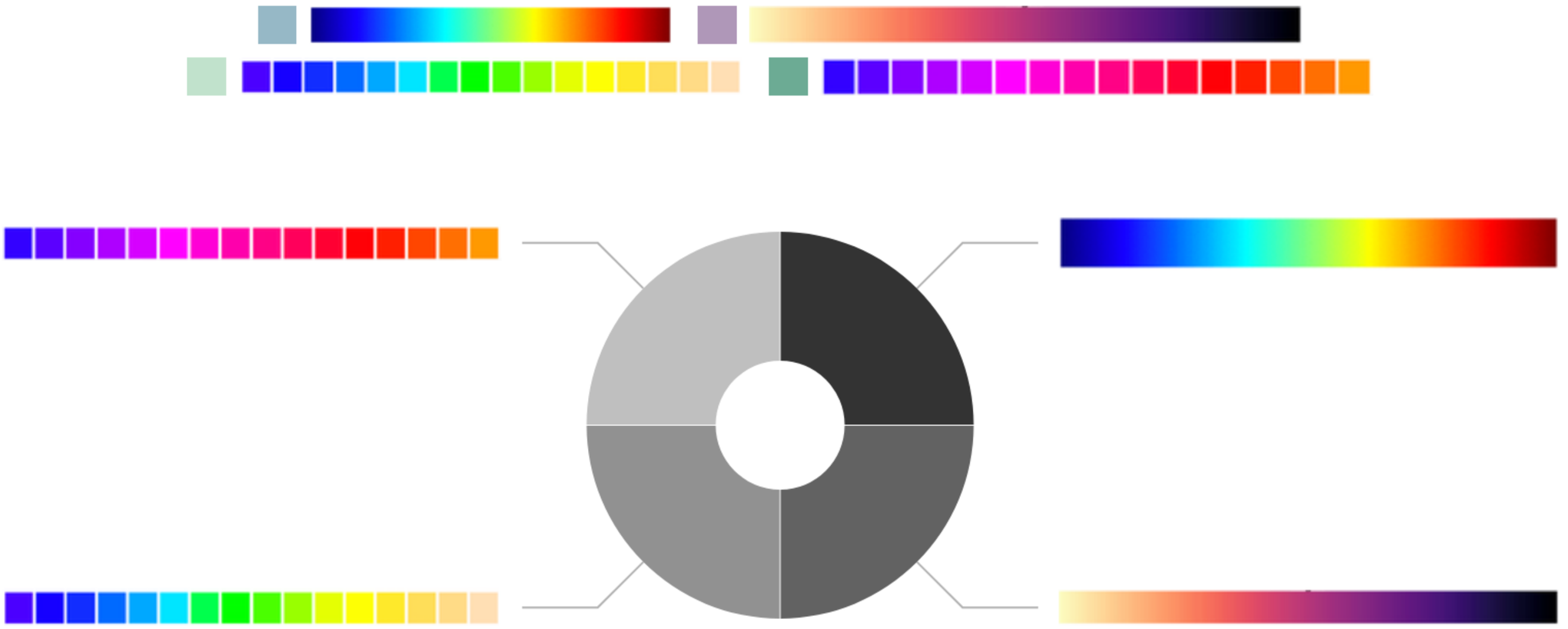
Review on posts

3 volunteers to talk about their visualization

Quick Assessment

A lab member wants to visualize the speed (milliseconds) of a subject's decision making cognitive test. If they were using color to visualize the range of speeds, what scale would be most effective?



A lab member wants to gather data on the number of times a mouse pointed it's nose towards a stimulus. How would they characterize that data?

Numerical, continuous

Numerical, discrete

Categorical, nominal

Categorical, ordinal

What best describes n_of_correct_responses in the attached image?

```{r}
n_of_correct_responses <- c(1,5,4,6,3)
```

A function computing a list of numbers

An assigned value

An object

A new variable

What best describes <- in the attached image?

```{r}
n_of_correct_responses <- c(1,5,4,6,3)
```

The equals operator

The assignment operator

A function

The creation operator

What best describes c() in the attached image?

```{r}
n_of_correct_responses <- c(1,5,4,6,3)
```

