Assignment 2: Creating Charts with D3 & Data Abstraction

Date Posted: February 2, 2024

Due Date: February 16, 2024 11:59pm

Description

This assignment contains two parts: (1) creating visualizations in D3 and (2) data abstraction.

In part 1 of this assignment you will re-create the scatterplot from HW1 part 2 using d3. You will also provide several design variations, including creating different color scales. The goal of this assignment is to familiarize you with the fundamental paradigms used by d3 and introduce you to more advanced concepts in d3, namely scales, axes, and transitions.

In part 2, you will practice abstracting data use the framework from our textbook. Specifically, you will be asked to identify the types of given attributes and practice abstracting a dataset into its dataset type, data types, and semantics.

Submission Requirements

Provided Template

In the GitHub repository, you will find five files: `index.html`, `scores.js`, `d3.v7.min.js`, `README.md`, and `spec.pdf`. The `index.html` provides a minimal html outline to set up your webpage and structure your responses. It includes two `<div>` tags, one for your visualization from part 1 and one for your written answers to part 2. The `scores.js` file contains the data for your scatterplot (the same data from HW 1). `d3.v7.min.js` is a copy of the d3 library. You will need to load this before any other scripts that use d3. `README.md` provides a template to use for documenting your repository as described below. Finally, `spec.pdf` contains a copy of the assignment specification (this document).

Submitted Files

You will create your visualizations in `index.html`, it should be the only HTML file in your submission. You may add any additional JavaScript and CSS files as you see fit. You must add appropriate documentation via commenting to all JavaScript code, whether it's in a separate `.js` file or written directly in your `index.html` file. Finally, you should include a `README.md` file (using the given template) that provides a text description of what is in your repository, how to run your program and any parameters that you used. Also, document any idiosyncrasies, behaviors, or bugs of note that you want us to be aware of. When you have completed your assignment, please submit the link to your GitHub repository on Canvas.

You must use `d3.js` to create your visualizations. However, you may also use any other external libraries that you see fit. If you opt to use additional libraries, please document which libraries and how you used them in your `README.md` file.

Part 1: Creating Visualizations in D3 (80 pts)

In this part of the assignment you will recreate your scatterplot from the previous assignment, as well as implement several design variations. The steps below describe all of the components that you should add to your visualization. The final product should be a single visualization with four buttons to switch between the different design variations described below.

Step 1: Re-create your scatterplot from HW 1, part 2.

The first part of this assignment is to reproduce the scatterplot from the previous homework, but this time using D3! You will use the same mappings as before, namely `SATM` should map to the x-axis, `SATV` should map to the y-axis, the `ACT` should map to the radius, and `GPA` should map to color. For each of these mappings, you will need to create a scale using D3 Scales to map the data values to their corresponding visual encodings, discussed below.

Scales

Recall in HW 1, you had to calculate the visual encodings (x & y coordinates, radius, and color) for your data by hand. This time, you will calculate them by defining appropriate D3 Scales for each value. You may find that `d3.min()`, `d3.min()`, and `d3.mean()` useful for creating your scales.

Circle Positions

For `cx` and `cx`, you will define two separate scales that map `SATM` to the x position and `SATV` to the y position. Both of these scales should be instances of `d3.scaleLinear()`.

Circle Radii

You will similarly create a function to control the mapping from `ACT` to the circle radius. The smallest circles should have radius 10 and the largest, radius 20. However, your radius scale should be created using `d3.scaleSqrt()` rather than `d3.scaleLinear()`. This is a better choice to control the *area* of a circle - as we will see in class, it is generally better to use area to control the size of a shape. By setting the radius relative to the square root, the area of each circle will correspond directly to the value. In contrast, if we set the radius linearly based on the value, small differences between values will be disproportionately encoded by the area. For example, if we use a linear scale where value *x* has radius *y*, then value *2x* will have radius *2y*. When we compare the areas of the resulting circles, the circle for *x* will be 1/4 of the size of circle for *2x* even though the actual value *x* is only 1/2 the value of *2x*.

