# Interactivity

#### **Session 10**

PMAP 8921: Data Visualization with R Andrew Young School of Policy Studies Summer 2021

## **Plan for today**

#### Making interactive graphics

#### **Sharing content**

# Making interactive graphics

### Three general methods

#### Single plots with **plotly**



#### Dashboards with **flexdashboard**

Slightly more complicated

Complete interactive apps with Shiny

Super complicated!

# Single plots with plotly

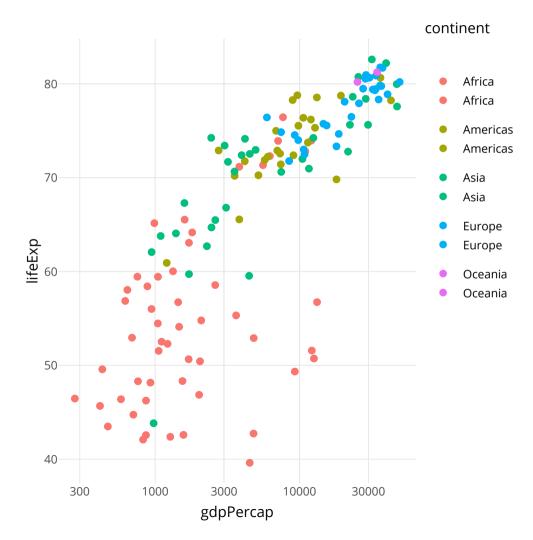
**Plotly** is special software for creating interactive plots with JavaScript

No knowledge of JavaScript needed!

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

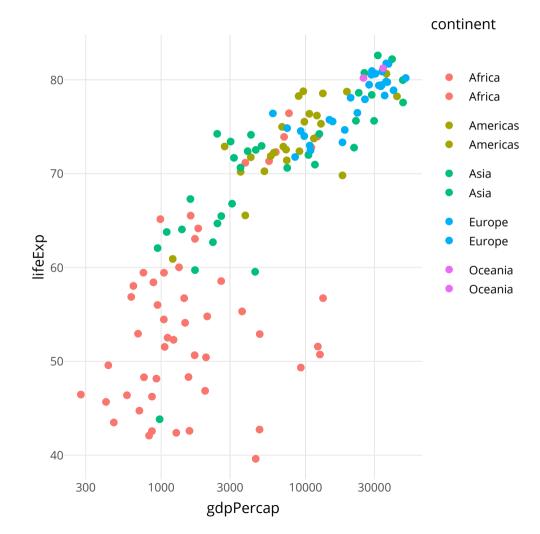


ggplotly(my\_plot)

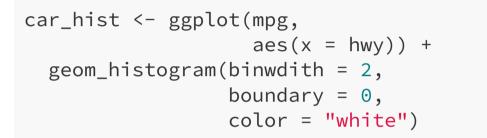


# **Plotly tooltips**

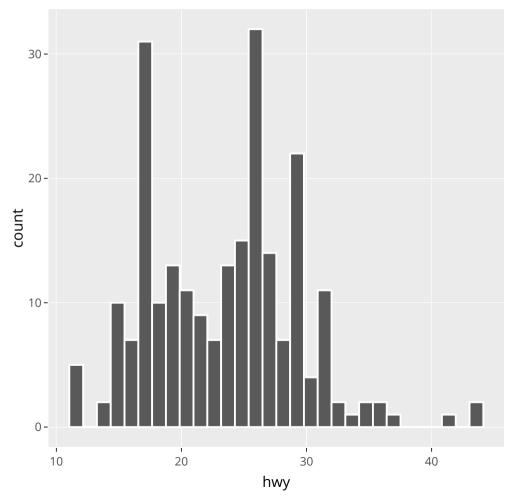
```
interactive_plot <- ggplotly(
   my_plot, tooltip = "text"
)
interactive_plot</pre>
```



### Works with most geoms!



ggplotly(car\_hist)





