

DSBA 5122: Visual Analytics

Why R (and RStudio)?

Ryan Wesslen

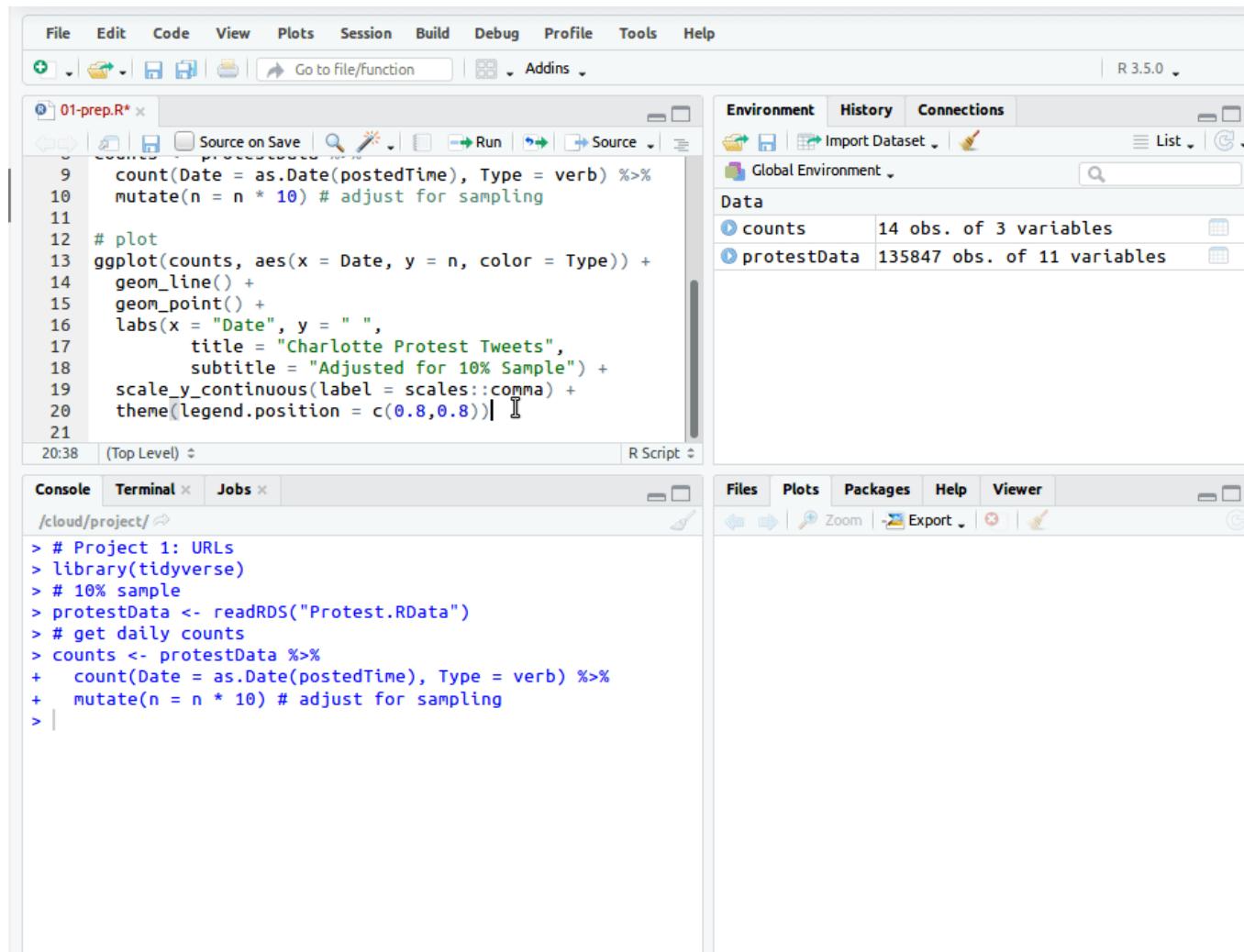
September 9, 2019

bit.ly/dsbaRintro

What is R?

```
Terminal  
R version 3.4.4 (2018-03-15) -- "Someone to Lean On"  
Copyright (C) 2018 The R Foundation for Statistical Computing  
Platform: x86_64-pc-linux-gnu (64-bit)  
  
R is free software and comes with ABSOLUTELY NO WARRANTY.  
You are welcome to redistribute it under certain conditions.  
Type 'license()' or 'licence()' for distribution details.  
  
Natural language support but running in an English locale  
  
R is a collaborative project with many contributors.  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.  
  
Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.  
  
[Previously saved workspace restored]  
> █
```

What is RStudio?



1. Why R / RStudio?

It's free, as in ... free beer?



2. Hanging with the big kids.



Miles McBain
@MilesMcBain



Well then. Another take on programming language popularity just published by IEEE Spectrum has [#rstats](#) at 5, edging out JavaScript. [spectrum.ieee.org/computing/soft...](https://spectrum.ieee.org/computing/software/the-top-programming-languages-2019) an incredible result for a DSL.



The Top Programming Languages 2019

Python remains the big kahuna, but specialist languages hold their...
[spectrum.ieee.org](https://spectrum.ieee.org/computing/software/the-top-programming-languages-2019)

♡ 57 11:17 PM - Sep 7, 2019



20 people are talking about this



bit.ly/dsbaRintro

3. R open source community makes it fun!



Tyler Morgan-Wall
@tylermorganwall



Did you know [#rayshader](#) comes with a function that can detect bodies of water directly from the elevation data? `detect_water()` looks for large contiguous flat regions of a user-defined minimum size and "flatness." Use `add_water()` to layer it on the map. [#rstats](#)



bit.ly/dsbaRintro

3. R open source community makes it fun!



Tyler Morgan-Wall
@tylermorganwall



Replying to @tylermorganwall

And here's a gist showing how to make the above "jumping River Derwent" visualization:[gist.github.com/tylermorganwal...](https://gist.github.com/tylermorganwall...)



Creating the River Derwent, jumping out of the map

Creating the River Derwent, jumping out of the map - jumping_wat...

[gist.github.com](https://gist.github.com/tylermorganwall...)

♡ 21 9:32 AM - Aug 27, 2019



[See Tyler Morgan-Wall's other Tweets](#)



bit.ly/dsbaRintro

4. Why R / RStudio?

 **JD Long**
@CMastication 

well this R=Batman, Python=Superman apparently showed up in a vendor pitch to one of my colleagues. That must make it official. [#rstats #python](#)

Analysis Tool	Similar Superhero	Super Powers in Common
R 	Batman 	<ul style="list-style-type: none">• Detective Work• Intelligence• Cunning• Usage of Tools• More Brain than Muscles
Python 	Superman 	<ul style="list-style-type: none">• Muscle Power• Super Strength• Elegance• Wide Range• More Muscles than Brain