A variable

An object

A function

The assignment operator

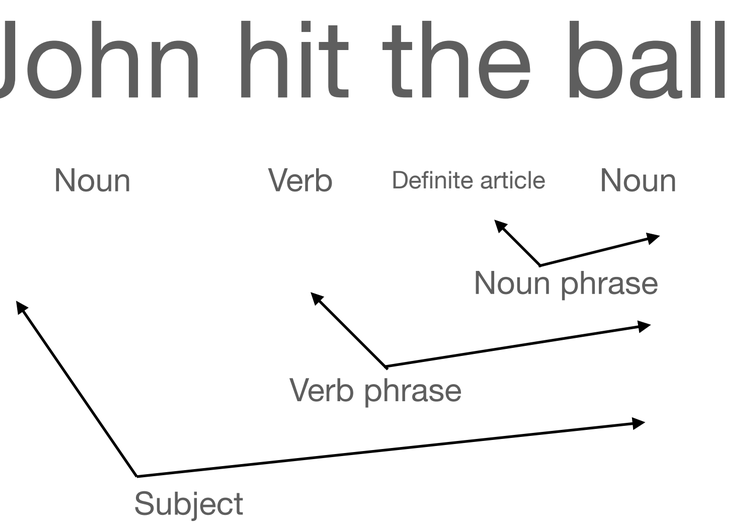
The Grammar of Graphics

ggplot

L. Andrew Bell, PhD Fall 2020

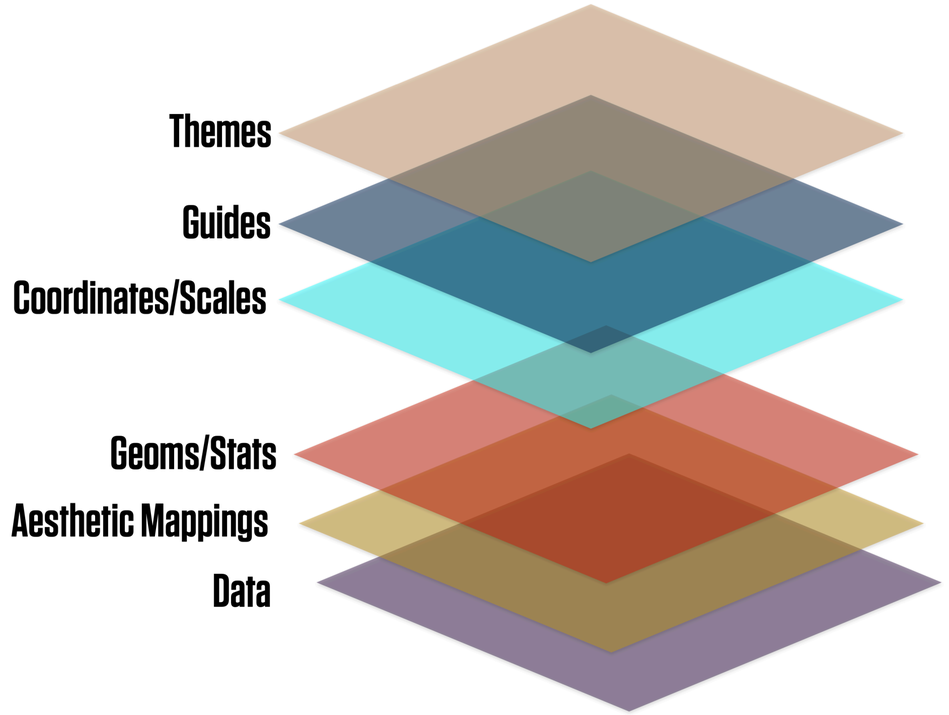
What is the grammar of linguistics?

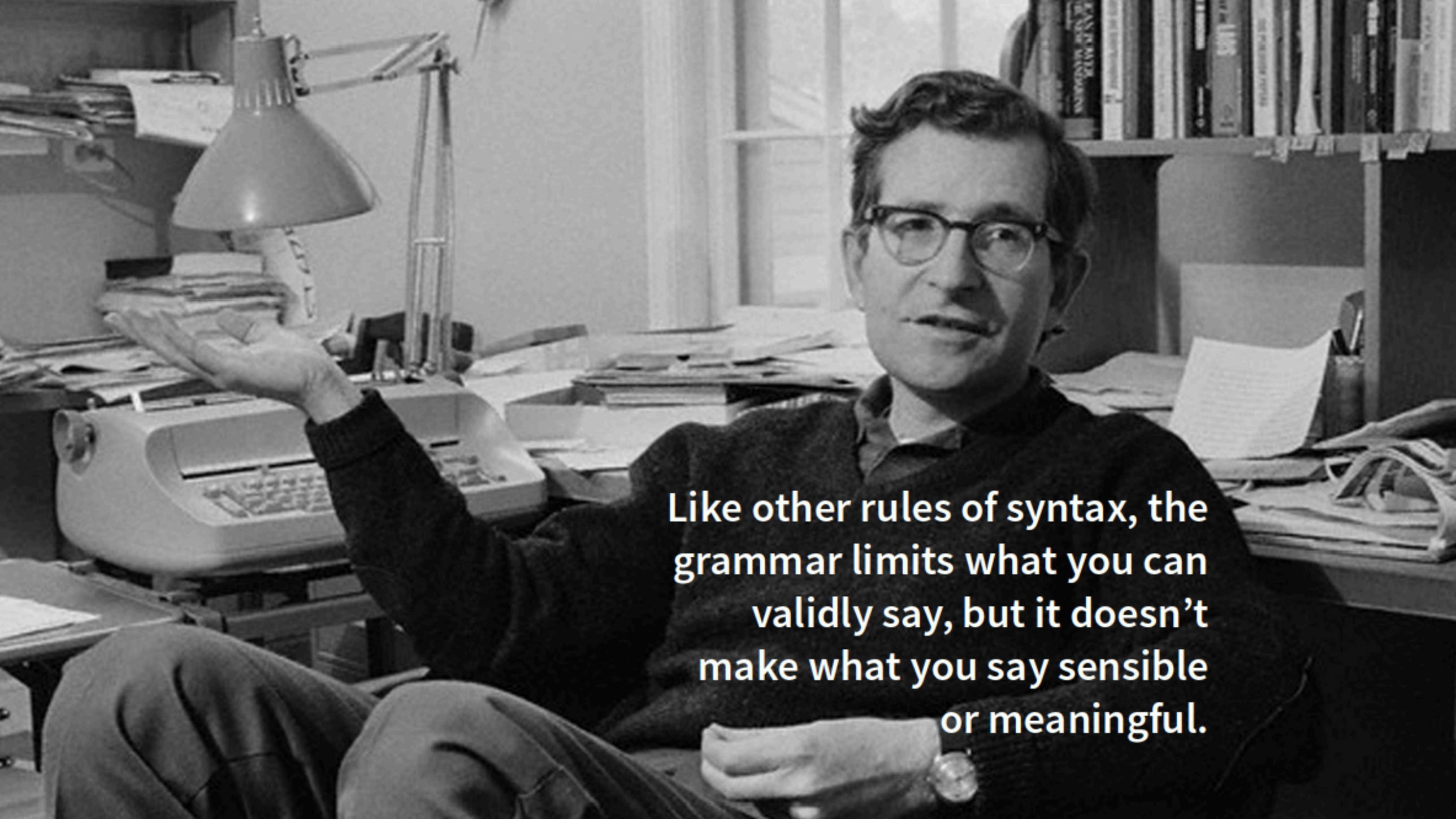
The grammar is the set of structural rules governing the composition of clauses, phrases, and words in a natural language



What is the grammar of graphics?

The grammar is a set of rules for how produce graphics from data, taking **pieces of data** and **mapping** them to **geometric objects** (like points and lines) **that have aesthetic attributes** (like position, color and size), together with further rules for **transforming the data if needed**, adjusting **scales**, or projecting the results onto a **coordinate system**.




