ggplot2

Etienne Low-Décarie

September 12, 2015

Waiting around?

Let R impress you:

```
demo(package = .packages(all.available = TRUE))
```

- you will get a text file with all available demos

```
library(googleVis)
demo(smooth)
```

Waiting around?

Let R impress you:

```
demo(package = .packages(all.available = TRUE))
```

- you will get a text file with all available demos
- choose one of interest

```
library(googleVis)
demo(smooth)
```

Waiting around?

Let R impress you:

```
demo(package = .packages(all.available = TRUE))
```

- you will get a text file with all available demos
- choose one of interest
- if needed load its package

```
library(googleVis)
demo(smooth)
```

Waiting around?

Let R impress you:

```
demo(package = .packages(all.available = TRUE))
```

- you will get a text file with all available demos
- choose one of interest
- if needed load its package
- type it in the demo brackets

```
library(googleVis)
demo(smooth)
```

Waiting around?

Let R impress you:

```
demo(package = .packages(all.available = TRUE))
```

- you will get a text file with all available demos
- choose one of interest
- if needed load its package
- type it in the demo brackets
- example

```
library(googleVis)
demo(smooth)
```

Waiting around?

Let R impress you:

```
demo(package = .packages(all.available = TRUE))
```

- you will get a text file with all available demos
- choose one of interest
- if needed load its package
- type it in the demo brackets
- example

```
library(googleVis)
demo(smooth)
```

- notice it shows you the code that creates the demo!

Waiting around?

Let R impress you:

```
demo(package = .packages(all.available = TRUE))
```

- you will get a text file with all available demos
- choose one of interest
- if needed load its package
- type it in the demo brackets
- example

```
library(googleVis)
demo(smooth)
```

- notice it shows you the code that creates the demo!
- look at packages on `http://crantastic.org`

Tell me about you

- Have you created a plot?

Tell me about you

- Have you created a plot?
- With what data?

Tell me about you

- Have you created a plot?
- With what data?
- What kind of plot?

Tell me about you

- Have you created a plot?
- With what data?
- What kind of plot?
- Have plotted with R?

Tell me about you

- Have you created a plot?
- With what data?
- What kind of plot?
- Have plotted with R?
- used ggplot?

Tell me about you

- Have you created a plot?
- With what data?
- What kind of plot?
- Have plotted with R?
- used ggplot?
- How many variables do you usually need to put into a plot?

Tell me about you

- Have you created a plot?
- With what data?
- What kind of plot?
- Have plotted with R?
- used ggplot?
- How many variables do you usually need to put into a plot?
- Do you need to produce maps?

Tell me about you

- Have you created a plot?
- With what data?
- What kind of plot?
- Have plotted with R?
- used ggplot?
- How many variables do you usually need to put into a plot?
- Do you need to produce maps?
- Have you used stackoverflow?

Tell me about you

- Have you created a plot?
- With what data?
- What kind of plot?
- Have plotted with R?
- used ggplot?
- How many variables do you usually need to put into a plot?
- Do you need to produce maps?
- Have you used stackoverflow?
- Are you in buddy pairs?

Tell me about you

- Have you created a plot?
- With what data?
- What kind of plot?
- Have plotted with R?
- used ggplot?
- How many variables do you usually need to put into a plot?
- Do you need to produce maps?
- Have you used stackoverflow?
- Are you in buddy pairs?
- Who uses Rstudio as their IDE?

A few pet peeves

- Always work from a script

```
object <- data.frame(argument1="value1",
                     argument2="value2",
                     argument3="value3")
```

A few pet peeves

- Always work from a script
- Use carriage returns and indentation

```
object <- data.frame(argument1="value1",
                     argument2="value2",
                     argument3="value3")
```

A few pet peeves

- Always work from a script
- Use carriage returns and indentation

```
object <- data.frame(argument1="value1",
                     argument2="value2",
                     argument3="value3")
```

- Create your own new script

A few pet peeves

- Always work from a script
- Use carriage returns and indentation

```
object <- data.frame(argument1="value1",
                     argument2="value2",
                     argument3="value3")
```

