# Data visualisation with R

Amy Li
amy.x.li@unsw.edu.au

*adapted in part from workshops by*
Danielle Navarro and Andrew Perfors

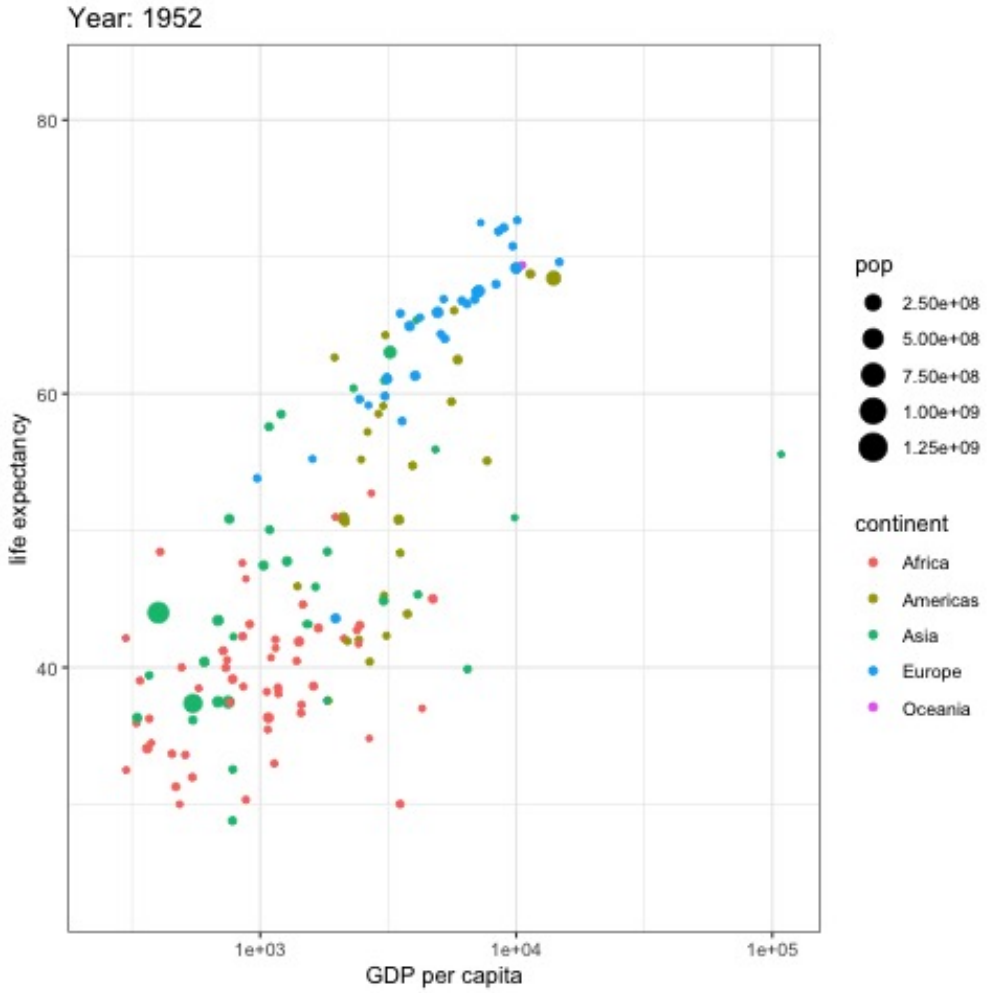
# introduction to R

# why R?

- open source
- tool of choice for researchers + data scientists
- analysis made extremely easy + reproducible
- it can do *a lot*

# why R?

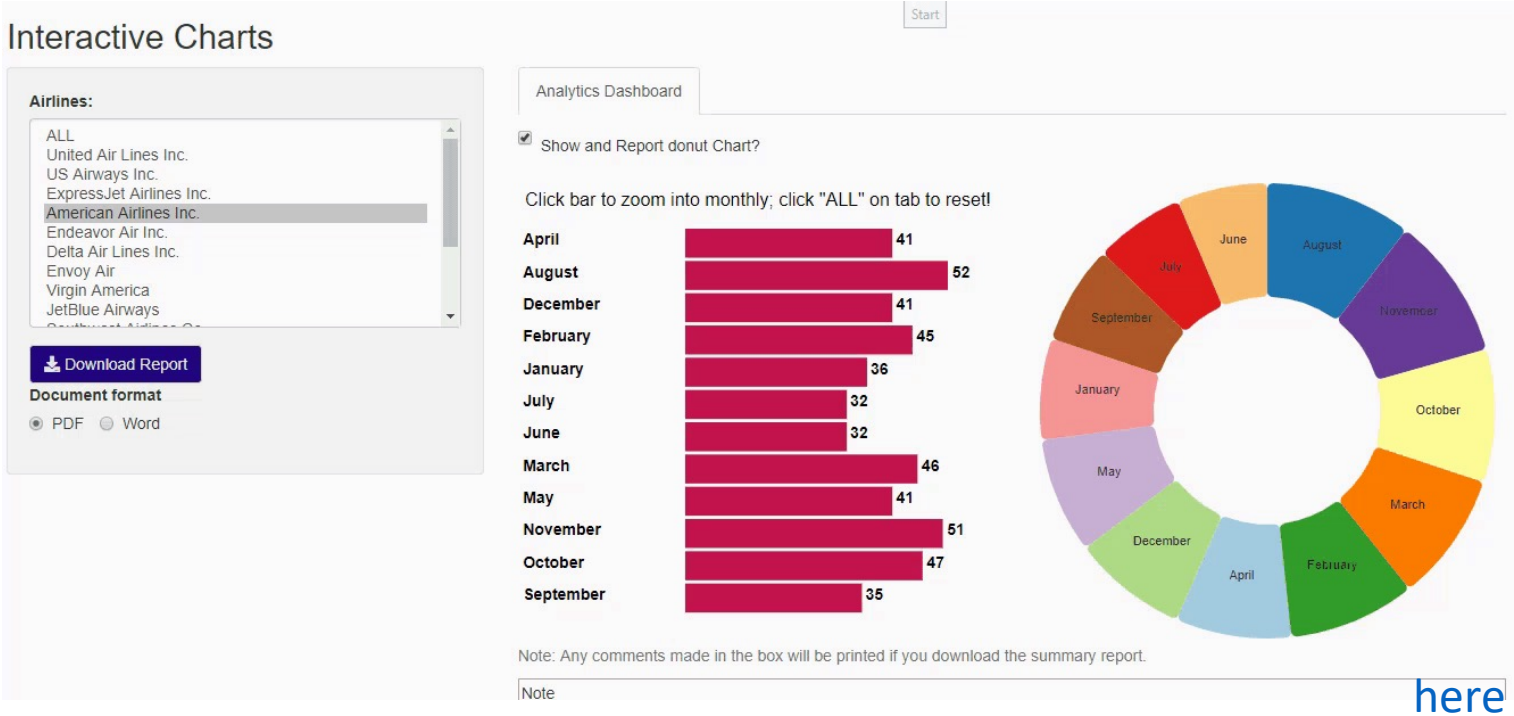
- open source
- tool of choice for researchers + data scientists
- analysis made extremely easy + reproducible
- it can do *a lot*



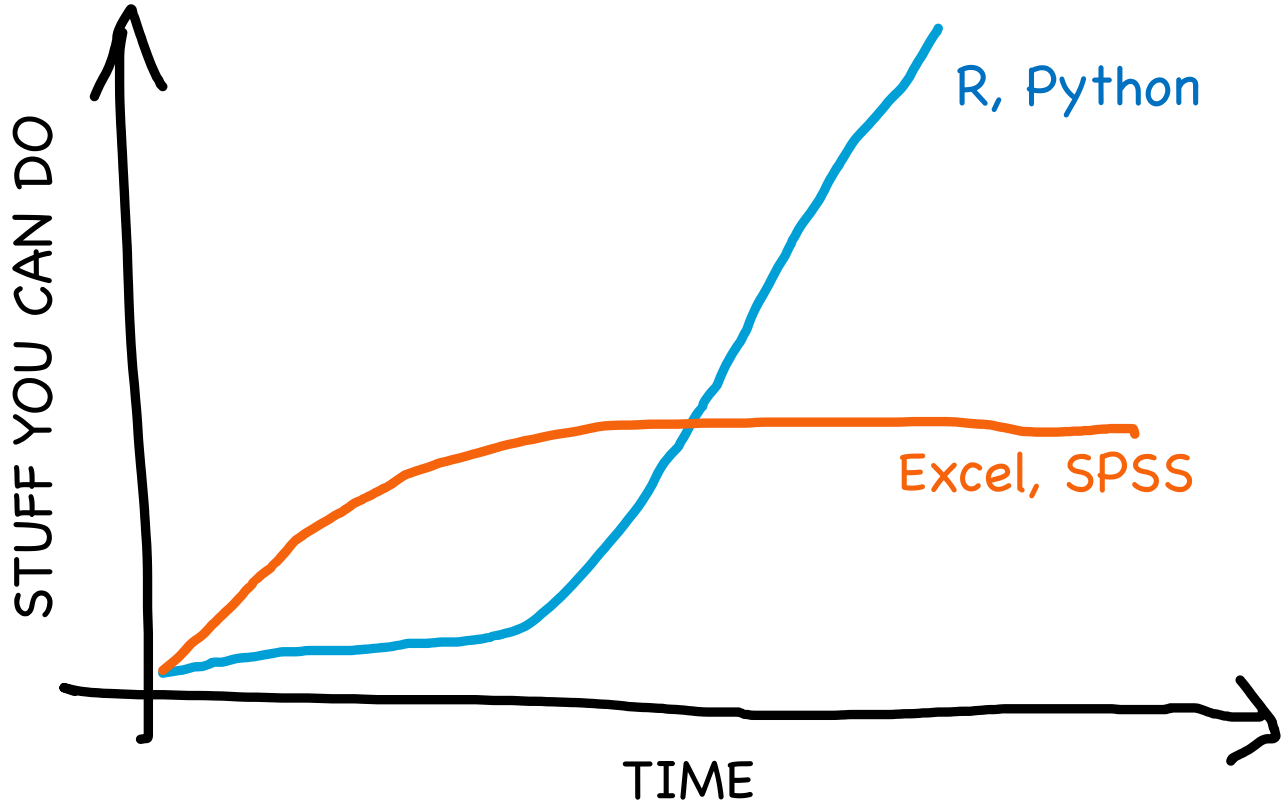

using `gganimate()`; code here

# why R?

- open source
- tool of choice for researchers + data scientists
- analysis made extremely easy + reproducible
- it can do *a lot*

