

Linking HTML Widgets with Crosstalk

JSM 2016

Joe Cheng

joe@rstudio.com

@jcheng

Background

- **R:** Great for data, statistics, (static) data viz, and reporting
- **JavaScript:** Great for interactive data viz, easy to deploy
- **HTML Widgets:** Wrap JavaScript viz libraries to make them feel “native” to R—as easy as plotting

	Native	Web
Building apps	JGR, gWidgets	Shiny
Authoring documents	Sweave, knitr	R Markdown/knitr
Grammar of graphics	ggplot2	ggvis, plotly
Custom plotting	Base graphics, grid, lattice	htmlwidgets
Coordinated multiple views	GGobi, cranvas	?

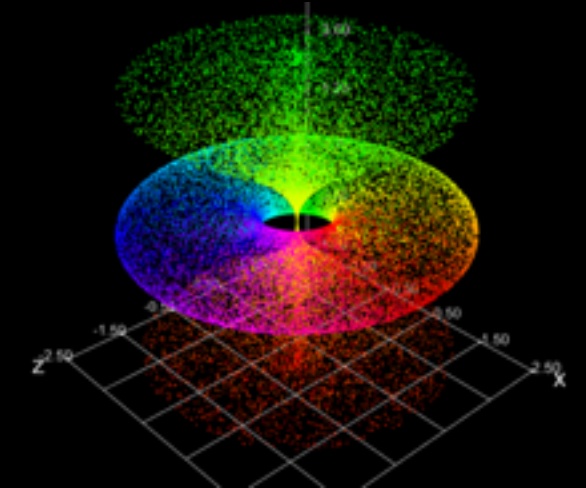
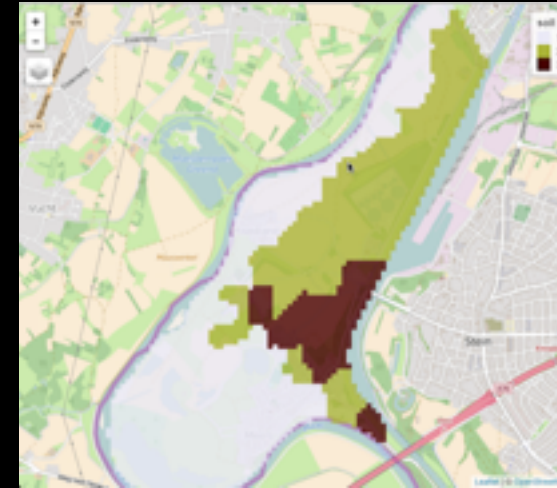
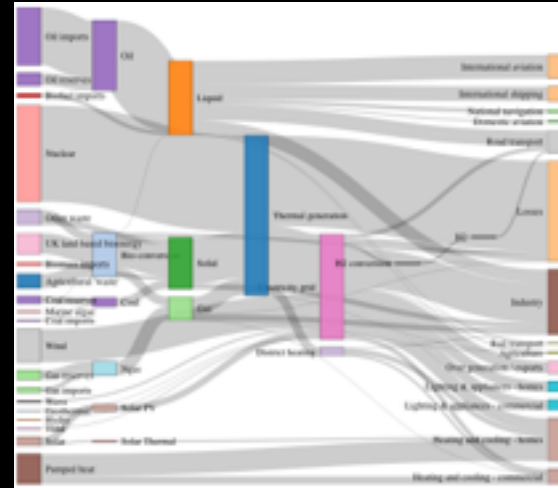
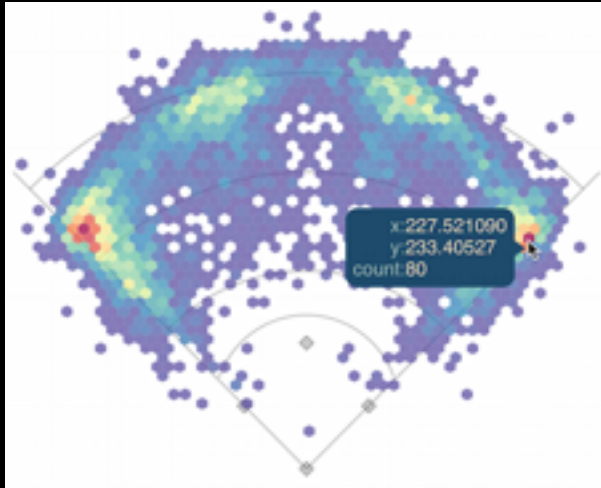
Demo 1