- Create your own new script
- refer to provided code only if needed

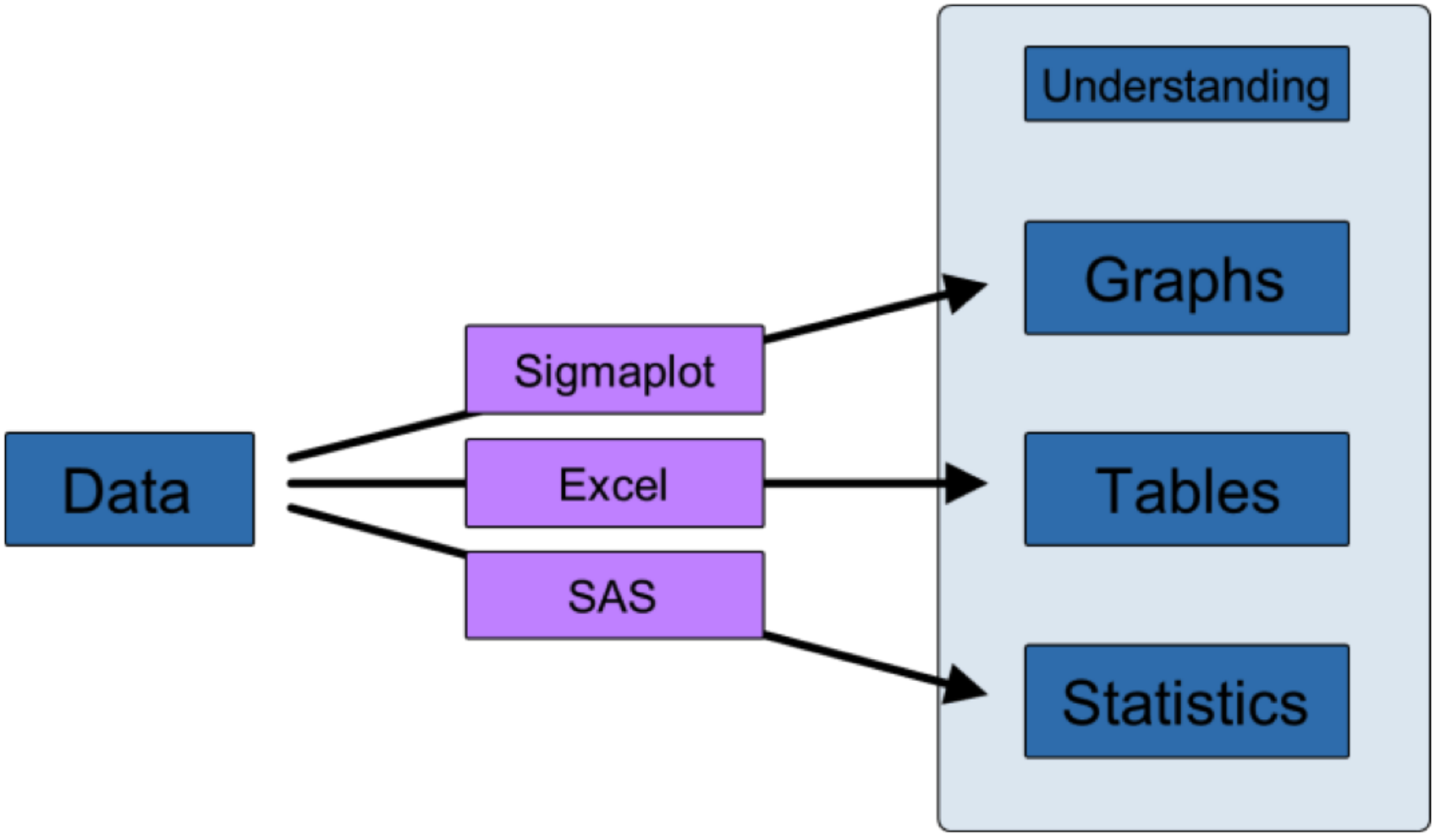
A few pet peeves

- Always work from a script
- Use carriage returns and indentation

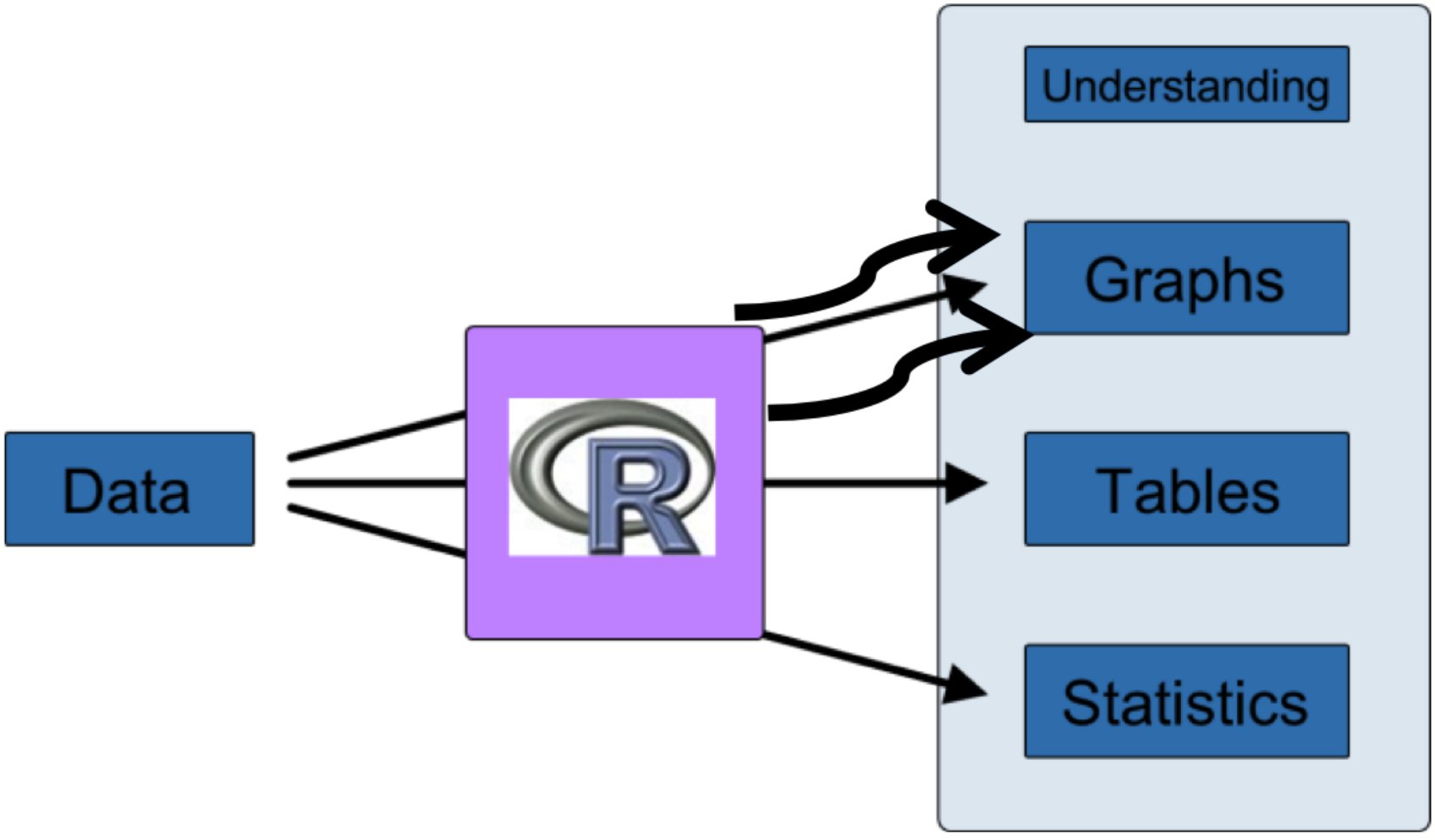
```
object <- data.frame(argument1="value1",
                     argument2="value2",
                     argument3="value3")
```

- Create your own new script
- refer to provided code only if needed
- don't use the one provided

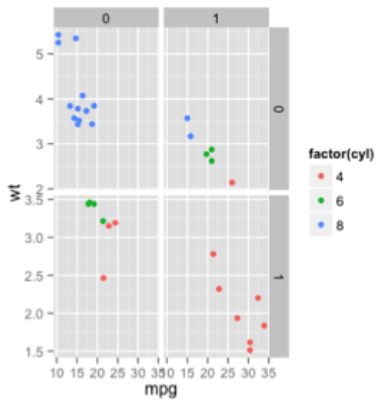
Without R

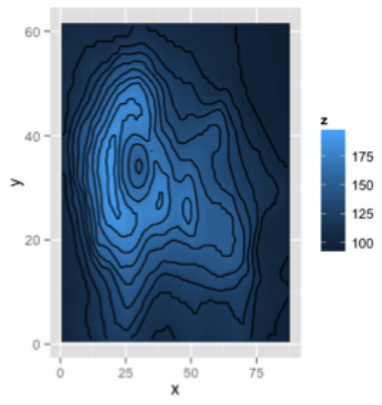

Data
Sigmaplot
Excel
SAS
Understanding
Graphs
Tables
Statistics

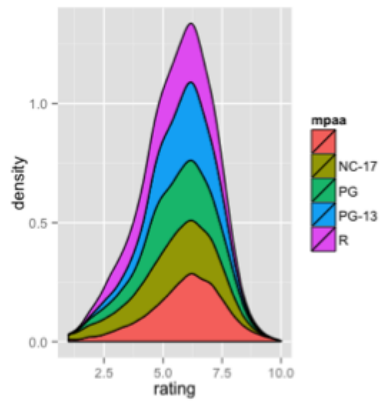
With R

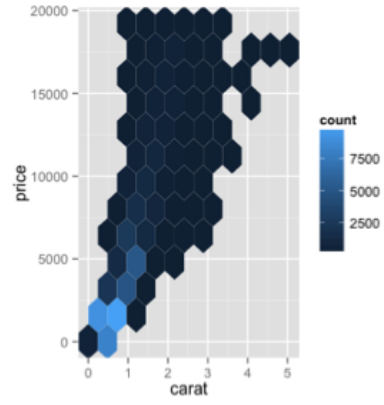


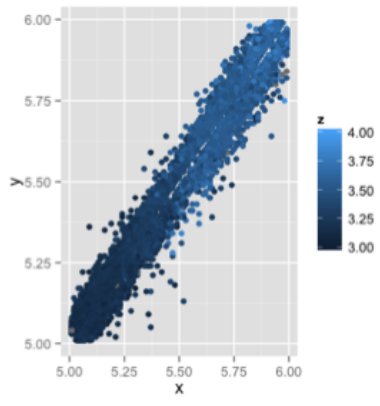
Beautiful and flexible!

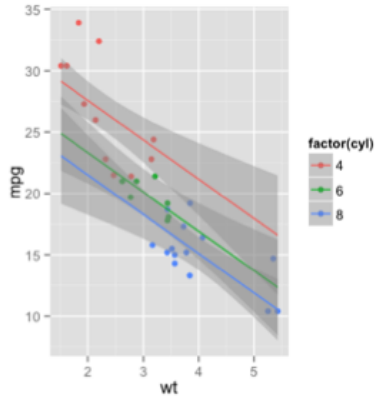


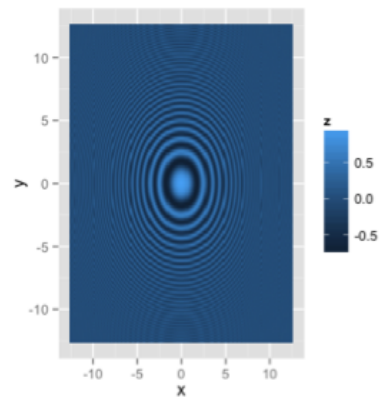


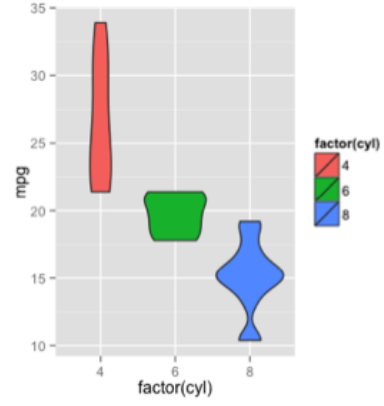












Install/load ggplot2

```
if(!require(ggplot2)){install.packages("ggplot2")}
```

```
## Loading required package: ggplot2
```

```
require(ggplot2)
```

Outline

Outline

1. Your first ggplot plot
    - basic scatter plot

Outline

1. Your first ggplot plot
    - basic scatter plot
    - Exercise 1

Outline

1. Your first ggplot plot
   - basic scatter plot
   - Exercise 1
2. Grammar of graphics

Outline

1. Your first ggplot plot
   - basic scatter plot
   - Exercise 1
2. Grammar of graphics
   - More advanced plots

Outline

1. Your first ggplot plot
   - basic scatter plot
   - Exercise 1
2. Grammar of graphics
   - More advanced plots
   - Available plot elements and when to use them

Outline

1. Your first ggplot plot
   - basic scatter plot
   - Exercise 1
2. Grammar of graphics
   - More advanced plots
   - Available plot elements and when to use them
   - Exercise 2

Outline

1. Your first ggplot plot
   - basic scatter plot
   - Exercise 1
2. Grammar of graphics
   - More advanced plots
   - Available plot elements and when to use them
   - Exercise 2
   - Saving a plot

Outline

1. Your first ggplot plot
   - basic scatter plot
   - Exercise 1
2. Grammar of graphics
   - More advanced plots
   - Available plot elements and when to use them
   - Exercise 2
   - Saving a plot
3. Fine tuning your plot

Outline

1. Your first ggplot plot
   - basic scatter plot
   - Exercise 1
2. Grammar of graphics
   - More advanced plots
   - Available plot elements and when to use them
   - Exercise 2
   - Saving a plot
3. Fine tuning your plot
   - colours