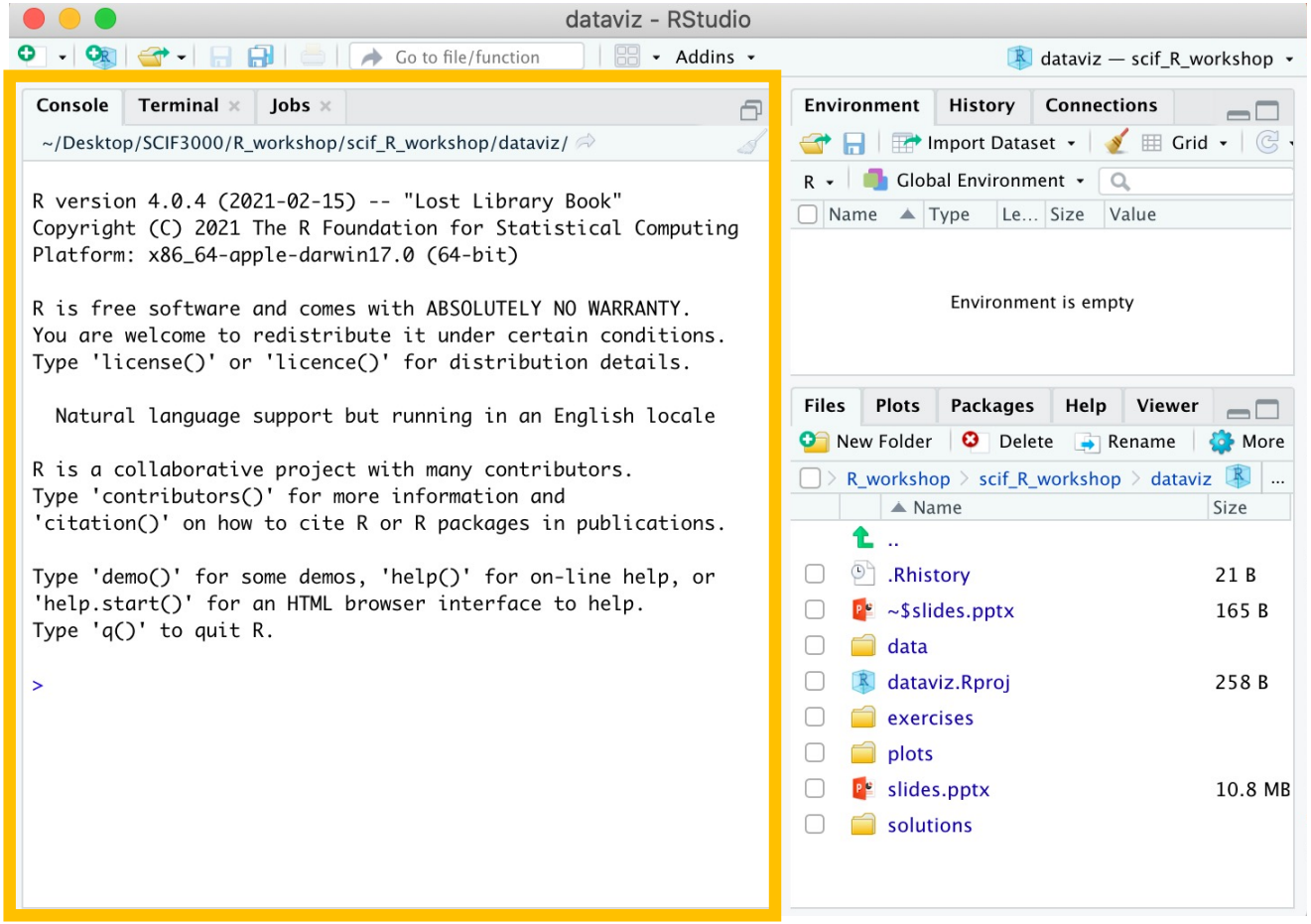



here

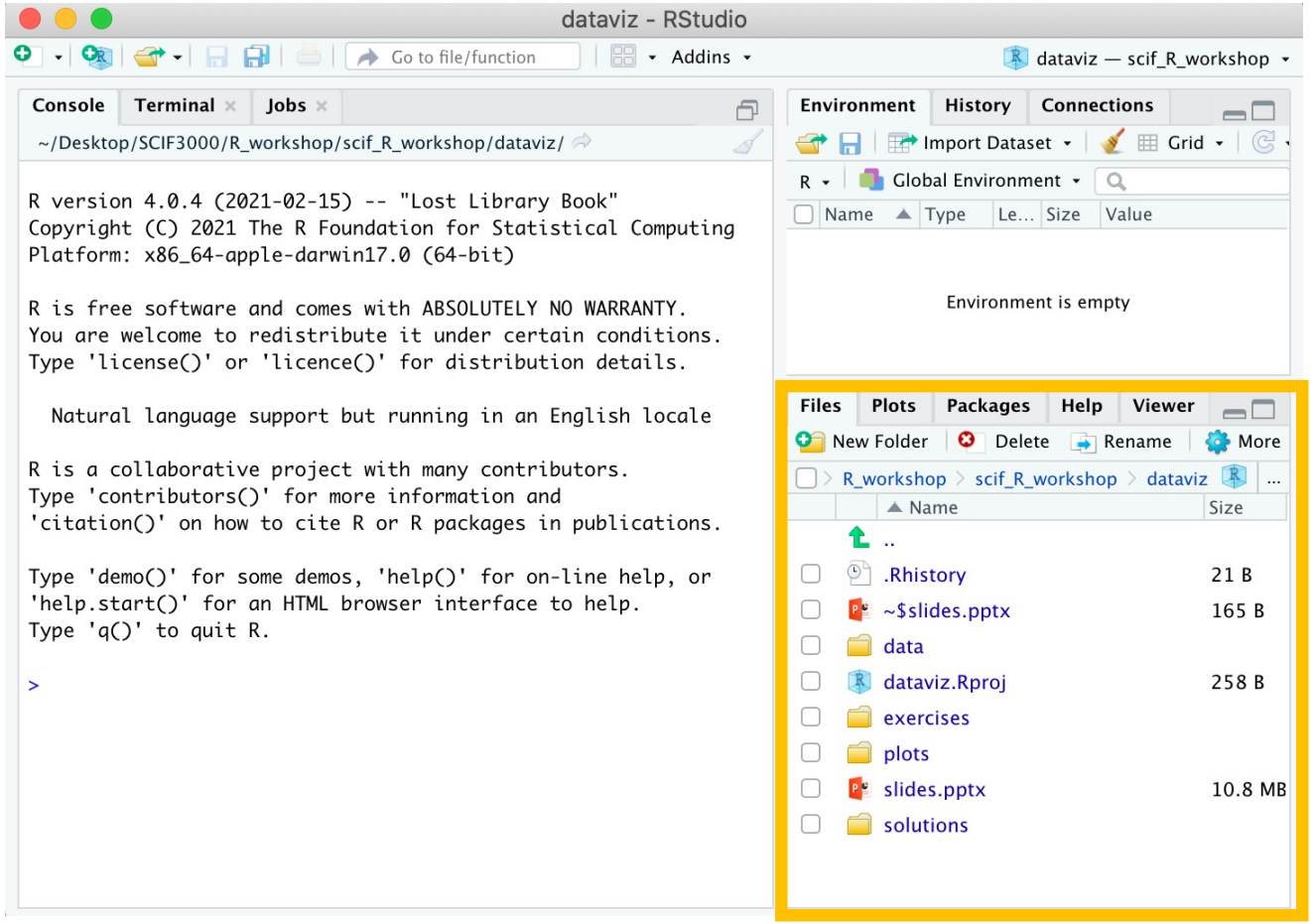

STUFF YOU CAN DO
TIME
R, Python
Excel, SPSS

# the RStudio interface

console:
where R evaluates
your code

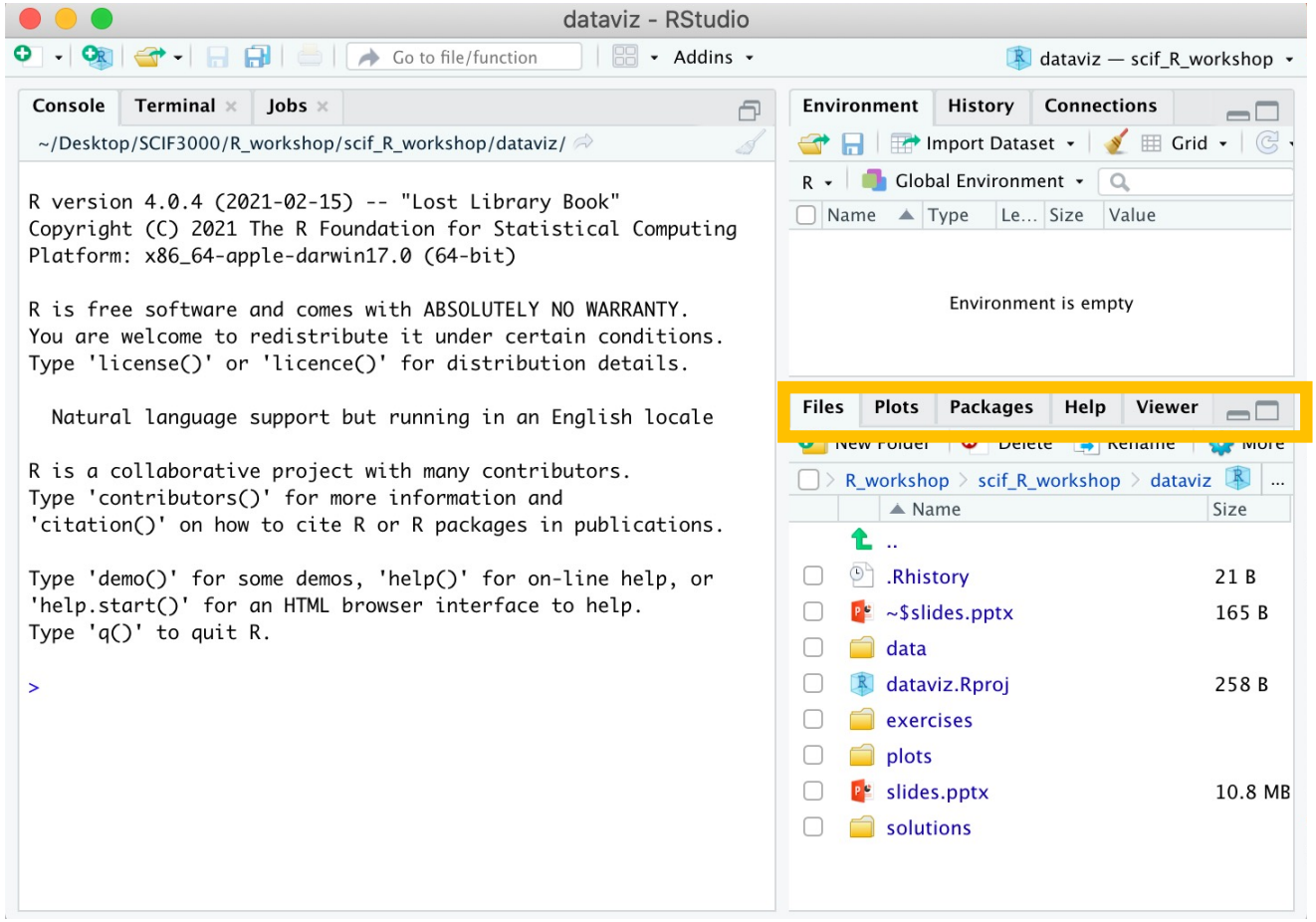


# the RStudio interface




file directory

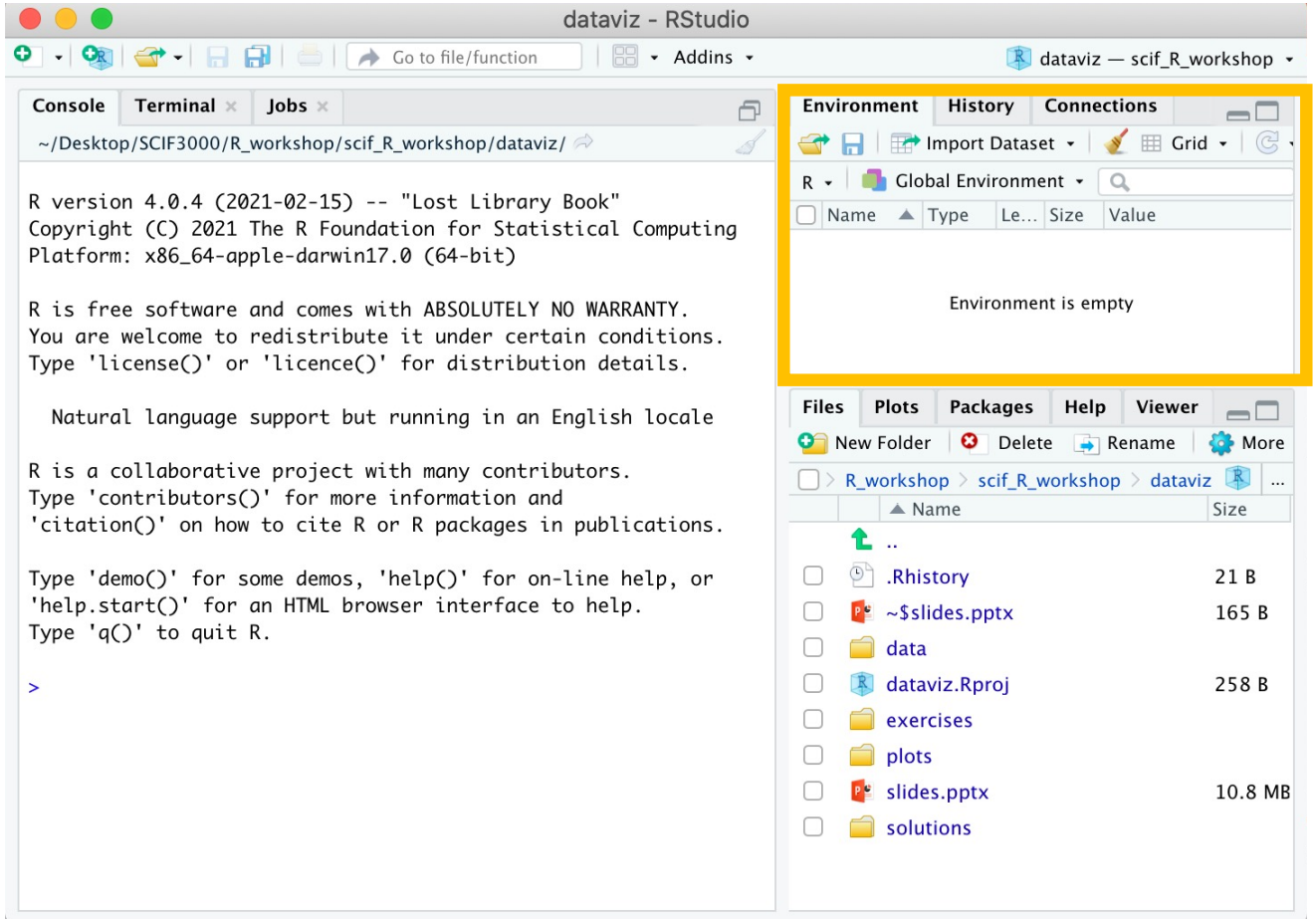
# the RStudio interface




plots, packages, help

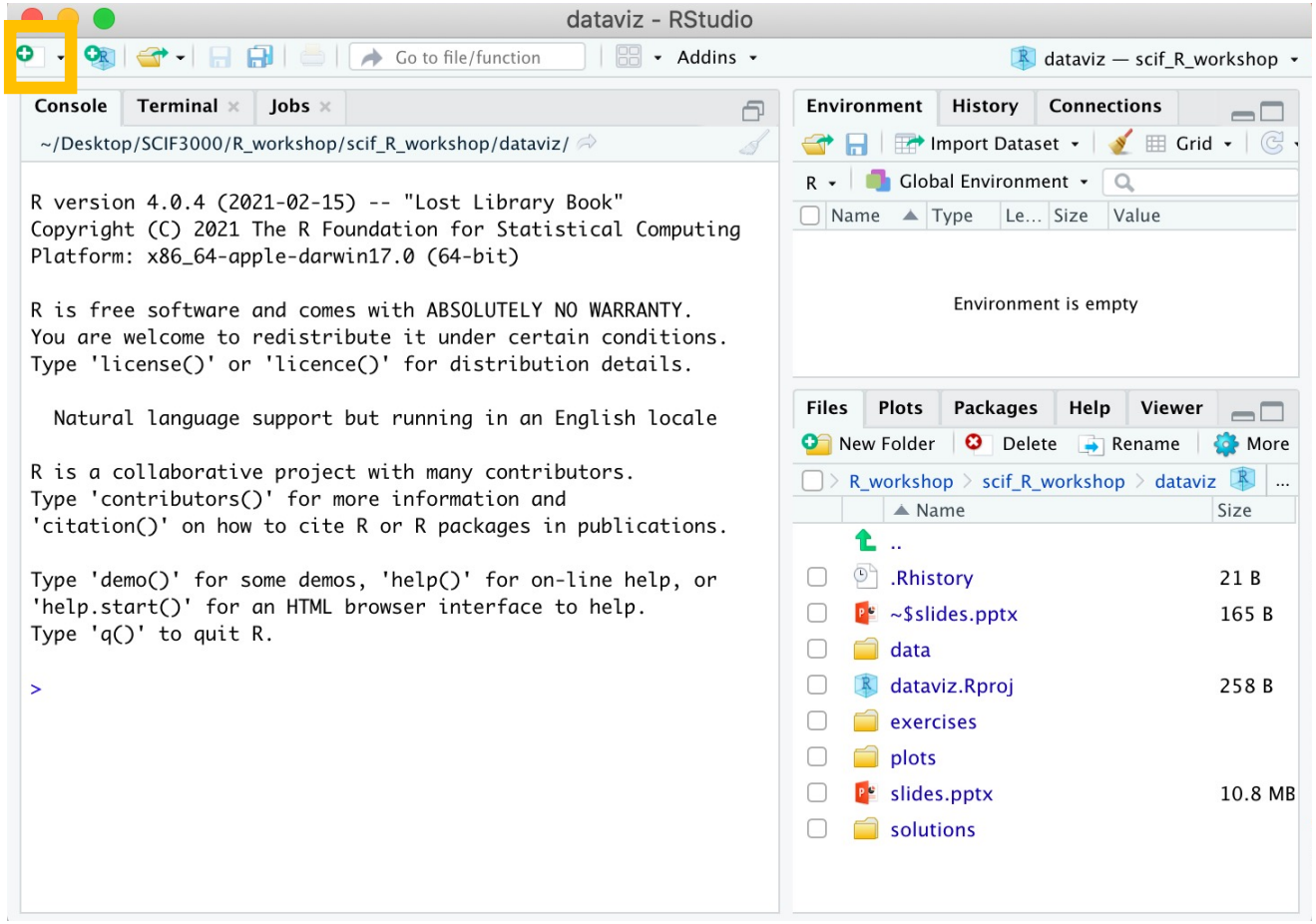
# the RStudio interface

environment:
stuff R knows right now



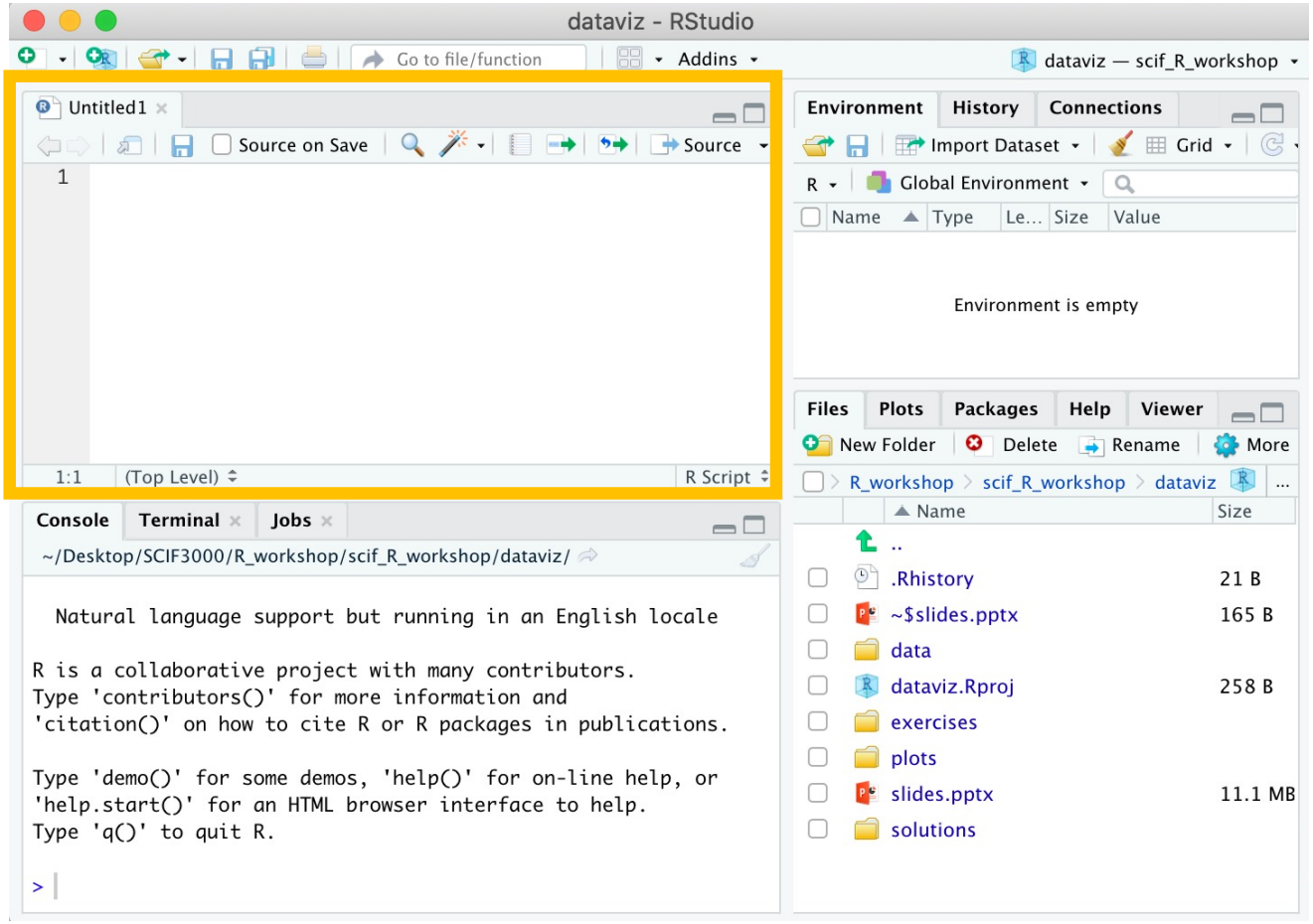
# the RStudio interface

new script

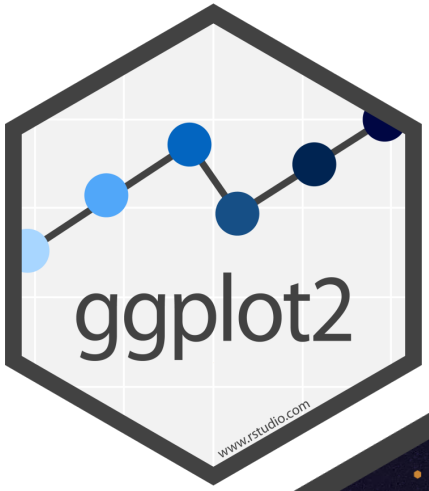


# the RStudio interface

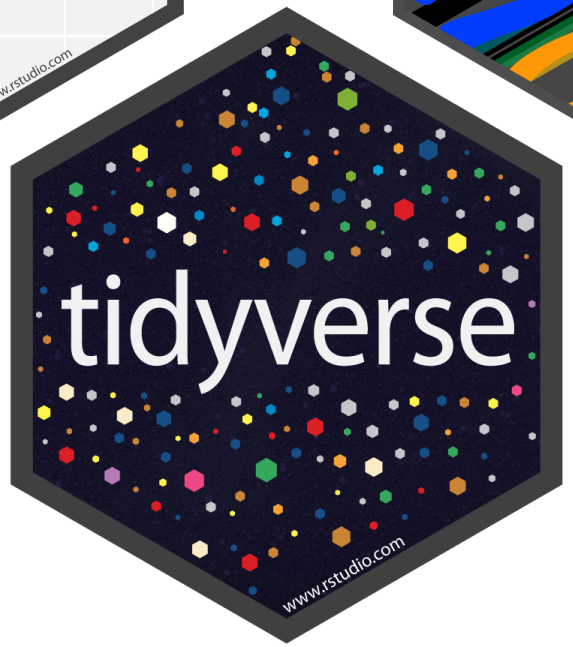
built-in
script editor



your main helpers








# installing and using packages

## installed means...

the package files are stored on your computer
your version of R is **able to load** the package

```
install.packages("PACKAGENAME")
```

## loaded means...

the package is now "switched on"
you can **use** the functions / data stored in the package

```
library(PACKAGENAME)
```

## As a result:

a package must be installed before you can load it
a package must be loaded before you can **use it**

# the working directory problem

R needs to know *exactly* where your files are.

```
setwd("C:\Users\amyxli\path\that\only\I\have")
```

# the working directory problem

R needs to know *exactly* where your files are.

```
setwd("C:\Users\amyxli\path\that\only\I\have")
```

Problem 1: This changes from computer to computer.

```
setwd("C:\Users\collaborator\path\in\folder\that\only\I\have")
```

Problem 2: This is typo prone.

```
setwd("C:\Users\callaborator\path\in\fodler\that\only\I\have")
```

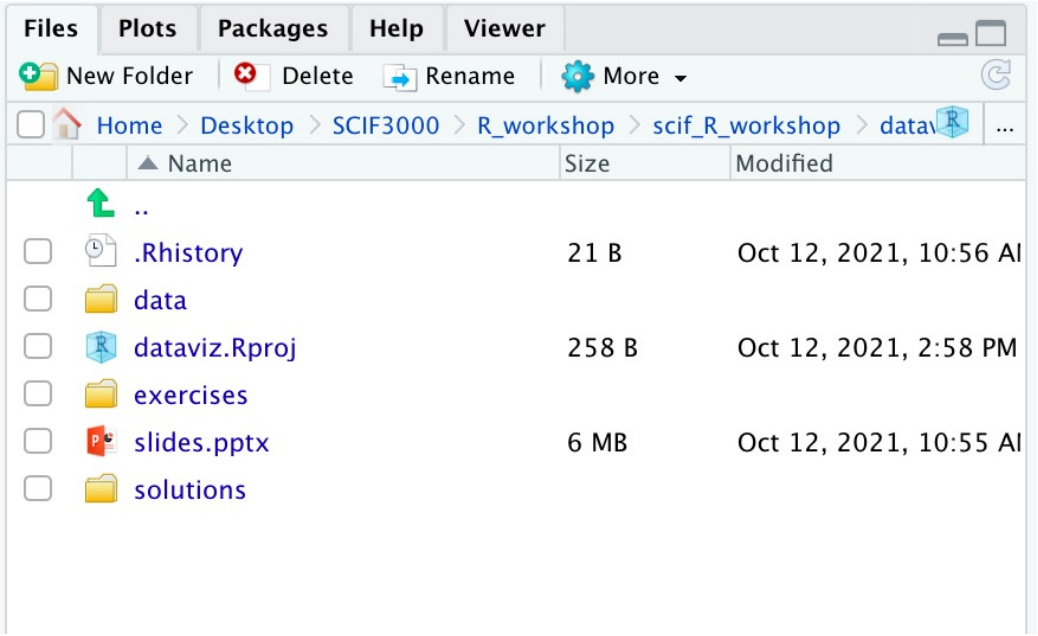
# the working directory problem

1. Have an organized, project-based folder structure.

2. Make an R project at the ”base” of your project folder.

3. Use `here()` to specify file paths.

4. Rest easy knowing that others can also run your code without fiddling with file paths.

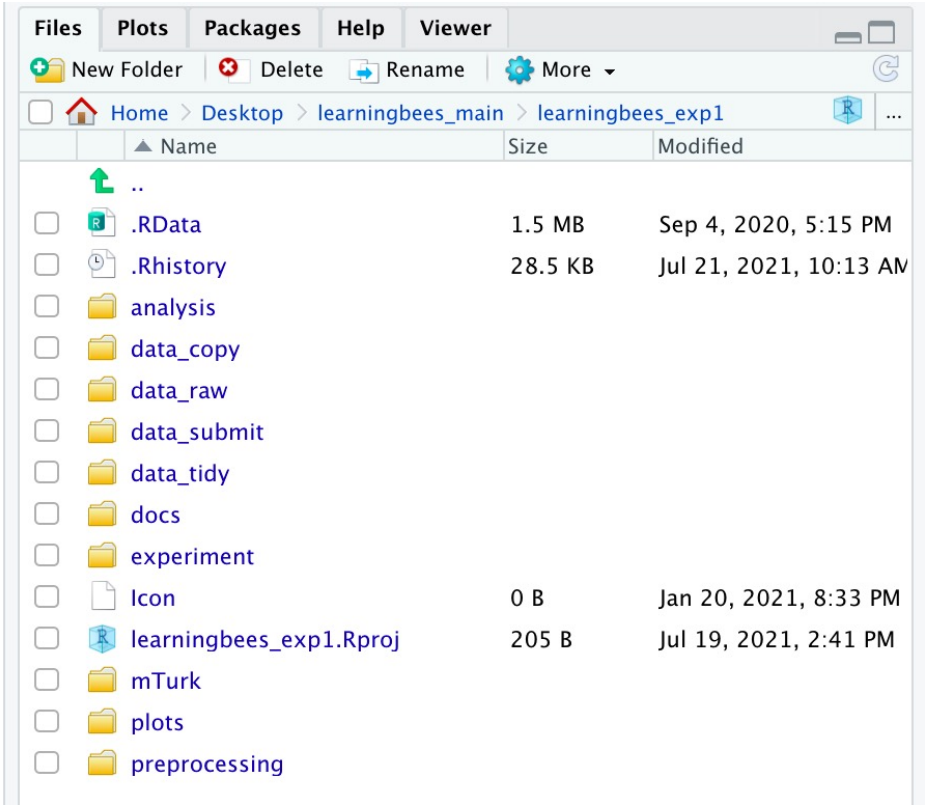



```
> library(here)
here() starts at /Users/amyli/Desktop/SCIF3000/R_workshop/scif_R_workshop/dataviz

> here("data","sydneybeaches.csv")
[1] "/Users/amyli/Desktop/SCIF3000/R_workshop/scif_R_workshop/dataviz/data/sydneybeaches.csv"
```

# the working directory problem

Another example of folder structure…



# visualising data with *ggplot2*

# beaches data

```
# load data
beaches <- read_csv(here("data", "sydneybeaches.csv"))

# show data
beaches
```

```
## # A tibble: 344 x 12
##    date        year month   day season rainfall temperature enterococci
##    <date>     <dbl> <dbl> <dbl>  <dbl>    <dbl>       <dbl>       <dbl>
##  1 2013-01-02  2013     1     2      1      0          23.4         6.7
##  2 2013-01-06  2013     1     6      1      0          30.3         2
##  3 2013-01-12  2013     1    12      1      0          31.4        69.1
##  4 2013-01-18  2013     1    18      1      0          46.4         9
##  5 2013-01-24  2013     1    24      1      0          27.5        33.9
##  6 2013-01-30  2013     1    30      1      0.6        26.6        26.5
##  7 2013-02-05  2013     2     5      1      0.1        25.7        66.9
##  8 2013-02-11  2013     2    11      1      8          22.2       118.
##  9 2013-02-17  2013     2    17      1     13.6        26.3        75
## 10 2013-02-23  2013     2    23      1      7.2        24.8       311.
## # ... with 334 more rows, and 4 more variables: day_num <dbl>,
## #   month_num <dbl>, month_name <chr>, season_name <chr>
```

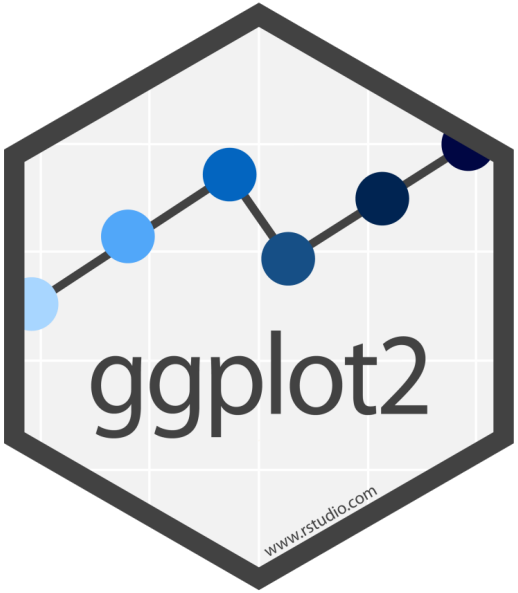
# exercise 1:
# introduction to ggplot2

# ggplot2

A “grammar of graphics”.

