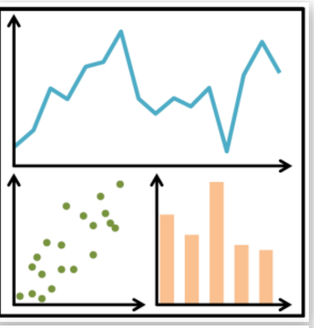
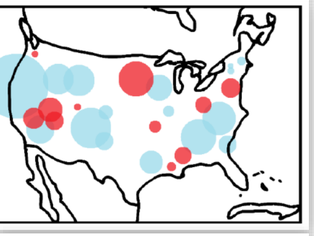
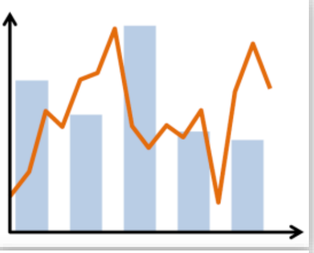
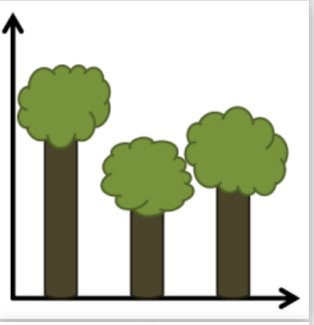
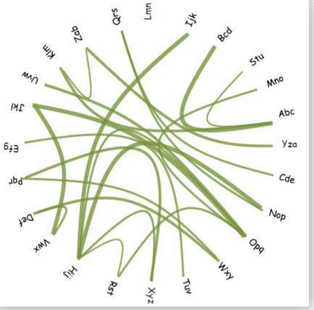






# Lecture 4: Tableau, D3

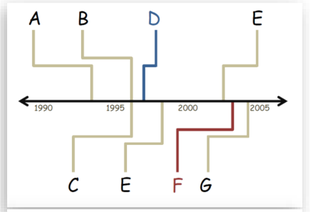
DS 4200

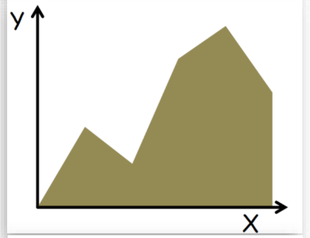
FALL 2020

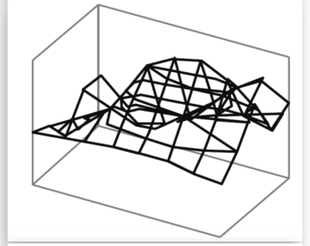
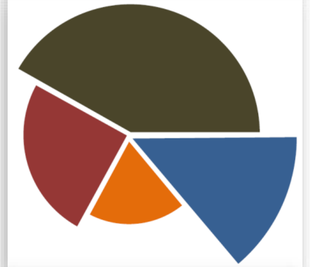
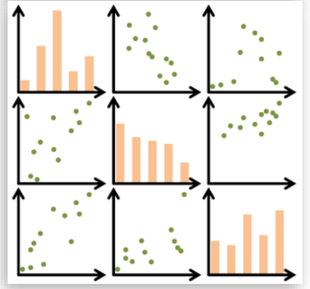
*Prof. Cody Dunne*

*NORTHEASTERN UNIVERSITY*

*Slides and inspiration from Michelle Borkin, Krzysztof Gajos, Hanspeter Pfister, Miriah Meyer, Jonathan Schwabish, and David Sprague*





