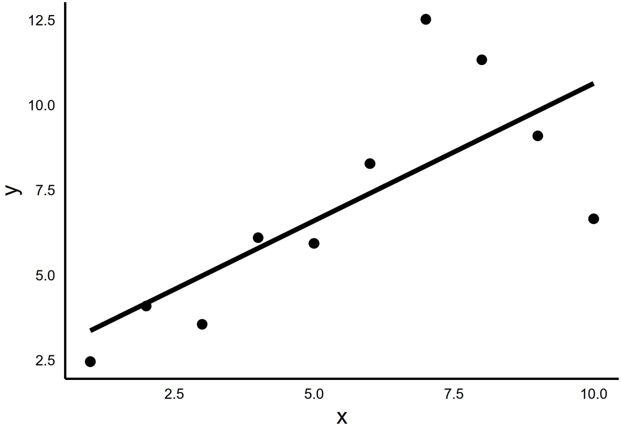
A Graphical Example



Source: Ben Elsner's slides

Review: Ordinary Least Squares Estimation

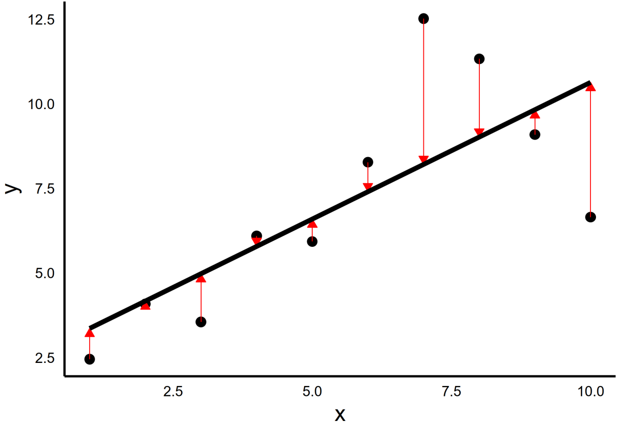
A Graphical Example



Source: Ben Elsner's slides

Review: Ordinary Least Squares Estimation

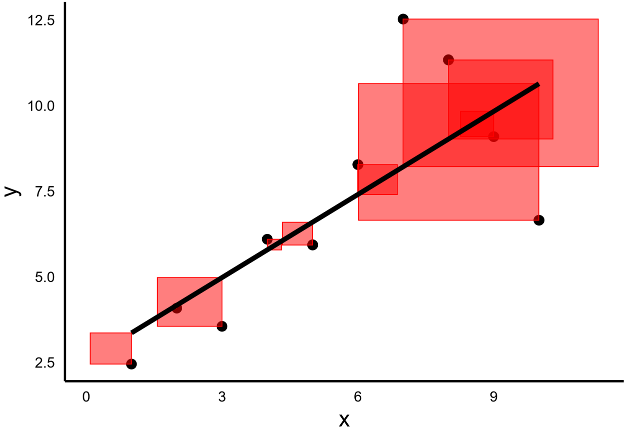
A Graphical Example



Source: Ben Elsner's slides

Review: Ordinary Least Squares Estimation

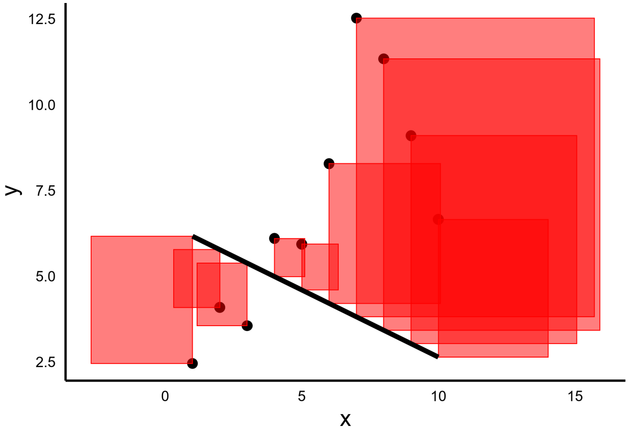
A Graphical Example



Source: Ben Elsner's slides

Review: Ordinary Least Squares Estimation

A Graphical Example



Source: Ben Elsner's slides

Review: Omitted Variable Bias

- OLS estimator for treatment effect α:

$$\hat{\alpha} = \frac{Cov(\mathrm{Y}_i, D_i)}{V(D_i)}$$

  - Covariance between Y_i and D_i: $\mathrm{Cov}(\mathrm{Y}_i, D_i) = \frac{1}{N}\sum_{i=1}^{N}(Y_i - \bar{Y})(D_i - \bar{D})$
  - Variance of D_i: $V(D_i) = \frac{1}{N}\sum_{i=1}^{N}(D_i - \bar{D})^2$

- Failure to include enough (right) control variables in the regression would result in bias

