Introduction

Gov 51: Data Analysis and Politics

Scott Cunningham

Harvard University

Week 1
January 27 & 29, 2026

Welcome to Gov 51

Who Am I?

Scott Cunningham

- Professor of Economics, Baylor University
- Visiting Professor, Harvard Government Department

- Background in English literature before economics
- Believer that statistics is a *humanistic* discipline

Email: anthony_cunningham@fas.harvard.edu

What Is This Course About?

Learning to use data to answer questions about politics and society

Questions like:

- How can we measure racial discrimination in job hiring?
- What is the best way to predict election outcomes?
- What factors drive the onset of civil wars?
- Do policies actually achieve their intended effects?

By The End of This Course

You will be able to:

1. Evaluate claims about causality
2. Summarize and visualize data
3. Apply linear regression to analyze data
4. Understand uncertainty in data analysis
5. Use professional tools: R, RStudio, git, GitHub

You'll be able to read most quantitative papers in political science.

Why should you care about data analysis?

Data Is Everywhere

In Academia

- Senior theses
- Graduate school applications
- Research assistantships
- Understanding what you read

In Industry

- Consulting
- Tech companies
- Campaigns and polling
- Policy analysis

The skills you learn here transfer everywhere.

These Skills Are in High Demand

Major tech companies have built teams around **causal inference** and **experimentation**:

- **Netflix**: Dedicated “Experimentation & Causal Inference” research team
- **Uber**: Developed CausalML, an open-source causal inference package
- **Microsoft**: Causality and Machine Learning group; created DoWhy and EconML
- **Meta**: Core Data Science team runs experiments at massive scale
- **Amazon, Google, Airbnb, Spotify**: All hire for these skills

Data scientist jobs are projected to grow 34% from 2024–2034 (BLS).

The Market Values These Skills

Median data scientist salary: $112,590 (BLS, 2024)

- Entry-level (0–2 years): $80,000–$105,000
- Mid-level (3–5 years): $100,000–$135,000
- Senior (6+ years): $140,000–$180,000+
- Big Tech (L5–L6): $180,000–$450,000+
- Principal level (L7 at Amazon): $700,000–$800,000+

The path to these roles starts early — the skills you build now compound.

You're not just learning academic methods — you're building marketable skills.

Course Structure

Component	Weight
Problem Sets (4)	40%
Midterm Exams (2)	40%
Final Project	20%

- Problem sets due Wednesdays 11:59pm via Gradescope
- In-class midterm exams (no notes, no computers)
- Final project: your own research question and data

Late policy: −10% per day; zero after 7 days.

Course Materials

Required Textbook (either edition is fine):

- Imai & Williams, *QSS: An Introduction in tidyverse* (2022), or
- Imai, *Quantitative Social Science* (2018)

Software (all free):

- R — statistical programming language
- RStudio — development environment
- Git & GitHub — version control

We'll get everything set up in Problem Set 1.

Weekly Rhythm

Day	Activity
Before Tuesday	Read the assigned QSS sections
Tuesday	Lecture (concepts)
Thursday	Lecture (application)
Section	Hands-on practice with TFs

Key principle: Predictable structure, every week.

What matters most is what you actually *learn* — not the grade.

If you're struggling, come see me.

AI Policy

AI Policy

Certain assignments in this course will permit or even encourage the use of generative artificial intelligence (GAI) tools such as ChatGPT.

- The **default is that such use is disallowed** unless otherwise stated
- Any such use must be **appropriately acknowledged and cited**
- It is each student's responsibility to **assess the validity** of any GAI output that is submitted
- You bear the **final responsibility**
- Violations of this policy will be considered **academic misconduct**

Different classes at Harvard may implement different AI policies. It is your responsibility to conform to expectations for each course.

Why This Policy?

The goal of this course is for you to **learn to think with data**.

Using AI to generate answers defeats that purpose and will leave you **unprepared for exams**, which are completed in-class without AI assistance.

But there's a deeper reason...

