Interactivity

Session 11

PMAP 8921: Data Visualization with R
Andrew Young School of Policy Studies
Spring 2025

Plan for today

Making interactive graphics

Sharing content

Making interactive graphics

Three general methods

Single plots with {plotly}

Easy!

Dashboards with {flexdashboard}

Slightly more complicated

Complete interactive apps with Shiny

Super complicated!

Single plots with plotly

Plotly is special software for creating interactive plots with JavaScript

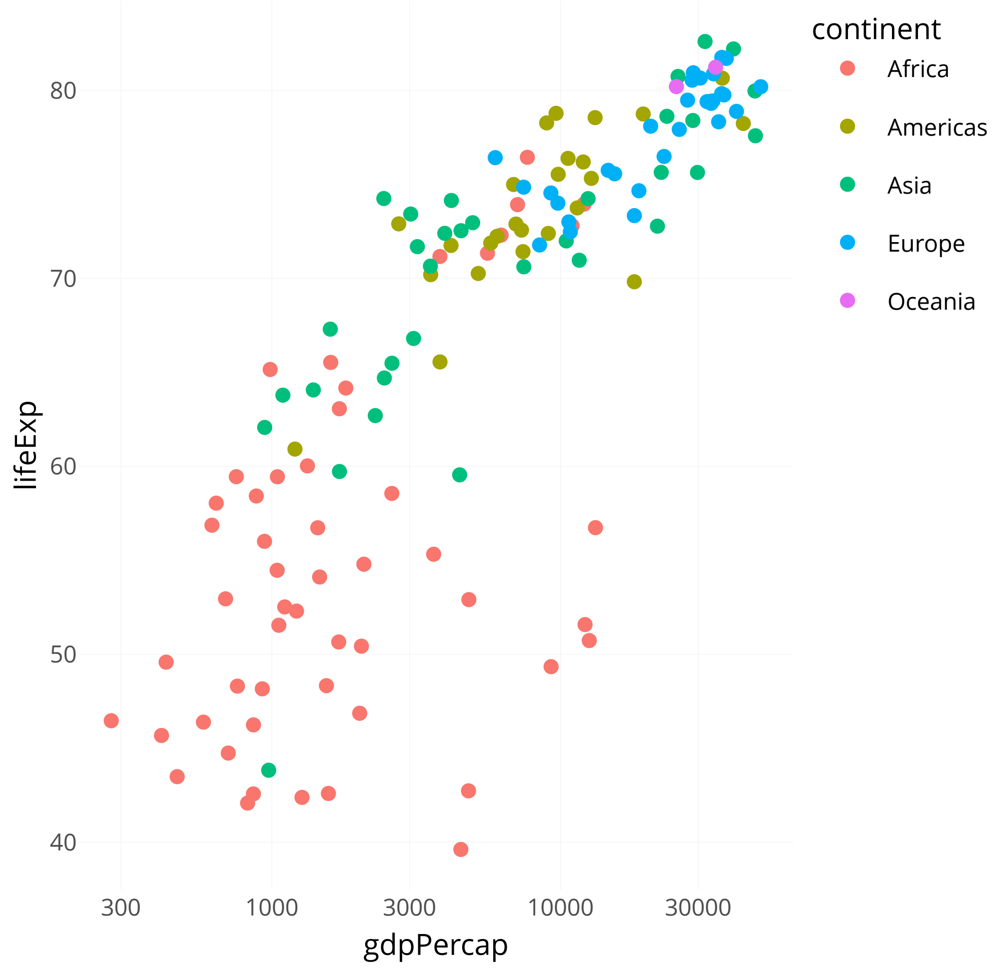
No knowledge of JavaScript needed!

`ggplotly()` in the {plotly} R package translates between R and Javascript for you!

Plotly

```r
library(gapminder)
library(plotly)

gapminder_2007 <- filter(gapminder,
                         year == 2007)

my_plot <- ggplot(
  data = gapminder_2007,
  mapping = aes(x = gdpPercap, y = lifeExp,
                color = continent)) +
  geom_point() +
  scale_x_log10() +
  theme_minimal()
```

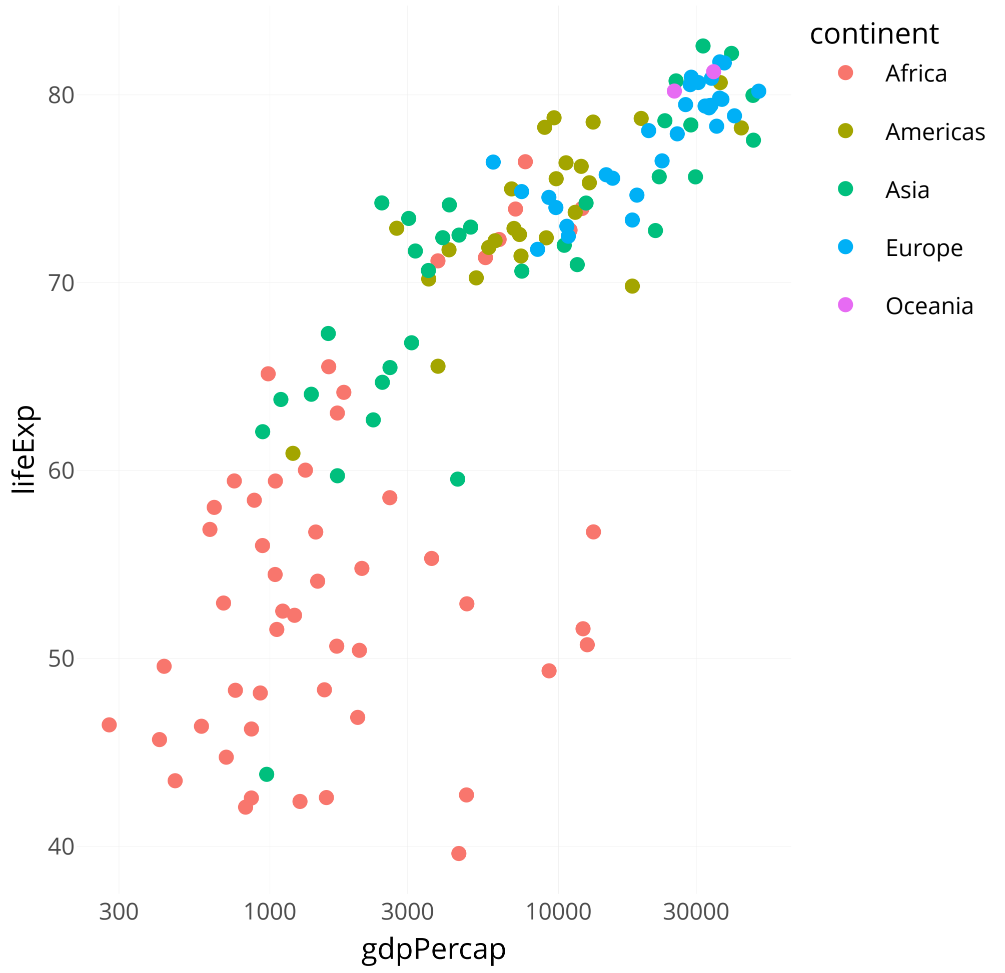
```r
ggplotly(my_plot)
```



Plotly tooltips

```r
my_plot <- ggplot(
  data = gapminder_2007,
  mapping = aes(x = gdpPercap, y = lifeExp,
                color = continent)) +
  geom_point(aes(text = country)) +
  scale_x_log10() +
  theme_minimal()
```

