Module 0: Introduction

Ian McCarthy | Emory University

Econ 771

Table of contents

Introductions

Mad libs ice breaker

We need some ideas:

- name
- area of study
- book
- food
- fear
- liquid
- gift
- area of study

Mad libs

(name) was excited to start school! Today, they were going to learn about *(area of study)*. They stayed up all night reading *(book)* and eating *(food)*. Now they are tired. They fall asleep in class, and dream about *(fear)*. They are awoken when the professor throws *(liquid)* at their desk. Feeling bad for sleeping, they give the professor a *(gift)*, apologize, and are careful to never sleep in class again. Now everyone is happy to learn more about *(area of study)*.

Motivation

Health and Healthcare

Everything hurts and I'm dying.

Health and Healthcare

We've made *major* improvements in life expectancy (and many other measures of health) across the world

- Poverty reduction
- Technology development and innovation
- Technology diffusion and adoption
- Access to better services, including healthcare

Evidence of better health

```r
library(gapminder)
gapminder %>%
  group_by(year) %>%
  summarize(lifeExp = median(lifeExp),
            gdpMed = median(gdpPercap)) %>%
  ggplot(aes(year,lifeExp)) + geom_line(alpha = 1/3) + theme_bw() +
    labs(x = "Year",
         y = "Life Expectancy (years)",
         title = "Median life expectancy across the world")
```
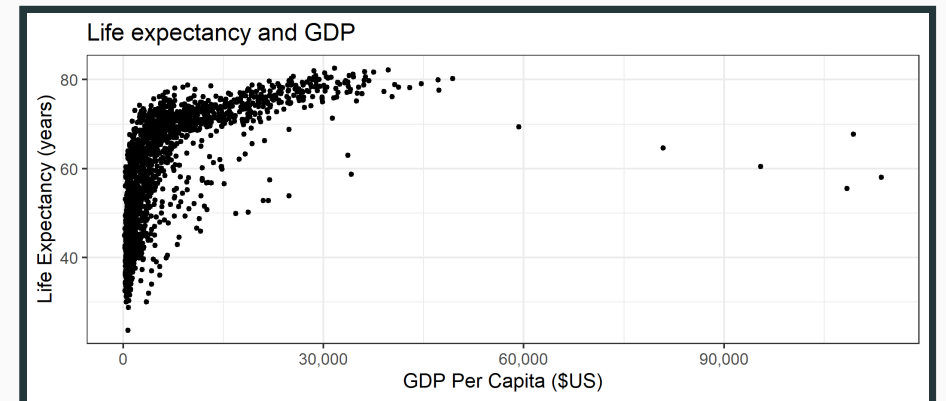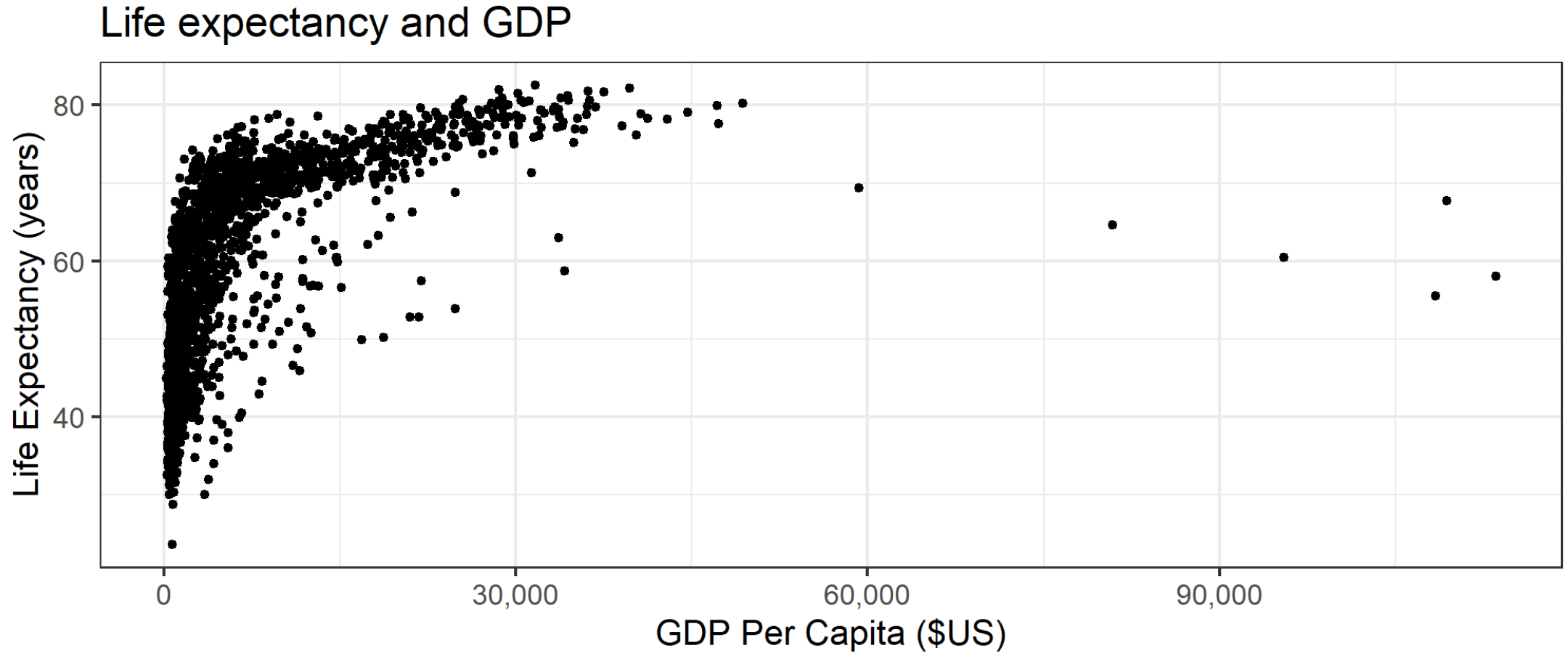
Evidence of better health