- composed of, and reuses, small parts
- build complex structures from simpler units

- uses the “painter’s model”:
  - plot is built in layers



# Start with a blank canvas

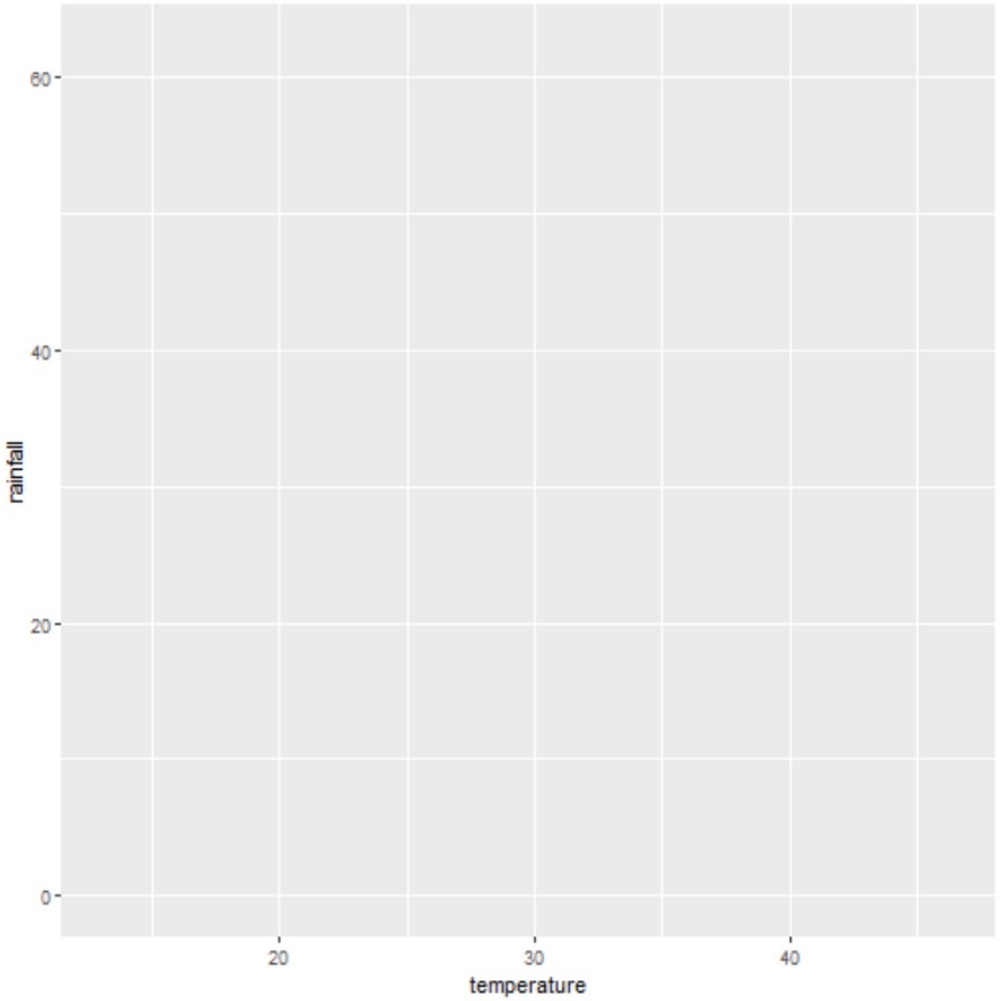
```
ggplot()
```

# Specify the data

```
ggplot(
  data = beaches
)
```

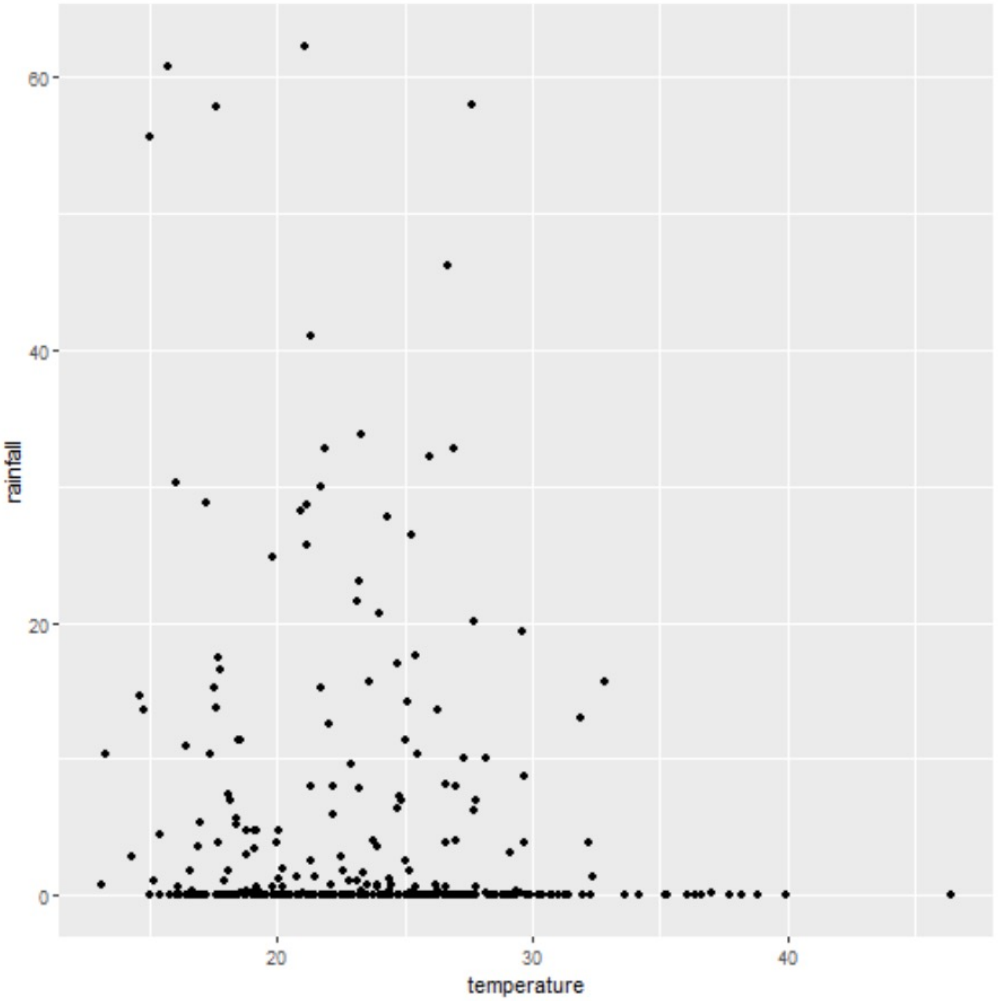
# Map data variables onto plot aesthetics

```
ggplot(
  data = beaches,
  mapping = aes(
    x = temperature,
    y = rainfall
  )
)
```



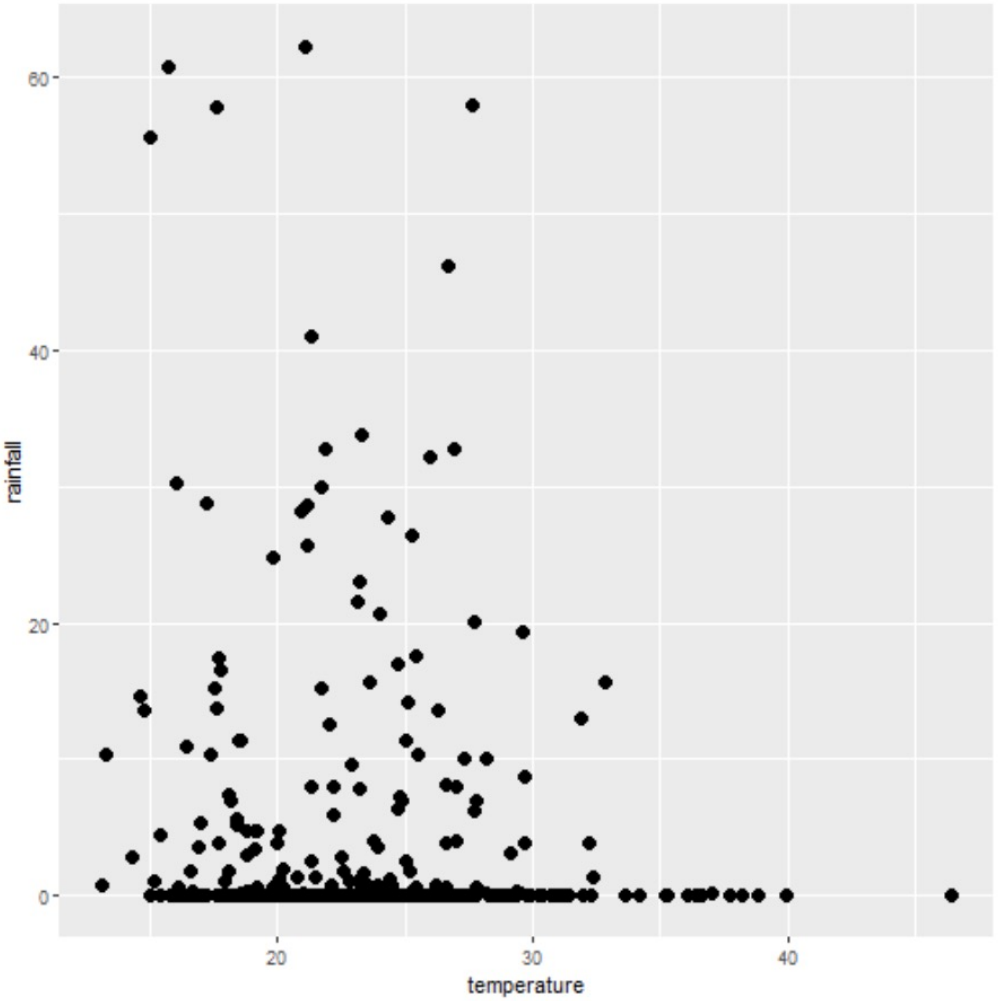
# Add a plot layer

```
ggplot(
  data = beaches,
  mapping = aes(
    x = temperature,
    y = rainfall
  )
) +
geom_point()
```



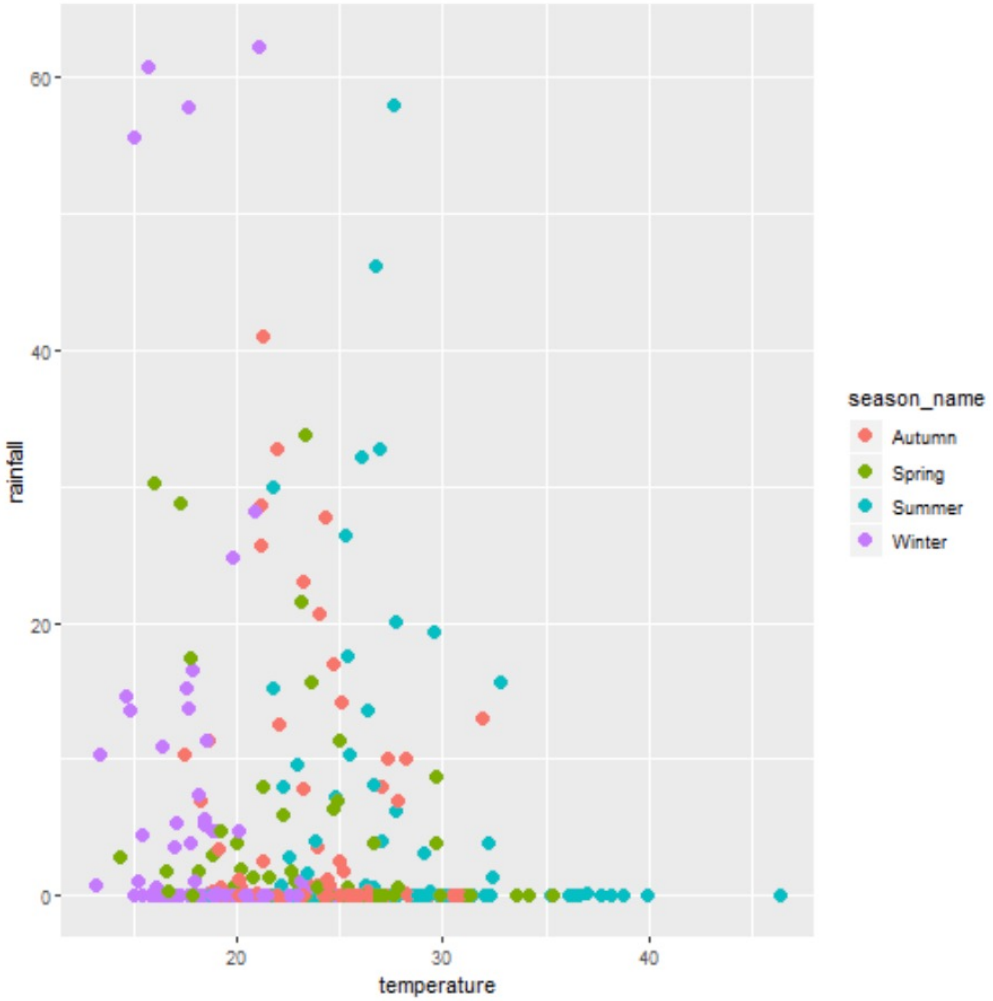
# Add layer-specific parameters

```
ggplot(
  data = beaches,
  mapping = aes(
    x = temperature,
    y = rainfall
  )
) +
geom_point(size = 3)
```



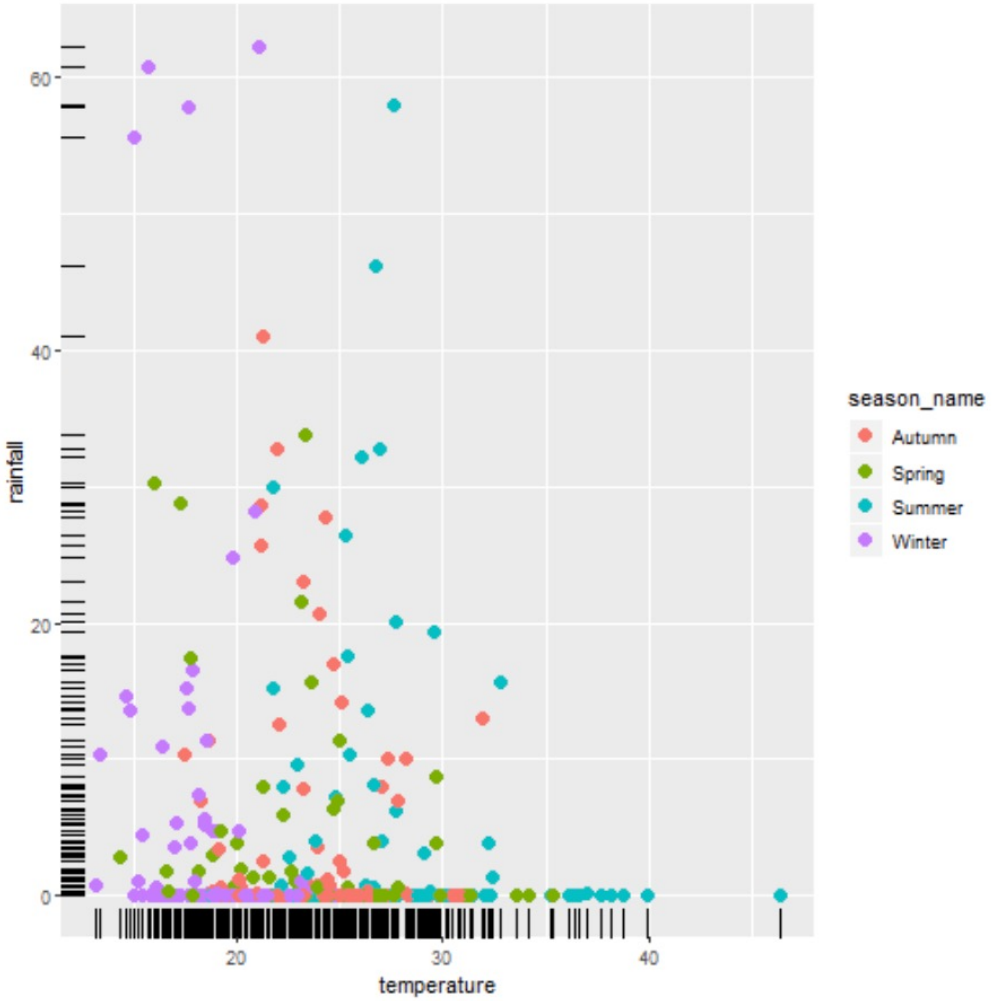
# Add layer-specific mappings

```
ggplot(
  data = beaches,
  mapping = aes(
    x = temperature,
    y = rainfall
  )
) +
geom_point(
  mapping = aes(colour = season_name),
  size = 3
)
```



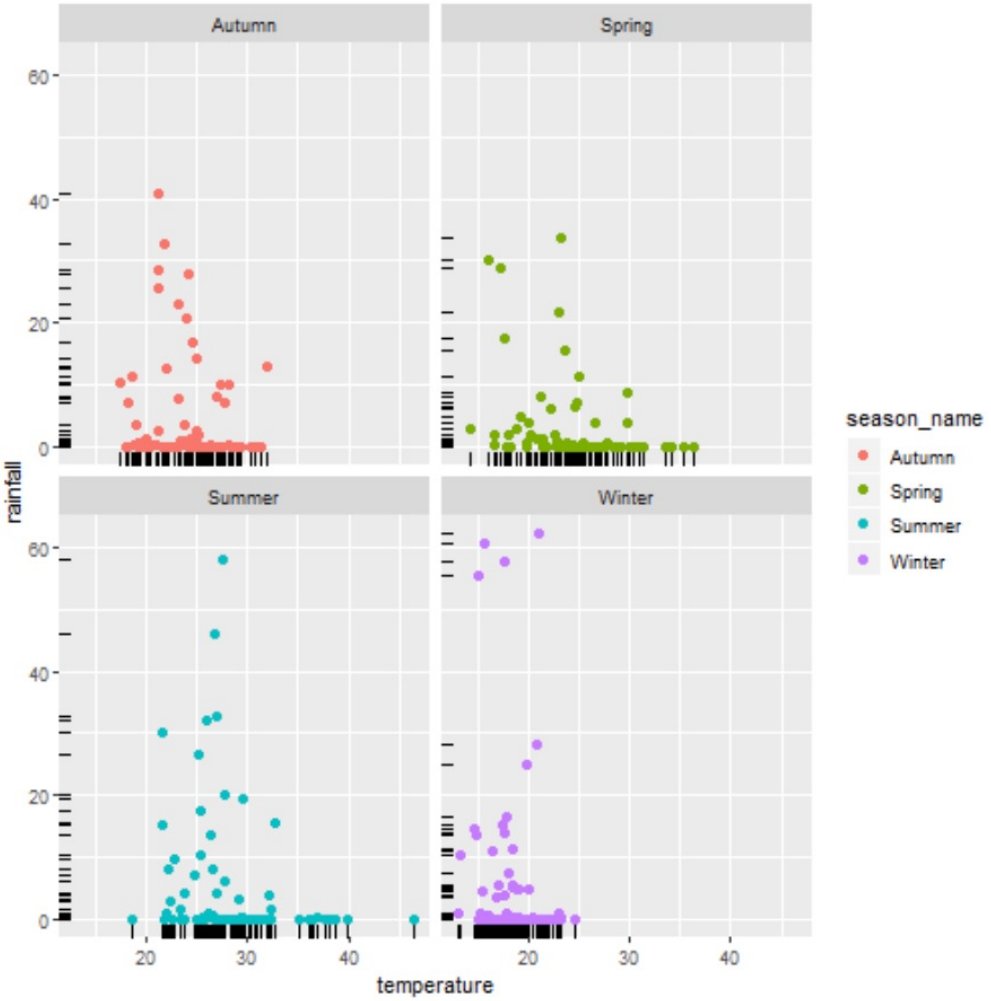
# Add more layers to the plot cake

```
ggplot(
  data = beaches,
  mapping = aes(
    x = temperature,
    y = rainfall
  )
) +
geom_point(
  mapping = aes(colour = season_name),
  size = 3
) +
geom_rug()
```