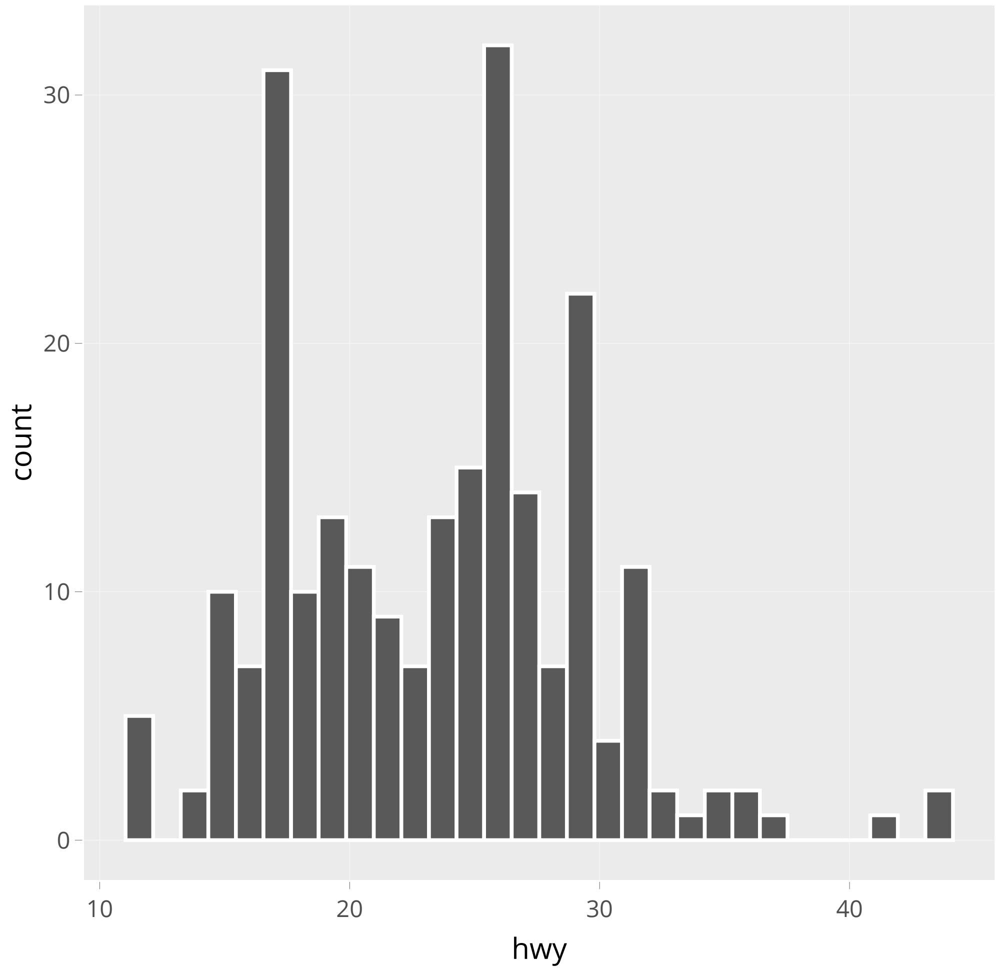
```r
interactive_plot <- ggplotly(
  my_plot, tooltip = "text"
)
interactive_plot
```



Works with most geoms!

```
car_hist <- ggplot(mpg,
                   aes(x = hwy)) +
  geom_histogram(binwdith = 2,
                 boundary = 0,
                 color = "white")
```

```
ggplotly(car_hist)
```



Save as HTML

Save a self-contained HTML version of it with `saveWidget()` in the {htmlwidgets} R package

```r
# This is like ggsave, but for interactive HTML plots
htmlwidgets::saveWidget(interactive_plot, "fancy_plot.html")
```

Fully documented

The documentation for ggplot2 + plotly is full of examples of how to customize everything

Rely on that ↑ + Google to make really fancy (and easy!) interactive plots

Three general methods

Single plots with {plotly}

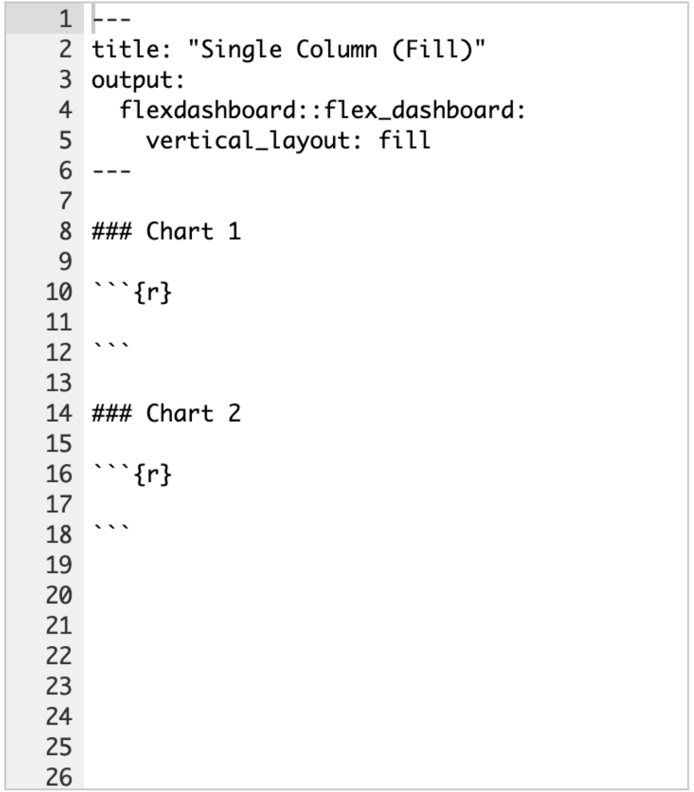
Easy!