Like other rules of syntax, the
grammar limits what you can
validly say, but it doesn’t
make what you say sensible
or meaningful.

Hadley Wickham and the Development of ggplot

A Layered Grammar of Graphics

Hadley WICKHAM

A grammar of graphics is a tool that enables us to concisely describe the components of a graphic. Such a grammar allows us to move beyond named graphics (e.g., the "scatterplot") and gain insight into the deep structure that underlies statistical graphics. This article builds on Wilkinson, Anand, and Grossman (2005), describing extensions and refinements developed while building an open source implementation of the grammar of graphics for R, `ggplot2`.

The topics in this article include an introduction to the grammar by working through the process of creating a plot, and discussing the components that we need. The grammar is then presented formally and compared to Wilkinson's grammar, highlighting the hierarchy of defaults, and the implications of embedding a graphical grammar into a programming language. The power of the grammar is illustrated with a selection of examples that explore different components and their interactions, in more detail. The article concludes by discussing some perceptual issues, and thinking about how we can build on the grammar to learn how to create graphical "poems."

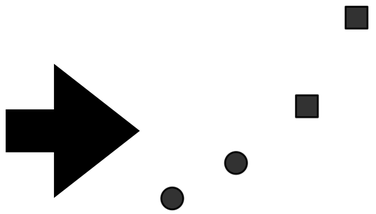
Supplemental materials are available online.

Wickham, 2010

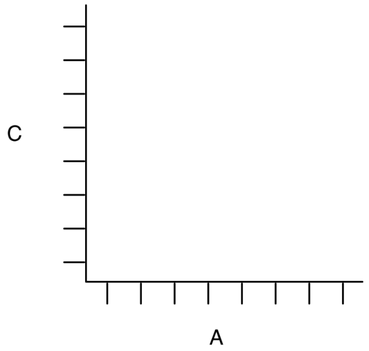
Hadley Wickham and the Development of ggplot

How to build a plot?

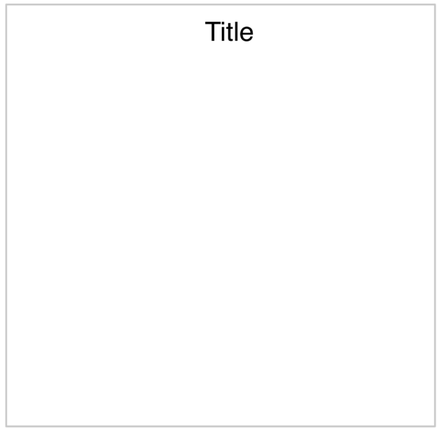
x	y	Shape
25	11	circle
0	0	circle
75	53	square
200	300	square



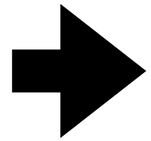
Geometric Objects

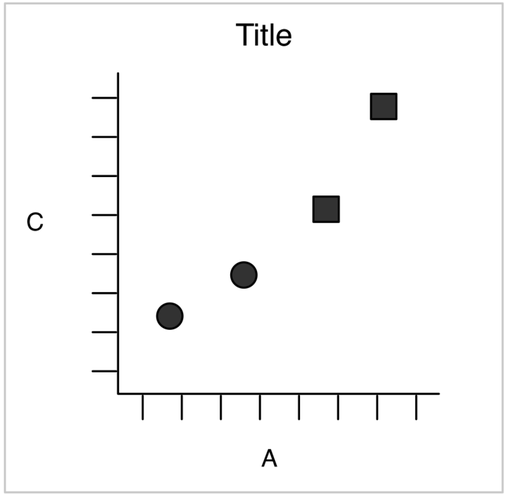


Scales &
Coordinate System



Plot Annotations

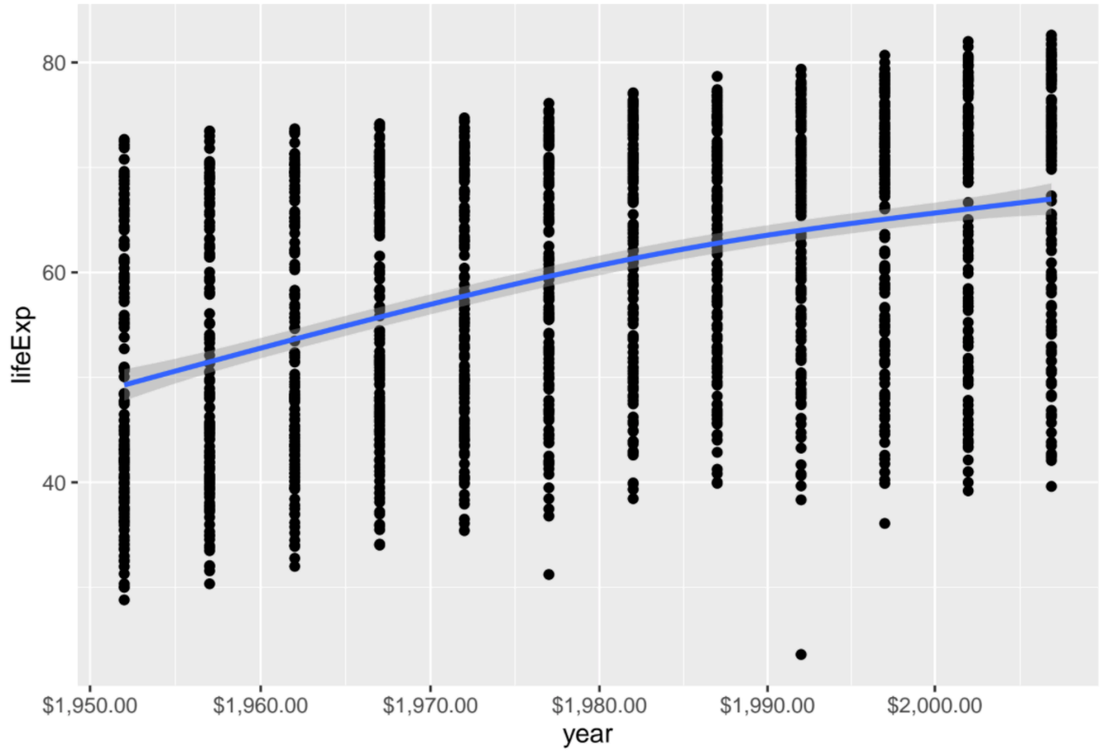




ggplot

BUILDING BLOCKS

- data
- aesthetic mappings
- geometric objects
- statistical transformations (*defaults*)
- scales (*defaults*)
- coordinate systems



ggplot

BUILDING BLOCKS - Geometric Objects (geom)

Geometric objects or geoms are the actual marks we put on a plot.
Examples include:

- points (geom_point, for scatter plots, dot plots, etc)
- lines (geom_line, for time series, trend lines, etc)
- boxplot (geom_boxplot, for, well, boxplots!)
- … and many more!