Circle Colors

You will create three colormap variations. Each of your colormaps will use colors from the RdYlBu-5 colorscale from ColorBrewer.

a. Your first color scale will be a continuous color scale that interpolates linearly from the minimum GPA to the maximum, using two colors - the first (the darkest red) and the last (the darkest blue) - of the RdYlBu-5 colormap. The minimum GPA should be colored with red (`#d7191c`) and the maximum should be colored with blue (`#2c7bb6`).
b. Your second color scale will be a continuos scale that uses three of the colors of RdYlBu-5. The minmum GPA should map to the darkest red (`#d7191c`), the average GPA should map to the center yellow (`#ffffbf`), and the maximum GPA should map to the darkest blue (`#2c7bb6`). Hint: read the documentation on d3 scales.
c. Your third color scale will be a quantized scale using all five colors of RdYlBu-5 such that the minimum color maps to red, the maximum maps to blue, and there are five "bands" of GPA values each mapped to one of the colors. Hint: read the documentation on d3 quantize scales.

Step 2: Creating Plot Axes in D3

Next, you will add axes to your scatterplot using `d3.axisLeft()` and `d3.axisBottom()`. For an example, see here.
Be sure to label your axes as well. You are welcome to pad your scatterplot as you see fit to make room for the axes and labels.

Step 3: Transitions

In this step, you will implement an animations using `.transition()`. You will create three buttons to change between the three color scales you designed above. When you click a button to change color scales, it should animate the transition from the current color to the new color. For handling button clicks, you should use d3's event handling, specifically the `.on("click",...)` function.

Extra Credit (10 pts):

Create two additional buttons to toggle the x-coordinate of your circles, alternating between `SATM` (the current x-axis) and the cumulative SAT score, `SATM + SATV`. This change should be animated using `.transition()`. Note, the range will change and therefore the scale of your horizontal axis along with the axis labeling should change. For full credit, the transition of both the positions of the circles and the x-axis (including ticks and labels) should be animated.

Part 2 - Data Abstractions (20 pts)

For the following written questions, add your answers in the corresponding locations in the `index.html` file.

1. Data Attributes (5 pts)

What type of attribute (categorical, ordinal, quantitative-interval, quantitative-ratio) are each of the following?

a. The rank of Tulane University.
b. The color of your favorite sweatshirt.
c. A cumulative SAT score.
d. Your letter grade in this course.
e. The state you were born in.
f. The population of Louisiana.
g. The length of time it takes you to commute to Tulane.
h. The temperature of water to brew tea.
i. The amount precipitation in each state from 2023.

2. Data Abstraction (15 pts)

The video at the this link contains a visualization driven story about the fallen of World War II. For this exercise, watch the video from 1:00 to 7:20 and identify the following, towards the goal of abstracting the data.

- The dataset type
- The semantics of the data items (i.e. what does each item represent?)
- At least three attributes, specifying their types and semantics.
- At least one derived attribute, specifying its type and semantics.

Note, you are encouraged to justify/explain your answers where appropriate but are not required to. However, justified answers are more likely to receive partial credit.

Grading

In part 1, your visualization will be graded on how well it fulfills the specified criteria. Additionally, points will be deducted for lack of documentation (i.e. commenting your JS code and filling out the README.md file). A visualization that gets full marks will: use only d3 to add/manipulate SVG elements, use d3 to bind the data to the svg circles, use d3 scales to map the data values to the visual encodings (x position, y position, radius, & the three described colormaps), have axes with appropriate labels, have three buttons the switch between the 3 colormaps, and transition between colors when each of the buttons is clicked. The below rubric describes the point values for individual components.

Using d3 to bind the data and create/manipulate SVG circles	10
Using d3 scales to create the circle positions	12
Using d3 scales to create the circle radii	12
Using d3 scales to create three colormaps	21

Creating axes for your visualization	10
Creating buttons to switch between color scales	5
Animate the transition between color scales	5
Proper documentation (code comments and README.md)	5

In part 2, for question 1 you will be graded on the correctness of your answer to each part. Question 2 will be graded on the accuracy and plausabliilty of the provided response to each part. There may be mulitple abstractions of the data presented in the video - I will accept all answers that are plausible for the shown visualizations. When in doubt, justify/explain your answers! I don't expect you to perfectly identify every aspect of the data nor are you limited to only listing the specified number of attrbutes and derived attributes. If more than the required number are listed, I will not subtract points for incorrect answers beyond the required minimum. For example, if you provide three correct attributes and one incorrect, you will still receive full credit.