# Check-in

# PREVIOUSLY, ON DS 4200…

# Assignment/Project deadlines:

# T 11:59pm

# Assignment 2b — Who Lives in the South End? (Tableau)

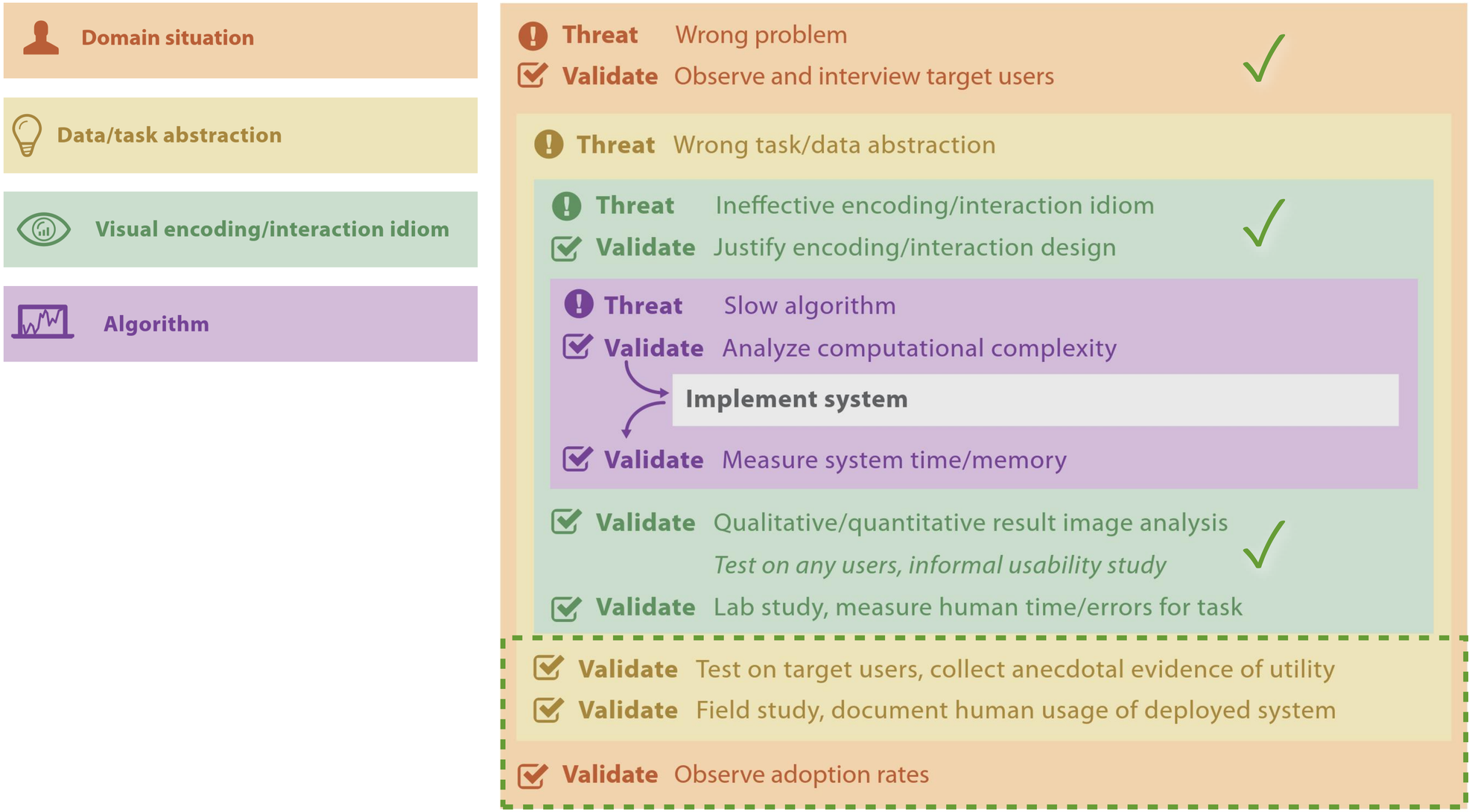
Now Due F 11:59pm

# JS TIPS AND TRICKS

*Slides and inspiration from Sara Di Bartolomeo*

# Threats to Validity

✓ *Final Project validation*




Final project follow-up

# PROJECTS

(Using the nested model via *design study "lite" methodology*)

https://northeastern.instructure.com/courses/18721/pages/project-overview

# PROJECTS

In-class project pitches: M 2020-09-30

What questions do you have for me?

