

1. Course Introduction

PUBH 6199: Visualizing Data with R, Summer 2026

Xindi (Cindy) Hu, ScD

2026-05-19



Outline for today

- Who?
- How?
- What?
- Why?
- Introduction to {ggplot2}



Meet your instructor



Xindi (Cindy) Hu, ScD

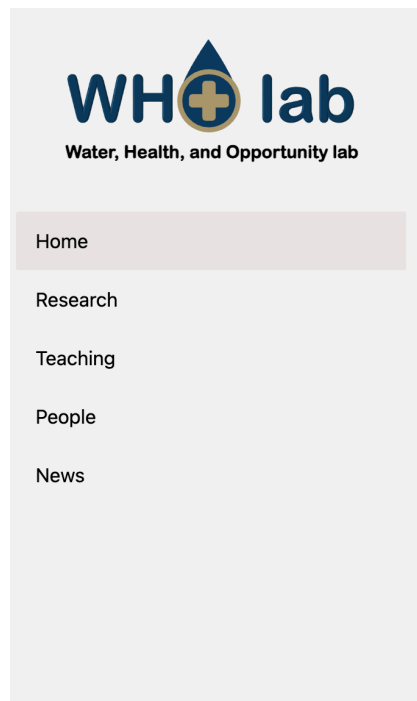
Assistant Professor, Department of Environmental and Occupational Health

ScD in Environmental Health, Harvard University

PUBH 6199: Visualizing Data with R



Water, Health, and Opportunity Lab



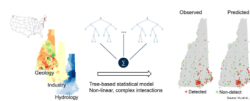
Welcome to the 🙌

Water, Health, Opportunity Lab (WHO Lab)

@ Milken Institute School of Public Health, George Washington University

We study how clean drinking water and other environmental factors impact population health and health disparities. We conduct rigorous and policy-relevant research to generate evidence at a large scale, leveraging transdisciplinary approaches spanning exposure science, geospatial data science, and health informatics.

Our latest projects



MEMCARE

Private drinking water wells are federally unregulated and more likely to be contaminated due to proximity to pollution sources. I developed a machine learning model capable of predicting the risks of chemical contamination in private wells.

Funding for this work is from National Institute of Environmental Health Sciences Superfund Research Program.

Our Research:

Environmental Data Science, Drinking Water Quality, Health Equity, Climate Change, Geospatial Analysis, Machine Learning

Meet your LA



Bogdan Bunea

BEng Candidate

Systems Engineering

GW SEAS

PUBH 6199: Visualizing Data with R



About you



Outline for today

- Who?
- **How?**
- What?
- Why?
- Introduction to {ggplot2}



PUBH 6199 Goals

Theory

- Apply fundamental principles & techniques

Design Skills

- Create, evaluate, and critique visualization designs

Coding Skills

- Implement static and interactive data visualizations

Develop a substantial visualization project!



Course component

- **Lectures:** Tuesdays 3:10-5:10 PM
- **Lab:** Thursdays 3:10-5:10 PM
- **Homework:** Weekly assignments due Mondays
- **Final project:** Team of 2-3 people



What about gradings???





Grading

- **Class Participation** (attendance, contribution to in-class activities, completion of end-of-class surveys): 10%
- **Homework** (4 weekly assignments): 35%
- **Lab** (4 weekly lab notebooks): 15%
- **Final Project** (team of 2): 40%
 - 25% for the final data visualization product
 - 10% for the presentation
 - 5% for peer evaluation



Prerequisites

- Programming experience at the level of PUBH 6131 or similar
- Willingness to learn new software & tools
 - This can be time consuming
 - Learning by doing is the best way of acquiring new skills, get on the bike



Course policies

- Be respectful and inclusive
- Get on the bike
- Don't cheat!



Device policy



- Bring laptop to lecture, lab, and office hour
- Please only use it for in-class activities!



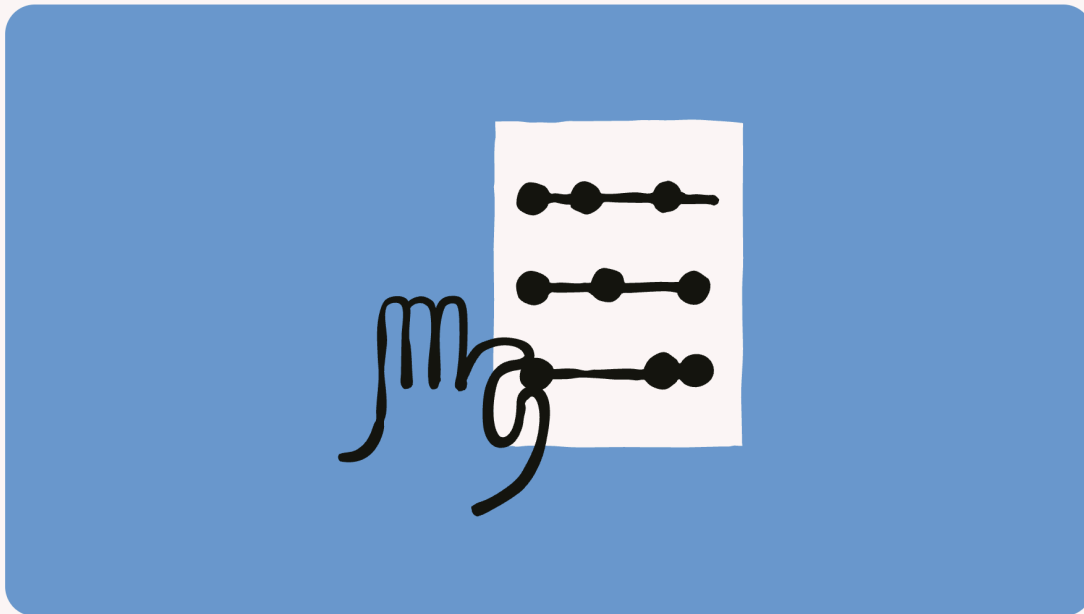
Before we talk about genAI policy

Source: Anthropic

How AI assistance impacts the formation of coding skills

Jan 29, 2026

[Read the paper](#)












A randomized controlled trial.

- Using AI assistance led to a statistically significant decrease in mastery (17% lower)
- Using AI sped up the task slightly, but not statistically significant.
- How someone used AI influenced how much information they retained.

GenAI policy

- Generative AI is another tool in your toolbox, use it but be prepared to be responsible.
- Usage of GenAI tools is permitted but please be transparent about it.
- Lazy usage of GenAI tools (homework prompt -> output -> submission) is **prohibited**.
- Include a “How I used GenAI” section in your homework and final project (include prompt, date, model version, and link to chat history).
- Use **Study Mode ChatGPT**, **Claude Learning Mode**, or **LearnLM in Gemini**

Active Learning Techniques to Use in Study Mode

 <p>Think-Pair-Share Reflect on a topic and discuss ideas</p>	 <p>Muddiest Point Identify areas of confusion and seek clarification</p>	 <p>Speak Out Loud Explain concepts in your own words</p>
 <p>Roleplay Simulate scenarios and practice different perspectives</p>	 <p>Discussion & Debates Engage in dialogue and explore different viewpoints</p>	 <p>Minute Paper/ One-Sentence Summary Summarize key points concisely</p>
 <p>Flipped Classroom Prep Review material and quiz yourself beforehand</p>	 <p>Peer Teaching Teach others to reinforce your own understanding</p>	 <p>Socratic Questioning Answer probing questions to deepen your reasoning</p>

Communication

- Slack: PUBH 6199 channel
- Course website: <https://pubh6199-data-viz-with-r.github.io/2026-Summer/>
- Blackboard
- Office hours:
 - Bogdan: Fridays 1-2pm
 - Cindy: By appointment
- Boundaries:
 - Please allow 24 hrs for slack/email response
 - Replies in the after hours (after 6pm ET) and over the weekends are not guaranteed



Class Mascot

Rubber Duck Debugging 🦆

Explain your problem out loud — as if you're talking to a rubber duck.

- Slows down your thinking
- Reveals skipped steps
- Helps you find mistakes
- Works even without another person!



Outline for today

- Who?
- How?
- **What?**
- Why?
- Introduction to {ggplot2}



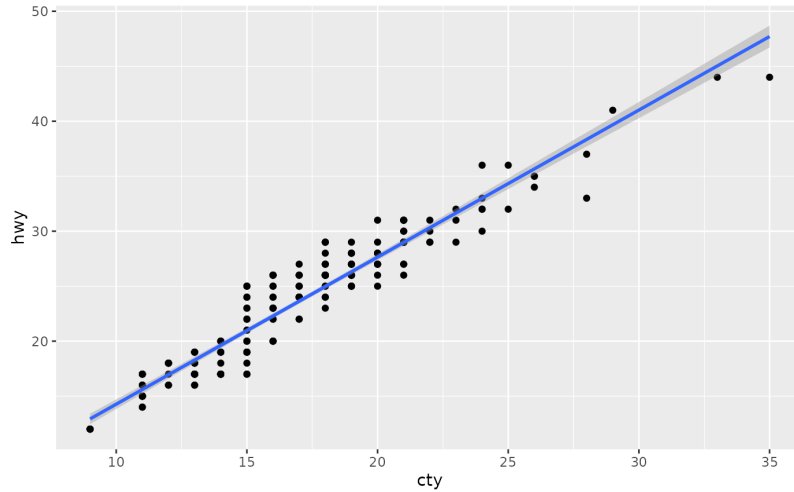
What is data visualization?

“The practice of designing and creating graphic or visual representations of a large amount of complex quantitative and qualitative data and information with the help of static, dynamic or interactive visual items.”

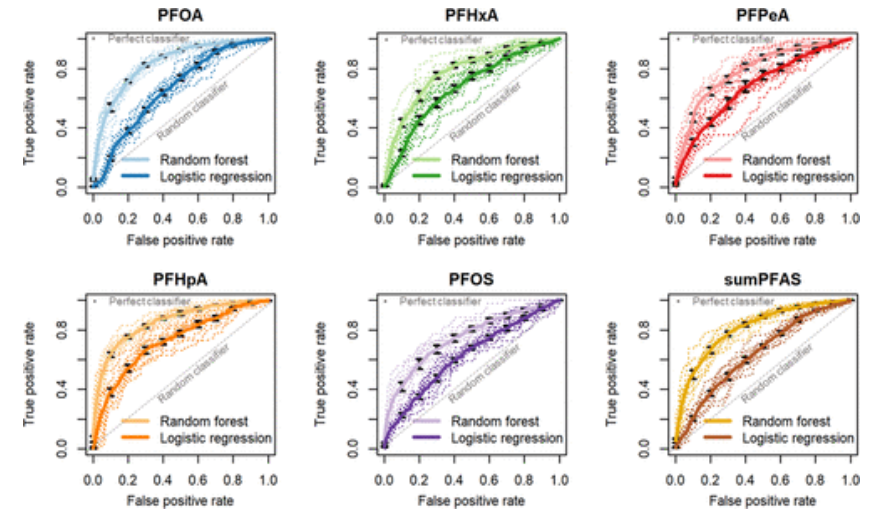
-from [Wikipedia](#)



