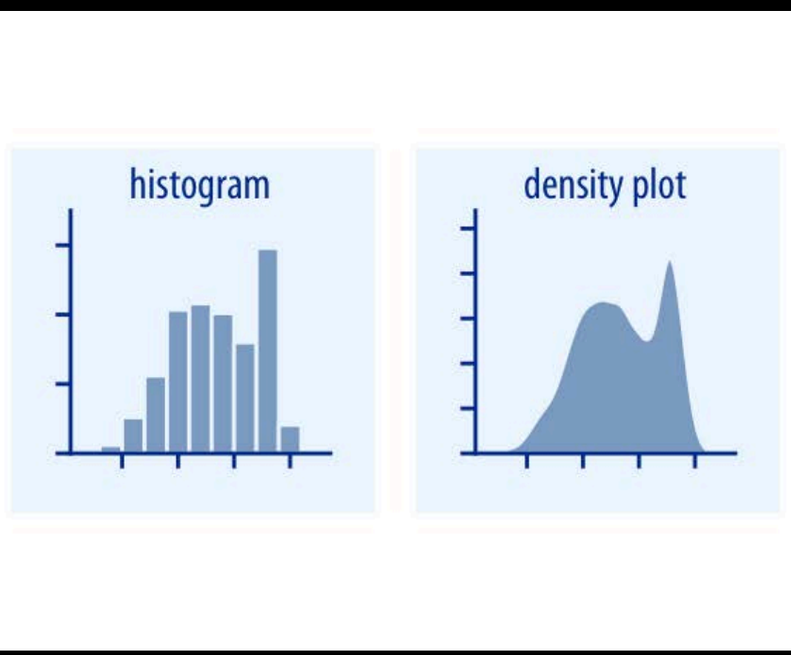
ANT 6973: DATA VISUALIZATION AND EXPLORATION

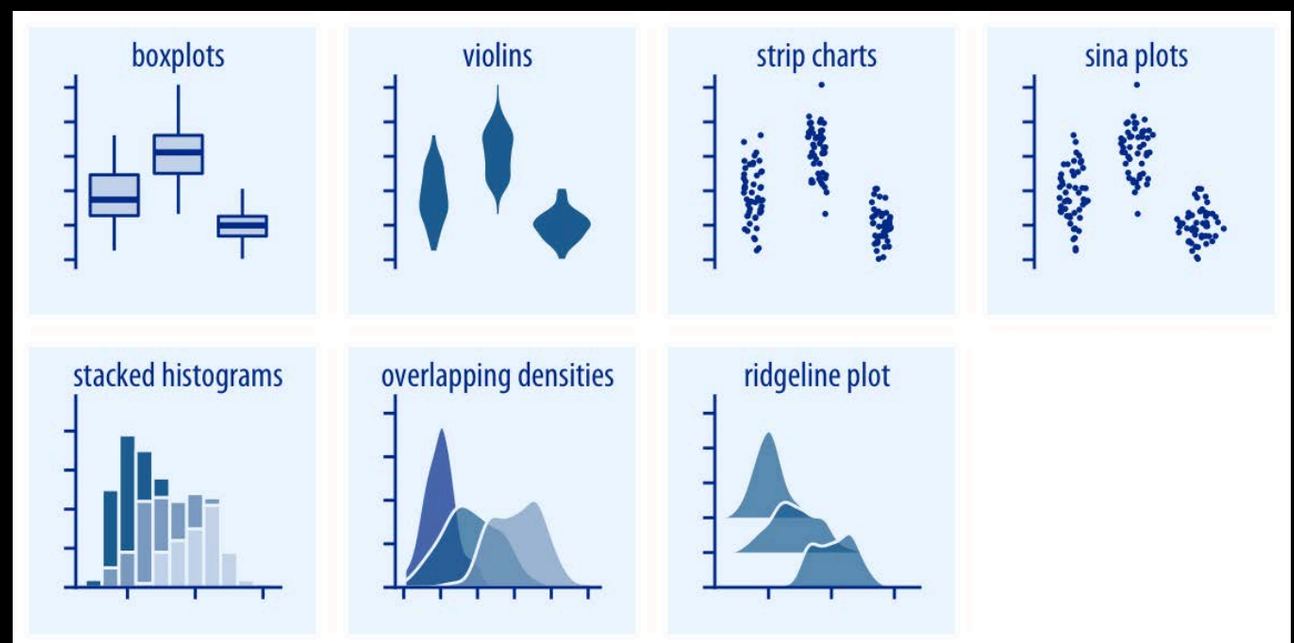
VISUALIZING RELATIONSHIPS AND CHANGE OVER TIME

PREVIOUSLY...

Single distributions



Multiple distributions



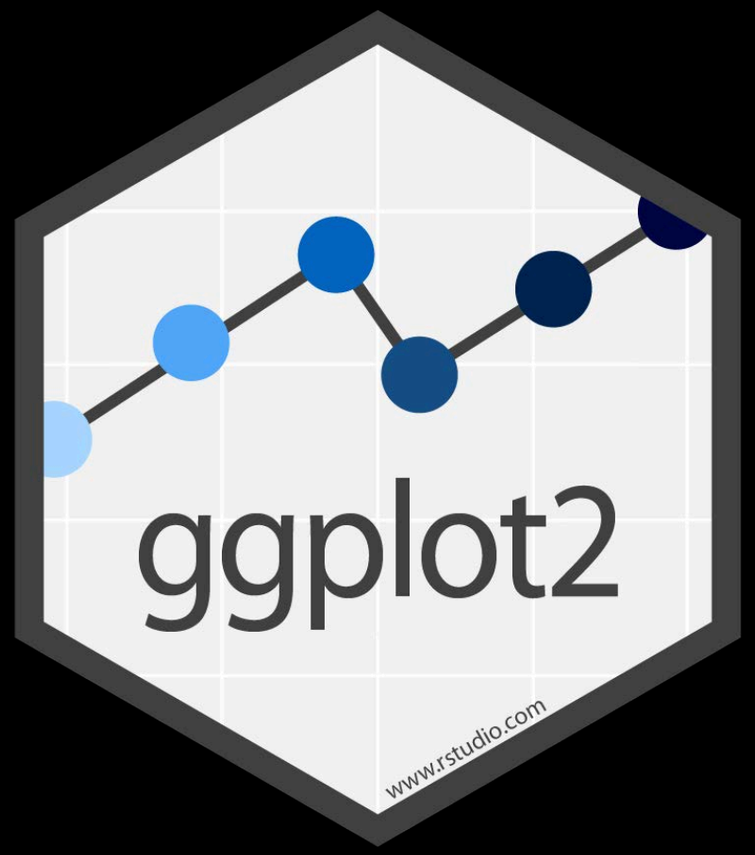
TODAY'S TOPICS

- Visualizing relationships between variables
- Visualizing change over time
- Activities:
  - Baby names
  - Katrina
  - Gender pay gap (if there's time)

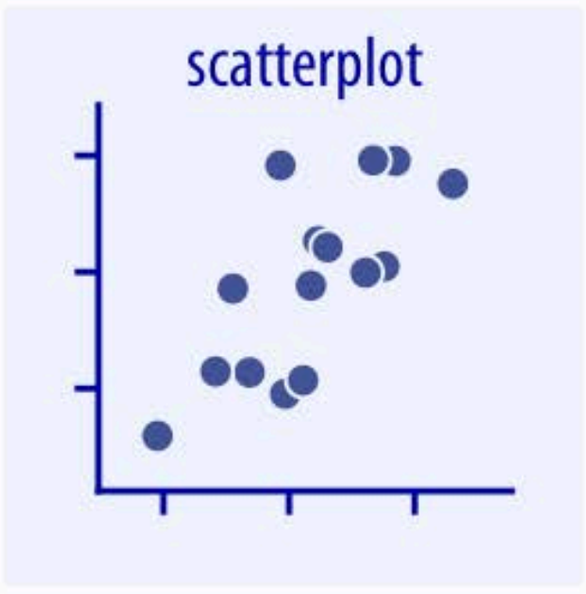
NEW SKILLS

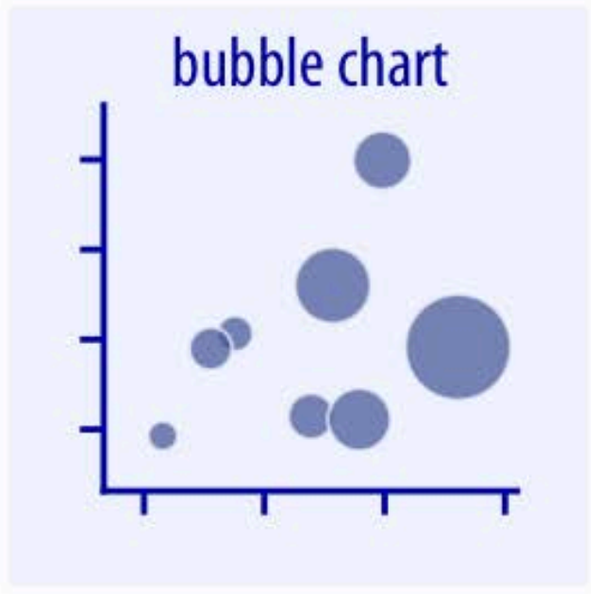
- More practice filtering: multiple and nested logical tests
- Annotations, including positioning and justifying text and other marks
- Extend the axes with `expand_limits()`
- A glimpse of ggplot2's spatial plotting capabilities
- Axis scales: breaks and labels
- Paste text to data with the `glue` package
- Fine-tuning appearance to make a publication-ready plot

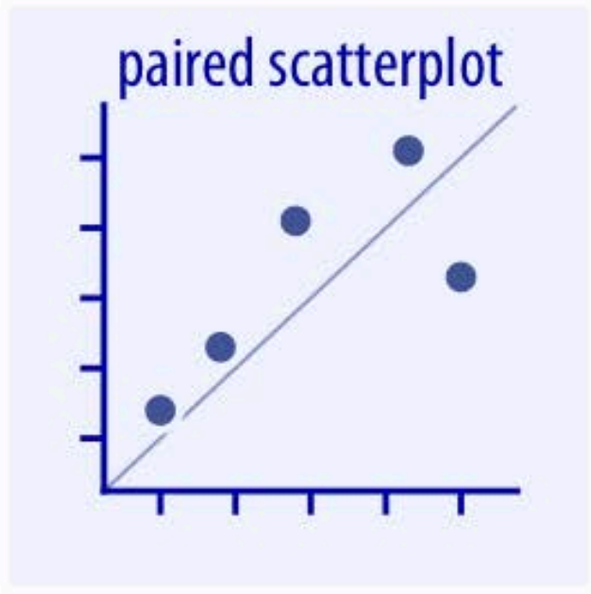
VISUALIZING RELATIONSHIPS AND CHANGE OVER TIME

