Plotly

INFO 201

Today’s Objectives

Discuss strategies for building new skills

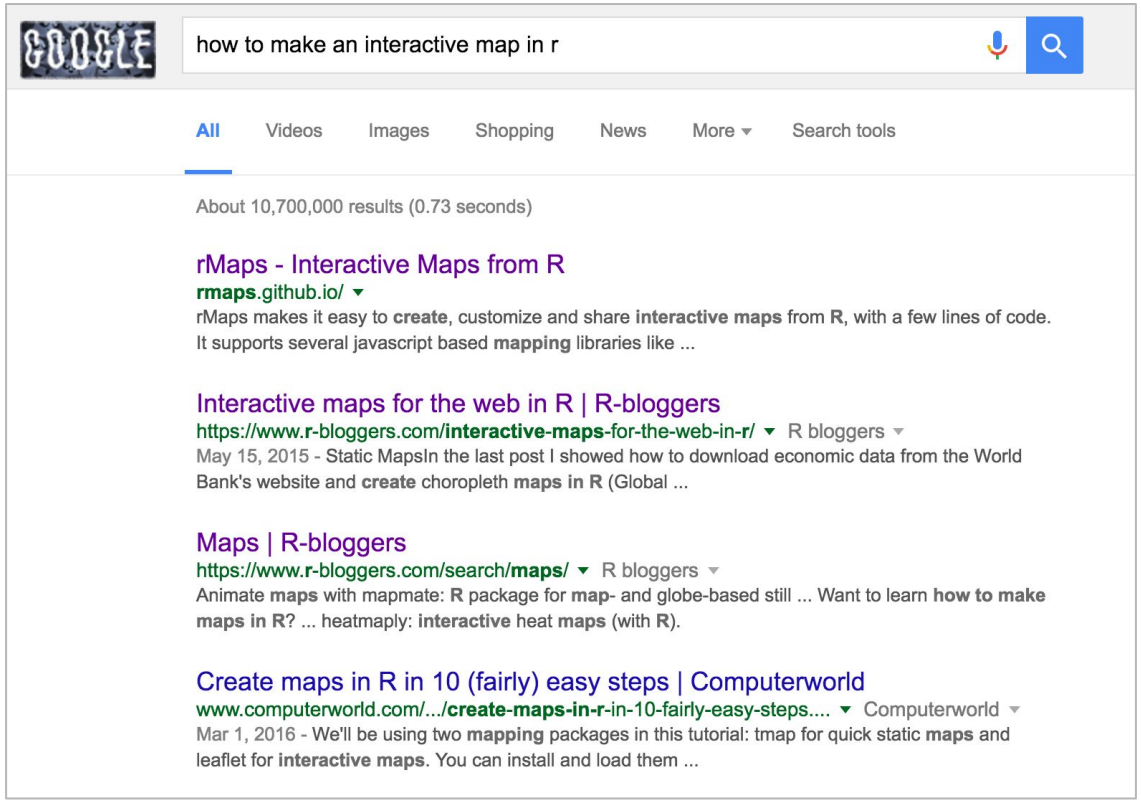
Build interactive graphics with the Plotly API

Practice creating websites (with interactive graphics)

Introduce the tidyr package

Building new skills

If you wanted to learn how to make a map in R, how would you do it?


GOOGLE
how to make an interactive map in r
All
Videos
Images
Shopping
News
More
Search tools
About 10,700,000 results (0.73 seconds)
rMaps - Interactive Maps from R
rmaps.github.io/
rMaps makes it easy to create, customize and share interactive maps from R, with a few lines of code. It supports several javascript based mapping libraries like ...
Interactive maps for the web in R | R-bloggers
https://www.r-bloggers.com/interactive-maps-for-the-web-in-r/ R bloggers
May 15, 2015 - Static MapsIn the last post I showed how to download economic data from the World Bank's website and create choropleth maps in R (Global ...
Maps | R-bloggers
https://www.r-bloggers.com/search/maps/ R bloggers
Animate maps with mapmate: R package for map- and globe-based still ... Want to learn how to make maps in R? ... heatmaply: interactive heat maps (with R).
Create maps in R in 10 (fairly) easy steps | Computerworld
www.computerworld.com/.../create-maps-in-r-in-10-fairly-easy-steps.... Computerworld
Mar 1, 2016 - We'll be using two mapping packages in this tutorial: tmap for quick static maps and leaflet for interactive maps. You can install and load them ...

Well...