- The **OLS version** of the **selection bias** generated by inadequate controls is called **Omitted Variable Bias (OVB)**

Review: Omitted Variable Bias

- Suppose the true model is:

$$\mathrm{Y}_i = \delta + \alpha D_i + \beta X_i + \epsilon_i$$

- X_i is the observed characteristics (e.g. family wealth)
- But we estimate this model:

$$\mathrm{Y}_i = \delta + \alpha D_i + u_i$$

  - where $u_i = \beta X_i + \epsilon_i$
  - Assume $\mathrm{E}[\epsilon_i | D_i, X_i] = 0$

Review: Omitted Variable Bias

- OVB formula:

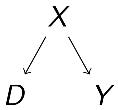
$$\hat{\alpha} \xrightarrow{p} \alpha + \frac{Cov(u_i, D_i)}{V(D_i)}$$
$$= \alpha + \beta \frac{Cov(X_i, D_i)}{V(D_i)}$$

  - Covariance between X_i and D_i: $\text{Cov}(X_i, D_i) = \frac{1}{N}\sum_{i=1}^{N}(X_i - \bar{X})(D_i - \bar{D})$
  - Variance of D_i: $V(D_i) = \frac{1}{N}\sum_{i=1}^{N}(D_i - \bar{D})^2$

- The difference between estimated treatment effect $\hat{\alpha}$ and true effect α depends on two components:

  1. β: The effect of omitted variable X_i on outcome variable Y_i
  2. $\frac{Cov(X_i, D_i)}{V(D_i)}$: The relationship between omitted variable X_i and treatment variable D_i

Review: Omitted Variable Bias



- The confounding factor X can result in the co-movement between treatment D and outcome Y
- Even if treatment D has no causal effect on outcome Y

Review: Omitted Variable Bias

Example

- OVB formula:

$$\hat{\alpha} \xrightarrow{p} \alpha + \frac{Cov(u_i, D_i)}{V(D_i)}$$
$$= \alpha + \beta \frac{Cov(X_i, D_i)}{V(D_i)}$$

- The difference between estimated effect of attending graduate school $\hat{\alpha}$ and true effect of attending graduate school α depends on two components:

  1. β: The effect of family wealth (omitted) X_i on earnings Y_i
  2. $\frac{Cov(X_i, D_i)}{V(D_i)}$: The relationship between family wealth X_i and attending graduate school D_i

Review: Omitted Variable Bias

- In RCT or other qusi-experimental methods, we can eliminate OVB since treatment assignment D_i is unrelated to other confounding factors X_i
  - $\frac{Cov(X_i, D_i)}{V(D_i)} = 0$
- In the regression, we can eliminate OVB by including all relevant confounding factors X_i into regression
  - $\frac{Cov(u_i, D_i)}{V(D_i)} = 0$
  - When we include X_i in regression model, $u_i = \epsilon_i$ which is unrelated to treatment status D_i

Review: Omitted Variable Bias

- OVB formula is a tool that allows us to consider the impact of controlling for variables we wish we had
  - We cannot use data to check the consequences of omitted variables that we do not observe
- But we can use the OVB formula to make a educated guess as to the likely consequences of their omission

$$\hat{\alpha} \xrightarrow{p} \alpha + \beta \frac{Cov(X_i, D_i)}{V(D_i)}$$

From OVB to Controlling for Covariates

- We saw that OVB arises when X_i is correlated with D_i and affects Y_i
- The solution: **partial out** the influence of X_i from treatment D_i
  - Isolate the variation in D_i that is **unrelated to** X_i
  - Use only that "clean" variation to estimate the effect on Y_i
- This logic is formalized by the **Frisch-Waugh-Lovell (FWL) Theorem**

Review: Frisch-Waugh-Lovell Theorem

Main Idea

- Consider the population model:

$$Y_i = \delta + \alpha D_i + \beta X_i + \epsilon_i$$

- The FWL Theorem states that the following OLS estimators for α are equivalent:
  1. Direct method: Regressing Y_i on D_i and X_i simultaneously
  2. Partitioned method:
     - Step 1: Regress D_i on X_i, obtain residuals $\tilde{D}_i$
     - Step 2: Regress Y_i on $\tilde{D}_i$
- The theorem also holds for multiple control variables

Review: Frisch-Waugh-Lovell Theorem

Key Insights

- Key insights from the partitioned method:
  - $\tilde{D}_i$ represents the part of D_i uncorrelated with X_i
  - Regressing Y_i on $\tilde{D}_i$ isolates the effect of D_i on Y_i, controlling for X_i

Bad Control Problem

Bad Control Problem

- Controlling for additional covariates increases the likelihood that regression estimates have a causal interpretation
- **Bad control problem**: more controls are not always better
  - Bad controls are **variables that could themselves be outcomes, which are also affected by treatment**
- **The bad control problem would lead to selection bias**