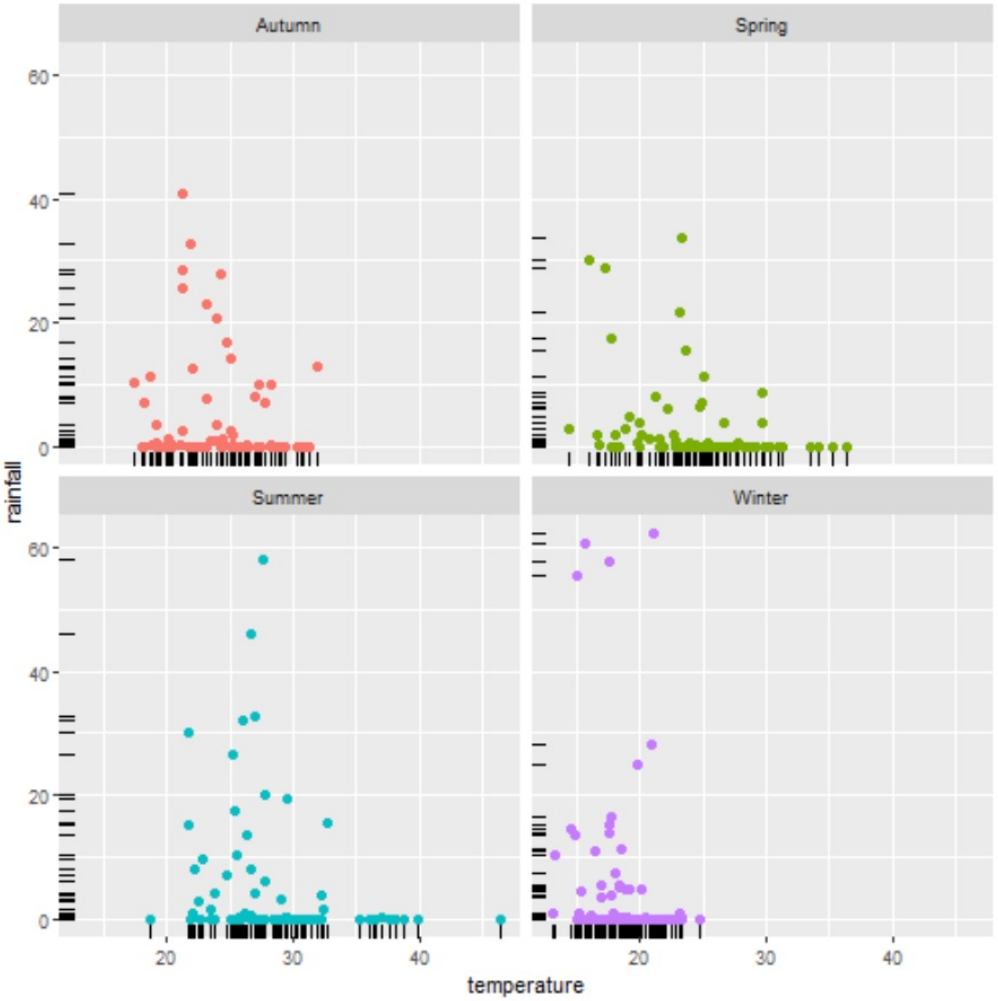
# Partition into facets

```
ggplot(
  data = beaches,
  mapping = aes(
    x = temperature,
    y = rainfall
  )
) +
geom_point(
  mapping = aes(colour = season_name),
  size = 2
) +
geom_rug() +
facet_wrap(vars(season_name))
```



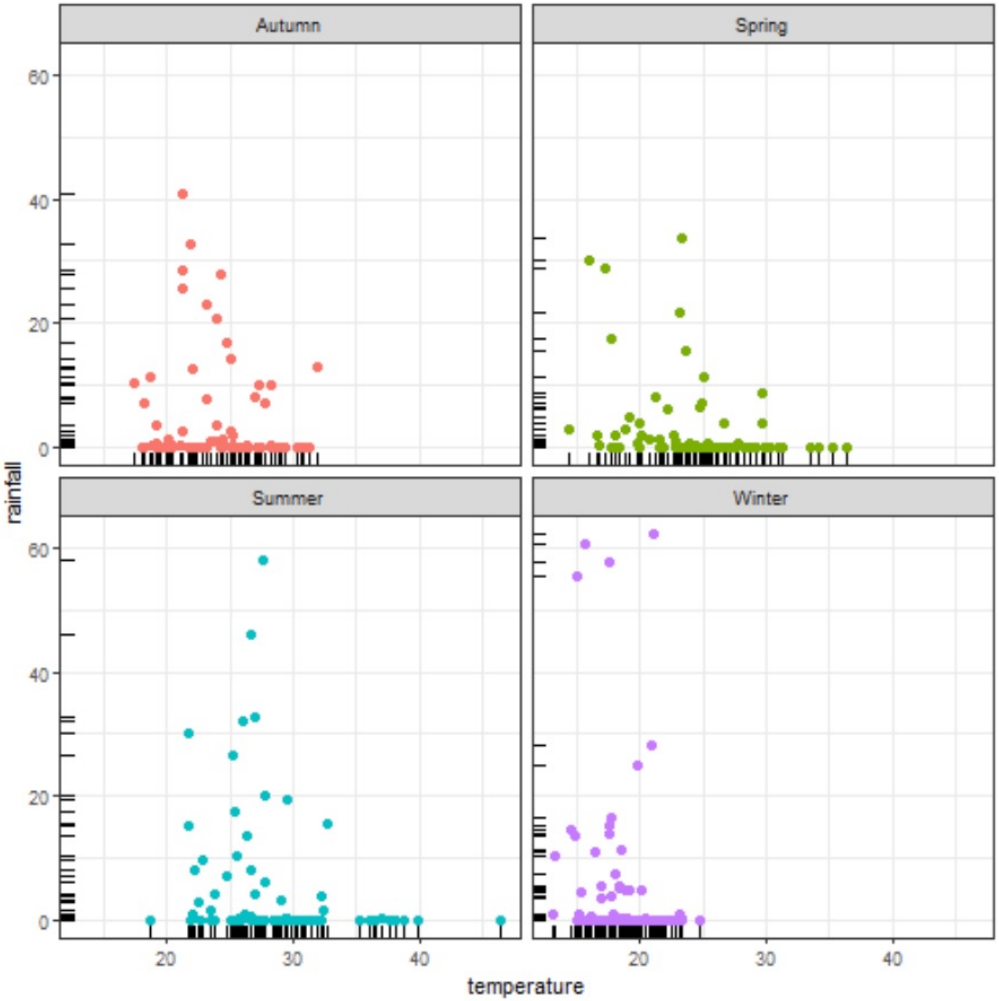
# Remove a redundant legend

```
ggplot(
  data = beaches,
  mapping = aes(
    x = temperature,
    y = rainfall
  )
) +
geom_point(
  mapping = aes(colour = season_name),
  size = 2,
  show.legend = FALSE
) +
geom_rug() +
facet_wrap(vars(season_name))
```



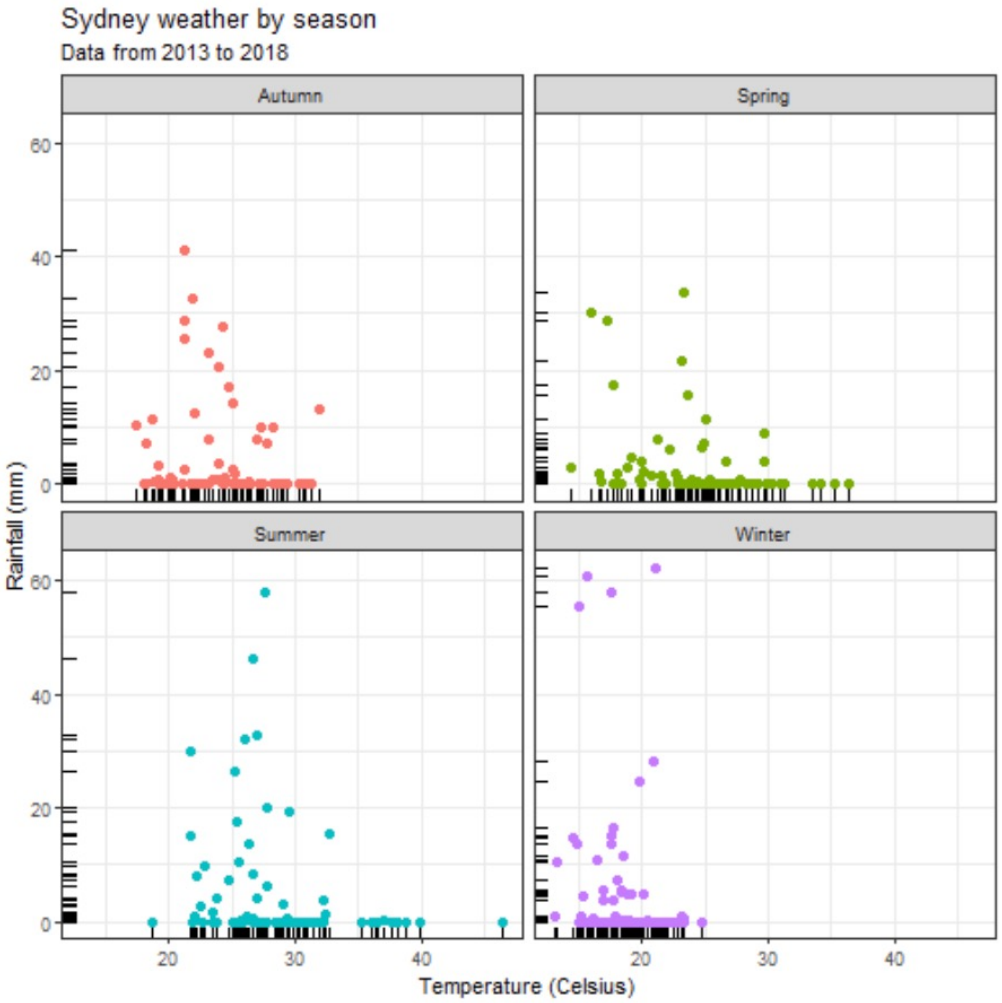
# Apply theme of choice

```
ggplot(
  data = beaches,
  mapping = aes(
    x = temperature,
    y = rainfall
  )
) +
geom_point(
  mapping = aes(colour = season_name),
  size = 2,
  show.legend = FALSE
) +
geom_rug() +
facet_wrap(vars(season_name)) +
theme_bw()
```



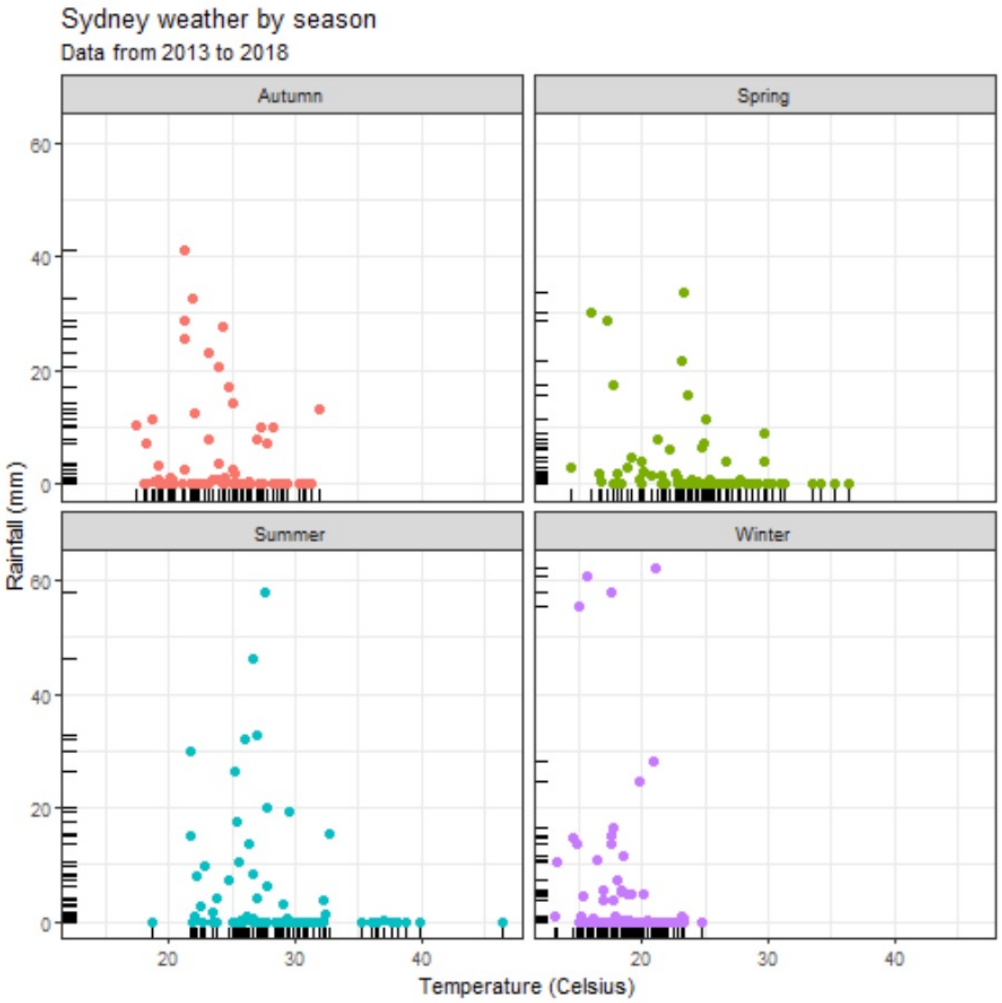
# Make some nicer titles

```
ggplot(
  data = beaches,
  mapping = aes(
    x = temperature,
    y = rainfall)) +
geom_point(
  mapping = aes(colour = season_name),
  size = 2,
  show.legend = FALSE) +
geom_rug() +
facet_wrap(vars(season_name)) +
theme_bw() +
labs(
  title = "Sydney weather by season",
  subtitle = "Data from 2013 to 2018",
  x = "Temperature (Celsius)",
  y = "Rainfall (mm)"
)
```



# Our plot painting is done!

```
ggplot(
  data = beaches,
  mapping = aes(
    x = temperature,
    y = rainfall)) +
geom_point(
  mapping = aes(colour = season_name),
  size = 2,
  show.legend = FALSE) +
geom_rug() +
facet_wrap(vars(season_name)) +
theme_bw() +
labs(
  title = "Sydney weather by season",
  subtitle = "Data from 2013 to 2018",
  x = "Temperature (Celsius)",
  y = "Rainfall (mm)"
)
```



# let’s get started!

open our first exercise script:
files panel > exercises > ex01_intro.R

**how this works:**

- 5 scripts – guided DIY exercises
  - **break** before ex. 5
- read slides that correspond to each exercise before you start

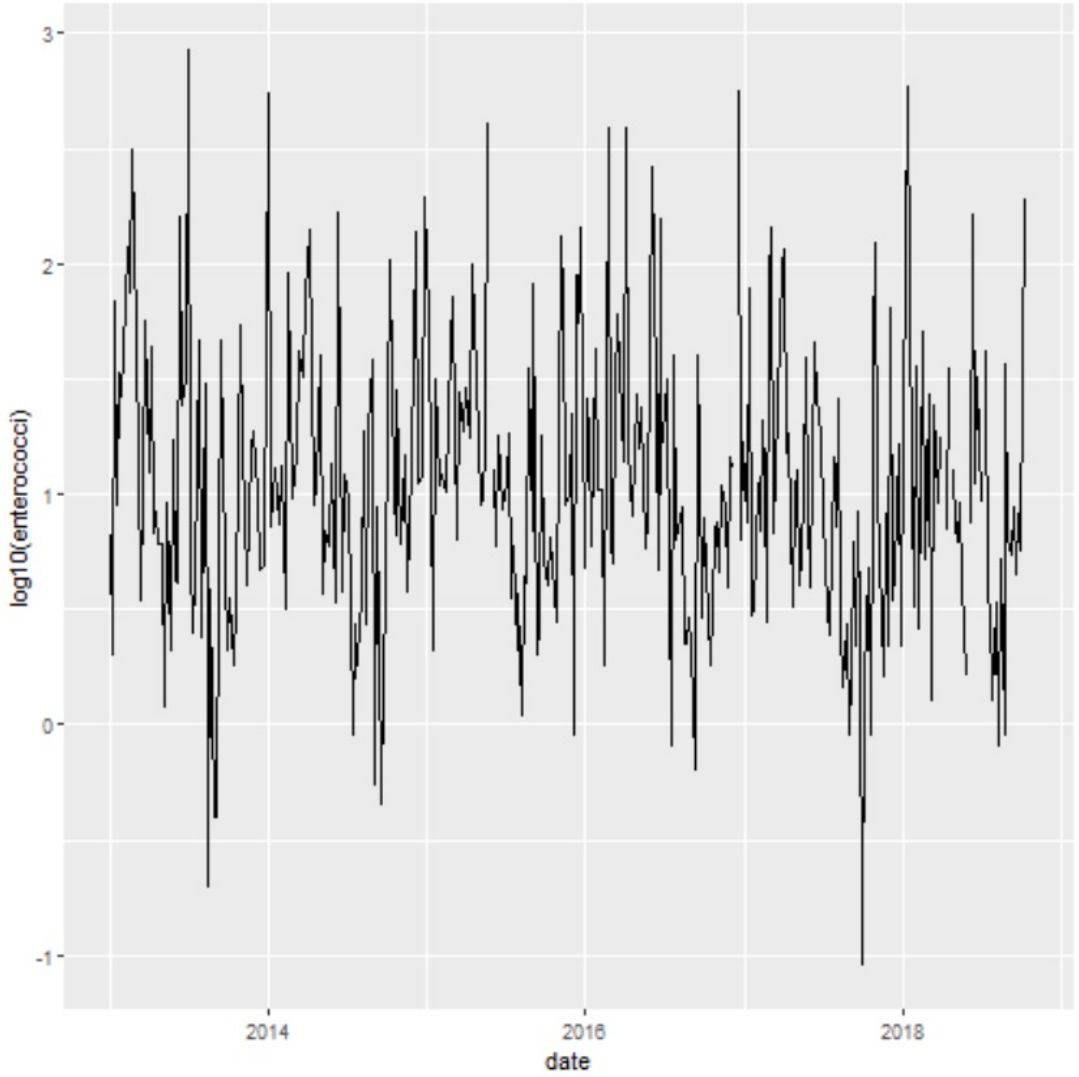
- call out + screenshare for questions
- when you’re done with each exercise: text in the chat or call out

# exercise 2:
# all about aesthetics

corresponds to:
exercises > ex02_aesthetics.R

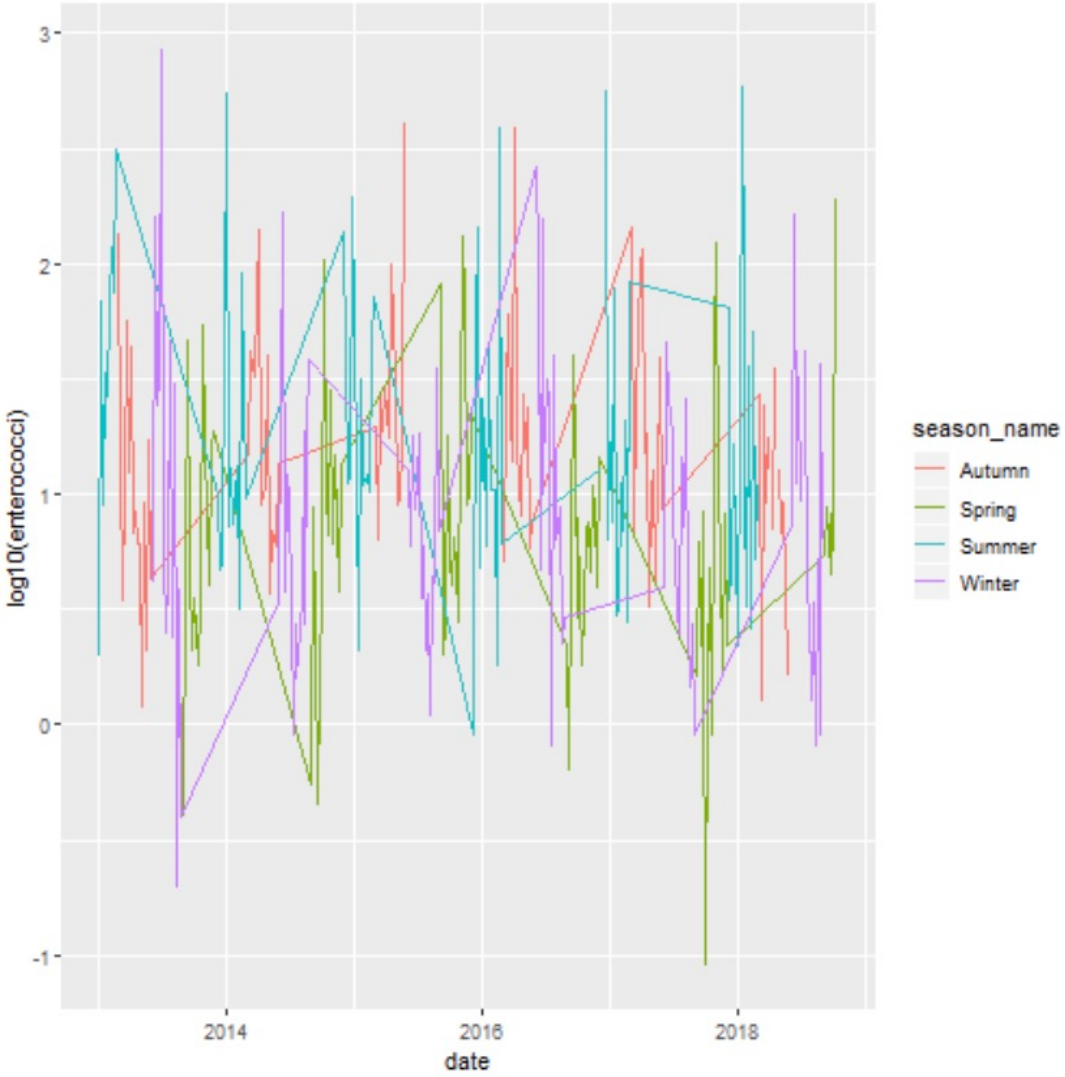
Locations (x and y) are aesthetics.

```
ggplot(
  data = beaches,
  mapping = aes(
    x = date,
    y = log10(enterococci)
  )
) +
geom_line()
```



