Exercise 3: Graphical Encoding

In this exercise, we have a dataset that contains the unemployment rate for each state for December 2019. Your task is to visualize this data in 4 different ways using D3. This exercise is about exploring possibilities, *so it's okay if some of your visualizations aren't the most effective, as long as they are still appropriate*. For example, the speedometer charts (see example) are not the most effective way, but they still work. A line chart, however, would not be appropriate and would not work for this type of data. Also, make sure to explore the largest possible set of marks and channels. Your solutions should include **at least 3 different channels and 3 different marks.**

Data

The map data (`usaGeo`) comes from [here](#) and is based on boundaries given by the U.S. Census Bureau.

The unemployment data (`unemployment`) comes from the [U.S. Bureau of Labor Statistics](#). We'll put the unemployment data in two formats:

- An array of objects where each object has the name and unemployment rate of the state.
- A single object where the keys are the state names and the values are the unemployment rates.

```
stateToRate = Object.fromEntries(new Map(unemployment.map(d => [d.state, d.rate])))
```

Result: `stateToRate = Object {North Dakota: 2.4, South Carolina: 2.4, … }`

We'll calculate the min and max unemployment rates:

```
extent = d3.extent(unemployment, d => d.rate)
```

Result: `extent = Array(2) [2.4, 6.1]`

Next, we'll create a continuous color scale. Feel free to modify this or use other color scales too (see, e.g., https://github.com/d3/d3-scale-chromatic). Note that different color maps do not count as different channels!

```
color = d3.scaleSequential().domain(extent).interpolator(d3. interpolatePurples)
```

Lastly, you may find it handy to have a mapping from state name to abbreviation (`stateToAbbr`). This data is from [World Population Review](#).

We provide you with a code template consisting of an HTML file and a JavaScript file. You can just put these two files on your hard drive and open the index.html in your browser. The image to the right shows the initial result.

Submit the code that you implemented for creating the visualizations to Brightspace (both the .html file and the .js file). Please also "decode" your visualization, e.g., specify which channels and marks you used and briefly discuss whether this is a good or bad solution for these data based on the effectiveness and expressiveness principle (you can put these short texts on the website you create).