 416 7:44 PM - Sep 5, 2018 

 172 people are talking about this 



bit.ly/dsbaRintro

Front end





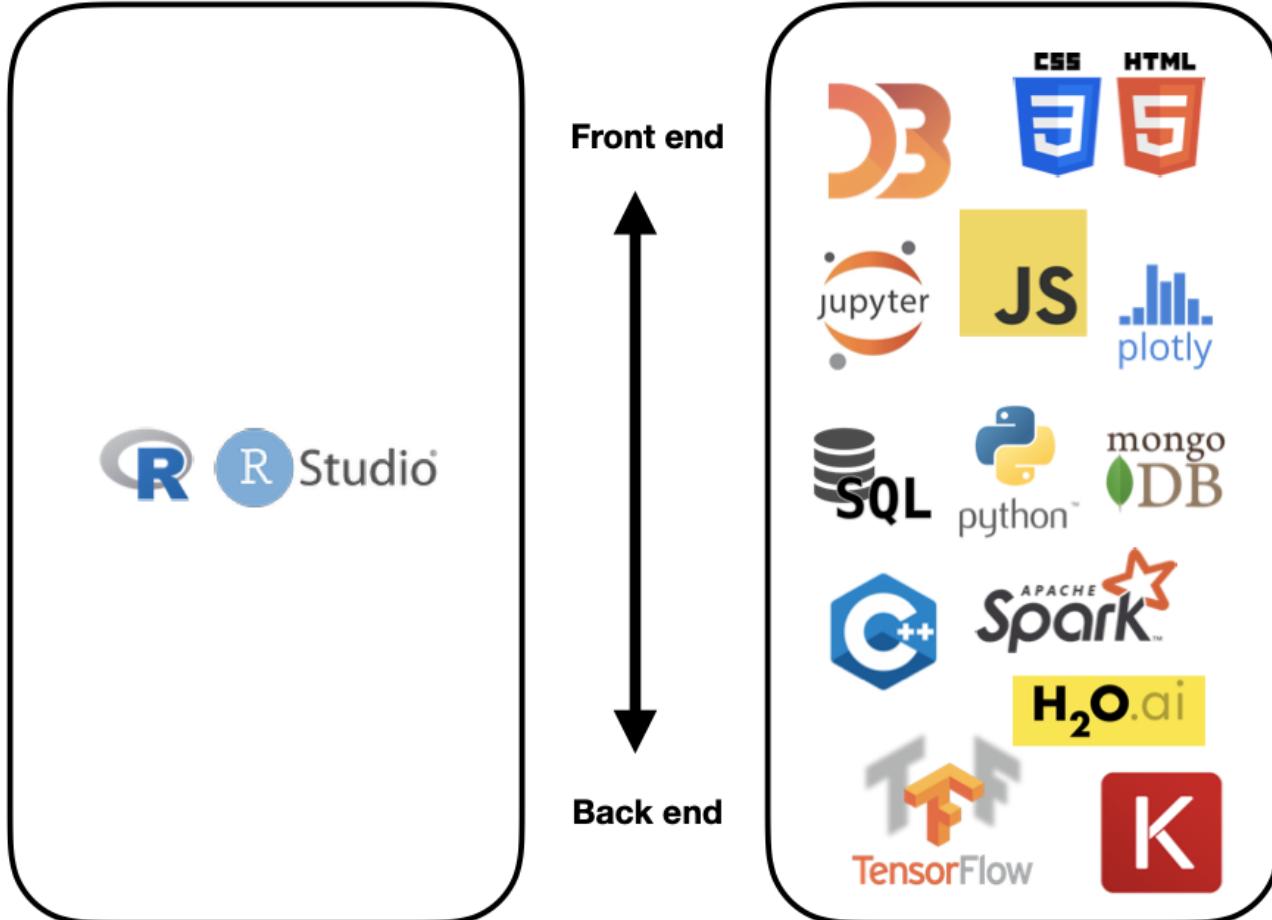
Front end



Back end



bit.ly/dsbaRintro



bit.ly/dsbaRintro

#1

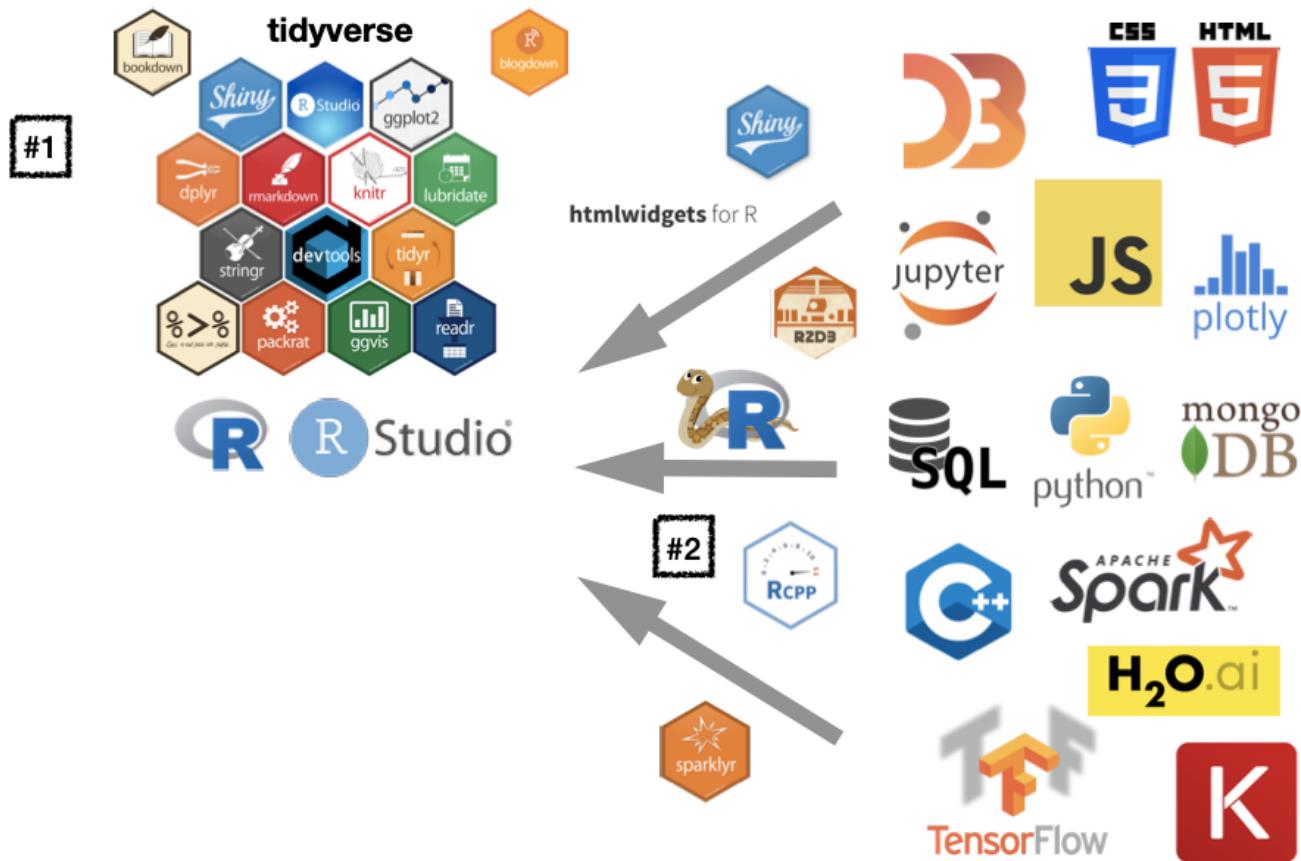


Front end

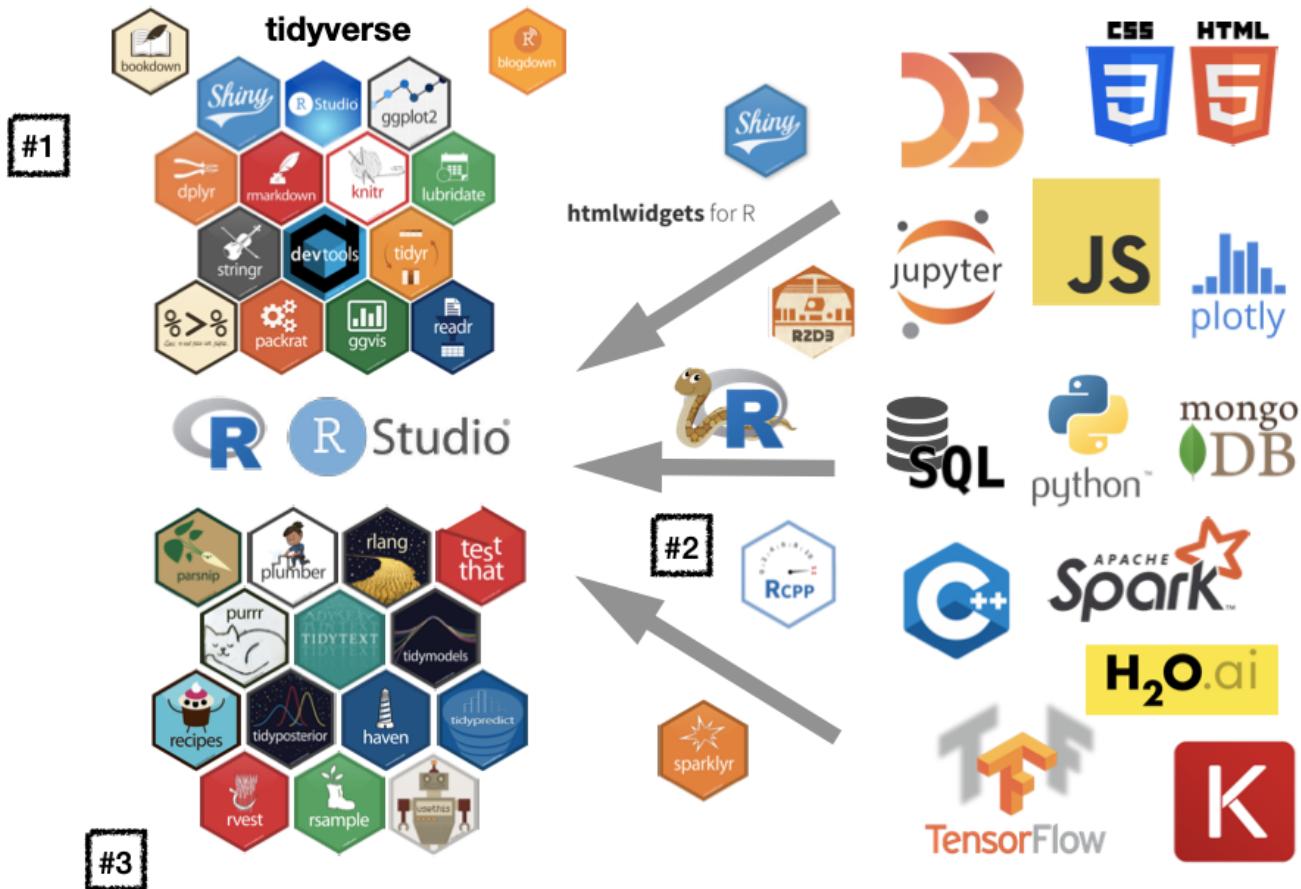
Back end



bit.ly/dsbaRintro



bit.ly/dsbaRintro



bit.ly/dsbaRintro

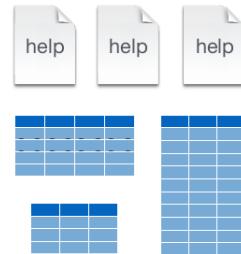
R / RStudio to the rescue



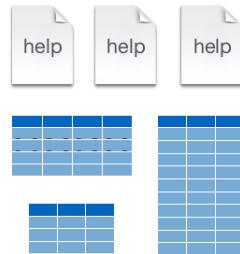
Artwork by @allison_horst

bit.ly/dsbaRintro

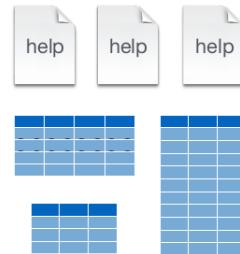
What are R packages?



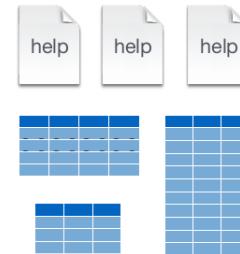
function1()
function2()
function3()
function4()



function5()
function6()
function7()
function8()



function9()
functionA()
functionB()
functionC()

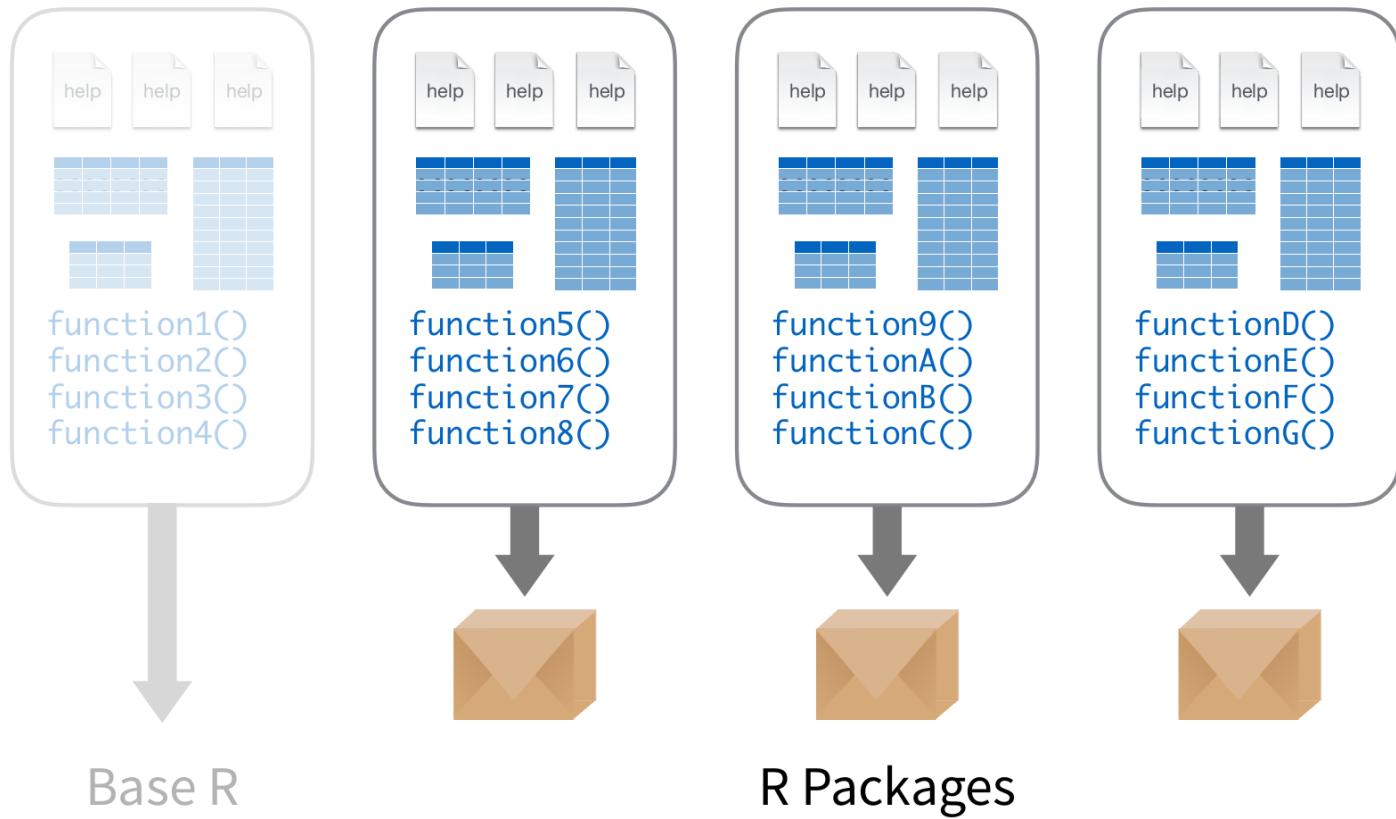


functionD()
functionE()
functionF()
functionG()