Median life expectancy across the world
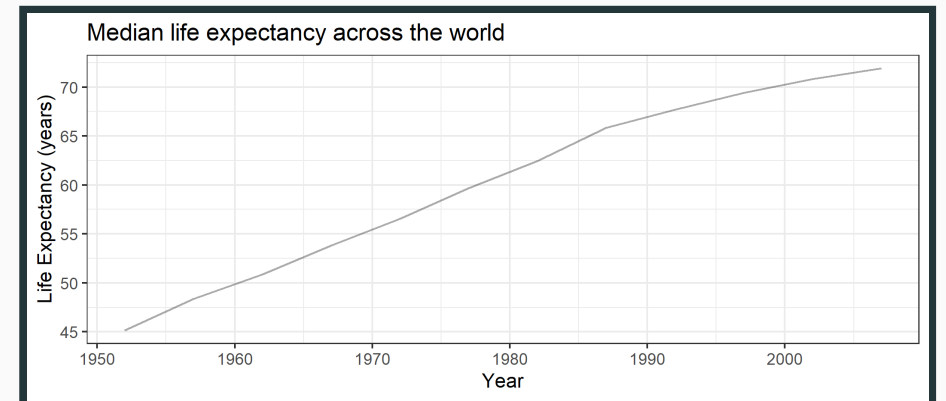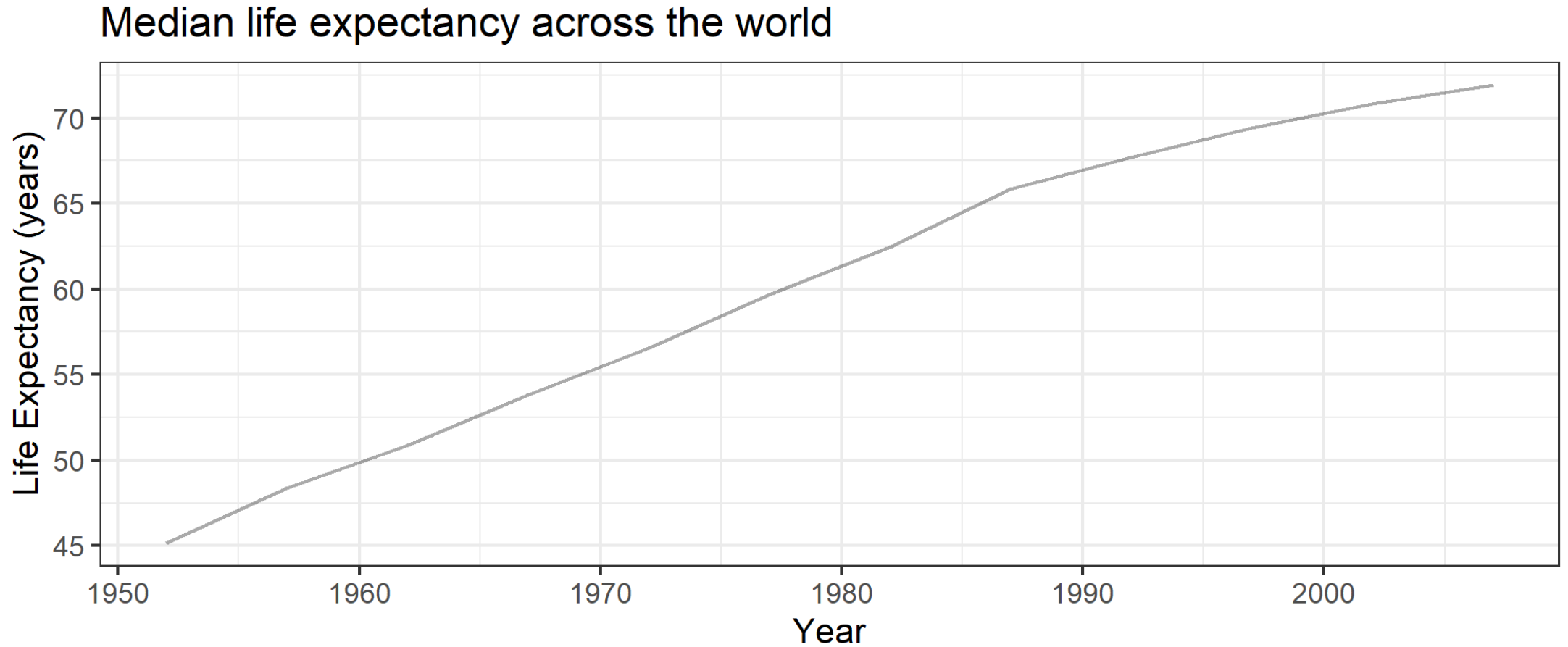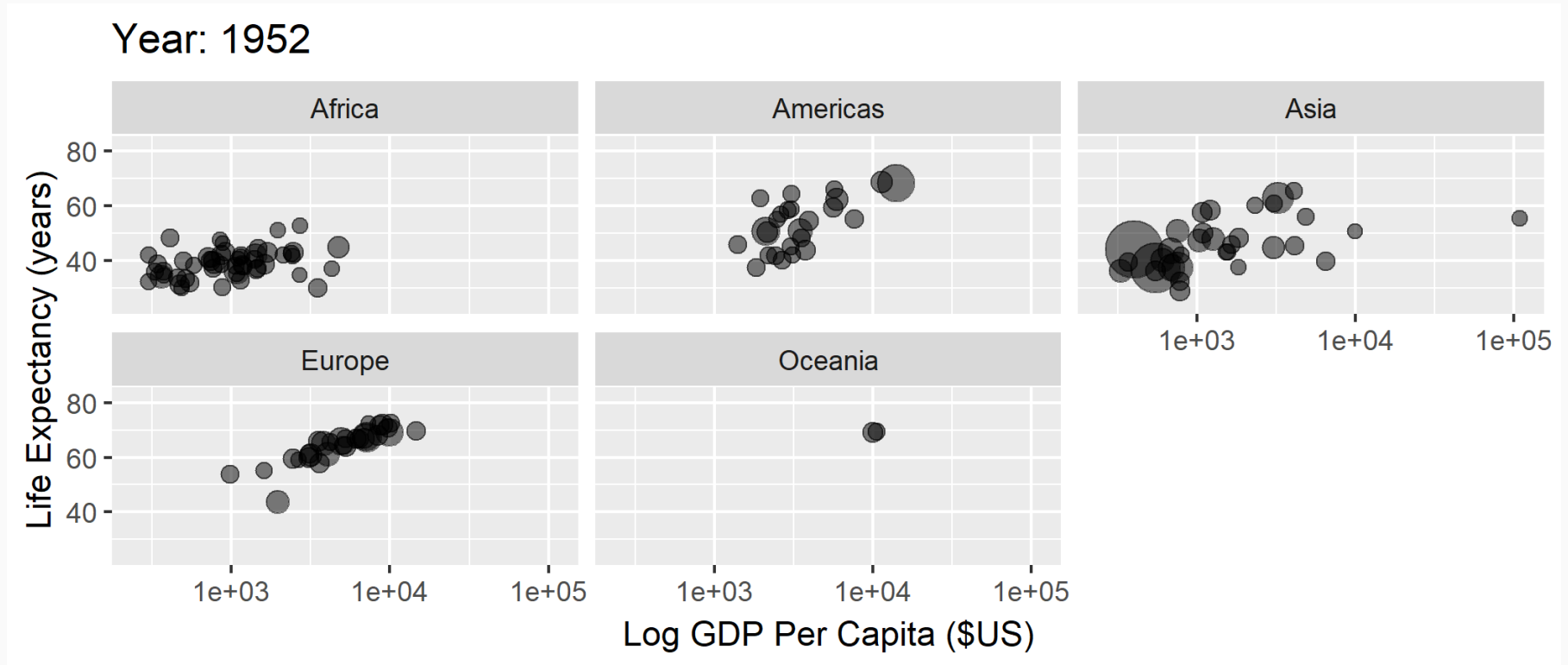
