

Applied Econometrics and Data Analysis

ECON 121

People

▶ Me: Tom Vogl

- ▶ User, not developer, of econometrics
- ▶ Researcher of health and population in developing countries
- ▶ Have been teaching variants of this course for over a decade
- ▶ Teaching is very important to me

▶ TA: Ida Grigoryeva

- ▶ Will hold biweekly problem set labs. Super useful.

▶ Reader: Songyu He

- ▶ Will have less direct contact with you.



Some Stuff You Didn't Learn in 120A-C

- ▶ How to use econometric tools in the real world?
 - ▶ e.g., R, Stata, Python
- ▶ How to deal with observations that are not independent?
 - ▶ e.g., siblings, neighbors, coworkers
- ▶ When to weight?
 - ▶ e.g., sampling weights, population weights
- ▶ How to model non-linear relationships?
 - ▶ Polynomials are not the only answer
- ▶ What to do with dependent variables that are not continuous?
 - ▶ e.g., voting, product choice, unemployment duration, work hours
- ▶ What is causality?
 - ▶ Hint: not a regression
- ▶ How to implement a difference-in-differences design?
 - ▶ Hint: can be a regression



Roadmap for the Semester

1. Estimation

- ▶ “Review” of OLS
- ▶ Departures from i.i.d.
- ▶ Maximum likelihood
- ▶ Limited dependent variables
- ▶ Panel data

2. Causality

- ▶ Difference-in-differences estimation
- ▶ Potential outcomes
- ▶ Randomized experiments
- ▶ Instrumental variables
- ▶ Regression discontinuity designs



Prerequisites

- ▶ Official prerequisite: ECON 120C
 - ▶ Targets Econ, Math/Econ, ManSci majors after full 120 series
- ▶ Unofficial alternative prerequisites: ECON 5 & ECON 120B
 - ▶ Targets BusEcon majors, who must take 5 but not 120C
- ▶ Unofficial will become official next quarter
- ▶ Either combo is solid
 - ▶ Students w/ 120C will have seen many 121 topics before
 - ▶ Students w/ 5 will have more experience with statistical computing



Course Structure

▶ Text

- ▶ No textbook, will rely on course notes

▶ Participation

- ▶ Participation matters for grade but can take many forms

▶ Deliverables

- ▶ Problem sets (5): R-based, can code in groups (max 4 people) but must write own answers, lowest score dropped
- ▶ Academic articles (4):
 - ▶ Group presentation for one article
 - ▶ Quizzes for all articles, lowest score dropped

▶ Final Exam

- ▶ Open book, R-based, basically an extra problem set
-



Lectures and Assignments

▶ Lectures:

- ▶ If people behind you might see non-course-related material on your laptop, please sit toward the back of the class
- ▶ I will post written notes by the night before each lecture, so you can print them or put them on your tablet for notetaking
- ▶ I will post the whiteboard and lecture video afterward

▶ Assignments:

- ▶ Work in groups on problem sets; let us know if you need help finding partners
- ▶ Late assignments not accepted
 - ▶ Lowest problem set grades are dropped
 - ▶ If you are late with an assignment, you should do it anyway



Statistical Computing

- ▶ Course used to use Stata

- ▶ Easier implementation of methods we study
- ▶ Used in ECON 120 series
- ▶ Less common outside academia (\$\$\$)

- ▶ This year, we will use R

- ▶ Free, more common in industry
- ▶ Avoided in the past because of the package zoo, but I figured out we can do most topics using `tidyverse` and `fixest`
- ▶ I hope you find it useful, but feedback is very useful

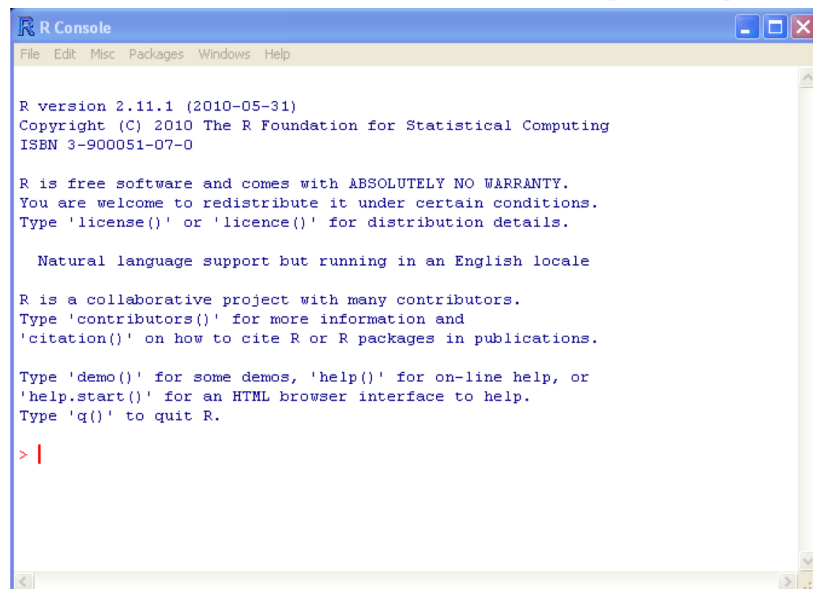
- ▶ Stuck on R?