Top-Down Approach

Start with Documentation

- marker
  - opacity (`number between or equal to 0 and 1`)

    default: `1`

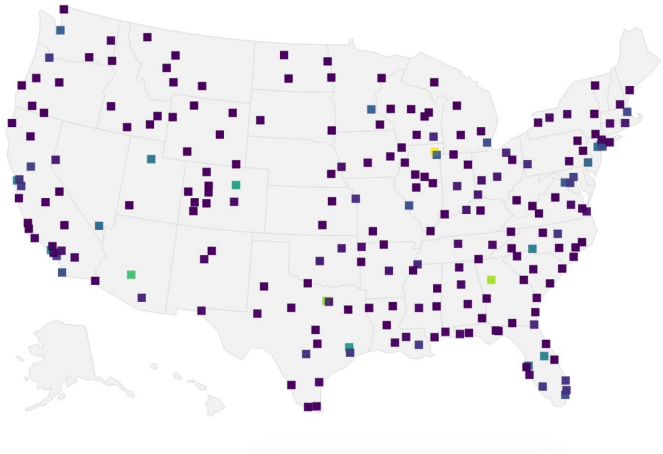
    Sets the marker opacity.
  - outliercolor (`color`)

    default: `"rgba(0, 0, 0, 0)"`

    Sets the color of the outlier sample points.
  - color (`color`)

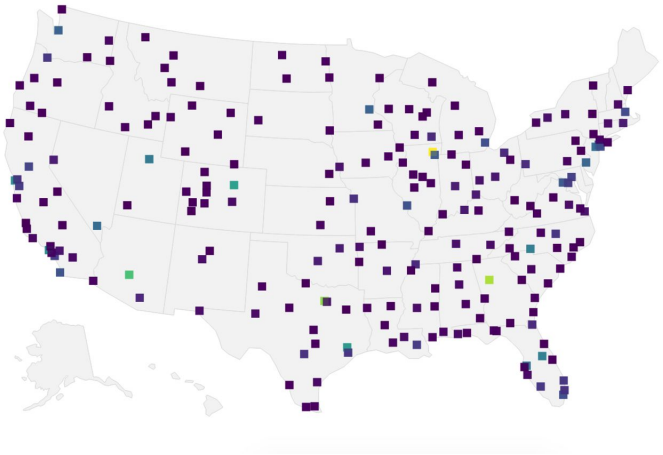
    Sets the marker color.

Build your project



Bottom-up Approach

Build your project



Start with Documentation

- marker
  - opacity (`number between or equal to 0 and 1`)

    default: `1`

    Sets the marker opacity.
  - outliercolor (`color`)

    default: `"rgba(0, 0, 0, 0)"`

    Sets the color of the outlier sample points.
  - color (`color`)

    Sets the marker color.

Today's API

module 12 exercise-1

Interpreting code

```r
# Create some data
Primates <- c('Potar monkey', 'Gorilla', 'Human', 'Rhesus monkey', 'Chimp')
Bodywt <- c(10.0, 207.0, 62.0, 6.8, 52.2)
Brainwt <- c(115, 406, 1320, 179, 440)
data <- data.frame(Primates, Bodywt, Brainwt)

# Using plotly
plot_ly(data, x = ~Bodywt, y = ~Brainwt, type = 'scatter',
        mode = 'text', text = ~Primates, textposition = 'middle right',
        textfont = list(color = '#000000', size = 16)) %>%
        layout(title = 'Primates Brain and Body Weight',
             xaxis = list(title = 'Body Weight (kg)',
                          zeroline = TRUE,
                          range = c(0, 250)),
             yaxis = list(title = 'Brain Weight (g)',
                          range = c(0,1400)))
```

What is happening in this code?

A few notes on plotly

Plot objects are created with the `plot_ly` function

Layout is controlled by passing a plot object to the `layout` function

Arguments are often **named lists**

module 12 exercise-2

Data Shape

What ***shape*** (rows, columns) should our data be in for analysis?

Wide data

```
> students
  names math_exam1 math_exam2 spanish_exam1 spanish_exam2
1 Mason         91         88            79            99
2  Tabi         82         79            88            92
3 Bryce         93         77            92            92
```

Long data

```
> students_long
   names          exam score
1  Bryce    math_exam2    77
2  Bryce spanish_exam1    92
3  Bryce spanish_exam2    92
4  Bryce    math_exam1    93
5  Mason spanish_exam1    79
6  Mason    math_exam2    88
7  Mason    math_exam1    91
8  Mason spanish_exam2    99
9   Tabi    math_exam2    79
10  Tabi    math_exam1    82
11  Tabi spanish_exam1    88
12  Tabi spanish_exam2    92
```

```r
# Make a data.frame
library(tidyr)
students <- data.frame(
  names=c('Mason', 'Tabi', 'Bryce'),
  math_exam1 = c(91, 82, 93),
  math_exam2 = c(88, 79, 77),
  spanish_exam1 = c(79, 88, 92),
  spanish_exam2 = c(99, 92, 92)
)
# Make long data (by student-exam)
students.exam.long <- gather(students, exam, score, -names)

# Make wide data (by student)
spread(students_long, exam, score)

# Make wide data (by exam)
spread(students_long, names, score)
```

Tidyr example

Upcoming...

By Thursday: Be confident with **module 12**

Due Tuesday, 11/8 (***before class***): a6-mapping-shootings