Dashboards with {flexdashboard}

Slightly more complicated

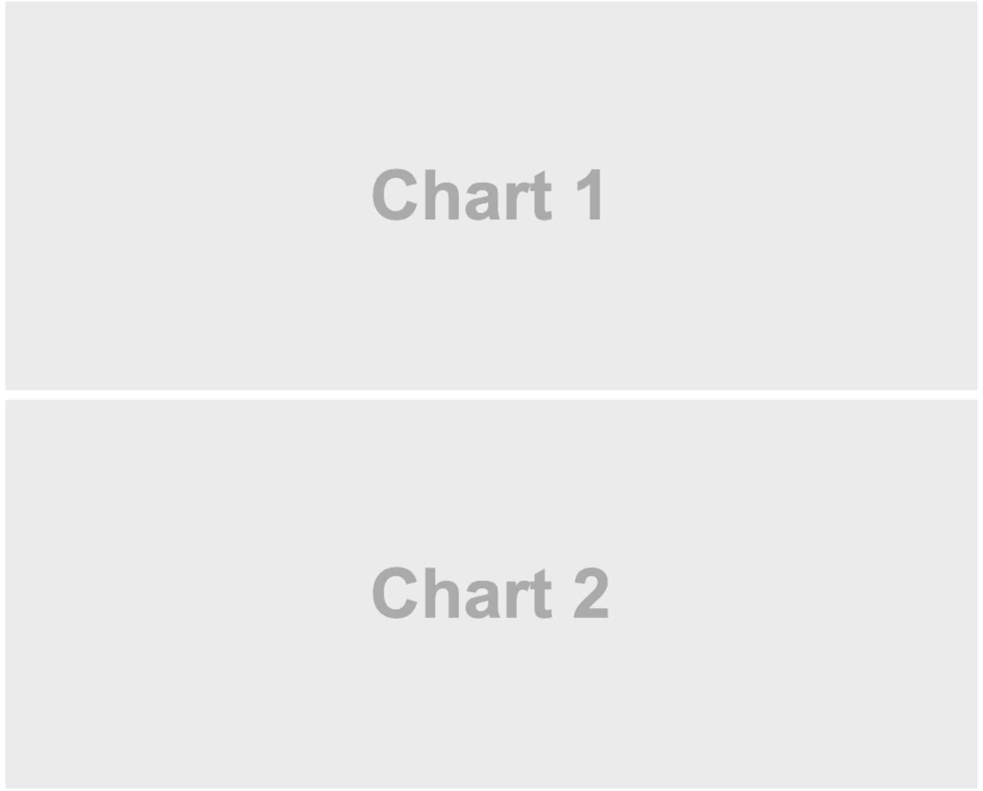
Dashboards with {flexdashboard}

Use basic R Markdown to build a dashboard!



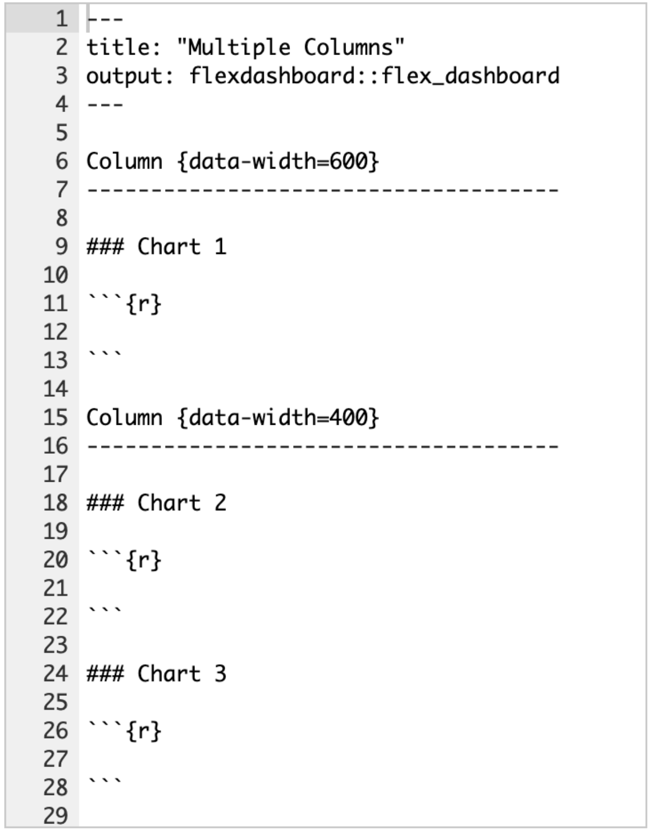
~~~
---
title: "Single Column (Fill)"
output:
  flexdashboard::flex_dashboard:
    vertical_layout: fill
---

### Chart 1

```{r}

~~~

Chart 2

```{r}

```
```


```

# Dashboards with {flexdashboard}

## Make any kind of block arrangement



~~~
---
title: "Multiple Columns"
output: flexdashboard::flex_dashboard
---

Column {data-width=600}
-------------------------------------

### Chart 1

```{r}

~~~

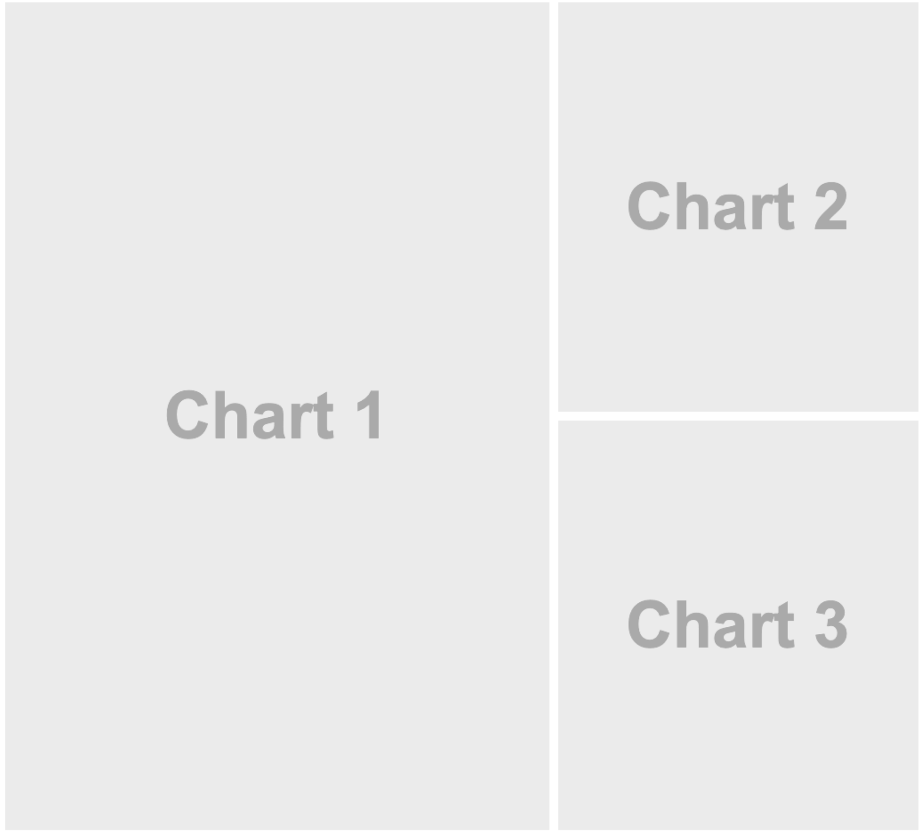
Column {data-width=400}
-------------------------------------

