**From:** Piruz Alemi

**Subject:** Report on Advance D3 Visuals

Date: Feb 24th, 2020

Run: <http://127.0.0.1:5500/alemiToolTips.html>

A picture containing screenshot

Description automatically generated

A close up of a device

Description automatically generated

**Background**

Welcome to the newsroom! I've just accepted a data visualization position for a major metro paper. I was tasked with analyzing the current trends shaping people's lives, as well as creating charts, graphs, and interactive elements to help readers understand our findings.

The editor wants to run a series of feature stories about the health risks facing particular demographics. She's counting on me to sniff out the first story idea by sifting through information from the U.S. Census Bureau and the Behavioral Risk Factor Surveillance System.

The data set I checked was based on 2014 ACS 1-year estimates: <https://factfinder.census.gov/faces/nav/jsf/pages/searchresults.xhtml>, but she also left me free to investigate a different data set. The current data set included data on rates of income, obesity, poverty, etc. by state with some MOE which stands for "margin of error."

A picture containing indoor, table, floor, building

Description automatically generated

**Before I Began**

I did some installations:

## Installing

If you use NPM, npm install d3-axis. Otherwise, download the [latest release](https://github.com/d3/d3-axis/releases/latest). You can also load directly from [d3js.org](https://d3js.org/), either as a [standalone library](https://d3js.org/d3-axis.v1.min.js) or as part of [D3 4.0](https://github.com/d3/d3). (To be useful, you’ll also want to use [d3-scale](https://github.com/d3/d3-scale) and [d3-selection](https://github.com/d3/d3-selection), but these are soft dependencies.) AMD, CommonJS, and vanilla environments are supported. In vanilla, a d3 global is exported:

1. I found Complete Bar Charts, In D3-Activities / 2 + the following sites very useful:
2. See <https://www.d3indepth.com/enterexit/>
3. I checked my sources, among them: <https://www.d3indepth.com/scales/>
4. See this example of create [D3 Band Scales & Bottom Axes](https://bl.ocks.org/biovisualize/9c0d30d0539914ecdb15).
   1. I Remembered that the domain and range must both be arrays.
5. For assistance with axis creation with D3, see the [d3-axis documentation](https://github.com/d3/d3-axis).
6. On scaling see: <https://www.d3indepth.com/scales/>
7. For line generating see: <https://www.d3indepth.com/shapes/#line-generator>
8. <https://github.com/d3/d3-shape#line>
9. For locale.parsing see: <https://github.com/d3/d3-time-format/blob/master/README.md#locale_parse>
10. <https://github.com/d3/d3-time-format/blob/master/README.md#locale_format>
11. <https://observablehq.com/@d3/line-chart>
12. <https://observablehq.com/@d3/scatterplot-with-shapes?collection=@d3/d3-shape>
13. <http://bl.ocks.org/Caged/6476579>
14. <https://stackoverflow.com/questions/44427544/embed-a-svg-shape-in-d3tip-tooltip>
15. For adding axes: <http://www.d3noob.org/2012/12/adding-axis-labels-to-d3js-graph.html>
16. <https://github.com/Caged/d3-tip>
17. On Transitions: <https://bl.ocks.org/d3noob/899a0b2490318a96f9ebd40a5a84e4a7>
18. On d3.js Tips & Tricks: <https://leanpub.com/d3-t-and-t-v4>

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Created a new repository for this project called D3-challenge.
2. Cloned the new repository to my computer.
3. Inside my local git repository, created a directory for the D3 challenge. Used the folder name to correspond to the challenge: **D3\_data\_journalism**.
4. This work utilized both **html** and **Javascript** so I made sure to add all the necessary files. These were the main files to run for my analysis.
5. I then Pushed the above changes to GitHub

**My Task**

**Level 1: D3 Dabbler**

1. I created a scatter plot between two of the data variables such as Healthcare vs. Poverty or Smokers vs. Age.
2. Used the D3 technique, I created a *scatter plot that represents each state with circle elements.*
3. I coded the above graphic in the app.js file —made sure I pulled in the data from data.csv by using the d3.csv function. My scatter plot appeared like the image at the top of this section.

* *Included state abbreviations in the circles.*
* Created and situated my axes and labels to the left and bottom of the chart.
* Note: I needed to use python -m http.server to run the visualization. This hosted the page at localhost:8000 in your web browser.

**Level 2: Impress the Boss (Optional Challenge Assignment)**

* Why make a static graphic when D3 lets us interact with our data?

**1. More Data, More Dynamics**

I was going to include more demographics and more risk factors. I Placed additional labels in my scatter plot and gave them click events so that my users can decide which data to display.

I Animated the transitions for my circles' locations as well as the range of my axes. I did this for two risk factors for each axis. Or, for the extreme challenge, I created three for each axis.

* Hint: *I Tried binding all of the CSV data to my circles.* This allowed me to easily determine the x or y values when I clicked the labels.