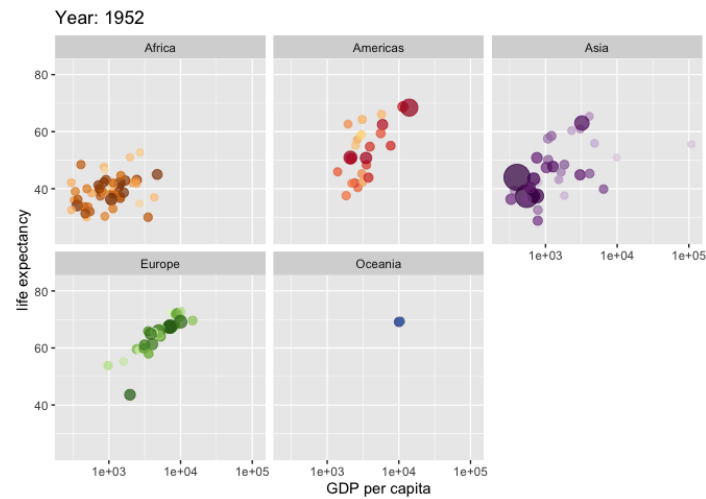
Made with {ggplot2}



Made with {ggplot2} and publication ready

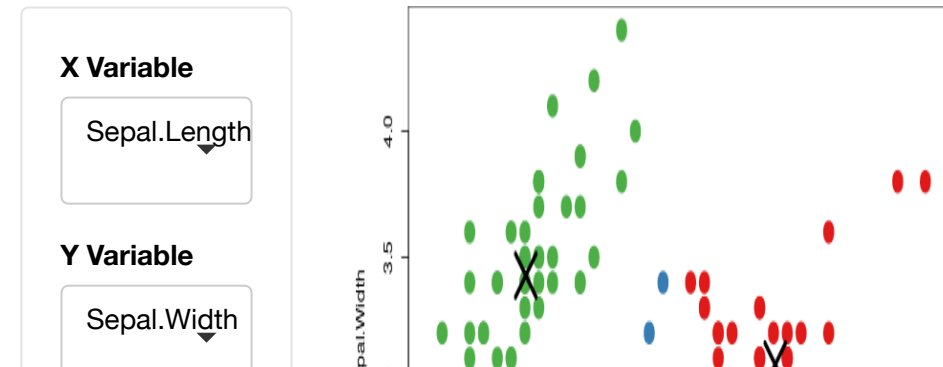


Made with {gganimate}



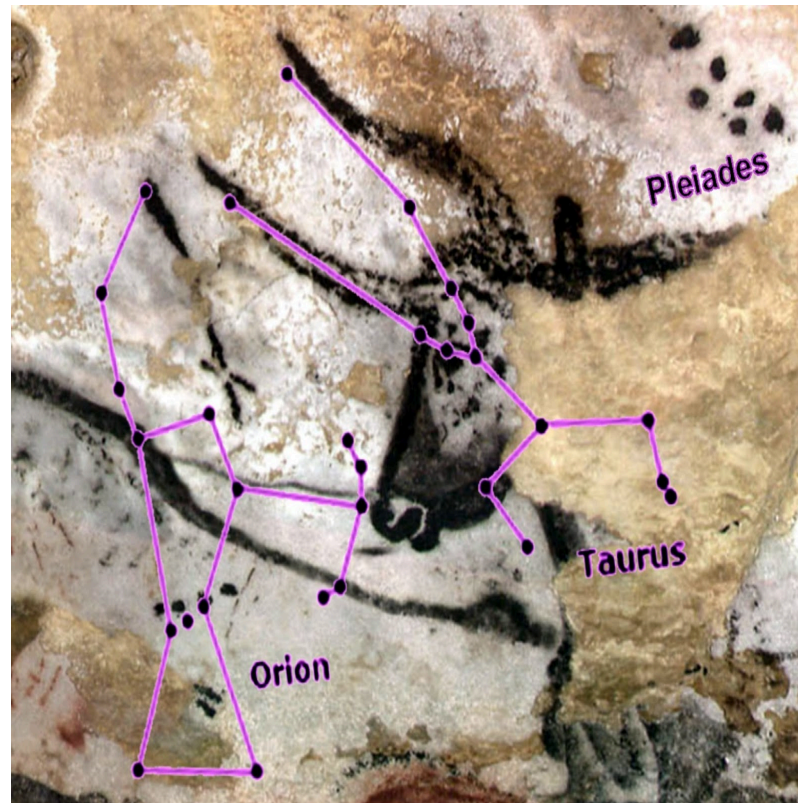
Made with Shiny

Iris k-means clustering



A brief history of Data Visualization (Adapted from [EDS 240](#))

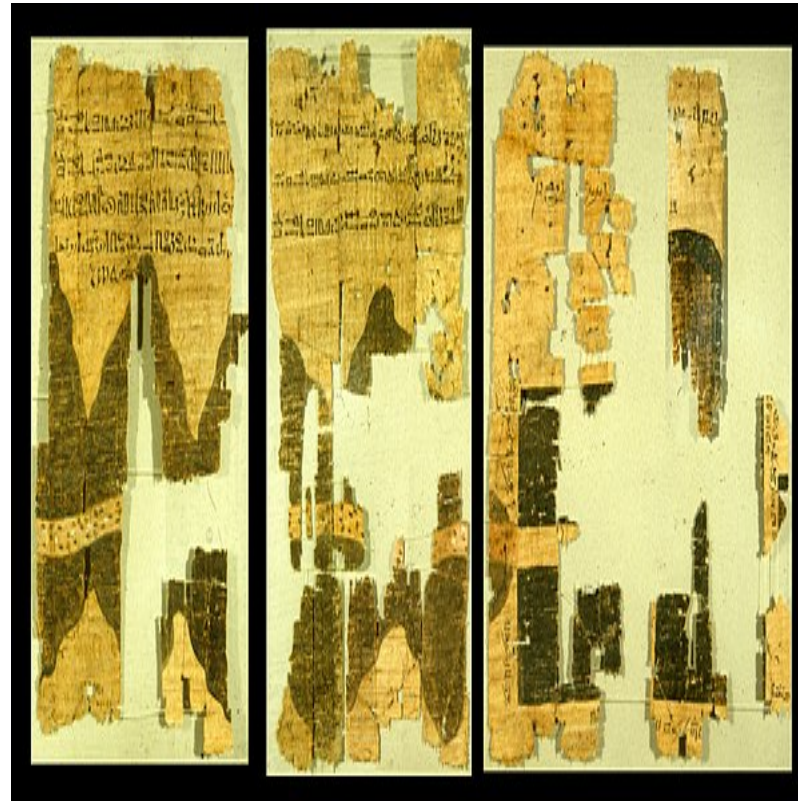
16,500 years ago, Pleistocene



Source: [BBC](#)

A brief history of Data Visualization

~1150 BC, Ancient Egypt



Source: [Wikipedia](#)

A brief history of Data Visualization

1400 - 1532 AD, Inca Empire



Quipus (kee-poos) were recording devices for data collection, census records, calendaring...

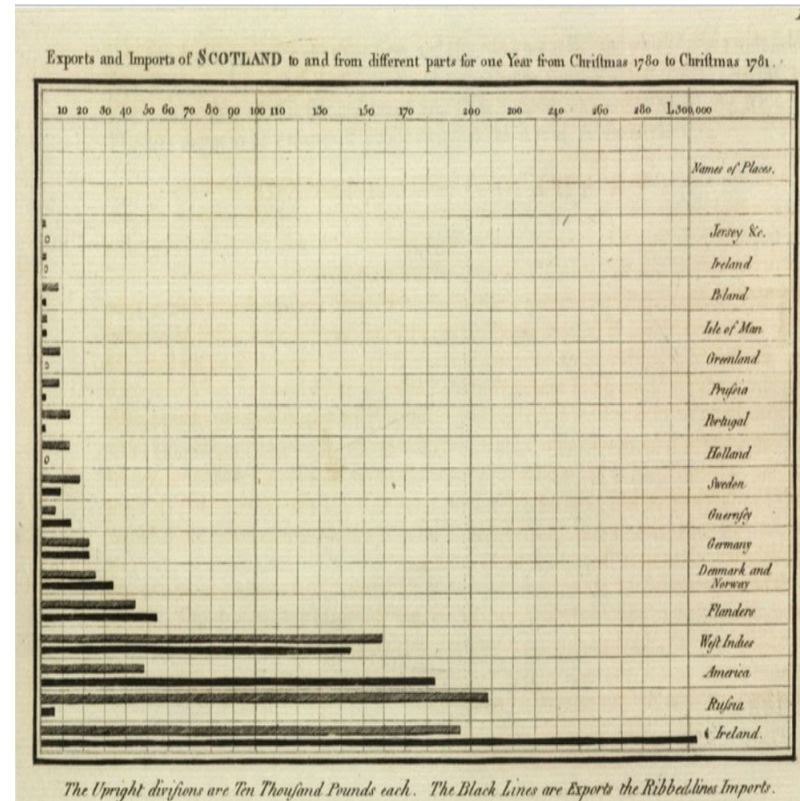
Source: [Smithsonian](#)

PUBH 6199: Visualizing Data with R



A brief history of Data Visualization

1786, William Playfair



Created first bar chart (featuring Scottish trade data, 1780 - 1781), as well as line and pie charts.

Source: [Wikipedia](#)

PUBH 6199: Visualizing Data with R



A brief history of Data Visualization

1854, John Snow

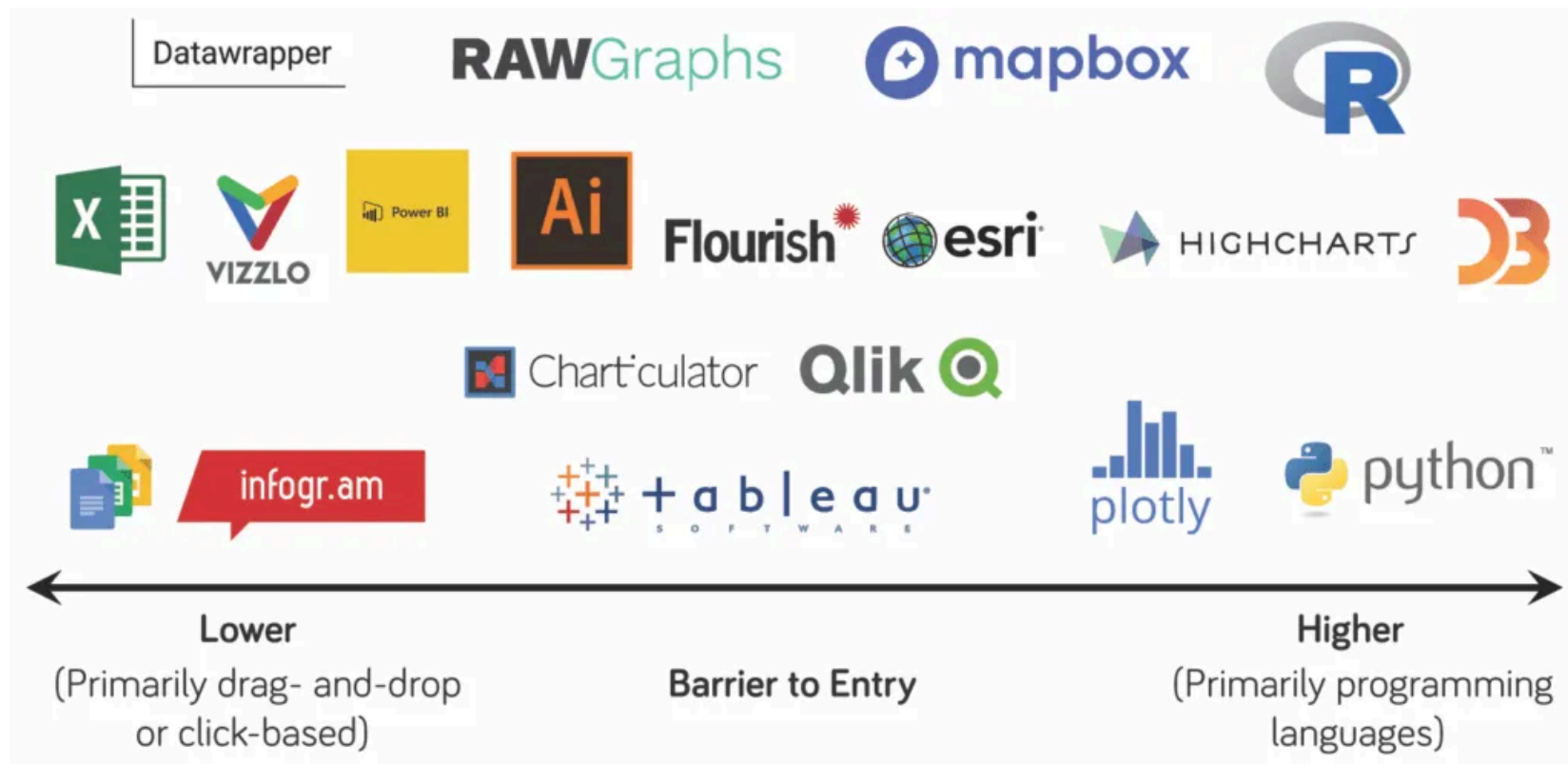


Used a dot map and showed the clusters of cholera cases in the London epidemic of 1854

Source: [Wikipedia](#)

A brief history of Data Visualization

Modern day



Source: [Jonathan Schwabish](#)



What will you learn in this class?