The colour is also an aesthetic.

```
ggplot(
  data = beaches,
  mapping = aes(
    x = date,
    y = log10(enterococci),
    colour = season_name
  )
) +
geom_line()
```



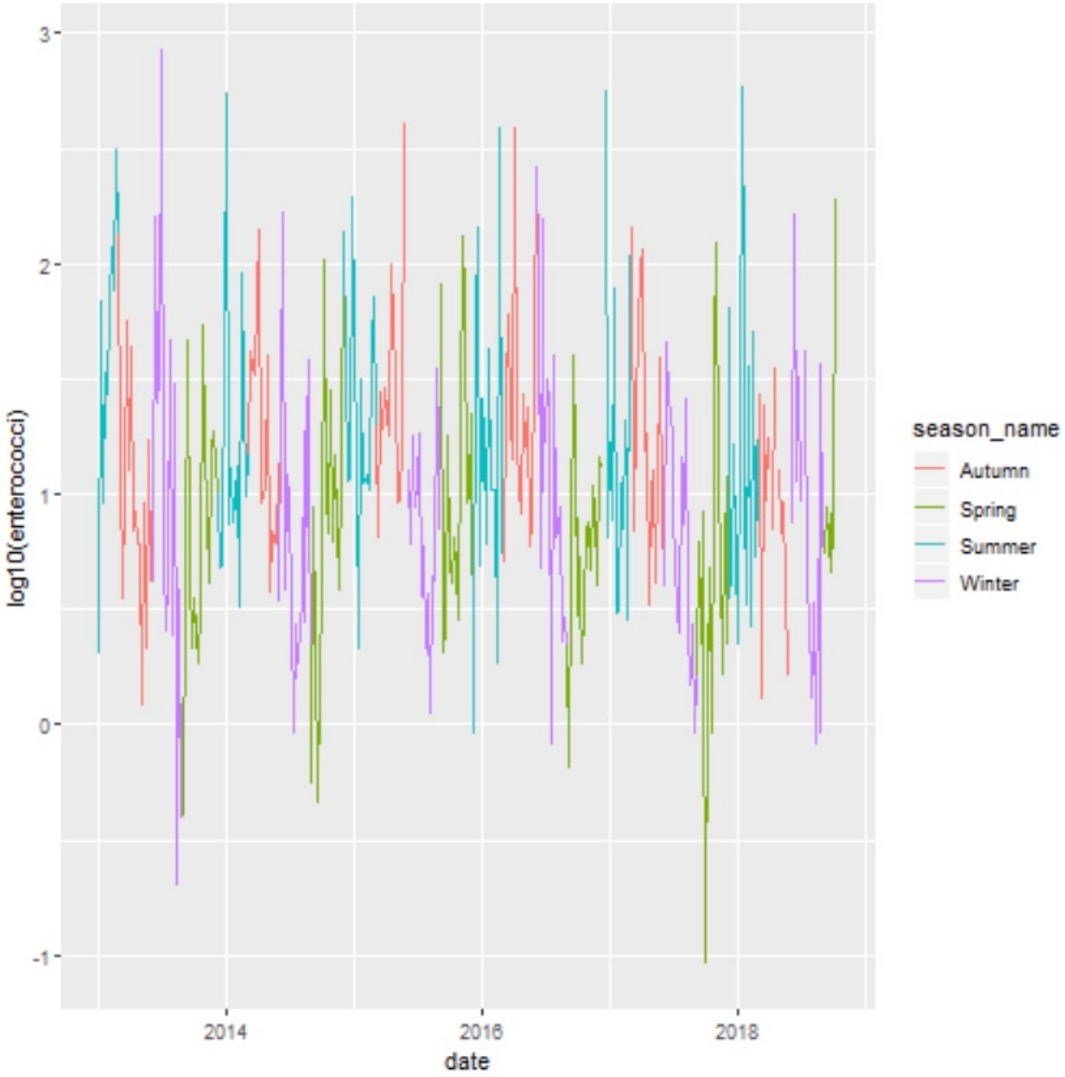

But what happened with the lines…?

Here, this is because setting colour = season_name set the default *group* aesthetic to *season_name.*

Let’s change the group aesthetic to = 1 …

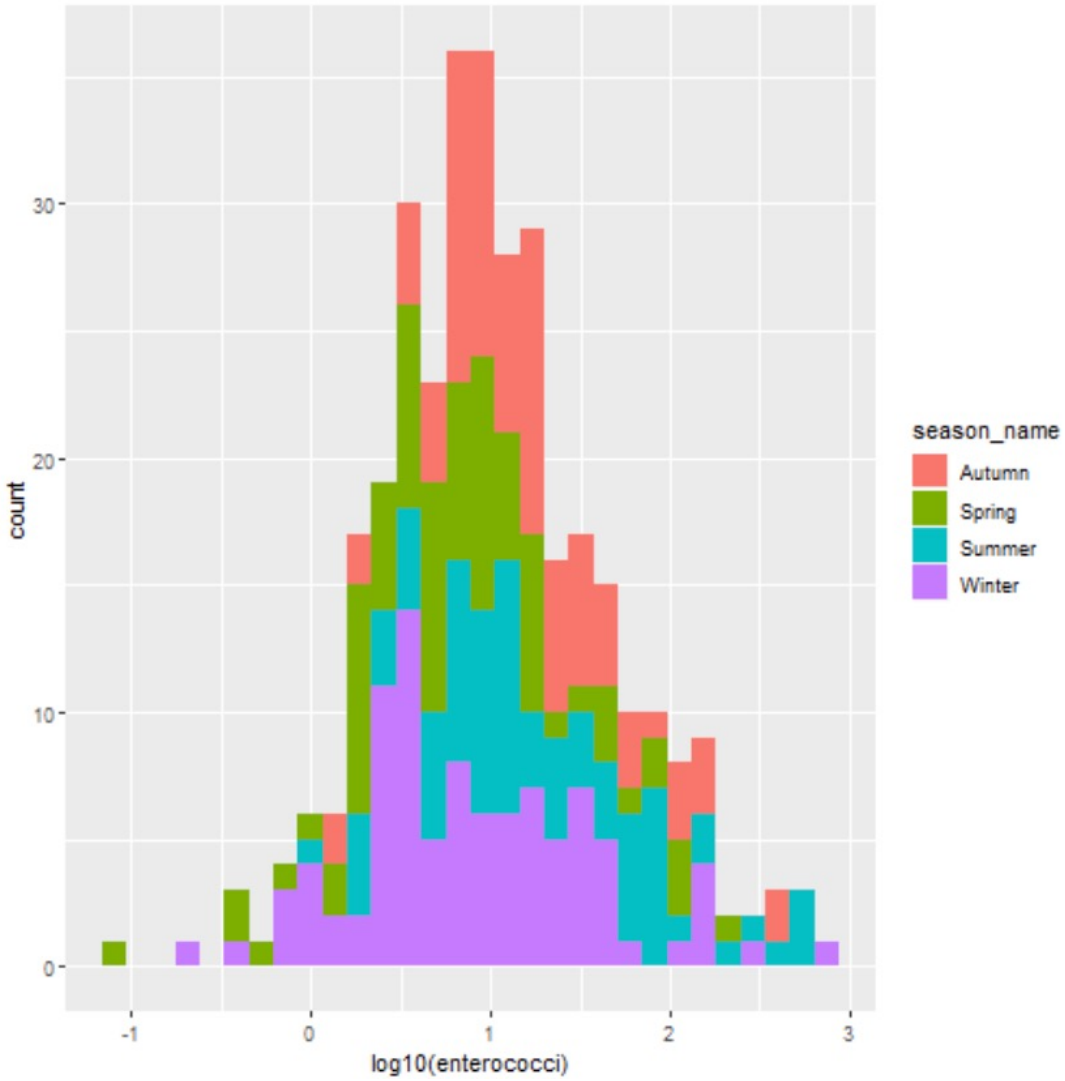
# The group aesthetic matters!

```
ggplot(
  data = beaches,
  mapping = aes(
    x = date,
    y = log10(enterococci),
    colour = season_name,
    group = 1,
  )
) +
geom_line()
```



The fill can also be an aesthetic.

```
ggplot(
  data = beaches,
  mapping = aes(
    x = log10(enterococci),
    fill = season_name
  )
) +
geom_histogram()
```

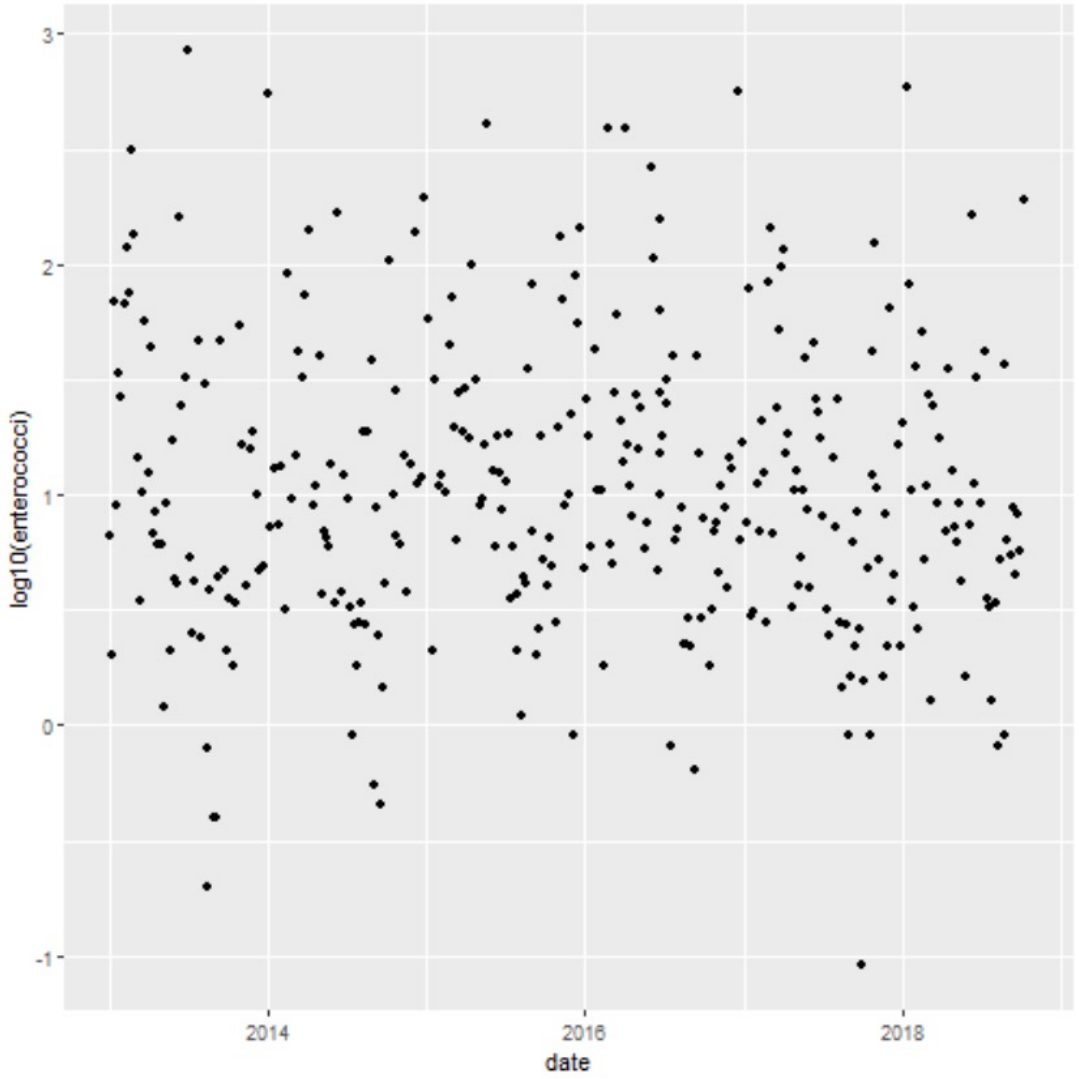


# exercise 3:
# geoms + more layering

corresponds to:
exercises > ex03_geoms.R

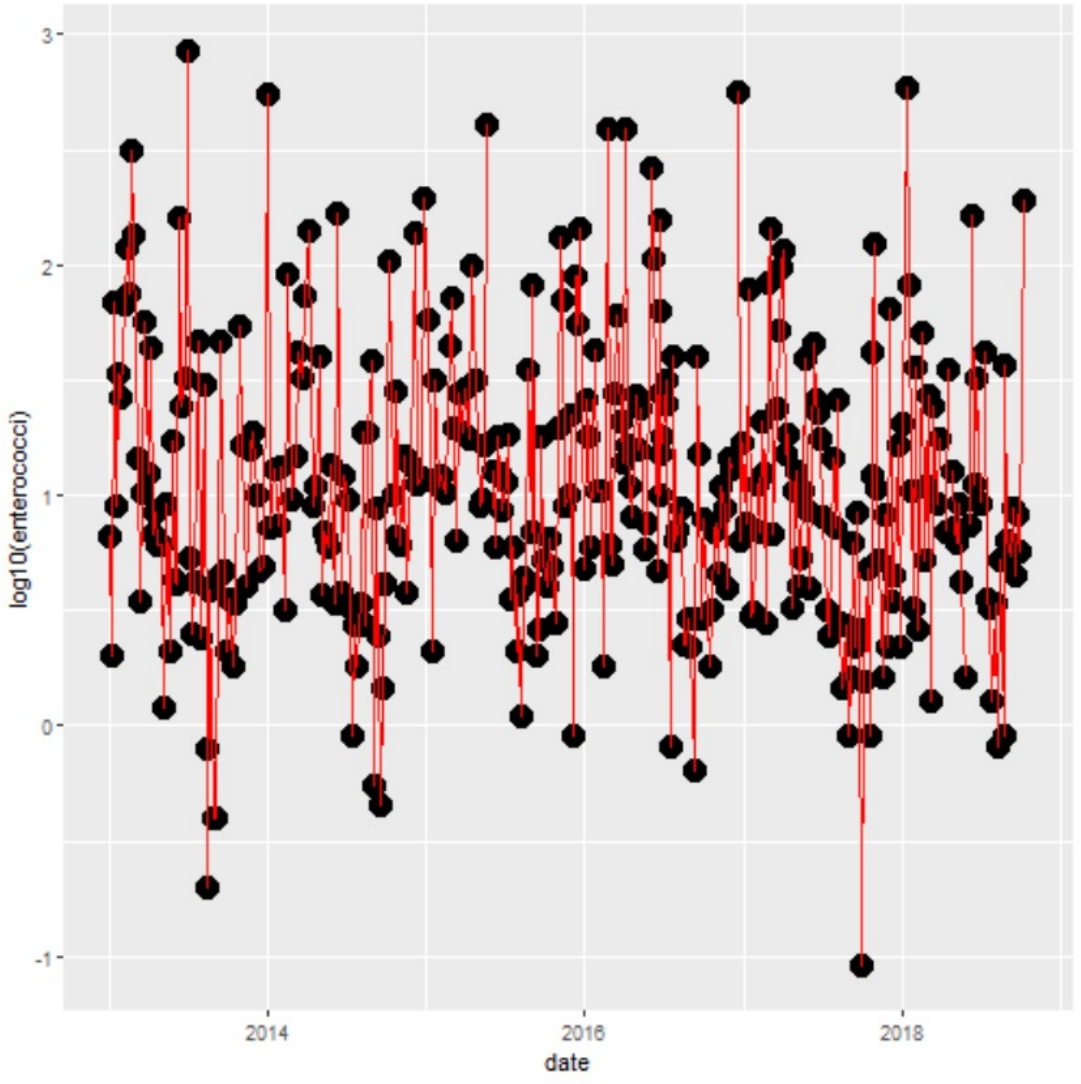
Point and lines are “simple” geoms.

```
ggplot(
  data = beaches,
  mapping = aes(
    x = date,
    y = log10(enterococci)
  )
) +
geom_point()
```



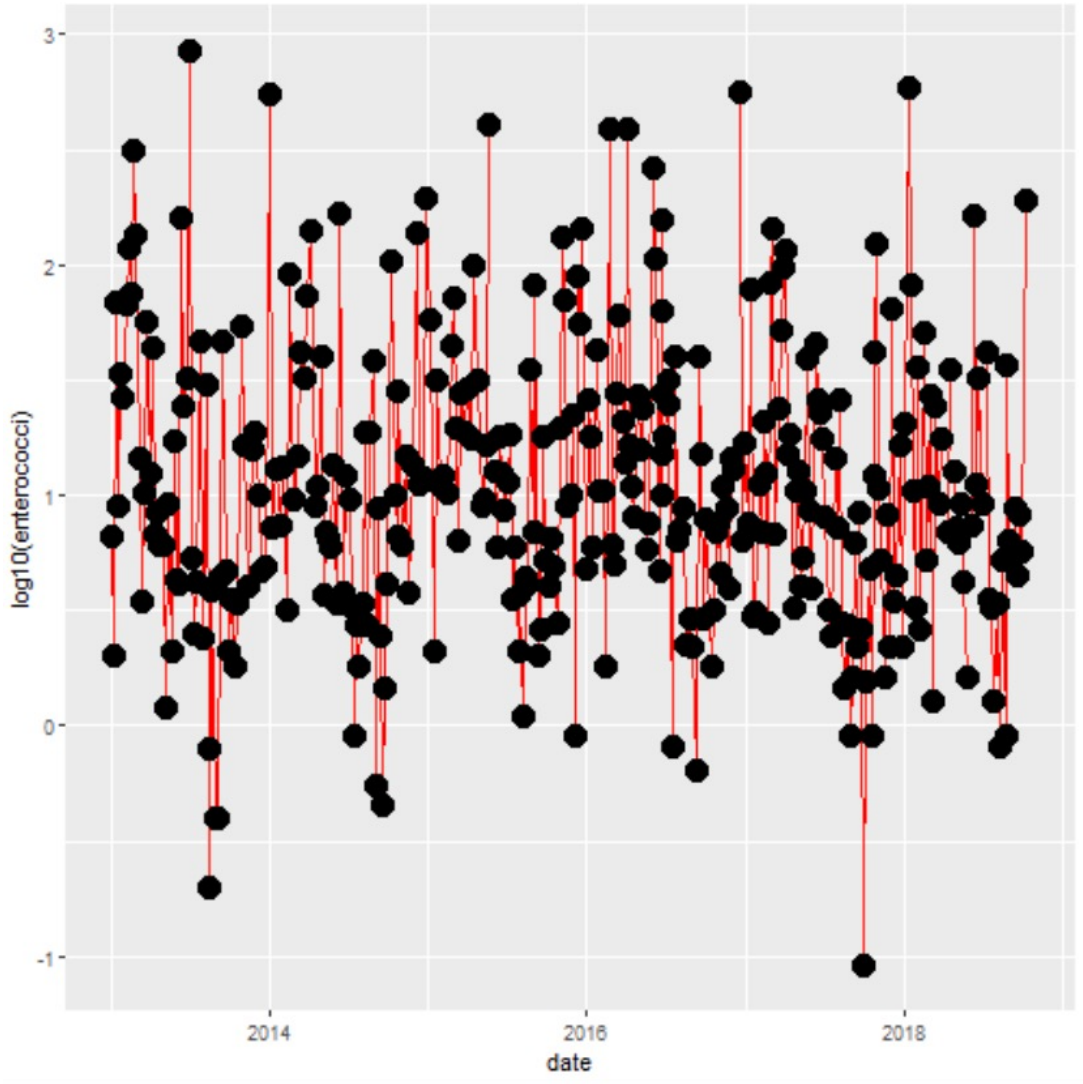
We can layer geoms on top of each other.

```
ggplot(
  data = beaches,
  mapping = aes(
    x = date,
    y = log10(enterococci)
  )
) +
geom_point(size = 5) +
geom_line(colour = "red")
```