Outline

1. Your first ggplot plot
   - basic scatter plot
   - Exercise 1
2. Grammar of graphics
   - More advanced plots
   - Available plot elements and when to use them
   - Exercise 2
   - Saving a plot
3. Fine tuning your plot
   - colours
   - themes

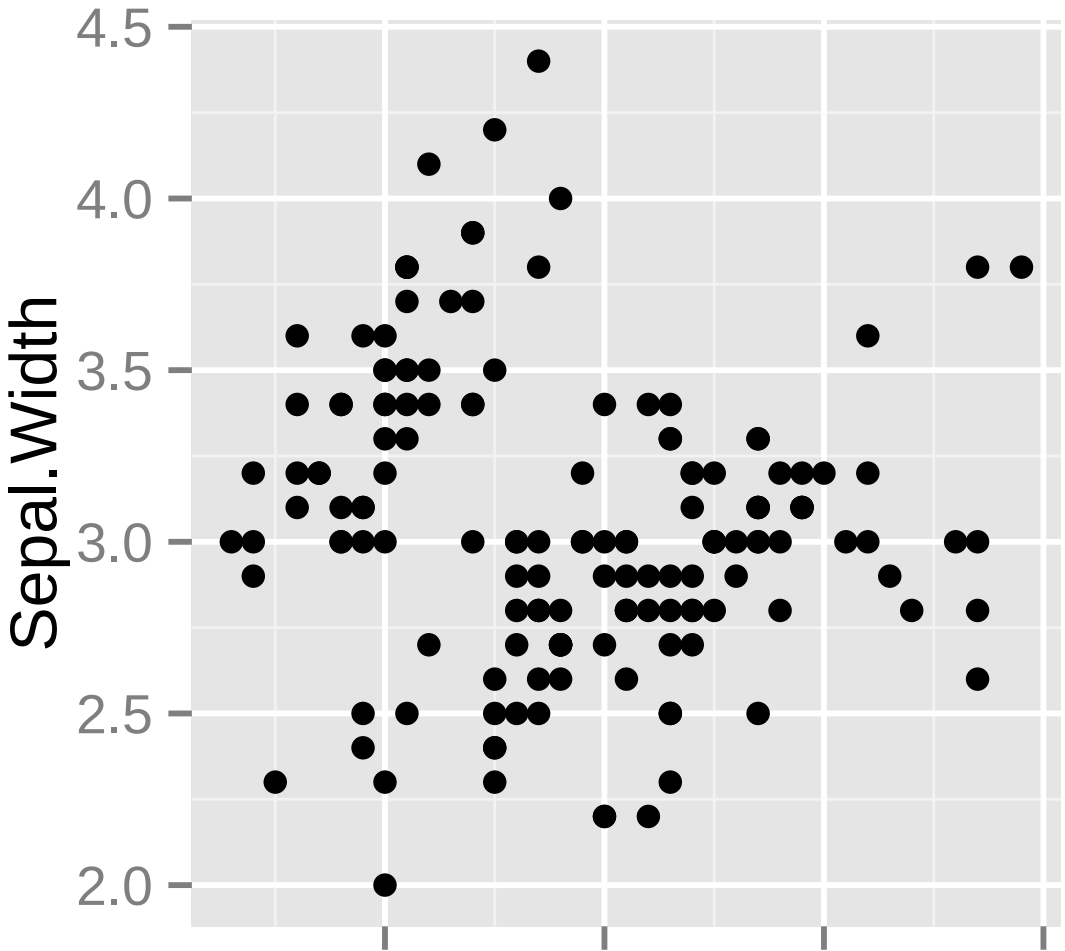
Outline

1. Your first ggplot plot
   - basic scatter plot
   - Exercise 1
2. Grammar of graphics
   - More advanced plots
   - Available plot elements and when to use them
   - Exercise 2
   - Saving a plot
3. Fine tuning your plot
   - colours
   - themes
4. Miscellaneous cool stuff

Your first ggplot

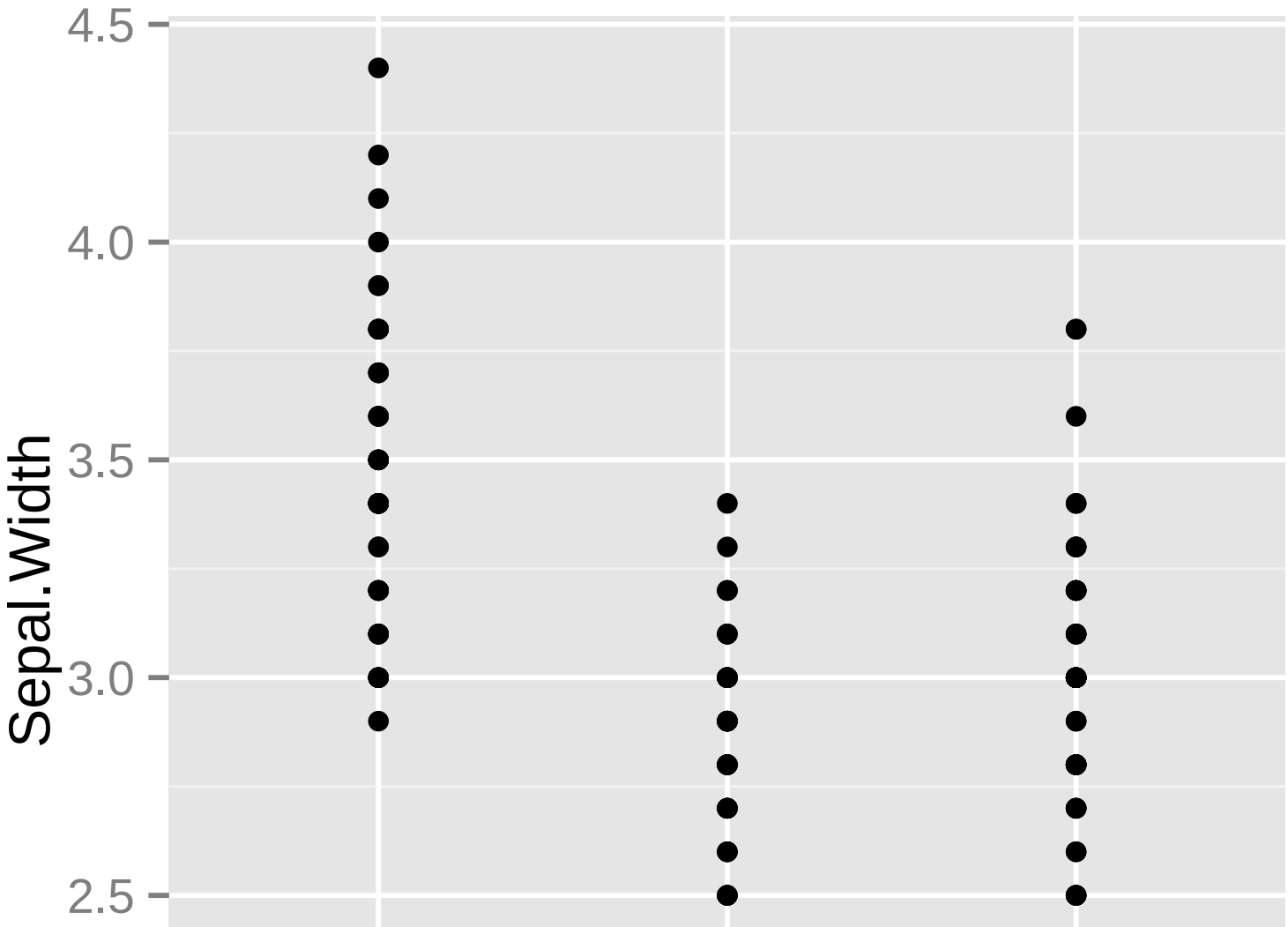
A basic scatter plot

```
qplot(data=iris,
      x=Sepal.Length,
      y=Sepal.Width)
```



Categorical x-axis

```
qplot(data=iris,
      x=Species,
      y=Sepal.Width)
```



