

3. Visualize

R³ Training

noaa-iaa.github.io/r3-train

Ben Best
2021-06-28

Building a `ggplot`: add layers & customization

```
# define aesthetics
ggplot(d, aes(x = time, y = revenue)) +

  # add geometry
  geom_line() +

  # add smooth
  geom_smooth() +

  # change labels
  labs(
    title = "Fisheries Revenue",
    y      = "Millions $ (year 2015)")
```

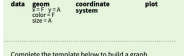
Cheatsheet for reference

Data Visualization with ggplot2 Cheatsheet (RStudio menu: Help > Cheat Sheets)

Data Visualization with ggplot2 : : CHEAT SHEET

Basics

ggplot2 is based on the **grammar of graphics**, the idea that you can build every graph from the same components: **a data set**, **a coordinate system**, and **geoms**—visual marks that represent data points.



Complete the template below to build a graph.

ggplot (data = **DATA**) +
 geom_function (mapping = aes(**AESTHETICS**))
 coord_cartesian (ylim = **COORDINATES**)
 facet_grid (**ROW** ~ **COLUMN**)
 theme (**THEME**)

ggplot2(data = **my_data**, aes = cty ~ hwy)) creates a plot that you finish by adding layers to. Add one geom function per layer.

geom_point (x = **var**, y = **hwy**) creates a scatter plot with **geom** data, **geom**, and **mapping**. Suggests many useful defaults.

last_plot() returns the last plot.

ggsave("file.png", width = 5, height = 5) Saves last plot as 5" x 5" file named "file.png" in working directory. Watch file type for file extension.

Geoms

Use a geom function to represent data points, use the geom's aesthetic properties to represent variables. Each function returns a layer.

geom_blank()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(x = long, y = lat))
c = ggplot(economics, aes(x = long, y = lat))
d = ggplot(economics, aes(x = long, y = lat))

geom_rect()
a = ggplot(economics, aes(x = long, y = lat))
b = ggplot(economics, aes(x = long, y = lat))
c = ggplot(economics, aes(x = long, y = lat))
d = ggplot(economics, aes(x = long, y = lat))

geom_line()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

geom_point()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

geom_smooth()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

geom_histogram()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

geom_density()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

Two Variables

geom_point()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

geom_line()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

geom_rect()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

geom_smooth()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

geom_histogram()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

geom_density()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

geom_area()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

geom_bar()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

Continuous Bivariate Distribution

geom_density_2d()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

geom_density_2d_r()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

geom_density_2d_r()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

geom_density_2d_r()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

geom_density_2d_r()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

geom_density_2d_r()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

geom_density_2d_r()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

geom_density_2d_r()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))



Stats

An alternative way to build a layer



Visualize a stat by changing the default stat of a geom function, **geom_bar(stat = "count")** or by using a stat function, **stat_count(geom = "bar")**, which calls a default geom to make a layer (equivalent to a geom function). Use **name** syntax to map stat variables to aesthetics.

stat_count()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

stat_density()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

stat_smooth()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

stat_summary()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

stat_summary()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

stat_summary()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

stat_summary()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

Scales

Scales map data values to the visual values of an aesthetic. To change a mapping, add a new scale.



Visualize a stat by changing the default stat of a geom function, **geom_bar(stat = "count")** or by using a stat function, **stat_count(geom = "bar")**, which calls a default geom to make a layer (equivalent to a geom function). Use **name** syntax to map stat variables to aesthetics.

scale_color_manual()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

scale_color_manual()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

scale_color_manual()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

scale_color_manual()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

scale_color_manual()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

scale_color_manual()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

scale_color_manual()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

Coordinate Systems

Coordinate systems map data values to the visual values of an aesthetic. To change a mapping, add a new scale.



Visualize a stat by changing the default stat of a geom function, **geom_bar(stat = "count")** or by using a stat function, **stat_count(geom = "bar")**, which calls a default geom to make a layer (equivalent to a geom function). Use **name** syntax to map stat variables to aesthetics.

coord_cartesian()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

coord_cartesian()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

coord_cartesian()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

coord_cartesian()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

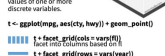
coord_cartesian()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

coord_cartesian()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

coord_cartesian()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

Faceting

Facets divide a plot into subplots based on the values of one or more discrete variables.



Visualize a stat by changing the default stat of a geom function, **geom_bar(stat = "count")** or by using a stat function, **stat_count(geom = "bar")**, which calls a default geom to make a layer (equivalent to a geom function). Use **name** syntax to map stat variables to aesthetics.

facet_grid()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

facet_grid()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

facet_grid()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

facet_grid()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

facet_grid()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

facet_grid()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))

facet_grid()
a = ggplot(economics, aes(date, unemp))
b = ggplot(economics, aes(date, unemp))
c = ggplot(economics, aes(date, unemp))
d = ggplot(economics, aes(date, unemp))



Lesson

noaa-iea.github.io/r3-train/visualize.html