# Now, on DS 4200…

# TABLEAU TUTORIAL

# In-Class Tool Introduction — Tableau

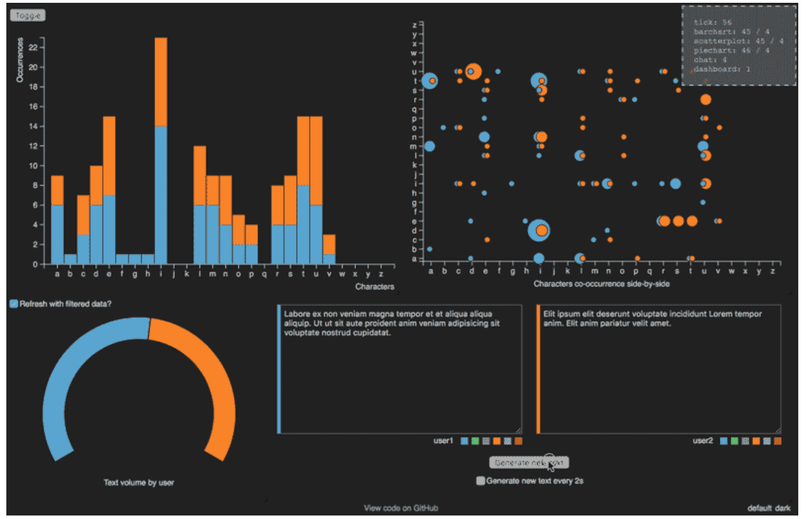
*~25 min total*

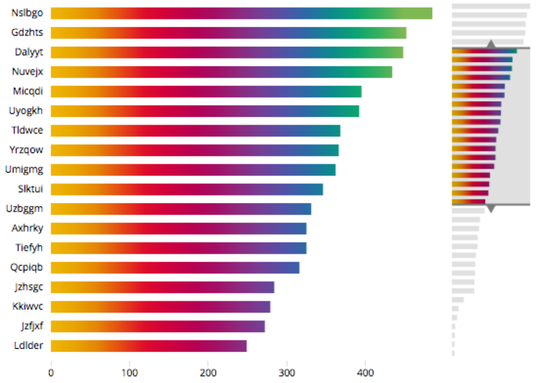
# D3 Tutorial

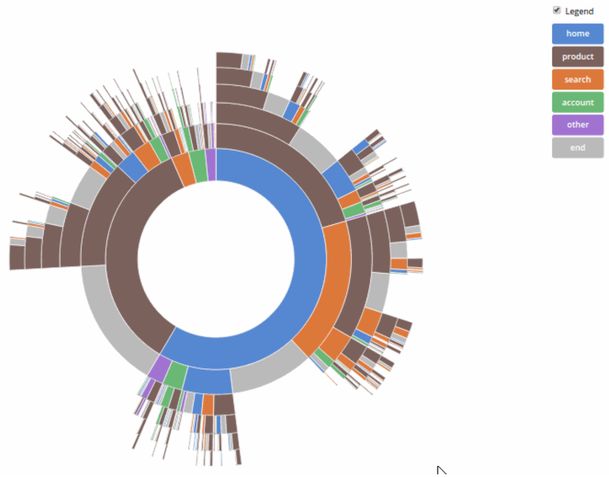
# Data Driven Documents

| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | Code | Name | Population | % with Health Care Coverage | % Adult current smokers | % Overweight | %Obese |
| 2 | AK | Alaska | 710231 | 82.4 | 20.4 | 40.7 | 25.2 |
| 3 | AL | Alabama | 4779736 | 83.8 | 21.9 | 37.0 | 33.0 |
| 4 | AR | Arkansas | 2915918 | 78.7 | 22.9 | 36.3 | 30.9 |
| 5 | AZ | Arizona | 6392017 | 89.5 | 13.5 | 40.7 | 24.7 |
| 6 | CA | California | 37253956 | 82.2 | 12.1 | 36.9 | 24.7 |
| 7 | CO | Colorado | 5029196 | 83.6 | 16.0 | 36.2 | 21.4 |
| 8 | CT | Connecticut | 3574097 | 90.2 | 13.2 | 37.5 | 23.0 |
| 9 | DC | District of Columbia | 601723 | 93.5 | 14.8 | 34.8 | 22.7 |
| 10 | DE | Delaware | 897934 | 90.0 | 17.3 | 35.3 | 28.7 |
| 11 | FL | Florida | 18801310 | 83.0 | 17.1 | 37.8 | 27.2 |
| 12 | GA | Georgia | 9687653 | 83.7 | 17.6 | 35.3 | 30.4 |
| 13 | HI | Hawaii | 1360301 | 93.2 | 14.5 | 34.1 | 23.1 |
| 14 | IA | Iowa | 3046355 | 89.6 | 16.1 | 37.1 | 29.1 |
| 15 | ID | Idaho | 1567582 | 80.9 | 15.7 | 36.0 | 26.9 |
| 16 | IL | | | | | | 28.7 |
| 17 | IN | | | | | | 30.2 |
| 18 | KS | | | | | | 30.1 |
| 19 | KY | | | | | | 31.8 |
| 20 | LA | | | | | | 31.7 |

| Month | Salesman | Region | Product | No. Customers | Net Sales | Profit / Loss |
|---|---|---|---|---|---|---|
| Jan-07 | Joseph | North | FastCar | 8 | 1,592 | 563 |
| Jan-07 | Joseph | North | RapidZoo | 8 | 1,088 | 397 |
| Jan-07 | Joseph | West | SuperGlue | 8 | 1,680 | 753 |
| Jan-07 | Joseph | West | FastCar | 9 | 2,133 | 923 |
| Jan-07 | Joseph | West | RapidZoo | 10 | 1,610 | 579 |
| Jan-07 | Joseph | Middle | SuperGlue | 10 | 1,540 | 570 |
| Jan-07 | Joseph | Middle | FastCar | 7 | 1,316 | 428 |
| Jan-07 | Joseph | Middle | RapidZoo | 7 | 1,799 | 709 |
| Jan-07 | Lawrence | North | SuperGlue | 8 | 1,624 | 621 |
| Jan-07 | Lawrence | North | FastCar | 6 | 726 | 236 |
| Jan-07 | Lawrence | North | RapidZoo | 9 | 2,277 | 966 |
| Jan-07 | Lawrence | West | SuperGlue | 6 | 714 | 221 |
| Jan-07 | Lawrence | West | FastCar | 9 | 2,682 | 1,023 |
| | Lawrence | West | RapidZoo | 6 | 1,500 | 634 |