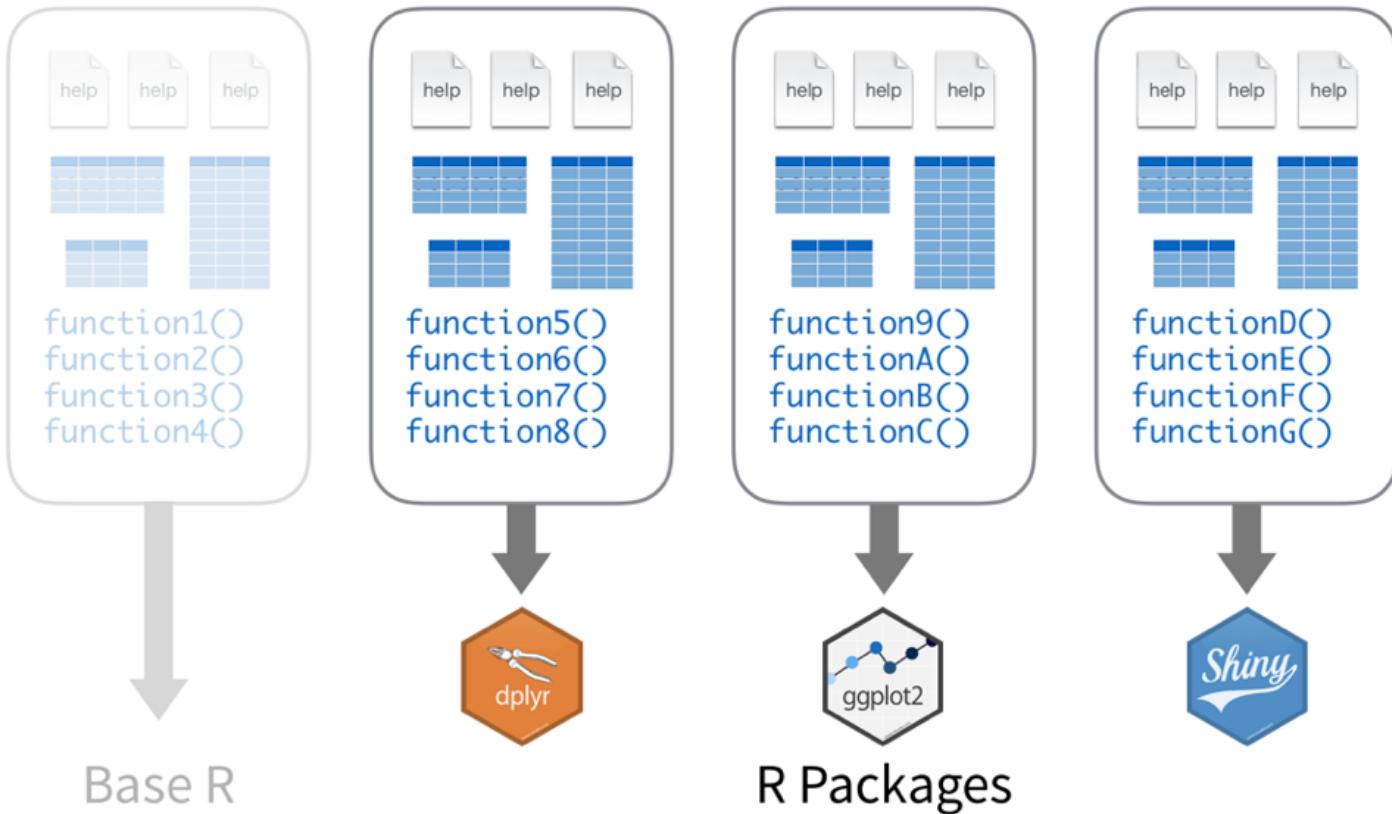
What are R packages?



What are R packages?



What are R packages?



How to install and run packages:

1

```
install.packages("foo")
```

Downloads files to computer

1 x per computer

How to install and run packages:

1

```
install.packages("foo")
```

Downloads files to computer

1 x per computer

2

```
library("foo")
```

Loads package

1 x per R Session

How to install and run packages:

1

```
install.packages("foo")
```

Downloads files to computer

1 x per computer

2

```
library("foo")
```

Loads package

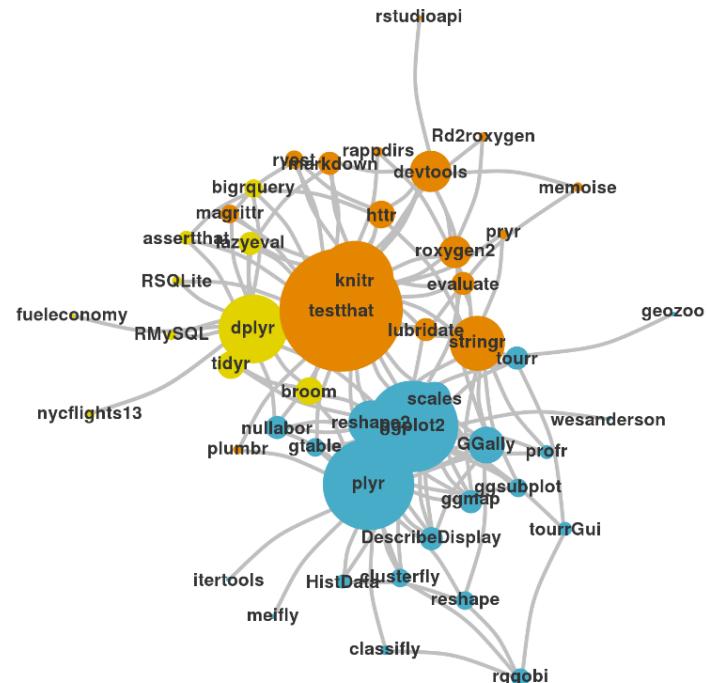
1 x per R Session

How to install and run packages:

```
install.packages("tidyverse")
```

does the equivalent of

```
install.packages("ggplot2")
install.packages("dplyr")
install.packages("tidyverse")
install.packages("readr")
install.packages("purrr")
install.packages("tibble")
install.packages("hms")
install.packages("stringr")
install.packages("lubridate")
install.packages("forcats")
install.packages("DBI")
install.packages("haven")
install.packages("httr")
install.packages("jsonlite")
install.packages("readxl")
install.packages("rvest")
install.packages("xml2")
install.packages("modelr")
install.packages("broom")
```



How to install and run packages:

```
install.packages("tidyverse")
```

does the equivalent of

```
install.packages("ggplot2")
install.packages("dplyr")
install.packages("tidyr")
install.packages("readr")
install.packages("purrr")
install.packages("tibble")
install.packages("hms")
install.packages("stringr")
install.packages("lubridate")
install.packages("forcats")
install.packages("DBI")
install.packages("haven")
install.packages("httr")
install.packages("jsonlite")
install.packages("readxl")
install.packages("rvest")
install.packages("xml2")
install.packages("modelr")
install.packages("broom")
```

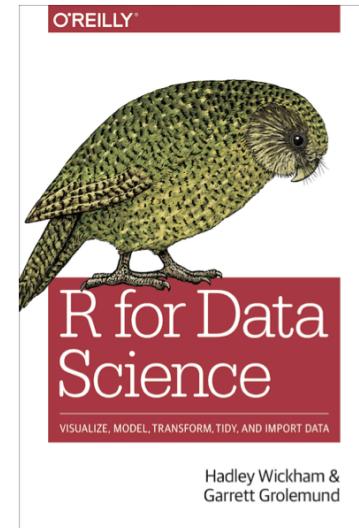
```
library("tidyverse")
```

does the equivalent of

```
library("ggplot2")
library("dplyr")
library("tidyr")
library("readr")
library("purrr")
library("tibble")
```

RStudio Visualization Ecosystem

tidyverse



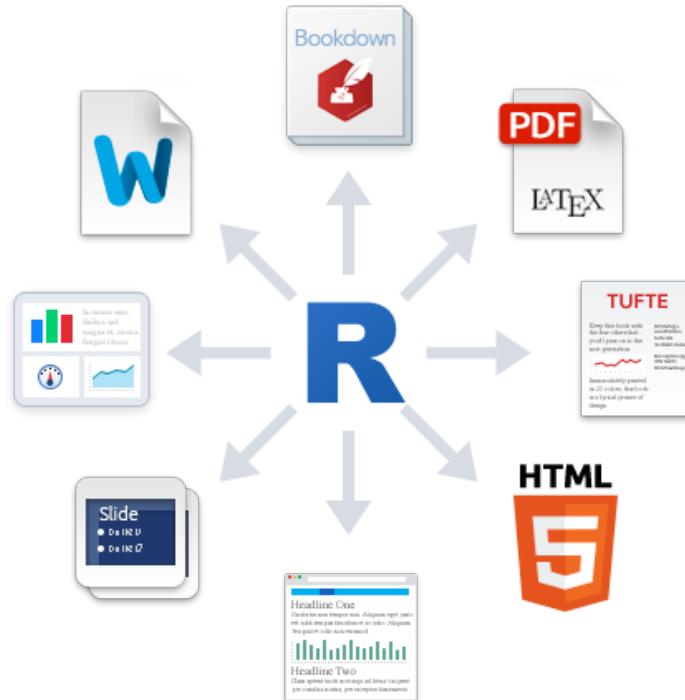
tidyverse.org

bit.ly/dsbaRintro

RStudio Visualization Ecosystem

tidyverse
rmarkdown

Analyze. Share. Reproduce.