### Chart 2

```{r}

```

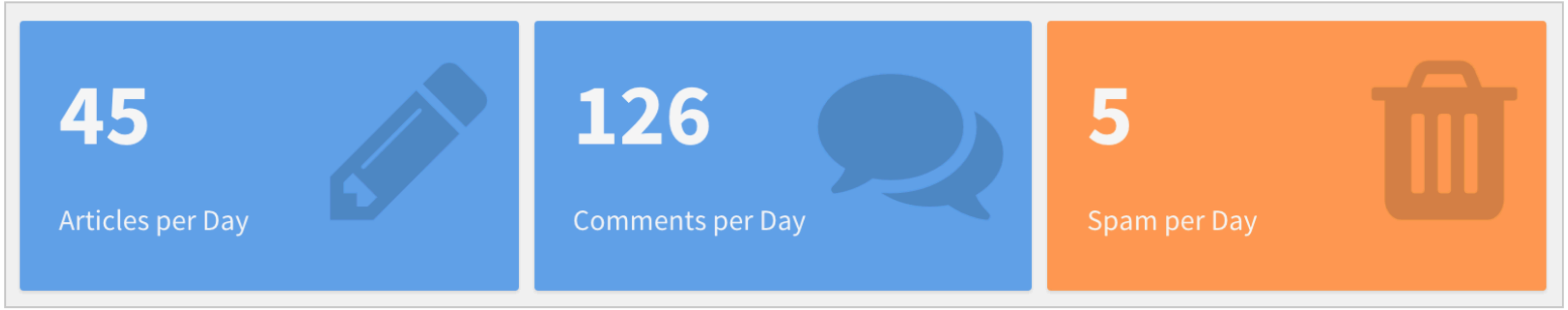
Chart 3

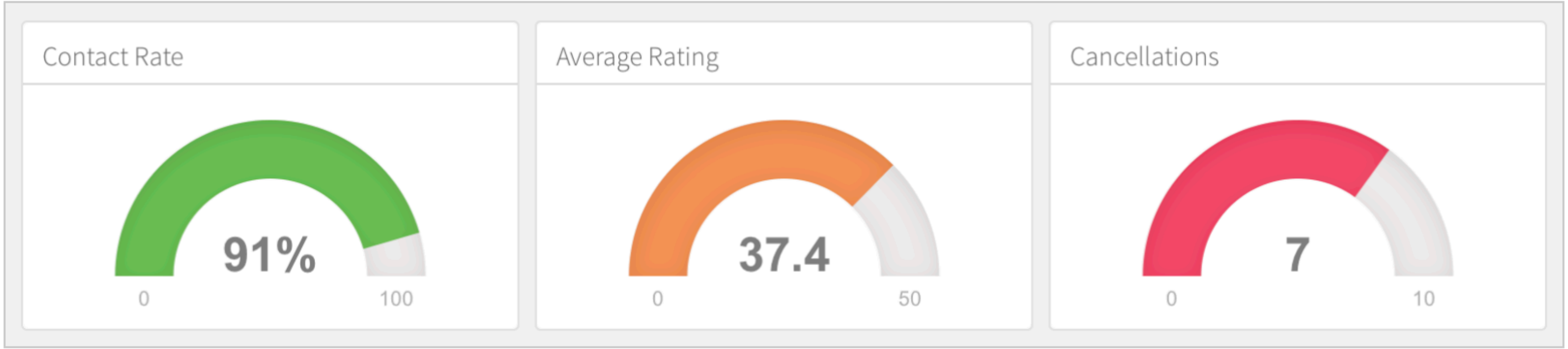
```{r}

```

```


```

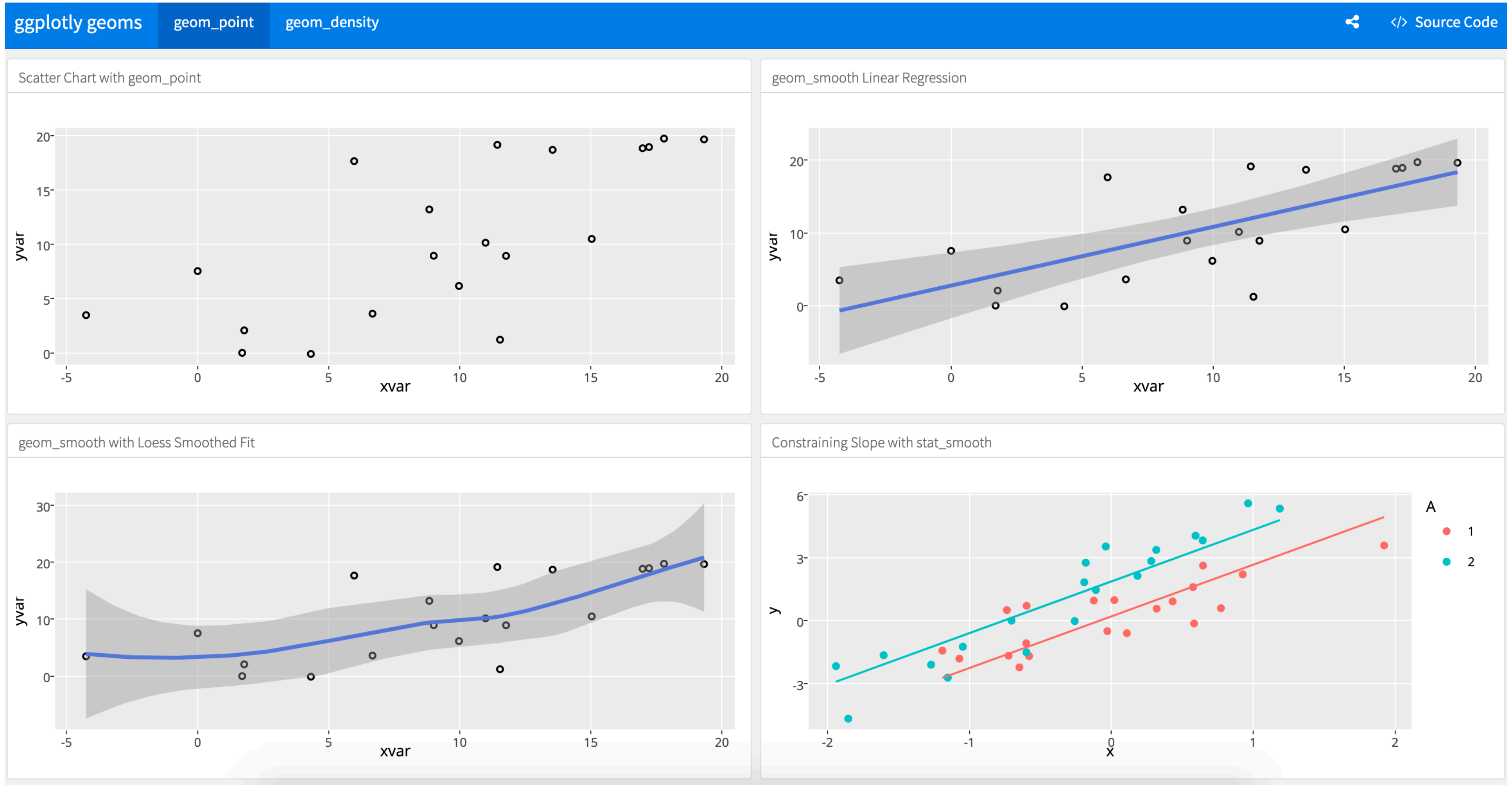
Dashboards with {flexdashboard}

Add other elements like text and gauges



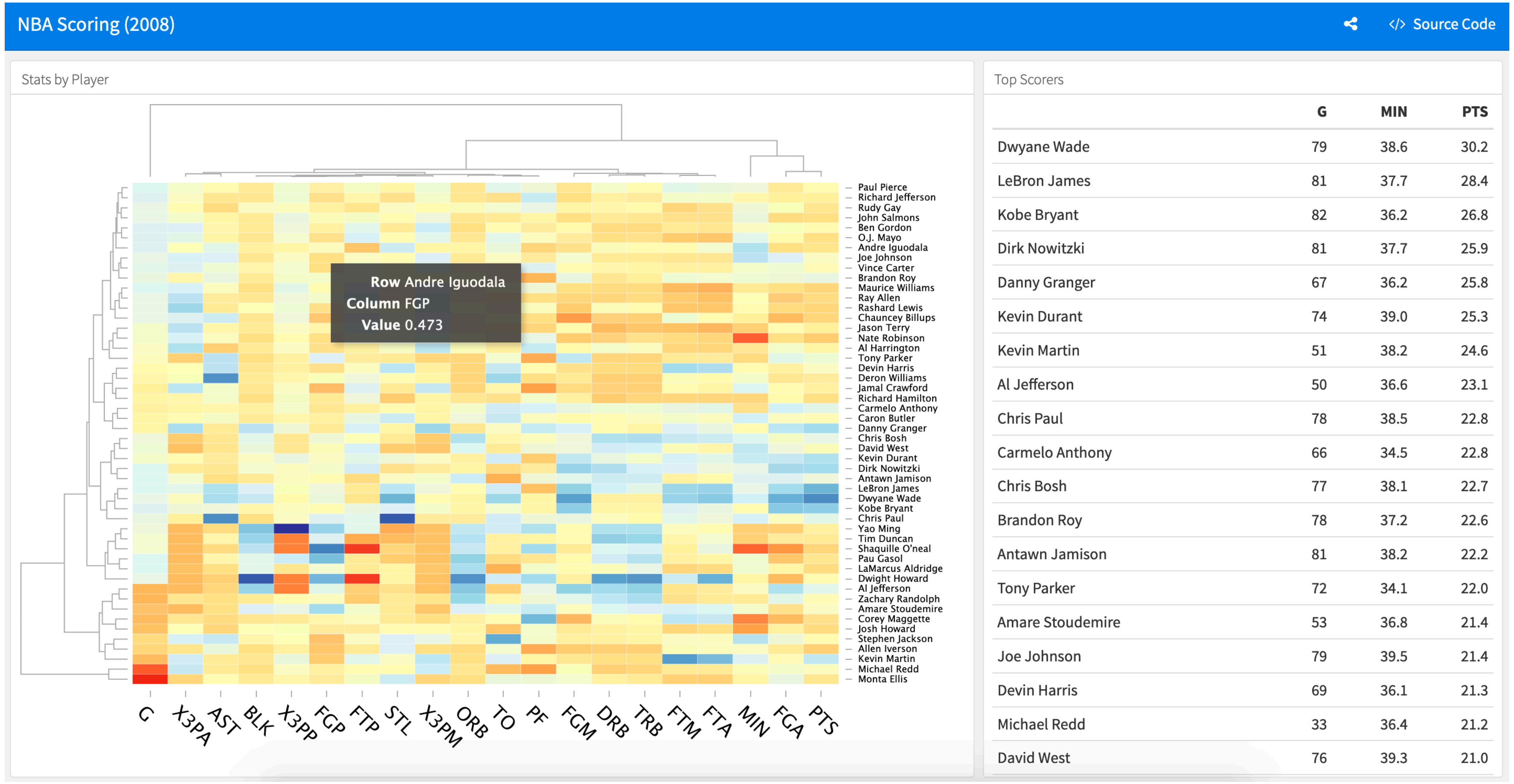


Example dashboards


ggplotly geoms
geom_point
geom_density
Source Code
Scatter Chart with geom_point
geom_smooth Linear Regression
geom_smooth with Loess Smoothed Fit
Constraining Slope with stat_smooth
xvar
yvar
x
y
A
1
2

ggplot2 geoms

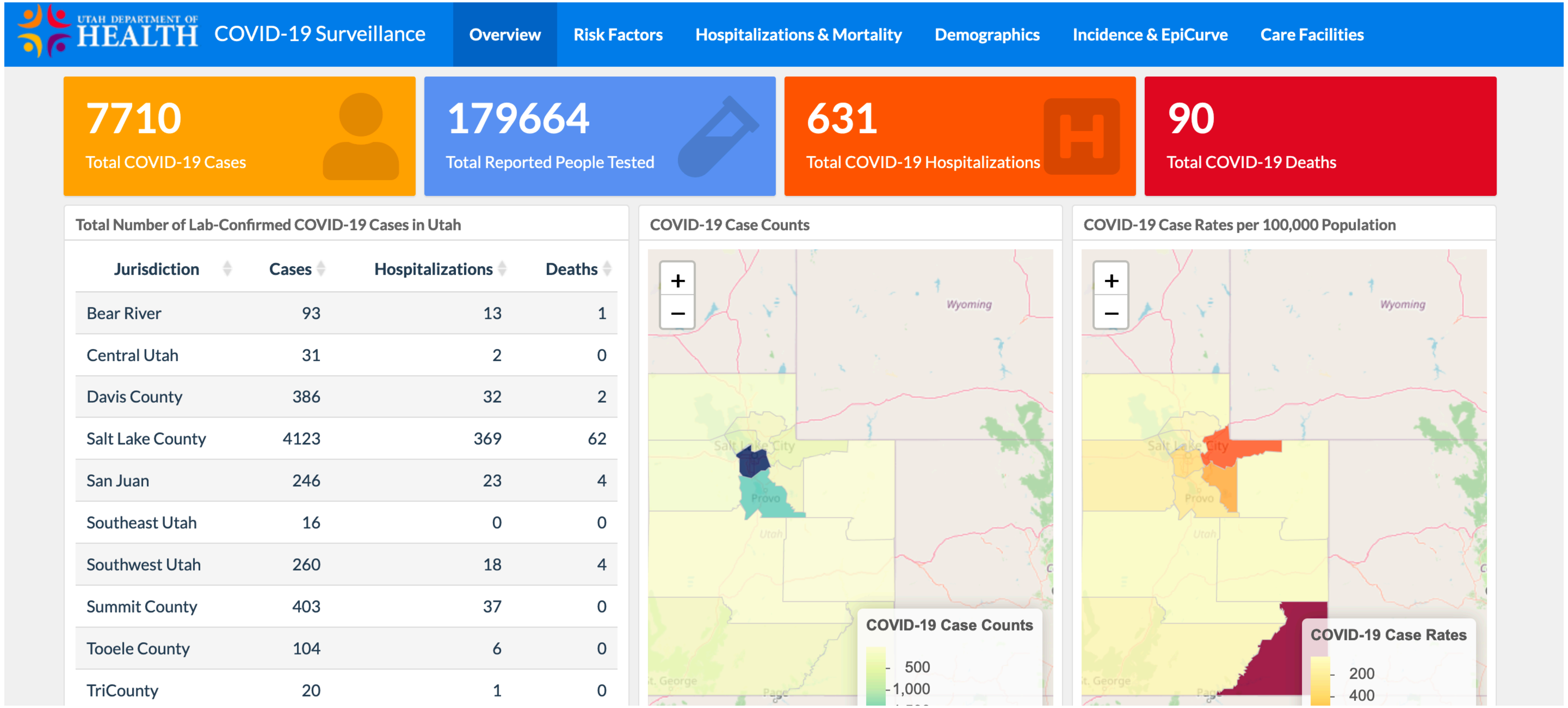
Example dashboards



	G	MIN	PTS
Dwyane Wade	79	38.6	30.2
LeBron James	81	37.7	28.4
Kobe Bryant	82	36.2	26.8
Dirk Nowitzki	81	37.7	25.9
Danny Granger	67	36.2	25.8
Kevin Durant	74	39.0	25.3
Kevin Martin	51	38.2	24.6
Al Jefferson	50	36.6	23.1
Chris Paul	78	38.5	22.8
Carmelo Anthony	66	34.5	22.8
Chris Bosh	77	38.1	22.7
Brandon Roy	78	37.2	22.6
Antawn Jamison	81	38.2	22.2
Tony Parker	72	34.1	22.0
Amare Stoudemire	53	36.8	21.4