AI and Learning

The Production of Cognitive Output

Cognitive tasks (research, code, analysis, homework) are produced with inputs:

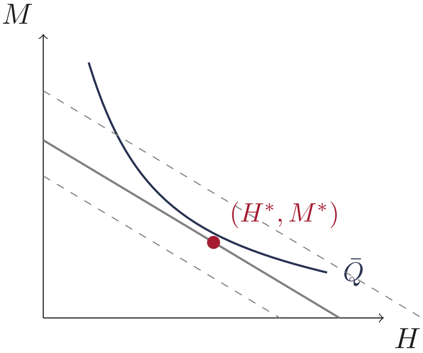
- H = Human time
- M = Machine time

The production function:

$$Q = f(H, M)$$

Key question: What is the shape of the isoquants?

Pre-AI: Quasi-Concave Isoquants



Tangency ⇒ interior solution with $H > 0$, $M > 0$

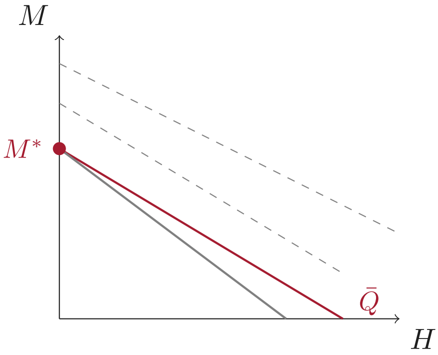
Cost minimization:

$$\min_{H,M} \; w_H H + w_M M \;\; \text{s.t.} \;\; f(H, M) = \bar{Q}$$

Result: Tangency condition

- MRTS $= w_H / w_M$
- Always use *some* human time
- Interior solution

Post-AI: Linear Isoquants (Perfect Substitutes)



Isocost flatter than isoquant ⇒ corner at M axis

With linear isoquants:

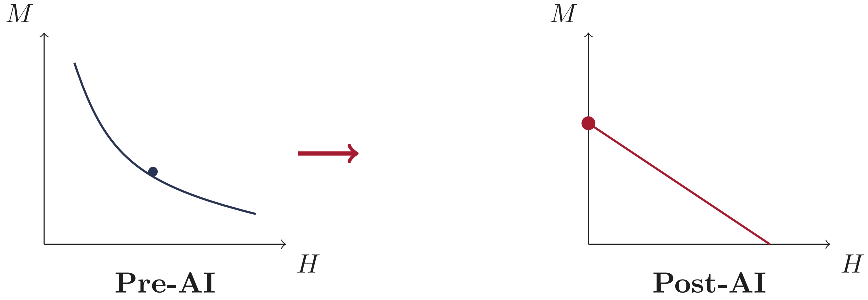
- MRS is constant
- Compare slopes: isocost vs isoquant

Corner solution:

- Isocost flatter ⇒ use only M
- Isocost steeper ⇒ use only H

AI makes w_M cheap ⇒ specialize in machine time.

The Homework Example

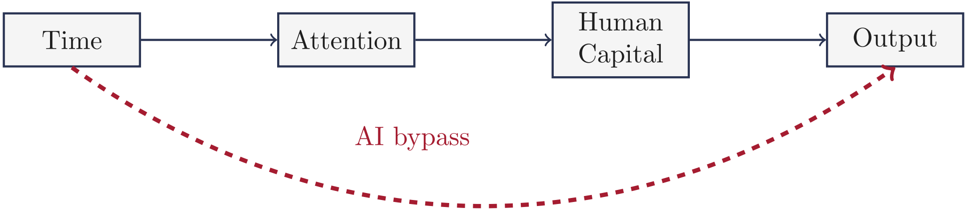


Pre-AI: Must invest human time to complete homework.

Post-AI: Can “complete” homework with $H = 0$.

The problem: “Homework” is completed, but was anything learned?

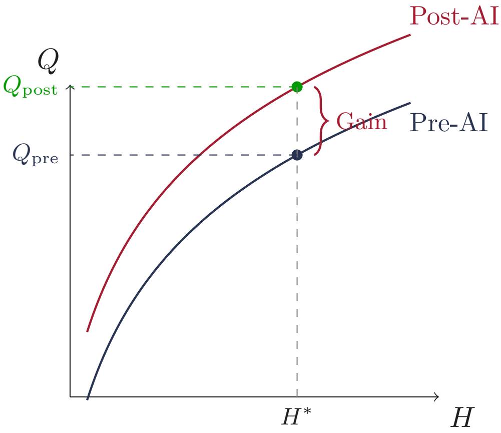
Time, Attention, and Human Capital



- **Time** → Attention: Can't attend to what you don't spend time on
- **Attention** → Human Capital: Learning requires focus
- **Human Capital** → Output: Knowledge produces results

AI allows bypassing the chain: Output without human capital accumulation.