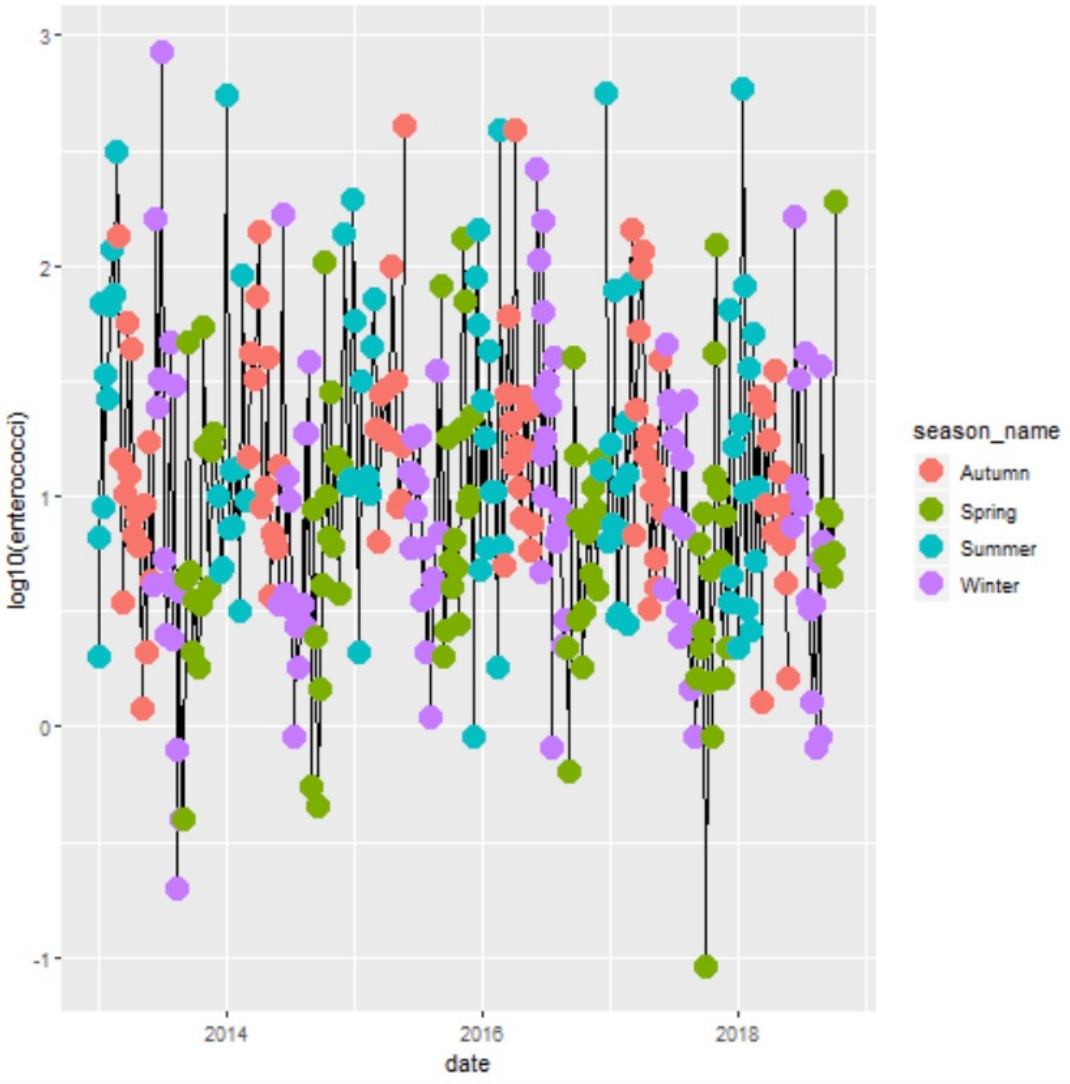
# The order of layers matters!

```
ggplot(
  data = beaches,
  mapping = aes(
    x = date,
    y = log10(enterococci)
  )
) +
geom_line(colour = "red") +
geom_point(size = 5)
```



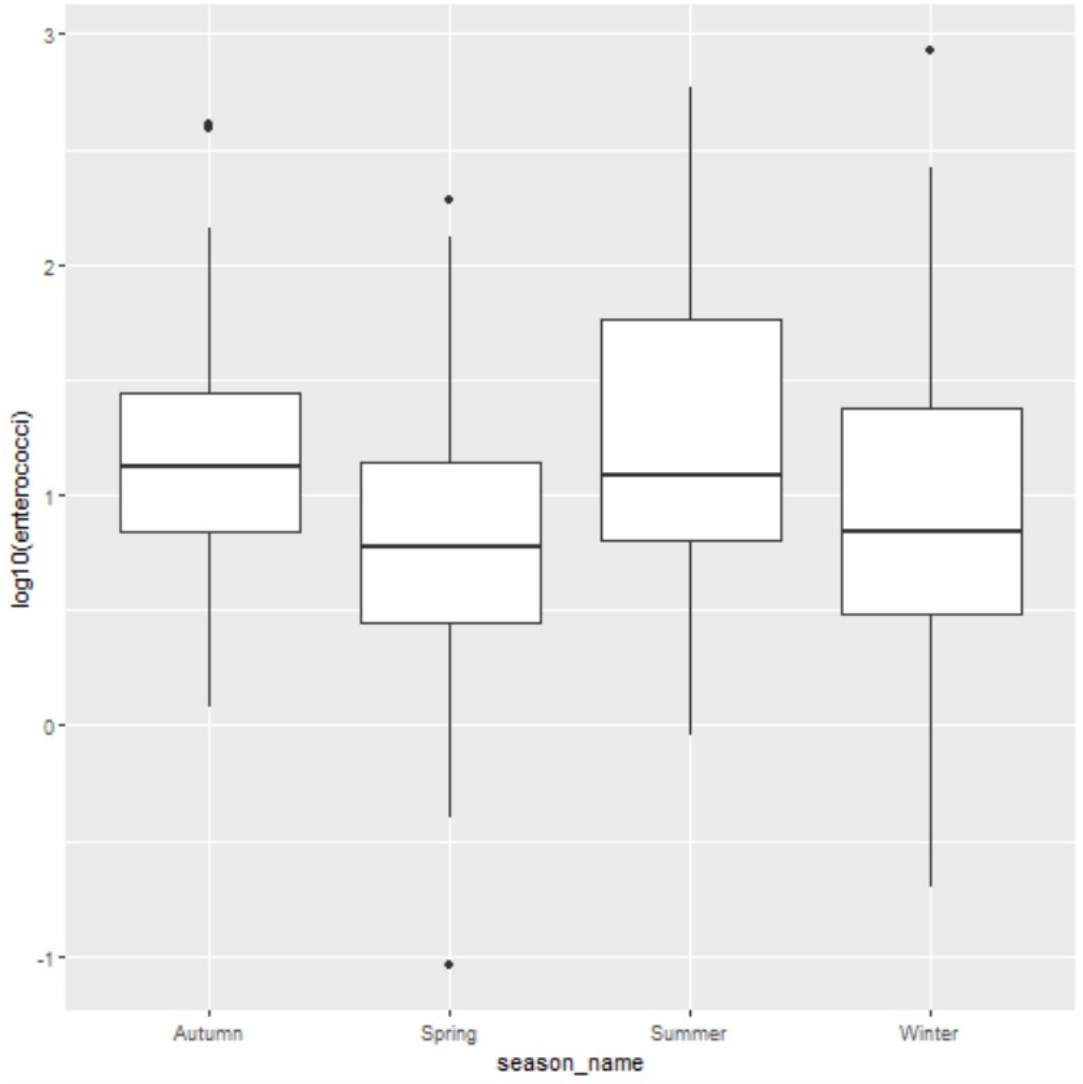
Geoms can have different aesthetic mappings.

```
ggplot(
  data = beaches,
  mapping = aes(
    x = date,
    y = log10(enterococci)) ) +
geom_line(
  linewidth = 2,
  colour = "black") +
geom_point(
  mapping = aes(colour = season_name),
  size = 5)
```



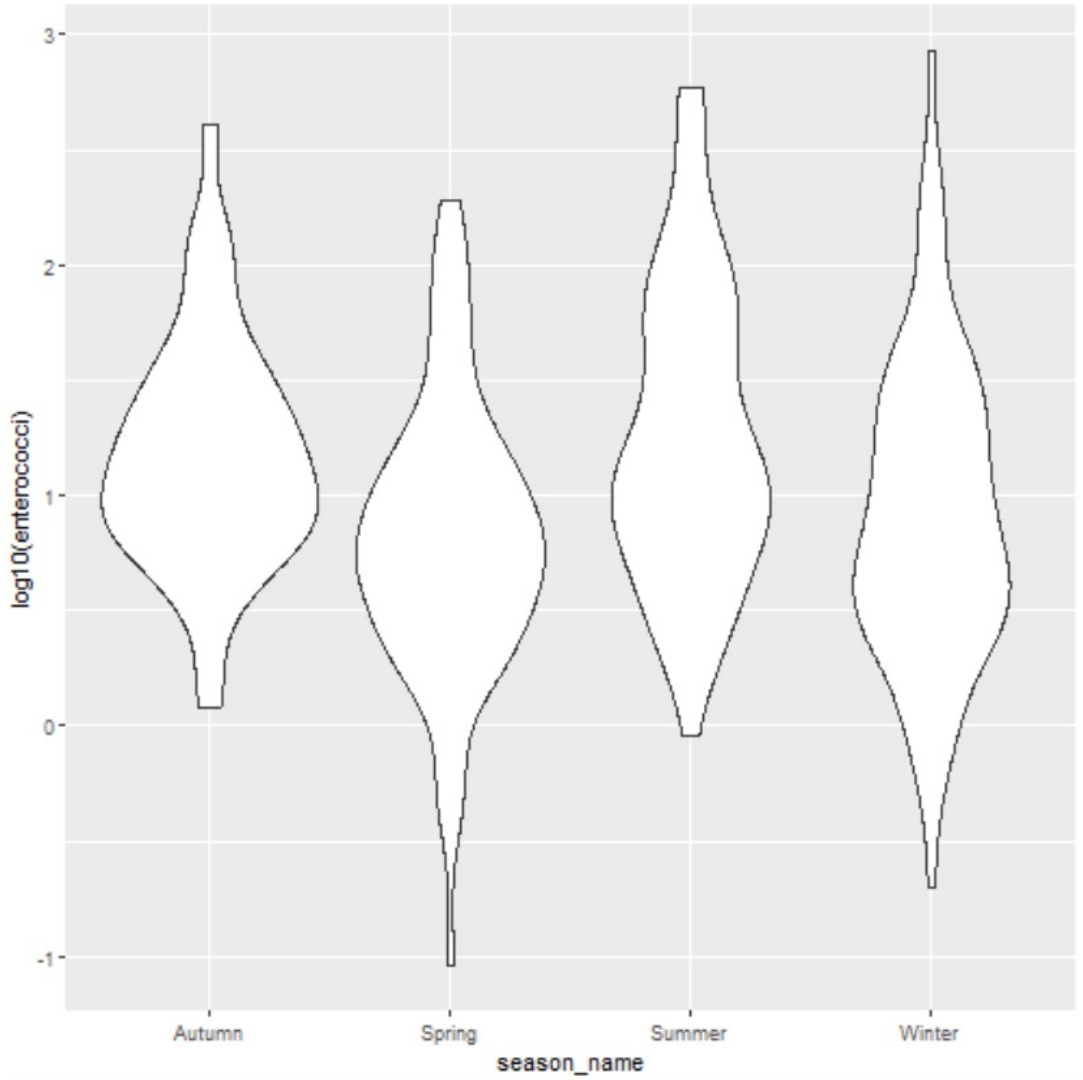
Geoms come in many useful varieties.

```
ggplot(
  data = beaches,
  mapping = aes(
    x = season_name,
    y = log10(enterococci))
  ) +
geom_boxplot()
```



Geoms come in many useful varieties.

```
ggplot(
  data = beaches,
  mapping = aes(
    x = season_name,
    y = log10(enterococci))
  ) +
geom_violin()
```

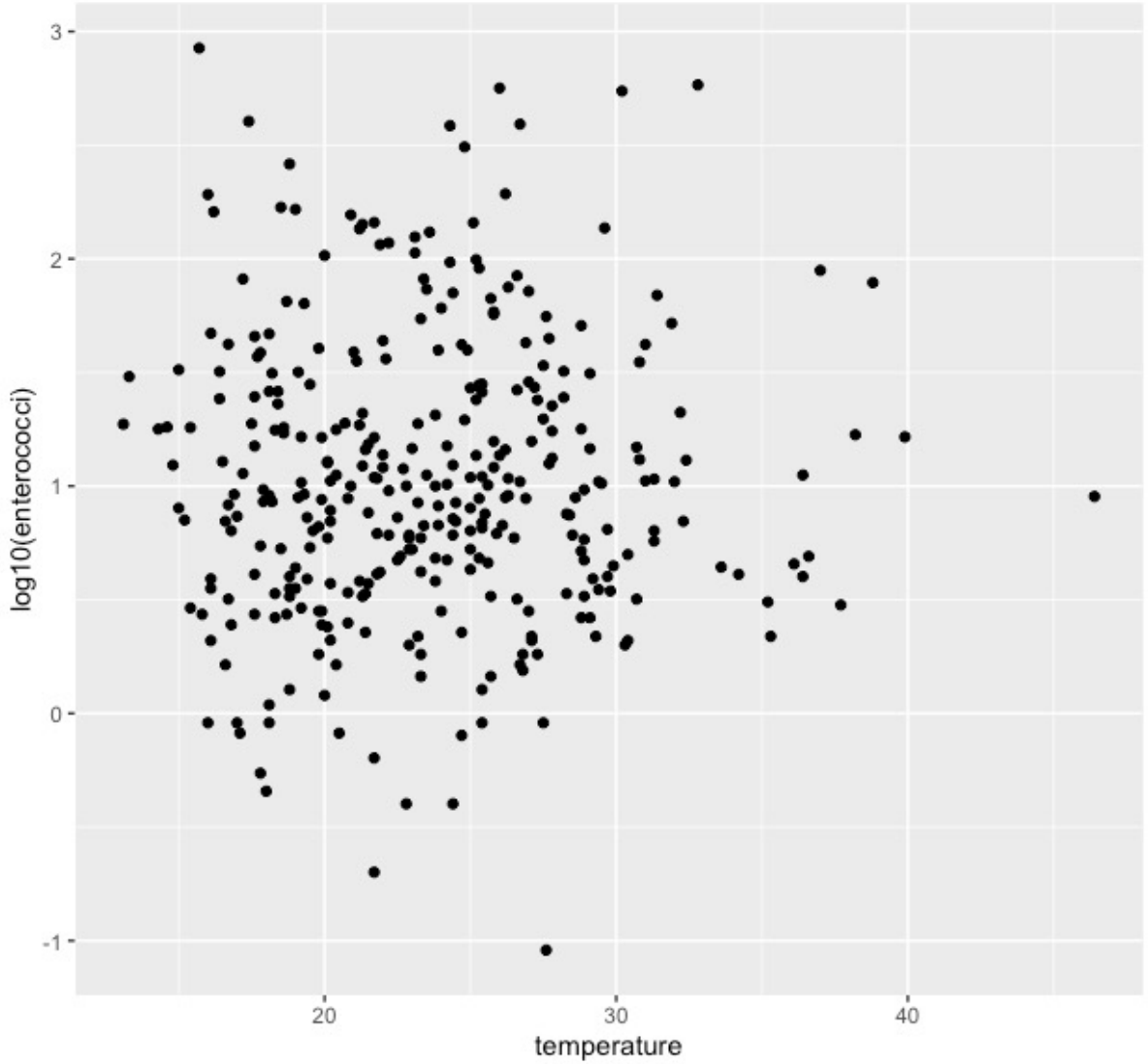


# exercise 4:
# facets

corresponds to:
exercises > ex04_facets.R

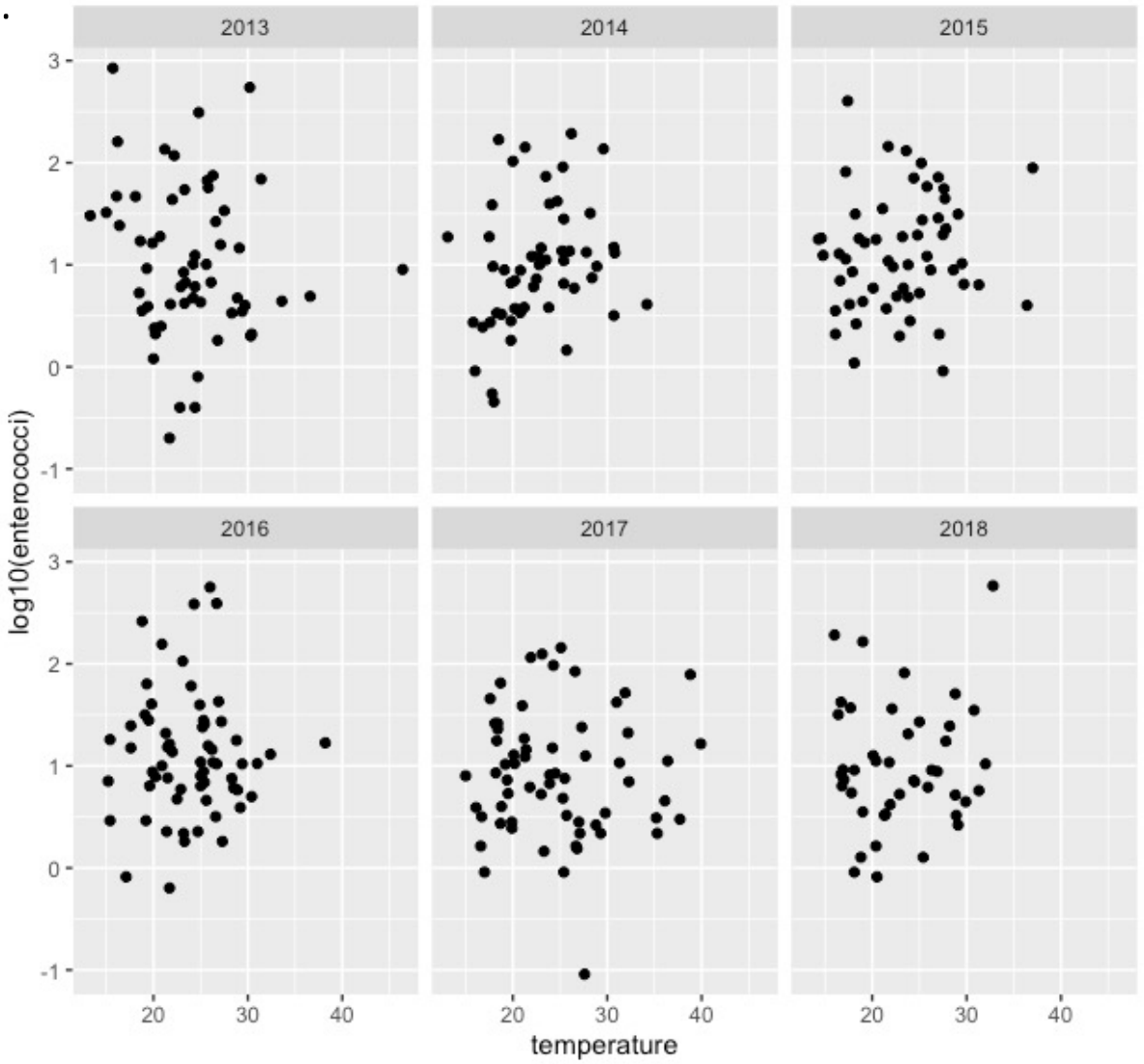
Basic viz of relationship between enterococci levels & temperature.

```
ggplot(
     data = beaches,
     mapping = aes(
          x = temperature,
          y = log10(enterococci)
     )
) +
geom_point()
```



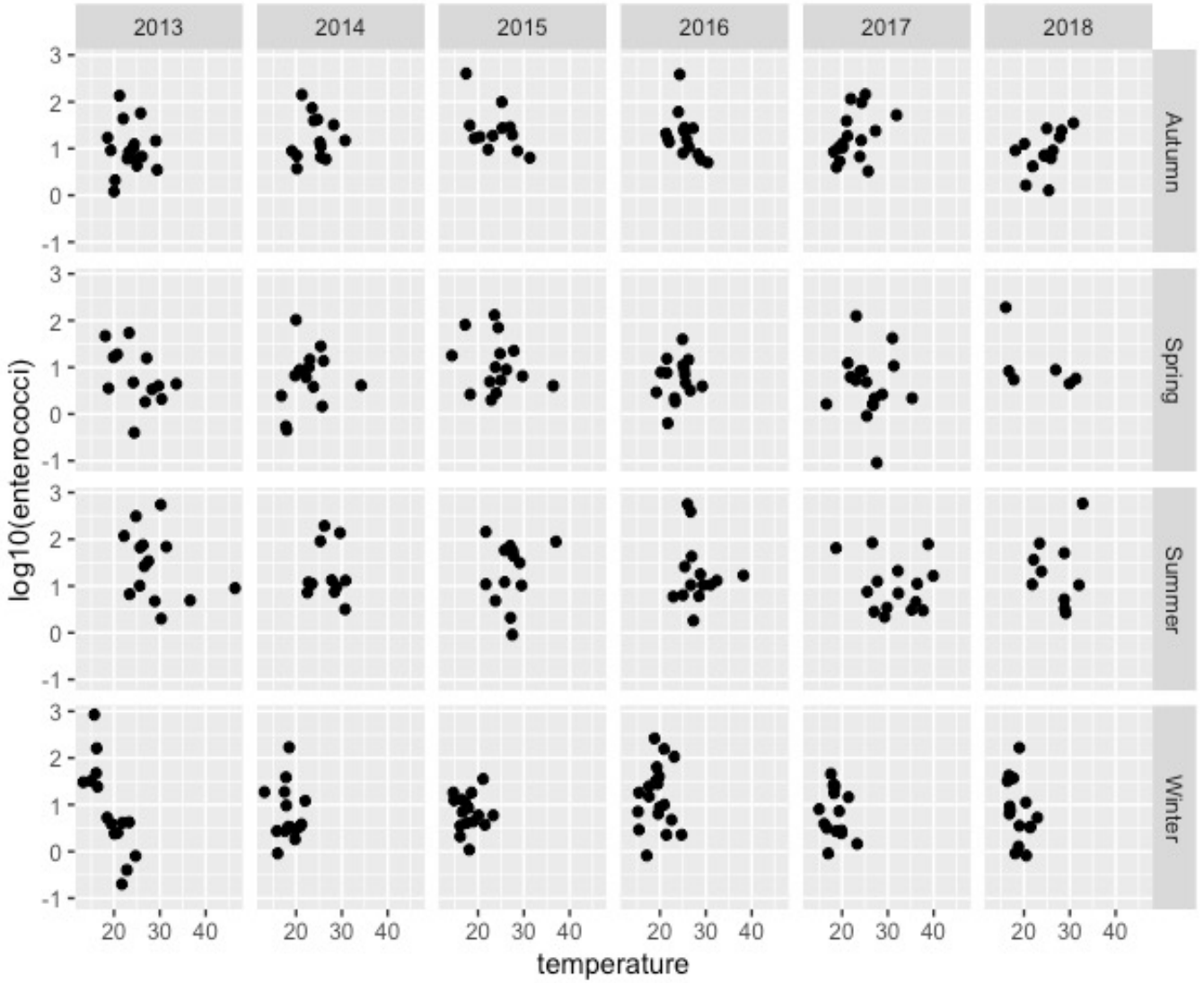
See if relationship changes between years.

```
ggplot(
     data = beaches,
     mapping = aes(
          x = temperature,
          y = log10(enterococci)
     )
) +
geom_point() +
facet_wrap(~year)
```



facet_grid lets us do even more.

```
ggplot(
     data = beaches,
     mapping = aes(
          x = temperature,
          y = log10(enterococci)
     )
) +
geom_point() +
facet_grid(season_name ~ year)
```



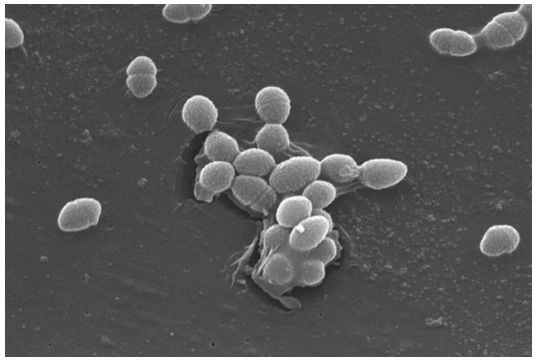
psst! have you taken a break yet?