Less basic scatter plot

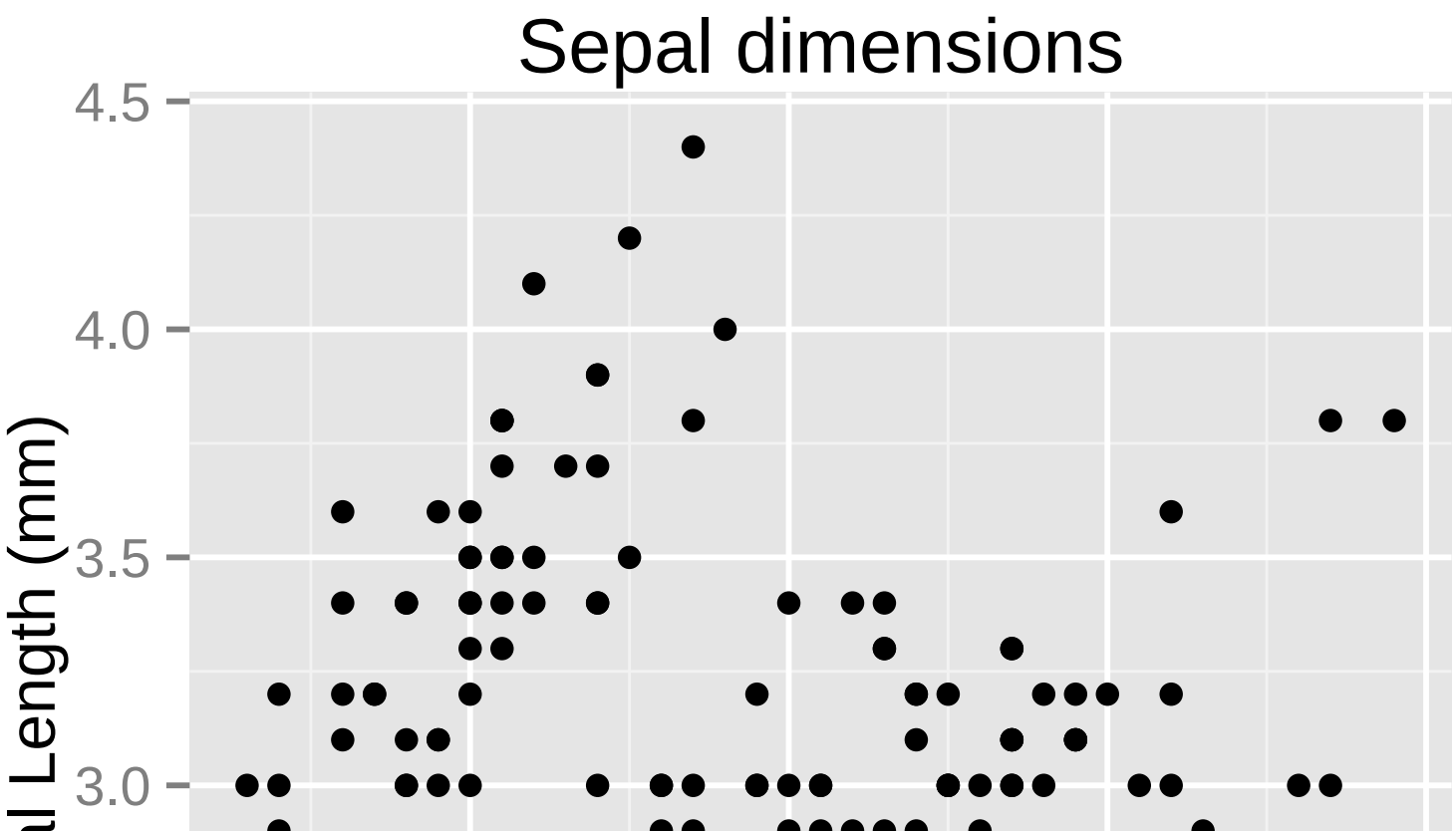
```
?qplot
```

Arguments

`x`
`y`
`...`
`data`
`xlab`
`ylab`
`main`

Less basic scatter plot

```
qplot(data=iris,
      x=Sepal.Length,
      xlab="Sepal Width (mm)",
      y=Sepal.Width,
      ylab="Sepal Length (mm)",
      main="Sepal dimensions")
```



Exercise 1

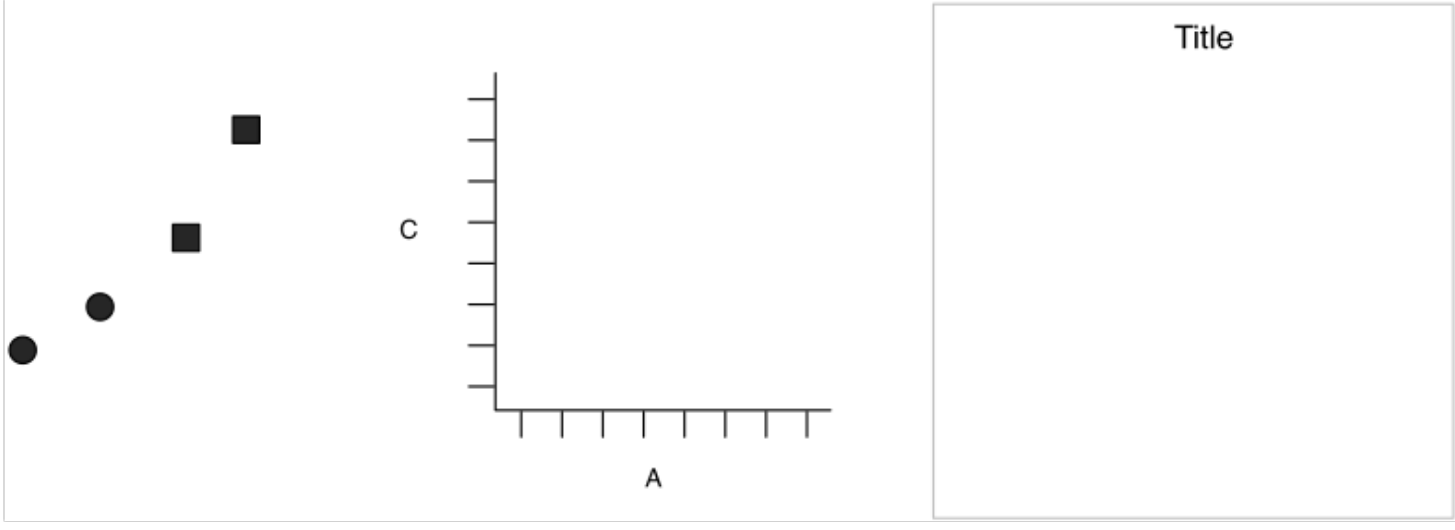
produce a basic plot with built in data

```
CO2
?CO2
BOD
data()
```

Grammar of graphics (gg)

A graphic is made of elements (layers)

- data

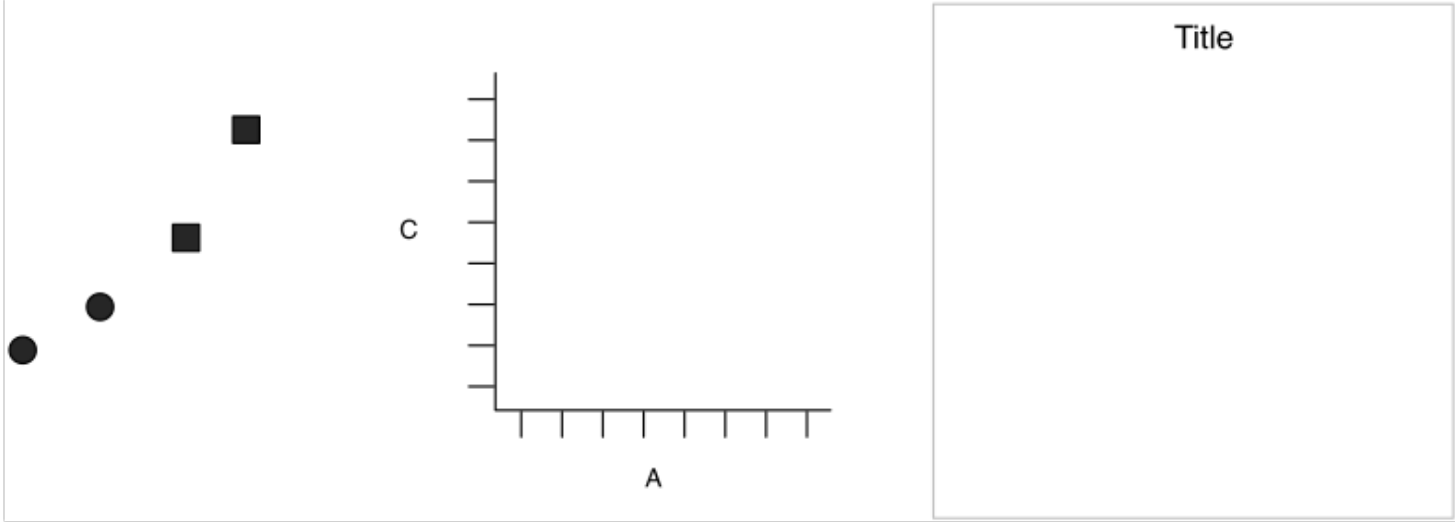


Aesthetics (aes) make data visible:

Grammar of graphics (gg)

A graphic is made of elements (layers)

- data
- aesthetics (aes)