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

# 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**

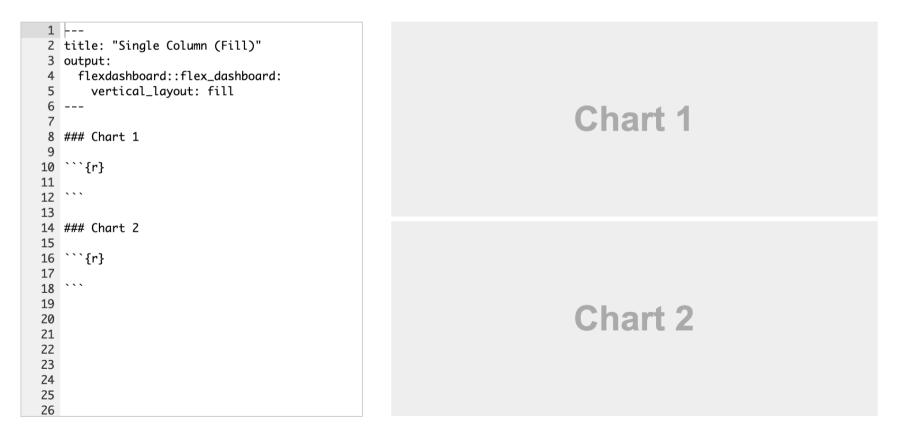


#### Dashboards with flexdashboard

Slightly more complicated

### Dashboards with flexdashboard

#### Use basic R Markdown to build a dashboard!



#### Dashboards with flexdashboard

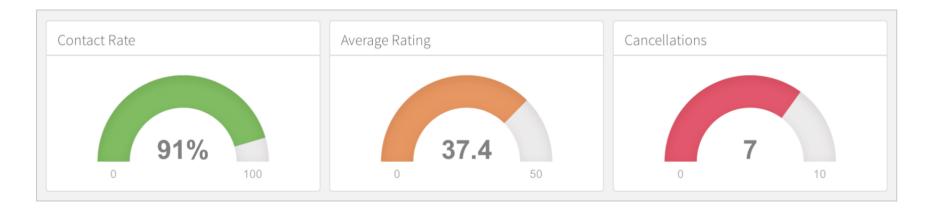
#### Make any kind of block arrangement

| <pre>1 2 title: "Multiple Columns" 3 output: flexdashboard::flex_dashboard 4 5 6 Column {data-width=600} 7 8 9 ### Chart 1 10 11 ```{r} 12 13 ``` 14 15 Column [data width 400]</pre> | Chart 1 | Chart 2 |
|---|---------|---------|
| <pre>15 Column {data-width=400} 16 17 18 ### Chart 2 19 20 ```{r} 21 22 ``` 23 24 ### Chart 3 25 26 ```{r} 27 28 ``` 29</pre>   |         | Chart 3 |