<http://rpubs.com/jcheng/dash1>

htmlwidgets

- An intentionally **unopinionated** framework
 - Unopinionated about **environment** – works in R console, RStudio viewer, R Markdown, Shiny
 - Unopinionated about **viz types** – traditional plots, slippy maps, tabular views, collapsible tree view
 - Unopinionated about **interactivity types** – anything you can do in JavaScript is fair game
 - Unopinionated about **viz API** – reuse almost any JS visualization library you want

Being unopinionated is great for diversity

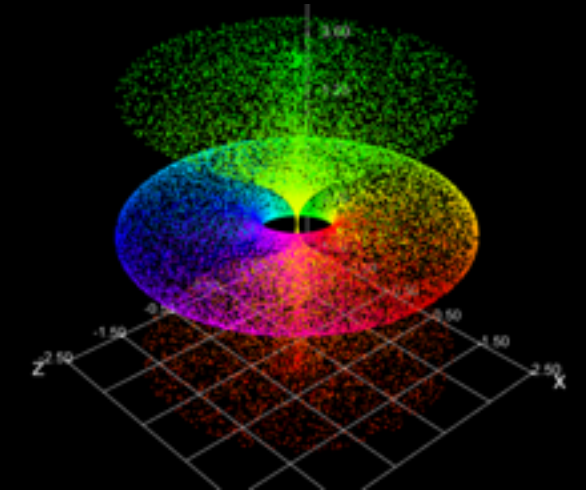
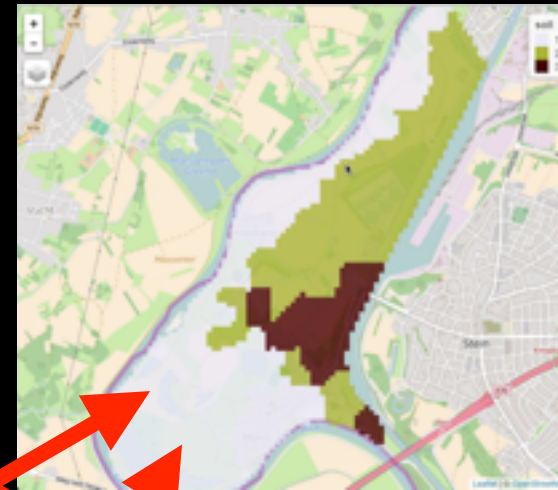
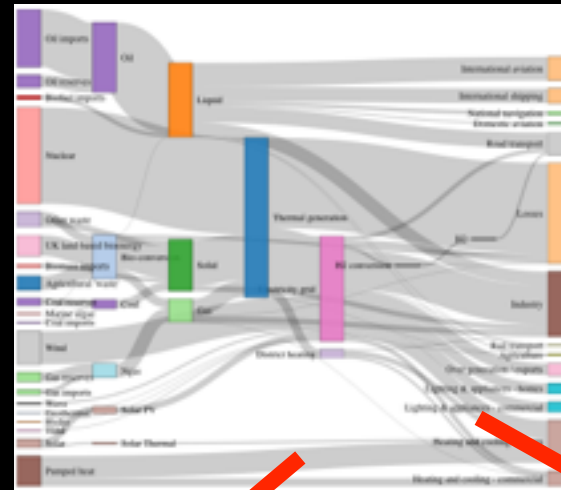
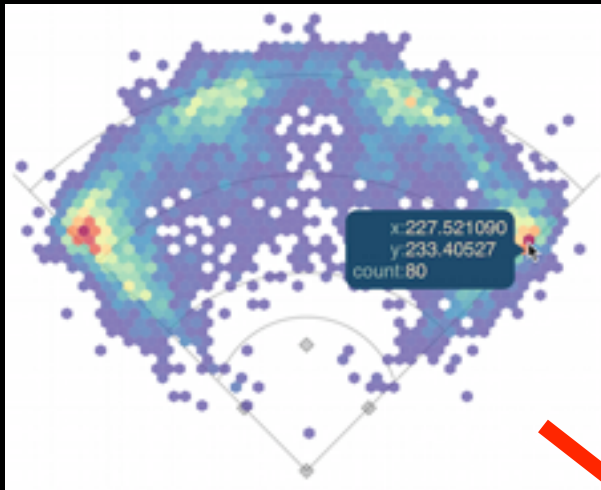