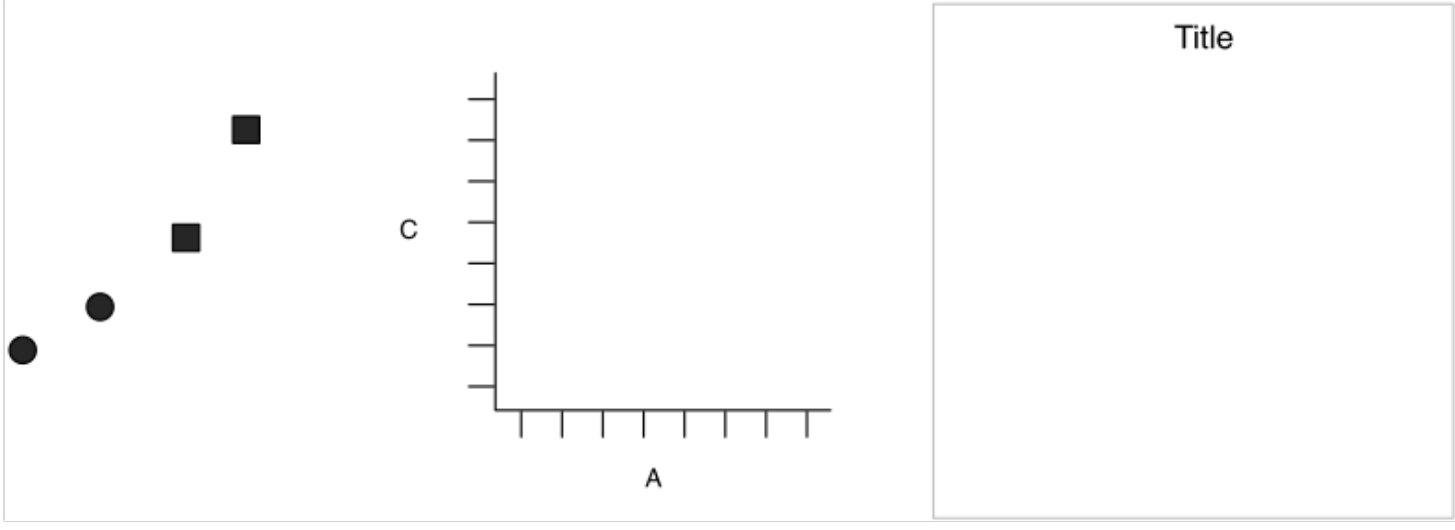


Aesthetics (aes) make data visible:

Grammar of graphics (gg)

A graphic is made of elements (layers)

- data
- aesthetics (aes)
- transformation

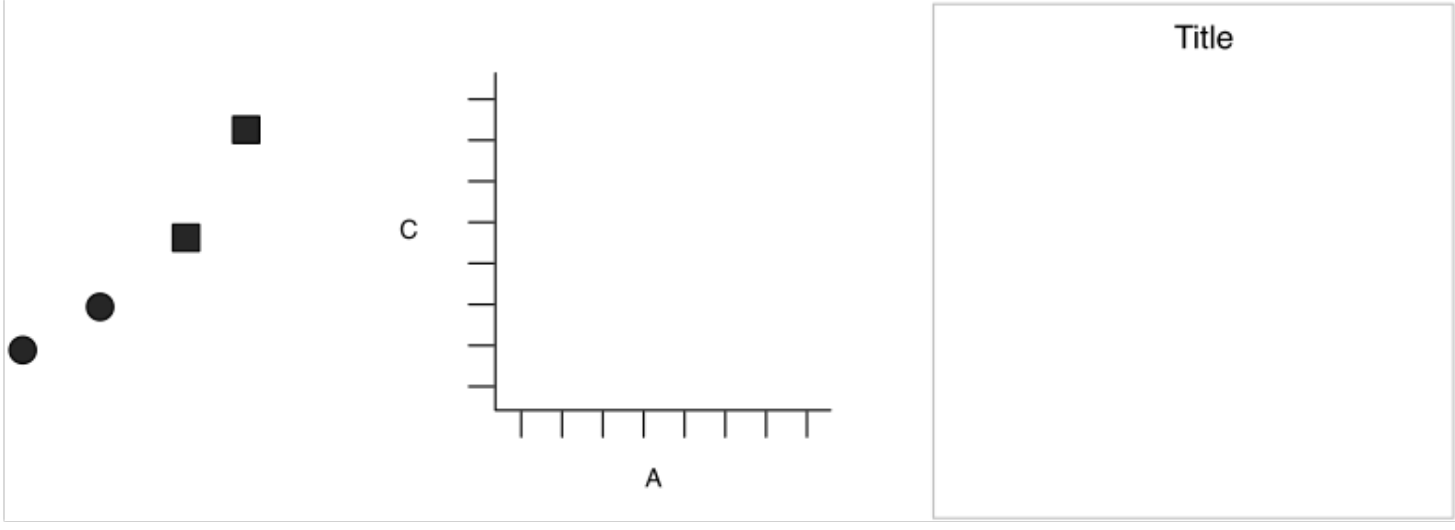


Aesthetics (aes) make data visible:

Grammar of graphics (gg)

A graphic is made of elements (layers)

- data
- aesthetics (aes)
- transformation
- geoms (geometric objects)

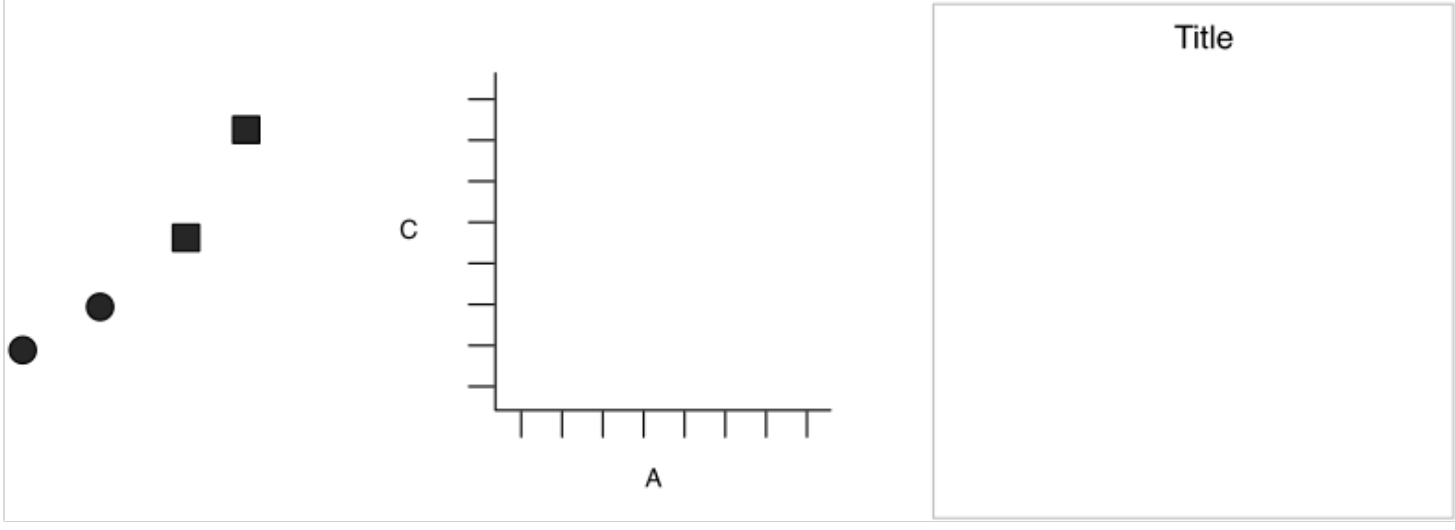


Aesthetics (aes) make data visible:

Grammar of graphics (gg)

A graphic is made of elements (layers)

- data
- aesthetics (aes)
- transformation
- geoms (geometric objects)
- axis (coordinate system)

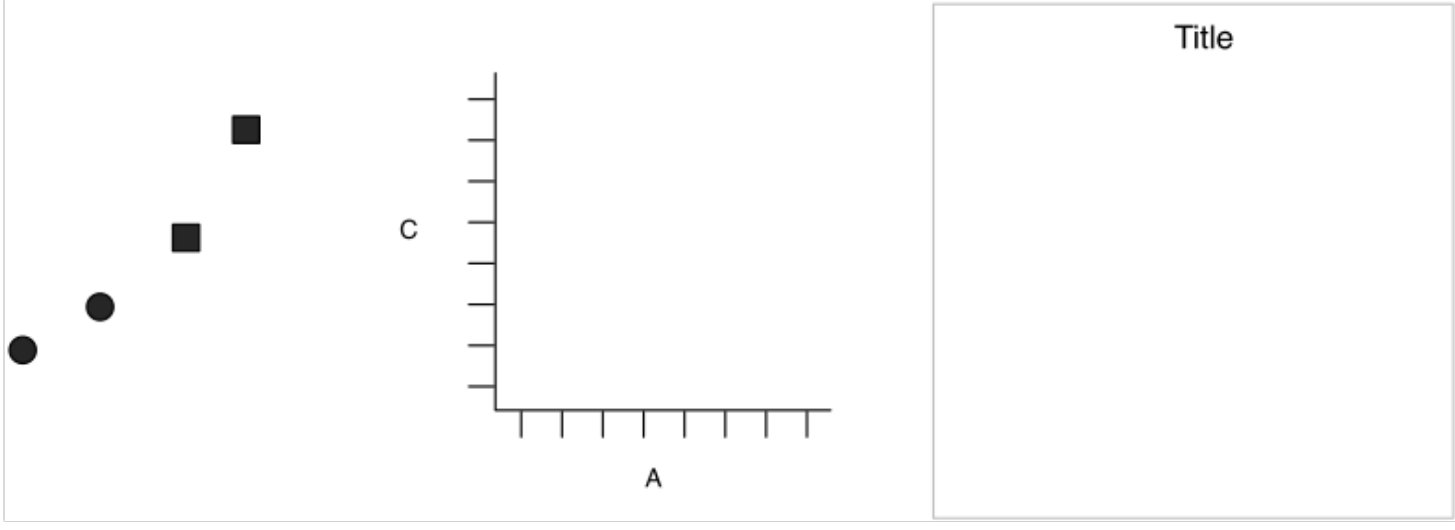


Aesthetics (aes) make data visible:

