Module 0: Getting Started

Part 1: Class Overview and Software Setup

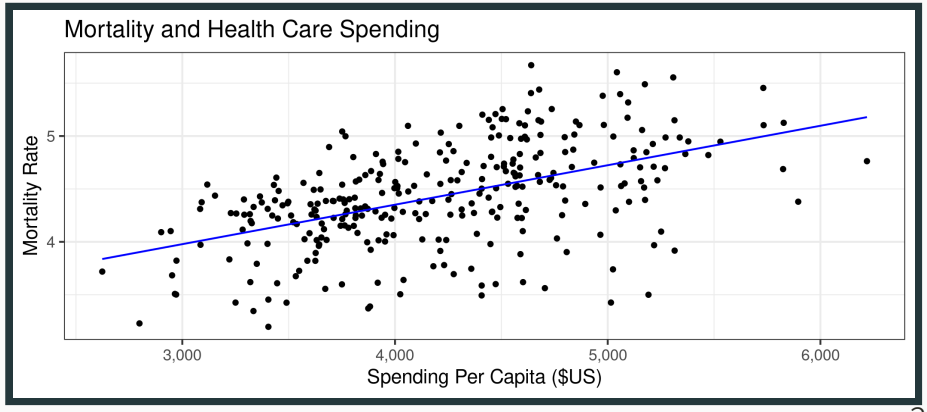
Ian McCarthy | Emory University
Econ 470 & HLTH 470

Motivating question

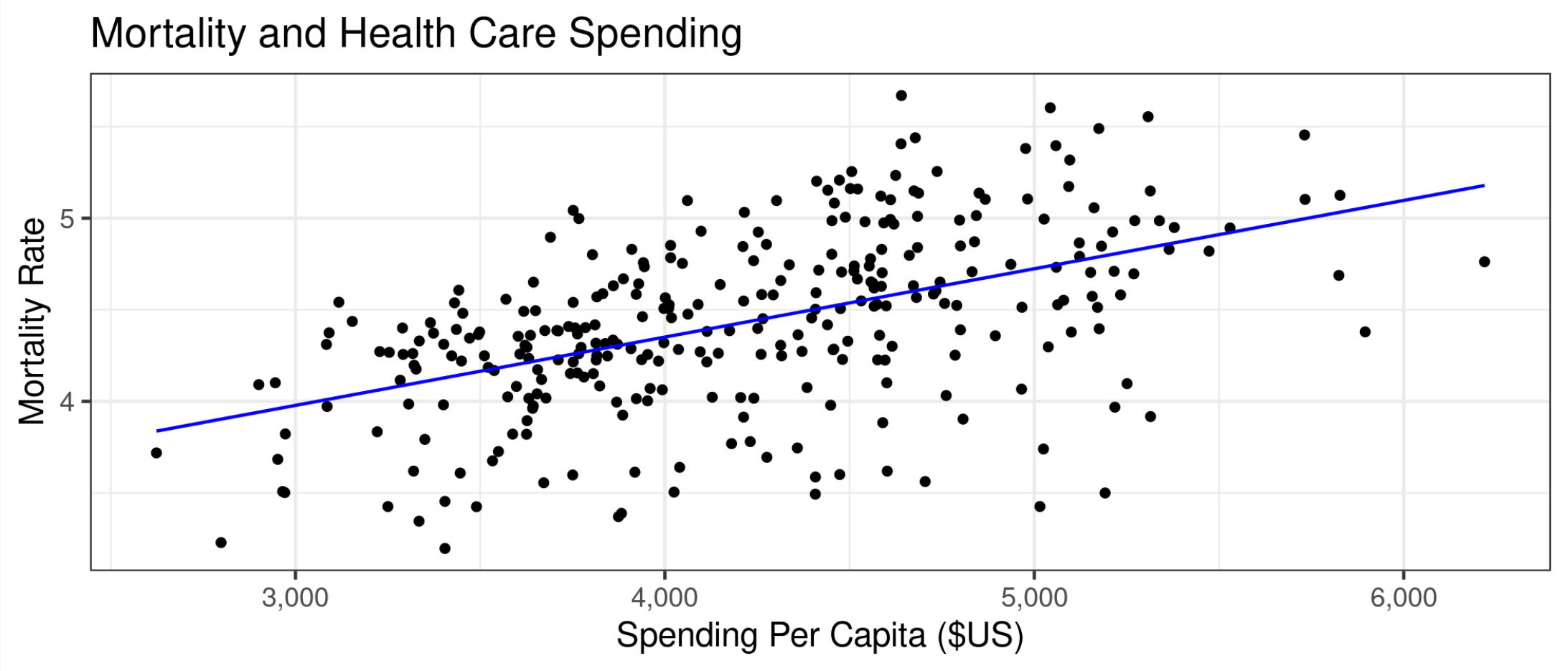
Does health care spending improve health?