	f1	f2
1	I need to buy apples.	I need to buy apples.
2	I need to run the laundry.	I need to do the laundry.
3	I need to wash the dog.	I need to wash the car.
4	I need to get the car detailed.	I need to get the dog detailed.

	val	bool	big	small	dt
1		✓	A	a	07/10/2015
2		✓	B	b	07/11/2015
3		✓	C	c	07/12/2015
4		✓	D	d	07/13/2015
5		✓	E	e	
6		✓	F	f	
7		✓	G	g	
8		✓	H	h	
9		✓	I	i	
10		✓	J	j	

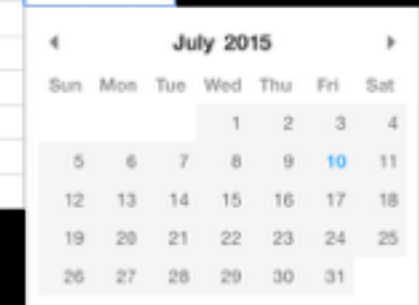
July 2015						
Sun	Mon	Tue	Wed	Thu	Fri	Sat
				1	2	3
				4	5	6
				7	8	9
				10	11	12
				13	14	15
				16	17	18
				19	20	21
				22	23	24
				25	26	27
				28	29	30
				31		

Being unopinionated is great for diversity





	f1	f2	
1	I need to buy apples.	I need to buy apples.	1
2	I need to run the laundry.	I need to do the laundry.	2
3	I need to wash the dog.	I need to wash the car.	3
4	I need to get the car detailed.	I need to get the dog detailed.	4

	val	bool	big	small	dt
1		<input checked="" type="checkbox"/>	A	a	07/10/2015
2		<input checked="" type="checkbox"/>	B	b	07/11/2015
3		<input checked="" type="checkbox"/>	C	c	07/12/2015
4		<input checked="" type="checkbox"/>	D	d	07/13/2015
5		<input checked="" type="checkbox"/>	E	e	
6		<input checked="" type="checkbox"/>	F	f	
7		<input checked="" type="checkbox"/>	G	g	
8		<input checked="" type="checkbox"/>	H	h	
9		<input checked="" type="checkbox"/>	I	i	
10		<input checked="" type="checkbox"/>	J	j	



but makes coordination between plots difficult.

Three approaches

- Automatic coordination (crosstalk) 
- Author directs coordination with R (Shiny)
- Author directs coordination with JS (robserve) 

Crosstalk

- Opinionated about **coordination** – currently limited to filtering and linked brushing
- Opinionated about **data model** – data frame (or sufficiently data-frame-like object) as input
- Opinionated about **visualization** – each observation/row should be displayed distinctly (no summarizing)
- Shiny supported but not required; static .html output is possible

	Native	Web
Building apps	JGR, gWidgets	Shiny
Authoring documents	Sweave, knitr	R Markdown/knitr
Grammar of graphics	ggplot2	ggvis, plotly
Custom plotting	Base graphics, grid, lattice	htmlwidgets
Coordinated multiple views	GGobi, cranvas	htmlwidgets + crosstalk?

```
` `` {r}  
df <- quakes
```

```
leaflet(df) %>% addMarkers()
```

```
datatable(df)  
` ``
```

```
` `` {r}  
shared_df <- SharedData$new(quakes)  
  
leaflet(shared_df) %>% addMarkers()  
  
datatable(shared_df)  
` ``
```