- Identify the effective types of data visualization for the data at hand and the intended audience
- Critique data visualizations and provide constructive feedback
- Prepare dataset for developing data visualization
- Create effective, ethical, and aesthetically-pleasing visualizations using R programming language
- Collaborate with classmates from diverse disciplinary background to carry out a visualization project



Outline for today

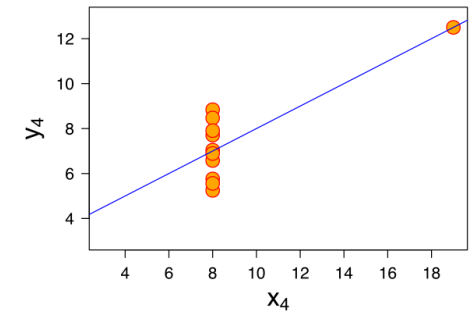
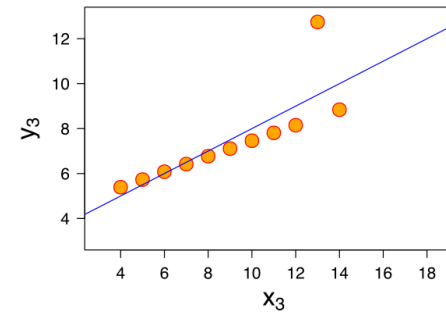
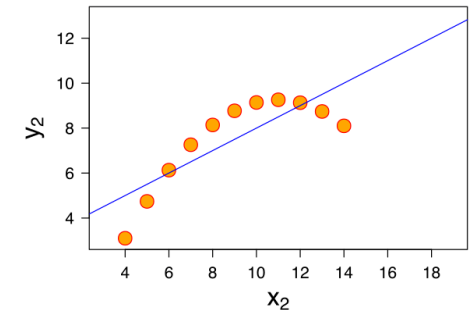
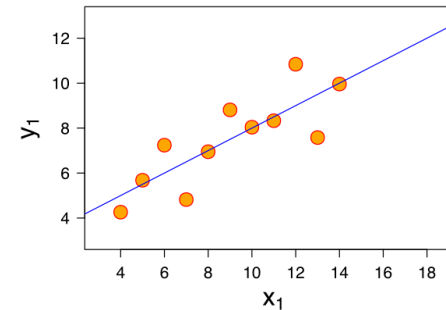
- Who?
- How?
- What?
- **Why?**
- Introduction to {ggplot2}



Why do we visualize data?

To reveal patterns that are hard to see in raw numbers...

Anscombe's Quartet: Raw Data								
	I		II		III		IV	
	x	y	x	y	x	y	x	y
	10.0	8.04	10.0	9.14	10.0	7.46	8.0	6.58
	8.0	6.95	8.0	8.14	8.0	6.77	8.0	5.76
	13.0	7.58	13.0	8.74	13.0	12.74	8.0	7.71
	9.0	8.81	9.0	8.77	9.0	7.11	8.0	8.84
	11.0	8.33	11.0	9.26	11.0	7.81	8.0	8.47
	14.0	9.96	14.0	8.10	14.0	8.84	8.0	7.04
	6.0	7.24	6.0	6.13	6.0	6.08	8.0	5.25
	4.0	4.26	4.0	3.10	4.0	5.39	19.0	12.50
	12.0	10.84	12.0	9.13	12.0	8.15	8.0	5.56
	7.0	4.82	7.0	7.26	7.0	6.42	8.0	7.91
	5.0	5.68	5.0	4.74	5.0	5.73	8.0	6.89
mean	9.0	7.5	9.0	7.5	9.0	7.5	9.0	7.5
var.	10.0	3.75	10.0	3.75	10.0	3.75	10.0	3.75
corr.		0.816		0.816		0.816		0.816



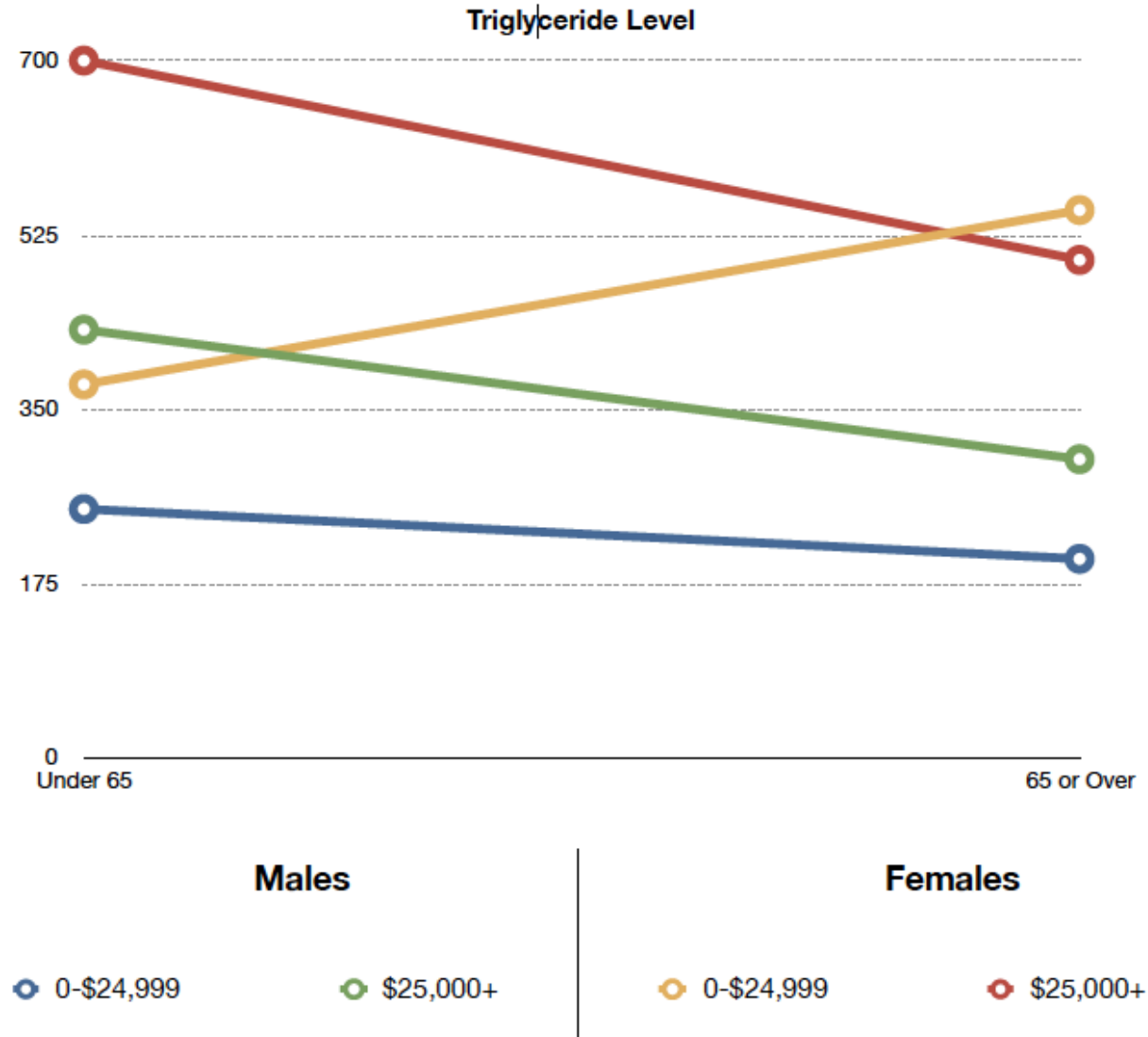
To communicate complex ideas quickly...

Income Group	Males: Under 65	Males: 65 or Over	Females: Under 65	Females: 65 or Over
0-\$24,999	250	200	375	550
\$25,000+	430	300	700	500

Is the effect of age on cholesterol levels the same for all subgroups defined by sex and income?



To communicate complex ideas quickly...



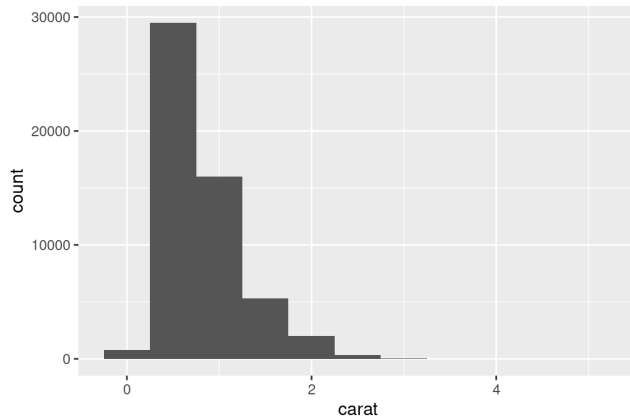
Source: [SM Kosslyn: Clear and to the point](#)



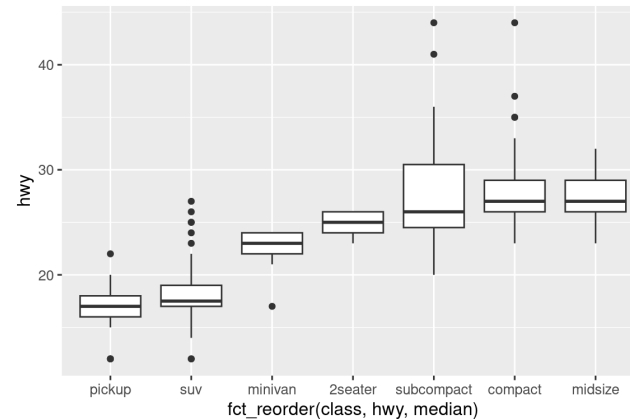
To explore and generate new questions

“Exploratory Data Analysis, or EDA, is a process to use visualization and transformation to explore your data in a systemic way. EDA is not a formal process with a strict set of rules. More than anything, EDA is a state of mind. During the initial phases of EDA you should feel free to investigate every idea that occurs to you. Some of these ideas will pan out, and some will be dead ends. As your exploration continues, you will home in on a few particularly productive areas that you’ll eventually write up and communicate to others.”