A plot should ***have at least one geom***, but there is no upper limit.

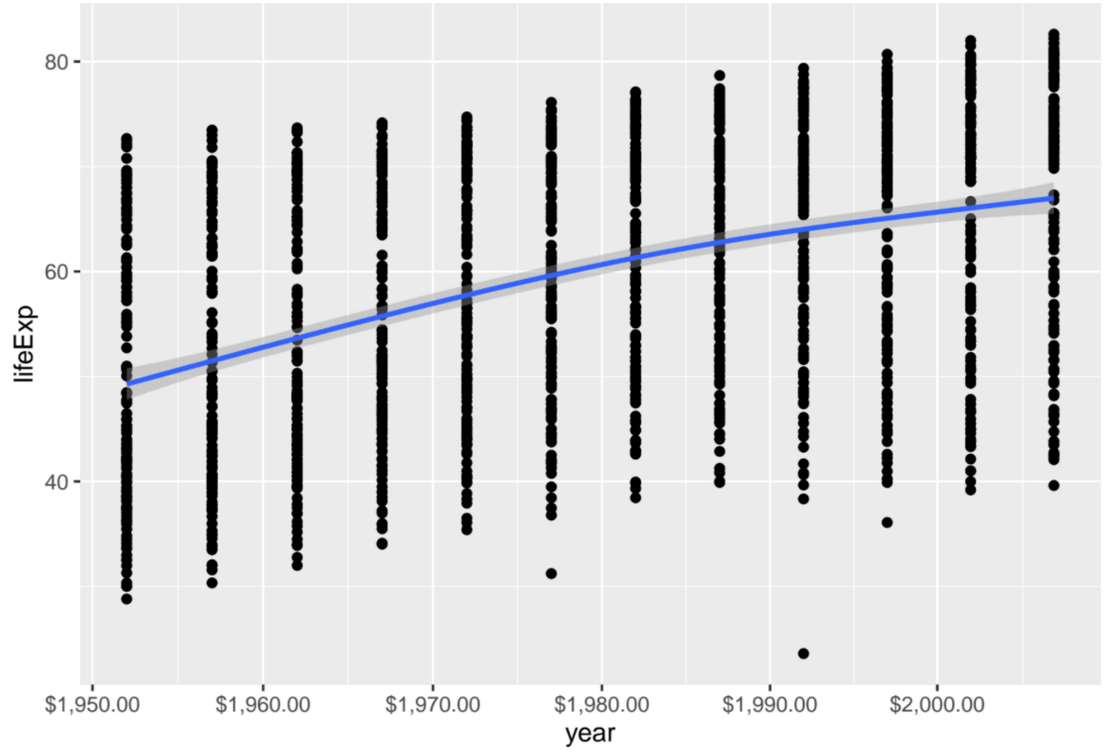
You can ***add a geom to a plot using the + operator***.

You can get a list of available geometric objects using the code:

help.search("geom", package = "ggplot2")

ggplot

BUILDING BLOCKS - Geometric Objects (geom)



ggplot

BUILDING BLOCKS - Aesthetic Mapping (aes)

In ggplot2, Aesthetic means “something you can see”. **Each aesthetic is a mapping between a visual cue and a variable.** *Examples include:*

- position (i.e., on the x and y axes)
- color (“outside” color)
- fill (“inside” color)
- shape (of points)
- line type
- size

Each type of geom accepts only a subset of all aesthetics—refer to the geom help pages to see what mappings each geom accepts.

Aesthetic mappings are set with the aes() function.

ggplot

BUILDING BLOCKS

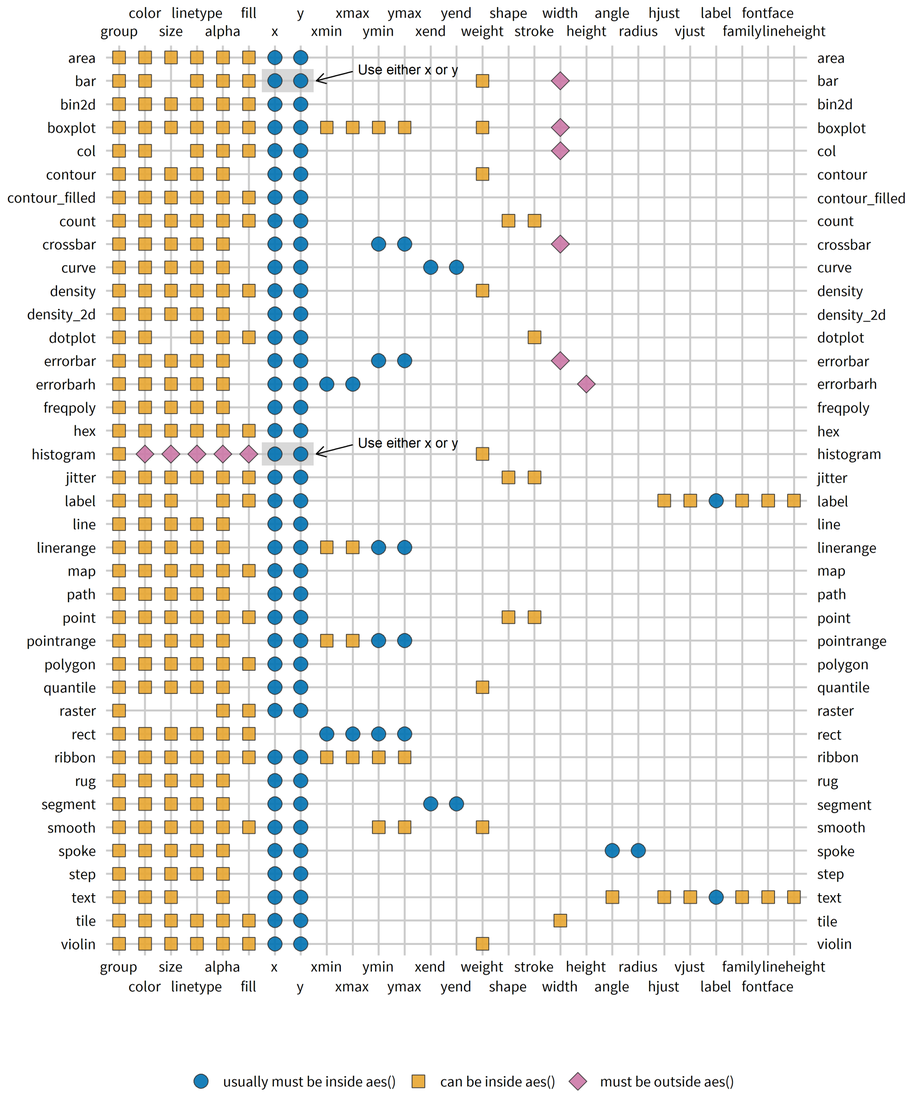
Aesthetic Mapping (aes)

