

Introduction

Instructor: Yuta Toyama

Last updated: 2020-03-30

Section 1

Introduction to the course

What do we do with econometrics?

1. Estimating economic relationships

1.1 Demand curve $\log(Q_t) = \alpha_0 + \alpha_1 P_t + \epsilon_t$

1.2 Production function $Y_{it} = A_{it} K_{it}^{\alpha} L_{it}^{\beta}$

2. Testing economic theory

- ▶ Does adverse selection exists in insurance markets?
- ▶ Are consumers rational?

3. Determine the effect of a given intervention (causal inference)

- ▶ What is the effect of increasing minimum wage on employment?
- ▶ Does democracy cause economic growth?
- ▶ Effects of going to private colleges on your future earnings.
- ▶ Note: Some questions may have underlying economic models, others may not.

4. Describe the data (prediction/forecasting)

- ▶ How does the distribution of wage look like?
- ▶ Relationship .b.w electricity consumption and temperature (possibly nonlinear).
- ▶ Related to machine learning (ML).

Goal of the Course

1. Understand program evaluation and/or causal inference approach in econometrics
 - ▶ Widely used in applied microeconomics (labor, health, development, IO)
2. read empirical papers that uses these methods
3. Learn how to conduct empirical analysis using R.

Relation to other courses

- ▶ This course does **NOT** cover:
 1. Discrete choice models
 2. Machine learning methods
 3. Structural estimation
- ▶ I recommend the following courses to learn these topics
 - ▶ Econometrics II or Applied Econometrics by Prof. Hoshino (for topic 1 and 2)
 - ▶ Economic Study (Microeconometrics) by Me (for topic 1 and 3)
 - ▶ Advanced Econometrics by Prof. Ueda and Prod. Dendup (for topic 1)
- ▶ I also highly recommend a special course on causal inference by Prof. Yamamoto
 - ▶ more rigorous treatment of the topic.
 - ▶ intensive course at the end of August (to be announced)

Why do we need to learn computation

1. Conduct statistical and empirical analysis using your own data set
 - 1.1 Construct the data set
 - 1.2 Describe the data
 - 1.3 Run regression or estimate an economic object
 - 1.4 Make tables and figures that show the results of your analysis.
2. Verify the econometric theory through numerical simulations.
 - ▶ Ex. Asymptotic theory considers the case when the sample size is large enough (i.e., $N \rightarrow \infty$)
 - ▶ Law of large numbers, central limit theorem
 - ▶ How well is the asymptotic approximation?
 - ▶ **Monte Carlo simulations**
- ▶ We will learn both aspects in this course.

Why do we use R?

► Many alternatives: Stata, Matlab, Python, etc. . .

1. Free software!!

- Stata and Matlab are expensive.
- Though you can use Matlab through the campus license from this April.

2. Good balance between flexibility in programming and easy-to-use for econometric analysis

- Stata is easy to use for econometric analysis, but hard to write your own program.
- Matlab is the opposite.
- You can do everything with R, including data construction, regression analysis, and complicated structural estimation.

3. Many users

- Popular in engineering.
- Many packages being developed (especially important for recently popular tools.)

► Note: Python seems also good, though I have not used it before.

Course Plan

1. Introduction to R programming (2-3 weeks)
2. Review of Statistics (1 week)
3. Linear Regression (3 weeks)
4. Instrumental Variable Regression (2 weeks)
5. Panel Data (2 weeks)
6. Program Evaluation, or Causal Inference Methods (4 weeks)

Section 2

Introduction of R and R studio

Getting Started

- ▶ You can use R/R studio in the PC room.
- ▶ However, I strongly recommend you install R/Studio in your laptop and bring it to the class.
- ▶ Install in the following order
 1. R: <https://www.r-project.org/>
 2. Rstudio: <https://www.rstudio.com/>
- ▶ Now open Rstudio.

Helps

- ▶ The RStudio team has developed a number of “cheatsheets” for working with both R and RStudio.
- ▶ This particular cheatsheet for Base R will summarize many of the concepts in this document.

Quick tour of Rstudio

- ▶ There are four panels
 1. Source: Write your own code here.
 2. Console:
 3. Environment/History:
 4. Files/Plots/Packages/Help:
- ▶ In the Source panel,
 - ▶ Write your own code.
 - ▶ Save your code in .R file
 - ▶ Click Run command to run your entire code.
- ▶ In the console panel,
 - ▶ After clicking Run in the source panel, your code is evaluated.
 - ▶ You can directly type your code here to implement.

Basic Calculations

To get started, we'll use R like a simple calculator.

Addition, Subtraction, Multiplication and Division

Math	R	Result
$3 + 2$	<code>3 + 2</code>	5
$3 - 2$	<code>3 - 2</code>	1
$3 \cdot 2$	<code>3 * 2</code>	6
$3/2$	<code>3 / 2</code>	1.5

```
1 + 3
```

```
## [1] 4
```

Exponents

Math	R	Result
3^2	<code>3 ^ 2</code>	9
$2^{(-3)}$	<code>2 ^ (-3)</code>	0.125
$100^{1/2}$	<code>100 ^ (1 / 2)</code>	10
$\sqrt{100}$	<code>sqrt(100)</code>	10

Mathematical Constants

Math	R	Result
π	pi	3.1415927
e	exp(1)	2.7182818

Logarithms

- ▶ Note that we will use \ln and \log interchangeably to mean the natural logarithm.
- ▶ There is no $\ln()$ in R, instead it uses $\log()$ to mean the natural logarithm.

Math	R	Result
$\log(e)$	<code>log(exp(1))</code>	1
$\log_{10}(1000)$	<code>log10(1000)</code>	3
$\log_2(8)$	<code>log2(8)</code>	3
$\log_4(16)$	<code>log(16, base = 4)</code>	2

Trigonometry

Math	R	Result
$\sin(\pi/2)$	<code>sin(pi / 2)</code>	1
$\cos(0)$	<code>cos(0)</code>	1

Getting Help

- ▶ In using R as a calculator, we have seen a number of functions: `sqrt()`, `exp()`, `log()` and `sin()`.
- ▶ To get documentation about a function in R, simply put a question mark in front of the function name and RStudio will display the documentation, for example:

```
?log  
?sin  
?paste  
?lm
```

Installing Packages

- ▶ One of the main strengths of R as an open-source project is its package system.
- ▶ To install a package, use the `install.packages()` function.
 - ▶ Think of this as buying a recipe book from the store, bringing it home, and putting it on your shelf.

```
install.packages("ggplot2")
```

- ▶ Once a package is installed, it must be loaded into your current R session before being used.
 - ▶ Think of this as taking the book off of the shelf and opening it up to read.

```
library(ggplot2)
```

- ▶ Once you close R, all the packages are closed and put back on the imaginary shelf.
- ▶ The next time you open R, you do not have to install the package again, but you do have to load any packages you intend to use by invoking `library()`.