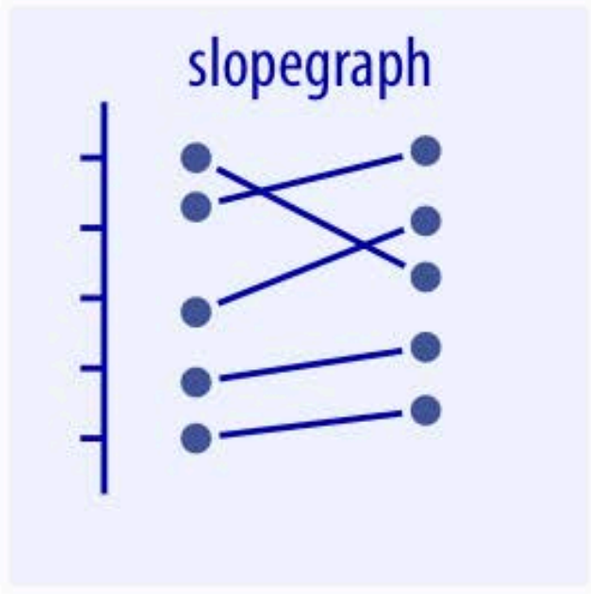


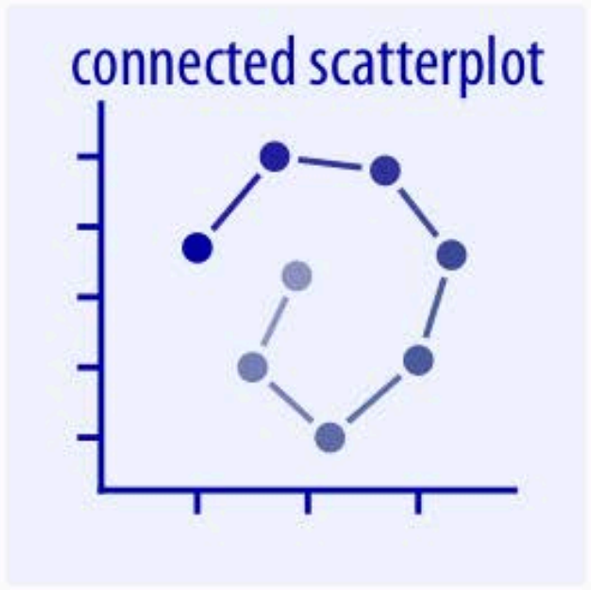
VISUALIZING RELATIONSHIPS AND CHANGE OVER TIME







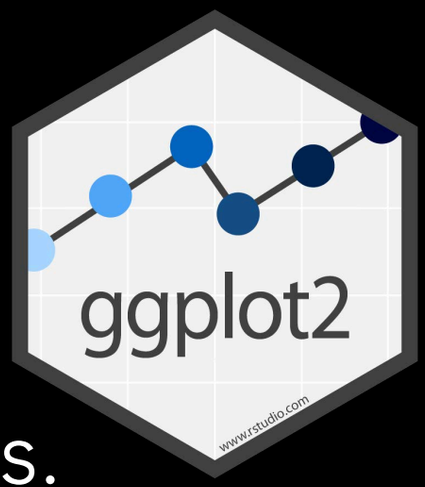




SCATTERPLOTS

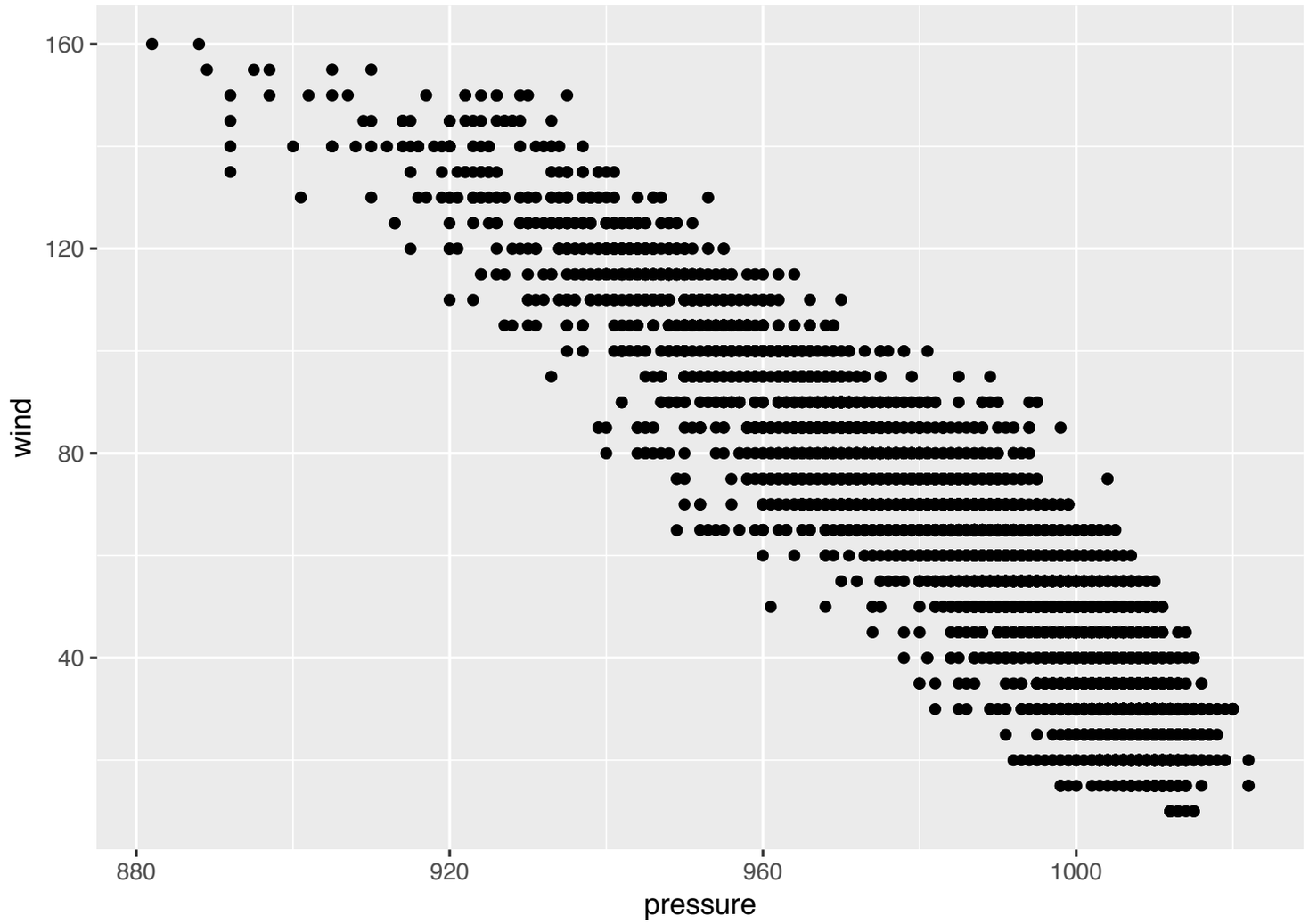
- Scatterplots are your bread and butter for examining relationships between two numerical variables.

SCATTERPLOTS

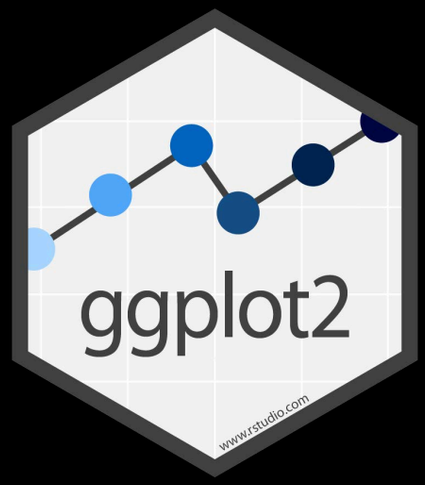


- We have already practiced making scatterplots.

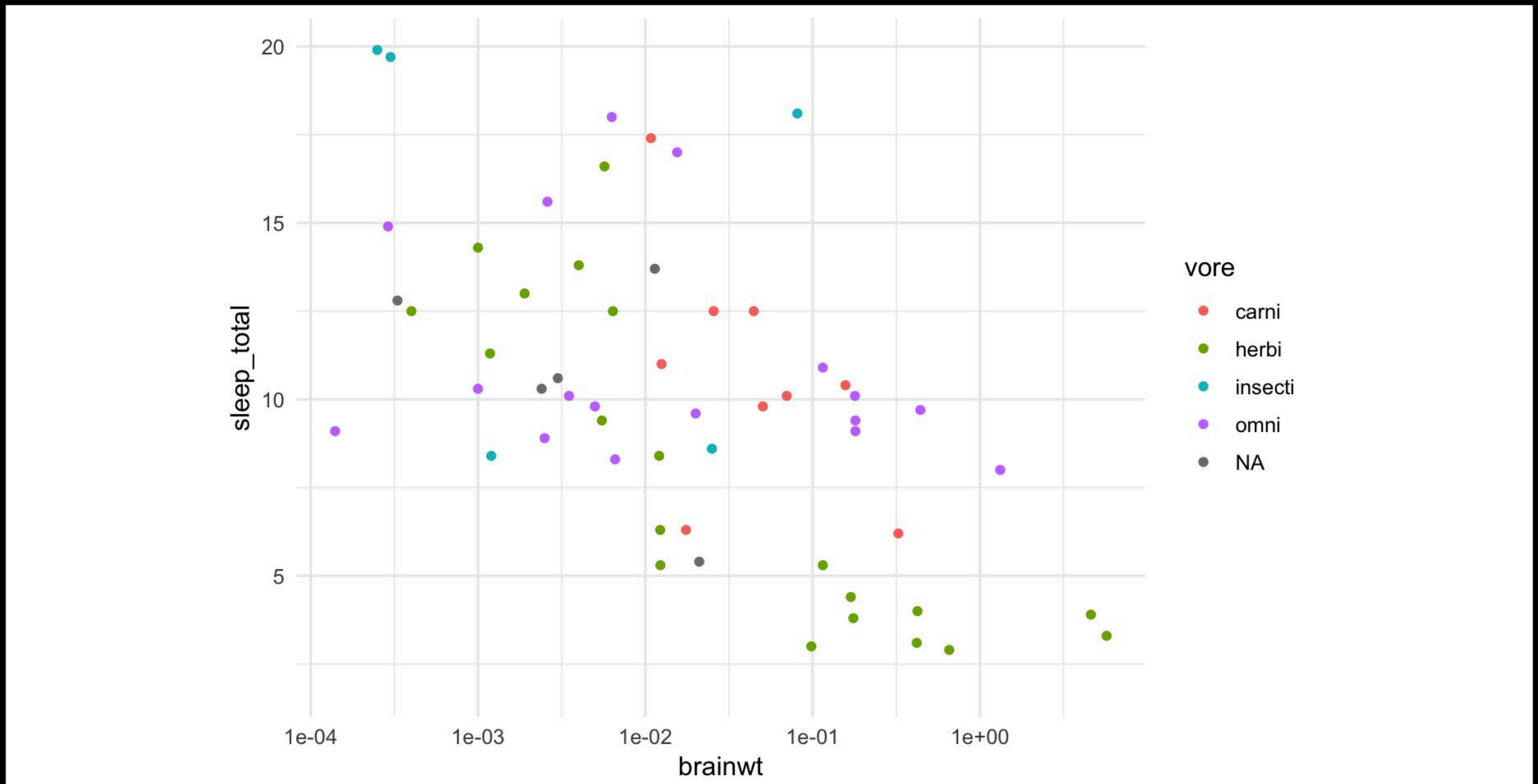
```
ggplot(storms, aes(x = pressure, y = wind)) +
  geom_point()
```