Demo 2

<http://rpubs.com/jcheng/dash2>

Compatibility

leaflet (rstudio/leaflet@joe/feature/crosstalk-filter)

DT (rstudio/DT@joe/feature/crosstalk)

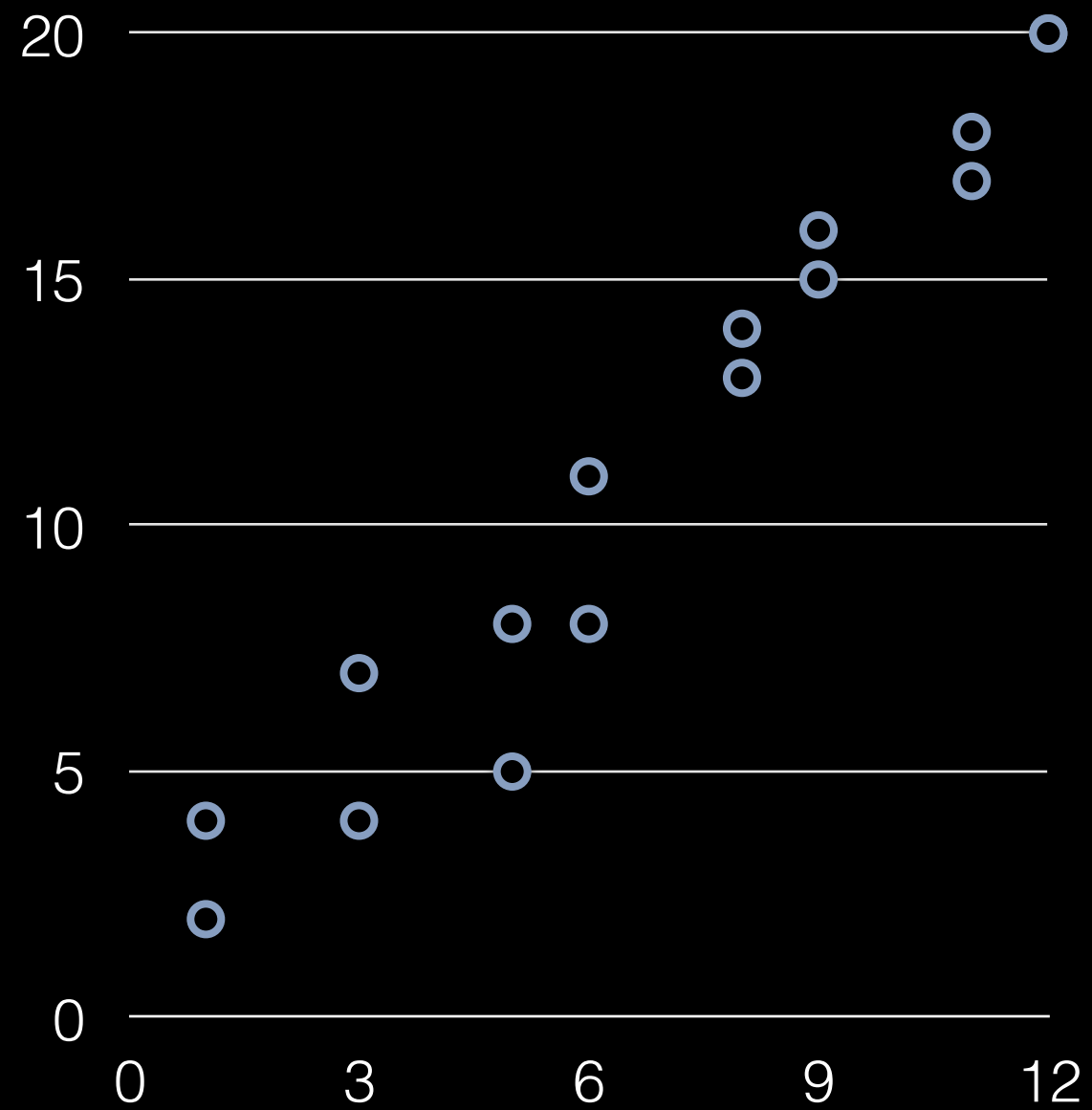
d3scatter (jcheng5/d3scatter)

plotly (jcheng5/plotly@joe/feature/crosstalk)