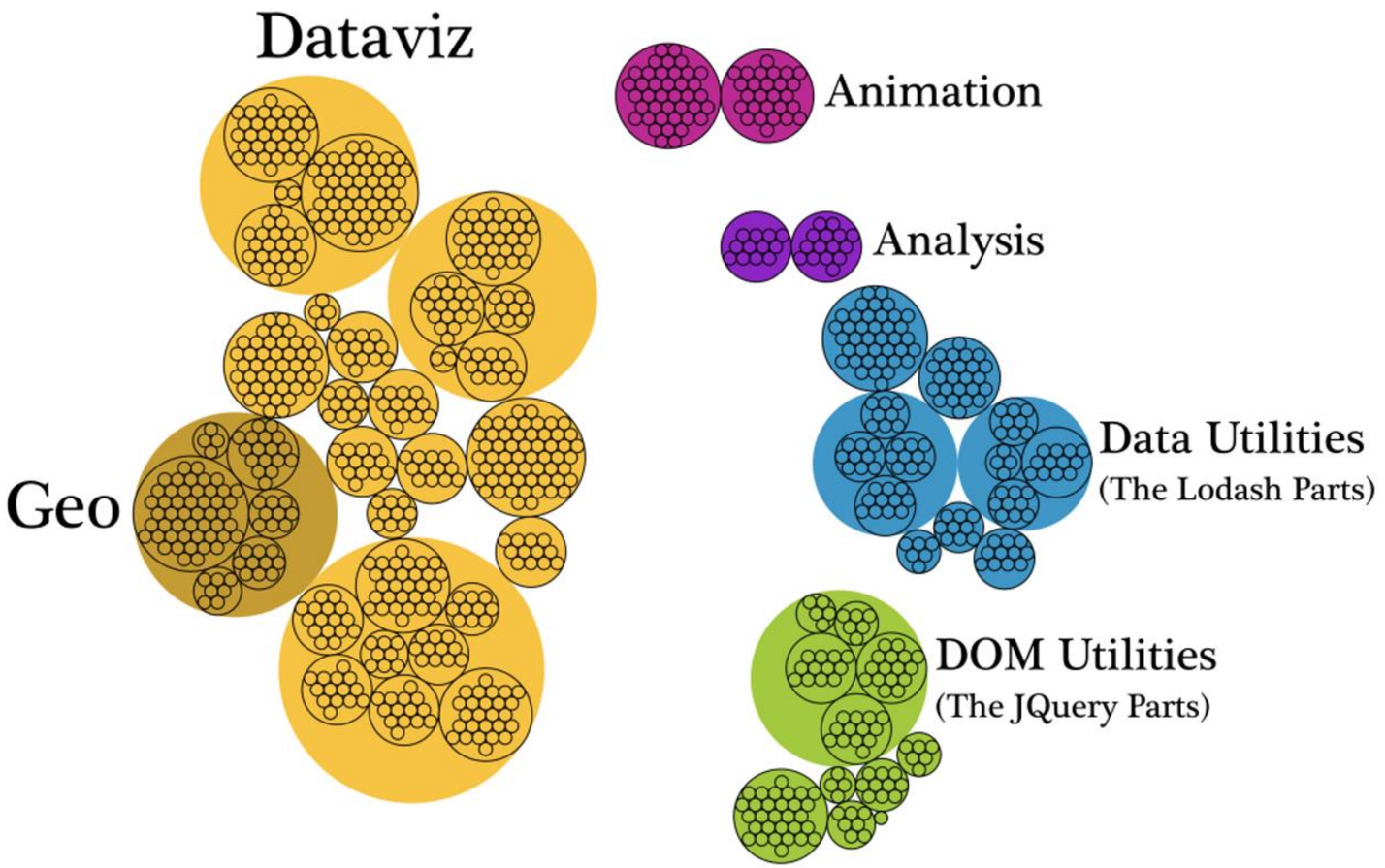
# Data Driven Documents

- https://d3js.org/
- D3 is a javascript library to manipulate documents based on data.

→ **not** a data visualization library (it's not like plotly, not matplotlib, ...)