SCATTERPLOTS

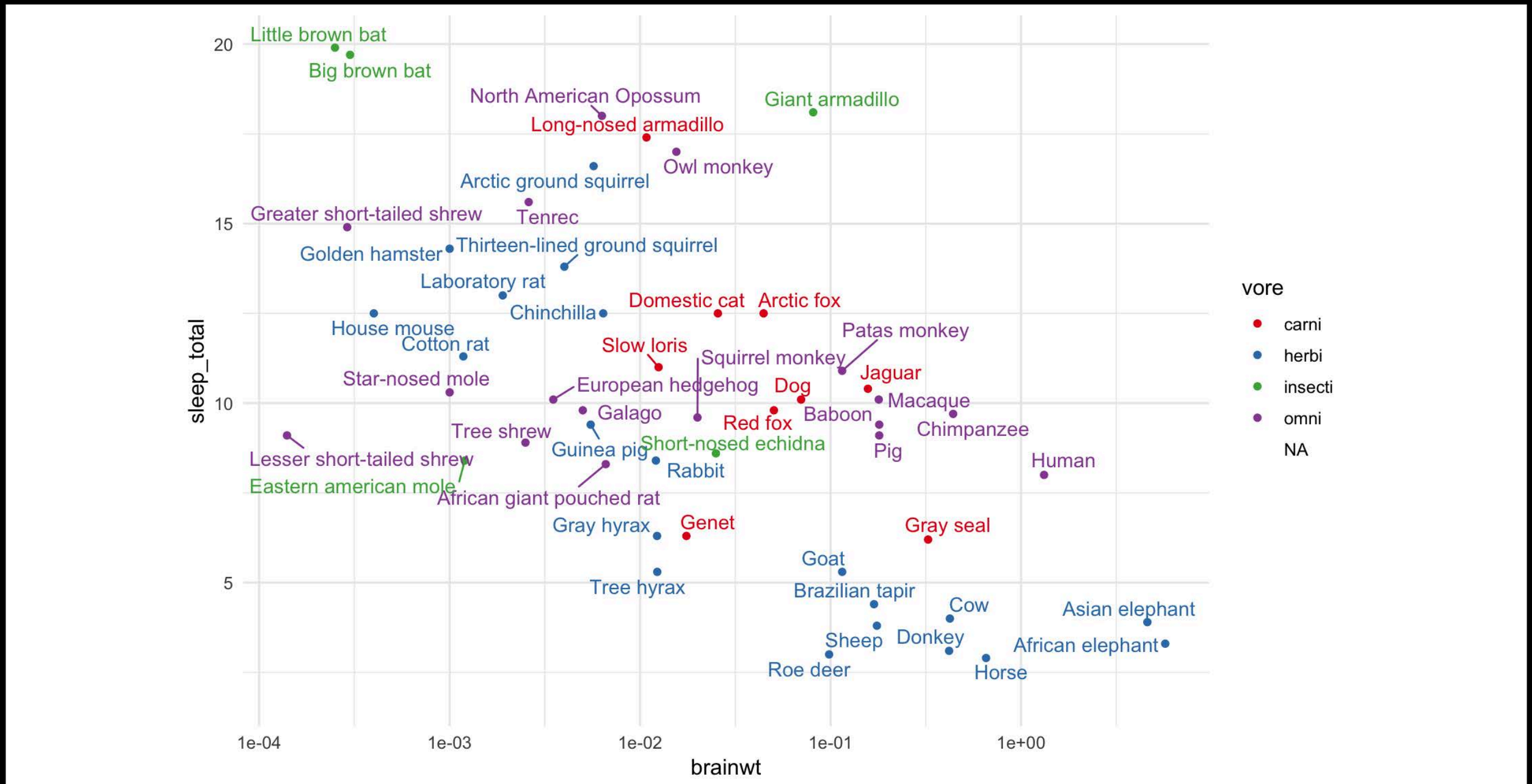


- Third *categorical* variable can be represented as color or shape.



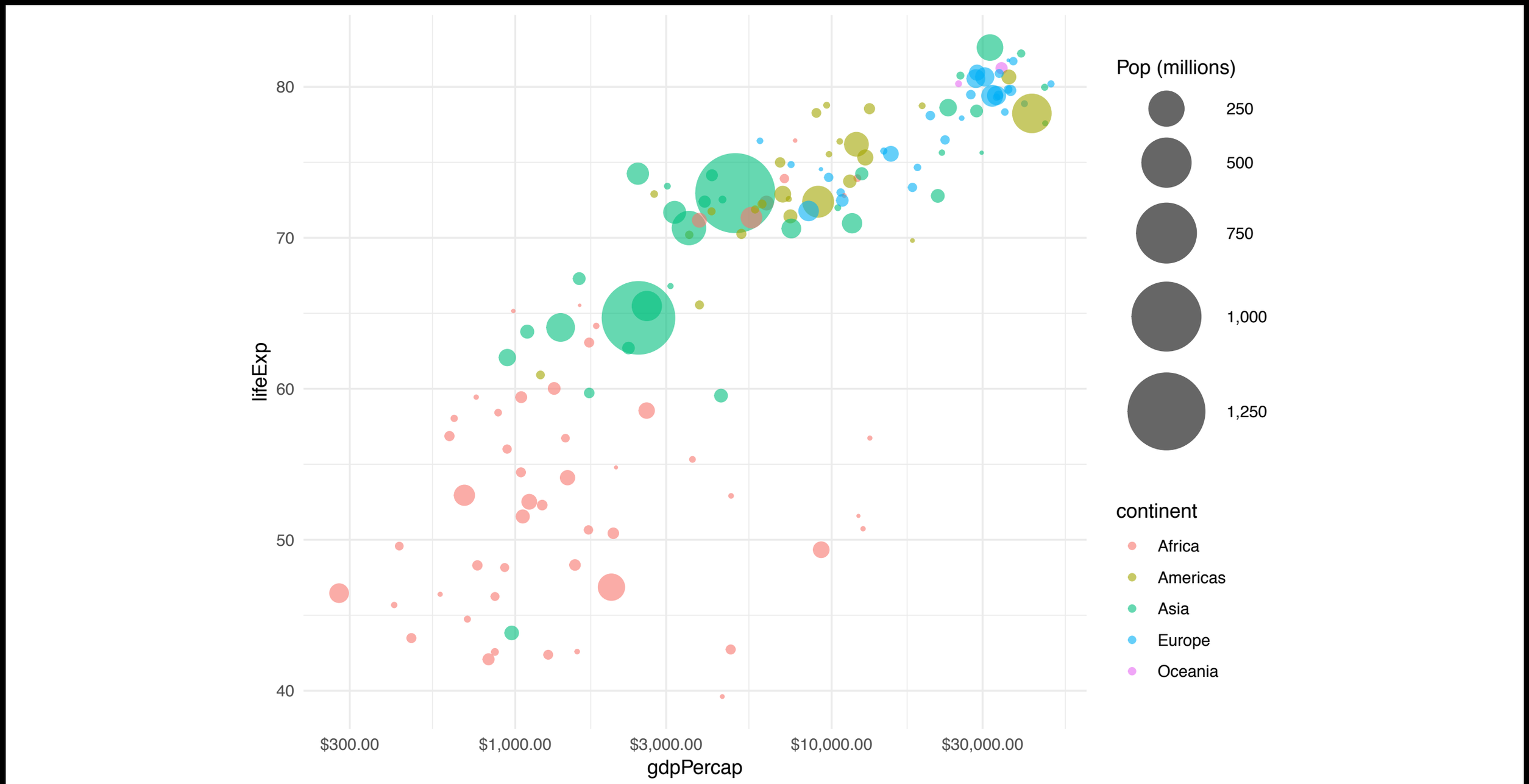
SCATTERPLOTS

- Points can be labeled (here using `ggrepel`)

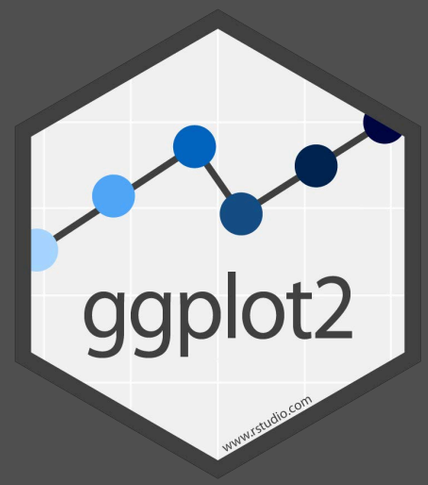


BUBBLE CHART

- Third *numerical* variable can be shown by color or size.



YOUR TURN



- Create a new folder for the week and a new R markdown file.

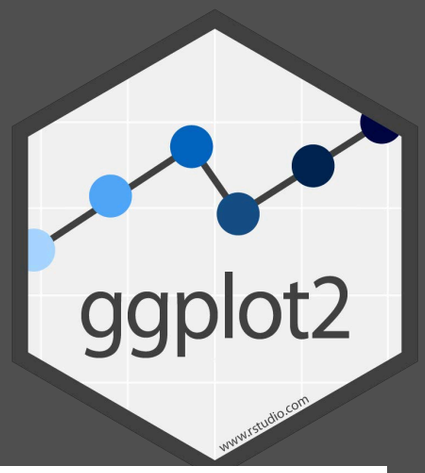
- Create a new R chunk and load the tidyverse package.