ggplot2 aesthetics cheat sheet

Use this table to find the right aesthetics for your geoms:

Aesthetics that usually must be mapped to the data: use inside aes()
Aesthetics that can be mapped to the data: use in or outside aes()
Aesthetics that cannot be mapped to the data: use outside aes()

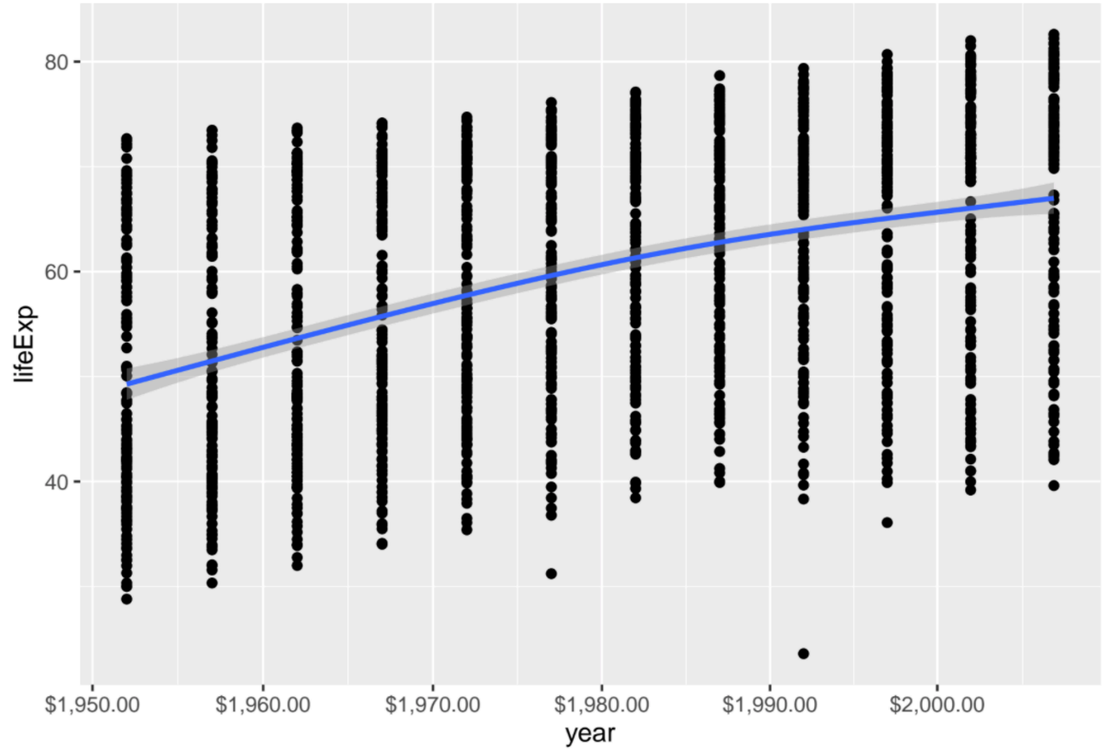
e.g., ggplot(mpg, aes(x = class, y = displ)) + geom_col(aes(fill = class), width = .9)



idea and design: Christian Burkhart
design advice: Ida Aarnio

ggplot

BUILDING BLOCKS - Aesthetic Mapping (aes)



ggplot

BUILDING BLOCKS - Scales (control aesthetic mapping)

Aesthetic mapping (i.e., with aes()) only says that a variable should be mapped to an aesthetic. It doesn't say how that should happen. For example, when mapping a variable to shape with aes(shape = z) you don't say what shapes should be used. Similarly, aes(color = z doesn't say what colors should be used. Describing what colors/shapes/ sizes etc. to use is done by modifying the corresponding scale. In ggplot2, scales include:

- position
- color, fill, and alpha
- size
- shape
- linetype

Scales are modified with a series of functions using a *scale_<aesthetic>_<type>* naming scheme. Try typing *scale_<tab>* to see a list of scale modification functions.

ggplot

BUILDING BLOCKS - Scales (control aesthetic mapping)