Bad Control Problem

- We should **NOT include bad controls** into regression or matching process
  - Even if including them can change estimated coefficients of treatment effect
- Good controls are variables that is **pre-determined**
  - **The value of variables have been determined before getting treatment**
  - Whether the variables are pre-determined or not, depending on timing of treatment
  - **Examples:**
    - The effect of master degree on earnings
    - **Pre-determined variables:** gender, age, birth place, father's wealth, mother's wealth
    - **Bad control variables:** occupation, employment, working industry

Bad Control Problem and Selection Bias

Example

- We are interested in the effect of master degree on earnings.
- People can work in two occupations:
  - White collar ($W_i = 1$)
  - Blue collar ($W_i = 0$)
- Occupation is highly correlated with both education (treatment) and earnings (outcome)
  - Occupation is a potential omitted variable, should we include it into our regression ?
  - Should we look at the effect of master degree on earnings for those within an occupation (e.g. white collar) ?

Bad Control Problem and Selection Bias

Example

- Note that having a master degree also increases the chance of getting a high-paying white collar job.
- That is, occupational choices are also affect by treatment (get a master degree): **Bad Controls**

Bad Control Problem and Selection Bias

Example

- Suppose master degree completion is **randomly assigned**
- Now consider comparing earnings **within white collar workers**:
  - Group A: Has a master degree **and** works white collar
  - Group B: No master degree, **but still** works white collar
- Group A is a **selected** sample — among those randomly assigned a master degree, only **some** end up in white collar jobs
- Group B is a **selected** sample — to obtain a white collar job without a master degree, these individuals likely have **higher unobserved ability**
- Conditioning on occupation **creates** a new form of selection bias, even when treatment was randomly assigned

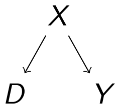
Bad Control Problem and Selection Bias

Intuition

- If our goal was to estimate the causal effect of having a master degree on earnings, it would be a bad idea to control for occupation
  - The reason is that one of the main ways that education can affect one's earning is through changing occupation
- If our regression controls for occupation, we might shut down this channel and underestimate the effect of having a master degree
  - The causal effect of having a master degree on earnings given the occupation does not change
- This is related to the concept of mediation effects, where occupation mediates the relationship between education and earnings

Good Controls

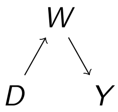
Example



- X is the confounding factor and good control variable
- If you want to estimate the (total) effect of treatment D, you should control for all confounding factors X
- In this case, there is no mediation effect to consider, as X is not on the causal path between D and Y

Bad Controls

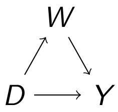
Example



- W is the mediator and bad control variable
- If you want to estimate the (total) effect of treatment D, you should NOT control for mediator W
- This is because W represents a mediation effect where part of the impact of D on Y flows through W

Bad Controls

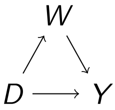
Example



- However, if you want to estimate the effect of treatment D on outcome Y NOT through the mediator W
  - You can get it by controlling for mediator W
  - This represents estimating the direct effect rather than the total effect (direct + indirect mediation effect)
- Mediation analysis would decompose the total effect into: direct effect and indirect effect through W

Mediation Effects

Decomposition



- Total Effect = Direct Effect + Indirect Effect
  - Direct Effect: Impact of D on Y not through W (control for W)
  - Indirect Effect (Mediation Effect): Impact of D on Y through W

Statistical Inference

Summary of Hypothesis Testing for Regression

- We estimate the following regression and want to test whether there is treatment effect:

$$Y_i = \delta + \alpha D_i + X_i\beta + \epsilon_i$$

1. Choose a null hypothesis:
   - We usually test whether there is **no average effect** of treatment
   - $H_0 : \alpha = 0$

Summary of Hypothesis Testing for Regression

2. Choose a test statistic

   - We use a t-statistic to measure whether our sample estimates support/against this null hypothesis
   - $t = \dfrac{(\hat{\alpha} - \alpha)}{\hat{\mathrm{SE}}(\hat{\alpha})}$

Summary of Hypothesis Testing for Regression

3. Estimate standard error of the estimator

- $\hat{\text{SE}}(\hat{\alpha}) = \sqrt{\dfrac{\sum_{i=1}^{N} \hat{\epsilon}_i^2 \widetilde{D}_i^2}{\left(\sum_{i=1}^{N} \widetilde{D}_i^2\right)^2}}$
  - $\hat{\epsilon}_i$ are the residuals from the main regression
  - $\widetilde{D}_i$ are the residuals obtained from regressing D_i on X_i
- The addition of covariates X has two opposing effects on $\hat{\text{SE}}(\hat{\alpha})$.
  1. $\hat{\epsilon}_i$ might decrease since addition covariates explain some of the variation in Y_i
  2. $\widetilde{D}_i$ falls when covariates that predict D_i are added to the regressions
- This is known as **heteroskedasticity-robust standard errors**
  - Provide valid standard errors of estimator α even in the presence of heteroskedasticity (i.e., non-constant variance)