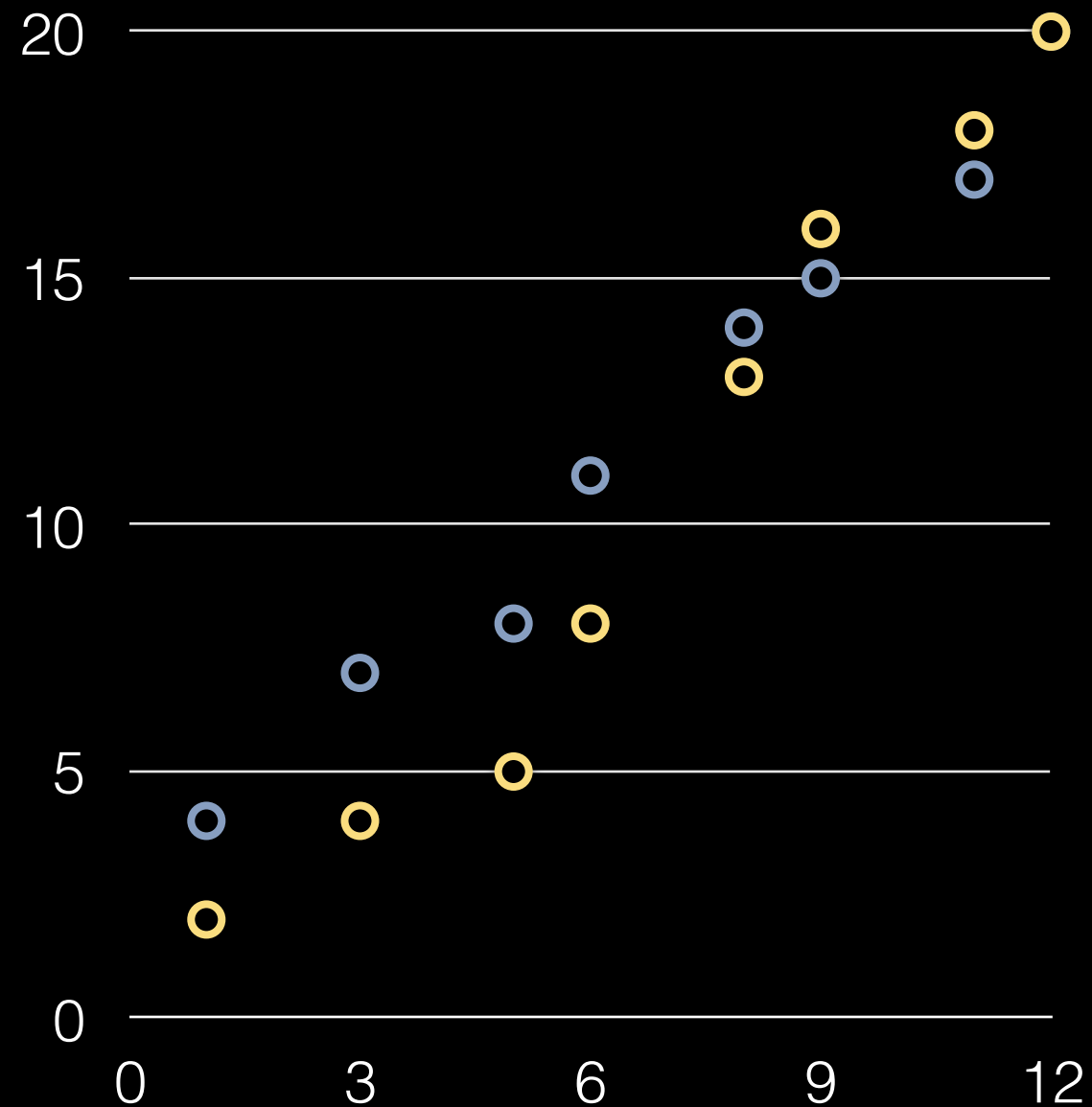
Significant limitations

- Only "standardized" behaviors supported
- Widget authors must do extra work to support each Crosstalk behavior
- Row-level data must be sent to the browser (i.e. potentially lots of data)
- **Can't use R to aggregate or summarize, fit models, smooth, quantize, etc.**

```
d3scatter(mtcars, ~disp, ~hp)
```