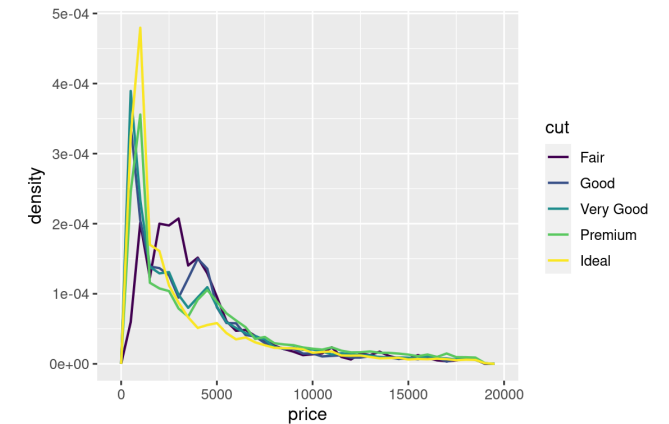
-from R for Data Science



```
1 ggplot(diamonds, aes(x = carat))
2   geom_histogram(binwidth = 0.5)
```



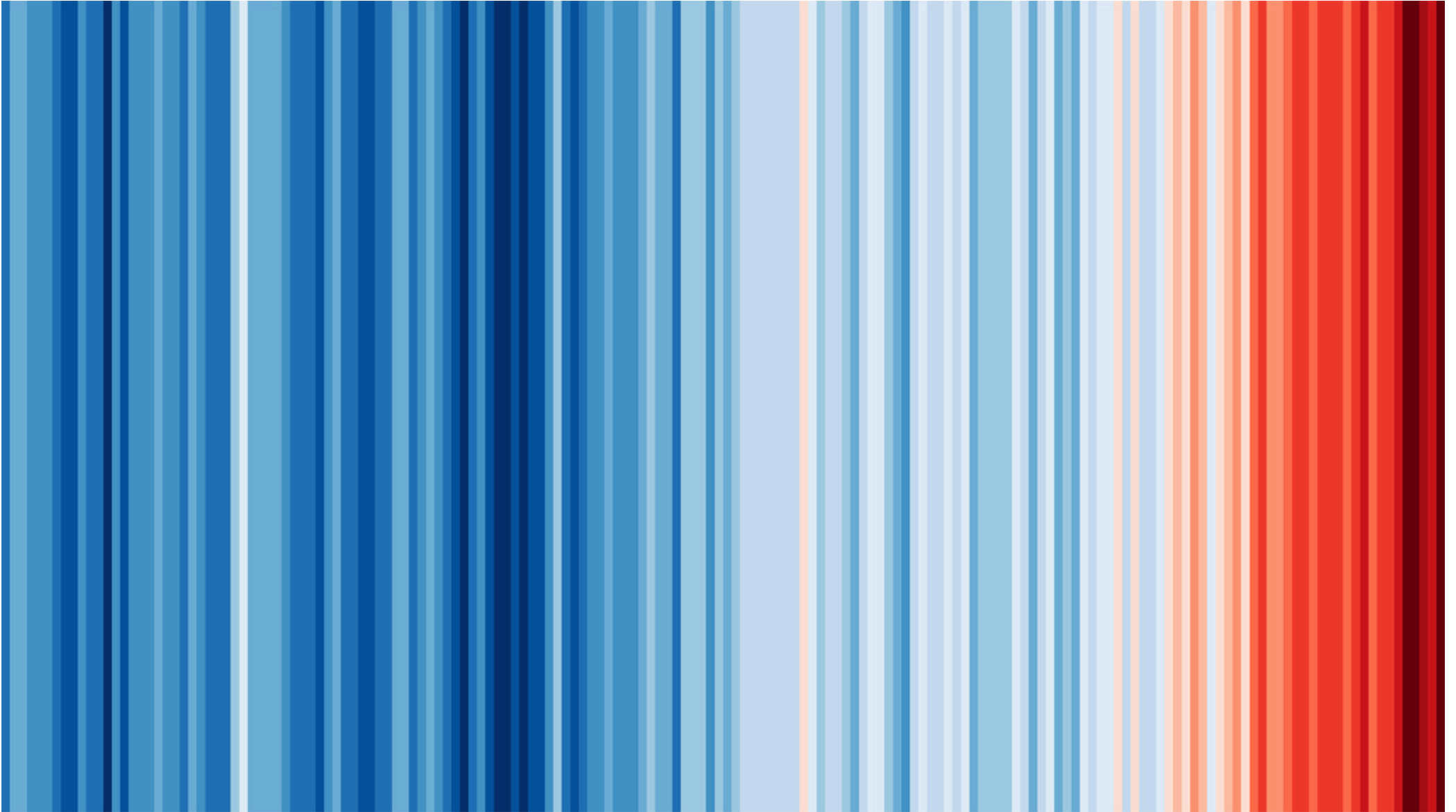
```
1 ggplot(mpg, aes(x = fct_reorder(
2   geom_boxplot()
```



```
1 ggplot(diamonds, aes(x = price,
2   geom_freqpoly(aes(color = cut)
```

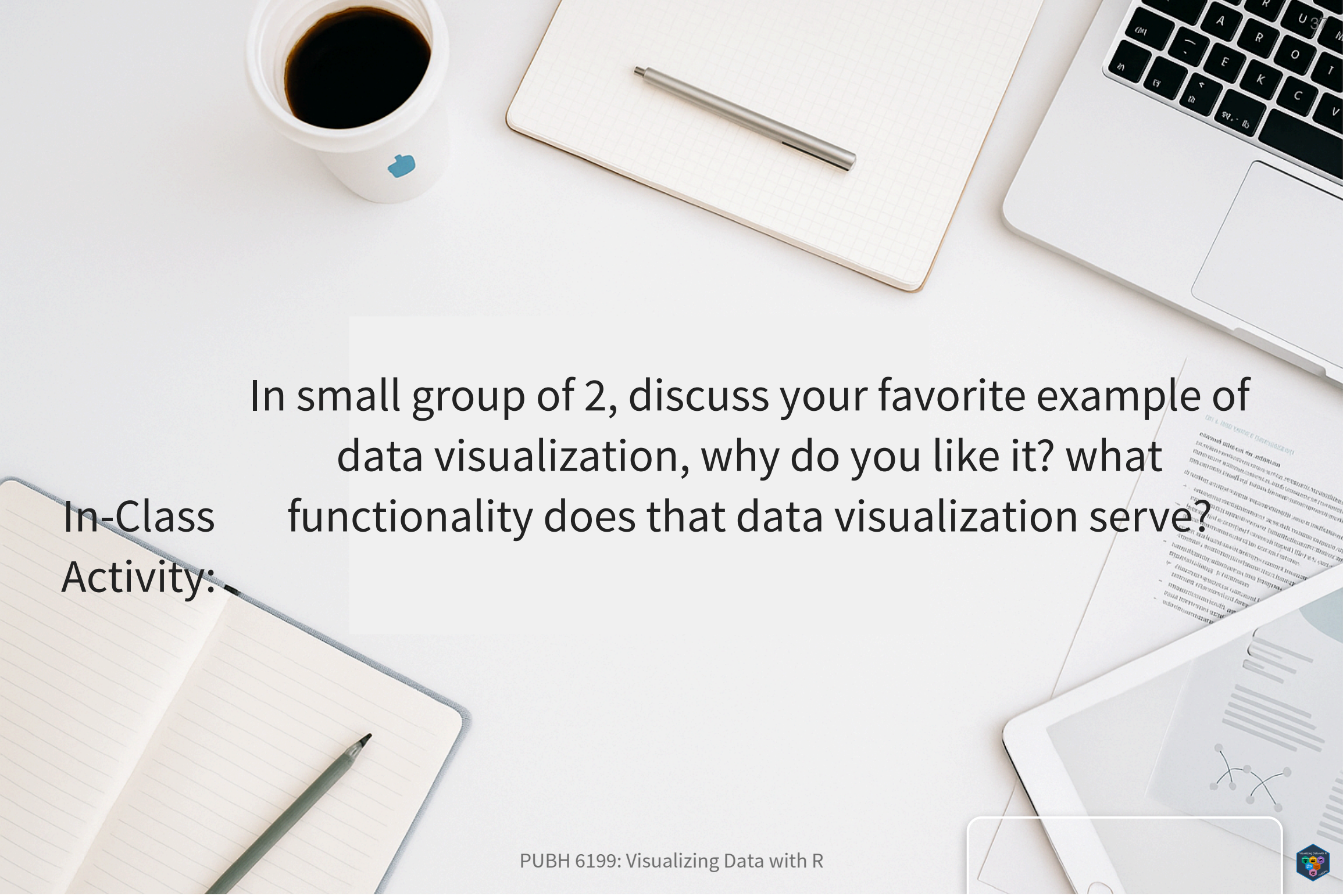


To tell a story and evoke emotions



Source: Ed Hawkins, [Climate Stripes](#)





In small group of 2, discuss your favorite example of data visualization, why do you like it? what functionality does that data visualization serve?

In-Class Activity:



Why R?

- Open-source and free
- Highly customizable
- Script-based and reproducible
- Data analysis and visualization in one language
- Large open-source community and ecosystem



Art by [Allison Horst](#)

 Take a Break

~ This is the end of part 1 ~

Outline for today

- Who?
- How?
- What?
- Why?
- **Introduction to {ggplot2}**



Welcome to {ggplot2}

- Based on Grammar of Graphics ([Wilkinson 2005](#))
- Hadley Wickham developed ggplot2 based on Wilkinson's grammar of graphics in 2009
- Allow you to compose graphs by combining independent components
- Designed to work iteratively:
 - Start with a simple layer that shows the raw data
 - Add layers of annotations and summary statistics
 - Each layer can be customized independently



Art by [Allison Horst](#)

What is the grammar of graphics?

“A graphic maps the data to the aesthetic attributes (color, shape, size) of geometric objects (points, lines, bars). The plot may also include statistical transformations of the data and information about the plot’s coordinate system. Facetting can be used to plot for different subsets of the data. The combination of these independent components are what make up a graphic.”

Wilkinson (2005)



{ggplot2} graphic layers

First these:

- **data**: in tidy format (Lab 1)
- **mapping**: how variables are mapped to aesthetic attributes
- **geom**: the geometric object used to display the data

Then these:

- **stat**: statistical transformations, e.g. binning and counting, fitting a linear model
- **scale**: maps values in data space to values in the aesthetics space, also draw the legend
- **coord**: normally the Cartesian coordinate system, but can be polar, map, etc.
- **facet**: display subsets of data as small multiples
- **theme**: non-data ink, e.g. background color, grid lines, font size, etc.



Airquality dataset

The `airquality` dataset contains daily air quality measurements in New York, May to September 1973. The data frame has 153 observations and 6 variables:

- **Ozone**: Ozone in parts per billion (ppb)
- **Solar.R**: Solar radiation in langley
- **Wind**: Average wind speed in miles per hour (mph)
- **Temp**: Maximum daily temperature in degrees Fahrenheit (F)
- **Month**: Month of the year (1-12)
- **Day**: Day of the month (1-31)

```
1 library(tidyverse)
2 data(airquality)
3 glimpse(airquality)
```