Summary of Hypothesis Testing for Regression

4. Evaluate whether the sample estimator is against null hypothesis or not

   - **Goal:** Calculate **p-value**
     - **p-value:** Given null hypothesis is true, the probability of obtaining the sample estimates or more extreme ones
     - If this probability is high, it means the sample estimate might support for null hypothesis
     - If this probability is low, it means the sample estimate might be against null hypothesis

Summary of Hypothesis Testing for Regression

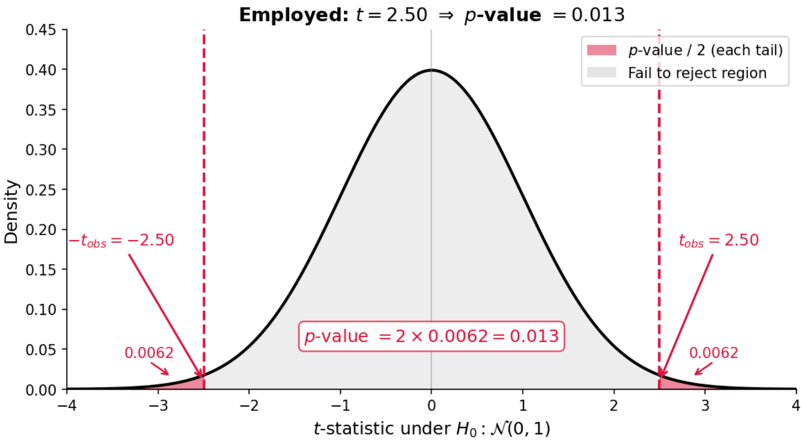
4. Evaluate whether the sample estimator is against null hypothesis or not

    - In order to calculate this probability (p-value), we need to know the distribution of the t-statistic under the null hypothesis

        - If sample size is sufficiently large, using **Central Limit Theorem (CLT)**, t-statistic will have standard normal distribution

Distribution of t-statistic

Visualizing the p-value



Summary of Hypothesis Testing for Regression

4. Evaluate whether the sample estimator is against null hypothesis or not
   - Based on standard normal distribution and sample estimator, we can get p-value
   - We reject the null hypothesis $H_0 : \alpha = 0$ when p-value is sufficiently low
     - We usually select an arbitrarily pre-defined threshold value θ, which is referred to as the **level of significance**
     - By convention, θ is commonly set to 0.1 or 0.05
   - If p-value is smaller than θ, we would say the sample estimate is **significantly different from the null hypothesis**

Interpretation of Regression Results

- We are only interested in α, the causal effect of treatment D on Y
  - The other coefficients $\beta_1, \beta_2, \ldots, \beta_k$ are NOT of interest
  - We include the covariates X to control for observed confounding factors
- Interpretation of α when controlling X
  - Holding all other variables X constant, a one unit increase in D leads to a α unit increase in Y

Interpretation of Regression Results

- Suppose the estimated regression is the following:

$$\hat{\mathrm{Y}}_{\mathrm{i}} = 35000 + 5000 D_i + 0.5 X_i$$

- Suppose the estimated standard error is:

$$\hat{\mathrm{SE}}(\hat{\alpha}) = 1000$$

- So the t-statistic for testing $H_0 : \alpha = 0$:

$$t = \frac{(\hat{\alpha} - \alpha)}{\hat{\mathrm{SE}}(\hat{\alpha})} = \frac{5000 - 0}{1000} = 5$$

Interpretation of Regression Results

- Using t-statistic, we can compute the p-value = 0.00001, which is much lower than 0.05 or 0.01
  - Given null hypothesis $H_0 : \alpha = 0$ is true, our estimate is unlikely to happen (but it happens!!)
  - It suggests our estimate is against the null hypothesis
  - Thus, we should reject the null hypothesis

- When Y is log-transformed, our model becomes:

$$\log(Y_i) = \delta + \alpha D_i + X_i'\beta + \epsilon_i$$

- This is known as a log-linear model
- α represents the log difference when D changes from 0 to 1
  - $\log(\mathrm{Y}_i^0) = \delta + X_i'\beta$
  - $\log(\mathrm{Y}_i^1) = \delta + \alpha + X_i'\beta$
- The exact percentage change in Y due to treatment is:
  - $\alpha = \log(\mathrm{Y}_i^1) - \log(\mathrm{Y}_i^0) = \log(\mathrm{Y}_i^1/\mathrm{Y}_i^0)$
  - % change in $Y = 100 \times (e^{\alpha} - 1)$
  - For small values of $|\alpha| < 0.1$, $\alpha \approx (\mathrm{Y}_i^1 - \mathrm{Y}_i^0)/\mathrm{Y}_i^0$