Joe Johnson	79	39.5	21.4
Devin Harris	69	36.1	21.3
Michael Redd	33	36.4	21.2
David West	76	39.3	21.0

NBA scoring

Example dashboards



UTAH DEPARTMENT OF HEALTH COVID-19 Surveillance

Overview | Risk Factors | Hospitalizations & Mortality | Demographics | Incidence & EpiCurve | Care Facilities

7710 Total COVID-19 Cases

179664 Total Reported People Tested

631 Total COVID-19 Hospitalizations

90 Total COVID-19 Deaths

Total Number of Lab-Confirmed COVID-19 Cases in Utah

Jurisdiction	Cases	Hospitalizations	Deaths
Bear River	93	13	1
Central Utah	31	2	0
Davis County	386	32	2
Salt Lake County	4123	369	62
San Juan	246	23	4
Southeast Utah	16	0	0
Southwest Utah	260	18	4
Summit County	403	37	0
Tooele County	104	6	0
TriCounty	20	1	0

COVID-19 Case Counts

COVID-19 Case Rates per 100,000 Population

Utah's COVID-19 dashboard

Outstanding documentation

The **documentation** for {flexdashboard} is
full of examples and details of everything you can do

Rely on that ↑ + Google to make
really fancy (and easy!) dashboards!