GDP and Health

```r
library(gapminder)
ggplot(data = gapminder, mapping = aes(x = gdpPercap, y = lifeExp)) +
    geom_point(size = 1) + theme_bw() + scale_x_continuous(label = comma) +
    labs(x = "GDP Per Capita ($US)",
         y = "Life Expectancy (years)",
         title = "Life expectancy and GDP")
```
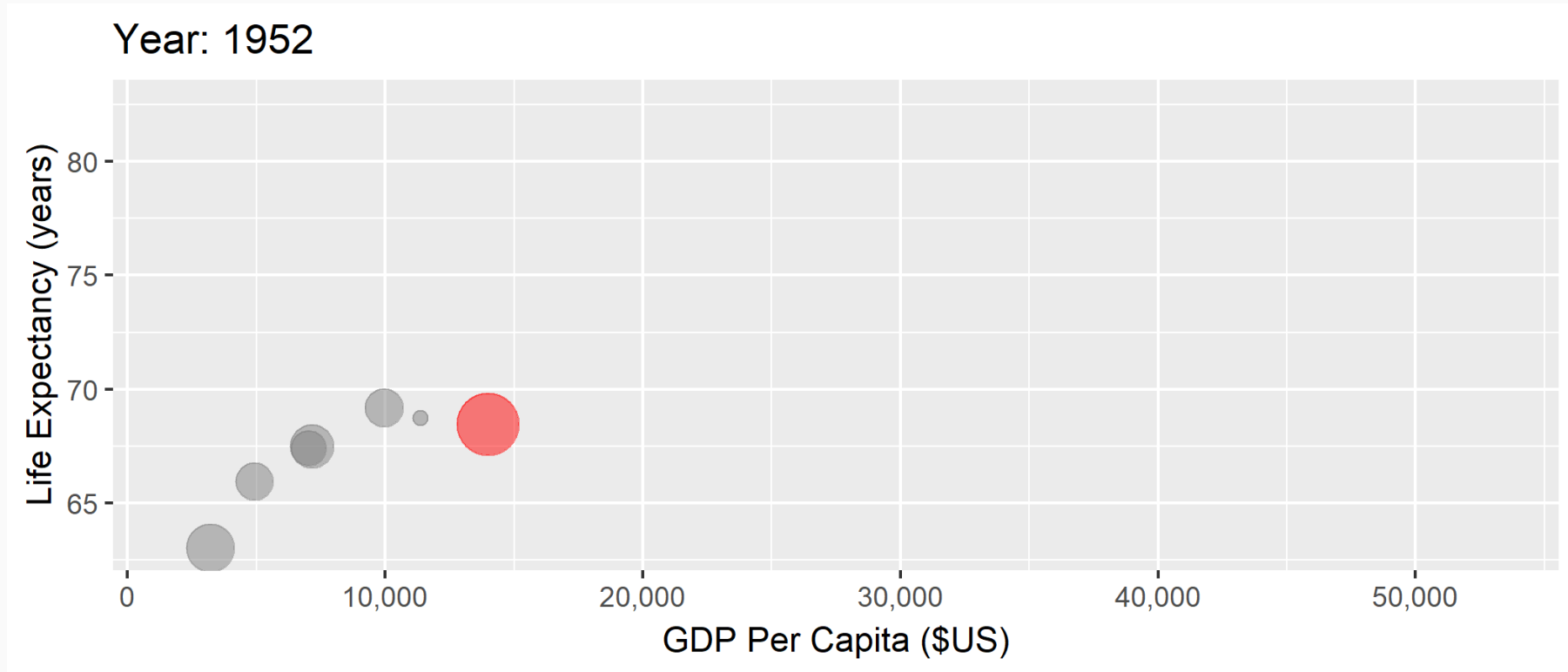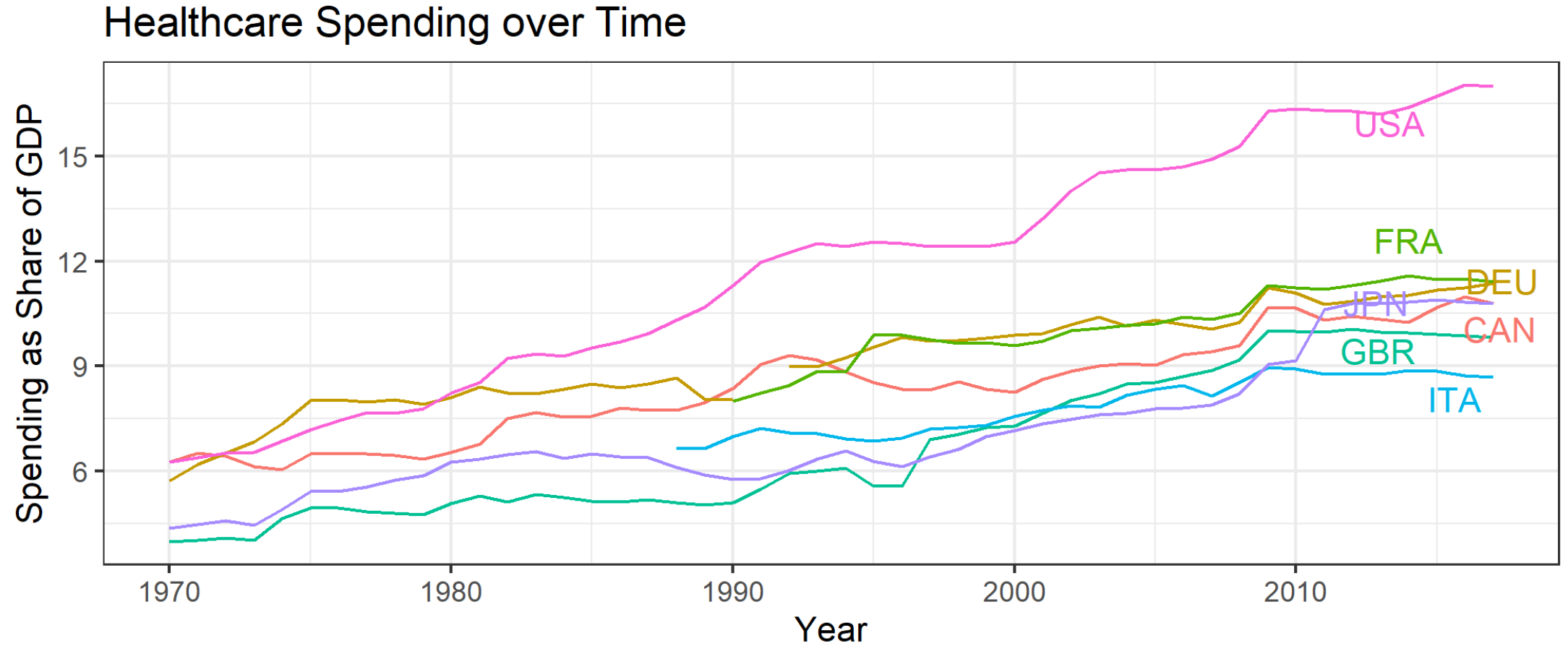
GDP and Health