Interpretation of Regression Results

When Y is log-transformed

- D represents whether an individual has a graduate degree (1) or not (0)
- Interpretation of α:
  - If $\alpha = 0.10$, individuals with graduate degrees earn approximately 10% more than those without
- D represents years of education
- Interpretation of α:
  - $100 \times \alpha$ is the percentage change in Y for a one-unit increase in D
  - If $\alpha = 0.06$, each additional year of education is associated with approximately a 6% increase in earnings

Interpretation of Regression Results

Heterogeneous Treatment Effects

- Same treatment may affect different individuals differently
  - This leads to the concept of Conditional Average Treatment Effect (CATE)
  - CATE measures how the treatment effect varies across subgroups

Interpretation of Regression Results

Heterogeneous Treatment Effects

- Example: Analyze the differential effect of graduate degree on earnings by gender
- We introduce a dummy variable M for gender:
  - $M = 1$ for males
  - $M = 0$ for females
- Estimation methods:
  1. Include interaction terms: $D_i \times M_i$
  2. Subgroup regression: Run separate regressions for each group

- Our new regression model becomes:

$$\log(Y_i) = \delta + \alpha_1 D_i + \alpha_2 M_i + \alpha_3 (D_i \times M_i) + X_i \beta + \epsilon_i$$

  - Y_i is earnings
  - D_i is the dummy for graduate education (1 if yes, 0 if no)
  - M_i is the dummy for gender (1 if male, 0 if female)
  - $D_i \times M_i$ is the interaction term

- α_1: Effect of graduate degree on earnings for baseline group (females)
- α_3: Differential effect of graduate degree for males compared to baseline group (females)
- $\alpha_1 + \alpha_3$: Effect of graduate degree for males

Interpretation of Regression Results

Heterogeneous Treatment Effects

Suppose we estimate:

$$\log(Y_i) = 10 + 0.3D_i + 0.2M_i + 0.1(D_i \times M_i) + \dots$$

- For females ($M_i = 0$), graduate degree increases salary by approximately 30%
- Differential effect of graduate degree for males compared to females is about 10 percentage points
- For males ($M_i = 1$), graduate degree increases salary by approximately 40% ($0.3 + 0.1$)

STATA Example

STATA Example

- See **regression.do**
- Use cps_2014_16.dta

Examine Data

misstable: Examining missing values in your data

```
misstable summarize
misstable summarize inctot
```

- **misstable summarize:** Displays patterns of missing values for all variables in the dataset
- **misstable summarize varname:** Shows missing value patterns for a specific variable (e.g., inctot)

Create Sample for Analysis

generate: Create new variables

```
generate college = educ99>= 15
generate gender = sex == 1
generate college_gender = college*gender
```

- **generate:** Create a binary variable `college` indicating if education level is college or above
- **generate:** Create a binary variable `gender` indicating if gender is male (sex=1)

Create Sample for Analysis

replace/drop

```
replace incwage=. if incwage==9999999
drop if incwage==.
```

- **replace:** Replace missing values in incwage with "." if incwage equals 9999999
- **drop:** Drop observations with missing values in incwage

Create Sample for Analysis

recode: Recoding Variable Values

```
recode sex (1=0 "Male") (2=1 "Female"), gen(female)
```

- **recode:** Transform the original `sex` variable by reassigning its values and labels
  - **value mapping:** Change value 1 to 0 with label "Male", and value 2 to 1 with label "Female"
  - **gen(female):** Create a new variable named `female` instead of modifying the original `sex` variable
- Useful for creating binary indicator variables and standardizing coding schemes across datasets

Create Sample for Analysis

forvalues: Looping Through Numeric Sequences

```
forv i=1(1)5{
gen health_`i' = health==`i'
}
```

- **forvalues:** Loop through values 1 to 5 and create binary variables health_1, health_2, ..., health_5 indicating if health equals the corresponding value
    - **generate:** Create a new binary variable based on the condition health==`i
    - The loop generates 5 binary variables capturing different values of the health variable

```
foreach var in age inctot incwage {
sum `var', detail
}
```

- **foreach:** Loop through a list of variables (`age`, `inctot`, `incwage`) and perform the same operations on each one
  - **summarize:** Calculate descriptive statistics for each variable with the `detail` option
- The loop efficiently executes multiple commands on several variables without repetitive code

STATA Command: regress

- **regress**: Implement a regression
- Syntax:

```
regress depvar [indepvars] [if] [in] [weight] [,options
    ]
```

Reducing OVB by including covariates

```
reg incwage college , vce(robust)
reg incwage college health_1 - health_4, vce(robust)
reg incwage college health_1 - health_4 age i.race, vce(
    robust)
```