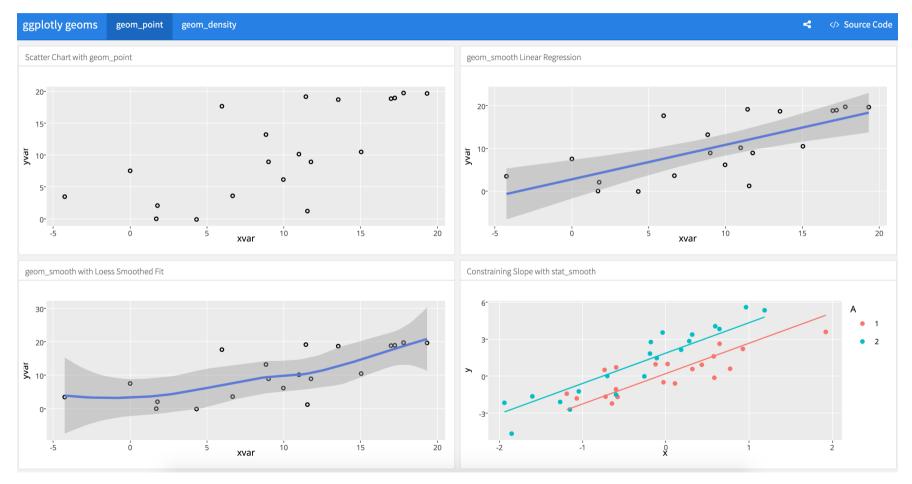
### Dashboards with flexdashboard

#### Add other elements like text and gauges





# Example dashboards



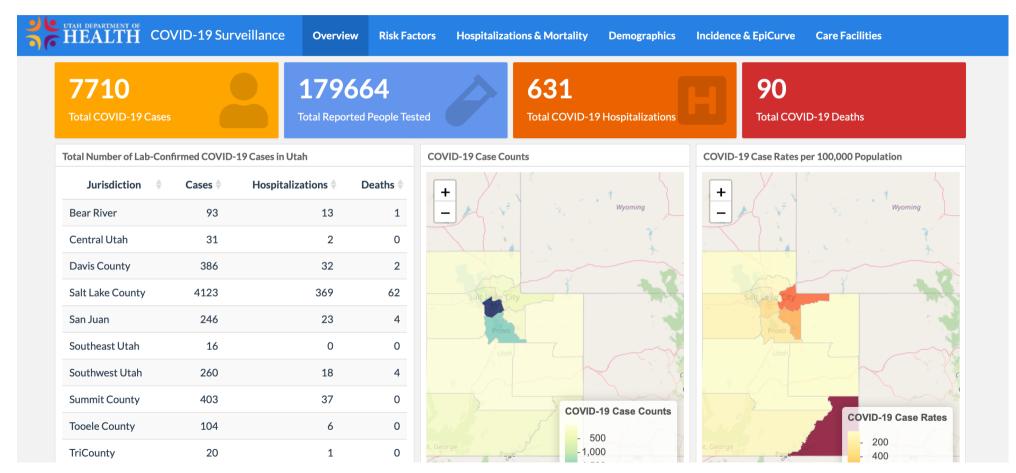
ggplot2 geoms

### **Example dashboards**

| NBA Scoring (2008)  |                  | 4  | > Sou | rce Code |
|---|------------------|----|-------|----------|
| Stats by Player   | Top Scorers      |    |       |          |
|   |                  | G  | MIN   | PTS      |
|   | Dwyane Wade      | 79 | 38.6  | 30.2     |
| - Paul Pierce     - Richard Jefferson   | LeBron James     | 81 | 37.7  | 28.4     |
| Construction     C         | Kobe Bryant      | 82 | 36.2  | 26.8     |
|   | Dirk Nowitzki    | 81 | 37.7  | 25.9     |
| Row Andre Iguodala     - Vince Carter       Column FGP     - RayAllen       Value 0.473     - Charter Willings  | Danny Granger    | 67 | 36.2  | 25.8     |
|   | Kevin Durant     | 74 | 39.0  | 25.3     |
| - Al Harrington - Al Harringto    | Kevin Martin     | 51 | 38.2  | 24.6     |
|   | Al Jefferson     | 50 | 36.6  | 23.1     |
| - Carmelo Anthony<br>- Carmelo Anthony<br>- Carmelo Anthony<br>- Danno Butler<br>- Danny Granger  | Chris Paul       | 78 | 38.5  | 22.8     |
| - Chris Bosh<br>- David West<br>- Kevin Durant  | Carmelo Anthony  | 66 | 34.5  | 22.8     |
| - Antwarianison<br>- LeBron James<br>- Dwyare<br>- Kobe Bryant<br>- Chris Paul  | Chris Bosh       | 77 | 38.1  | 22.7     |
| - Yao Ming  | Brandon Roy      | 78 | 37.2  | 22.6     |
| - Tim Duncan<br>- Shaquille O'neal<br>- Pau Gasol<br>- Lawrence Addridee  | Antawn Jamison   | 81 | 38.2  | 22.2     |
| - LiMarcus Adridge<br>- Diviging and a second s | Tony Parker      | 72 | 34.1  | 22.0     |
| <ul> <li>Amare Stoudemire</li> <li>Corey Magette</li> <li>Josh Howard</li> <li>Stouber Jackson</li> </ul>   | Amare Stoudemire | 53 | 36.8  | 21.4     |
| L L L Allen Verson - Allen Verson - Kevin Martin - Kickel Red   | Joe Johnson      | 79 | 39.5  | 21.4     |
| - Monta Ellis   | Devin Harris     | 69 | 36.1  | 21.3     |
| $c t_{s_{q}} t_{y} t_{s_{q}} c_{s} $    | Michael Redd     | 33 | 36.4  | 21.2     |
| ,   | David West       | 76 | 39.3  | 21.0     |

