

R: Introduction and Review

EC 421, Set 1(r)

Edward Rubin

Econometrics

Recall: Applied econometrics, data science, analytics require:

1. Intuition for the **theory** behind statistics/econometrics (assumptions, results, strengths, weaknesses).
2. Practical knowledge of how to **apply theoretical methods** to data.
3. Efficient methods for **working with data** (cleaning, aggregating, joining, visualizing).

This course aims to deepen your knowledge in each of these three areas.

- 1: As before.
- 2–3: R

R

What is R?

To quote the [R project website](#):

R is a free software environment for statistical computing and graphics. It compiles and runs on a wide variety of UNIX platforms, Windows and MacOS.

What is R?

To quote the [R project website](#):

R is a free software environment for statistical computing and graphics. It compiles and runs on a wide variety of UNIX platforms, Windows and MacOS.

What does that mean?

- R was created for the statistical and graphical work required by econometrics.
- R has a vibrant, thriving online community. ([stack overflow](#))
- Plus it's **free** and **open source**.

Why are we using R?

1. R is **free** and **open source**—saving both you and the university 💰💰💰.
2. *Related:* Outside of a small group of economists, private- and public-sector **employers favor R** over **Stata** and most competing softwares.
3. R is very **flexible and powerful**—adaptable to nearly any task, *e.g.*, 'metrics, spatial data analysis, machine learning, web scraping, data cleaning, website building, teaching (these slides).

Why are we using R?

4. *Related*: R imposes **no limitations** on your amount of observations, variables, memory, or processing power. (I'm looking at **you, Stata.**)