rmarkdown.rstudio.com

bit.ly/dsbaRintro

RStudio Visualization Ecosystem

tidyverse

JavaScript libraries as R packages!
htmlwidgets.org

rmarkdown

htmlwidgets

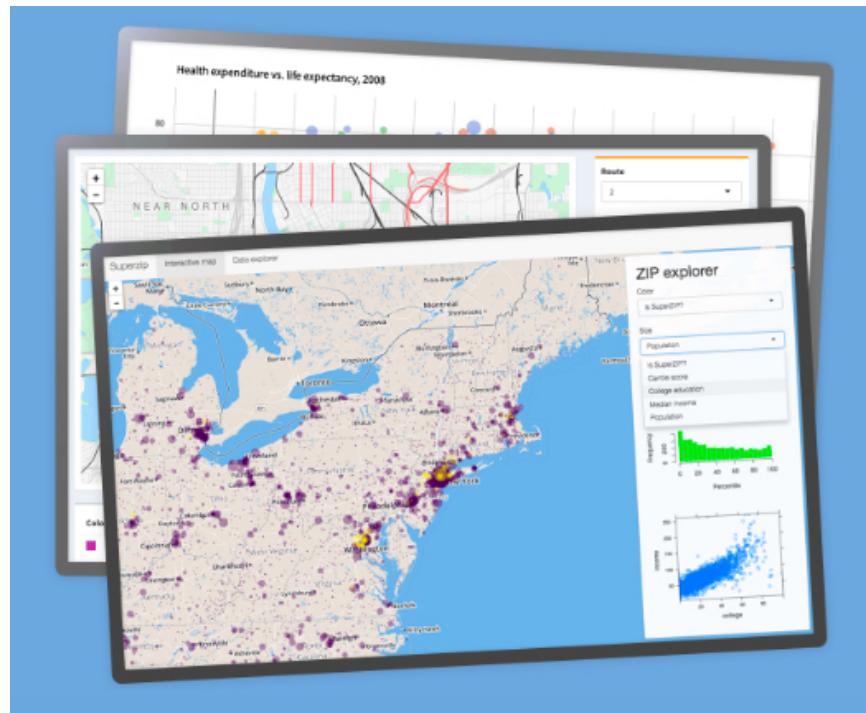
RStudio Visualization Ecosystem

tidyverse

rmarkdown

htmlwidgets

shiny



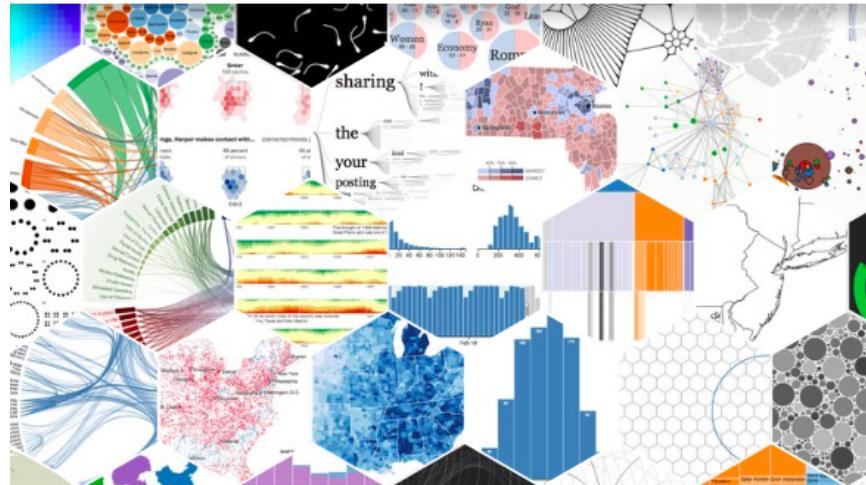
shiny.rstudio.com/

bit.ly/dsbaRintro

RStudio Visualization Ecosystem

tidyverse
rmarkdown
htmlwidgets
shiny
r2d3

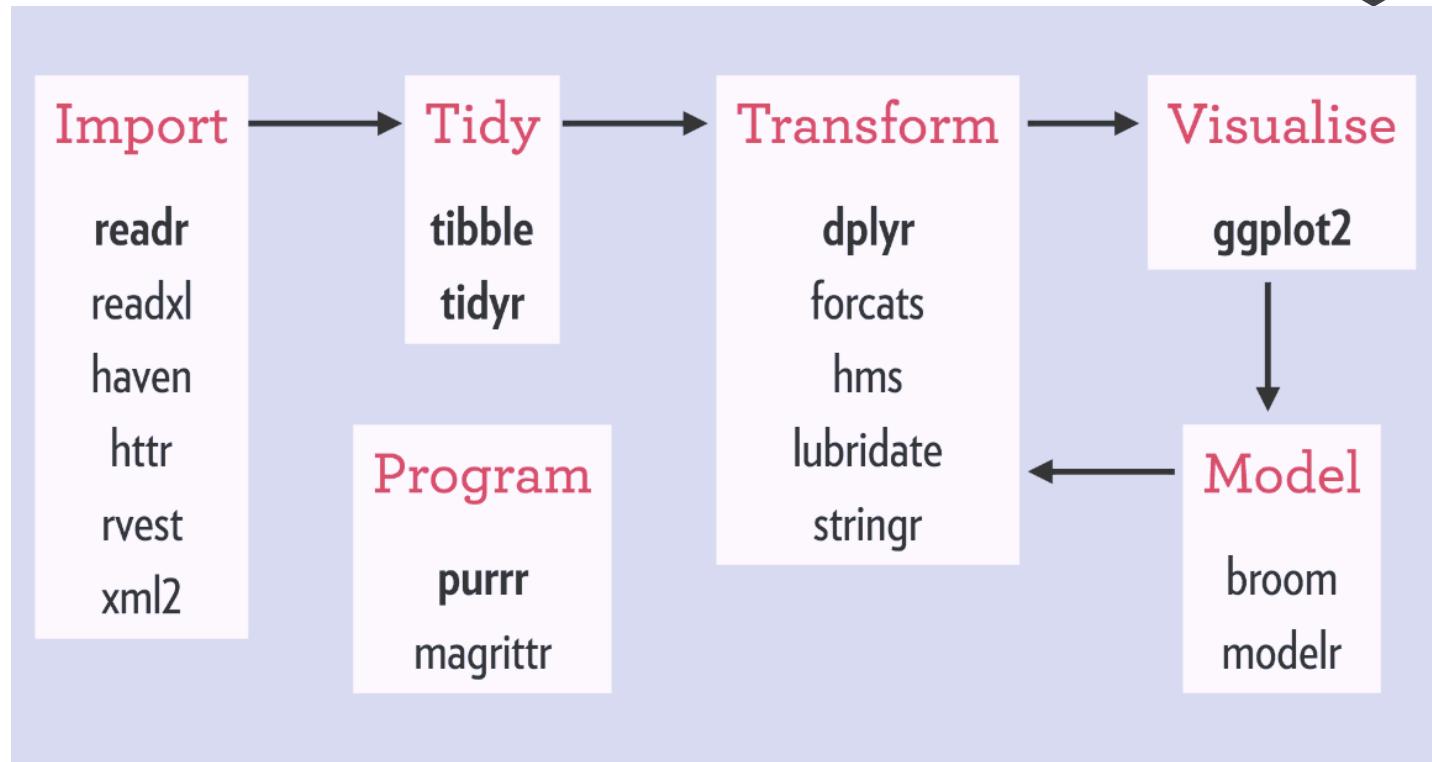
 Data-Driven Documents



d3js.org and r2d3 package

bit.ly/dsbaRintro

tidyverse: Data science process



tidyverse: dplyr + piping



US Unemployment rate

```
library(tidyverse)

# read in using readr's read_csv function
df <- read_csv("dsba-rintro_files/us-unemployment.csv",
  col_types = cols(
    date = col_date(),                  # read in as date
    unemployment = col_double()        # read in as double
  )
)

# view the data via dplyr's glimpse
glimpse(df)
```

```
## Observations: 853
## Variables: 2
## $ date          <date> 1948-01-01, 1948-02-01, 1948-03-01, 1948-04-01, 19
## $ unemployment <dbl> 3.4, 3.8, 4.0, 3.9, 3.5, 3.6, 3.6, 3.9, 3.8, 3.7, 3.
```

tidyverse: dplyr + piping



What are the years with the highest unemployment rate in December?

```
library(lubridate) # non-standard tidyverse for dates/times

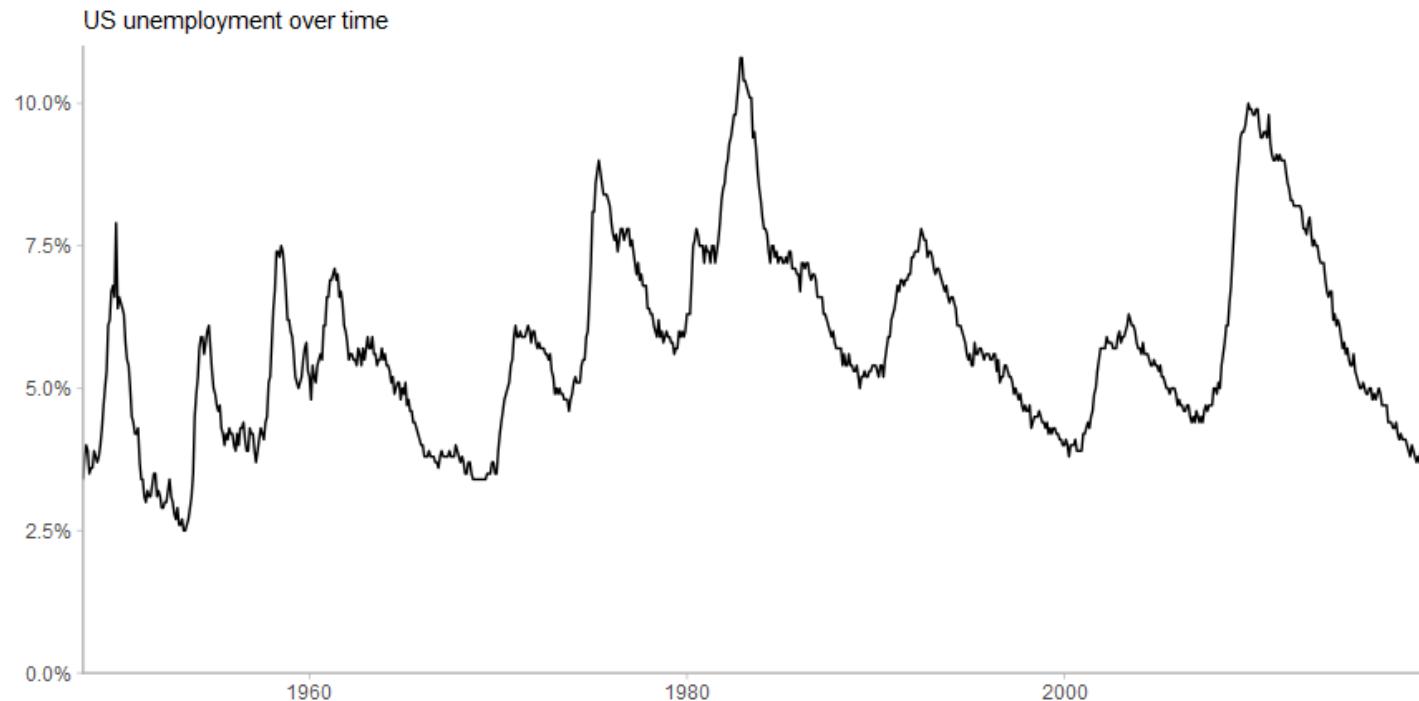
df %>%
  mutate(month = month(date)) %>% # create new column for month
  filter(month == 12) %>%          # filter for only December values
  arrange(desc(unemployment)) %>% # order by unemployment rate
  head(n = 6)                   # show top five
```

```
## # A tibble: 6 x 3
##   date      unemployment month
##   <date>          <dbl> <dbl>
## 1 1982-12-01     10.8    12
## 2 2009-12-01      9.9    12
## 3 2010-12-01      9.3    12
## 4 1981-12-01      8.5    12
## 5 2011-12-01      8.5    12
## 6 1983-12-01      8.3    12
```