- Install the packages `gapminder` and `scales`

- Load the packages `tidyverse`, `gapminder`, and `scales`

- Look at the gapminder data set

YOUR TURN



country	continent	year	lifeExp	pop	gdpPercap
Afghanistan	Asia	1952	28.801	8425333	779.4453
Afghanistan	Asia	1957	30.332	9240934	820.8530
Afghanistan	Asia	1962	31.997	10267083	853.1007
Afghanistan	Asia	1967	34.020	11537966	836.1971
Afghanistan	Asia	1972	36.088	13079460	739.9811
Afghanistan	Asia	1977	38.438	14880372	786.1134
Afghanistan	Asia	1982	39.854	12881816	978.0114
Afghanistan	Asia	1987	40.822	13867957	852.3959
Afghanistan	Asia	1992	41.674	16317921	649.3414
Afghanistan	Asia	1997	41.763	22227415	635.3414

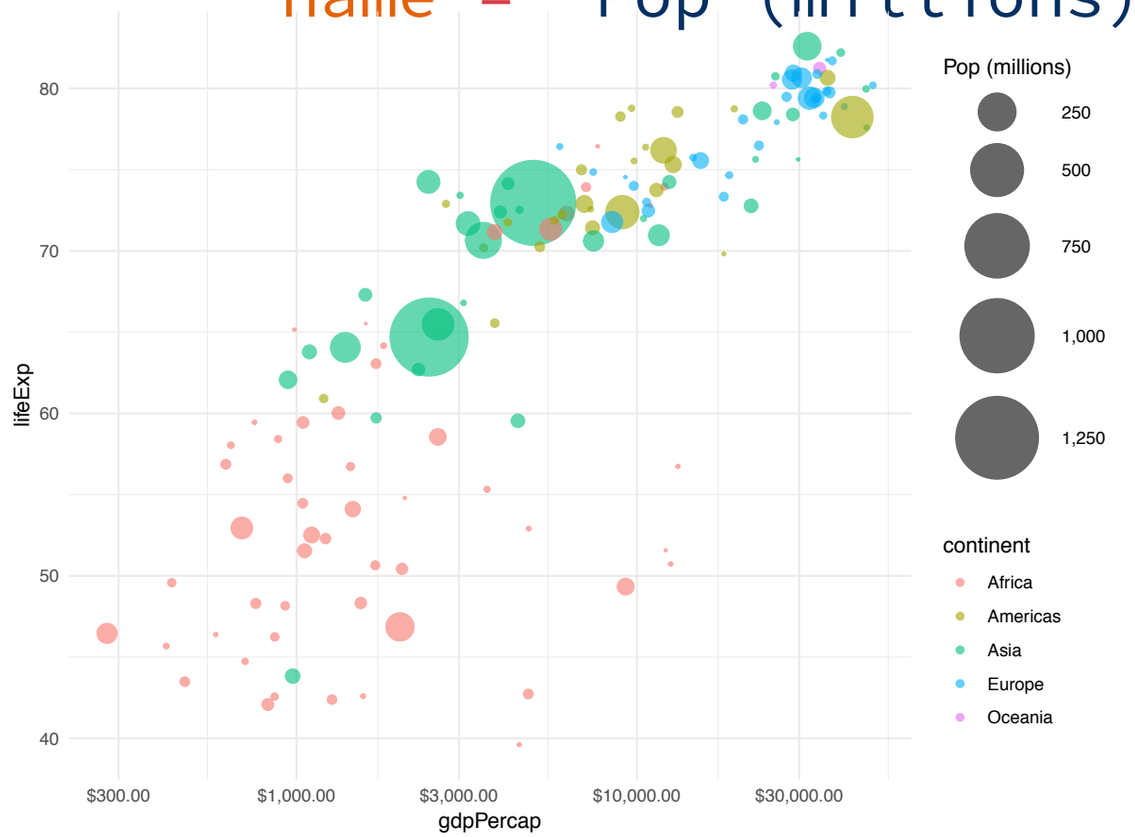
- Plot data lifeExp (y) vs. gdpPerCap (x) for 2007 only
- Divide pop by 1 million and encode as point size; color points by continent
- Use log 10 for x axis

YOUR TURN

```
ggplot(filter(gapminder, year == 2007),
       aes(x = gdpPercap, y = lifeExp,
           size = pop / 1e6, color = continent)) +
  geom_point(alpha = 0.6) +
  scale_x_log10()
```

YOUR TURN

```
ggplot(filter(gapminder, year == 2007),
       aes(x = gdpPercap, y = lifeExp,
           size = pop / 1e6, color = continent)) +
  geom_point(alpha = 0.6) +
  scale_x_log10(labels = dollar) +
  scale_size_area(max_size = 20, labels = comma,
                  name = "Pop (millions)")
```



LINE GRAPHS

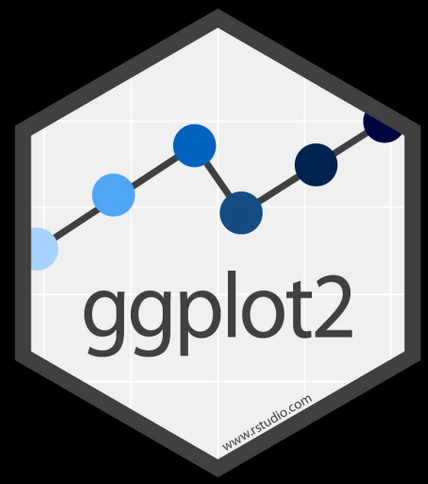
- Typically used when the x-axis represents date/time and the y-axis represents some other numerical variable.

- This is called a *time series*.

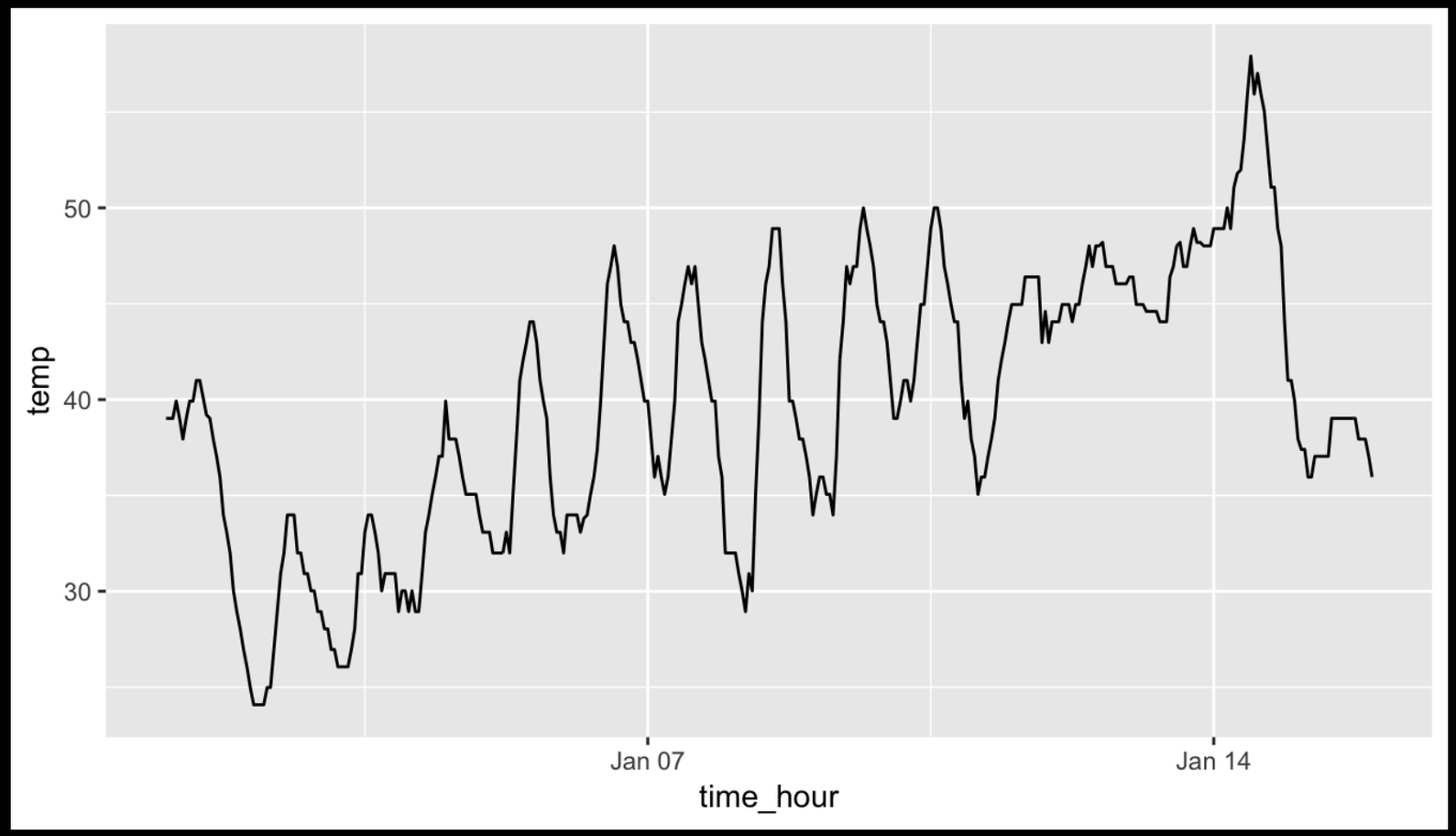
- The line between points implies that they are connected through some defined order.

- Line graphs should not be used when there is not a natural sequential ordering to the the data!

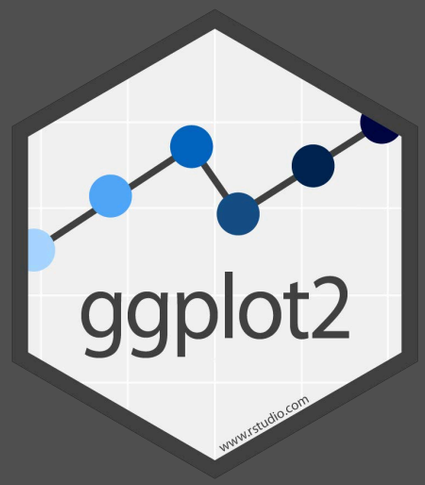
LINE GRAPHS



- Geometric object is `geom_line()`
- One numeric variable to mapped to x
- Another numeric variable to mapped to y



YOUR TURN



- Go to this week's assignments on the course website.

- Install the `babynames` package.

- Follow the instructions in the babynames.Rmd file, and answer the questions about trends in baby names.

- Submit the html report (not the Rmd).

SPATIAL DATA:
A GLIMPSE

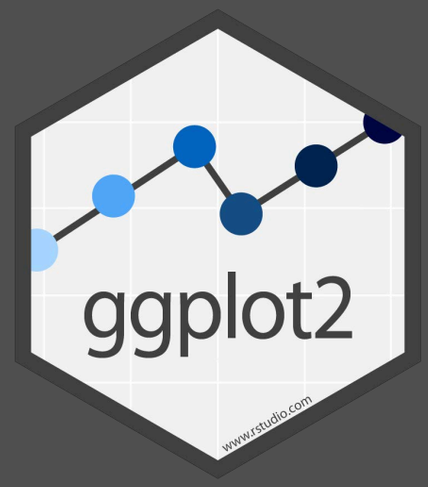
SPATIAL DATA: A GLIMPSE

- ggplot2 has robust spatial plotting capabilities.
- Creating maps in GIS software involves visually preparing a map, analogous to plotting in Excel.
- With ggplot2 (and other R plotting packages), we can draw maps programmatically.