5. If you put in the work,[†] you will come away with a **valuable and marketable** tool.

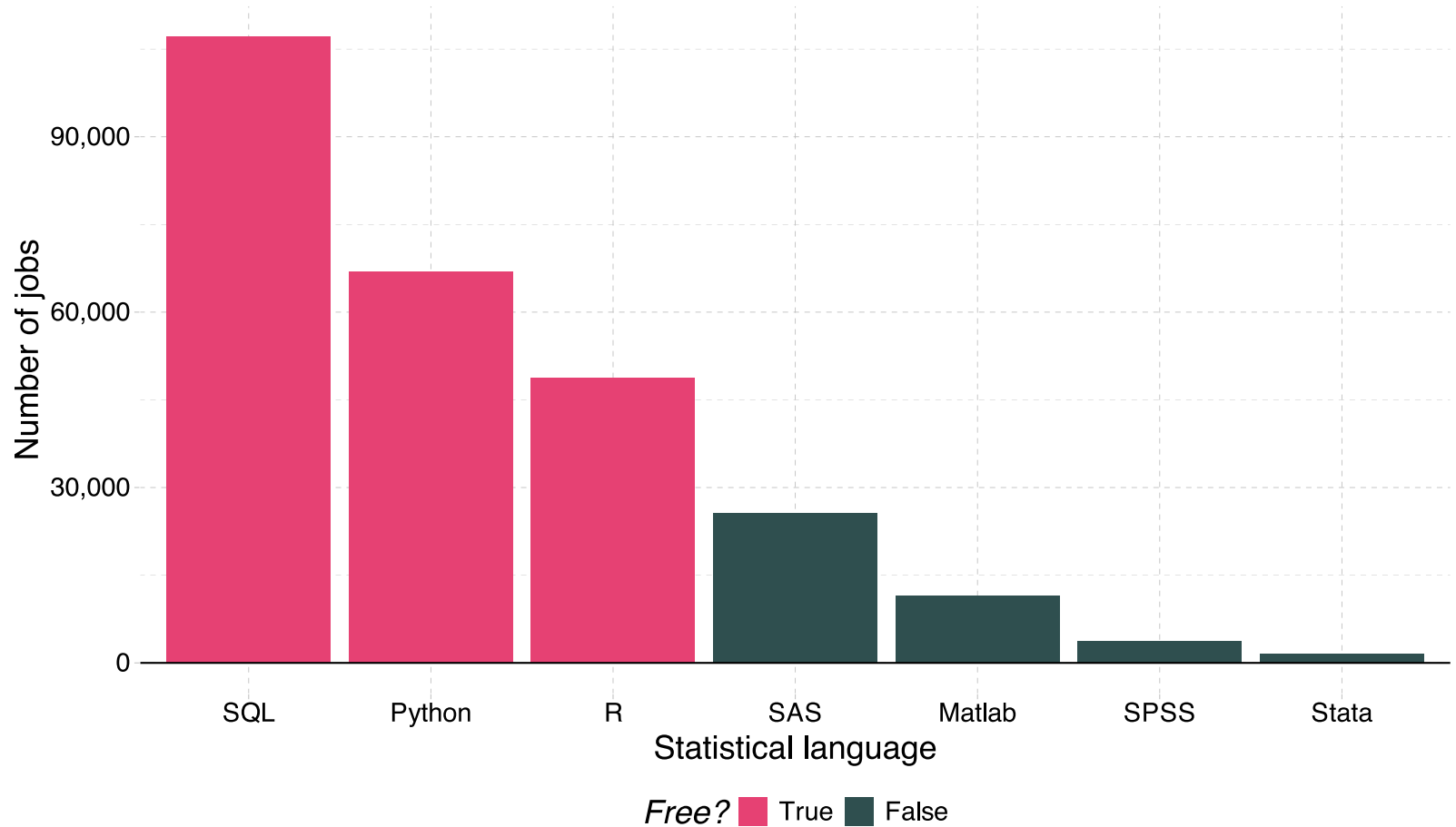
6. I  R

7. R is a nice gateway to (and plays well with) other programming languages (*e.g.*, Python, SQL, C++, JavaScript).

[†]: Learning R definitely requires time and effort.

Comparing statistical languages

Number of job postings on Indeed.com, 2019/01/06



R + Examples

R + Regression

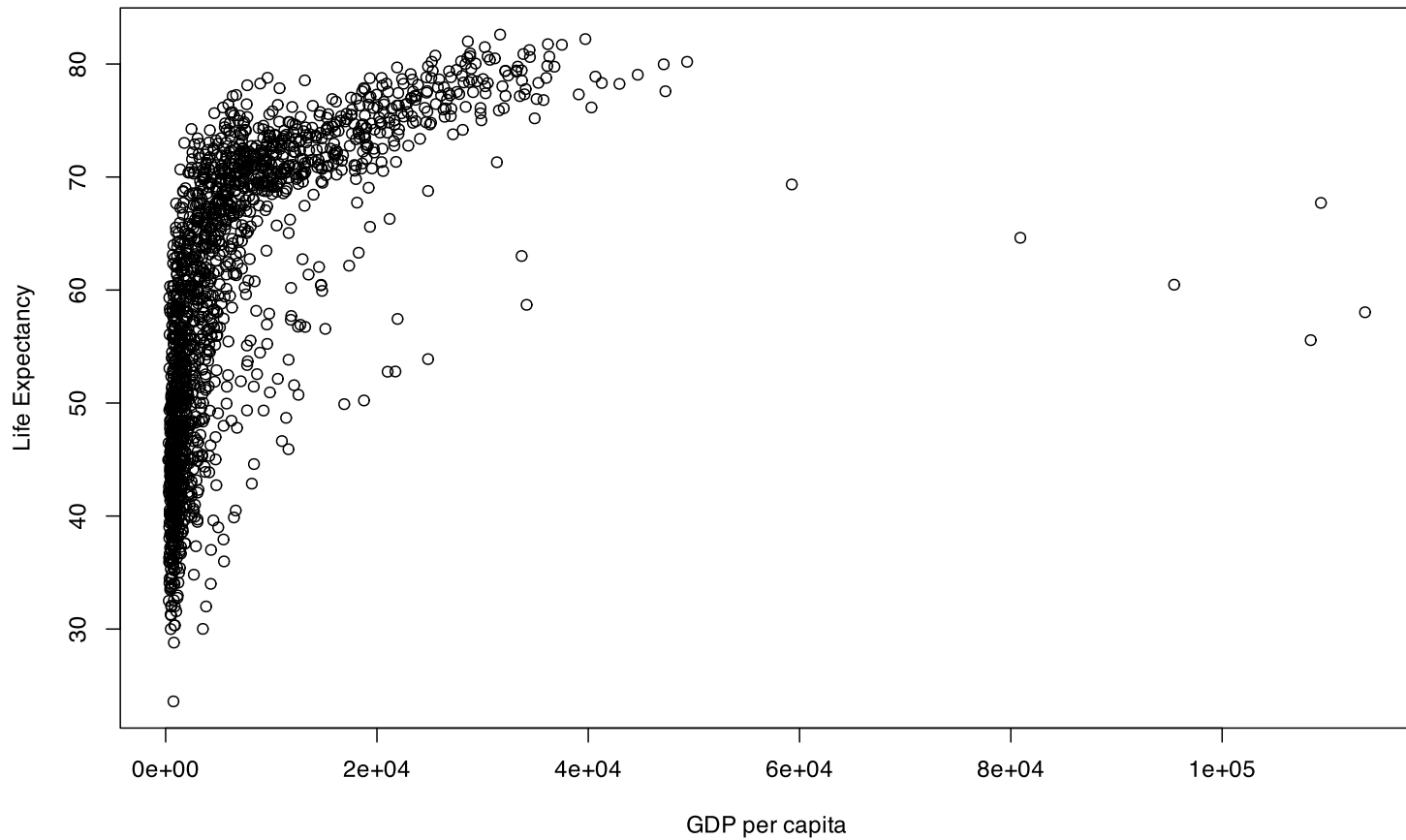
```
# A simple regression  
fit ← lm(dist ~ 1 + speed, data = cars)  
# Show the coefficients  
coef(summary(fit))
```

```
#>               Estimate Std. Error   t value    Pr(>|t|)  
#> (Intercept) -17.579095   6.7584402 -2.601058 1.231882e-02  
#> speed        3.932409    0.4155128  9.463990 1.489836e-12
```

```
# A nice, clear table  
library(broom)  
tidy(fit)
```

```
#> # A tibble: 2 × 5  
#>   term          estimate std.error statistic  p.value  
#>   <chr>          <dbl>     <dbl>     <dbl>    <dbl>  
#> 1 (Intercept)   -17.6       6.76      -2.60 1.23e- 2  
#> 2 speed          3.93      0.416      9.46 1.49e-12
```