Three general methods

Single plots with {plotly}

Easy!

Dashboards with {flexdashboard}

Slightly more complicated

Complete interactive apps with Shiny

Super complicated!

Shiny

Shiny is a complete web application framework for interactive statistics

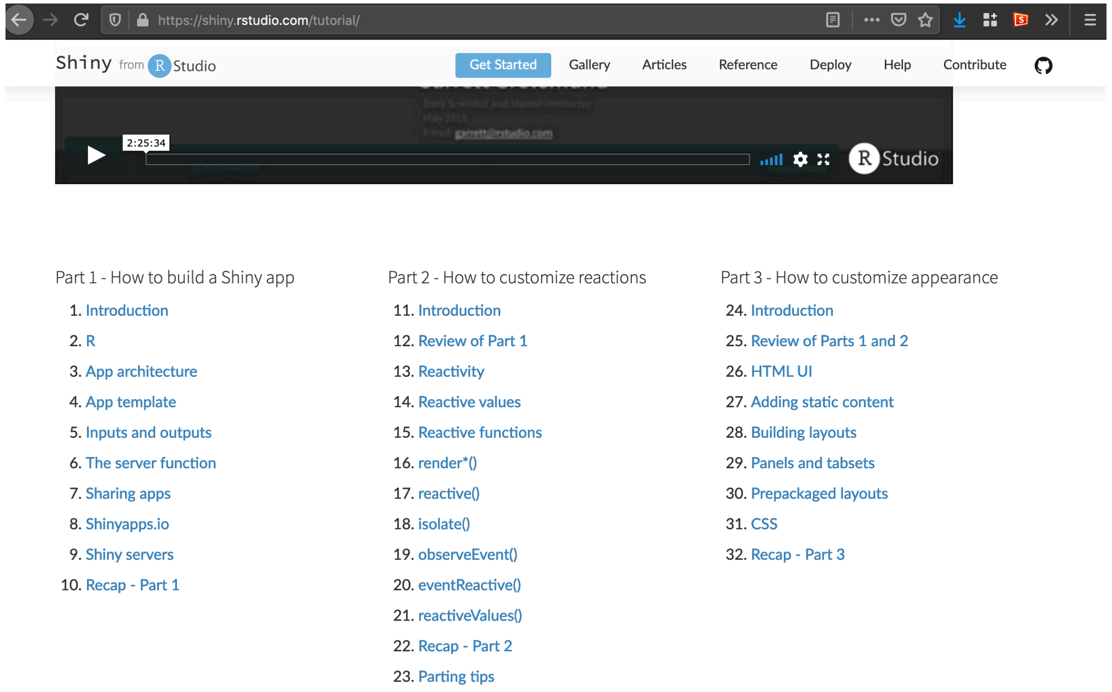
It's super complex and hard for beginners

I've never made a standalone Shiny app!

(And I don't plan on trying anytime soon)

Lots of resources to help start

RStudio has a whole website for helping you get started



Part 1 - How to build a Shiny app

1. Introduction
2. R
3. App architecture
4. App template
5. Inputs and outputs
6. The server function
7. Sharing apps
8. Shinyapps.io
9. Shiny servers
10. Recap - Part 1

Part 2 - How to customize reactions

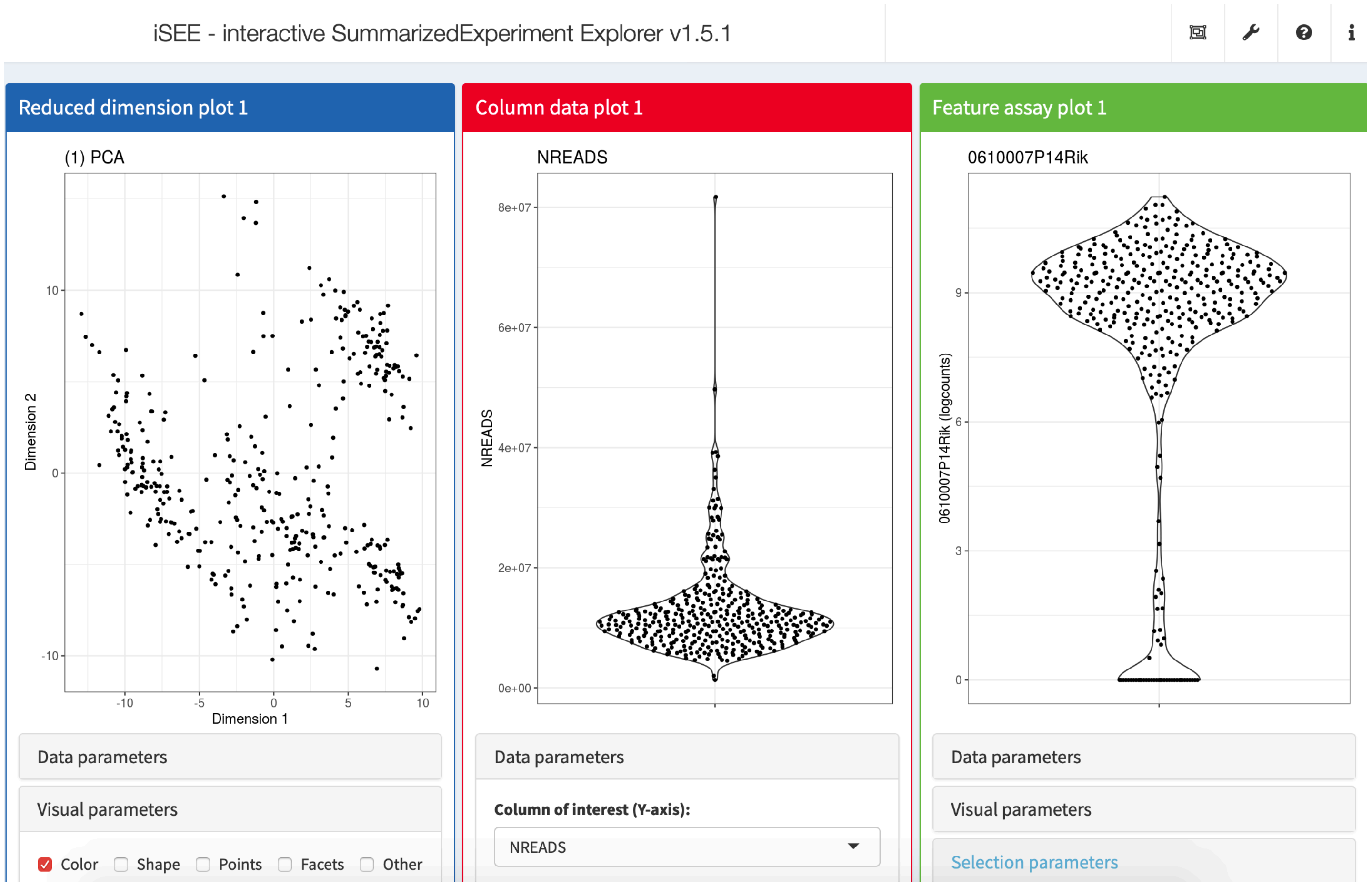
11. Introduction
12. Review of Part 1
13. Reactivity
14. Reactive values
15. Reactive functions
16. render*()
17. reactive()
18. isolate()
19. observeEvent()
20. eventReactive()
21. reactiveValues()
22. Recap - Part 2
23. Parting tips