- ▶ Refer to my examples from class (on GitHub)
 - ▶ Use help files / Stack Exchange / Google / ChatGPT
-



Background on R and RStudio

- ▶ R: programming language for statistical computing
 - ▶ Like the kitchen in my first apartment: you can cook, but not easily
- ▶ RStudio: integrated development environment for R
 - ▶ Like the kitchen inside Chancellor Khosla's house: neat, easy to use
- ▶ Download both: <https://posit.co/download/rstudio-desktop/>



```
R Console
File Edit Misc Packages Windows Help

R version 2.11.1 (2010-05-31)
Copyright (C) 2010 The R Foundation for Statistical Computing
ISBN 3-900051-07-0

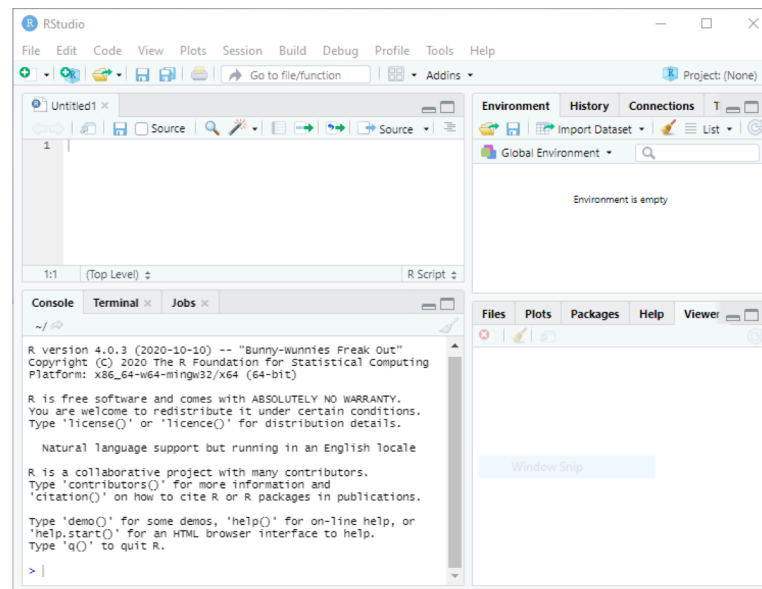
R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> |
```



Packages

- ▶ Of **many** user-written packages that extend R's capabilities, we will heavily use two:
 - ▶ `tidyverse` (a suite of packages) for basic tasks and graphing
 - ▶ `fixest` for regression estimation
 - ▶ Others will occasionally be necessary, but I will try to keep them to a minimum to avoid confusion
- ▶ To download a package, type:
 - ▶ `install.packages('packageName')`
- ▶ To load the package, type:
 - ▶ `library(packageName)`
- ▶ You only need to install each package once, but you must load it every time you open R or RStudio



Operators

- ▶ Operators perform operations on variables and values
- ▶ Arithmetic operators: `+` `-` `*` `/` `^`
- ▶ Comparison operators: `==` `!=` `>` `<` `>=` `<=`
- ▶ Logical operators: `&` `|`
- ▶ Assignment operators: `<-` `=`
 - ▶ Assign values to objects
 - ▶ They are the same, don't be confused when used interchangeably
- ▶ Pipe operators: `%>%` `|>`
 - ▶ String together sequences of operations: 'and then'
 - ▶ Original is `%>%` from `tidyverse`, `|>` is new addition to base R
 - ▶ Very similar, but we will mostly use `%>%`



Data Frames and Variables

- ▶ Unlike Stata, R has always been able to store multiple datasets (“data frames”) in memory simultaneously
- ▶ You need to specify the data frame when you ask R to perform calculations on a variable (or set of variables)
- ▶ The `$` operator is the standard way
 - ▶ `mean(census$educ)` estimates the mean of the variable `educ` from the data frame `census`
- ▶ For tidyverse functions, pipes or first argument do it
 - ▶ `census %>% summarize(mean(educ))` or `summarize(census, mean(educ))`
- ▶ Also `attach()`, but cumbersome w/ multiple data frames
 - ▶ `attach(census)` and then on a new line `mean(educ)`
- ▶ Missing values? Change to `mean(educ, na.rm=TRUE)`



Tidyverse Functions

- ▶ We will use many R functions over the term.
- ▶ A few from `tidyverse` (specifically `dplyr`) that we will use often:

- ▶ **Modifying data:**

`arrange()` orders observations by the variable(s) inside the parentheses

`filter()` subsets the data to observations that satisfy the statement inside the parentheses

`mutate()` creates, modifies, or deletes variables

`select()` keeps the variables inside the parentheses

- ▶ **Grouping data:**

`group_by()` groups the data by the variable(s) inside the parentheses

`summarize()` summarizes the data in a group → useful w/ `mean()`, `sd()`, `sum()`

`n()` gives the number of observations in a group

- ▶ **Evaluating conditional statements:**

`if_else()` evaluates truth of statement in parentheses → useful to create binary variables

`case_when()` is like `if_else()` but with multiple categories



Guidelines for Programming in R

- ▶ I will do most programming instruction by writing/running R scripts in class, but here are a few basic principles:
 1. Write code in a script (*.R) or Markdown (*.Rmd) file
 - ▶ R scripts are just code, do not automatically save output
 - ▶ R Markdown files save code, prose, and output to html or PDF
 2. Keep track of your working directory
 - ▶ If you are using or saving files locally, set a working directory
 - ▶ `getwd()` tells the current directory, `setwd()` sets a new one
 3. Annotate, annotate, annotate
 - ▶ Write comments to explain each step of your code
 - ▶ The `#` symbol starts a comment