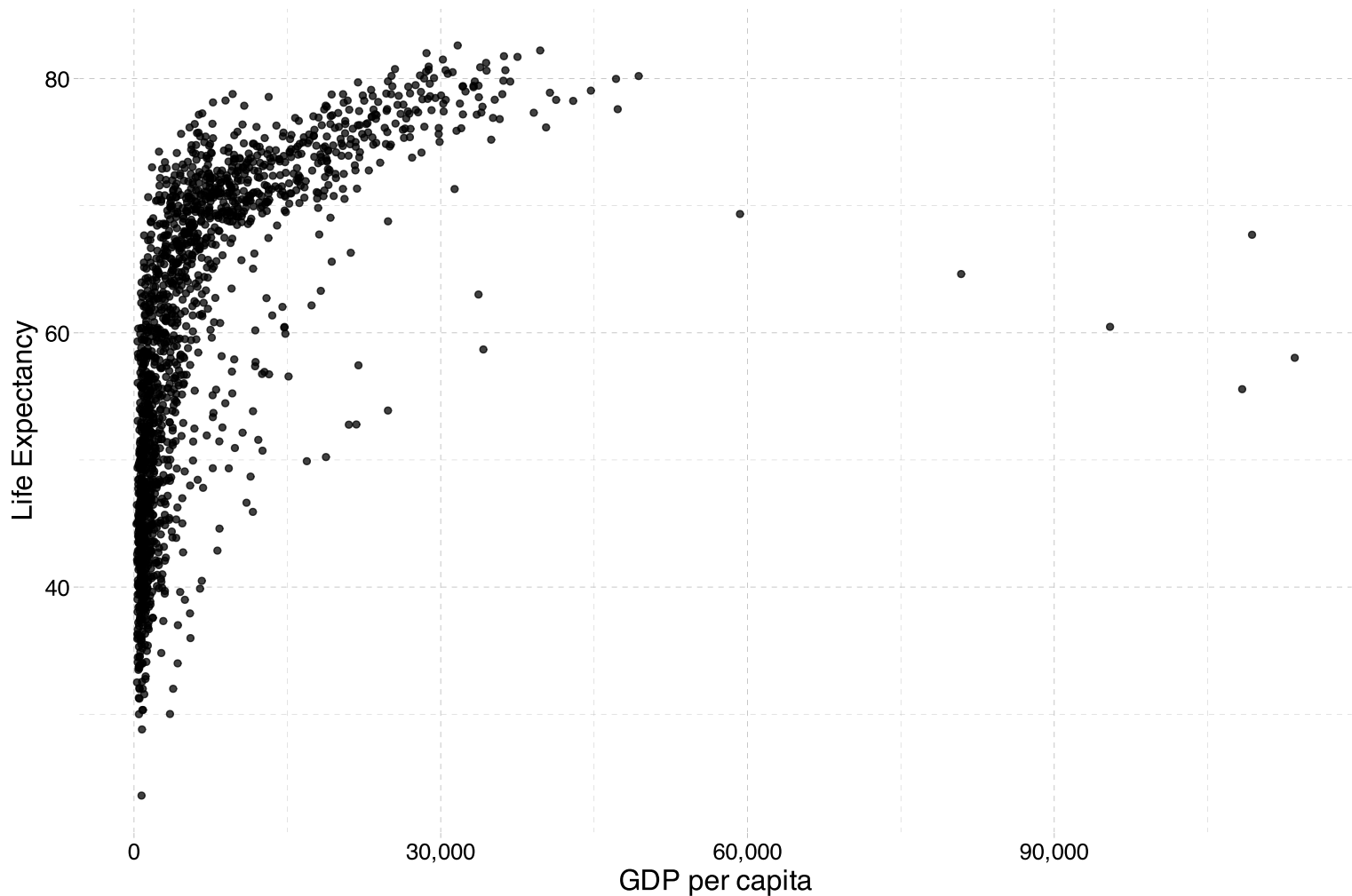
R + Plotting (w/ plot)