Spending and Health

```
ggplot(data = (dartmouth.data %>% filter(Year==2015)),
       mapping = aes(x = Expenditures, y = Total_Mortality)) +
  geom_point(size = 1) + theme_bw() + scale_x_continuous(label = comma) +
  geom_smooth(method="lm", se=FALSE, color="blue", size=1/2) +
  labs(x = "Spending Per Capita ($US)",
       y = "Mortality Rate",
       title = "Mortality and Health Care Spending")
```

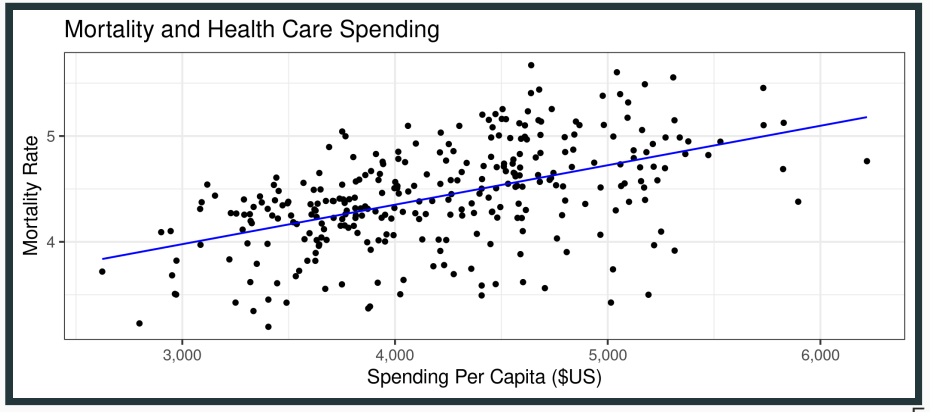


Spending and Health



Spending and Health

- Does medical spending make us sicker?
- What else might explain this relationship?



Goals of this course.

1. Understand and implement selected methods for causal inference
2. Along the way...data management and version control with real data
3. Summarize, visualize, and explain research results

Syllabus highlights

(Read the full document here.)

Why this course?

1. Major problems that need solutions

2. Need good, convincing empirical work for policy

3. Working with data is hard, particularly health care data

4. Your work should be transparent and reproducible

Structure

- Very applied in nature
- *Methods* for causal inference
  - Selection on observables (regression, re-weighting, matching, propensity scores)
  - Instrumental variables
  - Regression discontinuity
  - Difference-in-differences

Structure

- *Substantive* areas
  - Hospital pricing, policy, and competition
  - Cigarette taxes and demand
  - Medicare Advantage and quality disclosure
  - Medicaid expansion and health insurance

Structure

- *Datasets* from the real world
  - Hospital Cost Report Information System (HCRIS)
  - Centers for Disease Control (CDC)
  - Medicare Advantage data
  - Behavioral Risk Factor Surveillance System (BRFSS), Medicaid, Health Insurance Exchanges

Assignments

- Homework (x5)
- Research project
- Participation

Grading

Component	Weight
5 × homework assignments (11% each)	55%
Research project	40%
Participation	5%

Software Installation

Software Installation

1. Download R
2. Download RStudio
3. Download Git
4. Create an account on GitHub

For help and troubleshooting with Git and GitHub, take a look at Jenny Bryan's http://happygitwithr.com.

Checklist

☑ Do you have the most recent version of R?

```
version$version.string
```

```
## [1] "R version 4.1.2 (2021-11-01)"
```

☑ Do you have the most recent version of RStudio? (The preview version is fine.)

```
RStudio.Version()$version
## Requires an interactive session but should return something like "[1] ‘1.4.1717’"
```

☑ Have you updated all of your R packages?

```
update.packages(ask = FALSE, checkBuilt = TRUE)
```

Checklist

- Open up the shell
- Windows users, make sure that you installed a Bash-compatible version of the shell. If you installed Git for Windows, then you should be good to go.

Checklist

☑ Which version of Git have you installed?

```
git --version
```

☑ Did you introduce yourself to Git? (Substitute in your details.)

```
git config --global user.name 'Ian McCarthy'
git config --global user.email 'ian.mccarthy@emory.edu'
git config --global --list
```

☑ Did you register an account in GitHub?

Altnernative setup...

Just use the cloud!

- We have our own virtual computer via AWS
- This computer has all the space you need and the data are already available
- Downside: the computer will be "on" during designated times only
- Details of this are on Canvas

Practice with Git and RStudio

Before next class (see http://happygitwithr.com)

1. Download R
2. Download RStudio
3. Download Git
4. Create an account on GitHub
5. Connect RStudio to Git and GitHub
6. Start/clone/fork a repository for this class

Setting things up

Now we're going to clone a GitHub repository (repo) using RStudio. The video below is from Grant McDermott's class.

Data Science for Economists

Lecture 2: Version control with Git(Hub)

Some common mistakes for windows users

- Windows folders are *not* files...there is no content without a file. You can't commit or push changes without content.
- Let RStudio/GitHub create the directory (main folder) for you.
- If you're working across devices on your own repo, be sure to pull before starting and push afterward.
- Avoid spaces in file names. Avoid them at all costs. *DO NOT PUT SPACES IN YOUR FILE NAMES.*

> *"A space in a file name is a space in your soul."*

Ideal workflow

Until you are a Git(Hub) expert...

1. Start project on GitHub (fork from another repo if needed)

2. Clone to desktop with RStudio

3. See http://happygitwithr.com for instructions on linking your local repo with a new upstream remote

Working with AWS

Let's do this live!