Part 3 - How to customize appearance

24. Introduction
25. Review of Parts 1 and 2
26. HTML UI
27. Adding static content
28. Building layouts
29. Panels and tabsets
30. Prepackaged layouts
31. CSS
32. Recap - Part 3

Getting started with Shiny

Really neat examples!

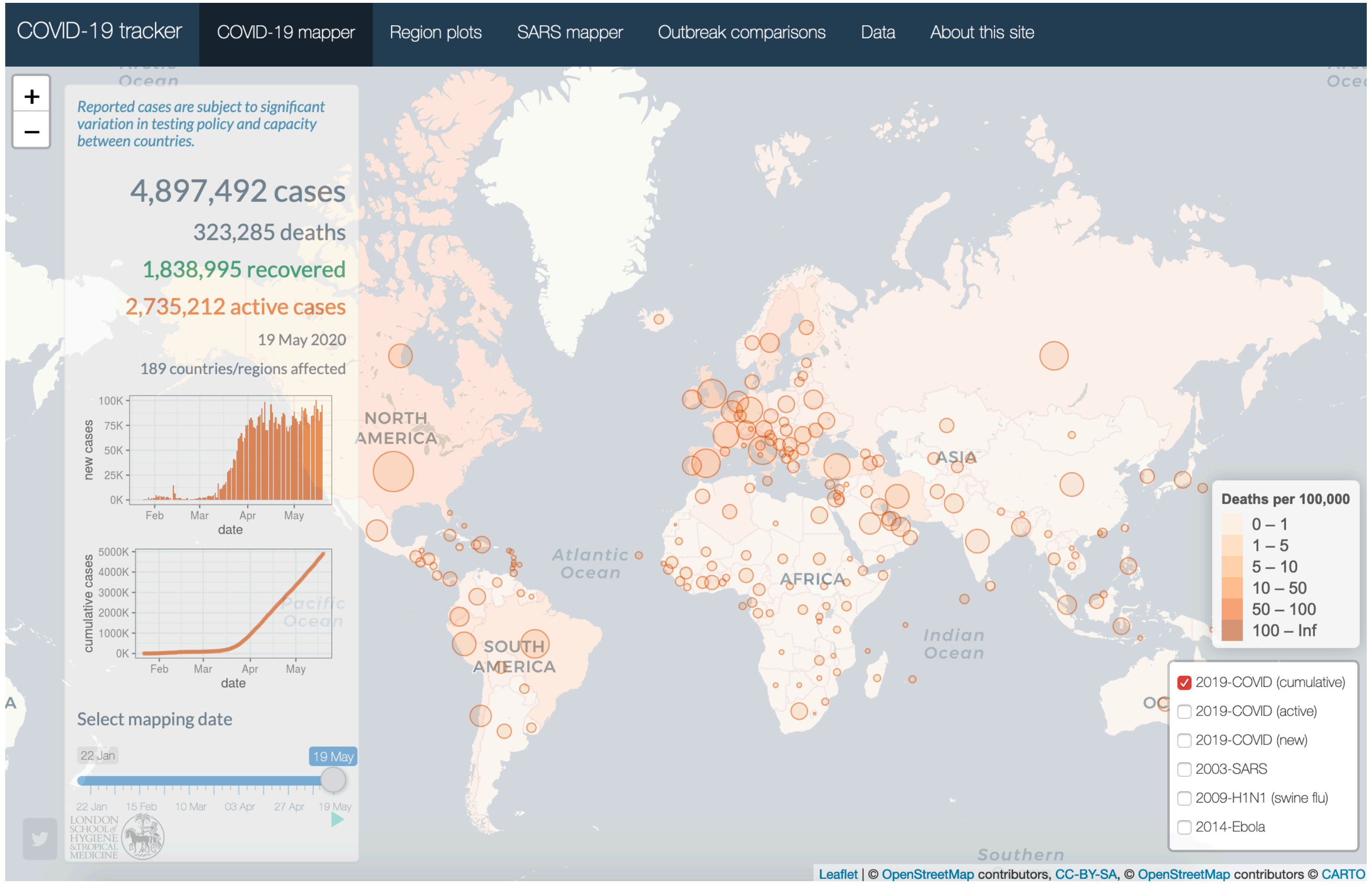


Column of interest (Y-axis): NREADS

[✓ Color] Shape Points Facets Other

iSEE (interactive SummarizedExperiment Explorer)

Really neat examples!


COVID-19 tracker | COVID-19 mapper | Region plots | SARS mapper | Outbreak comparisons | Data | About this site

Reported cases are subject to significant variation in testing policy and capacity between countries.