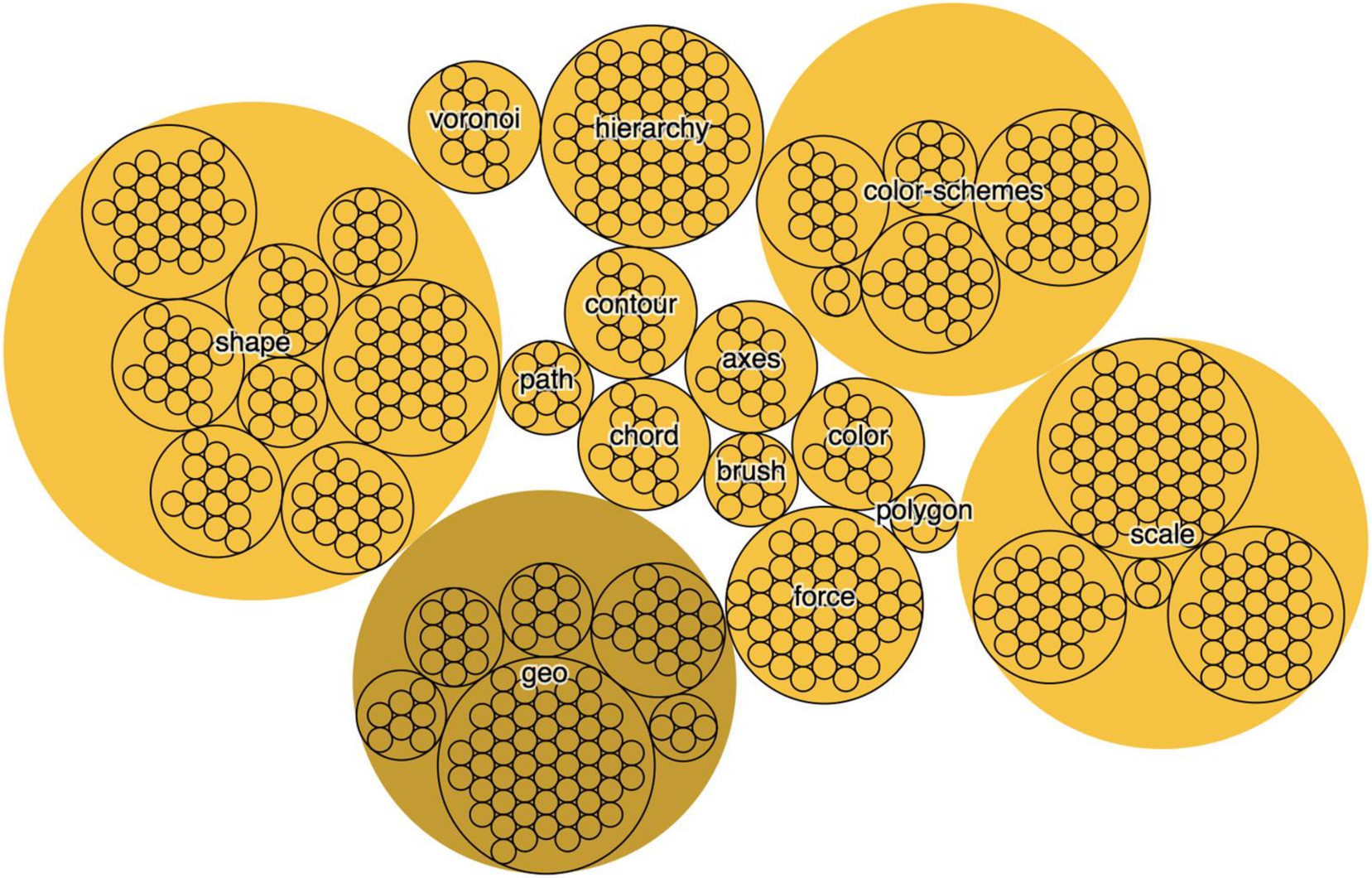
D3 is not a Data Visualization Library - Elijah Meeks

→ no out of the box charts (no functions to automatically build a chart)




The D3 API

Meeks, 2018


voronoi
hierarchy
color-schemes
shape
contour
axes
path
chord
color
brush
polygon
scale
force
geo

Meeks, 2018

# Vector (svg) vs. raster (canvas, png, jpg, ...)

- Formulas that describe the lines and points that make up an image
- Independent from the size of an image
- Always looks crisp, no matter how much you zoom in or distort the picture
- Graphics in SVG will be heavier to process

- Describe the color content of each pixel
- Will appear blurry/pixelated if you zoom in too much