Rows: 153

Columns: 6

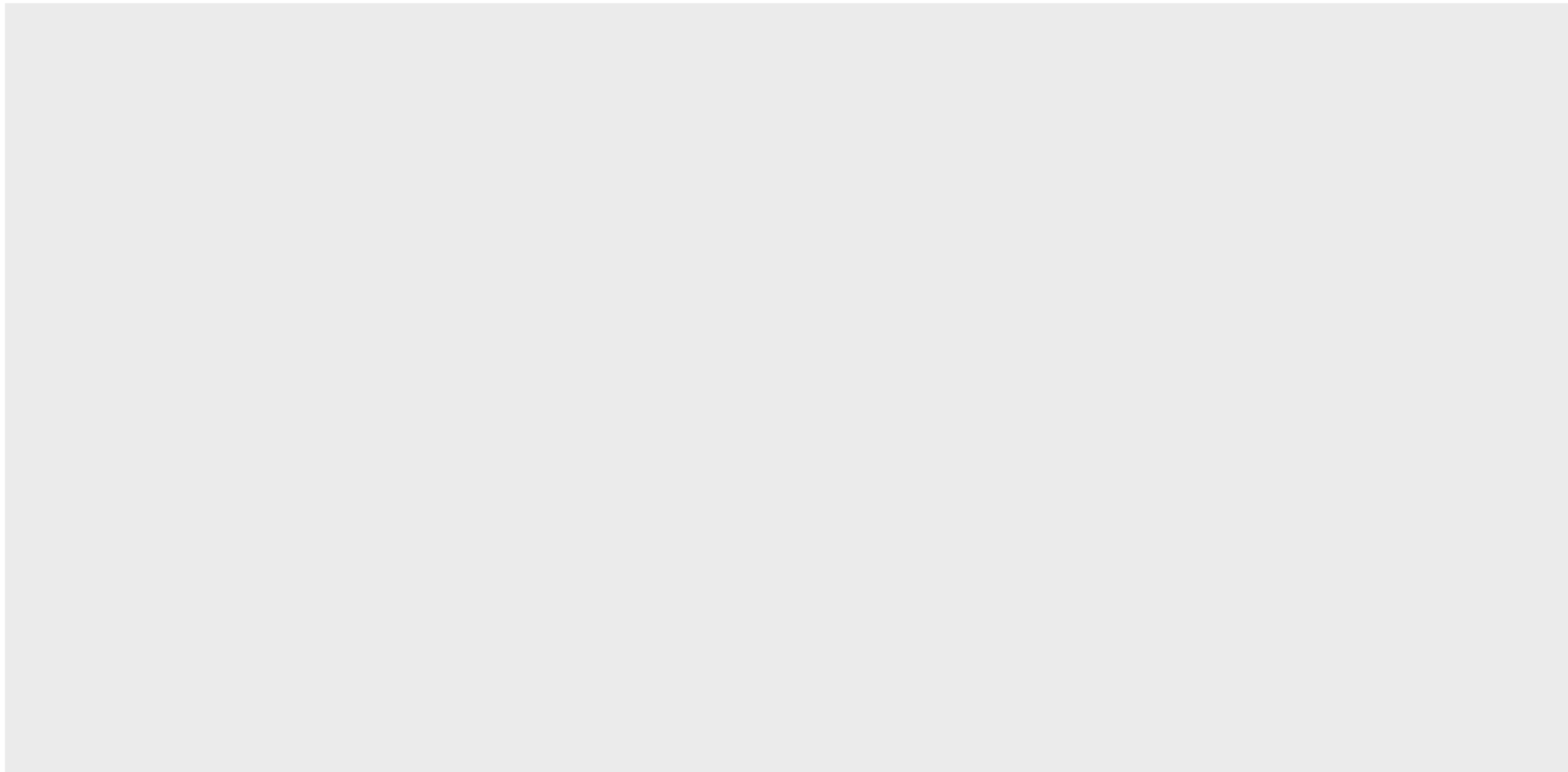
```
$ Ozone   <int> 41, 36, 12, 18, NA, 28, 23, 19, 8, NA,
7, 16, 11, 14, 18, 14, ...
$ Solar.R <int> 190, 118, 149, 313, NA, NA, 299, 99,
19, 194, NA, 256, 290, 27...
$ Wind    <dbl> 7.4, 8.0, 12.6, 11.5, 14.3, 14.9, 8.6,
13.8, 20.1, 8.6, 6.9, 9...
$ Temp    <int> 67, 72, 74, 62, 56, 66, 65, 59, 61, 69,
74, 69, 66, 68, 58, 64...
$ Month   <int> 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5,
5, 5, 5, 5, 5, 5, 5, ...
$ Day     <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12,
13, 14, 15, 16, 17, 18, ...
```



Step 0: Initialize a plot object

Initialize the plot using `ggplot()`. It is empty because we haven't told ggplot how to map the data to the plot yet.

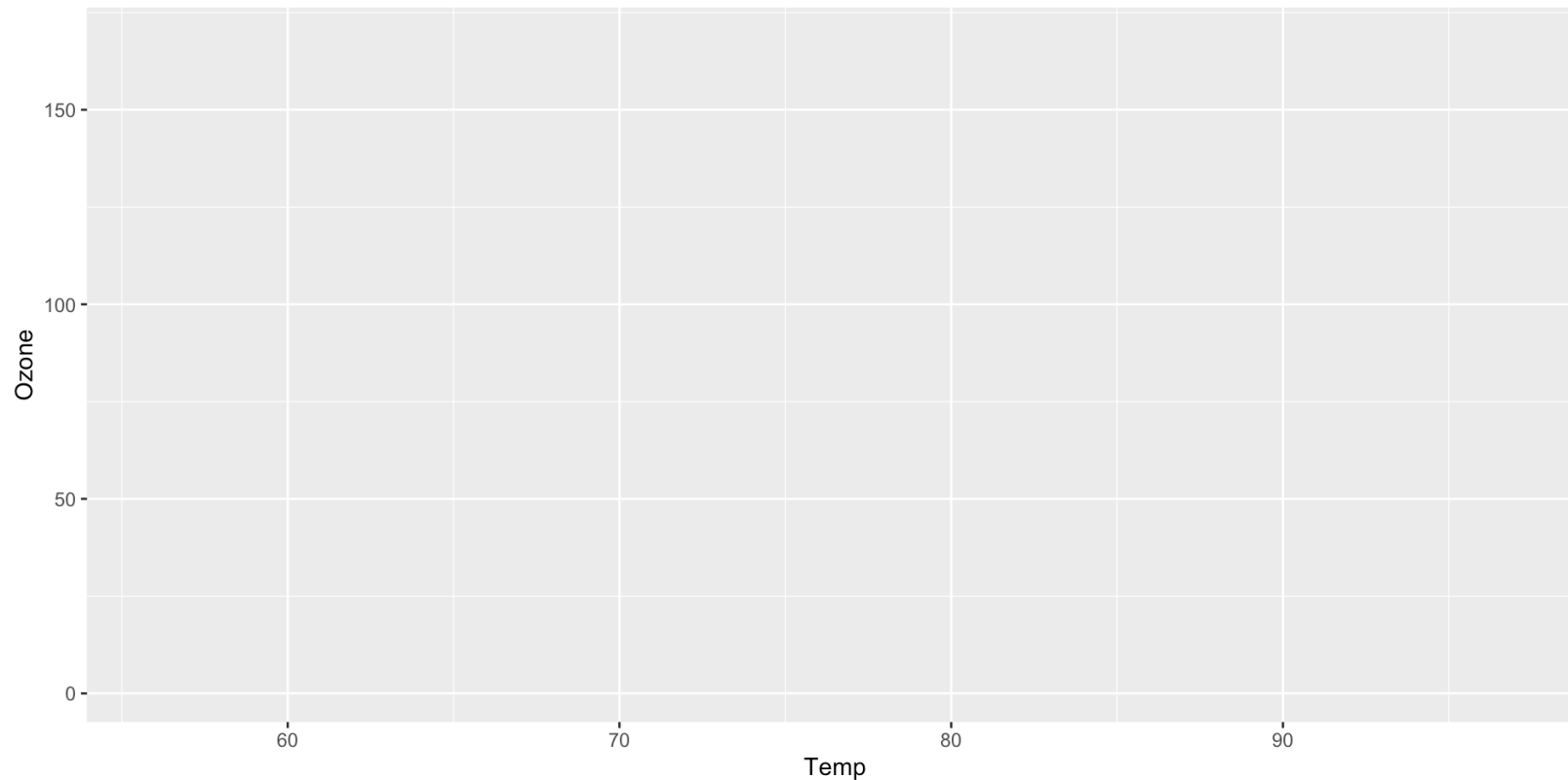
```
1 library(tidyverse)
2 data(airquality)
3 ggplot(data = airquality)
```



Step 1: Map the variables

The `mapping` argument is used to specify how variables in the data are mapped to aesthetic attributes of the plot. The `aes()` function is used to define the mapping.

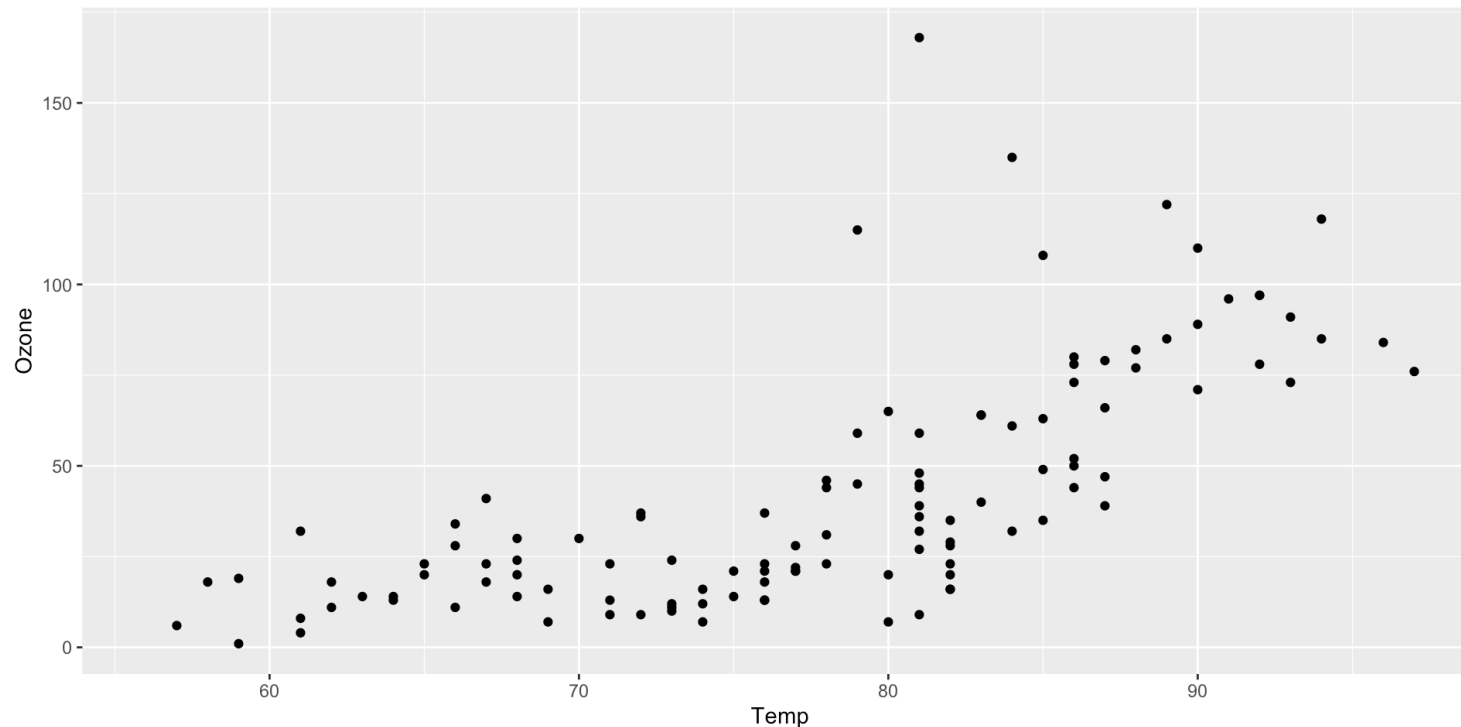
```
1 library(tidyverse)
2 data(airquality)
3 ggplot(data = airquality,
4       mapping = aes(x = Temp, y = Ozone)) # x-axis and y-axis
```



Step 2: Add points (`geom_point`)

Next, we add a geometric object (`geom`) that represents the data. In this case, we use `geom_point()` to add points to the plot. There are many more geoms (`geom_*()`) built into `{ggplot2}` and extension packages.