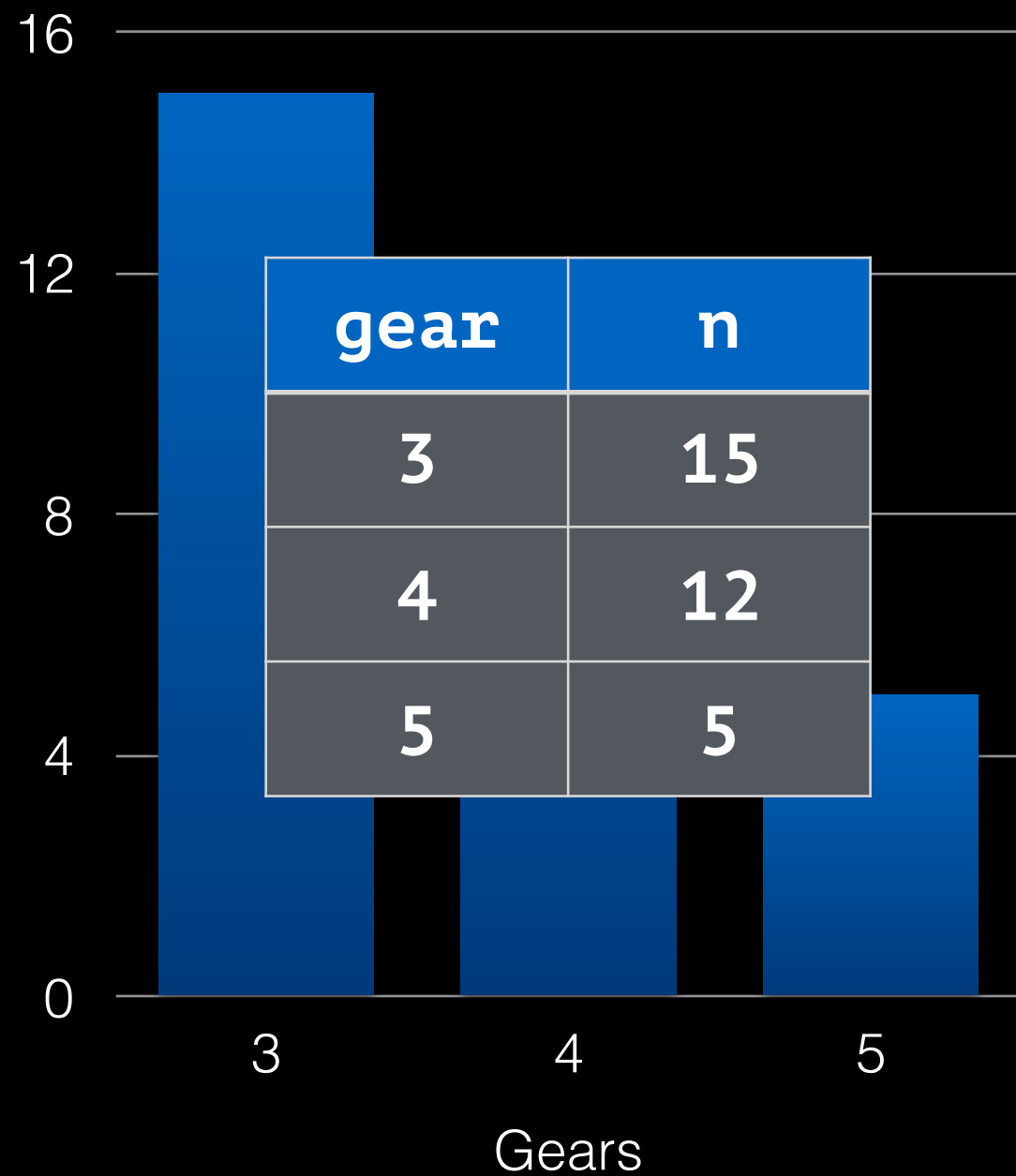



```
d3scatter(mtcars, ~disp, ~hp)
```