Grammar of graphics (gg)

A graphic is made of elements (layers)

- data
- aesthetics (aes)
- transformation
- geoms (geometric objects)
- axis (coordinate system)
- scales

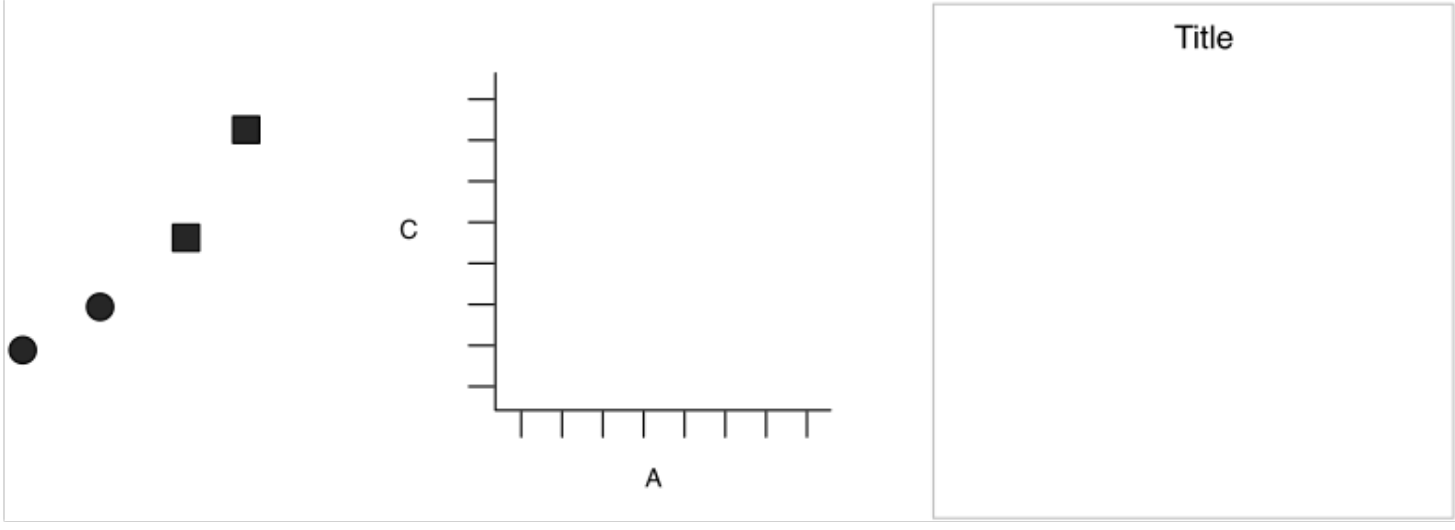


Aesthetics (aes) make data visible:

Grammar of graphics (gg)

A graphic is made of elements (layers)

- data
- aesthetics (aes)
- transformation
- geoms (geometric objects)
- axis (coordinate system)
- scales

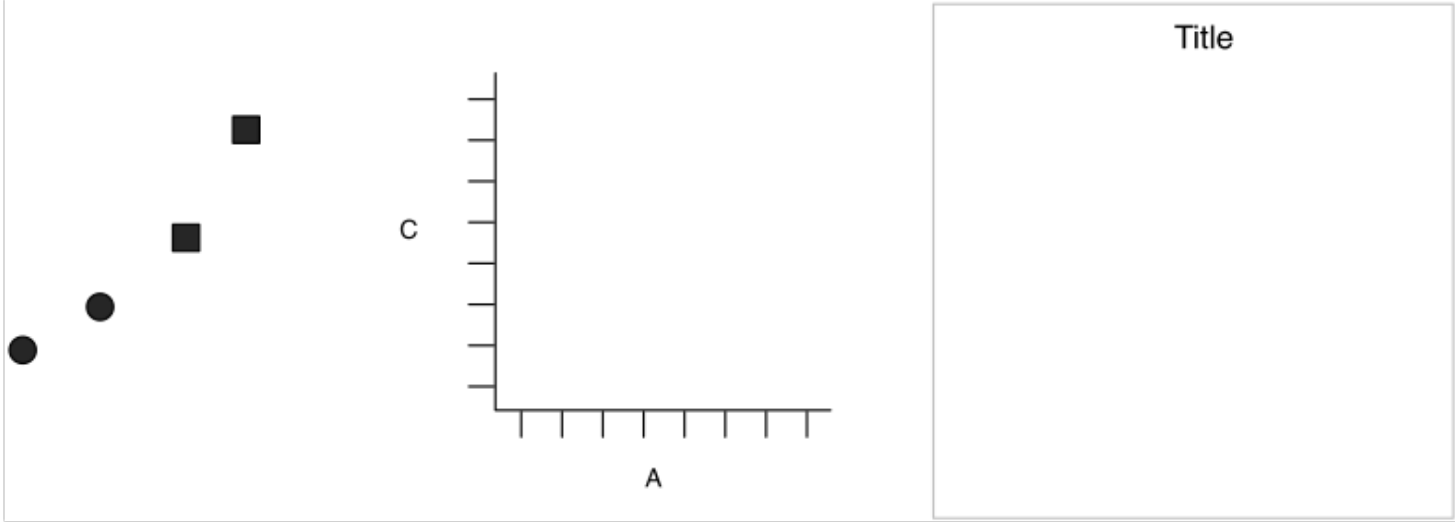


Aesthetics (aes) make data visible:

Grammar of graphics (gg)

A graphic is made of elements (layers)

- data
- aesthetics (aes)
- transformation
- geoms (geometric objects)
- axis (coordinate system)
- scales

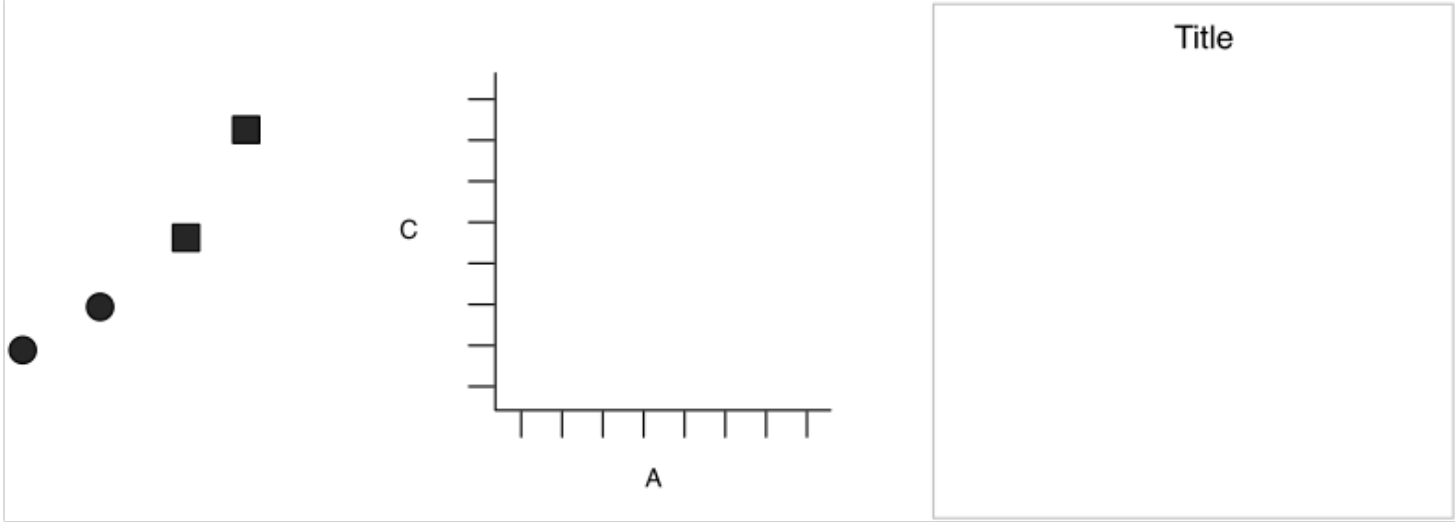


Aesthetics (aes) make data visible:

Grammar of graphics (gg)

A graphic is made of elements (layers)

- data
- aesthetics (aes)
- transformation
- geoms (geometric objects)
- axis (coordinate system)
- scales