SPATIAL DATA: A GLIMPSE

- The advantages are much the same as with other kinds of plots that we have been making.
    - Elements of a map can be added or removed with ease — R code can be tweaked to make major enhancements with a stroke of a key.
    - Easy to reproduce the same maps for different data sets.
    - It uses that same ggplot2 syntax that is now (sort of) familiar to you.

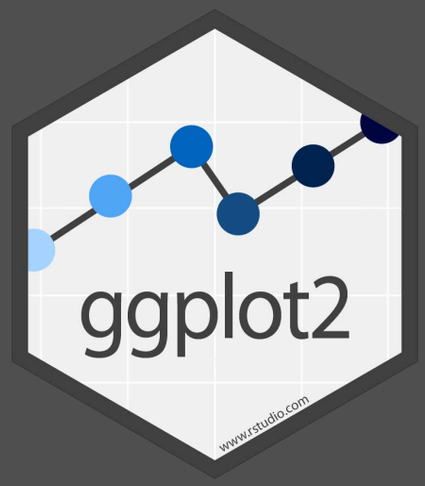
YOUR TURN



- Install the packages `rnaturalearth` and `rnaturalearthdata`.

- Create a new R markdown file for this interactive activity (nothing to turn in).

YOUR TURN

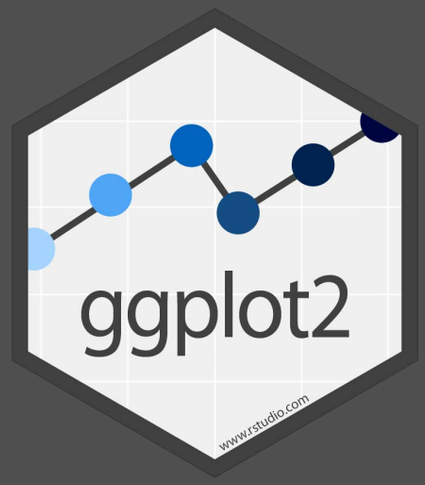


- Let's create a map in ggplot2 using the `storms` dataset that comes with `tidyverse`

name	year	month	day	hour	lat	long	status	category	wind	pressure	ts_diameter	hu_diameter
Amy	1975	6	27	0	27.5	-79.0	tropical depression	-1	25	1013	NA	NA
Amy	1975	6	27	6	28.5	-79.0	tropical depression	-1	25	1013	NA	NA
Amy	1975	6	27	12	29.5	-79.0	tropical depression	-1	25	1013	NA	NA
Amy	1975	6	27	18	30.5	-79.0	tropical depression	-1	25	1013	NA	NA
Amy	1975	6	28	0	31.5	-78.8	tropical depression	-1	25	1012	NA	NA
Amy	1975	6	28	6	32.4	-78.7	tropical depression	-1	25	1012	NA	NA

- Specifically, we want to plot the progression of hurricane Katrina (2005), showing its path and windspeed at each reading.

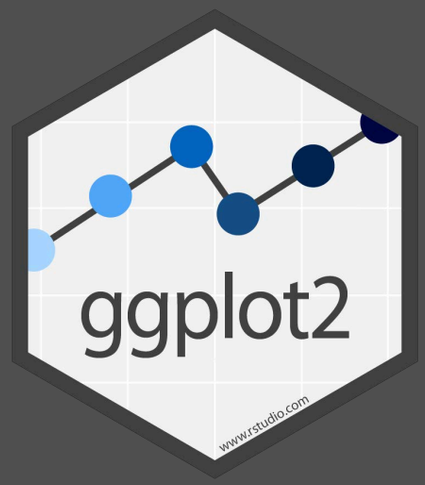
STEP 1: GET THE DATA



- Obtain the data for Katrina only

```
katrina <- filter(storms, name == "Katrina" & year == 2005)
```

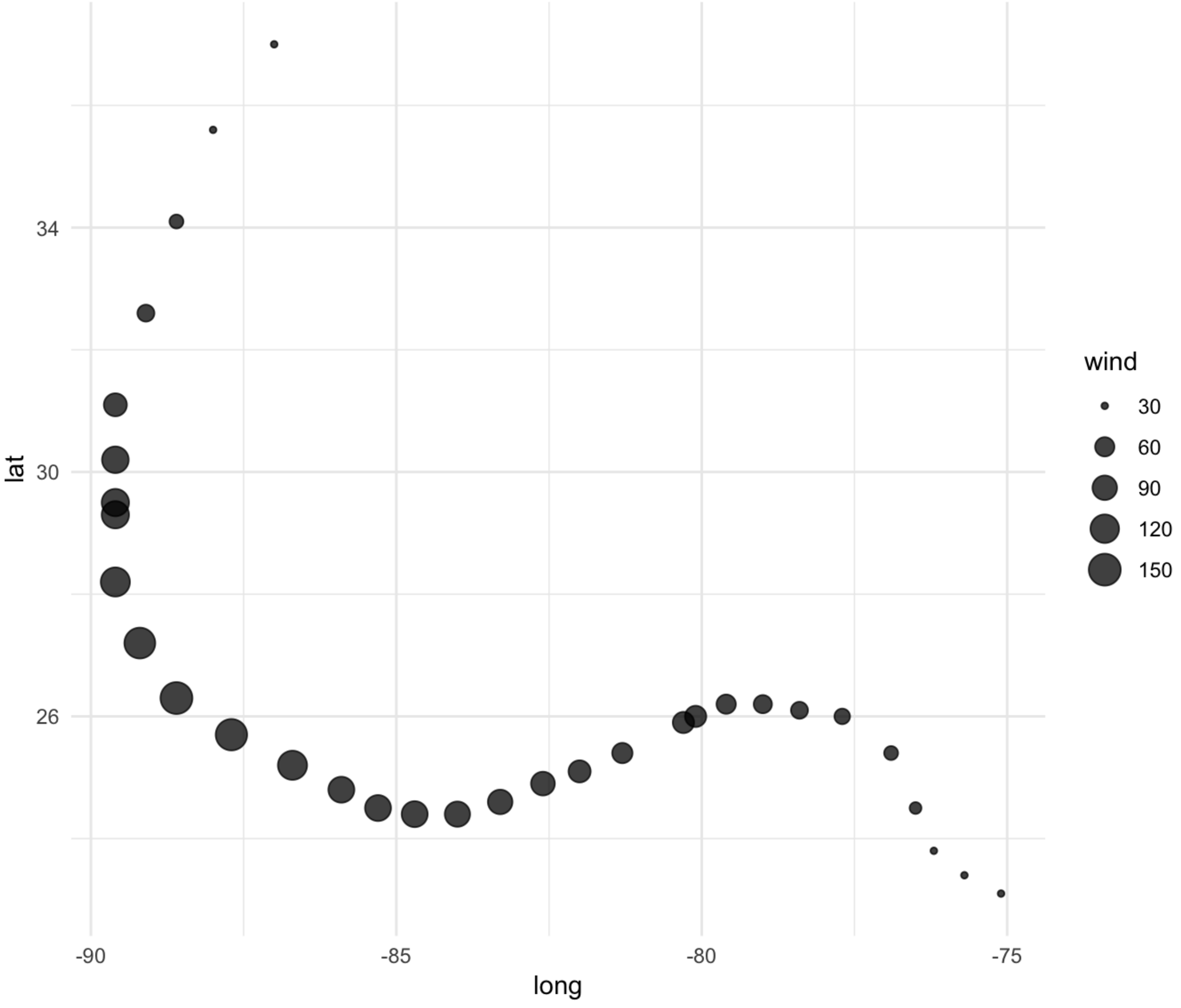
STEP 2: BUBBLE CHART



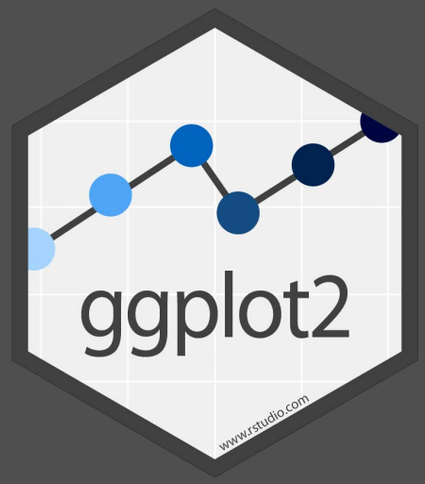
- Create a bubble chart using longitude and latitude for our x-y, and wind for size.

- Use `coord_equal()` to make x and y axis equal units

```
ggplot(katrina, aes(x = long, y = lat, size = wind)) +
  geom_point(alpha = 0.75) +
  coord_equal()
```