Common Scale Arguments

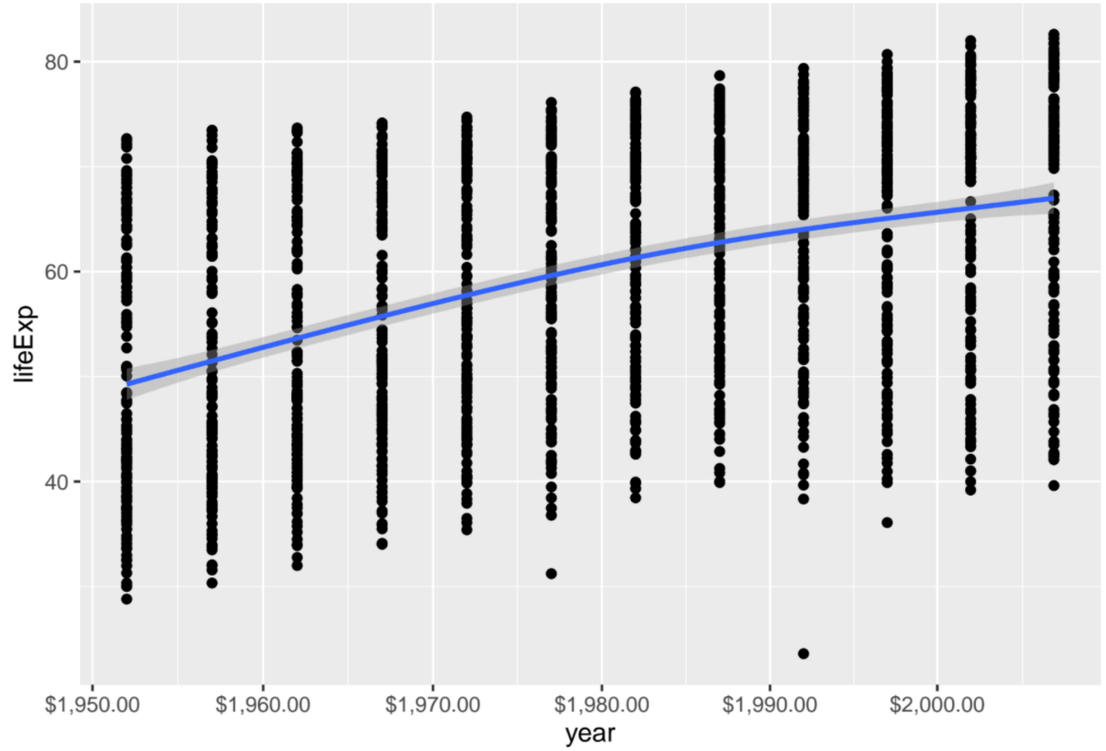
The following arguments are common to most scales in ggplot2:

- name: the first argument specifies the axis or legend title
- limits: the minimum and maximum of the scale
- breaks: the points along the scale where labels should appear
- labels: the text that appear at each break

Specific scale functions may have additional arguments; for example, the scale_color_continuous() function has arguments low and high for setting the colors at the low and high end of the scale

ggplot

BUILDING BLOCKS - Scales (control aesthetic mapping)



ggplot

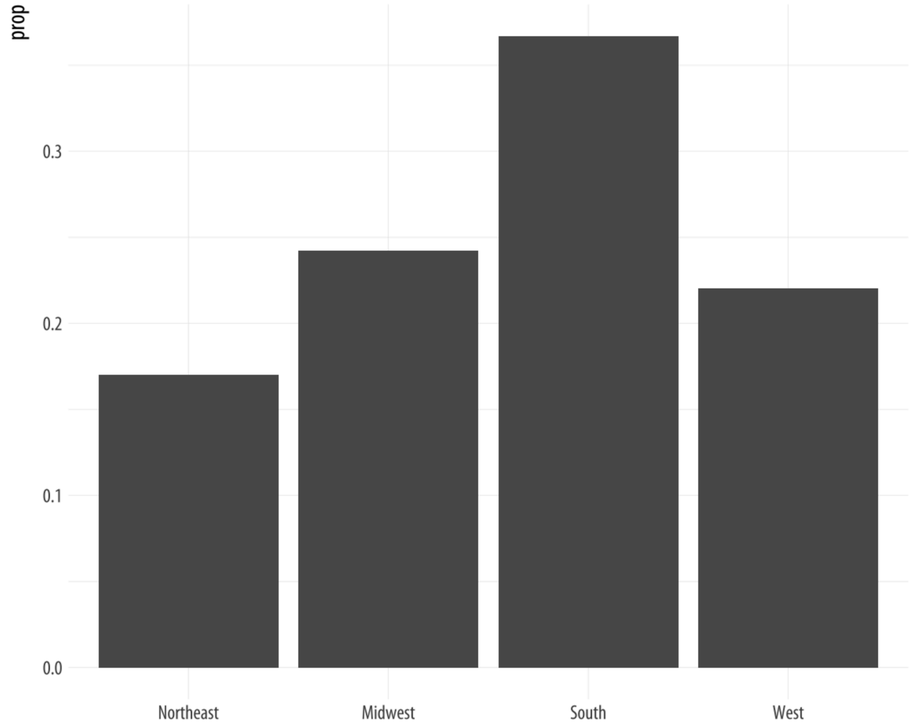
BUILDING BLOCKS

- data
- aesthetic mappings
- geometric objects
- statistical transformations
- scales
- coordinate systems

ggplot

BUILDING BLOCKS

- data
- aesthetic mappings
- geometric objects
- statistical transformations
- scales
- coordinate systems