R + Plotting (w/ plot)

```
# Load packages with dataset  
library(gapminder)  
  
# Create dataset  
plot(  
  x = gapminder$gdpPercap, y = gapminder$lifeExp,  
  xlab = "GDP per capita", ylab = "Life Expectancy"  
)
```

R + Plotting (w/ ggplot2)



R + Plotting (w/ ggplot2)

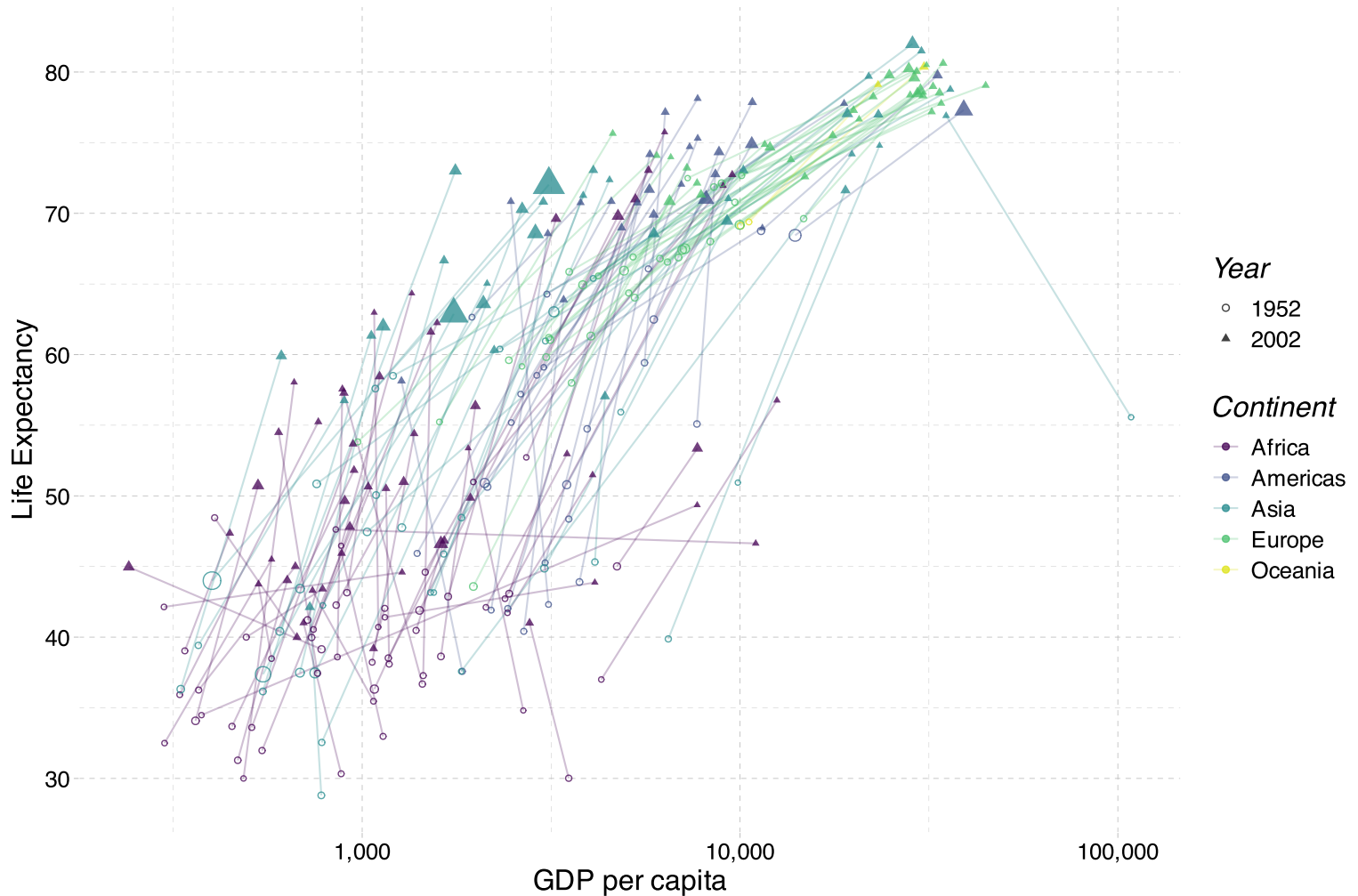
```
# Load packages
```

```
library(gapminder); library(dplyr)
```

```
# Create dataset
```

```
ggplot(data = gapminder, aes(x = gdpPercap, y = lifeExp)) +  
geom_point(alpha = 0.75) +  
scale_x_continuous("GDP per capita", label = scales::comma) +  
ylab("Life Expectancy") +  
theme_pander(base_size = 16)
```

R + More plotting (w/ ggplot2)