The Productivity Curve: AI Shifts It Up



AI shifts the curve up:

- Same human time H^*
- Higher output $Q_{\text{post}} > Q_{\text{pre}}$

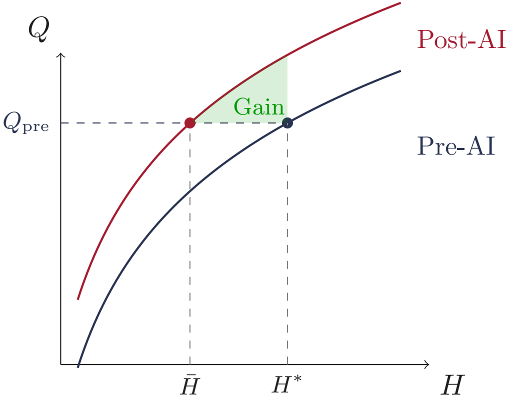
The productivity gain:

$$\Delta Q = Q_{\text{post}} - Q_{\text{pre}}$$

- Same time: Best outcome

If you maintain H^*, AI is unambiguously good.

The Productivity Curve: Moderate Time Reduction



AI induces time reduction:

- Tasks feel "easier"
- Temptation to reduce H

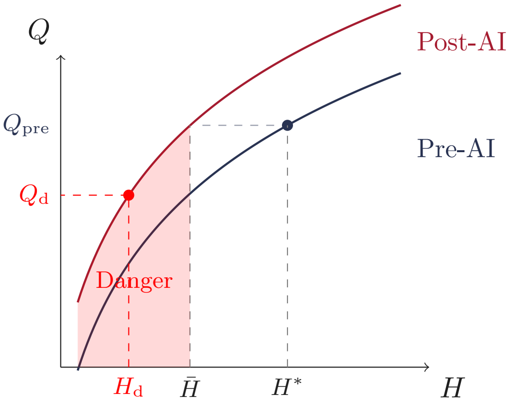
But there's a safe zone:

- As long as $H > \bar{H}$
- Output stays above Q_{pre}
- Shaded region: output-enhancing even with reduced time

- Moderate reduction: OK

Some time savings are fine—the curve shifted up enough to absorb them.

The Productivity Curve: The Danger Zone



Excessive time reduction:

- When $H < \bar{H}$
- $Q_{\text{danger}} < Q_{\text{pre}}$
- Worse off than before AI!

The paradox:
Productivity-enhancing technology can reduce actual output if behavioral response is large enough.

This is the homework problem: you "finish" faster but learn less.

The Bottom Line on AI

1. AI can genuinely help you learn *if* you use it right
2. The danger: using it to skip the struggle that produces learning
3. Exams are in-class, closed-book, no AI—if you don't learn, you'll know

Use AI to understand more deeply, not to avoid thinking.

More on this in the AI Policy page on Canvas.

What Is Quantitative Social Science?

The Big Picture

Using data to learn about the social world

Four key activities:

1. **Describe** — What happened? What does the data show?
2. **Predict** — What will happen next?
3. **Explain** — Why did it happen? What causes what?
4. **Recommend** — What should we do?

Description

What happened?

- What was voter turnout in 2024?
- How has polarization changed over time?
- What is the unemployment rate?

Seems simple, but:

- How do we measure “polarization”?
- Whose data do we trust?
- What counts as “unemployed”?

Prediction

What will happen?

- Who will win the next election?
- Which voters are likely to turn out?
- Where will conflict break out?

Prediction is about **patterns**:

- Find relationships in historical data
- Apply them to new situations
- Accept that you'll sometimes be wrong

Causal Explanation

Why did it happen?

- Does voter ID laws reduce turnout?
- Do campaign ads change votes?
- Does economic growth reduce conflict?