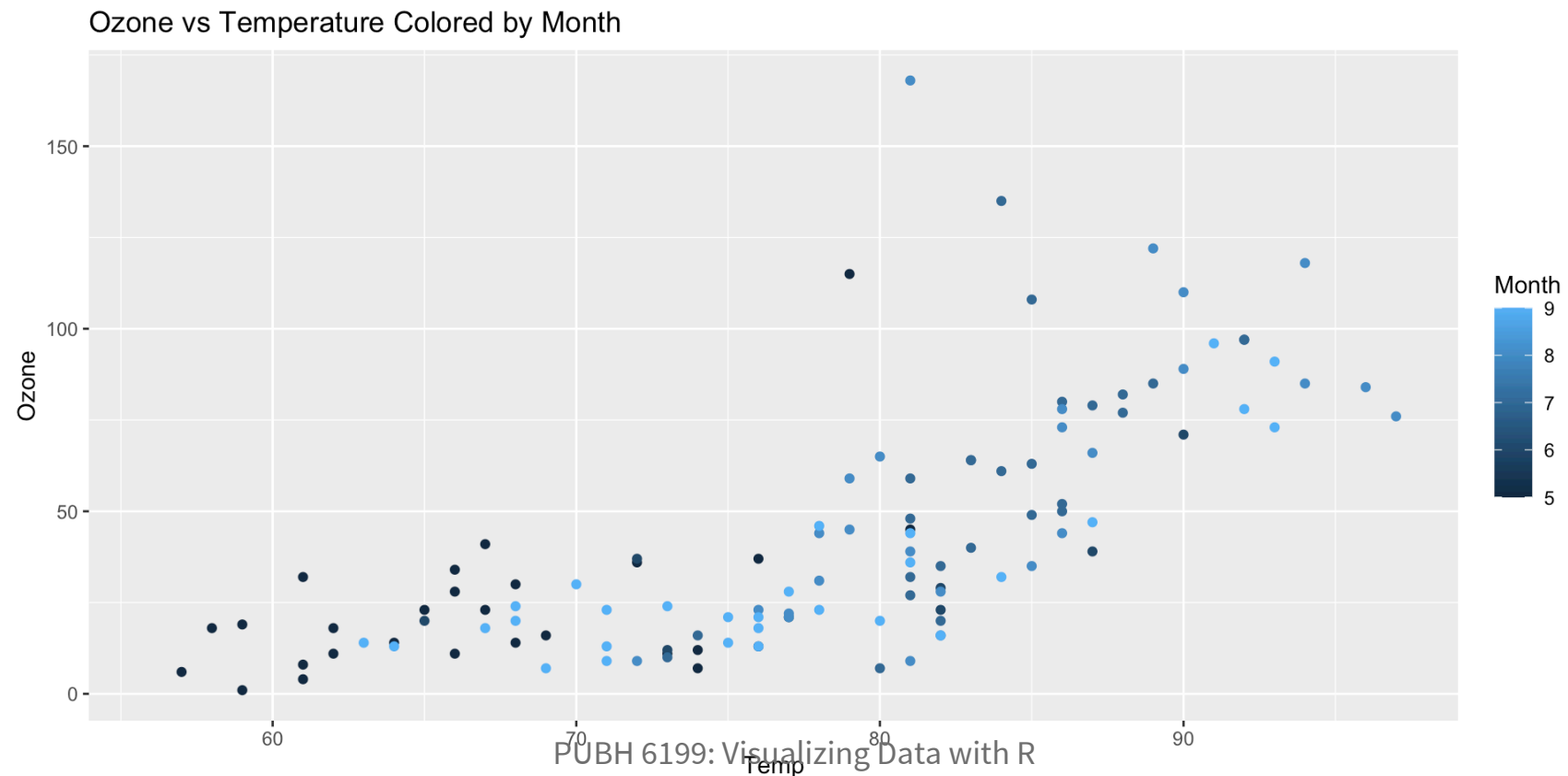
```
1 library(tidyverse)
2 data(airquality)
3 ggplot(data = airquality, mapping = aes(x = Temp, y = Ozone)) + # x-axis and y-axis
4   geom_point() # add points
```



Step 3: Color aesthetic (Month)

If we like to add more information to the plot, we can use the `color` aesthetic to map another variable to the color of the points. In this case, we will use `Month` as the color aesthetic.

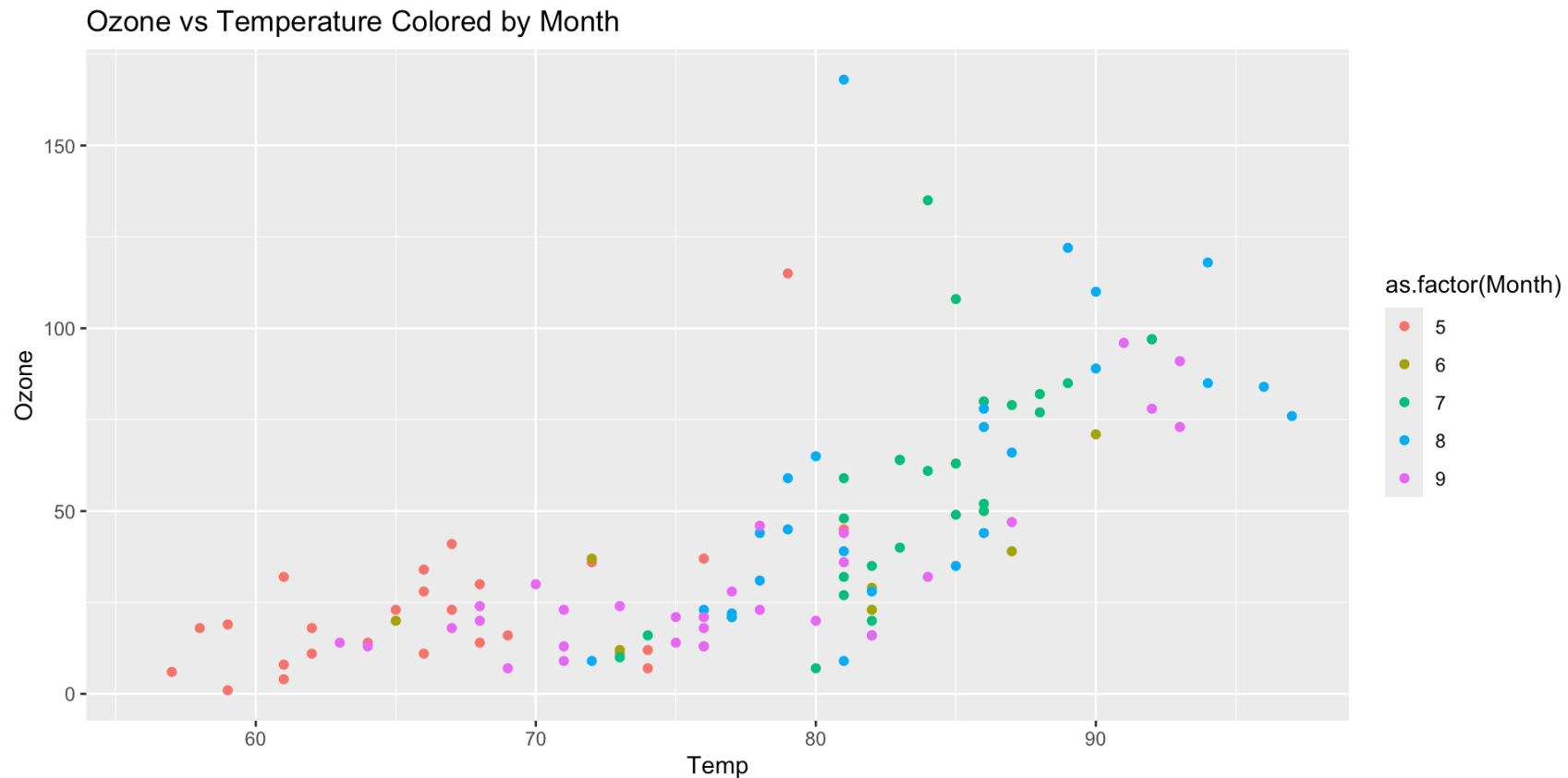
```
1 ggplot(airquality, aes(x = Temp, y = Ozone, color = Month)) +  
2   geom_point() +  
3   labs(title = "Ozone vs Temperature Colored by Month")
```



Step 3: Color aesthetic (Month)

Instead of treating `Month` as a continuous variable, maybe we want to treat `Month` like a categorical variable.

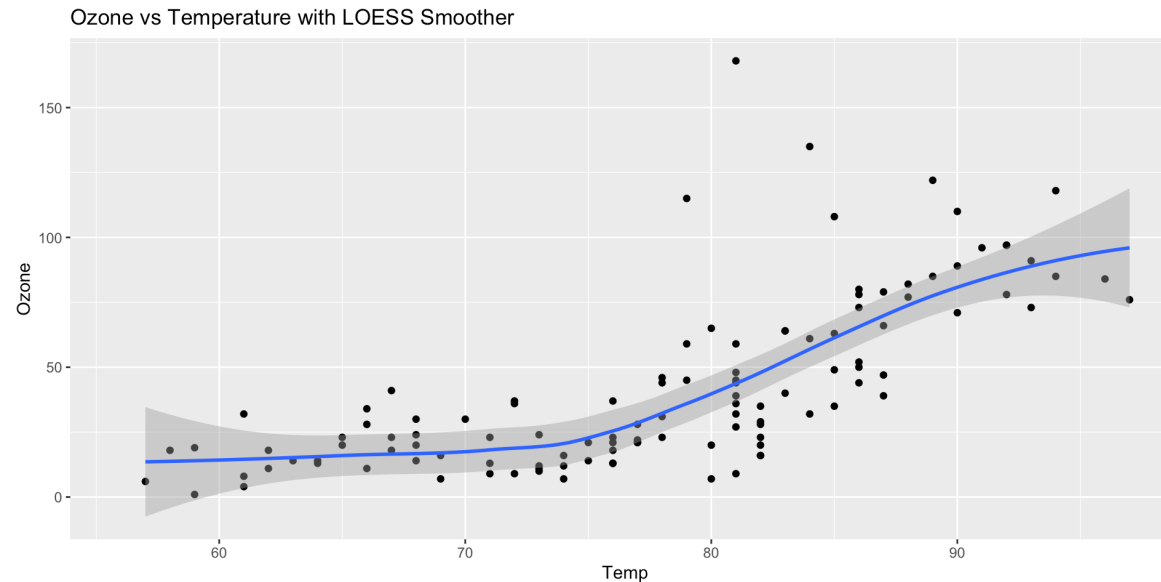
```
1 ggplot(airquality, aes(x = Temp, y = Ozone, color = as.factor(Month))) +  
2   geom_point() +  
3   labs(title = "Ozone vs Temperature Colored by Month")
```



Step 4: Add layers (smoother lines with `geom_smooth`)

We can add a smoother line to the plot using `geom_smooth()`. The default method is linear regression, but we can also use other methods like LOESS (locally weighted scatterplot smoothing).