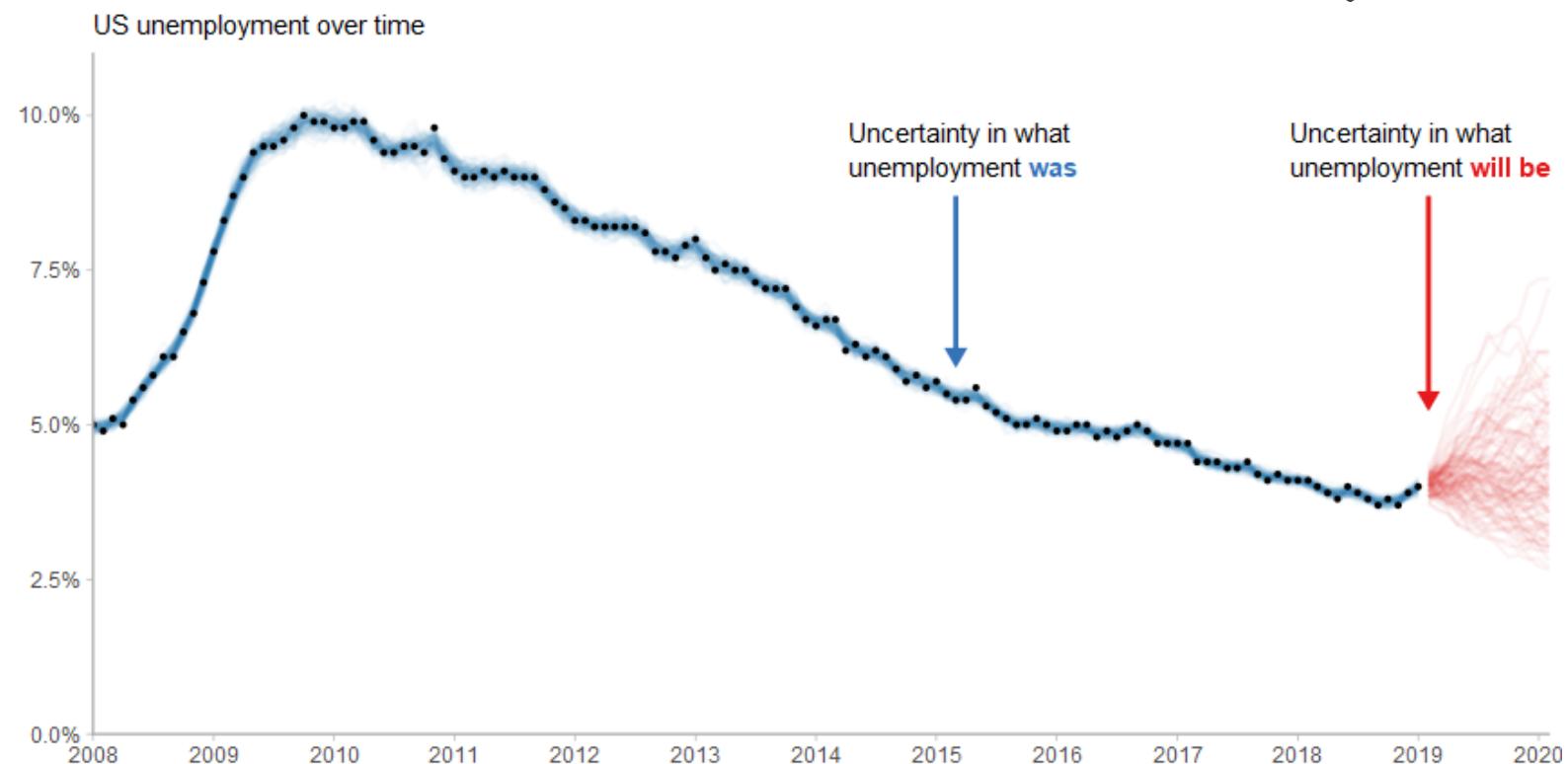
tidyverse: dplyr + piping + ggplot2



```
df %>%
  ggplot(aes(x = date, y = unemployment)) +
  geom_line() +
  coord_cartesian(ylim = c(0, .11), expand = FALSE),
  scale_y_continuous(labels = scales::percent) +
  labs(x = NULL, y = NULL, subtitle = "US unemployment over time")
```