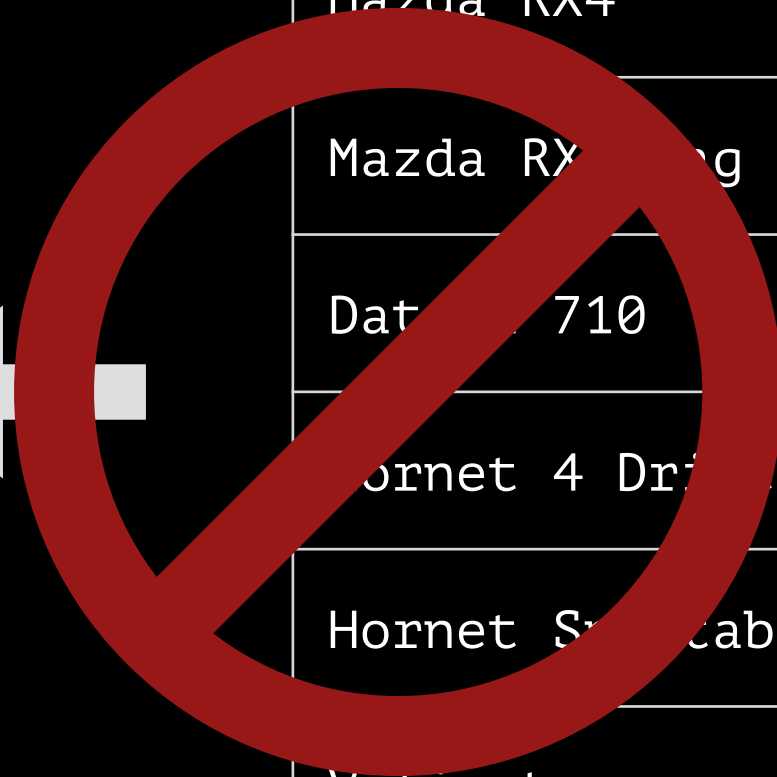
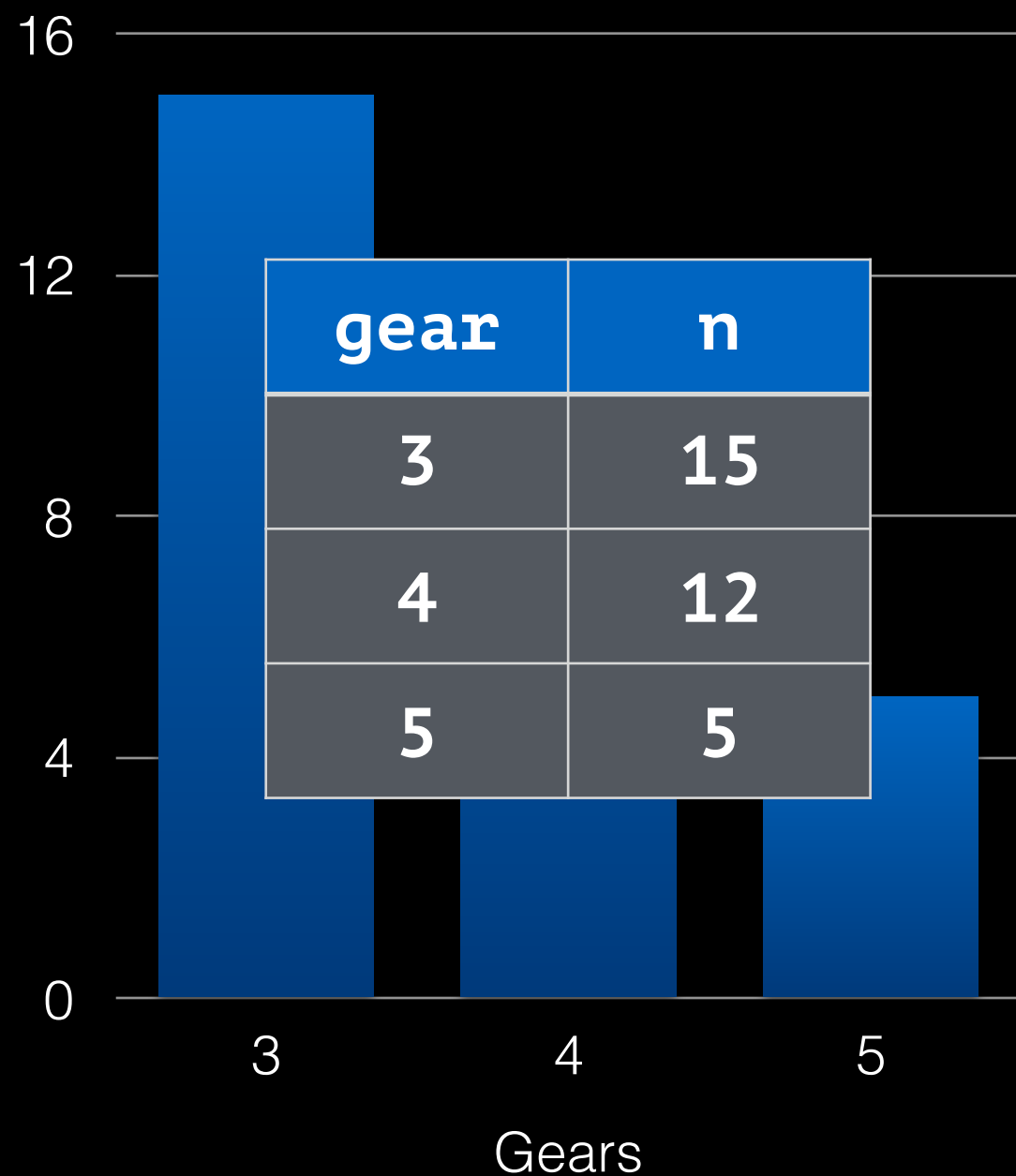