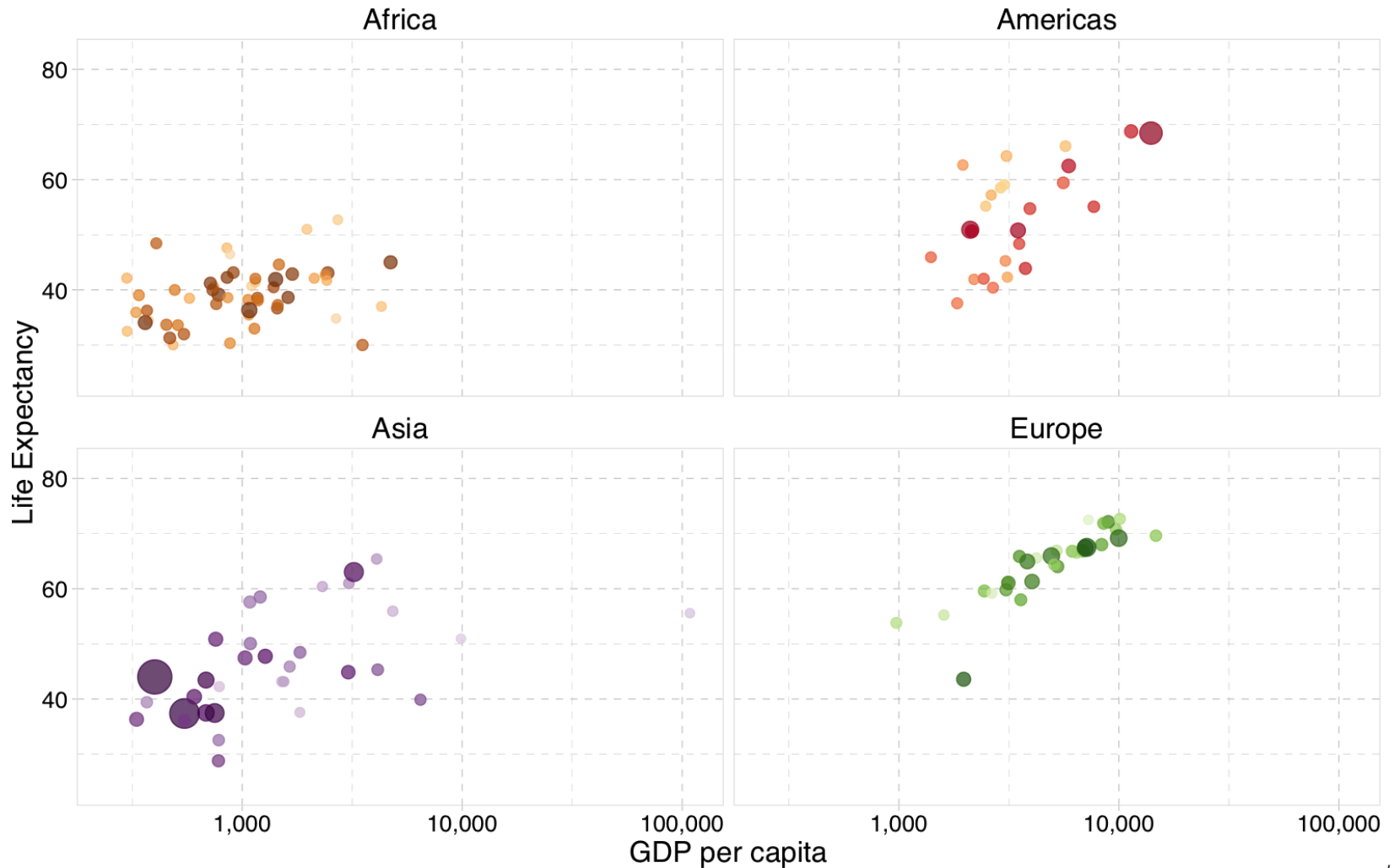
R + More plotting (w/ ggplot2)

```
# Load packages
library(gapminder); library(dplyr)

# Create dataset
ggplot(
  data = filter(gapminder, year %in% c(1952, 2002)),
  aes(x = gdpPercap, y = lifeExp, color = continent, group = country)
) +
  geom_path(alpha = 0.25) +
  geom_point(aes(shape = as.character(year), size = pop), alpha = 0.75) +
  scale_x_log10("GDP per capita", label = scales::comma) +
  ylab("Life Expectancy") +
  scale_shape_manual("Year", values = c(1, 17)) +
  scale_color_viridis("Continent", discrete = T, end = 0.95) +
  guides(size = F) +
  theme_pander(base_size = 16)
```


R + Animated plots (w/ gganimate)