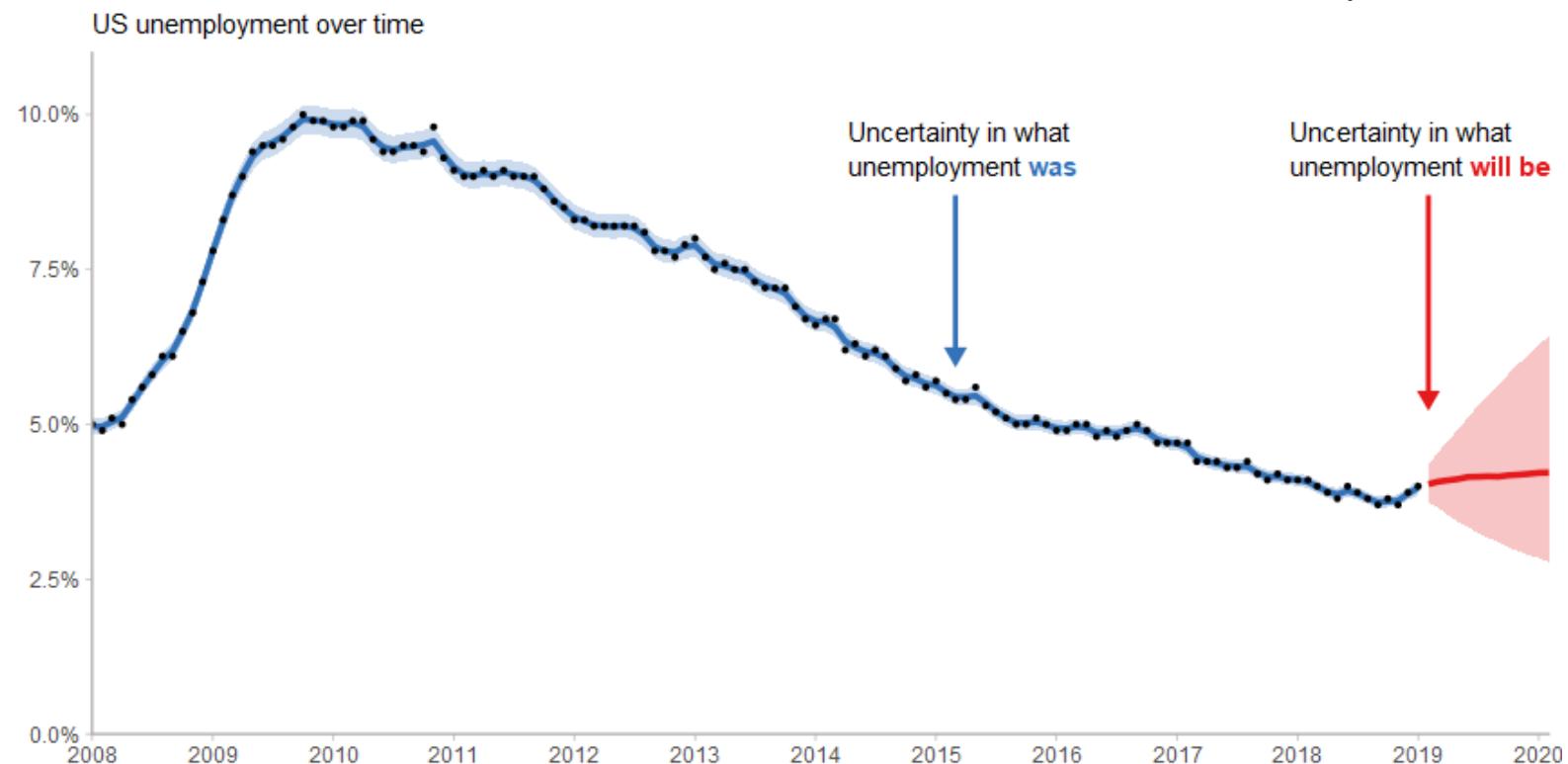
ggplot2 extensions



Source: Matthew Kay

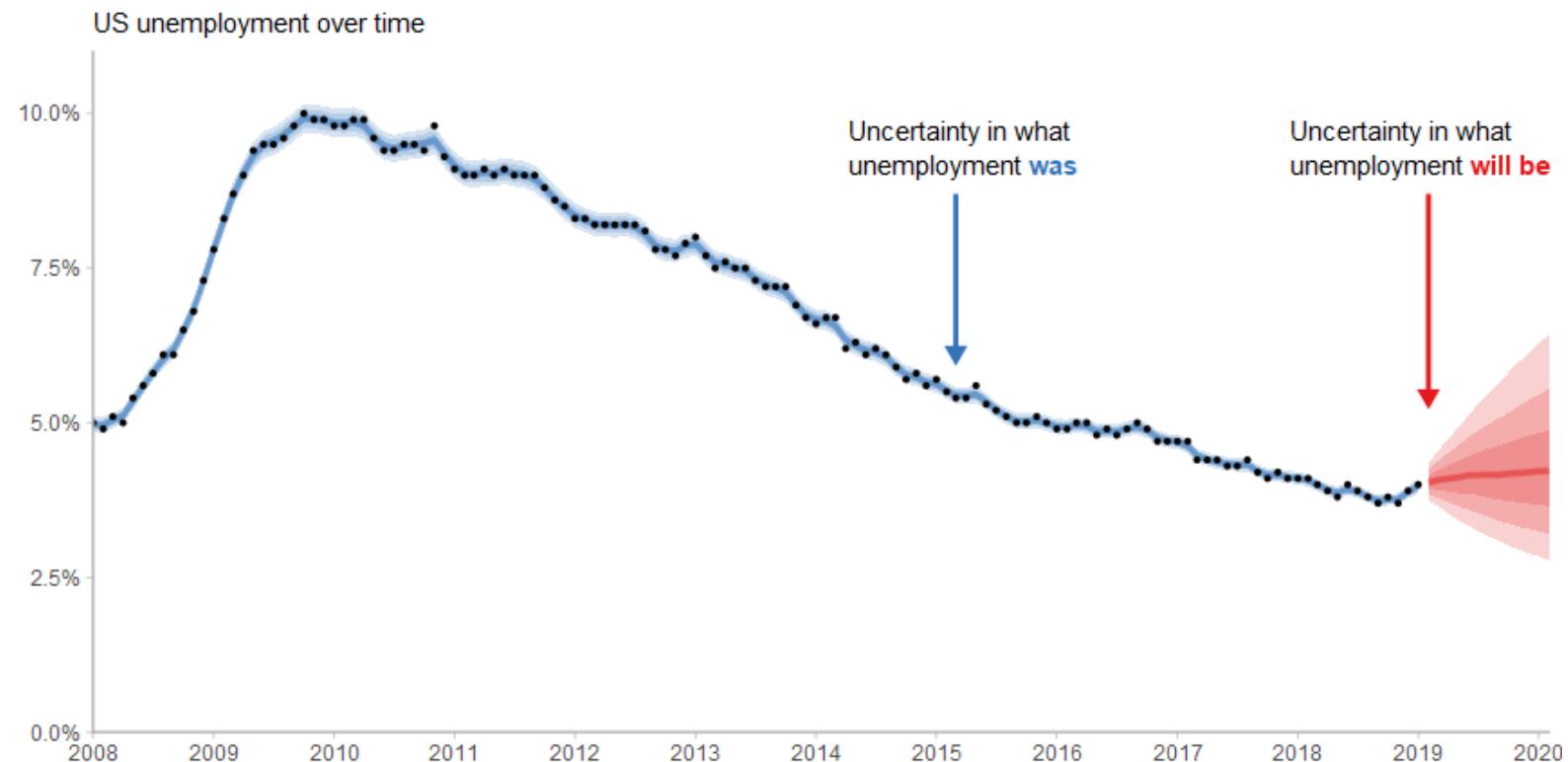
bit.ly/dsbaRintro

ggplot2 extensions



Source: Matthew Kay

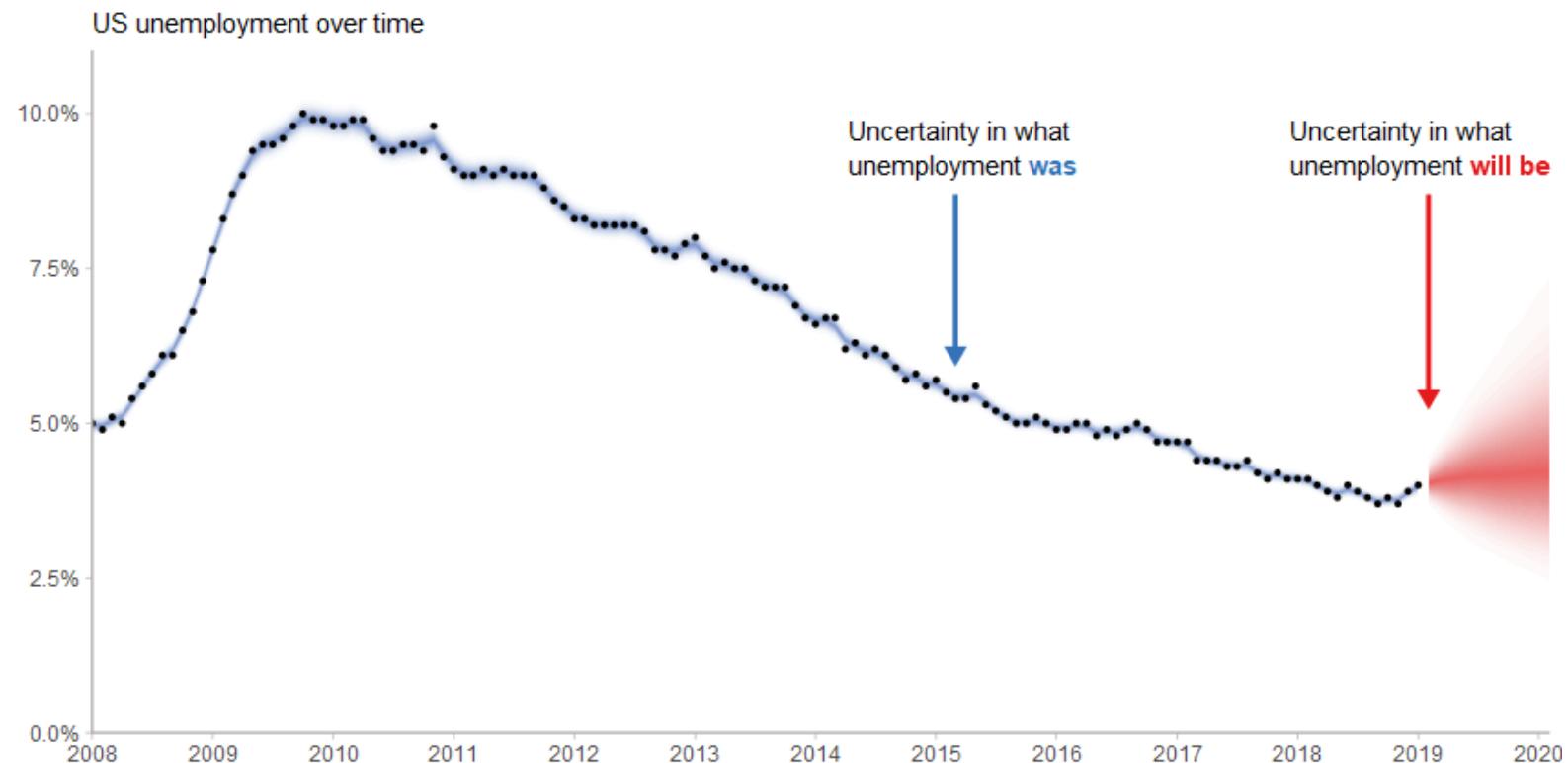
ggplot2 extensions



Source: Matthew Kay

bit.ly/dsbaRintro

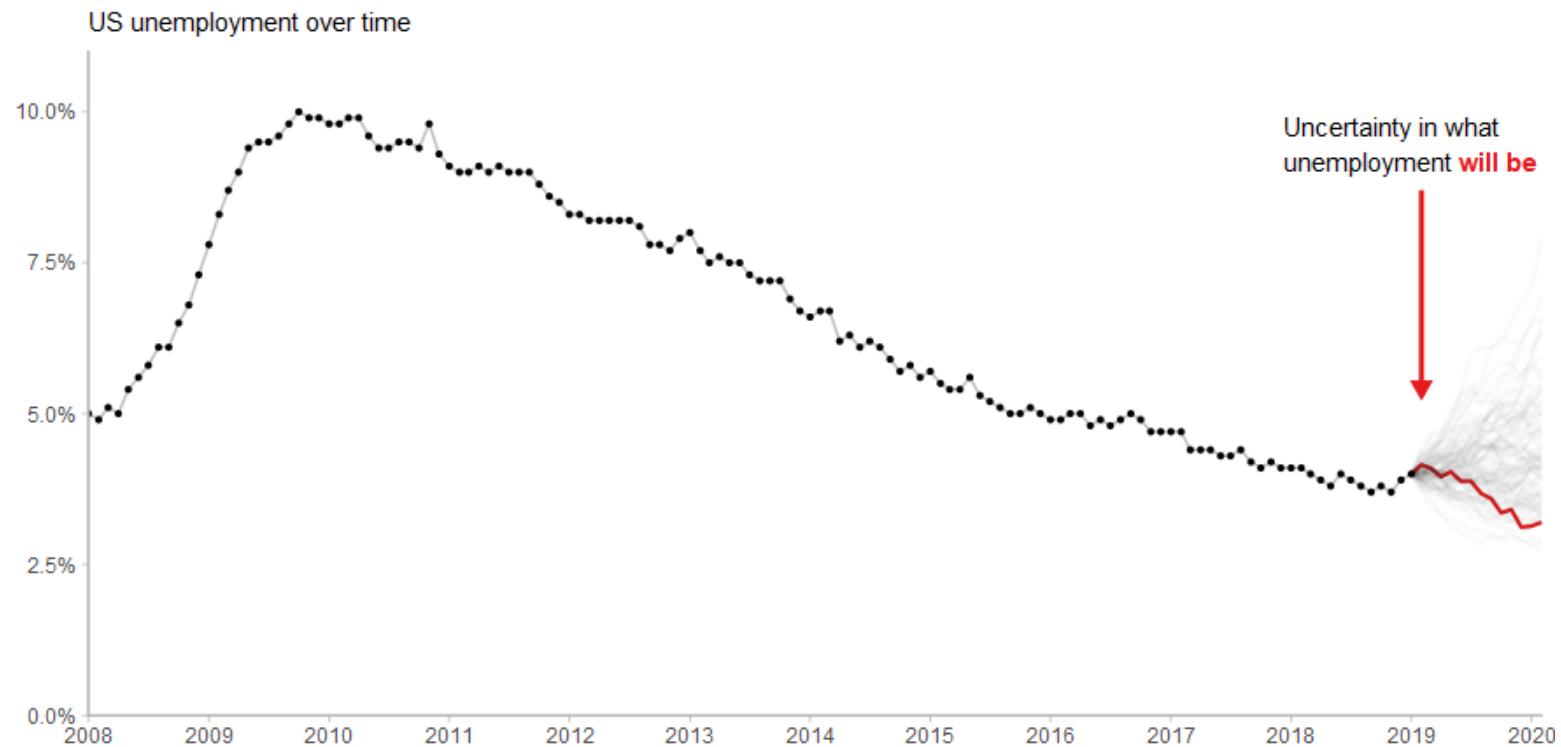
ggplot2 extensions



Source: Matthew Kay

bit.ly/dsbaRintro

ggplot2 extensions: ganimate



Hypothetical Outcome Plots (HOP): `tidybayes` & `ganimate`

Rmarkdown: reproducibility



RStudio RMarkdown documentation

The screenshot shows the RStudio interface with the following details:

- Code Editor:** The left pane displays the RMarkdown file `2-chunks.Rmd`. The code includes chunks of R code with specific options like `include = FALSE` and `echo = FALSE`, and a section describing the Magma color palette.
- Preview Area:** The right pane shows the rendered output of the RMarkdown file. It features a title "Magma Demo" and a section titled "Magma colors". Below this is a contour plot of Maunga Whau volcano in Auckland, New Zealand, using the Magma color palette. The plot has axes ranging from 0.0 to 1.0.
- Status Bar:** At the bottom, the status bar indicates "2:14" and "Magma Demo".
- Console:** A small "Console" tab is visible at the bottom left.

bit.ly/dsbaRintro

rmarkdown: word doc's



~/Downloads/text-popup - RStudio

my-doc-file.Rmd x Addins ▾

```
1 * ---  
2 title: "My report"  
3 output:  
4   _word_document: default  
5 ---  
6  
7 `r setup, include=FALSE}  
8 knitr::opts_chunk$set(echo = TRUE)  
9  
10  
11 ## R Markdown  
12  
13 Write here to describe your code!  
14  
15 `r mpg}  
16 library(tidyverse)  
17 library(ggalluvial)  
18  
19 data(vaccinations)  
20  
21 levels(vaccinations$response) <- rev(levels(vaccinations$response))  
22  
23 ggplot(vaccinations,  
24   aes(x = survey, stratum = response, alluvium = subject,  
25     y = freq,  
26     fill = response, label = response)) +  
27   scale_x_discrete(expand = c(.1, .1)) +  
28   geom_flow() +  
29   geom_stratum(alpha = .5) +  
30   geom_text(stat = "stratum", size = 3) +  
31   theme(legend.position = "none") +  
32   ggtitle("vaccination survey responses at three points in time")  
33 ...
```

R Console

vaccination survey responses at three points in time

1000

Always Sometimes Never

1:4 My report R Markdown

Console Terminal R Markdown Jobs

~/Downloads/text-popup/

> |

Files Plots Packages Help Viewer

New Folder Delete Rename More

Home Downloads text-popup

Name	Size	Modified
text-popup.Rproj	205 B	May 19, 2019, 3:06 PM
.Rhistory	19.1 KB	May 18, 2019, 12:38 PM
cfpb-birdseye		
R		
config.yaml	0 B	May 19, 2019, 3:15 PM

bit.ly/dsbaRintro

Even more of rmarkdown



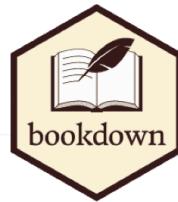
The screenshot shows a bookdown document structure. On the left, a sidebar lists sections: 3.3 PDF document, 3.3.1 Table of contents, 3.3.2 Figure options, 3.3.3 Data frame printing, 3.3.4 Syntax highlighting, 3.3.5 LaTeX options, 3.3.6 LaTeX packages for citations, 3.3.7 Advanced customization, 3.3.8 Other features, 3.4 Word document, 3.4.1 Other features, 3.5 OpenDocument Text document, 3.5.1 Other features, 3.6 Rich Text Format document, 3.6.1 Other features, 3.7 Markdown document, and 3.7.1 Markdown variants. The main content area displays the '3.3 PDF document' section, which includes a heading, a note about specifying the `pdf_document` output format in YAML metadata, and a code block showing raw LaTeX code for a document titled "Habits". Below the code block, there is a note about raw LaTeX macros and Pandoc's documentation, and another note about LaTeX requirements for Beamer slides.

- "Data Visualization: A Practical Introduction by Kieran Healy"
- "Fundamentals of Data Visualization" by Claus Wilke
- "R for Data Science" by Garrett Grolemund and Hadley Wickham

bit.ly/dsbaRintro

Even more of rmarkdown

3.3 PDF document
3.3.1 Table of contents
3.3.2 Figure options
3.3.3 Data frame printing
3.3.4 Syntax highlighting
3.3.5 LaTeX options
3.3.6 LaTeX packages for citations
3.3.7 Advanced customization
3.3.8 Other features
3.4 Word document
3.4.1 Other features
3.5 OpenDocument Text document
3.5.1 Other features
3.6 Rich Text Format document
3.6.1 Other features
3.7 Markdown document
3.7.1 Markdown variants



3.3 PDF document

To create a PDF document from R Markdown, you specify the `pdf_document` output format in the YAML metadata:

```
---
```

```
title: "Habits"
author: John Doe
date: March 22, 2005
output: pdf_document
---
```

Within R Markdown documents that generate PDF output, you can use raw LaTeX, and even define LaTeX macros. See Pandoc's documentation on the [raw_tex extension](#) for details.

Note that PDF output (including Beamer slides) requires an installation of LaTeX (see Chapter 1).



DSBA 5122: VISUAL ANALYTICS

[SYLLABUS](#) [SCHEDULE](#) [ASSIGNMENTS](#) [CANVAS](#) [RSTUDIO.CLOUD](#)
[CLASS SLACK](#) [DATA CAMP](#) [REFERENCE](#)



DATA VISUALIZATION

THIS SITE CONTAINS the syllabus, schedule, and assignments for DSBA5122: Visual Analytics, held during Spring 2019 at University of North Carolina at Charlotte.