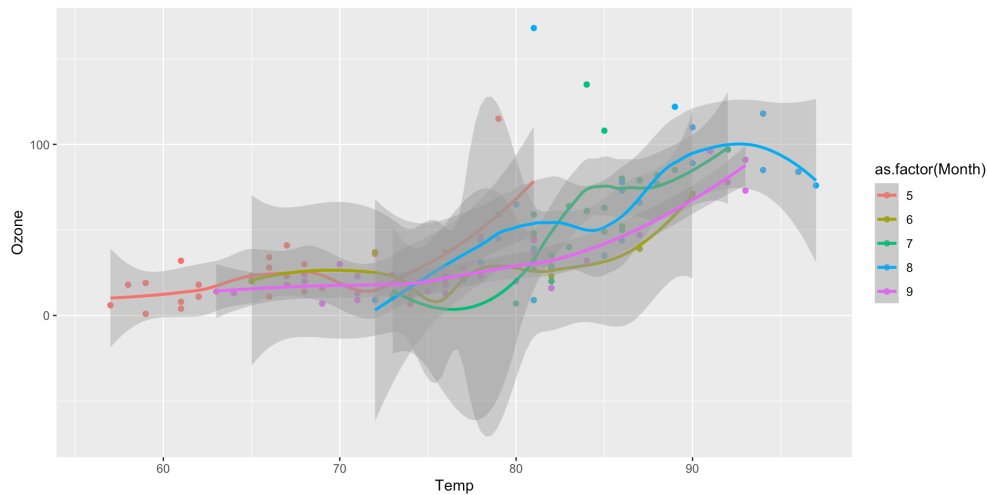
```
1 ggplot(airquality, aes(x = Temp, y = Ozone)) +  
2   geom_point() +  
3   geom_smooth(method = "loess") +  
4   labs(title = "Ozone vs Temperature with LOESS Smoother")
```



Global mapping v.s. Local mapping

Global mapping are passed down to each subsequent layer

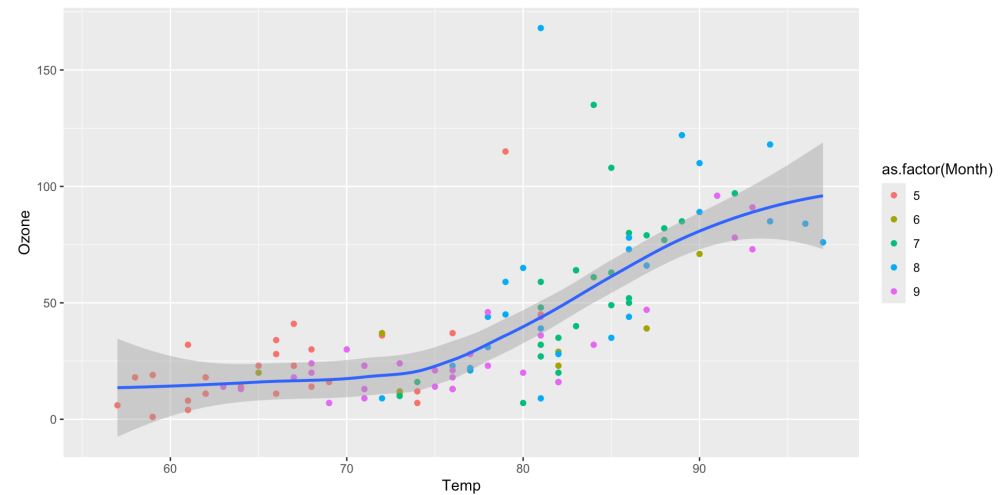
```
1 ggplot(airquality, aes(x = Temp, y = Ozone, color = 
2   geom_point() + 
3   geom_smooth(method = "loess"))
```



`color = as.factor(Month)` is passed to both `geom_point()` and `geom_smooth()`, so the points and the line are colored by month.

Local mapping are only used in that layer and don't affect other layers

```
1 ggplot(airquality, aes(x = Temp, y = Ozone)) + 
2   geom_point(aes(color = as.factor(Month))) + 
3   geom_smooth(method = "loess")
```



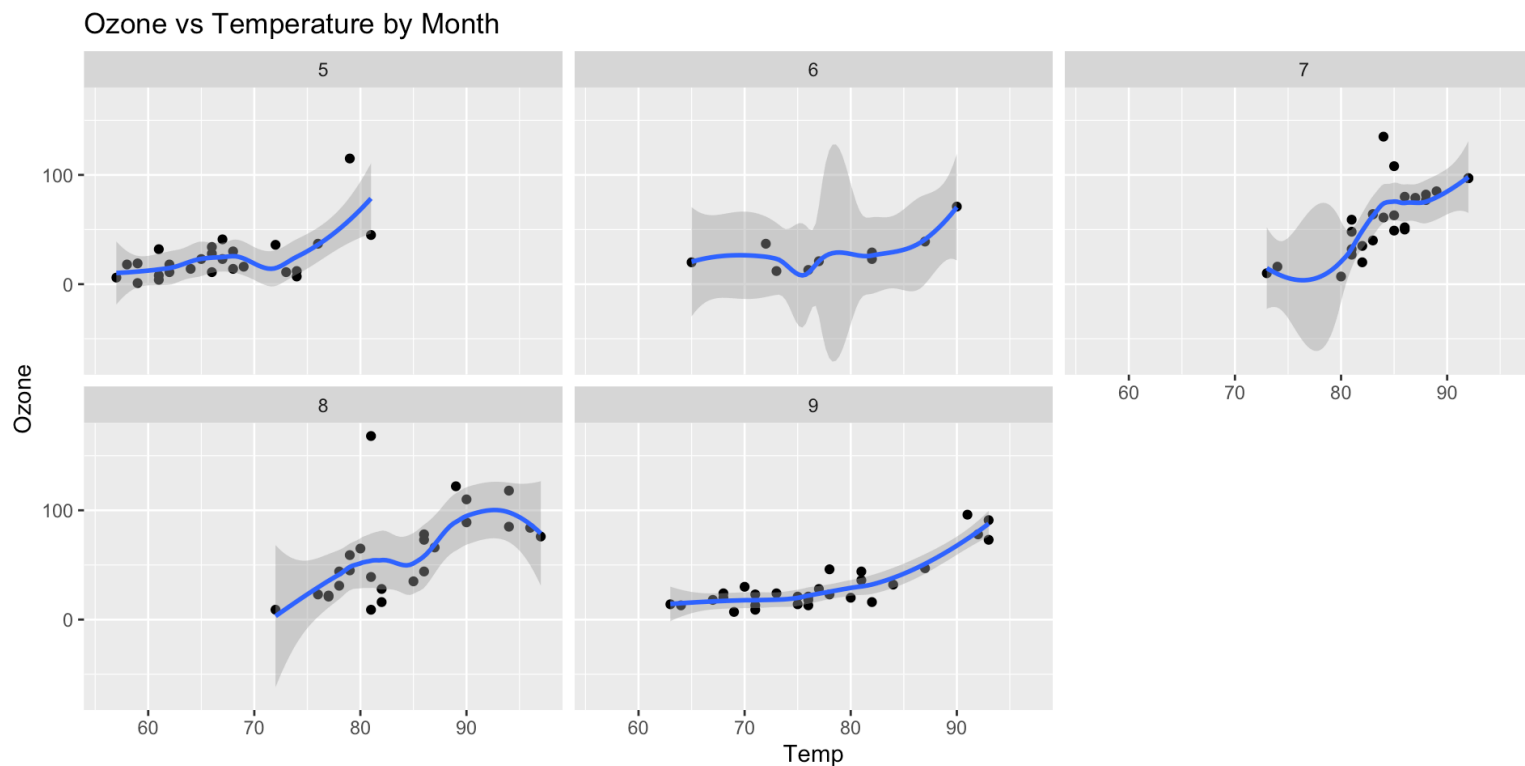
`color = as.factor(Month)` is only passed to `geom_point()`, so the points are colored by month, but the line is not colored by month.



Step 5: Facet by month (`facet_wrap`)

We can use `facet_wrap()` to create small multiples of the plot, one for each month. This allows us to see how the relationship between temperature and ozone varies by month.

```
1 ggplot(airquality, aes(x = Temp, y = Ozone)) +
2   geom_point() +
3   facet_wrap(~ Month) +
4   geom_smooth(method = "loess") +
5   labs(title = "Ozone vs Temperature by Month")
```



Step 6: Customize the appearance

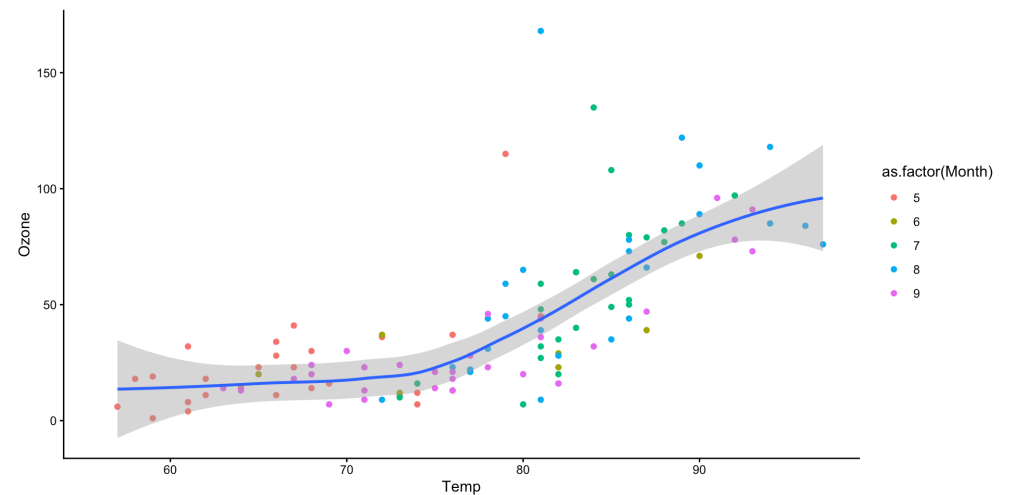
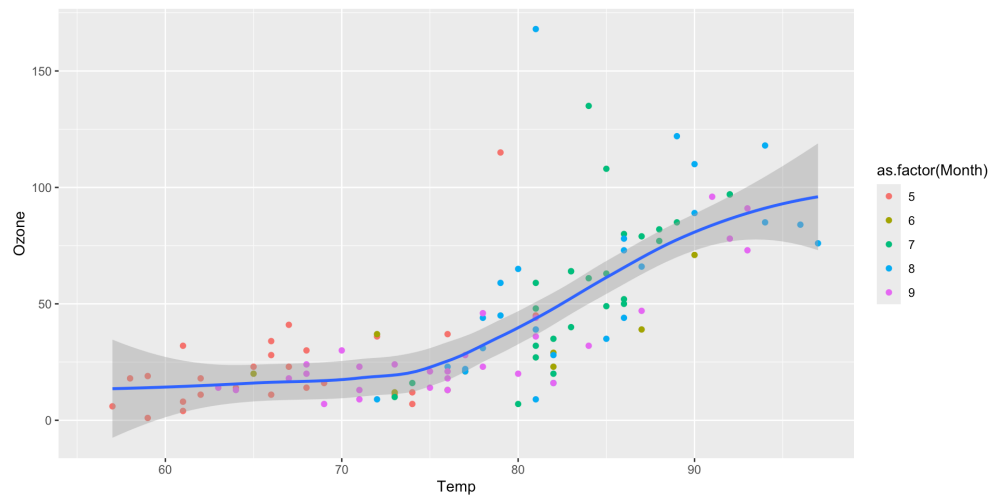
`ggplot2` has a number of built-in themes, which control all non-data display.