Year: 1952

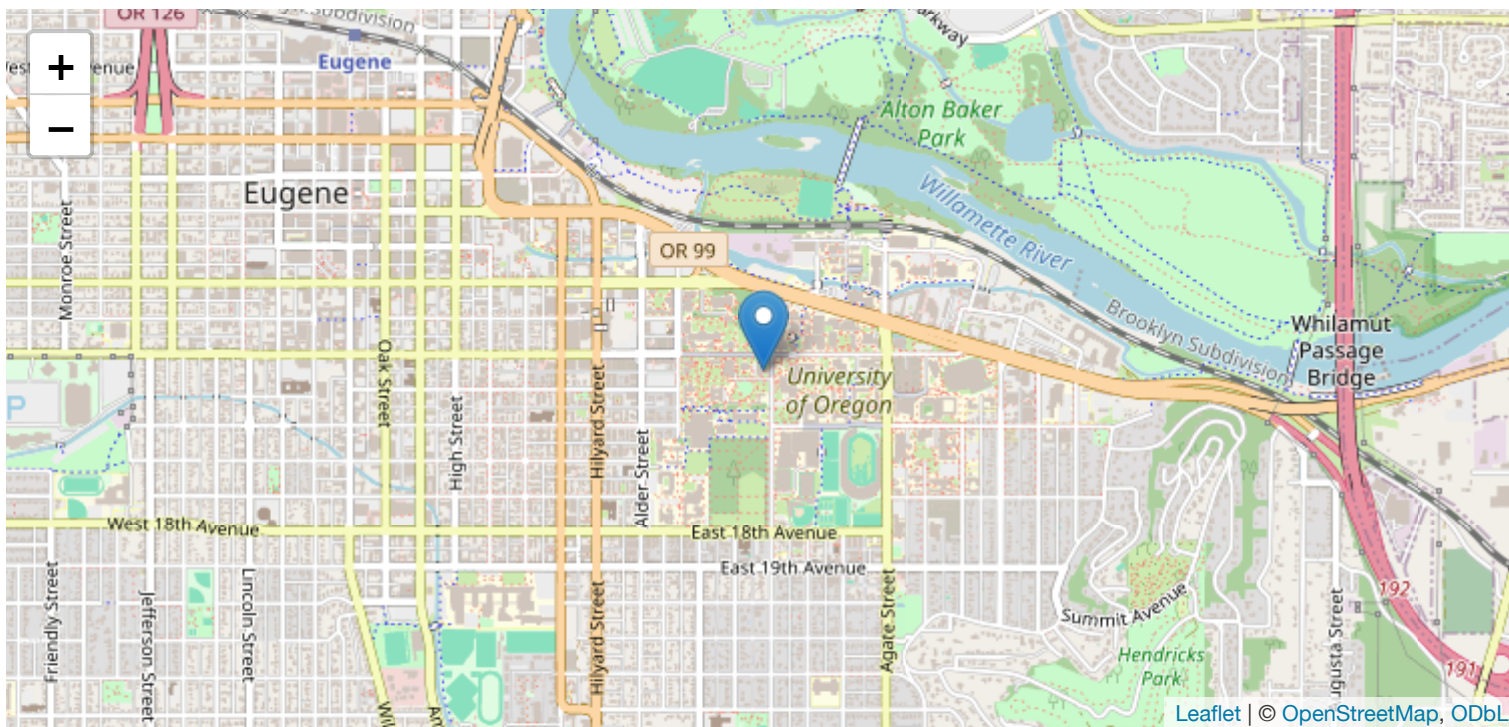


R + Animated plots (w/ gganimate)

```
# The package for animating ggplot2
library(gganimate)
# As before
ggplot(
  data = gapminder %>% filter(continent ≠ "Oceania"),
  aes(gdpPercap, lifeExp, size = pop, color = country)
) +
geom_point(alpha = 0.7, show.legend = FALSE) +
scale_colour_manual(values = country_colors) +
scale_size(range = c(2, 12)) +
scale_x_log10("GDP per capita", label = scales::comma) +
facet_wrap(~continent) +
theme_pander(base_size = 16) +
theme(panel.border = element_rect(color = "grey90", fill = NA)) +
# Here comes the gganimate-specific bits
labs(title = "Year: {frame_time}") +
ylab("Life Expectancy") +
transition_time(year) +
ease_aes("linear")
```

R + Maps

```
library(leaflet)
leaflet() %>%
  addTiles() %>%
  addMarkers(lng = -123.075, lat = 44.045, popup = "The University of Oregon")
```



Getting started with R

Starting R

Installation

- Install **R**.
- Install **RStudio**.
- **Optional/Overkill:** **Git**
 - Create an account on **GitHub**
 - Register for a student/educator **discount**.
 - For installation guidance and troubleshooting, check out Jenny Bryan's **website**.
- **Note:** Many UO labs have R installed and ready. That said, having a copy of R on your own computer will likely be very convenient for homework, projects, *etc*.

Starting R

Resources

Free(-ish)

- Google (which inevitably leads to StackOverflow)
- Time
- ChatGPT, Copilot, and other AI assistants
- [Data services at the UO library](#)
- Your classmates
- Your GE and me
- R resources [here](#) and [here](#)
- [swirl](#) and [learnr](#)

Money

Short online courses, e.g., [DataCamp](#)

Starting R

Some R basics

You will dive deeper into R in lab, but here six big points about R:

1. Everything is an **object**.

```
foo
```

2. Every object has a **name** and **value**.

```
foo = 2
```

3. You use **functions** on these objects.

```
mean(foo)
```

4. Functions come in **libraries (packages)**

```
library(dplyr)
```

5. R will try to **help** you.

```
?dplyr
```

6. R has its **quirks**.

```
NA; error; warning
```

Next: Metrics review(s)