

CS/INFO 3300; INFO 5100

Homework 3

Due 11:59pm Wednesday, September 22

Goals: Practice using d3 to create SVG elements and set their aesthetic properties. Recognize the effect of data transformations through direct data changes and through scale functions. Practice working with color scales.

Your work should be in the form of an HTML file called index.html with one `<p>` element per problem. Wrap any SVG code for each problem in a `<svg>` element following the `<p>` element. For this homework we will be using d3.js. **In the `<head>` section of your file, please import d3 using this tag: `<script src="https://d3js.org/d3.v7.min.js"></script>`**

Create a zip archive containing your **HTML file and all associated data files** (such as diamonds.json) and upload it to CMS before the deadline. Submissions that do not include data files may be penalized. Your submission will be graded using a Python web server run in a parent directory containing your zip file contents (e.g. server started in `~/student_hw`, with your homework at `~/student_hw/your_netid/hw3/index.htm`) – be sure that it works.

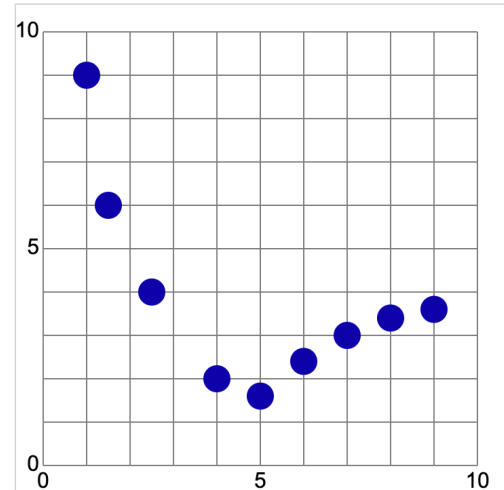
1. In your HTML, please create a **300x300 pixel SVG element**. Then, select it using `d3.select()` in the `<script>` section of your code. Unlike in HW2 where you drew things by hand, in this problem you are going to use `.append()` and `.style()` functions to build and decorate this canvas. Please use d3 functions to **create the following elements** in your canvas:

- A `<text>` element with the word "INFO3300" centered in the exact middle of the SVG canvas. Use `.attr()` to locate it. You are welcome to use text-anchor or pixels to center it. The `<text>` element should be styled to use a black 20px Verdana typeface.
- A `<circle>` element at (150, 150) with a 4px radius so that we can verify that the text is correctly centered. Please give it a light pink fill color and no stroke. It should appear **behind** the `<text>` element.
- **Three (3) `<rect>` elements** located in the white space around the text. They should be no larger than 50px x 50px. Give each of them a different stroke color and fill color. No two rectangles should overlap or be the same size. Make sure that the colors you choose make them stand out to the grader.

(see next page)

2. In HW2 you reproduced a plot from scratch using SVG. Now **create the same plot again**, but this time use **d3 functions to create it programmatically in a `<script>` tag**. While it should resemble the example image to the right, you don't need to recreate it exactly, so long as your point and line positions are correct.

Create a **360x360 pixel SVG element in HTML**. Use a CSS style to give the canvas a **1px light grey solid border**. The main plot region, excluding labels, should be a square **320x320 pixels** in size, running from (20,20) to (340, 340). Reserve the remaining pixels as padding for the labels.



On the last page of this assignment, we have included a code version of the dataset. Go ahead and copy it into your `<script>` tag. First create **x and y scale functions** that map from data coordinates to SVG coordinates, using the same minimum and maximum values as the chart *domain* (0 to 10 for both axes). Remember to account for the "padding" pixels when determining the *range* of point positions. You can choose whether to implement the margins by adjusting the range of your scales or adding a `<g>` element and `translate()`.

Next, create the **grid of lines for your chart**. While there is a way to make gridlines using d3 axes, please **manually create gridlines using a for loop**. You should create one horizontal line and one vertical line for each number between 0 and 10 (inclusive) in a grey color.

Afterwards, **add the text labels programmatically**. There are a few ways to do this. You could create an array containing the values that need labels (e.g. `[0,5,10]`) and then loop through it with `forEach`. You also could do some clever modulo arithmetic while you are looping to make lines. In either case, make a new `<text>` element for each label with **Arial font**. Use your scales to help place the text and adjust the `dominant-baseline` and `text-anchor` attributes like you did in HW2. You may need to add or subtract a few pixels to position the text nicely.

Now, add **circles** for each point with positions determined by your scales. You don't need to use a data join; it's fine if you just create circles one-by-one in a `forEach` loop. Circles should have a radius of 10px and have a dark color.

3. Instead of a `<p>` element, for this question please create a `` element. For each of the following scales, create a `` sub-element and answer the following questions: *(If you have a color vision deficiency and cannot perceive hues in a color scale in order to answer a subitem, please instead describe what you see.)*

(see next page)

A:



Is this a **sequential** or a **divergent** color scale?

Do you think this an **effective color scale**? Justify your answer in **1-2 sentences**.

B:



This scale is being used to color scatterplot points based on a **numeric data attribute** that captures the positive or negative sentiment of tweets. Values range from -1 to 1, with negative values moving towards yellow (the left side) and positive values moving towards red (the right side). Middle values remain blue. Is this an **effective color scale for this task**? Justify your answer in **1-2 sentences**.

C:



To a majority of individuals, this color scale will appear to vary in both hue and luminosity (greyish blue on the left, lime green on the right). However, the hue channel of this scale is not visible for individuals with certain color vision deficiencies. This poses usability issues. Use an online color blindness image testing tool to identify and list **which kind(s) of anomalous trichromatic and/or dichromatic color vision deficiencies (e.g. deuteranomaly)** would cause a loss of perceivable hue variation (file included in ZIP).

[If you have color vision deficiencies that make the scale's hue hard to interpret, you have two choices: You can either a) self-disclose your color vision deficiency and describe what the image looks like to you, or b) ask a trusted friend to describe what they see to you.]

D:



A data scientist is designing a choropleth map for a new **continuous, numeric county-by-county "average life expectancy" data attribute** they developed. **Would you recommend that they use this rainbow scale to color the counties in their map?** Justify your answer in 1-2 sentences.

(see last page for data for problem 2)

Data for scatterplot in #1

	X	Y
p1	1.0	9.0
p2	1.5	6.0
p3	2.5	4.0
p4	4.0	2.0
p5	5.0	1.6
p6	6.0	2.4
p7	7.0	3.0
p8	8.0	3.4
p9	9.0	3.6

For easier import into your code:

```
data = [{"x":1.0 ,"y":9.0},  
        {"x":1.5 ,"y":6.0},  
        {"x":2.5 ,"y":4.0},  
        {"x":4.0 ,"y":2.0},  
        {"x":5.0 ,"y":1.6},  
        {"x":6.0 ,"y":2.4},  
        {"x":7.0 ,"y":3.0},  
        {"x":8.0 ,"y":3.4},  
        {"x":9.0 ,"y":3.6}]
```