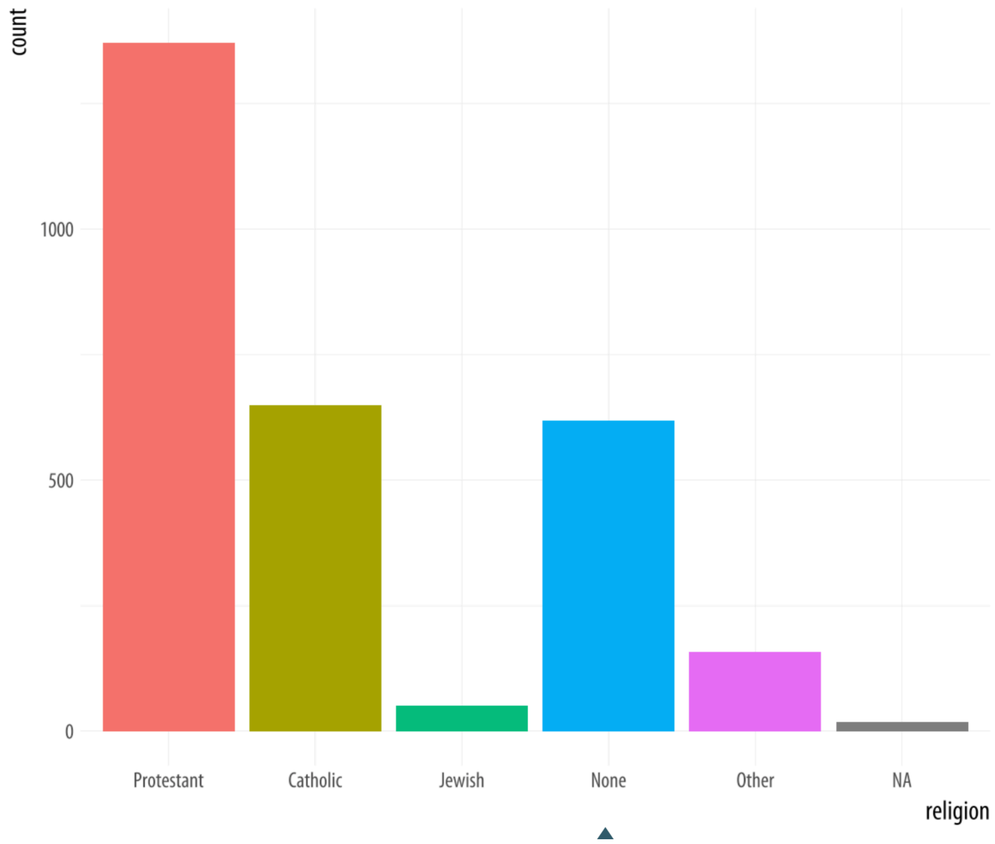



```
ggplot(data = gss_data, mapping = aes(x = bigregion)) +
  geom_bar(mapping = aes(y = ..prop.., group = 1))
```

ggplot

BUILDING BLOCKS

- data
- aesthetic mappings
- geometric objects
- statistical transformations
- scales
- coordinate systems

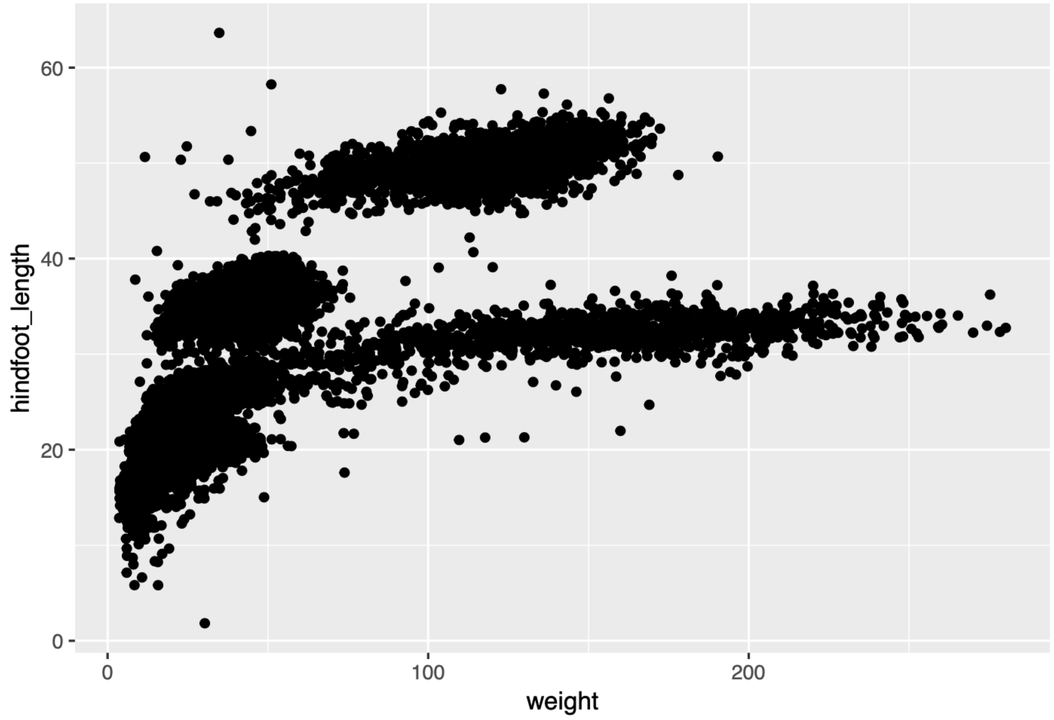



```
ggplot(data = gss_data, mapping = aes(x = religion, fill = religion)) +
  geom_bar() +
  guides(fill = FALSE)
```

ggplot

BUILDING BLOCKS

- data
- aesthetic mappings
- geometric objects
- statistical transformations
- scales
- coordinate systems