By the end of this course, you will become:

INSTRUCTOR
• Ryan Wesslen
• CenterCity
• rwesslen@uncc.edu
• [@ryanwesslen](https://twitter.com/ryanwesslen)
• Office Hours via Calendly

My blogdown website (all materials public) for my DSBA 5122: Visual Analytics course: dsba5122.com

- "Data Visualization: A Practical Introduction" by Kieran Healy
- "Fundamentals of Data Visualization" by Claus Wilke
- "R for Data Science" by Garrett Grolemund and Hadley Wickham

bit.ly/dsbaRintro

What if you need JavaScript libraries?

```
<!doctype html>
...<html> == $0
  ><head>...</head>
  ><body class=">
    ><div id="mapid" style="width: 600px; height: 400px; position: relative;" class="leaflet-container leaflet-touch leaflet-fade-anim leaflet-grab leaflet-drag leaflet-touch-zoom" tabindex="0">...</div>
    ><script>

      var mymap = L.map('mapid').setView([51.505, -0.09], 13);

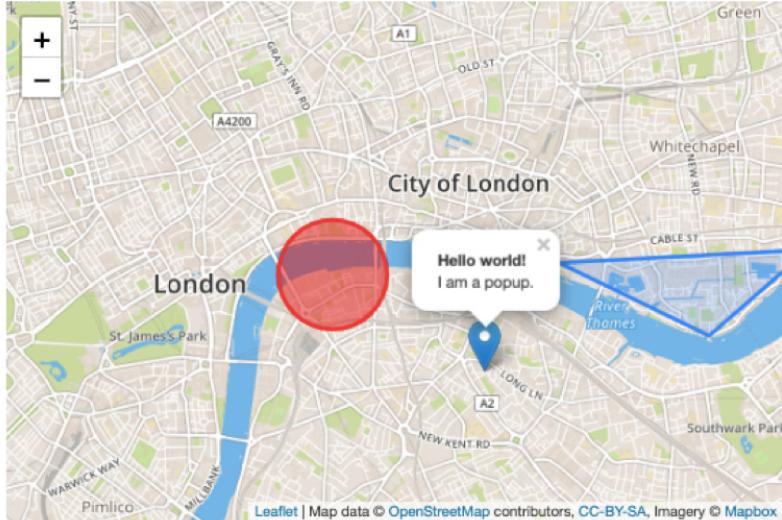
      L.tileLayer('https://api.tiles.mapbox.com/v4/{id}/{z}/{x}/{y}.png?
access_token=pk.eyJ1IjoibWFwYm94IiwiYSI6ImNpejY4NXVycTA2emYycXBndHRqcmZ3N3gifQ.rJcFIG214Ari
ISLb86B5aw', {
        maxZoom: 18,
        attribution: 'Map data © OpenStreetMap contributors, ' +
          '<a href="https://creativecommons.org/licenses/by-sa/2.0/">CC-BY-SA</a>, ' +
          'Imagery © <a href="https://www.mapbox.com/">Mapbox</a>',
        id: 'mapbox.streets'
      }).addTo(mymap);

      L.marker([51.5, -0.09]
        .bindPopup("<b>Hello world!</b>
I am a popup.");

      L.circle([51.508, -0.1
        color: 'red',
        fillColor: '#f03',
        fillOpacity: 0.5
      ]).addTo(mymap).bindPo

      L.polygon([
        [51.509, -0.08],
        [51.503, -0.06],
        [51.51, -0.047]
      ]).addTo(mymap).bindPo

      var popup = L.popup();
```



Leaflet | Map data © OpenStreetMap contributors, CC-BY-SA, Imagery © Mapbox

bit.ly/dsbaRintro

htmlwidgets: JavaScript Packages in R

plotly 861

Create interactive web graphics via Plotly's JavaScript graphing library.

- **author:** cpsievert
- **tags:** d3, webgl
- **js libraries:** plotly.js

DiagrammeR 680

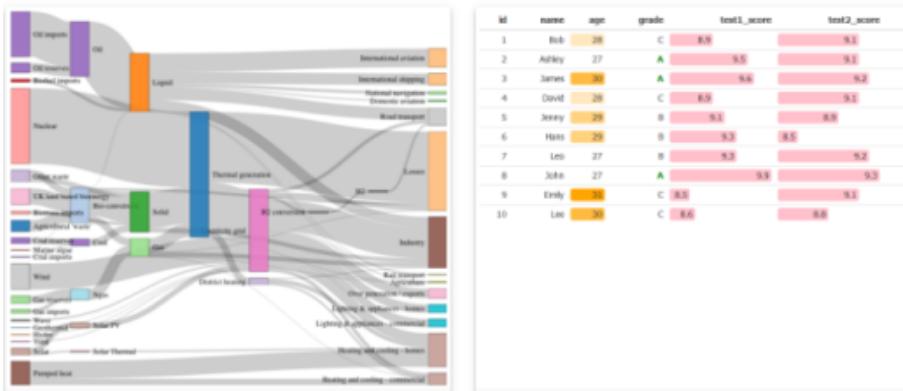
Easily create graph diagrams using R.

- **author:** rich-iannone
- **tags:** visualization, diagram, networks
- **js libraries:** d3,viz,mermaid

leaflet 306

Leaflet is an open-source JavaScript library for interactive maps. This package makes it easy to create Leaflet maps from R.

- **author:** rstudio
- **tags:** visualization, maps
- **js libraries:** leaflet



Show 10 entries Search:

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5	3.4	1.5	0.2	setosa
9	4.4	2.9	1.4	0.2	setosa
10	4.9	3.3	1.5	0.1	setosa

Showing 1 to 10 of 150 entries

Previous ... Next

bit.ly/dsbaRintro

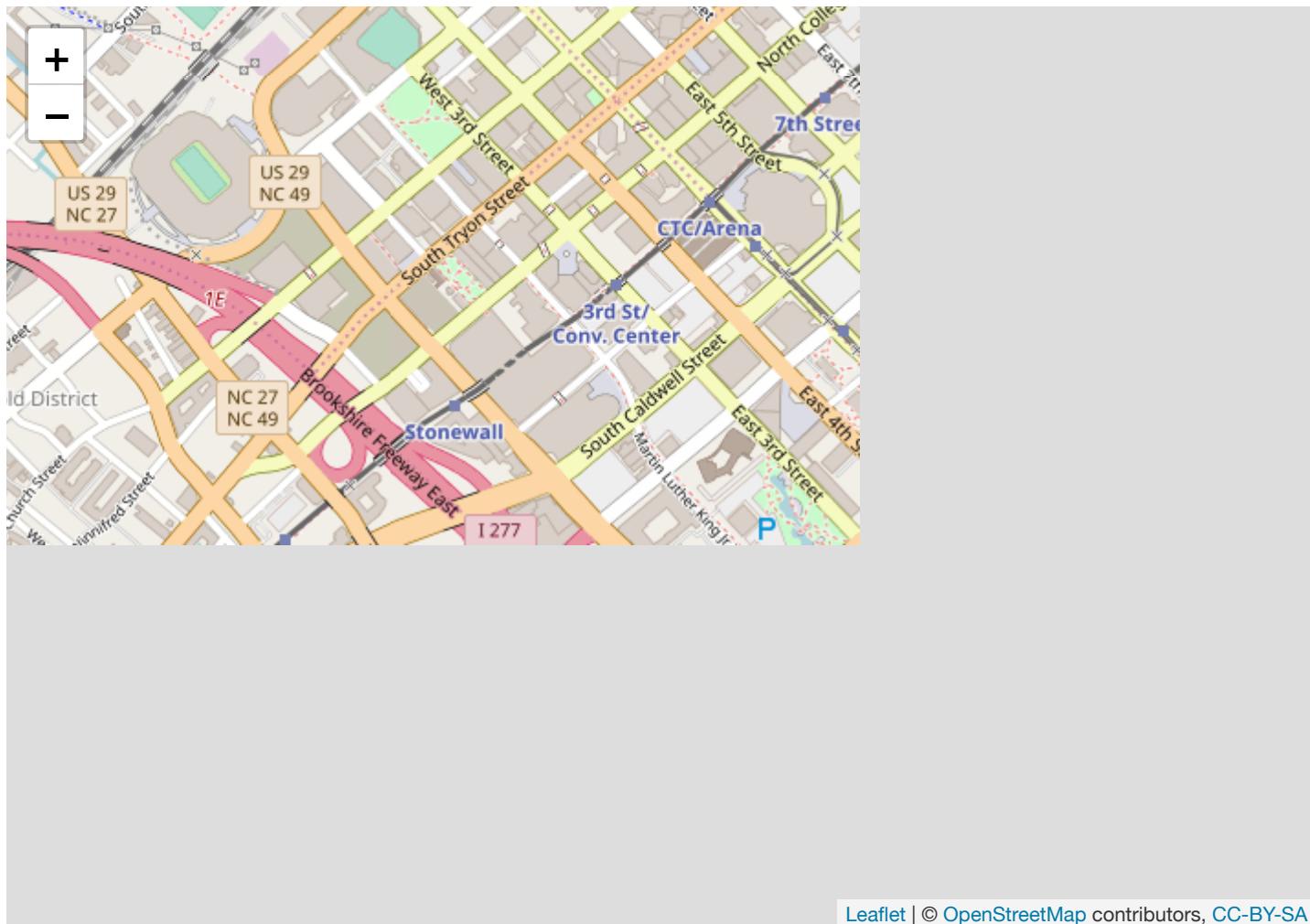
htmlwidgets: leaflet + rtweet

```
library(leaflet); library(rtweet) # see rtweet.info

tweets <- search_tweets(
  q = "beer",                      # query term
  geocode = "35.23,-80.84,10mi", # geolocation: charlotte + 10mi
  n = 3200) %>%                  # max tweets via public api
  lat_lng() %>%                  # convert lng/lat to dataframe
  filter(!is.na(lng))            # remove records missing lng/lat

tweets %>%                      # tweets data frame
  leaflet() %>%                  # pass to leaflet
  addTiles() %>%                  # add tiles
  addCircleMarkers(
    lng=tweets$lng,             # latitude
    lat=tweets$lat,              # longitude
    popup = tweets$text, # add tweet text
    clusterOptions = markerClusterOptions()
  )
```

htmlwidgets: leaflet



bit.ly/dsbaRintro

htmlwidgets: JavaScript's DT (DataTable)

source	text
Untappd	Drinking a Just Too Likeable Lemongrass Miso Gose by @OozlefinchBeers at @SaludNODA — https://t.co/cXWDL7tpdc
Untappd	Drinking a Psychedelic Flow by @SycamoreBrewing @ Tru by Hilton Charlotte Ayrshire — https://t.co/8P7TStfCI1
Instagram	Exactly how I'm going to sleep tonight after the day I've had. Dead Sleep S'mores Stout Brewed and conditioned with graham, marshmallow, and chocolate 8% @dg_beer @ Charlotte, North Carolina https://t.co/tXCjsM4jc5

Showing 4 to 6 of 15 entries

Previous

1

2

3

4

5

bit.ly/dsbaRintro Next

Shiny



A Shiny app is a web page (UI) connected to a computer running a live R session (Server).



Users can manipulate the UI, which will cause the server to update the UI's displays (by running R code).