Rows selected
Mazda RX4
Mazda RX4 Wag
Datsun 710
Hornet 4 Drive
Hornet Sportabout
Valiant
Duster 360

```
df <- mtcars %>% group_by(gear) %>% tally()
d3bar(df, ~gear, ~n)
```



```
df <- mtcars %>% group_by(gear) %>% tally()
d3bar(df, ~gear, ~n)
```



Rows selected	
Mazda RX4	
Mazda RX4 Sport	
Datsun 710	
Hornet 4 Door	
Hornet Sportabout	
Valiant	
Duster 360	

Significant limitations

- Only "standardized" behaviors supported
- Widget authors must do extra work to support each Crosstalk behavior
- Row-level data must be sent to the browser (i.e. potentially lots of data)
- **Can't use R to aggregate or summarize, fit models, smooth, quantize, etc.**

Significant limitations

Shiny

- Only "standardized" behaviors supported
- Widget authors must explicitly support each Crosstalk behavior
- Row-level data must be sent to the browser (i.e. potentially all data)
- **Can't use R to aggregate or summarize, fit models, smooth, quantize, etc.**

Demo 3

<https://github.com/jcheng5/crosstalk-demo/blob/master/demo3.Rmd>

Crosstalk

Shiny

Pros

Deploy anywhere
Most responsive UI
Infinitely scalable

Can use R at runtime
Use any widget, any data
Use large data sets

Cons

Specific interactions only
Tabular data only
Smaller data sets
Requires widget support

Requires server with R
Takes effort to scale
Higher latency in UI
More coding required

Thank you

- <https://github.com/rstudio/crosstalk>
- Examples: <https://github.com/jcheng5/crosstalk-demo>
- Your feedback on Crosstalk matters, even more than usual!
- [@jcheng, joe@rstudio.com](mailto:joe@rstudio.com), or file a GitHub issue