NBA scoring

#### **Example dashboards**



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**



#### Dashboards with **flexdashboard**

Slightly more complicated

Complete interactive apps with Shiny

Super complicated!



Shiny is a complete web application framework for interactive statistics

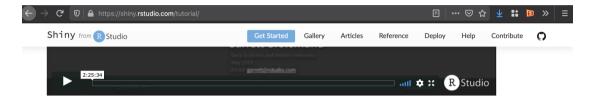
It's super complex and hard for beginners

I've never made a standalone Shiny app!

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

### Lots of resources to help start

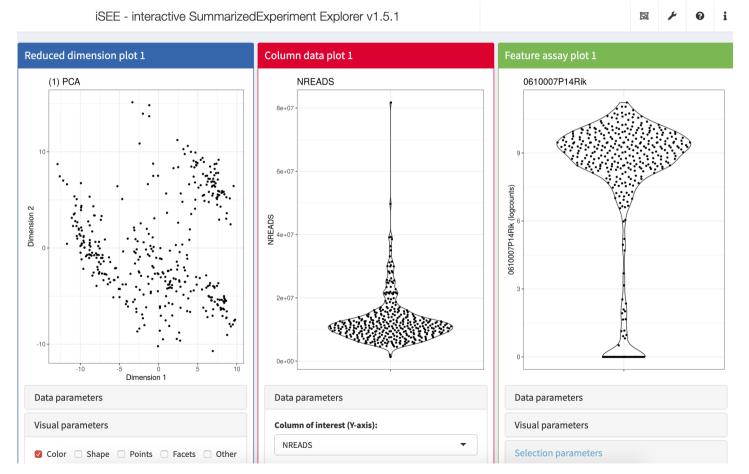
#### RStudio has a whole website for helping you get started



| Part 1 - How to build a Shiny app | Part 2 - How to customize reactions | Part 3 - How to customize appearance |
|-----------------------------------|-------------------------------------|--------------------------------------|
| 1. Introduction                   | 11. Introduction                    | 24. Introduction                     |
| 2. R                              | 12. Review of Part 1                | 25. Review of Parts 1 and 2          |
| 3. App architecture               | 13. Reactivity                      | 26. HTML UI                          |
| 4. App template                   | 14. Reactive values                 | 27. Adding static content            |
| 5. Inputs and outputs             | 15. Reactive functions              | 28. Building layouts                 |
| 6. The server function            | 16. render*()                       | 29. Panels and tabsets               |
| 7. Sharing apps                   | 17. reactive()                      | 30. Prepackaged layouts              |
| 8. Shinyapps.io                   | 18. isolate()                       | 31. CSS                              |
| 9. Shiny servers                  | 19. observeEvent()                  | 32. Recap - Part 3                   |
| 10. Recap - Part 1                | 20. eventReactive()                 |                                      |
|                                   | 21. reactiveValues()                |                                      |
|                                   | 22. Recap - Part 2                  |                                      |
|                                   | 23. Parting tips                    |                                      |