A close up of a logo

Description automatically generated

A close up of a device

Description automatically generated

**2. Incorporate d3-tip**

While the ticks on the axes allow us to infer approximate values for each circle, it's impossible to determine the true value without adding another layer of data.

**Enter tooltips:** developers can implement these in their D3 graphics to reveal a specific element's data when the user hovers their cursor over the element. I Added tooltips to my circles and displayed each tooltip with the data that the user has selected. I Used the d3-tip.js plugin developed by [Justin Palmer](https://github.com/Caged)—which was already included the plugin in my assignment directory.

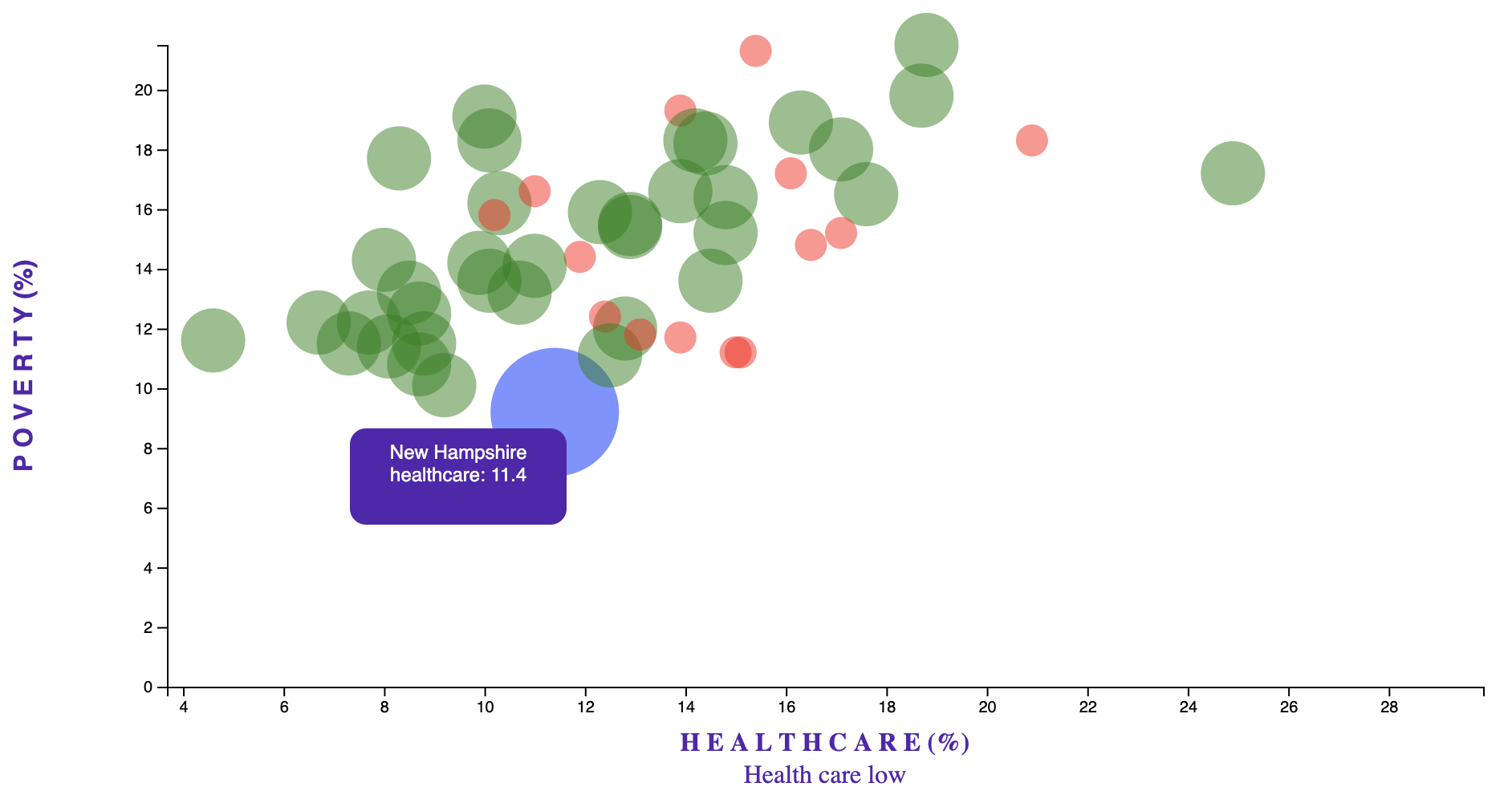
* I Checked out [David Gotz's example](https://bl.ocks.org/davegotz/bd54b56723c154d25eedde6504d30ad7) to see how we should implement tooltips with d3-tip.

**Assessment**

The final product was assessed on the following metrics:

* Creation of a **new** repository on GitHub called D3-Challenge (note the kebab-case).
* Completion of all steps in chosen level
* Coherency of scatter plot (labels, ticks)
* Visual attraction
* Professionalism

**I had some luck & this was the result!**

****