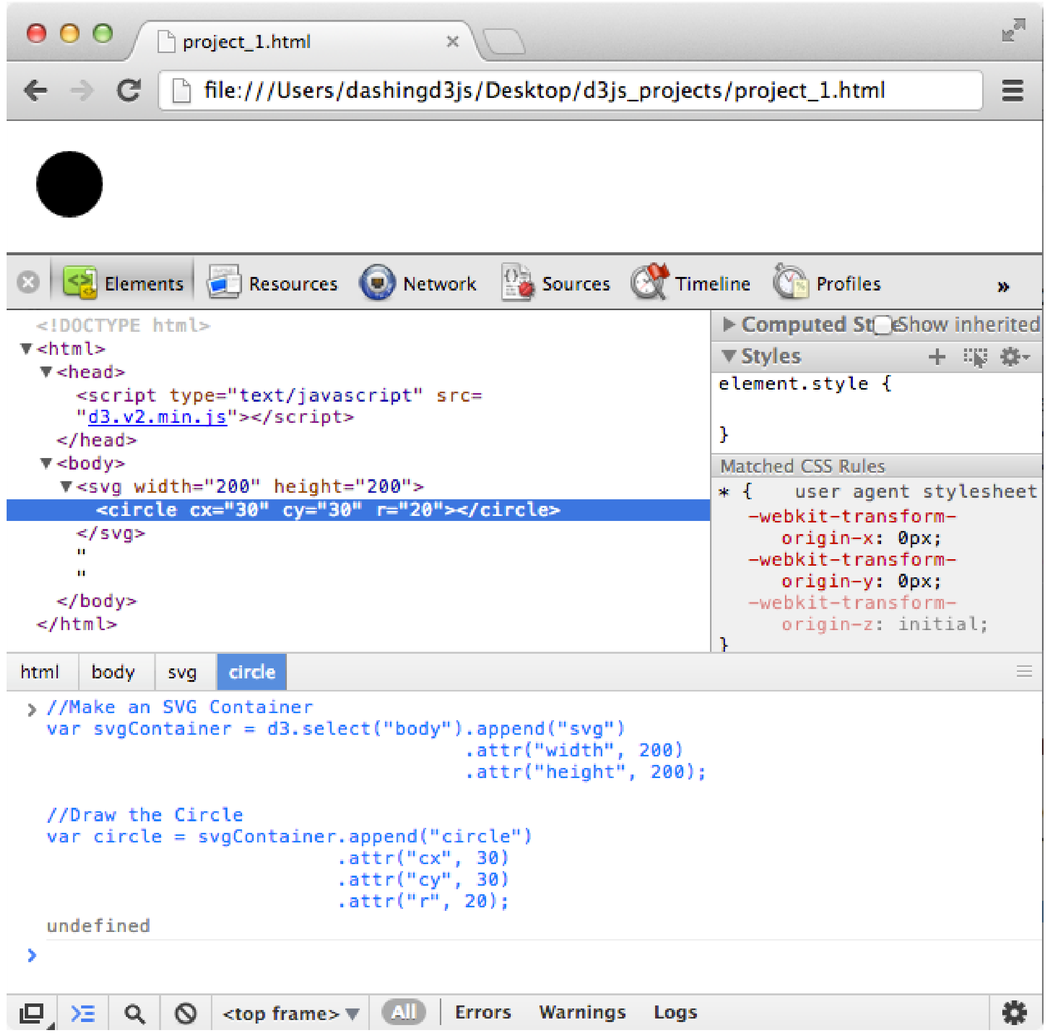

Minervini, 2017

# SVG

- <svg> tag. E.g., <svg width='500' height='500'>
- can add <style> attributes
- Basic SVG shapes: rect, circle, line, text, polyline
- Can group elements using the <g> tag

```
svg = d3.select('body').append('svg')
    .attr('width', 200)
    .attr('height', 200)

var circle = svg.append('circle')
    .attr('cx', 30)
    .attr('cy', 30)
    .attr('r', 20)
    .attr('fill', 'black')
```




Example: drawing

# Selections

Selections:

.select ('selectors’)

.selectAll ('selectors’)

```
.select('tagname')      // select by name of the element
.select('#id')          // select by id of the element
.select('.classname')   // select by class name
```

More info on selections: https://bost.ocks.org/mike/selection/

Example: selections-GoT

# Data Binding

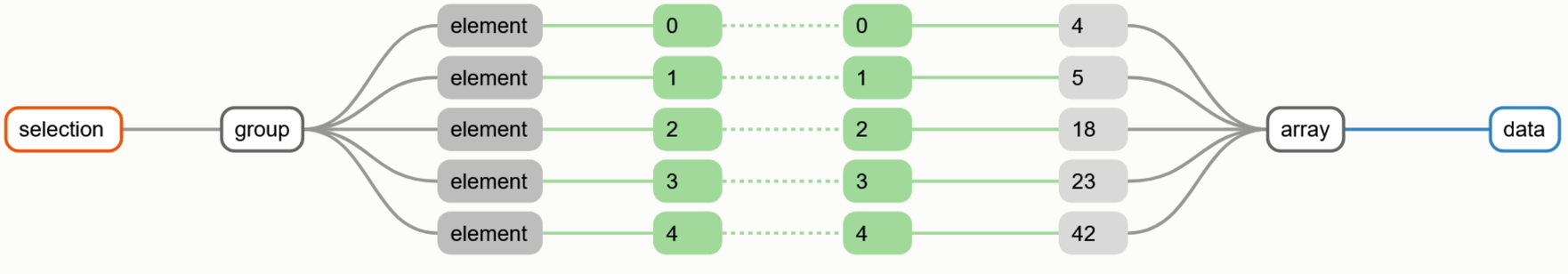
Data can be added in a number of different ways

Simplest way is through→ .data( )