This is the hardest question:

- Correlation is not causation
- We need special research designs
- Uncertainty is unavoidable

What’s the difference between prediction and causation?

Prediction vs. Causation

Prediction: Ice cream sales predict drowning deaths.

Causation: Does ice cream *cause* drowning?

For prediction:
Correlation is enough.
We just need patterns.

For causation:
We need to rule out confounders.
(Summer causes both!)

This course teaches both — and when to use each.

Getting Set Up

What Is R? What Is RStudio?

R is the engine

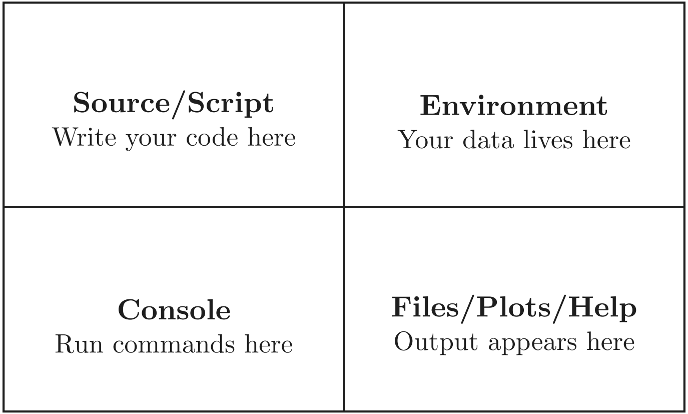
- The programming language
- Does the actual computation
- Free and open source
- `https://r-project.org`

RStudio is the dashboard

- Makes R easier to use
- Write scripts, see plots
- Manage files and projects
- `https://posit.co/downloads`

You need both. Install R first, then RStudio.

The RStudio Interface



Source/Script
Write your code here

Environment
Your data lives here

Console
Run commands here

Files/Plots/Help
Output appears here

PS 1 walks you through this step by step.

Introduction to R

Why R?

- Free and open source
- Dominant in academic social science
- Powerful for data analysis and visualization
- Huge community and package ecosystem
- Transferable skill for industry

Python is also excellent. R is our choice for this course.

R as a Calculator

R can do basic arithmetic:

```r
5 + 3       # Addition
5 - 3       # Subtraction
5 * 3       # Multiplication
5 / 3       # Division
5 ^ 3       # Exponentiation
sqrt(16)    # Square root
```

Try these in RStudio's console.

Objects and Assignment

We store values in **objects** using `<-`

```
result <- 5 + 3
result
## [1] 8

my_name <- "Scott"
my_name
## [1] "Scott"
```

- Object names are case-sensitive: `Result` ≠ `result`
- Choose meaningful names: `voter_turnout` not `x`

Vectors

A **vector** is a collection of values of the same type.

```r
# Election years in our data
years <- c(1980, 1984, 1988, 1992, 1996, 2000, 2004, 2008)

# Access elements
years[1]      # First element: 1980
years[c(1,3)] # First and third: 1980, 1988
years[-1]     # All except first
```

Vectors are the building blocks of data in R.

Vector Operations

Operations apply to *every element*:

```
# Total votes cast (thousands) in presidential elections
total_votes <- c(86515, 92653, 91595, 104405,
                 96263, 105375, 122295, 131304)

# Convert to millions
total_votes / 1000
## [1] 86.5 92.7 91.6 104.4 96.3 105.4 122.3 131.3

# Growth relative to 1980
total_votes / total_votes[1]
## [1] 1.00 1.07 1.06 1.21 1.11 1.22 1.41 1.52
```

Functions

Functions take inputs and produce outputs:

```
total_votes <- c(86515, 92653, 91595, 104405,
                 96263, 105375, 122295, 131304)

length(total_votes)  # Number of elections
## [1] 8

mean(total_votes)    # Average votes (thousands)
## [1] 103801

min(total_votes)     # Minimum
## [1] 86515
```

A Real Question: Measuring Voter Turnout

How Do We Measure Turnout?

What fraction of Americans voted?

Seems simple: votes cast ÷ population eligible to vote

But what's the denominator?

- **VAP**: Voting-Age Population (everyone 18+)
- **VEP**: Voting-Eligible Population (citizens who can legally vote)

VAP includes non-citizens and felons who can't vote.