Never use the default theme

`theme_classic()` is a good starting point

```
1 ggplot(airquality, aes(x = Temp, y = Ozone)) +
2   geom_point(aes(color = as.factor(Month))) +
3   geom_smooth(method = "loess")
```

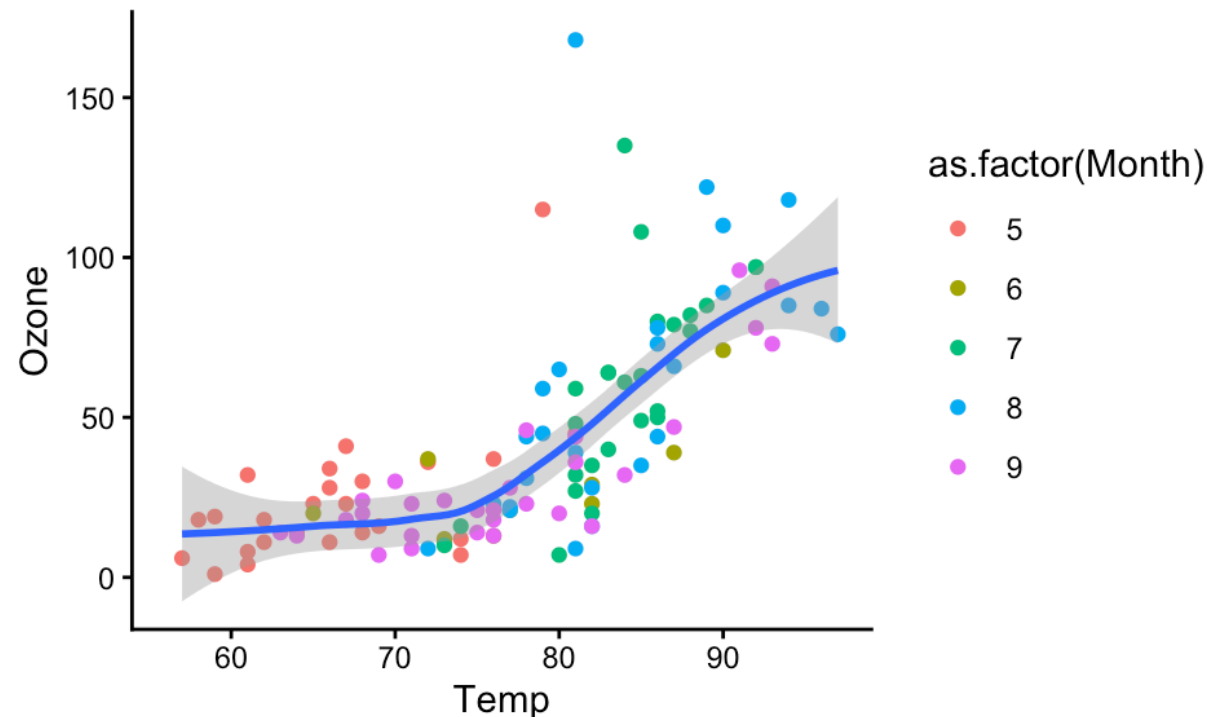
```
1 ggplot(airquality, aes(x = Temp, y = Ozone)) +
2   geom_point(aes(color = as.factor(Month))) +
3   geom_smooth(method = "loess") +
4   theme_classic()
```



Step 6: Customize the appearance

Almost always the default font size in `ggplot2` are too small. This is because the font size is set to 11 by default, but the size of the figure is set to 10 inches by 5 inches, so when you insert the figure to a Word or Powerpoint, it ends up being too small

```
1 ggplot(airquality, aes(x = Temp, y = Ozone)) +  
2   geom_point(aes(color = as.factor(Month))) +  
3   geom_smooth(method = "loess") +  
4   theme_classic(base_size = 11) # I set the output figure size to be 5 inches by 3 inches
```

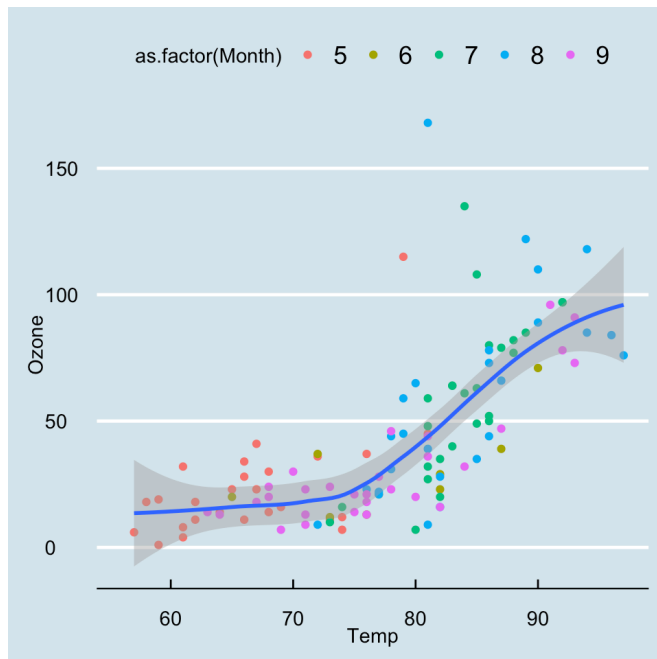


Step 6: Customize the appearance

You can explore other pre-built themes in the `{ggthemes}`.

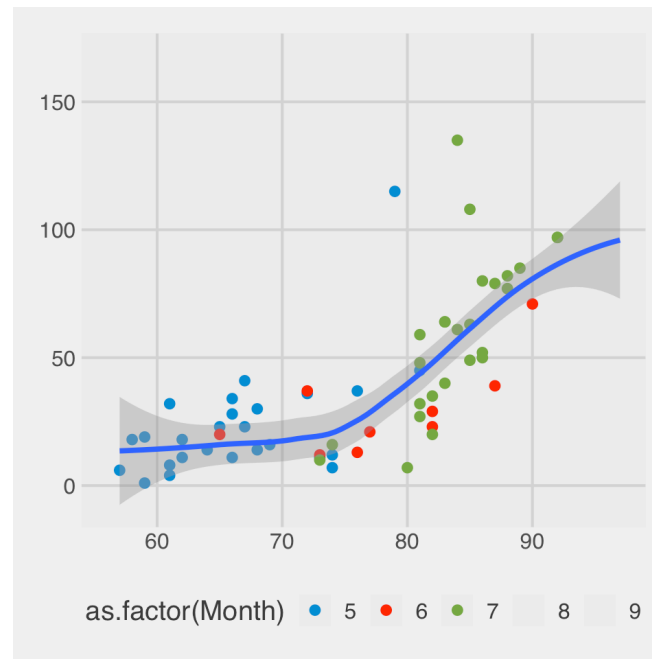
Theme economist

```
1 library(ggthemes)
2 ggplot(airquality, aes(x = Temp,
3   geom_point(aes(color = as.factor
4   geom_smooth(method = "loess")
5   theme_economist()
```



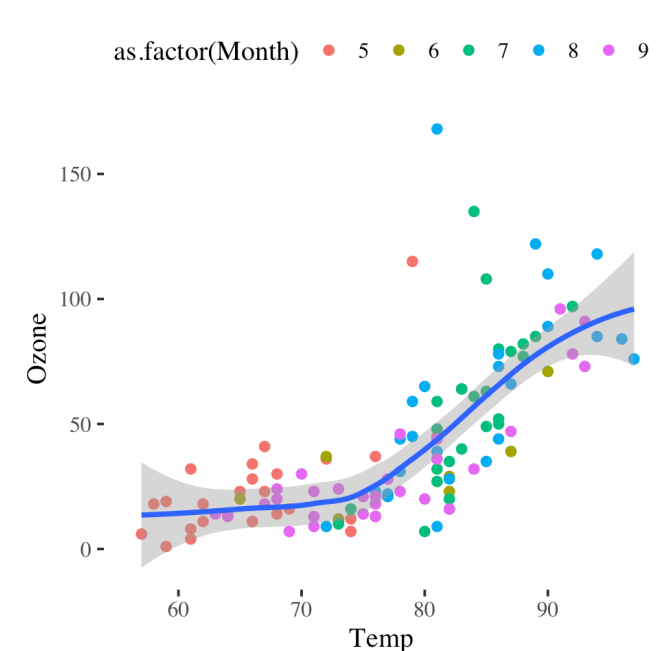
Theme 538

```
1 ggplot(airquality, aes(x = Temp,
2   geom_point(aes(color = as.factor
3   geom_smooth(method = "loess")
4   scale_color_fivethirtyeight()
5   theme_fivethirtyeight(base_siz
```



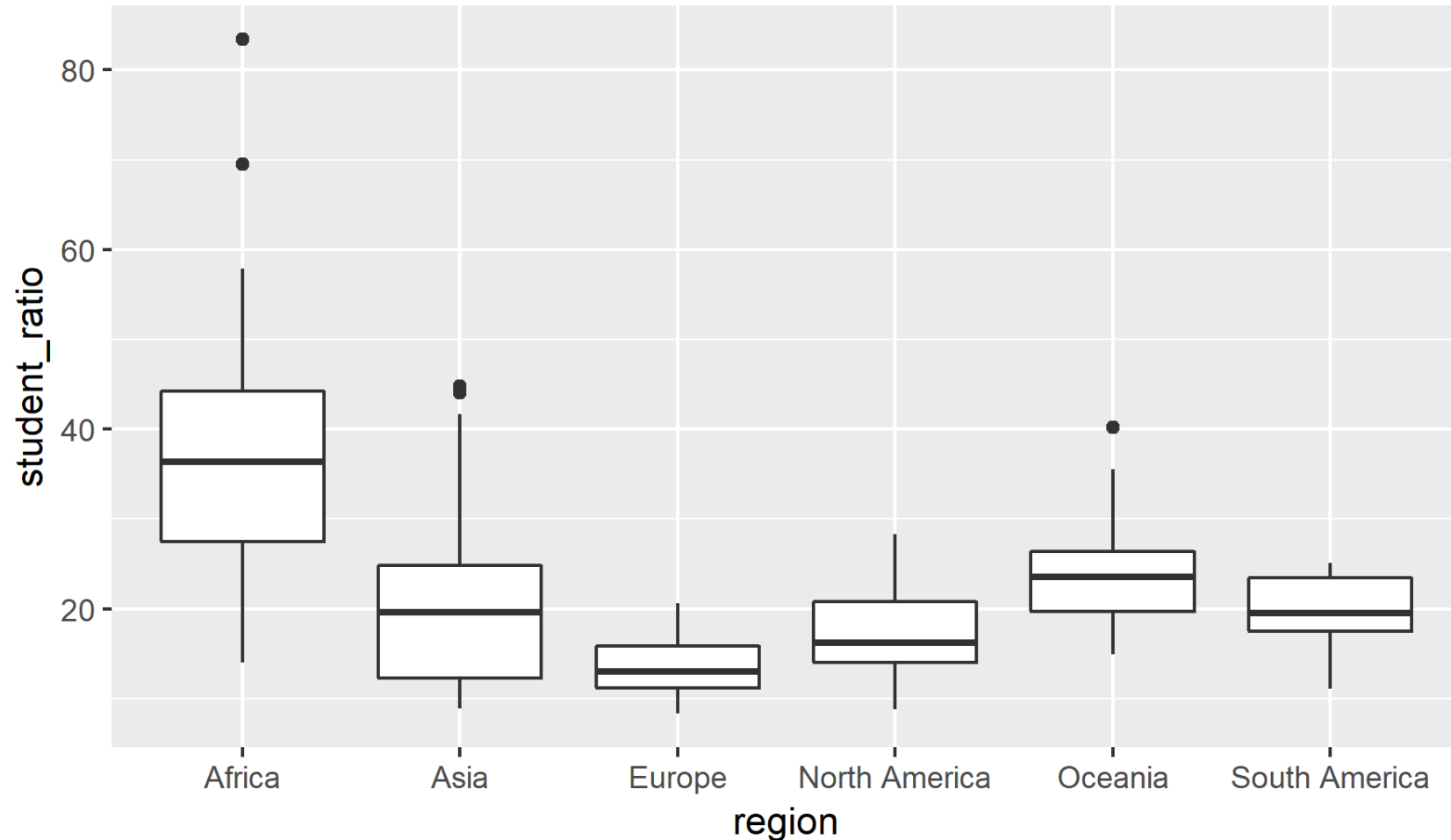
Theme Tufte

```
1 ggplot(airquality, aes(x = Temp,
2   geom_point(aes(color = as.factor
3   geom_smooth(method = "loess")
4   theme_tufte(base_size = 15) +
5   theme(legend.position = "top")
```



Building a data viz is an interative process!

The Evolution of a ggplot



Data: UNESCO Institute for Statistics
Visualization by Cédric Scherer

Make your own ggplot evolution using the `{camcorder}` package

PUBH 6199: Visualizing Data with R



End-of-Class Survey

 Fill out the end-of-class survey

~ This is the end of Lecture 1 ~