- Regress `incwage` on `college` using robust standard errors
  - Add health indicator variables (`health_1` to `health_4`) to the regression
  - Further control for `age` and `race` (using indicator variables) in the regression
- Option **vce(robust)**: use robust standard errors

Reducing OVB by including covariates

Output

```
. reg incwage college health_1 - health_4 age i.race, vce(robust)

Linear regression                               Number of obs     =     46,299
                                                F(20, 46276)      =          .
                                                Prob > F          =          .
                                                R-squared         =     0.1106
                                                Root MSE          =      48789
```

incwage	Coef.	Robust Std. Err.	t	P>|t|	[95% Conf.	Interval]
college	32661.38	659.6271	49.51	0.000	31368.5	33954.26
health_1	25663.82	771.8364	33.25	0.000	24151.01	27176.63
health_2	24268.25	639.2598	37.96	0.000	23015.29	25521.21
health_3	18432.33	610.8693	30.17	0.000	17235.02	19629.65
health_4	7670.004	667.9725	11.48	0.000	6360.768	8979.241
age	89.56983	10.82239	8.28	0.000	68.35778	110.7819

Understanding the Frisch-Waugh-Lovell Theorem

```
reg college health_1 - health_4 age i.race, vce(robust)
predict college_rid, residuals

reg incwage college health_1 - health_4 age i.race, vce(robust)
reg incwage college_rid, vce(robust)
```

- Regress `college` on all other covariates to obtain residuals `college_rid`
  - `college_rid` represents the part of `college` that is unrelated to other covariates
- Regress `incwage` on `college` and other covariates
- Regress `incwage` on `college_rid` gives same coefficient as previous regression

Understanding the Frisch-Waugh-Lovell Theorem

Output

```
. reg incwage college health_1 - health_4 age i.race, vce(robust)

Linear regression                               Number of obs     =     46,299
                                                F(20, 46276)      =          .
                                                Prob > F          =          .
                                                R-squared         =     0.1106
                                                Root MSE          =      48789

------------------------------------------------------------------------------
             |               Robust
     incwage |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     college |   32661.38   659.6271    49.51   0.000      31368.5    33954.26
    health_1 |   25663.82   771.8364    33.25   0.000     24151.01    27176.63
    health_2 |   24268.25   639.2598    37.96   0.000     23015.29    25521.21
    health_3 |   18432.33   610.8693    30.17   0.000     17235.02    19629.65
    health_4 |   7670.004   667.9725    11.48   0.000     6360.768    8979.241
         age |   89.56983   10.82239     8.28   0.000     68.35778    110.7819
```

Understanding the Frisch-Waugh-Lovell Theorem

Output

```
. reg incwage college_rid, vce(robust)

.inear regression                               Number of obs     =     46,299
                                                F(1, 46297)       =    2381.93
                                                Prob > F          =     0.0000
                                                R-squared         =     0.0772
                                                Root MSE          =      49684

------------------------------------------------------------------------------
             |               Robust
     incwage |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
 college_rid |   32661.38   669.2215    48.81   0.000     31349.69    33973.06
       _cons |   29208.86   230.9059   126.50   0.000     28756.28    29661.44
------------------------------------------------------------------------------
```

Subgroup Analysis

```
reg incwage college i.health age year i.race if sex==1, vce(
    robust)
```

- Option **if**: restrict sample to specific subgroup

Subgroup Analysis

Output

```
. reg incwage college health_1 - health_4 age i.race if sex==1, vce(robust)

Linear regression                               Number of obs     =     22,173
                                                F(17, 22150)      =          .
                                                Prob > F          =          .
                                                R-squared         =     0.1303
                                                Root MSE          =      58414
```

incwage	Coef.	Robust Std. Err.	t	P>|t|	[95% Conf.	Interval]
college	43138.43	1208.246	35.70	0.000	40770.18	45506.68
health_1	34501.94	1382.755	24.95	0.000	31791.64	37212.24
health_2	31922.23	1145.355	27.87	0.000	29677.25	34167.21
health_3	24665.21	1109.318	22.23	0.000	22490.86	26839.55
health_4	11095.74	1286.766	8.62	0.000	8573.588	13617.89
age	170.9962	19.49211	8.77	0.000	132.7903	209.2021

Subgroup Analysis

```
reg incwage college gender college_gender i.health age year
    i.race, vce(robust)
```

- Examine differential effect of `college` by gender

Reproducible Research: What is `markstat`?