Life expectancy and GDP

GDP and Health over Time

But the US is unique

Healthcare spending
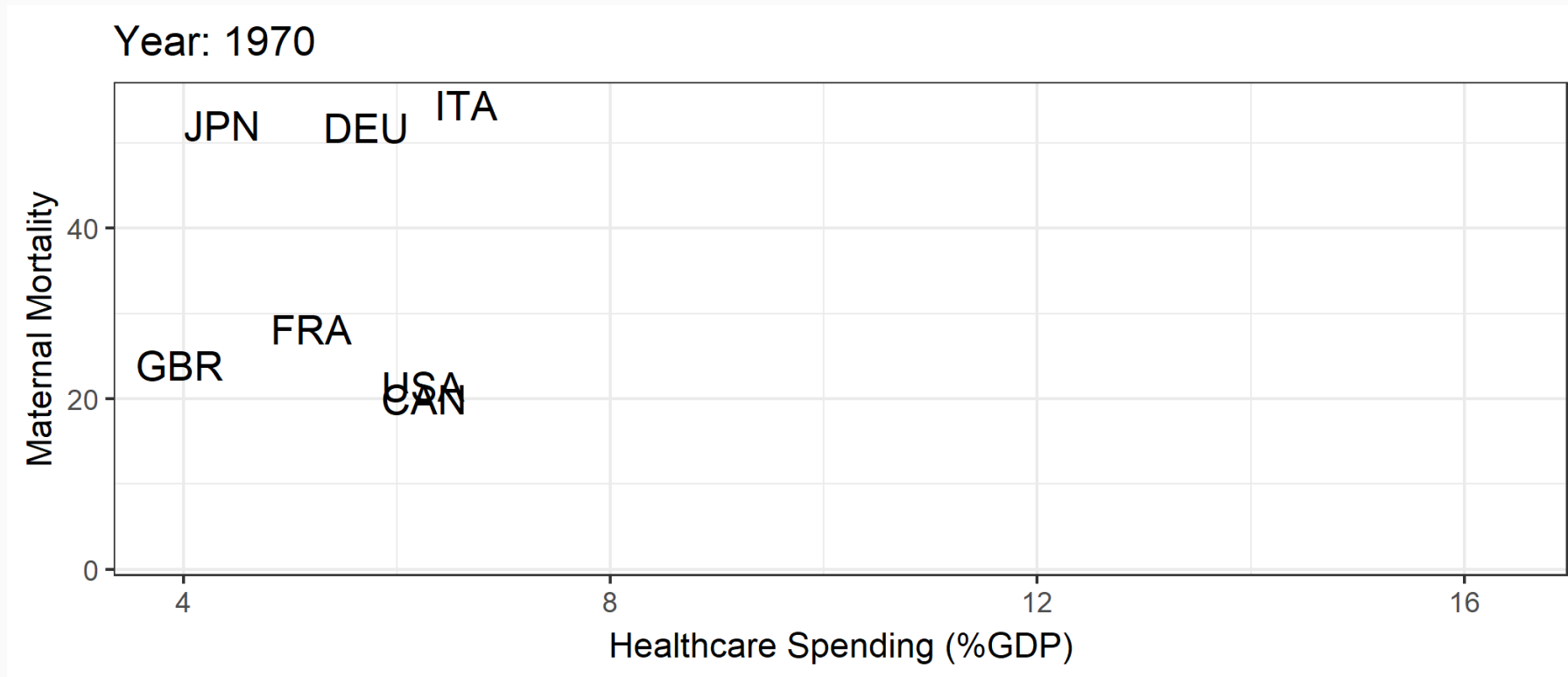

Healthcare Spending over Time

Spending and infant mortality
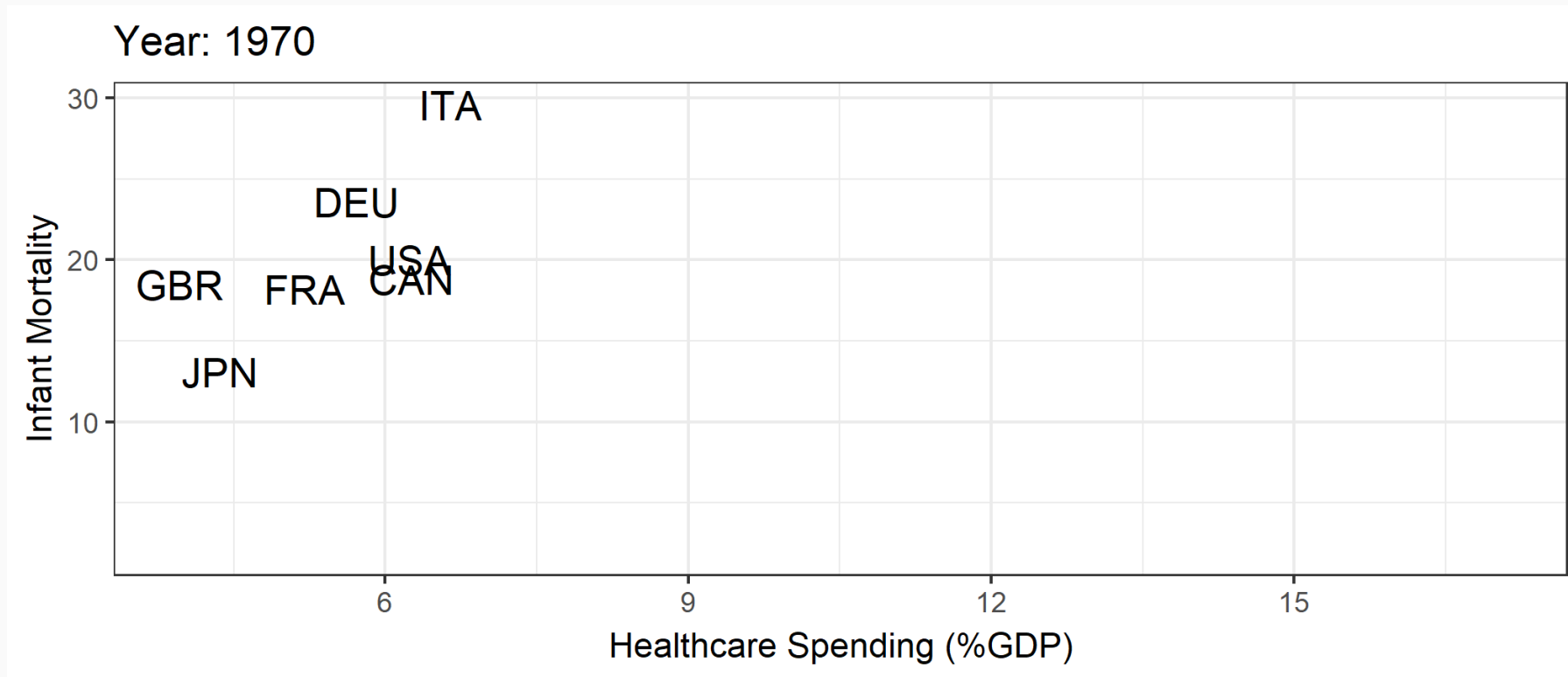
Spending and maternal mortality

Year: 1970

Why study healthcare

1. Health and healthcare are constantly changing

2. $3.5 trillion, 17.9% of GDP, $10,739 per person

3. Coronavirus shows interconnected web of economics, health policy, public policy, public health

4. U.S. healthcare is uniquely inefficient in many ways

Economic issues in healthcare

Lots of interesting economic issues in healthcare (not all unique to the U.S.):

1. Extremely heterogeneous products

2. Asymmetric information between patients and physicians

3. Unobservable quality (experience good)

4. Unpredictable need (inability to shop in many cases)

5. Distortion of incentives due to insurance

6. Adverse selection (asymmetric information between patients and insurers)

What about this class specifically?

1. Overlap with industrial organization and healthcare

2. Data challenges and opportunities

3. Good blend of microeconomic theory and applied empirical micro

Syllabus highlights

(Read the full document here)

Structure

- *Substantive* areas

  - Hospital ownership and production

  - Physician agency

  - Information disclosure and choice

  - Competition and industrial organization

Structure

- *Applying* canonical methods for causal inference

  - Instrumental variables

  - Regression discontinuity

  - Difference-in-differences

Structure

- *Datasets* from the real world

    - Hospital Cost Report Information System (HCRIS)

    - Inpatient Prospective Payment System (IPPS) Final Rule Files

    - Provider of Services (POS) files

    - Provider Utilization and Payment Public Use Files (PUF)

    - Medicare Advantage Insurance Data

Structure

- Practical *workflow* for replicable work

  - Git and GitHub

  - Practice with cloud computing

  - Extracting and managing raw data

Assignments

- Empirical exercises (x4)