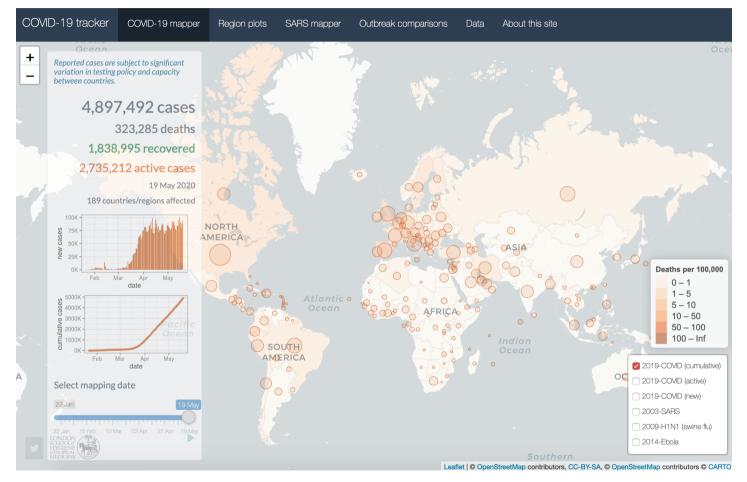
Getting started with Shiny

### **Really neat examples!**



iSEE (interactive SummarizedExperiment Explorer)

### **Really neat examples!**



COVID-19 tracker

### **Really neat examples!**

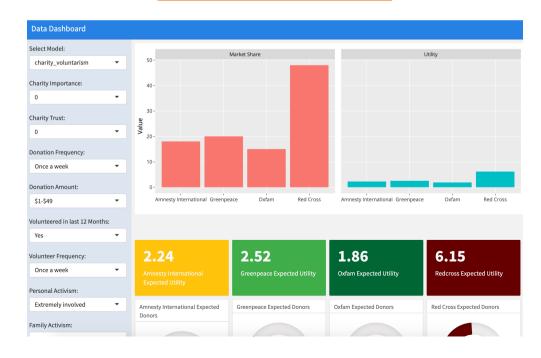
| Living in the Lego World  | Demographics        | Fashion                                | Moods                  | Ecology         | About |
|---|---------------------|--|------------------------|-----------------|-------|
| Ethnicity and gender Ethnic diversity and gene  | der parity by theme | Find sets with                         | a specific eth         | nicity or gende | r     |
| Filter to one or more themes:   |                     |  |                        |                 |       |
| Nothing selected -  |                     |  |                        |                 |       |
| Filter to one or more genders:  |                     |  | <b>₩</b>               |                 |       |
| Nothing selected -  | 8                   | Total pieces: 86<br>Minifig Head Skele | 6<br>eton, Standard Sl | kull Print      |       |
| <ul> <li>Large graphs (e.g., of the full dataset) may take<br/>a few seconds to render. The first graph may<br/>take up to two minutes if the app is retrieving<br/>new data from Rebrickable.</li> <li>Hover to see the part name.</li> <li>Each circle represents a unique minifigure or<br/>minidoll head.</li> <li>Area is proportional to the number of pieces<br/>across all sets.</li> </ul> |                     |  |                        |                 |       |
| <b>"Ethnicity"</b> is the color of the piece. Yes, it's silly.  |                     |  |                        |                 |       |
| <b>Gender</b> is inferred from keywords in the part<br>name ("Male", "Female", etc., plus references<br>to facial hair).  |                     |  |                        |                 |       |
| Some heads are not labeled male/female but<br>contain the name of a character of known<br>gender (e.g., "Han Solo"). Incorporating this   |                     |  |                        |                 |       |

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



# 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

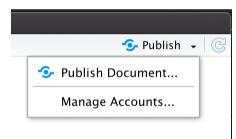
Standalone files won't work well if there's anything interactive

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