# exercise 5: data wrangling

corresponds to:
exercises > ex05_wrangling.R

# our data

**goal:** use enterococci data to help inform our beach-going decisions.



```
beaches2
```

```
# A tibble: 3,690 x 18
   council  long   lat date       site  enterococci    id region rainfall temperature  year month    day season day_num month_num month_name
   <chr>   <dbl> <dbl> <date>     <chr>       <dbl> <dbl> <chr>     <dbl>       <dbl> <dbl> <dbl>  <dbl>  <dbl>   <dbl>     <dbl> <chr>
 1 Randwi…  151. -33.9 2013-01-02 Clov…          19    25 Sydne…      0          23.4  2013     1      2      1       2         1 January
 2 Randwi…  151. -33.9 2013-01-06 Clov…           3    25 Sydne…      0          30.3  2013     1      6      1       6         1 January
 3 Randwi…  151. -33.9 2013-01-12 Clov…           2    25 Sydne…      0          31.4  2013     1     12      1      12         1 January
 4 Randwi…  151. -33.9 2013-01-18 Clov…          13    25 Sydne…      0          46.4  2013     1     18      1      18         1 January
 5 Randwi…  151. -33.9 2013-01-30 Clov…           8    25 Sydne…      0.6        26.6  2013     1     30      1      30         1 January
 6 Randwi…  151. -33.9 2013-02-05 Clov…           7    25 Sydne…      0.1        25.7  2013     2      5      1      36         2 February
 7 Randwi…  151. -33.9 2013-02-11 Clov…          11    25 Sydne…      8          22.2  2013     2     11      1      42         2 February
 8 Randwi…  151. -33.9 2013-02-23 Clov…          97    25 Sydne…      7.2        24.8  2013     2     23      1      54         2 February
 9 Randwi…  151. -33.9 2013-03-07 Clov…           3    25 Sydne…      0          29.1  2013     3      7      2      66         3 March
10 Randwi…  151. -33.9 2013-03-25 Clov…           0    25 Sydne…      0          25.8  2013     3     25      2      84         3 March
# … with 3,680 more rows, and 1 more variable: season_name <chr>
```

# data wrangling

**what?** manipulate data

**why?** real-life data is messy

**how?** with *tidyverse*

# the %>%

regular code

```
beaches2
```

piped version

```
beaches2
```

# the %>%

regular code

```
do_one_thing(
     beaches2
)
```

piped version

```
beaches2 %>%
     do_one_thing()
```

# the %>%

## regular code

```
then_another(
     do_one_thing(
          beaches2
     )
)
```

## piped version

```
beaches2 %>%
     do_one_thing() %>%
     then_another()
```

# the %>%

## regular code

```
and_then_one_more(
     then_another(
          do_one_thing(
               beaches2
          )
     )
)
```

## piped version

```
beaches2 %>%
     do_one_thing() %>%
     then_another() %>%
     and_then_one_more()
```

# the %>%

## regular code

```
then_finally_this(
     and_then_one_more(
          then_another(
               do_one_thing(
                    beaches2
               )
          )
     )
)
```

## piped version

```
beaches2 %>%
     do_one_thing() %>%
     then_another() %>%
     and_then_one_more() %>%
     then_finally_this()
```

# why?

- we pipe our dataset through operations
- the data flows through our analysis
- improves readability

piped version

```
beaches2 %>%
     do_one_thing() %>%
     then_another() %>%
     and_then_one_more() %>%
     then_finally_this()
```

```
beaches2
```

```
# A tibble: 3,690 x 18
   council  long   lat date       site  enterococci    id region rainfall temperature  year month   day season day_num month_num month_name
   <chr>   <dbl> <dbl> <date>     <chr>       <dbl> <dbl> <chr>     <dbl>       <dbl> <dbl> <dbl> <dbl>  <dbl>   <dbl>     <dbl> <chr>
 1 Randwi…  151. -33.9 2013-01-02 Clov…          19    25 Sydne…      0          23.4  2013     1     2      1       2         1 January
 2 Randwi…  151. -33.9 2013-01-06 Clov…           3    25 Sydne…      0          30.3  2013     1     6      1       6         1 January
 3 Randwi…  151. -33.9 2013-01-12 Clov…           2    25 Sydne…      0          31.4  2013     1    12      1      12         1 January
 4 Randwi…  151. -33.9 2013-01-18 Clov…          13    25 Sydne…      0          46.4  2013     1    18      1      18         1 January
 5 Randwi…  151. -33.9 2013-01-30 Clov…           8    25 Sydne…      0.6        26.6  2013     1    30      1      30         1 January
 6 Randwi…  151. -33.9 2013-02-05 Clov…           7    25 Sydne…      0.1        25.7  2013     2     5      1      36         2 February
 7 Randwi…  151. -33.9 2013-02-11 Clov…          11    25 Sydne…      8          22.2  2013     2    11      1      42         2 February
 8 Randwi…  151. -33.9 2013-02-23 Clov…          97    25 Sydne…      7.2        24.8  2013     2    23      1      54         2 February
 9 Randwi…  151. -33.9 2013-03-07 Clov…           3    25 Sydne…      0          29.1  2013     3     7      2      66         3 March
10 Randwi…  151. -33.9 2013-03-25 Clov…           0    25 Sydne…      0          25.8  2013     3    25      2      84         3 March
# … with 3,680 more rows, and 1 more variable: season_name <chr>
```

# select (drop columns)

```
beaches2 %>%
    select(-c(site, year, month, day, temperature))
```

```
# A tibble: 3,690 x 13
   council           long   lat date       enterococci    id region                    rainfall season day_num month_num month_name season_nar
   <chr>            <dbl> <dbl> <date>           <dbl> <dbl> <chr>                        <dbl>  <dbl>   <dbl>     <dbl> <chr>      <chr>
 1 Randwick Council  151. -33.9 2013-01-02          19    25 Sydney City Ocean Beaches      0        1       2         1 January    Summer
 2 Randwick Council  151. -33.9 2013-01-06           3    25 Sydney City Ocean Beaches      0        1       6         1 January    Summer
 3 Randwick Council  151. -33.9 2013-01-12           2    25 Sydney City Ocean Beaches      0        1      12         1 January    Summer
 4 Randwick Council  151. -33.9 2013-01-18          13    25 Sydney City Ocean Beaches      0        1      18         1 January    Summer
 5 Randwick Council  151. -33.9 2013-01-30           8    25 Sydney City Ocean Beaches      0.6      1      30         1 January    Summer
 6 Randwick Council  151. -33.9 2013-02-05           7    25 Sydney City Ocean Beaches      0.1      1      36         2 February   Summer
 7 Randwick Council  151. -33.9 2013-02-11          11    25 Sydney City Ocean Beaches      8        1      42         2 February   Summer
 8 Randwick Council  151. -33.9 2013-02-23          97    25 Sydney City Ocean Beaches      7.2      1      54         2 February   Summer
 9 Randwick Council  151. -33.9 2013-03-07           3    25 Sydney City Ocean Beaches      0        2      66         3 March      Autumn
10 Randwick Council  151. -33.9 2013-03-25           0    25 Sydney City Ocean Beaches      0        2      84         3 March      Autumn
# … with 3,680 more rows
```

# select (keep & order columns)

```
beaches2 %>%
    select(c(site, year, month, day, temperature))
```

```
# A tibble: 3,690 x 5
   site            year month   day temperature
   <chr>          <dbl> <dbl> <dbl>       <dbl>
 1 Clovelly Beach  2013     1     2        23.4
 2 Clovelly Beach  2013     1     6        30.3
 3 Clovelly Beach  2013     1    12        31.4
 4 Clovelly Beach  2013     1    18        46.4
 5 Clovelly Beach  2013     1    30        26.6
 6 Clovelly Beach  2013     2     5        25.7
 7 Clovelly Beach  2013     2    11        22.2
 8 Clovelly Beach  2013     2    23        24.8
 9 Clovelly Beach  2013     3     7        29.1
10 Clovelly Beach  2013     3    25        25.8
# … with 3,680 more rows
```

# arrange

```
beaches2 %>%
     select(c(site, year, month, day, temperature)) %>%
     arrange(temperature)
```

```
# A tibble: 3,690 x 5
   site                    year month   day temperature
   <chr>                  <dbl> <dbl> <dbl>       <dbl>
 1 Clovelly Beach          2014     8    11        13.1
 2 Coogee Beach            2014     8    11        13.1
 3 Gordons Bay (East)      2014     8    11        13.1
 4 Little Bay Beach        2014     8    11        13.1
 5 Malabar Beach           2014     8    11        13.1
 6 Maroubra Beach          2014     8    11        13.1
 7 South Maroubra Beach    2014     8    11        13.1
 8 South Maroubra Rockpool 2014     8    11        13.1
 9 Bondi Beach             2014     8    11        13.1
10 Bronte Beach            2014     8    11        13.1
# … with 3,680 more rows
```

# arrange

```
beaches2 %>%
     select(c(site, year, month, day, temperature)) %>%
     arrange(site, temperature)
```

```
# A tibble: 3,690 x 5
   site         year month   day temperature
   <chr>       <dbl> <dbl> <dbl>       <dbl>
 1 Bondi Beach  2014     8    11        13.1
 2 Bondi Beach  2013     8     8        13.3
 3 Bondi Beach  2015     9    23        14.3
 4 Bondi Beach  2015     7     8        14.6
 5 Bondi Beach  2015     6    20        14.8
 6 Bondi Beach  2013     6    24        15
 7 Bondi Beach  2017     6    30        15
 8 Bondi Beach  2016     8     2        15.2
 9 Bondi Beach  2016     6    28        15.4
10 Bondi Beach  2016     8    26        15.4
# … with 3,680 more rows
```

# filter (+ necessary logical operators)

```
beaches2 %>%
     select(c(site, year, month, day, temperature)) %>%
     arrange(site, temperature)
     filter(year == 2014)
```

```
# A tibble: 582 x 5
   site         year month   day temperature
   <chr>       <dbl> <dbl> <dbl>       <dbl>
 1 Bondi Beach  2014     8    11        13.1
 2 Bondi Beach  2014     7    18        15.8
 3 Bondi Beach  2014     7    14        16
 4 Bondi Beach  2014     9    12        16.8
 5 Bondi Beach  2014     8    21        17.5
 6 Bondi Beach  2014     8    15        17.6
 7 Bondi Beach  2014     9     2        17.8
 8 Bondi Beach  2014     8    27        17.8
 9 Bondi Beach  2014     7     3        17.9
10 Bondi Beach  2014     9    18        18
# … with 572 more rows
```

# filter (+ necessary logical operators)

```
beaches2 %>%
     select(c(site, year, month, day, temperature)) %>%
     arrange(site, temperature)
     filter(site == “Malabar Beach”)
```

```
# A tibble: 343 x 5
   site           year month   day temperature
   <chr>         <dbl> <dbl> <dbl>       <dbl>
 1 Malabar Beach  2014     8    11        13.1
 2 Malabar Beach  2013     8     8        13.3
 3 Malabar Beach  2015     9    23        14.3
 4 Malabar Beach  2015     7     8        14.6
 5 Malabar Beach  2015     6    20        14.8
 6 Malabar Beach  2013     6    24        15
 7 Malabar Beach  2017     6    30        15
 8 Malabar Beach  2016     8     2        15.2
 9 Malabar Beach  2016     6    28        15.4
10 Malabar Beach  2016     8    26        15.4
# … with 333 more rows
```

# filter (+ necessary logical operators)

```
beaches2 %>%
     select(c(site, year, month, day, temperature)) %>%
     arrange(site, temperature)
     filter(site == “Malabar Beach” & year == 2014)
```

```
# A tibble: 53 x 5
   site           year month   day temperature
   <chr>         <dbl> <dbl> <dbl>       <dbl>
 1 Malabar Beach  2014     8    11        13.1
 2 Malabar Beach  2014     7    18        15.8
 3 Malabar Beach  2014     7    14        16
 4 Malabar Beach  2014     9    12        16.8
 5 Malabar Beach  2014     8    21        17.5
 6 Malabar Beach  2014     8    15        17.6
 7 Malabar Beach  2014     8    27        17.8
 8 Malabar Beach  2014     9     2        17.8
 9 Malabar Beach  2014     7     3        17.9
10 Malabar Beach  2014     9    18        18
# … with 43 more rows
```

# logical operators refresher

== equality
!= inequality
> greater than
>= greater than or equal to
< less than
<= less than or equal to

& AND
| OR
! NOT

```
> 10 < 100
[1] TRUE

> 2 + 2 == 5
[1] FALSE
```

```
> (10 < 100) | (2 + 2 == 5)
[1] TRUE

> (10 < 100) & (2 + 2 == 5)
[1] FALSE
```

# mutate

```
beaches2
```

```
# A tibble: 3,690 x 18
   council  long   lat date       site  enterococci    id region rainfall temperature  year month   day season day_num month_num month_name
   <chr>   <dbl> <dbl> <date>     <chr>       <dbl> <dbl> <chr>     <dbl>       <dbl> <dbl> <dbl> <dbl>  <dbl>   <dbl>     <dbl> <chr>
 1 Randwi…  151. -33.9 2013-01-02 Clov…          19    25 Sydne…      0          23.4  2013     1     2      1       2         1 January
 2 Randwi…  151. -33.9 2013-01-06 Clov…           3    25 Sydne…      0          30.3  2013     1     6      1       6         1 January
 3 Randwi…  151. -33.9 2013-01-12 Clov…           2    25 Sydne…      0          31.4  2013     1    12      1      12         1 January
 4 Randwi…  151. -33.9 2013-01-18 Clov…          13    25 Sydne…      0          46.4  2013     1    18      1      18         1 January
 5 Randwi…  151. -33.9 2013-01-30 Clov…           8    25 Sydne…      0.6        26.6  2013     1    30      1      30         1 January
 6 Randwi…  151. -33.9 2013-02-05 Clov…           7    25 Sydne…      0.1        25.7  2013     2     5      1      36         2 February
 7 Randwi…  151. -33.9 2013-02-11 Clov…          11    25 Sydne…      8          22.2  2013     2    11      1      42         2 February
 8 Randwi…  151. -33.9 2013-02-23 Clov…          97    25 Sydne…      7.2        24.8  2013     2    23      1      54         2 February
 9 Randwi…  151. -33.9 2013-03-07 Clov…           3    25 Sydne…      0          29.1  2013     3     7      2      66         3 March
10 Randwi…  151. -33.9 2013-03-25 Clov…           0    25 Sydne…      0          25.8  2013     3    25      2      84         3 March
# … with 3,680 more rows, and 1 more variable: season_name <chr>
```

# mutate

```
beaches2 %>%
     mutate(id_real = id*100)
```

```
# A tibble: 3,690 x 18
   council  long   lat date       site  enterococci    id id_real region rainfall temperature  year month    day season day_num month_num
   <chr>   <dbl> <dbl> <date>     <chr>       <dbl> <dbl>   <dbl> <chr>      <dbl>       <dbl> <dbl> <dbl>  <dbl>  <dbl>   <dbl>     <dbl>
 1 Randwi…  151. -33.9 2013-01-02 Clov…          19    25    2500 Sydne…       0          23.4  2013     1      2      1       2         1
 2 Randwi…  151. -33.9 2013-01-06 Clov…           3    25    2500 Sydne…       0          30.3  2013     1      6      1       6         1
 3 Randwi…  151. -33.9 2013-01-12 Clov…           2    25    2500 Sydne…       0          31.4  2013     1     12      1      12         1
 4 Randwi…  151. -33.9 2013-01-18 Clov…          13    25    2500 Sydne…       0          46.4  2013     1     18      1      18         1
 5 Randwi…  151. -33.9 2013-01-30 Clov…           8    25    2500 Sydne…       0.6        26.6  2013     1     30      1      30         1
 6 Randwi…  151. -33.9 2013-02-05 Clov…           7    25    2500 Sydne…       0.1        25.7  2013     2      5      1      36         2
 7 Randwi…  151. -33.9 2013-02-11 Clov…          11    25    2500 Sydne…       8          22.2  2013     2     11      1      42         2
 8 Randwi…  151. -33.9 2013-02-23 Clov…          97    25    2500 Sydne…       7.2        24.8  2013     2     23      1      54         2
 9 Randwi…  151. -33.9 2013-03-07 Clov…           3    25    2500 Sydne…       0          29.1  2013     3      7      2      66         3
10 Randwi…  151. -33.9 2013-03-25 Clov…           0    25    2500 Sydne…       0          25.8  2013     3     25      2      84         3
# … with 3,680 more rows, and 1 more variable: month_name <chr>
```

*

# group_by and summarise

```
beaches2 %>%
     group_by( GROUP ) %>%
     summarise( EXPRESSION ) %>%
     ungroup()
```

use `group_by()` to define groups

use `summarise()` to create a new summary variable

where `EXPRESSION` usually looks like `NEW VARIABLE = DOTHIS(VARIABLE)`

*e.g.* `sd_rain = sd(rainfall)`

use `ungroup()` to remove grouping

# group_by and summarise

```
beaches2
```

```
# A tibble: 3,690 x 18
   council  long   lat date       site  enterococci    id region rainfall temperature  year month   day season day_num month_num month_name
   <chr>   <dbl> <dbl> <date>     <chr>       <dbl> <dbl> <chr>     <dbl>       <dbl> <dbl> <dbl> <dbl>  <dbl>   <dbl>     <dbl> <chr>
 1 Randwi…  151. -33.9 2013-01-02 Clov…          19    25 Sydne…      0          23.4  2013     1     2      1       2         1 January
 2 Randwi…  151. -33.9 2013-01-06 Clov…           3    25 Sydne…      0          30.3  2013     1     6      1       6         1 January
 3 Randwi…  151. -33.9 2013-01-12 Clov…           2    25 Sydne…      0          31.4  2013     1    12      1      12         1 January
 4 Randwi…  151. -33.9 2013-01-18 Clov…          13    25 Sydne…      0          46.4  2013     1    18      1      18         1 January
 5 Randwi…  151. -33.9 2013-01-30 Clov…           8    25 Sydne…      0.6        26.6  2013     1    30      1      30         1 January
 6 Randwi…  151. -33.9 2013-02-05 Clov…           7    25 Sydne…      0.1        25.7  2013     2     5      1      36         2 February
 7 Randwi…  151. -33.9 2013-02-11 Clov…          11    25 Sydne…      8          22.2  2013     2    11      1      42         2 February
 8 Randwi…  151. -33.9 2013-02-23 Clov…          97    25 Sydne…      7.2        24.8  2013     2    23      1      54         2 February
 9 Randwi…  151. -33.9 2013-03-07 Clov…           3    25 Sydne…      0          29.1  2013     3     7      2      66         3 March
10 Randwi…  151. -33.9 2013-03-25 Clov…           0    25 Sydne…      0          25.8  2013     3    25      2      84         3 March
# … with 3,680 more rows, and 1 more variable: season_name <chr>
```