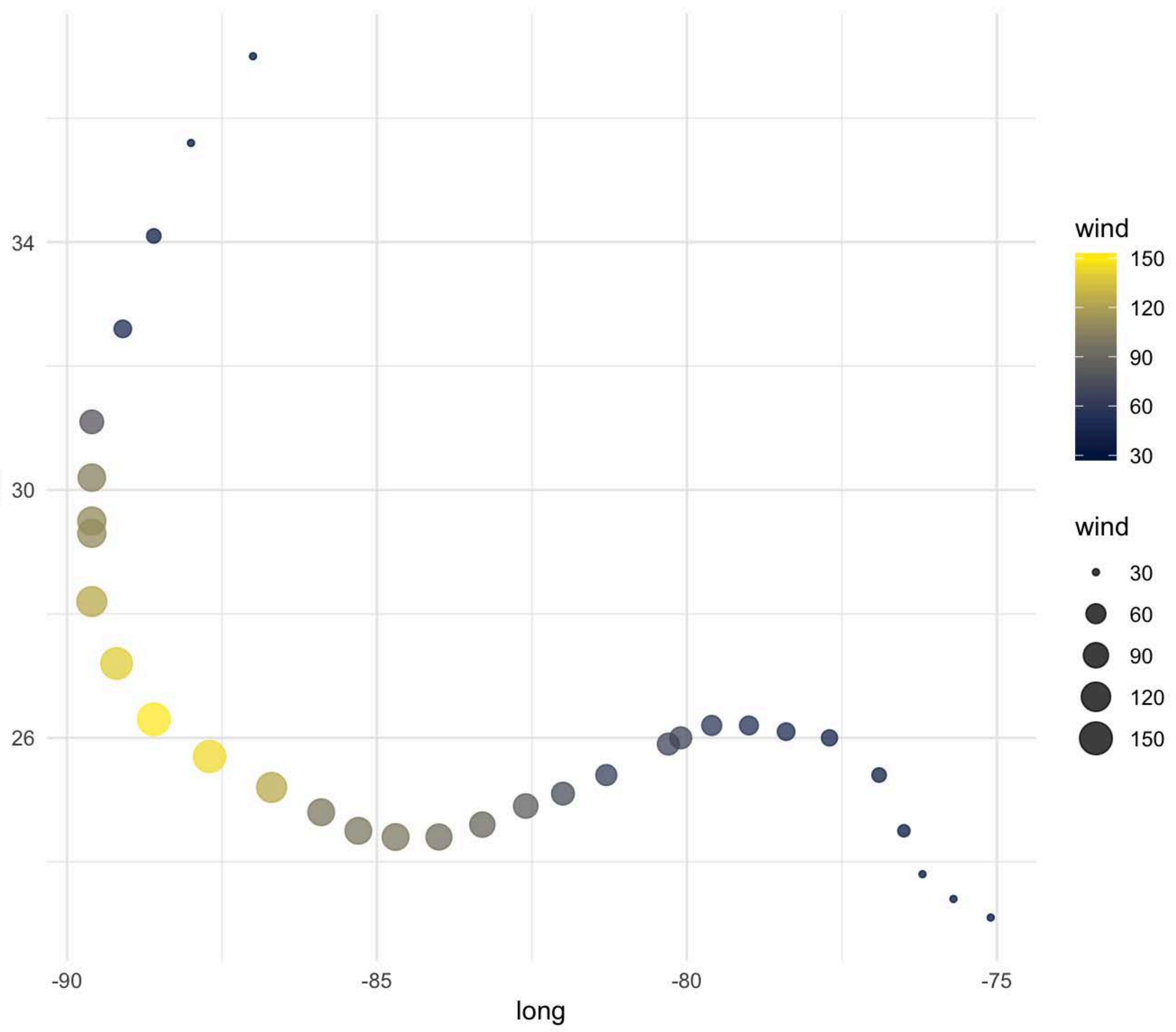

lat
34
30
26
-90
-85
-80
-75
long
wind
30
60
90
120
150

STEP 3: ADD COLOR

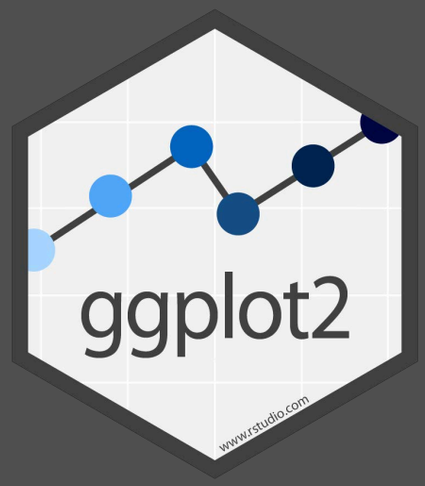


- Map color to wind as well, and use the "cividis" color option.

```
ggplot(katrina, aes(x = long, y = lat,
                    size = wind, color = wind)) +
  geom_point(alpha = 0.75) +
  scale_color_viridis_c(option = "cividis") +
  coord_equal()
```


lat
34
30
26
-90
-85
-80
-75
long
wind
150
120
90
60
30
wind
30
60
90
120
150

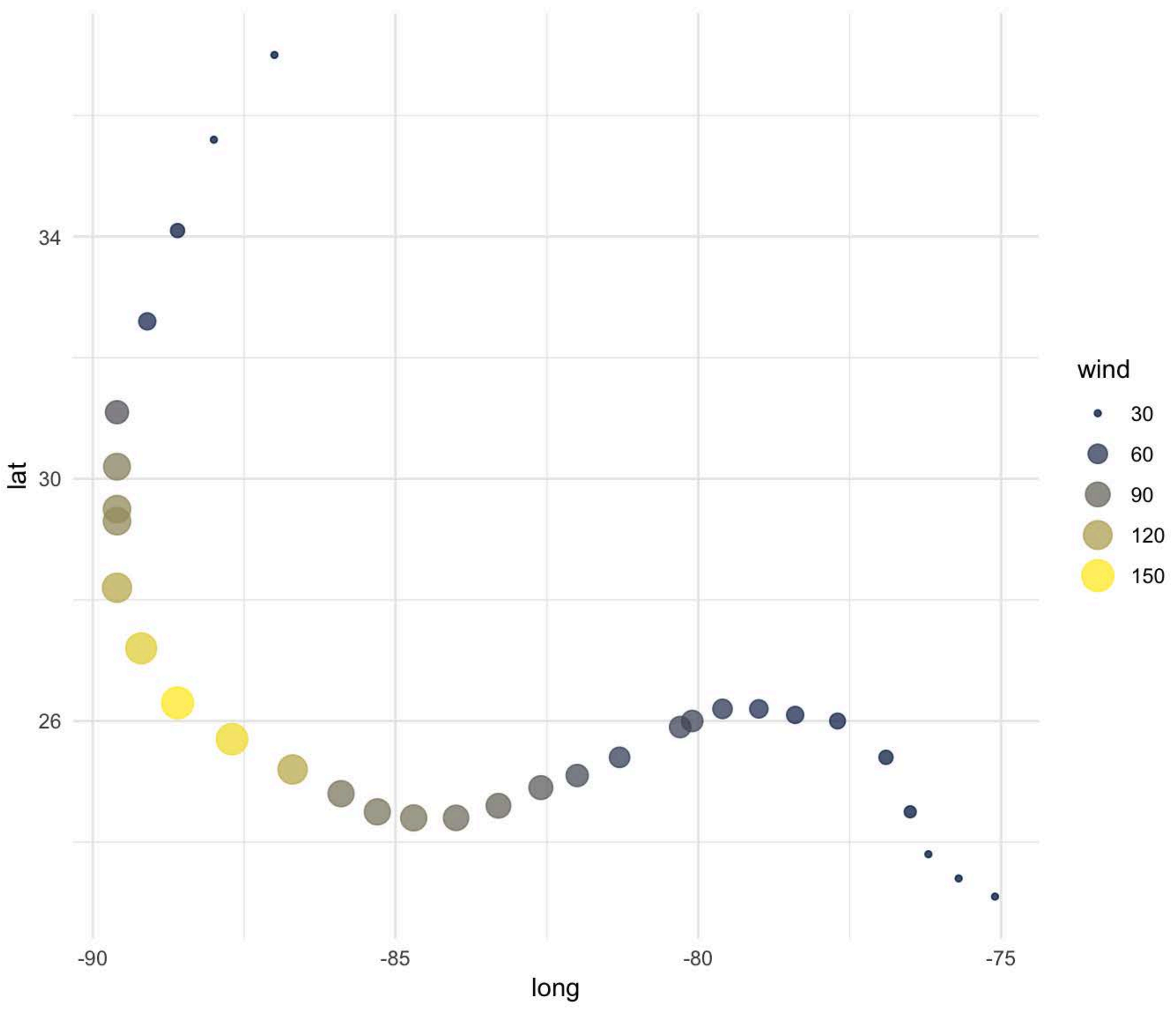
STEP 4: COMBINE LEGENDS



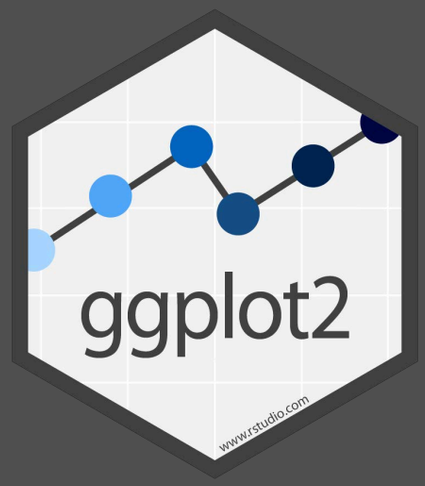
- Combine the legend and color bar using the `guides()` function

  - By default, `scale_colour_viridis_c()` uses a colorbar, but we want it to use a legend (like size)

```
ggplot(katrina, aes(x = long, y = lat,
                    size = wind, color = wind)) +
  geom_point(alpha = 0.75) +
  scale_color_viridis_c(option = "cividis") +
  coord_equal() +
  guides(color = guide_legend())
```


lat
34
30
26
-90
-85
-80
-75
long
wind
30
60
90
120
150

STEP 5: OBTAIN MAP DATA



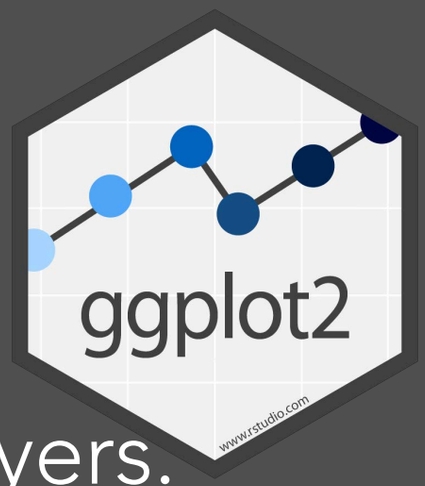
- Load the rnaturalearth package and, when prompted, download the data.

- Maps include:

1. `ne_countries()` for country boundaries
2. `ne_states()` for boundaries within countries
3. `ne_coastline()` for world coastline