- Presentation (x2)

- Lit review or draft paper

- Replication

- Discussion

Grading

Component	Weight
Online discussions	5%
4 × empirical exercies (5% each)	20%
2 x presentations (10% each)	20%
Lit review/draft paper	25%
Replication	30%

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 3.6.0 (2019-04-26)"
```

☑ 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] '1.1.463'"
```

☑ 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?
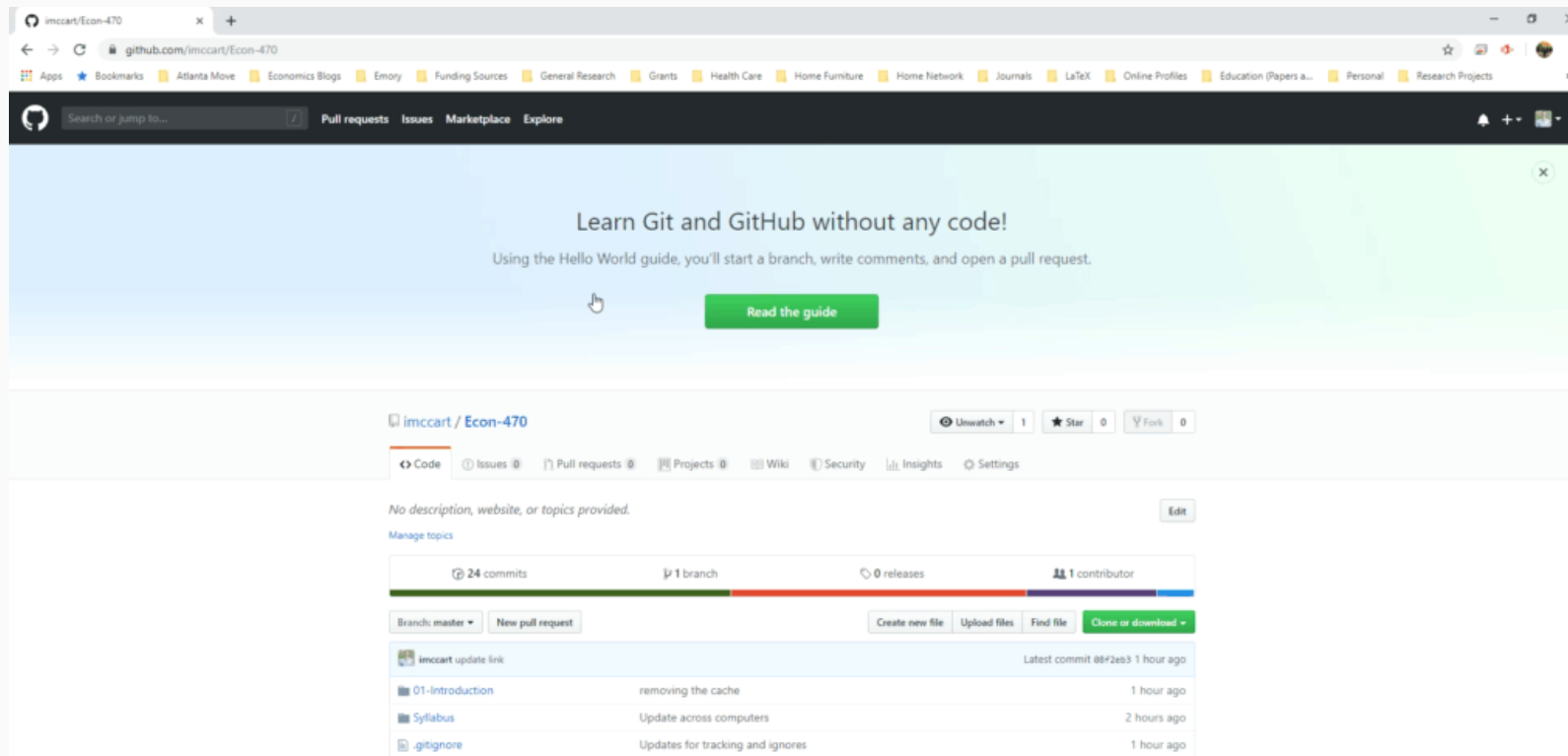
Practice with Git and RStudio

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.

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