Why Does This Matter?

VEP = VAP + overseas voters − ineligible voters

Ineligible voters include:

- Non-citizens (grew from 5.8M in 1980 to 19.4M in 2008)
- Disenfranchised felons (grew from 0.8M to 3.1M)
- Those who don’t meet residency requirements

Using VAP makes turnout look lower than it actually is.

Loading the Turnout Data

```r
# Load the data
turnout <- read.csv("turnout.csv")

# What do we have?
dim(turnout)
## [1] 14  9

names(turnout)
## [1] "year" "VEP" "VAP" "total" "ANES"
## [6] "felons" "noncit" "overseas" "osvoters"
```

14 elections (1980–2008), 9 variables.

Examining the Data

```
# First few rows
head(turnout, 3)
##   year    VEP    VAP total ANES felons noncit overseas
## 1 1980 159635 164445 86515   71    802   5756     1803
## 2 1982 160467 166028 67616   60    960   6641     1982
## 3 1984 167702 173995 92653   74   1165   7482     2361
```

- `VEP`, `VAP`, `total`: in thousands
- `ANES`: self-reported turnout (%)
- Notice 1982 has lower `total` — midterm election

Accessing Columns

Use $ to extract a column:

```
# Get the years
turnout$year
## [1] 1980 1982 1984 1986 1988 1990 1992 1994
## [9] 1996 1998 2000 2002 2004 2008

# Get total votes
turnout$total
## [1]  86515  67616  92653  64991  91595  67859
## [7] 104405  75106  96263  72537 105375  78382 ...
```

Each column is a vector.

Computing VAP Turnout Rate

```
# VAP turnout = total votes / (VAP + overseas) * 100
VAP_turnout <- turnout$total /
               (turnout$VAP + turnout$overseas) * 100

VAP_turnout
## [1] 52.0 40.6 52.9 36.4 50.0 36.3 54.4 38.3
## [9] 47.5 35.2 49.7 36.2 55.2 55.7
```

Presidential years: around 50%. Midterms: around 37%.

Computing VEP Turnout Rate

```r
# VEP turnout = total votes / VEP * 100
VEP_turnout <- turnout$total / turnout$VEP * 100

VEP_turnout
## [1] 54.2 42.1 55.2 38.1 52.8 38.4 58.1 41.1
## [9] 51.7 38.1 54.2 39.5 60.1 61.6
```

VEP turnout is *higher* — because denominator is smaller.

How Different Are They?

```
# Difference between VEP and VAP turnout
VEP_turnout - VAP_turnout
## [1] 2.2 1.5 2.3 1.7 2.8 2.1 3.7 2.8
## [9] 4.2 2.9 4.5 3.3 4.9 5.9

mean(VEP_turnout - VAP_turnout)
## [1] 3.2
```

On average, VAP understates turnout by 3.2 percentage points.

The gap is *growing* over time as the ineligible population grows.

Do People Lie About Voting?

The ANES survey asks people if they voted.

```r
# Compare self-reported (ANES) to actual (VEP)
turnout$ANES - VEP_turnout
## [1] 16.8 17.9 18.8 14.9 17.2  8.6 16.9 14.9
## [9] 21.3 13.9 18.8 22.5 16.9 16.4

mean(turnout$ANES - VEP_turnout)
## [1] 16.8
```

People overreport voting by about 17 percentage points!

This is called **social desirability bias**.

Why would people lie about voting?

What We Just Learned

1. **Measurement matters**: VAP vs VEP gives different answers
2. **Self-reports are biased**: People overreport socially desirable behavior
3. **The gap is growing**: As ineligible population increases, VAP becomes more misleading

This is what quantitative social science looks like:

- Start with a question
- Get data
- Compute and compare
- Draw conclusions

Visualizing Data with ggplot2

Why Visualize?

- Tables tell, pictures *show*
- Patterns are easier to see in plots
- Good figures communicate instantly
- ggplot2 makes publication-quality graphics

We'll use ggplot2 throughout this course.