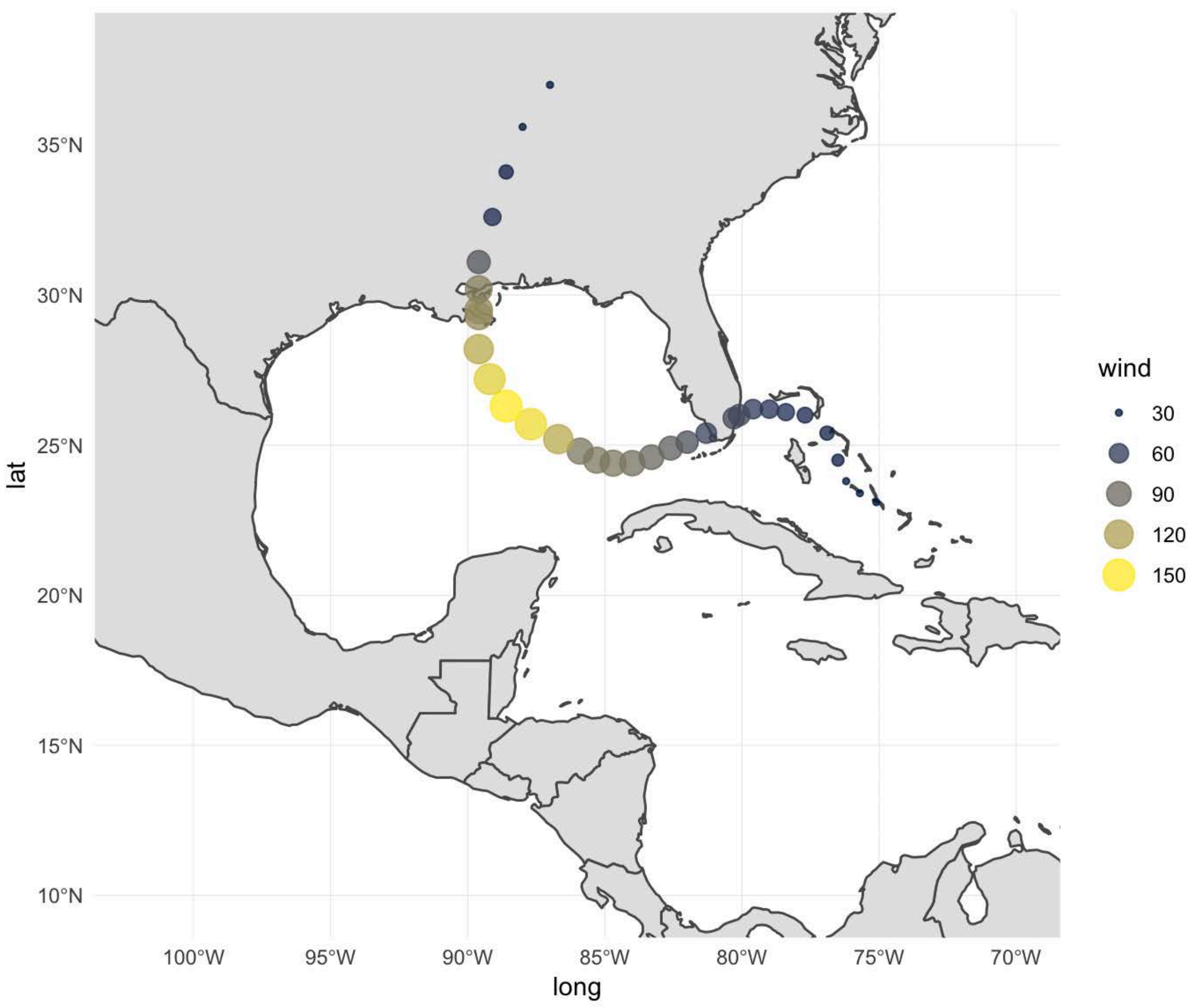
```
world <- ne_countries(scale = "medium", returnclass = "sf")
```

STEP 6: ADD THE MAP

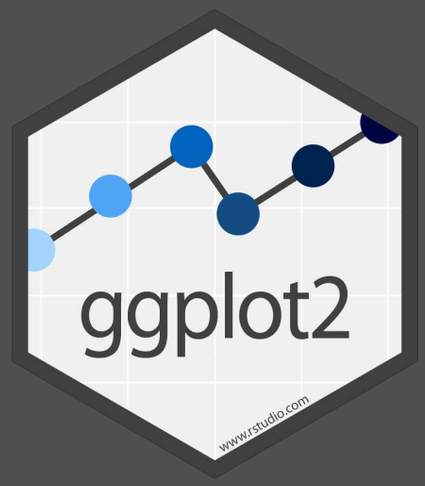


- `geom_sf(data = world)` beneath other layers.
- `coord_sf()` function allows us to "crop" the world map to our area of interest, and provides nice lat/long formatting.

```
ggplot() +
  geom_sf(data = world) +
  geom_point(data = katrina,
             aes(x = long, y = lat, size = wind, color = wind),
             alpha = 0.75) +
  scale_color_viridis_c(option = "cividis") +
  coord_sf(xlim = c(-102, -70), ylim = c(10, 38)) +
  guides(color = guide_legend())
```


35°N
30°N
25°N
20°N
15°N
10°N
lat
100°W
95°W
90°W
85°W
80°W
75°W
70°W
long
wind
30
60
90
120
150

STEP 7: THEME TWEAKS



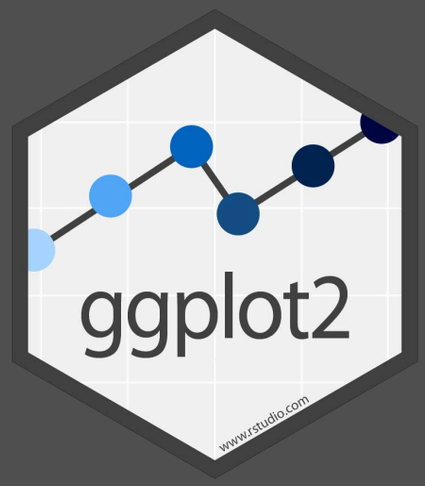
- We will not talk much about customizing the built-in themes.

- But the appearance of just about every non-data element of the plot can be customized using the `theme()` function.

- Lots of examples in the online documentation: https://ggplot2.tidyverse.org/reference/theme.html

- Warning: tons of options; sort of tedious to learn

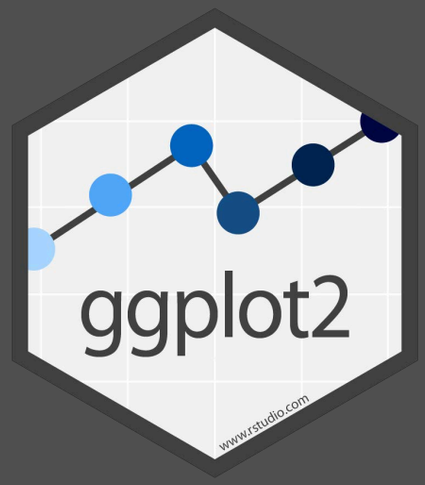
STEP 7: THEME TWEAKS



- Just as an example...

```
ggplot() +
  geom_sf(data = world, fill = "antiquewhite1") +
  geom_point(data = katrina,
             aes(x = long, y = lat, size = wind, color = wind),
             alpha = 0.75) +
scale_color_viridis_c(option = "cividis") +
coord_sf(xlim = c(-102, -70), ylim = c(10, 38), expand = FALSE) +
guides(color = guide_legend()) +
labs(x = "Longitude", y = "Latitude") +
theme(panel.grid.major = element_line(color = gray(0.5),
                                      linetype = "dashed",
                                      size = 0.5),
      panel.background = element_rect(fill = "aliceblue"),
      panel.border = element_rect(fill = NA))
```

STEP 7: THEME TWEAKS



- Just as an ex

Fill color of land

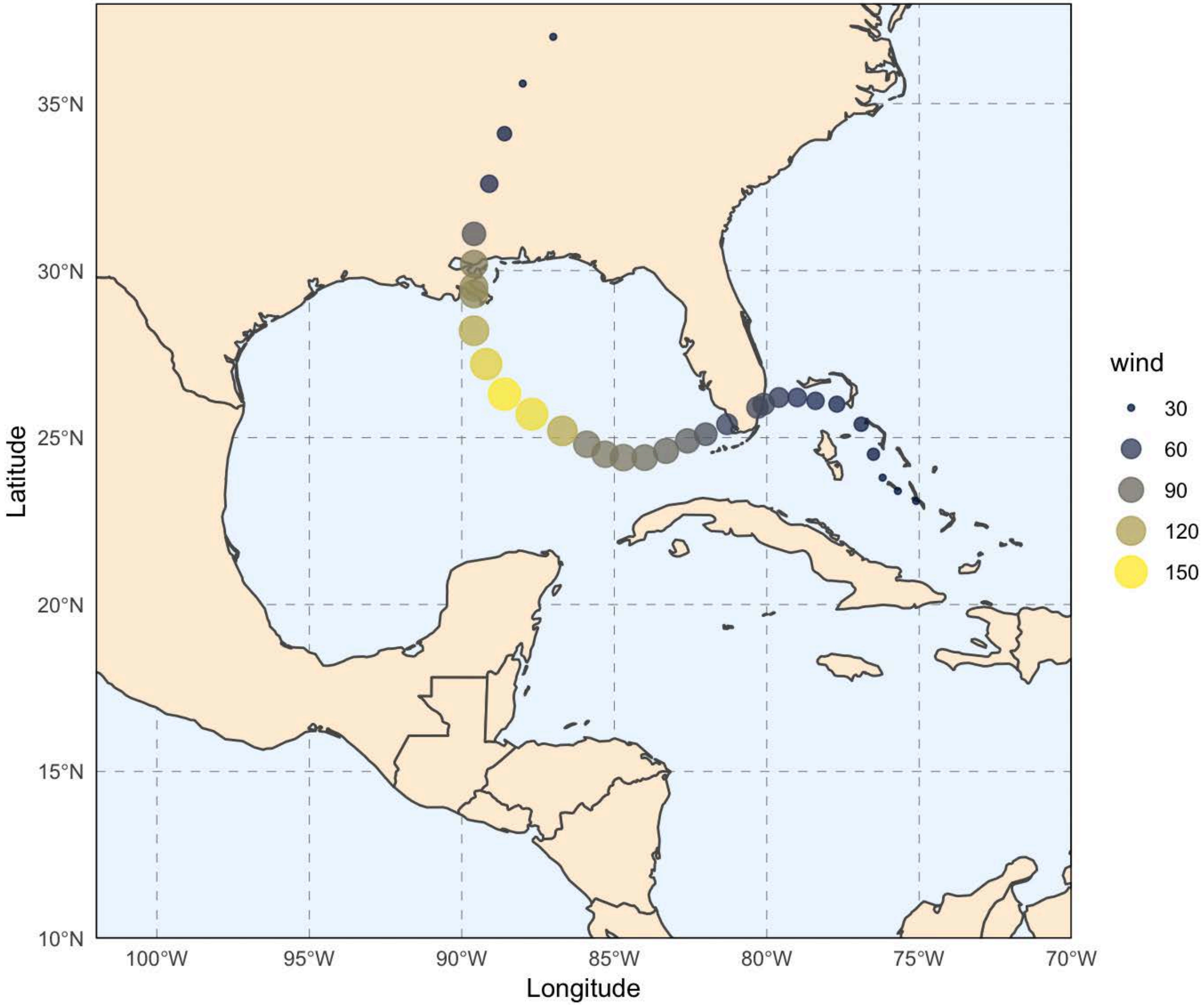
Trim the edges of the axes

Gray dashed graticules (lat/lon grid)

Light blue background for plot area (like water)