The .data( ) method joins the current selection with entries in your dataset

# Data Binding

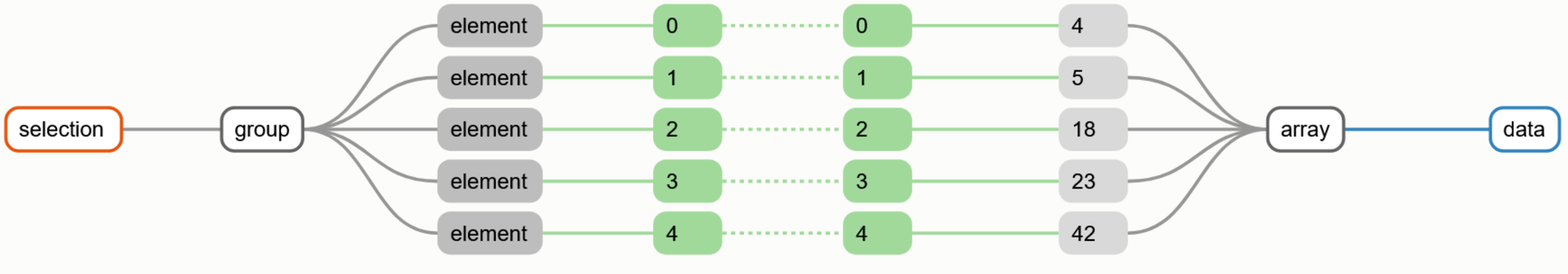
```
var numbers = [4, 5, 18, 23, 42];
```



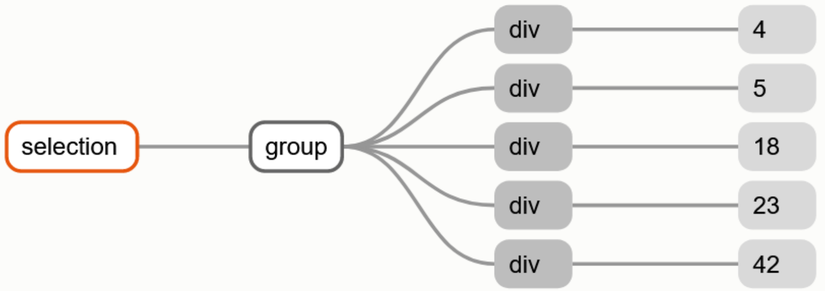

Bostock, 2013

# Data Binding

```
var numbers = [4, 5, 18, 23, 42];
```




```
d3.selectAll("div").data(numbers);
```

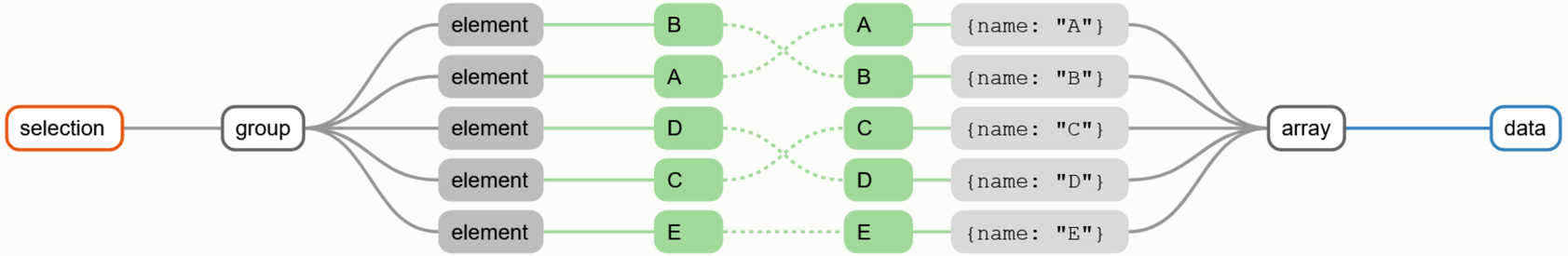



Bostock, 2013

# Data Binding

```
var letters = [
  {name: "A", frequency: .08167},
  {name: "B", frequency: .01492},
  {name: "C", frequency: .02780},
  {name: "D", frequency: .04253},
  {name: "E", frequency: .12702}
];

function name(d) {
  return d.name;
}
```