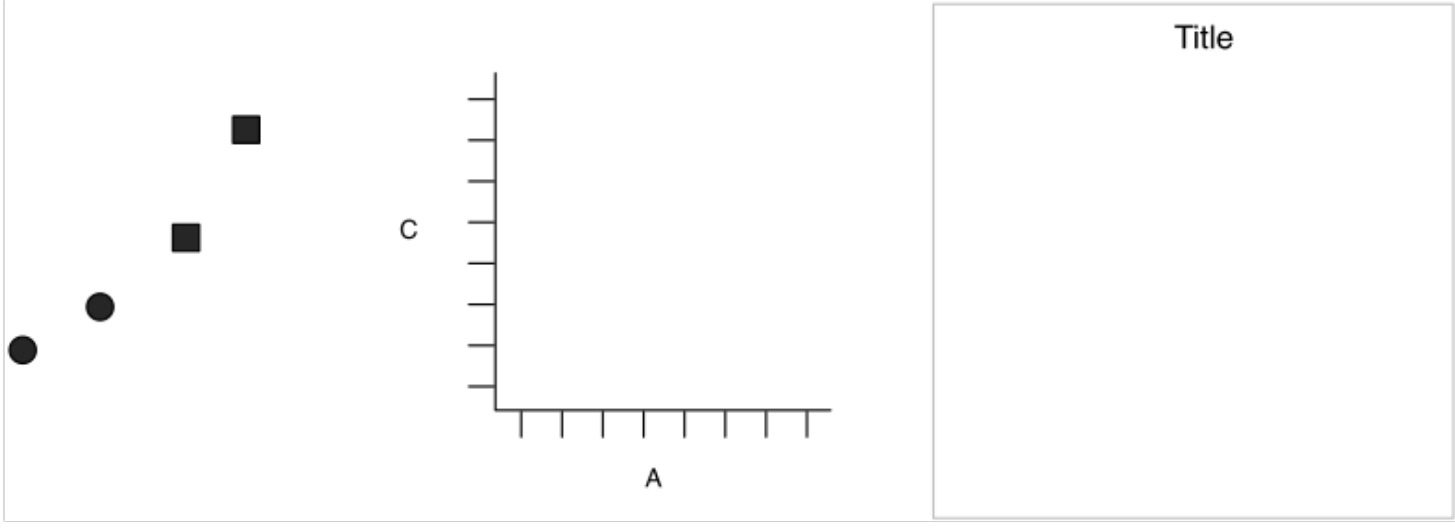


Aesthetics (aes) make data visible:

Grammar of graphics (gg)

A graphic is made of elements (layers)

- data
- aesthetics (aes)
- transformation
- geoms (geometric objects)
- axis (coordinate system)
- scales

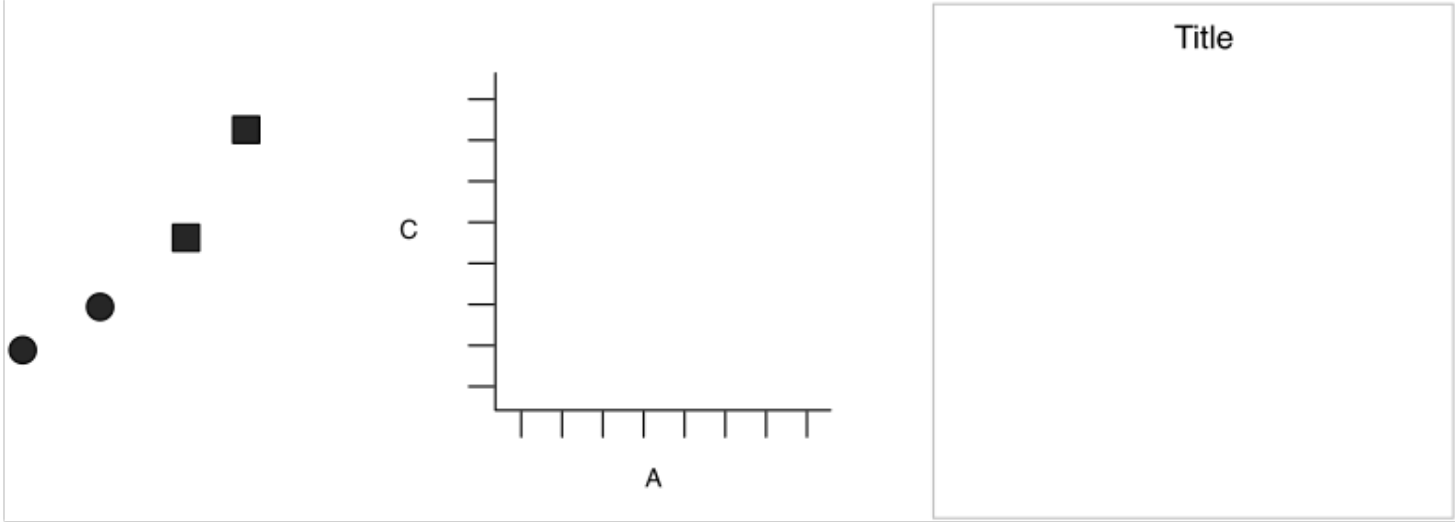


Aesthetics (aes) make data visible:

Grammar of graphics (gg)

A graphic is made of elements (layers)

- data
- aesthetics (aes)
- transformation
- geoms (geometric objects)
- axis (coordinate system)
- scales

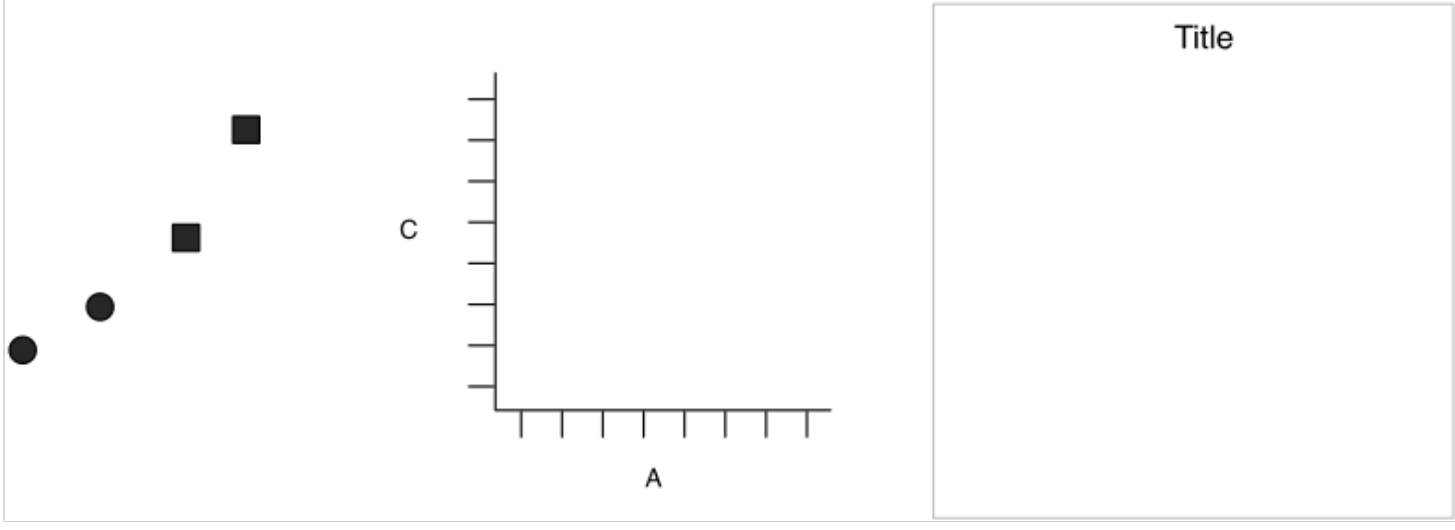


Aesthetics (aes) make data visible:

Grammar of graphics (gg)

A graphic is made of elements (layers)

- data
- aesthetics (aes)
- transformation
- geoms (geometric objects)
- axis (coordinate system)
- scales

