**Name**

D3-challenge – D3 Dabbler

**Overview**

This project required taking the 2014 ACS 1-year health information estimates from the US Census Bureau at (<https://data.census.gov/cedsci/>) and creating an interactive data visualization that compares at least two health elements by state. In this project, the dashboard was built to compare either Poverty levels as a percent of the population or the median Age for the state and compare it to Obesity. Other health factors are readily available if the visualization needs to be adjusted for user preference or need.

**Approach & Methodology**

The project called for using JavaScript, D3 and other tools to build interactive dashboards.

The Core Assignment: “D3 Dabbler” involved

* Creating a scatter plot between two of the data variables such as `Obesity vs. Poverty`
* Each state was represented in a scatter plot with circle elements.
* State abbreviations were provided in the circles for easy recognition
* X and Y Axes were created with representative labels.
* The visualization was built using a live server in the Edge browser
* Stye sheets were utilized for both the d3 and chart aspect of the visualization

Additional challenges

* Median Age was added as a health comparison element on the X-axis through an interactive label. The x-axis and tick labels adjust at the user changes their focus area (Age v. Poverty)
* A tooltip with the state, and selected data are provided when the user hovers over the appropriate circle.

**Key Takeaways**

D3 provides highly interactive and rich visualizations

How information is layered and sequenced in the JavaScript is important to overall success

**Repository Summary & Deliverables Locator**

The project repository is located at: https://github.com/rodgerskent/d3-challenge

The HTML dashboard is available at: <https://rodgerskent.github.io/d3-challenge/>

**Support**

Appreciate the course instructors, TA’s and the tutor program

**Contributing**

This project was complete on an individual basis with tutor and standard online support resources

**Project status**

Project is complete

D3-challenge working document

# D3 Homework - Data Journalism and D3

## Background

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

The editor wants to run a series of feature stories about the health risks facing particular demographics. She's counting on you 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 included with the assignment is based on 2014 ACS 1-year estimates from the [US Census Bureau](https://data.census.gov/cedsci/), but you are free to investigate a different data set. The current data set includes data on rates of income, obesity, poverty, etc. by state. MOE stands for "margin of error."

### Before You Begin

1. Create a new repository for this project called `D3-Challenge`.
   1. \*\*Do not add this homework to an existing repository\*\*.
2. Clone the new repository to your computer.
3. Inside your local git repository, create a directory for the D3 challenge.
   1. Use the folder name to correspond to the challenge: \*\*D3\_data\_journalism\*\*.
4. This homework utilizes both \*\*html\*\* and \*\*Javascript\*\* so be sure to add all the necessary files. These will be the main files to run for analysis.
5. Push the above changes to GitHub or GitLab.

## Your Task

### Core Assignment: D3 Dabbler (Required Assignment)

* You need to create a scatter plot between two of the data variables such as `Healthcare vs. Poverty` or `Smokers vs. Age`.
* Using the D3 techniques we taught you in class, create a scatter plot that represents each state with circle elements. You'll code this graphic in the `app.js` file of your homework directory—make sure you pull in the data from `data.csv` by using the `d3.csv` function. Your scatter plot should ultimately appear like the image at the top of this section.
* Include state abbreviations in the circles.
* Create and situate your axes and labels to the left and bottom of the chart.
* Note: You'll need to use `python -m http.server` to run the visualization. This will host the page at `localhost:8000` in your web browser.

### Bonus: Impress the Boss (Optional Assignment)

Why make a static graphic when D3 lets you interact with your data?

#### 1. More Data, More Dynamics

You're going to include more demographics and more risk factors. Place additional labels in your scatter plot and give them click events so that your users can decide which data to display. Animate the transitions for your circles' locations as well as the range of your axes. Do this for two risk factors for each axis. Or, for an extreme challenge, create three for each axis.

\* Hint: Try binding all of the CSV data to your circles. This will let you easily determine their x or y values when you click the labels.

#### 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. Add tooltips to your circles and display each tooltip with the data that the user has selected. Use the `d3-tip.js` plugin developed by [Justin Palmer](https://github.com/Caged)—we've already included this plugin in your assignment directory.

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

- - -

### Assessment

Your final product will be assessed on the following metrics:

1. Creation of a \*\*new\*\* repository on GitHub called `D3-Challenge` (note the kebab-case). Do not add to an already existing repo.
2. Completion of all steps in the core assignment
3. Coherency of scatter plot (labels, ticks)
4. Visual attraction
5. Professionalism
6. Ensure your repository has regular commits (i.e. 20+ commits) and a thorough README.md file