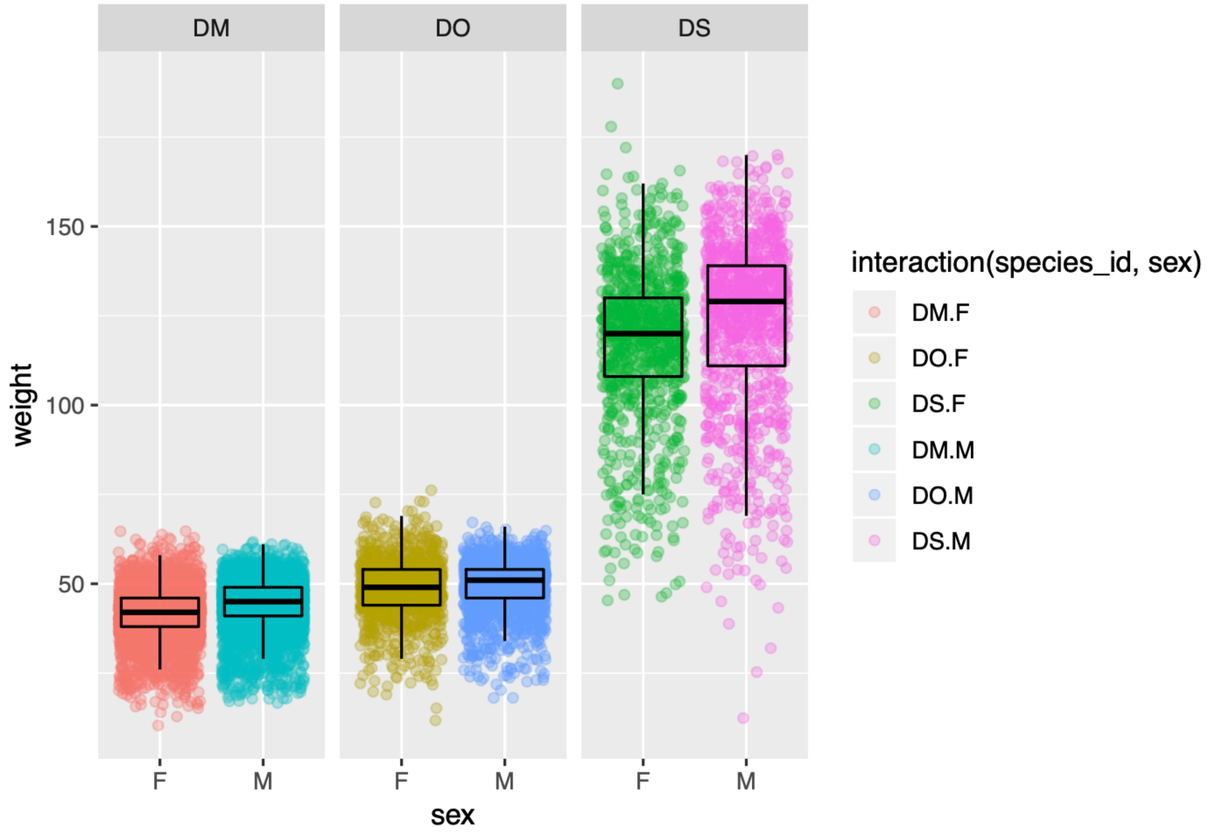



```
ggplot(bio, aes(weight, hindfoot_length)) +
  geom_point(position = position_jitter())
```

ggplot

BUILDING BLOCKS

- data
- aesthetic mappings
- geometric objects
- statistical transformations
- scales
- coordinate systems
- position adjustments




```
ggplot(subset(bio, species_id %in% c("DO", "DM", "DS") & sex %in% c("F", "M")),
       aes(x = sex, y = weight,  color = interaction(species_id,sex))) + facet_wrap( ~ species_id) +
       geom_point(alpha = 0.3, position = "jitter") +
       geom_boxplot(alpha = 0, colour = "black")
```

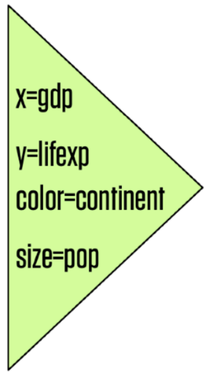
ggplot's FLOW OF ACTION

1. Tidy Data

gdp	lifexp	pop	continent
340	65	31	Euro
227	51	200	Amer
909	81	80	Euro
126	40	20	Asia

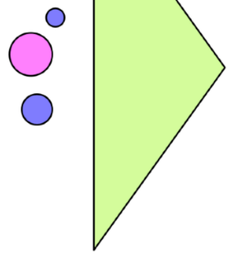
ggplot(data = gapminder,

2. Mapping



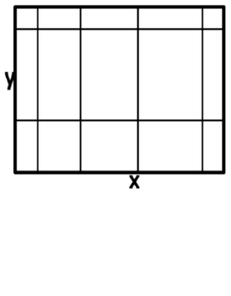
mapping =
aes(x = gdp,
y = lifespan,
color = continent,
size = pop))

3. Geom



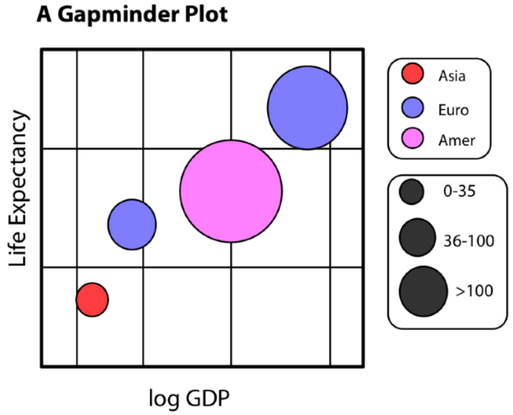
geom_point()

4. Co-Ordinates, Scales



coord_cartesian()
scale_x_log10()

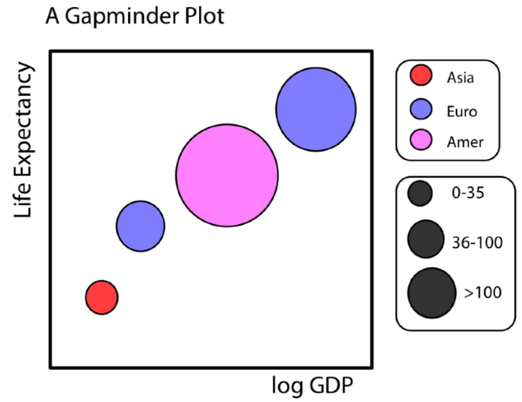
5. Labels & Guides



labs()

guides()

6. Themes



theme_minimal()

15 min break

Next - TidyTuesday workshop

https://mobile.twitter.com/jakepscott2020/status/1304124131558985728/photo/1

https://github.com/rfordatascience/tidytuesday/tree/master/data/2020/2020-09-08

https://docs.google.com/document/d/1Ps_zD629ZAfrsTNu4yQWkoR12QtOHzQdGAiGw6NwhTI/edit?usp=sharing