Shiny



```
library(shiny)  
  
ui <- fluidPage()  
  
server <- function(input, output) {}  
  
shinyApp(ui = ui, server = server)
```

A screenshot of a web browser window titled " ~/Downloads/text-popup/R - Shiny". The address bar shows "http://127.0.0.1:6183". Below the address bar are three circular icons and two buttons: "Open in Browser" and "Publish". The main content area of the browser is completely blank, indicating that the Shiny application has not yet loaded or is in its initial state.

bit.ly/dsbaRintro

Shiny



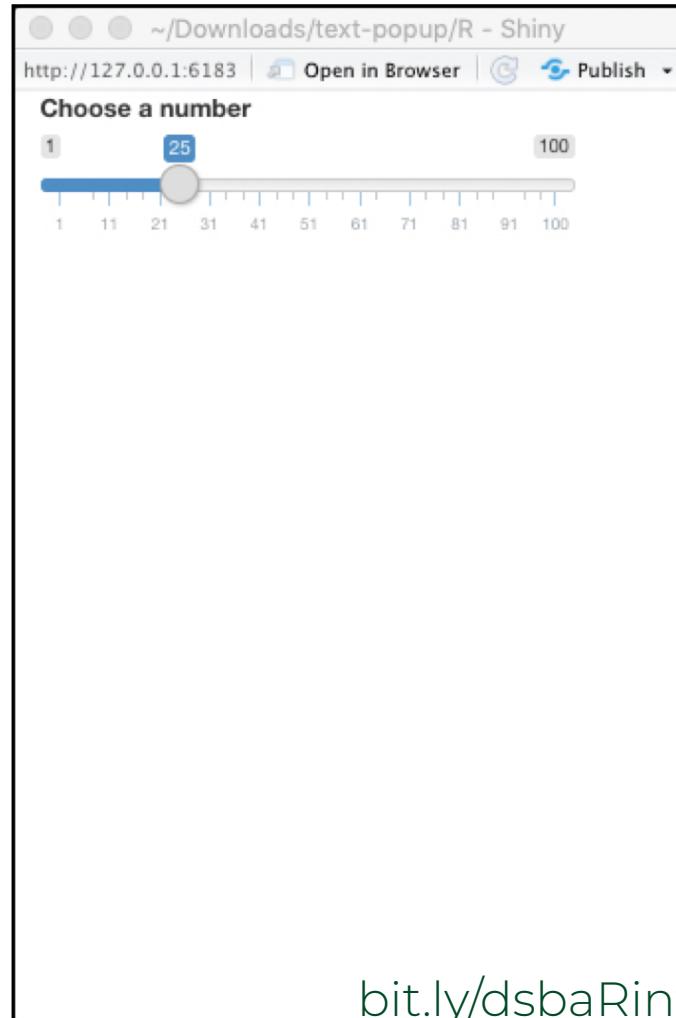
```
library(shiny)

ui <- fluidPage(
  sliderInput(inputId = "num",
    label = "Choose a number",
    value = 25, min = 1, max = 100)
)

server <- function(input, output) {

}

shinyApp(ui = ui, server = server)
```



bit.ly/dsbaRintro

Shiny



```
library(shiny)

ui <- fluidPage(
  sliderInput(inputId = "num",
    label = "Choose a number",
    value = 25, min = 1, max = 100),
  plotOutput("hist")
)

server <- function(input, output) {

}

shinyApp(ui = ui, server = server)
```

A screenshot of a web browser displaying a Shiny application. The title bar shows the path ~./Downloads/text-popup/R - Shiny and the URL http://127.0.0.1:6183. The browser interface includes tabs for 'Open in Browser' and 'Publish'. The main content area is titled 'Choose a number' and features a horizontal slider with a blue track and a grey handle. The handle is positioned at the value 25, which is also displayed in a small blue box above the slider. Below the slider, numerical tick marks are visible at intervals of 10, ranging from 1 to 100. To the right of the slider, there is a large, empty rectangular box with a black border, representing a plot area. Below this box, a text message reads: 'Need to specify in the server function what to render in this output.' At the bottom of the slide, there is a red footer bar with the text 'bit.ly/dsbaRintro'.

Shiny

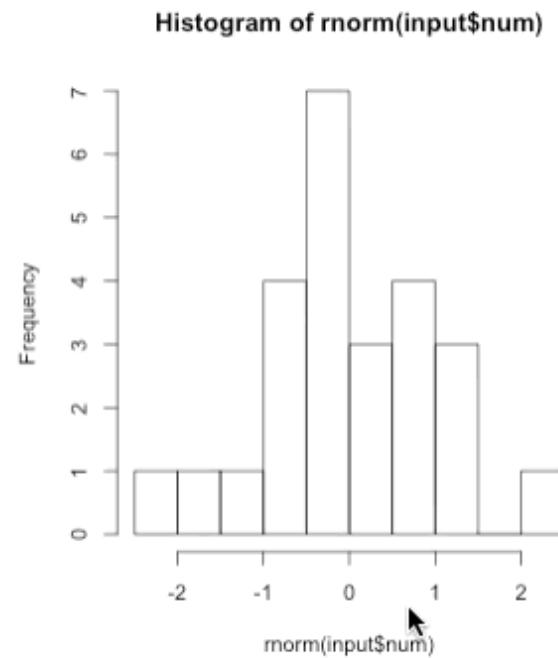


```
library(shiny)

ui <- fluidPage(
  sliderInput(inputId = "num",
    label = "Choose a number",
    value = 25, min = 1, max = 100),
  plotOutput("hist")
)

server <- function(input, output) {
  output$hist <- renderPlot({
    hist(rnorm(input$num))
  })
}

shinyApp(ui = ui, server = server)
```



bit.ly/dsbaRintro

RStudio's Shiny Gallery

[Products](#)[Resources](#)[Pricing](#)[About Us](#)[Blogs](#)

Shiny User Showcase

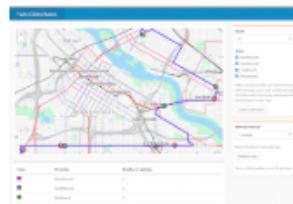
Shiny is designed for fully interactive visualization, using JavaScript libraries like d3, Leaflet, and Google Charts. Our users create fantastic examples, and some have shared them with the community. Here are some examples that we particularly like.

Shiny Apps for the Enterprise



MARKETING EFFECTS

See the effects of your marketing campaigns.



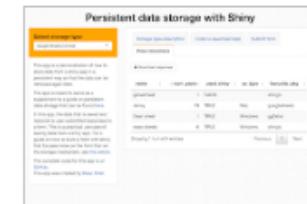
LOCATION TRACKER

Track locations over time with streaming data.



DOWNLOAD MONITOR

Streaming download rates visualized as a bubble chart.



PERSISTENT STORAGE

Save data from your apps to local files, servers, databases, and more.

<https://shiny.rstudio.com/gallery/>

bit.ly/dsbaRintro

Can we do more?

More
flexibility?

Interactivity?

Best in
class?

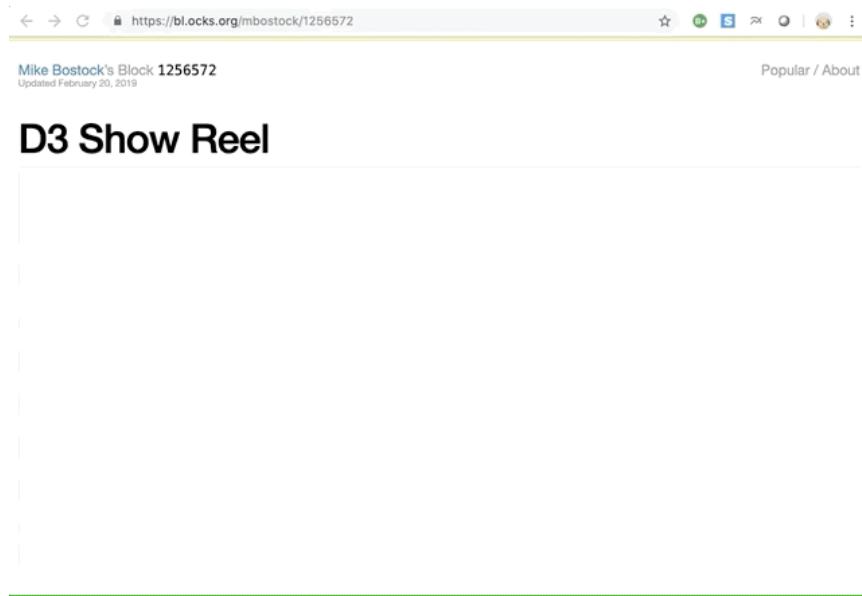
Can we do more?

More
flexibility?

Interactivity?

Best in
class?

Use D3 in R: r2d3.



<https://bl.ocks.org/mbostock/1256572>

bit.ly/dsbaRintro

r2d3



```
// baranims.js saved locally
var barHeight = Math.floor(height / data.length);

var bars = r2d3.svg.selectAll('rect')
  .data(r2d3.data);

bars.enter()
  .append('rect')
    .attr('width', function(d) { return d * width; })
    .attr('height', barHeight)
    .attr('y', function(d, i) { return i * barHeight; })
    .attr('fill', 'steelblue');

bars.exit().remove();

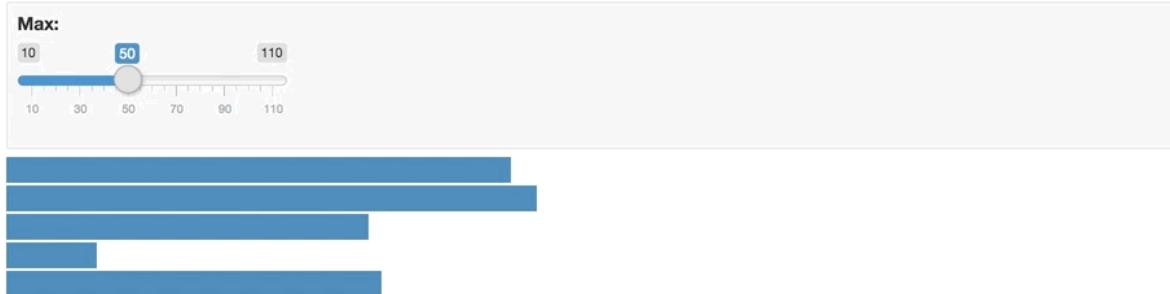
bars.transition()
  .duration(100)
  .attr("width", function(d) { return d * width; });
```

```
# shiny code with r2d3
library(shiny)
library(r2d3)

ui <- fluidPage(
  inputPanel(
    sliderInput(
      "bar_max", label = "Max:",
      min = 0.1, max = 1.0, value = 0.2, step = 0.1
    ),
    d3Output("d3")
  )

server <- function(input, output) {
  output$d3 <- renderD3(
    r2d3(
      runif(5, 0, input$bar_max),
      script = "baranims.js" # see left
    )
  )
}

shinyApp(ui = ui, server = server)
```



bit.ly/dsbaRintro

Takeaways

R/RStudio is not your old base R tools.

- Tidyverse, shiny, RMarkdown, etc.

Takeaways

R/RStudio is not your old base R tools.

- Tidyverse, shiny, RMarkdown, etc.

Choose the best tool for the job, maybe Excel or Tableau.

- But if its a data science pipeline + data visualizations: R is hard to compete with.

Takeaways

R/RStudio is not your old base R tools.

- Tidyverse, shiny, RMarkdown, etc.

Choose the best tool for the job, maybe Excel or Tableau.

- But if its a data science pipeline + data visualizations: R is hard to compete with.

A lot of times, don't need to choose R vs. Python, C++, JS.

- Take the best of each and make it into an R package.

Takeaways

R/RStudio is not your old base R tools.

- Tidyverse, shiny, RMarkdown, etc.

Choose the best tool for the job, maybe Excel or Tableau.

- But if its a data science pipeline + data visualizations: R is hard to compete with.

A lot of times, don't need to choose R vs. Python, C++, JS.

- Take the best of each and make it into an R package.

Data visualization is part art and science (and design!).

- The best way to learn: do it, get feedback, iterate.

Happy R programming!



Artwork by @allison_horst

bit.ly/dsbaRintro