The Grammar of Graphics

ggplot2 builds plots in layers:

```r
library(ggplot2)

ggplot(data, aes(x = ..., y = ...)) +
  geom_*()
```

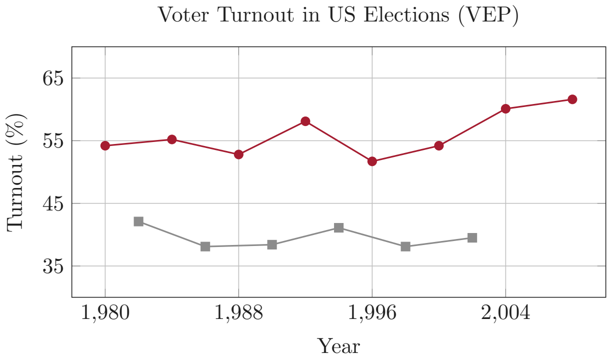
- `ggplot()`: Initialize with data
- `aes()`: Map variables to aesthetics (x, y, color, etc.)
- `geom_*()`: Add geometric objects (points, lines, bars)

Our First Plot: Turnout Over Time

```r
# Add turnout rates to our data
turnout$VEP_turnout <- turnout$total / turnout$VEP * 100
turnout$VAP_turnout <- turnout$total /
                       (turnout$VAP + turnout$overseas) * 100

# Plot VEP turnout over time
ggplot(turnout, aes(x = year, y = VEP_turnout)) +
  geom_line() +
  geom_point() +
  labs(x = "Year", y = "Turnout (%)",
       title = "Voter Turnout in US Elections")
```

Turnout Over Time



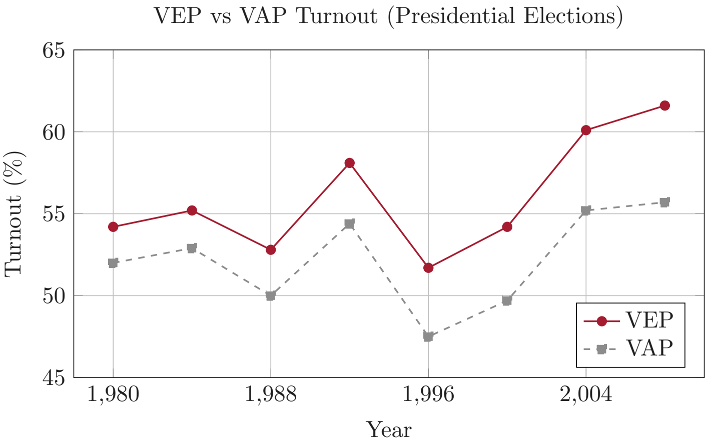
Clear pattern: Presidential elections (red) have much higher turnout than midterms (gray).

Comparing VAP vs VEP Turnout

```
# Reshape data for comparison (we’ll learn this later)
# For now, just see the result:

ggplot(turnout, aes(x = year)) +
  geom_line(aes(y = VEP_turnout, color = "VEP")) +
  geom_line(aes(y = VAP_turnout, color = "VAP")) +
  labs(x = "Year", y = "Turnout (%)",
       title = "VEP vs VAP Turnout Rates",
       color = "Measure")
```

VEP vs VAP: The Gap Grows



The gap between VEP and VAP grows over time.

What You Just Did

1. Loaded real data into R
2. Computed new variables from existing columns
3. Compared different measurement approaches
4. Made publication-quality visualizations

This is data analysis. You just did it.

Getting Started

This Week's Tasks

1. **Read** QSS Sections 1.1–1.4
2. **Install** R and RStudio (instructions on Canvas)
3. **Attend** your first section

R setup is built into Problem Set 1.

If You Get Stuck

1. Read the error message carefully
2. Google it (seriously — this is what professionals do)
3. Check the course discussion board
4. Ask in section
5. Come to office hours

TF: George Yean | CA: Harrison Huang

Getting stuck is normal. Asking for help is smart.

Looking Ahead

Next class: Causality and Randomized Experiments

- What does it mean to say X *causes* Y?
- Why are experiments the “gold standard”?
- How do we analyze experimental data?

Reading: QSS 2.1–2.4

Data analysis is a skill.

Skills require practice.

Start today.

Questions?

Scott: Tue/Thu 3–5pm | George: Thu 2–3pm, K455 | CA: Harrison Huang