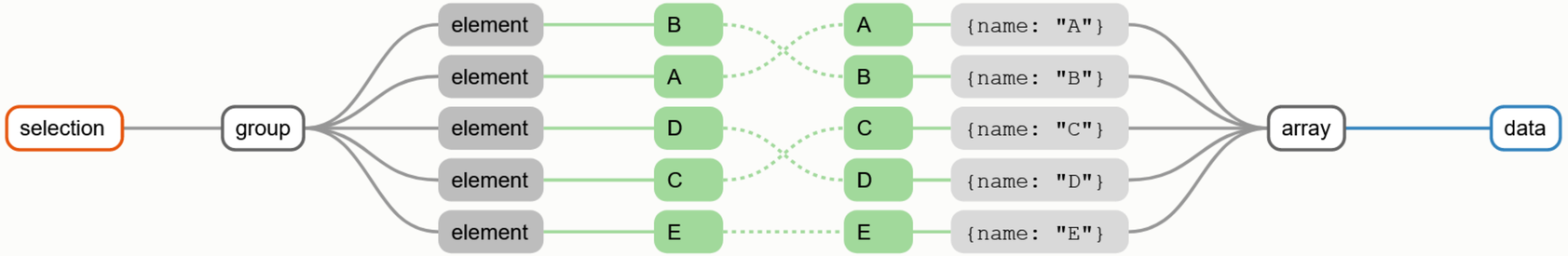



Bostock, 2013

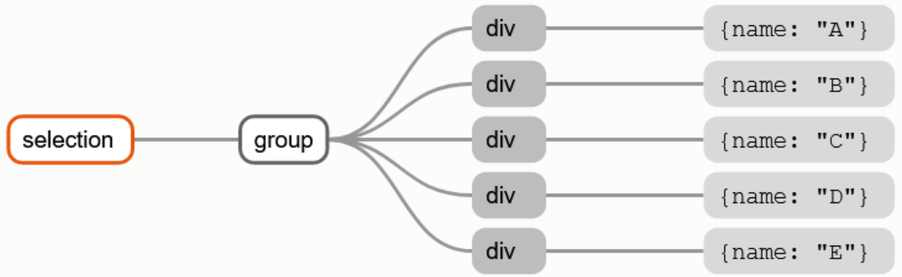
# Data Binding

```
var letters = [
  {name: "A", frequency: .08167},
  {name: "B", frequency: .01492},
  {name: "C", frequency: .02780},
  {name: "D", frequency: .04253},
  {name: "E", frequency: .12702}
];

function name(d) {
  return d.name;
}
```




```
d3.selectAll("div").data(letters, name);
```




Bostock, 2013

# Data Binding

If you ever get lost:

"How selections work:" https://bost.ocks.org/mike/selection/

Example: data_binding

# Modifying Elements

- text( ) // changes the text of the selection
- html( ) // allows you to modify the html
- append( ) // add element to the last child of the selection
- insert( ) // adds element to a more specific position
- remove( ) // deletes element

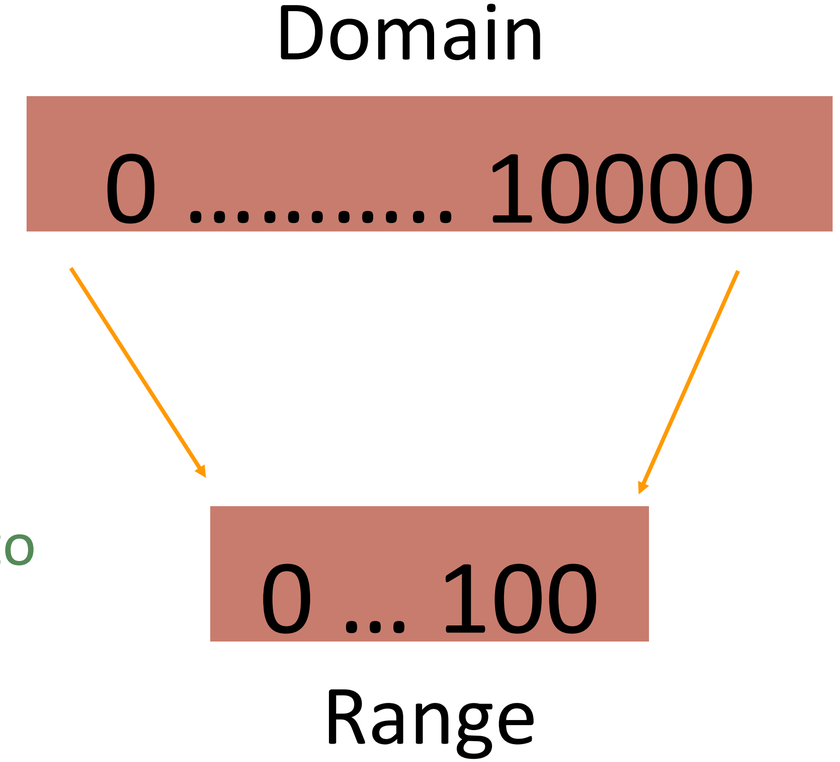
# Controlling Attributes

- style( ) // gives access to any CSS styles
- classed( ) // allows you to toggle classes on and off
- attr( ) // allows you to access any attributes
- property( ) // almost same as attr()

Example: selections-GoT

# Linear Scales

- scaleLinear( )  // Quantitative attributes
- domain( )  // Original values that you will modify
- range ( )  // Values that we want to scale our data to



# Ordinal Scales

- scaleBand( )  // categorical attributes
- domain( )  // original values that you will modify
- range ( )  // Values that we want to scale our data to
- padding()  // e.g., to control the spacing in between the bars