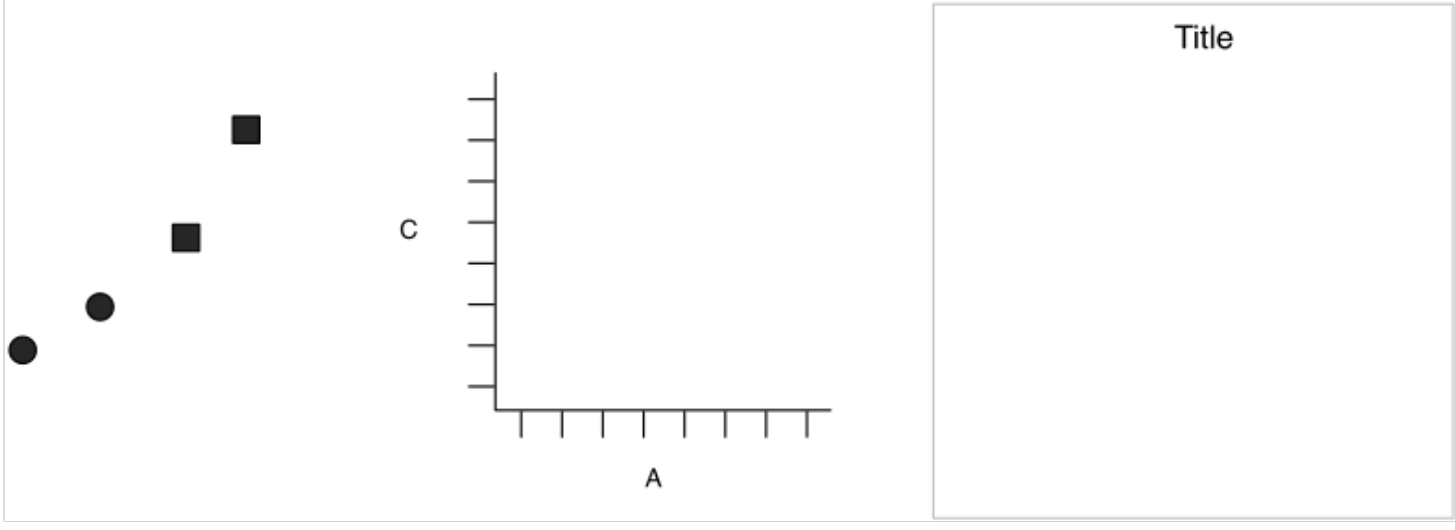


Aesthetics (aes) make data visible:

Grammar of graphics (gg)

A graphic is made of elements (layers)

- data
- aesthetics (aes)
- transformation
- geoms (geometric objects)
- axis (coordinate system)
- scales



Aesthetics (aes) make data visible:

geometric objects(geoms)

- point: scatterplot

geometric objects(geoms)

- point: scatterplot
- line: line plot, where lines connect points by increasing x value

geometric objects(geoms)

- point: scatterplot
- line: line plot, where lines connect points by increasing x value
- path: line plot, where lines connect points in sequence of appearance

geometric objects(geoms)

- point: scatterplot
- line: line plot, where lines connect points by increasing x value
- path: line plot, where lines connect points in sequence of appearance
- boxplot: box-and-whisker plots, for categorical y data

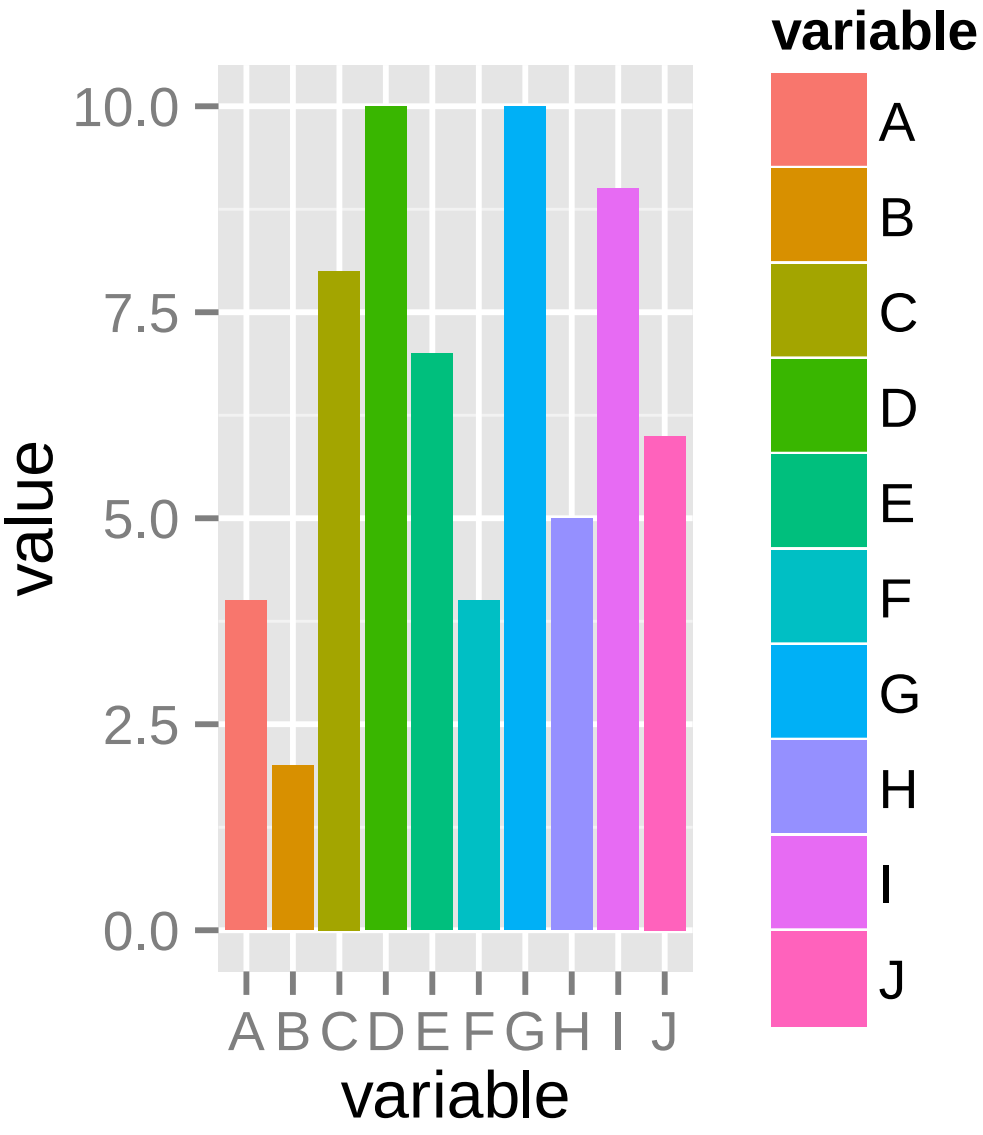
geometric objects(geoms)

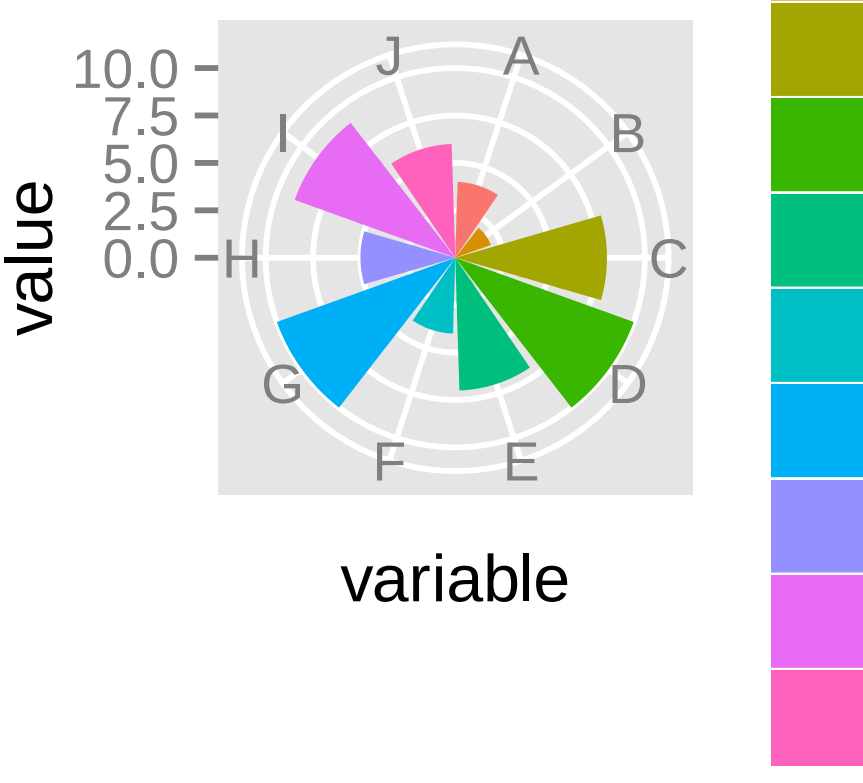
- point: scatterplot
- line: line plot, where lines connect points by increasing x value
- path: line plot, where lines connect points in sequence of appearance
- boxplot: box-and-whisker plots, for categorical y data
- bar: barplots

geometric objects(geoms)

- point: scatterplot
- line: line plot, where lines connect points by increasing x value
- path: line plot, where lines connect points in sequence of appearance
- boxplot: box-and-whisker plots, for categorical y data
- bar: barplots
- histogram: histograms (for 1-dimensional data)

Editing an element produces a new graph e.g. just change the coordinate system





How it works

1 create a simple plot object

```
plot.object<-qplot()
```

2 add graphical layers/complexity

```
plot.object<-plot.object+layer()
```

3 repeat step 2 until satisfied
4 print your object to screen (or to graphical device)

```
print(plot.object)
```

Resources

cheatsheets:
`https://www.rstudio.com/resources/cheatsheets/`