- **markstat** is a Stata package for **reproducible research**
  - Write Markdown text and Stata code together in **one file** (extension: `.stmd`)
  - Compile to produce a **formatted document** with code, output, and explanation interleaved
  - Output format: **HTML** (open in browser) or **PDF** (requires LaTeX)
- Compared to a regular `.do` file:
  - A `.do` file only produces a log file with raw text output
  - A `.stmd` file produces a **readable, shareable document** with section headings, formatted tables, and narrative explanation

Reproducible Research: Installation (One-Time Setup)

- **Step 1: Install Pandoc** — an external program (not a Stata package)
  - Download and install from `pandoc.org`
  - Pandoc converts Markdown to HTML/PDF; markstat calls it internally
- **Step 2: Install markstat in Stata**

```
ssc install markstat
ssc install whereis
```

- **Step 3: Tell Stata where Pandoc is installed**

```
* Adjust path to match your Pandoc installation:
whereis pandoc "C:\Program Files\Pandoc\pandoc.exe"
```

- Steps 1–3 are done **only once**

Reproducible Research: Compiling a .stmd File

- **Step 4: Write your document** and save as regression.stmd
  - Use any plain text editor: Notepad, VS Code, or Stata's do-file editor
  - The full file regression.stmd is available on the course dropbox
- **Step 5: Compile** from Stata

```
* Navigate to the folder containing your .stmd file:
cd "C:\...\do"

* Compile to HTML (recommended):
markstat using regression, bundle

* Compile to PDF (requires LaTeX installed):
markstat using regression, pdf
```

- bundle packs everything (tables, graphs) into a single .html file — easy to share

Reproducible Research: `.stmd` File Structure (1)

- A `.stmd` file starts with a **YAML header** (title, author, date), followed by **Markdown text** and **Stata code blocks**
- **Quiet block** (`` ```s/q ``): code runs but **nothing is shown** — use for paths and data setup

~~~~
---
title: "Regression Analysis with CPS Data"
author: "Tzu-Ting Yang, Academia Sinica"
date: "Causal Data Course"
---

## 1. Data Setup

```s/q
* Runs silently: paths and data loading not shown in output
global rawdata = "C:\...\rawdata"
use "$rawdata\cps_2014_16.dta", replace
gen college = educ99 >= 15
```
~~~~

Reproducible Research: `.stmd` File Structure (2)

- **Regular block** (` ```s `): shows **both code and Stata output** in the document
- **Inline value** (`` ` s expr` ``): embed a computed result directly in a sentence

````
## 2. OLS Regression

```s
* Code and full Stata output both appear in document
reg incwage college health_1-health_4 age i.race, vce(robust)
```

The college wage premium is `s %12.0fc _b[college]` USD per year.
````

- The inline expression `` ` s %12.0fc _b[college]` `` inserts the estimated coefficient directly into the sentence — updates automatically when the model changes

Reproducible Research: Key Syntax Elements

- Three types of code blocks / inline expressions:

Syntax	What it does	Use for
``s ...```	Show code **and** output	regressions, summary stats
``s/q ...```	Run quietly, **nothing shown**	paths, data cleaning
` s *expr*`	Embed value **inline** in text	report coefficients

R Example

R Example

- See **regression.R**
- Use cps_2014_16.dta

Examine Data

skim(): Check for Missing Values

```
# Load the skimr package
library(skimr)

# Check for missing values with skim()
skim(acs_2015)
```

- **skim()** from the **skimr** package provides a comprehensive data summary
- The function automatically reports:
  - Number of missing values for each variable
  - Proportion of missing data
  - Data type information and summary statistics
- More efficient than manually checking with **is.na()** or **complete.cases()**

Create Sample for Analysis

Create new variables

```r
# Create a dummy variable for college education
acs_2015$college <- ifelse(acs_2015$educ99 >= 15, 1, 0)

# Create new gender variable (1 for male, 0 for female)
acs_2015$gender <- ifelse(acs_2015$sex == 1, 1, 0)
```

- `ifelse()` function:
    - Syntax: `ifelse(condition, value_if_true, value_if_false)`
    - Creates a new variable based on a condition

Create Sample for Analysis

Create new variables

```
# Replace missing income wage values and remove NA rows
acs_2015$incwage[acs_2015$incwage == 9999999] <- NA
acs_2015 <- na.omit(acs_2015, cols = "incwage")

# Generate log of incwage
acs_2015$log_incwage <- log(acs_2015$incwage)
```

- Handle missing values in `incwage`:
  - Replace 9999999 (missing value code) with `NA`
  - `na.omit()` removes rows with `NA` in `incwage`
- Create log-transformed income variable:
  - `log()` function calculates natural logarithm
  - Useful for analyzing proportional effects and normalizing skewed distributions

Create Sample for Analysis

mutate(): Create new variables

```
# Generate health dummy variables using dplyr
acs_2015 <- acs_2015 %>%
mutate(
health_1 = as.integer(health == 1),
health_2 = as.integer(health == 2),
health_3 = as.integer(health == 3),
health_4 = as.integer(health == 4),
health_5 = as.integer(health == 5)
)
```

- Use `dplyr`'s **mutate()** function to create all dummy variables in one step
    - The %>% pipe operator makes the code more readable by passing data through operations
    - **as.integer()** converts logical values to 0 or 1
- More explicit approach that clearly shows all variables being created