# group_by and summarise

```
beaches2 %>%
    group_by(site, year)
```

```
# A tibble: 3,690 x 18
# Groups:   site, year [66]
   council  long   lat date       site  enterococci    id region rainfall temperature  year month   day season day_num month_num month_name
   <chr>   <dbl> <dbl> <date>     <chr>       <dbl> <dbl> <chr>     <dbl>       <dbl> <dbl> <dbl> <dbl>  <dbl>   <dbl>     <dbl> <chr>
 1 Randwi…  151. -33.9 2013-01-02 Clov…          19    25 Sydne…      0          23.4  2013     1     2      1       2         1 January
 2 Randwi…  151. -33.9 2013-01-06 Clov…           3    25 Sydne…      0          30.3  2013     1     6      1       6         1 January
 3 Randwi…  151. -33.9 2013-01-12 Clov…           2    25 Sydne…      0          31.4  2013     1    12      1      12         1 January
 4 Randwi…  151. -33.9 2013-01-18 Clov…          13    25 Sydne…      0          46.4  2013     1    18      1      18         1 January
 5 Randwi…  151. -33.9 2013-01-30 Clov…           8    25 Sydne…      0.6        26.6  2013     1    30      1      30         1 January
 6 Randwi…  151. -33.9 2013-02-05 Clov…           7    25 Sydne…      0.1        25.7  2013     2     5      1      36         2 February
 7 Randwi…  151. -33.9 2013-02-11 Clov…          11    25 Sydne…      8          22.2  2013     2    11      1      42         2 February
 8 Randwi…  151. -33.9 2013-02-23 Clov…          97    25 Sydne…      7.2        24.8  2013     2    23      1      54         2 February
 9 Randwi…  151. -33.9 2013-03-07 Clov…           3    25 Sydne…      0          29.1  2013     3     7      2      66         3 March
10 Randwi…  151. -33.9 2013-03-25 Clov…           0    25 Sydne…      0          25.8  2013     3    25      2      84         3 March
# … with 3,680 more rows, and 1 more variable: season_name <chr>
```

# group_by and summarise

```
beaches2 %>%
     group_by(site, year) %>%
     summarise(avg_rain = mean(rainfall)) %>%
     ungroup()
```

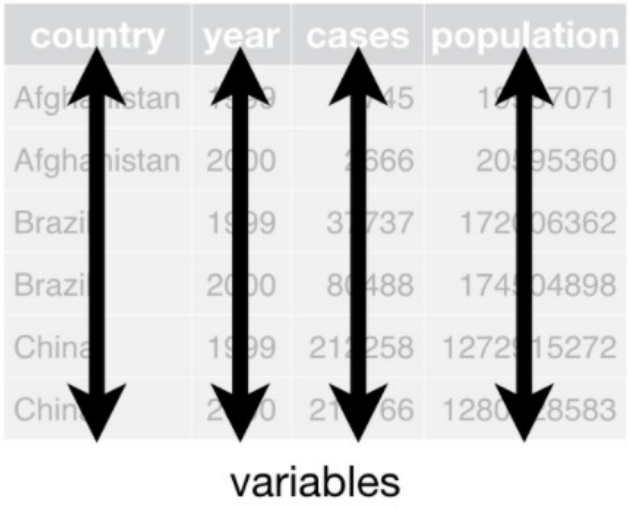
```
# A tibble: 66 x 3
   site          year avg_rain
   <chr>        <dbl>    <dbl>
 1 Bondi Beach   2013    NA
 2 Bondi Beach   2014    NA
 3 Bondi Beach   2015     5.58
 4 Bondi Beach   2016     5.39
 5 Bondi Beach   2017     4.09
 6 Bondi Beach   2018     2.39
 7 Bronte Beach  2013    NA
 8 Bronte Beach  2014    NA
 9 Bronte Beach  2015     5.58
10 Bronte Beach  2016     5.39
# … with 56 more rows
```

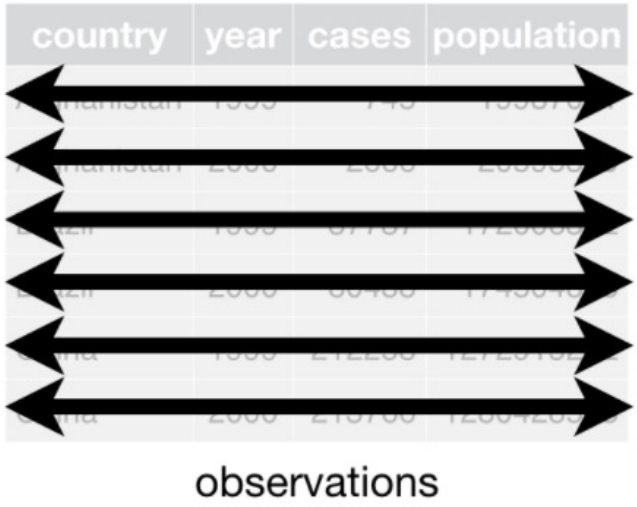
# group_by and summarise

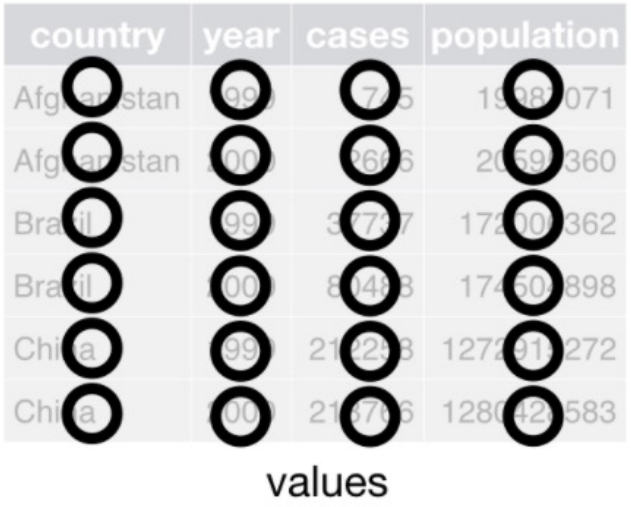
```
beaches2 %>%
     group_by(site, year) %>%
     summarise(avg_rain = mean(rainfall, na.rm=TRUE)) %>%
     ungroup()
```

```
# A tibble: 66 x 3
   site          year avg_rain
   <chr>        <dbl>    <dbl>
 1 Bondi Beach   2013     4.64
 2 Bondi Beach   2014     2.60
 3 Bondi Beach   2015     5.58
 4 Bondi Beach   2016     5.39
 5 Bondi Beach   2017     4.09
 6 Bondi Beach   2018     2.39
 7 Bronte Beach  2013     4.64
 8 Bronte Beach  2014     2.60
 9 Bronte Beach  2015     5.58
10 Bronte Beach  2016     5.39
# … with 56 more rows
```

# fyi: ggplot2 needs a “tidy” data format




this is sometimes called a “long” format

# pivoting from long to wide

```
rain_data <- beaches2 %>%
     group_by(site, year) %>%
     summarise(avg_rain = mean(rainfall, na.rm=TRUE)) %>%
     ungroup()

rain_data
```

```
# A tibble: 66 x 3
   site          year avg_rain
   <chr>        <dbl>    <dbl>
 1 Bondi Beach   2013     4.64
 2 Bondi Beach   2014     2.60
 3 Bondi Beach   2015     5.58
 4 Bondi Beach   2016     5.39
 5 Bondi Beach   2017     4.09
 6 Bondi Beach   2018     2.39
 7 Bronte Beach  2013     4.64
 8 Bronte Beach  2014     2.60
 9 Bronte Beach  2015     5.58
10 Bronte Beach  2016     5.39
# … with 56 more rows
```

# pivoting from long to wide

```
rain_data %>%
    pivot_wider(names_from = year, values_from = avg_rain)
```

```
# A tibble: 11 x 7
   site                    `2013` `2014` `2015` `2016` `2017` `2018`
   <chr>                    <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
 1 Bondi Beach               4.64   2.60   5.58   5.39   4.09   2.39
 2 Bronte Beach              4.64   2.60   5.58   5.39   4.09   2.39
 3 Clovelly Beach            4.64   2.60   5.58   5.39   4.09   2.39
 4 Coogee Beach              4.64   2.65   5.58   5.49   4.03   2.39
 5 Gordons Bay (East)        4.8    2.60   5.58   5.39   4.09   2.39
 6 Little Bay Beach          4.64   2.60   5.58   5.39   4.09   2.39
 7 Malabar Beach             4.64   2.60   5.58   5.49   4.03   2.39
 8 Maroubra Beach            4.64   2.60   5.58   5.39   4.09   2.39
 9 South Maroubra Beach      4.64   2.60   5.58   5.39   4.09   2.39
10 South Maroubra Rockpool   3.60   2.60   5.58   5.39   4.09   2.39
11 Tamarama Beach            4.64   2.60   5.58   5.48   4.09   2.39
```

# pivoting from long to wide... and back

```
rain_data %>%
     pivot_wider(names_from = year, values_from = avg_rain) %>%
     pivot_longer(cols = c(2:7), names_to = "year")
```

```
# A tibble: 11 x 7
   site                    `2013` `2014` `2015` `2016` `2017` `2018`
   <chr>                    <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
 1 Bondi Beach               4.64   2.60   5.58   5.39   4.09   2.39
 2 Bronte Beach              4.64   2.60   5.58   5.39   4.09   2.39
 3 Clovelly Beach            4.64   2.60   5.58   5.39   4.09   2.39
 4 Coogee Beach              4.64   2.65   5.58   5.49   4.03   2.39
 5 Gordons Bay (East)        4.8    2.60   5.58   5.39   4.09   2.39
 6 Little Bay Beach          4.64   2.60   5.58   5.39   4.09   2.39
 7 Malabar Beach             4.64   2.60   5.58   5.49   4.03   2.39
 8 Maroubra Beach            4.64   2.60   5.58   5.39   4.09   2.39
 9 South Maroubra Beach      4.64   2.60   5.58   5.39   4.09   2.39
10 South Maroubra Rockpool   3.60   2.60   5.58   5.39   4.09   2.39
11 Tamarama Beach            4.64   2.60   5.58   5.48   4.09   2.39
```

# pivoting from long to wide... and back

```
rain_data %>%
     pivot_wider(names_from = year, values_from = avg_rain) %>%
     pivot_longer(cols = c(2:7), names_to = "year")
```

```
# A tibble: 66 x 3
   site         year  value
   <chr>        <chr> <dbl>
 1 Bondi Beach  2013   4.64
 2 Bondi Beach  2014   2.60
 3 Bondi Beach  2015   5.58
 4 Bondi Beach  2016   5.39
 5 Bondi Beach  2017   4.09
 6 Bondi Beach  2018   2.39
 7 Bronte Beach 2013   4.64
 8 Bronte Beach 2014   2.60
 9 Bronte Beach 2015   5.58
10 Bronte Beach 2016   5.39
# … with 56 more rows
```

we're back where we started

# combining datasets

```
left_join(data1, data2)
```

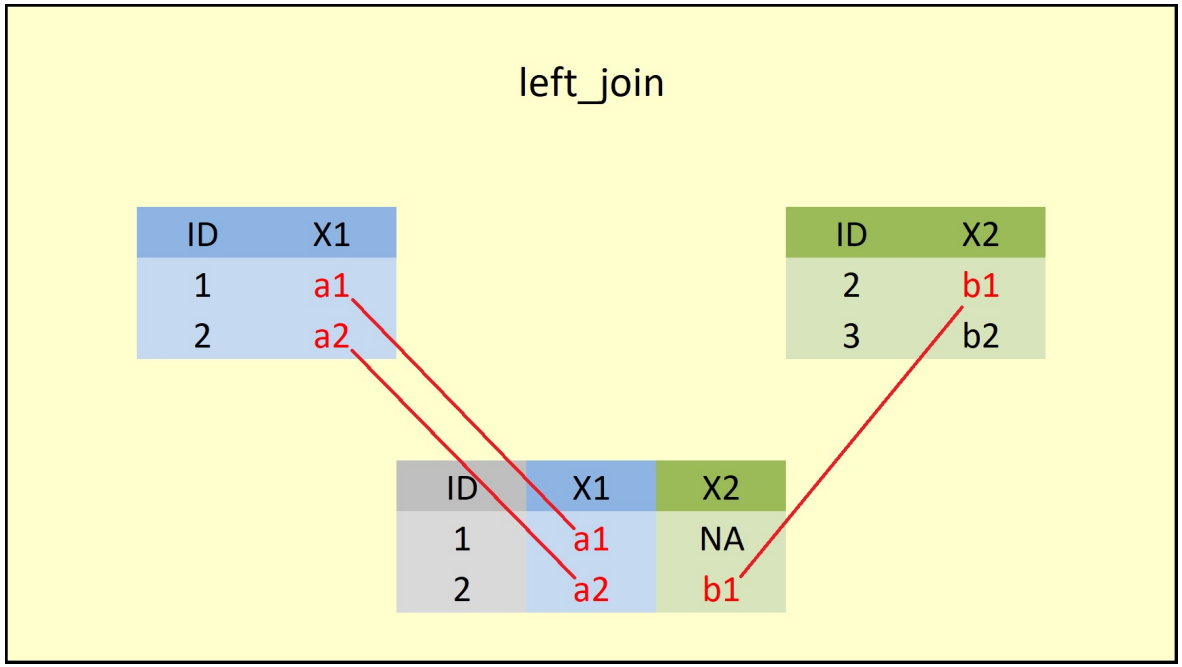



Image credit: Joachim Schork

# combining datasets

```
right_join(data1, data2)
```

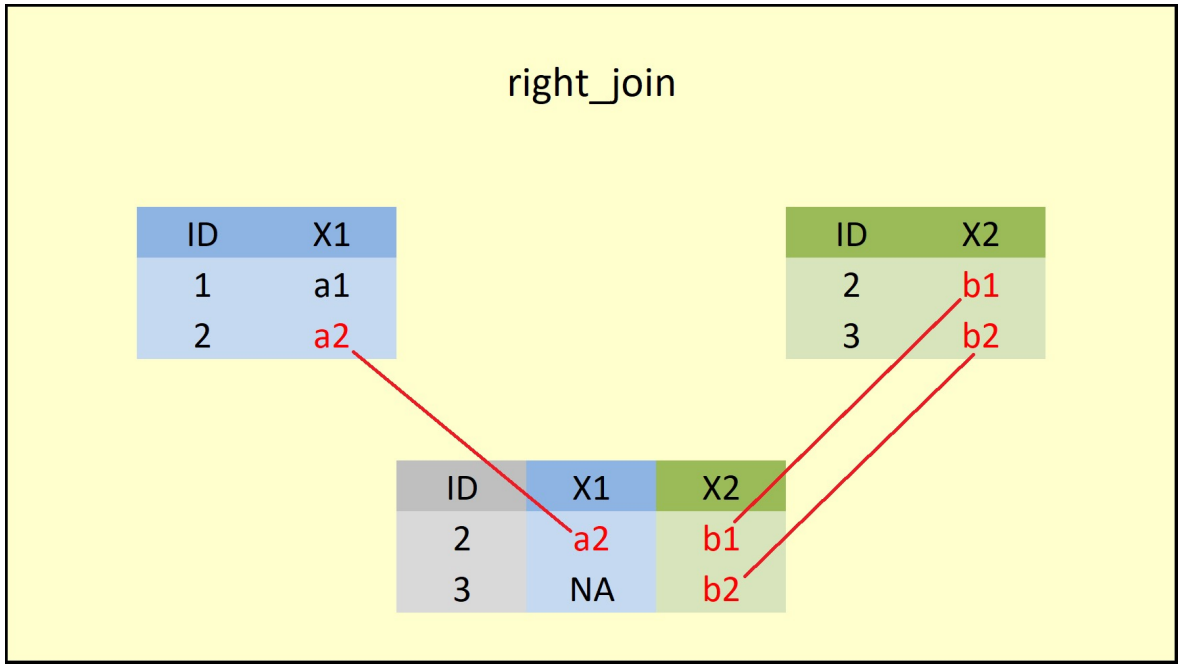



Image credit: Joachim Schork

# combining datasets

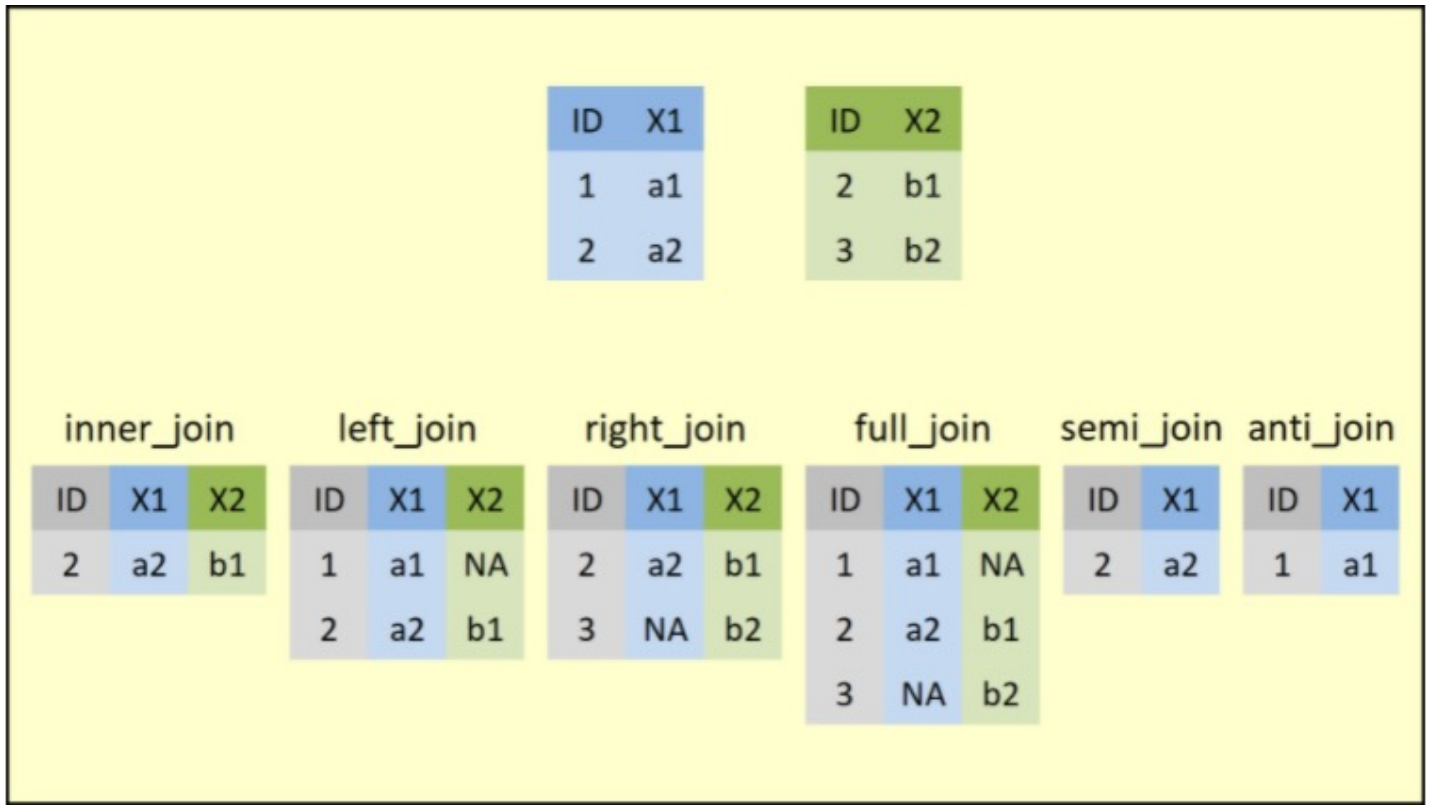



Image credit: Joachim Schork

# combining datasets

```
merge(data1, data2, by = “id”)
```

(both `data1` and `data2` contain `id`)

| | data1 | | data2 | |
|---|---|---|---|---|
| id | x1 | x2 | y1 | y2 |
| 1 | 5 | A | | |
| 2 | 1 | Y | | |
| 3 | 4 | G | | |
| 4 | 9 | F | 3 | a |
| 5 | 1 | G | 3 | x |
| 6 | 2 | Y | 4 | a |
| 7 | | | 1 | x |
| 8 | | | 2 | a |
| 9 | | | 9 | x |

Image credit: Joachim Schork

# combining datasets

**left/right/inner/etc. joins**

https://www.youtube.com/watch?v=Yg-pNqzDuN4

https://statisticsglobe.com/r-dplyr-join-inner-left-right-full-semi-anti

**merge**

https://statisticsglobe.com/r-merging-data-frames-by-column-names-merge-function

**rbind / cbind**