4,897,492 cases

323,285 deaths

1,838,995 recovered

2,735,212 active cases

19 May 2020

189 countries/regions affected

Select mapping date

22 Jan — 19 May

Deaths per 100,000

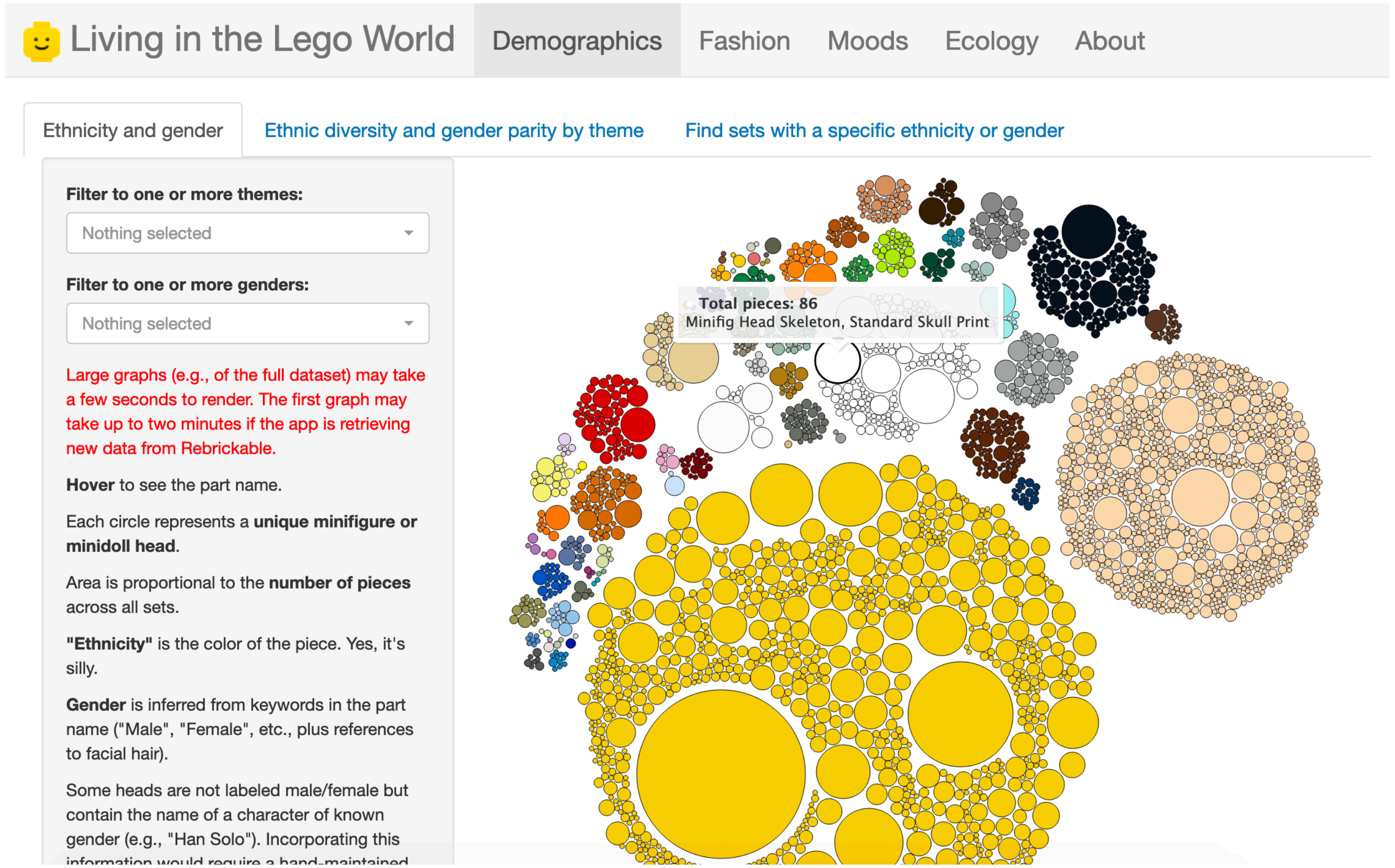
- 0 – 1
- 1 – 5
- 5 – 10
- 10 – 50
- 50 – 100
- 100 – Inf

- ☑ 2019-COVID (cumulative)
- ☐ 2019-COVID (active)
- ☐ 2019-COVID (new)
- ☐ 2003-SARS
- ☐ 2009-H1N1 (swine flu)
- ☐ 2014-Ebola

Leaflet | © OpenStreetMap contributors, CC-BY-SA, © OpenStreetMap contributors © CARTO

COVID-19 tracker

Really neat examples!


Living in the Lego World
Demographics
Fashion
Moods
Ecology
About
Ethnicity and gender
Ethnic diversity and gender parity by theme
Find sets with a specific ethnicity or gender
Total pieces: 86
Minifig Head Skeleton, Standard Skull Print

Filter to one or more themes:

Nothing selected

Filter to one or more genders:

Nothing selected

Large graphs (e.g., of the full dataset) may take a few seconds to render. The first graph may take up to two minutes if the app is retrieving new data from Rebrickable.

Hover to see the part name.

Each circle represents a **unique minifigure or minidoll head**.

Area is proportional to the **number of pieces** across all sets.

"Ethnicity" is the color of the piece. Yes, it's silly.

Gender is inferred from keywords in the part name ("Male", "Female", etc., plus references to facial hair).

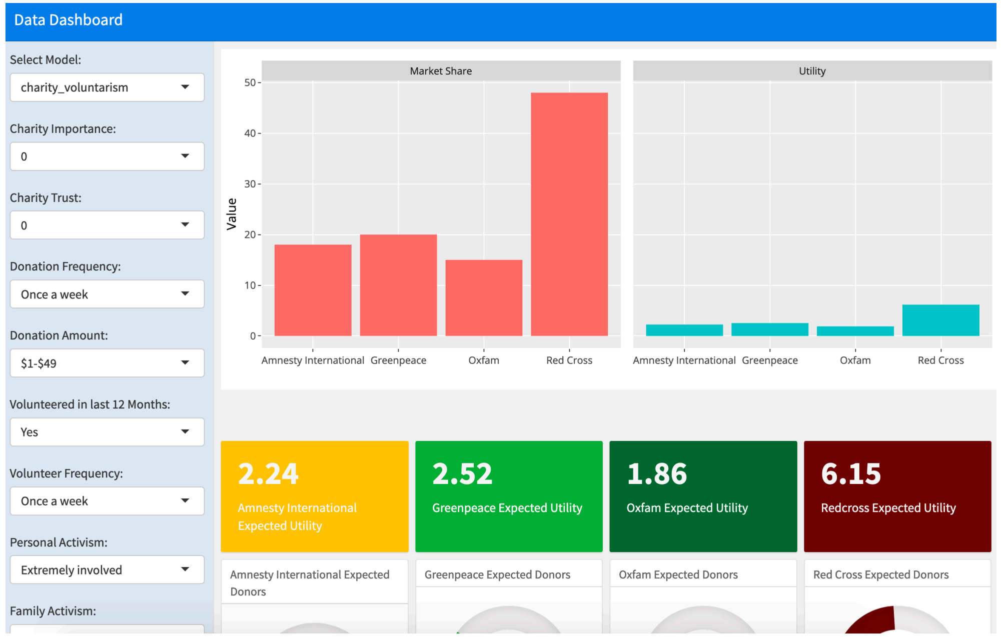
Some heads are not labeled male/female but contain the name of a character of known gender (e.g., "Han Solo"). Incorporating this information would require a hand-maintained

Living in the LEGO world

flexdashboard + Shiny

You can use reactive Shiny things in flexdashboards without building a complete Shiny app!

I *have* done this


Data Dashboard

Select Model: charity_voluntarism

Charity Importance: 0

Charity Trust: 0

Donation Frequency: Once a week

Donation Amount: $1-$49

Volunteered in last 12 Months: Yes

Volunteer Frequency: Once a week

Personal Activism: Extremely involved

Family Activism:

2.24 Amnesty International Expected Utility

2.52 Greenpeace Expected Utility

1.86 Oxfam Expected Utility

6.15 Redcross Expected Utility

Amnesty International Expected Donors

Greenpeace Expected Donors

Oxfam Expected Donors

Red Cross Expected Donors

Sharing content

What do you do after you knit?

When knitting to PDF or Word, you make a standalone file

E-mail it, message it, Slack it, whatever

When knitting to HTML, you make a website

By default it's a standalone `.html` file with graphics embedded,
so you can still e-mail it, etc., but it can get huge if there are lots of images

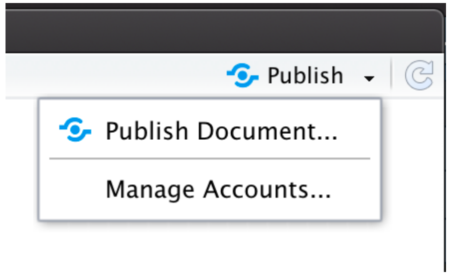
Standalone files won't work well if there's anything interactive

You can also post it online!

Places to put HTML documents

RPubs for knitted HTML documents

Built in to RStudio; works with ggplotly!



RPubs or shinyapps.io for flexdashboards

Your own web server for anything, if you have one