R Command: lm_robust()

- `lm_robust()`: Linear Models with Robust Standard Errors in R
- Syntax:

```
lm_robust(formula, data, subset, weights, se_type, ...)
```

- Required package:

```
library(estimatr)
```

Reducing OVB by including covariates

```r
# Define your models with robust standard errors
model1 <- lm_robust(incwage ~ college, data = acs_2015, se_type = "HC1")
model2 <- lm_robust(incwage ~ college + age + health_1 + health_2 + health_3 + health_4, data = acs_2015, se_type = "HC1")

# Print summaries to get robust standard errors
summary(model1)
summary(model2)
```

- Regress `incwage` on `college` using robust standard errors
- Use **lm_robust()** with **se_type = "HC1"** for direct computation of robust standard errors

Subgroup Analysis

```
model3 <- lm_robust(incwage ~ college + age + factor(race) +
    health_1 + health_2 + health_3 + health_4, data =
    acs_2015, subset = (gender == 1), se_type = "HC1")

# Print summary to get robust standard errors
summary(model3)
```

- subset parameter in **lm_robust()**:
  - Allows you to specify a condition for selecting observations
  - In this case, **subset = (gender == 1)** selects only male observations
  - Equivalent to filtering the data before running the regression

Subgroup Analysis

```
# Create interaction term
acs_2015$college_gender <- acs_2015$college *
    acs_2015$gender

# Define the model with interaction term and robust standard
     errors
model_interaction <- lm_robust(incwage ~ college + gender +
    college_gender +
health_1 + health_2 + health_3 + health_4 +
age + factor(race), data = acs_2015, se_type = "HC1")

# Print summary to get robust standard errors
summary(model_interaction)
```

- Create an interaction term between college and gender
- Include the interaction term in the regression model with robust standard errors

Reproducible Research: What is R Markdown?

- **R Markdown** is a file format (`.Rmd`) for reproducible research in R
  - Write R code and narrative text together in **one file**
  - Compile to produce a formatted **HTML, PDF, or Word** document with code, output, and explanation interleaved
- Compared to a regular `.R` script:
  - A `.R` script only produces console output
  - A `.Rmd` file produces a **readable, shareable document** with section headings, formatted tables, and equations
- Built into **RStudio** — no extra installation of external programs needed

Reproducible Research: Installation and Compile

- **Installation** (one-time, in R console):

```
install.packages("rmarkdown")
install.packages("knitr")
```

- **Compile** in RStudio: open `regression.Rmd` and click the **Knit** button

- Or compile from the R console:

```
rmarkdown::render("regression.Rmd")              # HTML output
rmarkdown::render("regression.Rmd",
  output_format = "pdf_document")                # PDF (requires
      LaTeX)
```

- Output file (`regression.html`) is saved in the same folder

- The full file `regression.Rmd` is available on the course dropbox

Reproducible Research: .Rmd File Structure (1)

- A .Rmd file starts with a **YAML header**, followed by Markdown text and R code chunks
- The **setup chunk** (include=FALSE): runs silently — use for paths and package loading

```
---
title: "Regression Analysis with CPS Data"
author: "Tzu-Ting Yang, Academia Sinica"
output:
  html_document:
    toc: true
    theme: flatly
---

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
library(haven);  library(estimatr);  library(dplyr)
rawdata <- "C:/nest/Dropbox/.../rawdata"
acs_2015 <- read_dta(paste0(rawdata, "/cps_2014_16.dta"))
```
```

Reproducible Research: .Rmd File Structure (2)

- **Regular chunk** ({r}): shows **both code and R output** in the document
- **Inline value** (`` ` r expr` ``): embed a computed result directly in a sentence

```
## OLS Regression

```{r model3}
model3 <- lm_robust(incwage ~ college + health_1 + health_2 +
            health_3 + health_4 + age + factor(race),
            data = acs_2015, se_type = "HC1")
summary(model3)
```


The college wage premium is
`r round(coef(model3)["college"], 0)` USD per year.
```

- The inline expression `` ` r round(coef(model3)["college"], 0)` `` inserts the estimated coefficient directly into the sentence

Reproducible Research: Key Syntax Elements

- Three types of code chunks / inline expressions:

Syntax	What it does	Use for
``{r} ...```	Show code **and** output	regressions, summary stats
``{r, include=FALSE} ...```	Run quietly, **nothing shown**	paths, data loading
`r *expr*`	Embed value **inline** in text	report coefficients

- echo=FALSE: hides code, shows output
- include=FALSE: hides both

Suggested Readings

- Chapter 2, Mastering Metrics: The Path from Cause to Effect
- Chapter 3, Mostly Harmless Econometrics
- Chapter 2, Causal Inference: The Mixtape