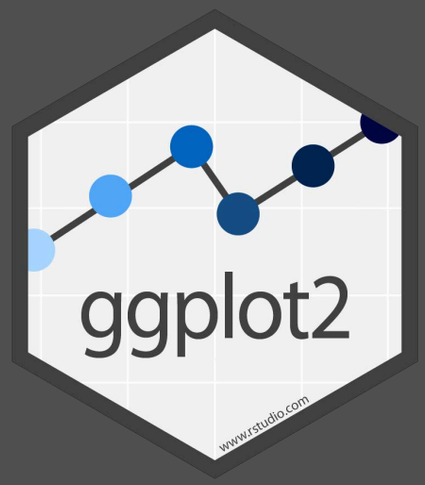
Rectangular border (no fill) around entire panel

```
ggplot() +
  geom_sf(data = world, fill = "antiquewh
  geom_point(data = katrina,
             aes                    ize = wind, color = wind),
             alp
  scale_color_vi                    idis") +
  coord_sf(xlim = c(-102, -70), ylim =                    ALSE) +
  guides(color =
  labs(x = "Long                    "
  theme(panel.gr                    ne(color = gray(0.5),
                                       linetype = "dashed",
                                       size = 0.5),
        panel.background = element_rect(fill = "aliceblue"),
        panel.border = element_rect(fill = NA))
```


35°N
30°N
25°N
20°N
15°N
10°N
100°W
95°W
90°W
85°W
80°W
75°W
70°W
Latitude
Longitude
wind
30
60
90
120
150

VISUALIZING CHANGE IN PAIRED DATA

YOUR TURN

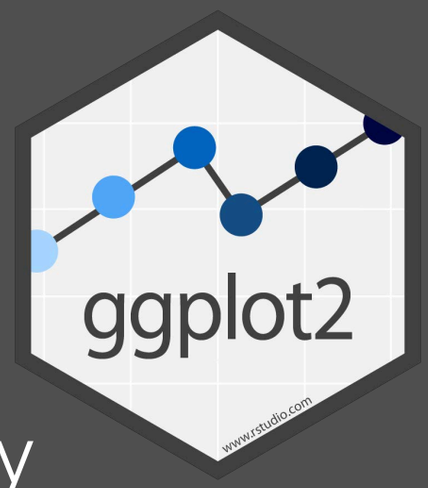


- Go to this week's assignments on the course website and download the file "school_earnings.csv".

- Install the packages `rcartocolor` and `glue`.

- Create a new R markdown file for this interactive activity (nothing to turn in).

GOAL

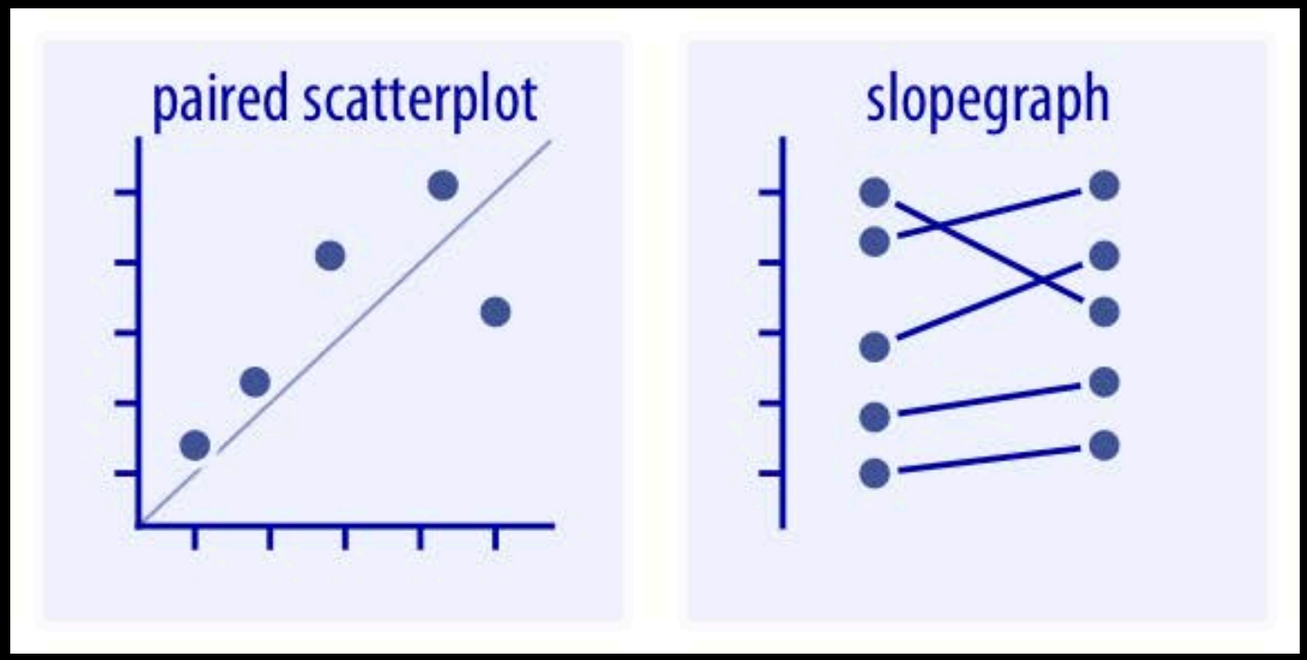


- The data show median mid-career annual salary (in thousands) for men and women who graduated from various elite universities.

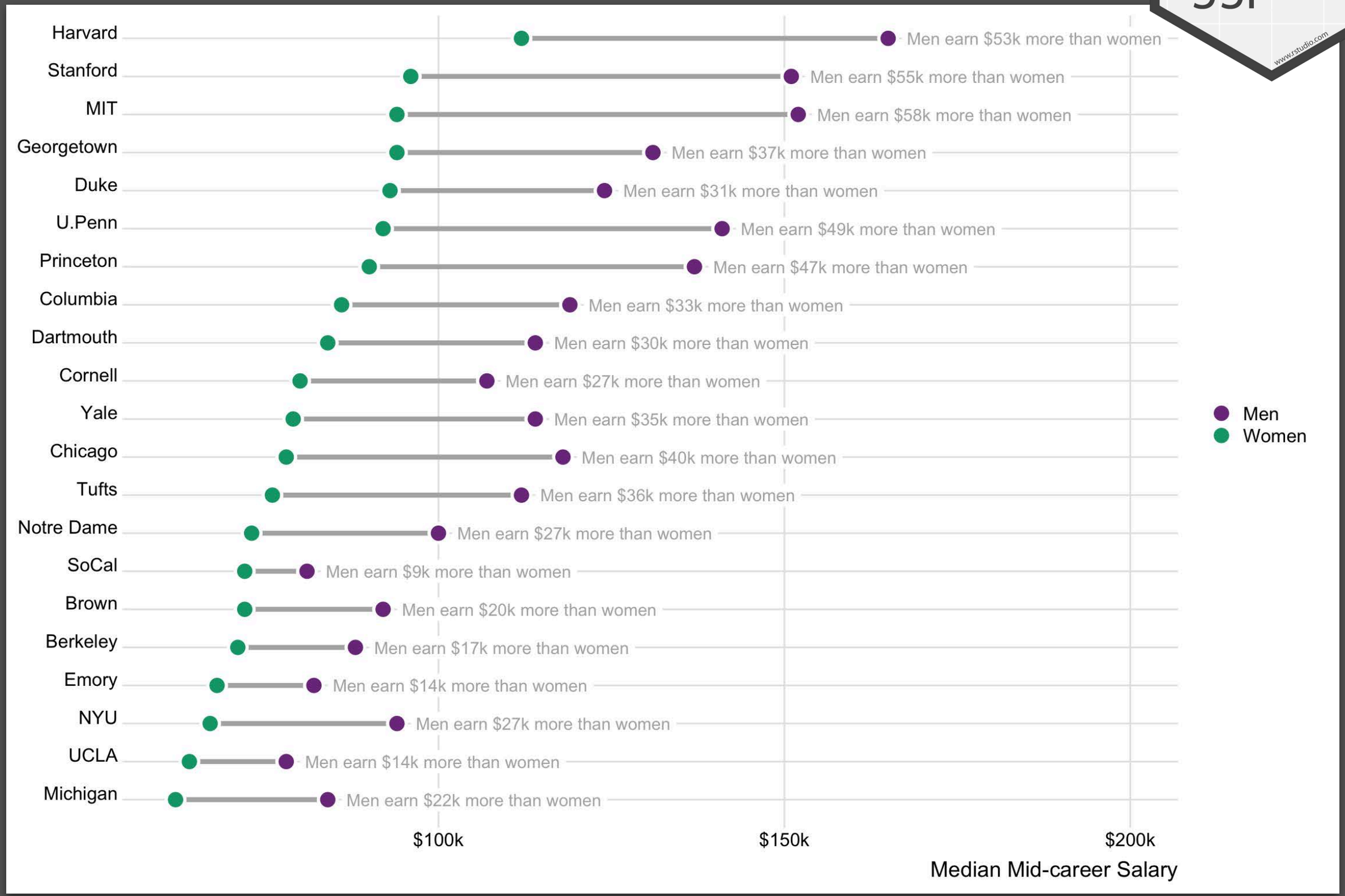
- Task: visualize the gender pay gap for each school.

School	Gap	Gender	Pay
Berkeley	17	Men	88
Berkeley	17	Women	71
Brown	20	Men	92
Brown	20	Women	72
Chicago	40	Men	118
Chicago	40	Women	78
Columbia	33	Men	119
Columbia	33	Women	86
Cornell	27	Men	107
Cornell	27	Women	80

VISUALIZING CHANGE IN PAIRED DATA



OUR TARGET


Harvard
Men earn $53k more than women
Stanford
Men earn $55k more than women
MIT
Men earn $58k more than women
Georgetown
Men earn $37k more than women
Duke
Men earn $31k more than women
U.Penn
Men earn $49k more than women
Princeton
Men earn $47k more than women
Columbia
Men earn $33k more than women
Dartmouth
Men earn $30k more than women
Cornell
Men earn $27k more than women
Yale
Men earn $35k more than women
Chicago
Men earn $40k more than women
Tufts
Men earn $36k more than women
Notre Dame
Men earn $27k more than women
SoCal
Men earn $9k more than women
Brown
Men earn $20k more than women
Berkeley
Men earn $17k more than women
Emory
Men earn $14k more than women
NYU
Men earn $27k more than women
UCLA
Men earn $14k more than women
Michigan
Men earn $22k more than women
$100k
$150k
$200k
Median Mid-career Salary
Men
Women

WE DID NOT HAVE TIME FOR THE PAIRED PLOT ACTIVITY

See the “pay-gaps.html” file on the course website if you’re interested in how to make the dumbbell plot.

ASIDE: THE GLUE PACKAGE


```
library("glue")

my_name <- "Fernando"
glue("My name is {my_name}.")

my_nums <- 1:5
glue("This is row {my_nums}.")

glue("The price is now ${1:5}.00 higher.")

glue("The average of 10, 14, and 33 is: {mean(c(10, 14, 33))}")


name_sequence <- c("first name", "middle name", "last name")
my_names <- c("Fernando", "Alonso", "Campos")
glue("My {name_sequence} is {my_names}.")
```