

Let's make a scatterplot matrix!

In this lab you will be working with multivariate data, data with multiple variables, and you will look for relationships between the different variables. You will do this by making a scatterplot matrix. What is a scatterplot matrix? It is a matrix of scatterplots! Basically, it is a quick way to look at the relationship between pairs of variables. Feel free to look up scatterplot matrices on the Internet before going on.

We will continue to work with Dr. Johnson's stream data. Start by downloading the newest version of the data from Moodle (with the conductance variables corrected).

Setup Lab 3

Open NetBeans and create a New Project for lab 3 with a `.html`, `.css`, and `.js` file.

Using Online Code

As you know, there are many d3.js projects on the web that you may use as a basis for assignments. In this lab we will start with the scatterplot matrix code found here

<https://b1.ocks.org/Fil/6d9de24b31cb870fed2e6178a120b17d>.

We will be using this to make scatterplots comparing discharge, temperature, and conductance.

- Start by copying the code in the `<style>` brackets into your `.css` file, and the code in the `<script>` brackets into your `.js` file.
- Our matrix will be 3×3 instead of their 4×4 matrix. Start by updating the `size` of each plot to be 320 (1/3 of 960).
- Update the `color` to use a color scheme with 20 colors since you have 20 streams.
<http://b1.ocks.org/aaizemberg/78bd3dade9593896a59d>.
- Read in the correct data file.
- Now we need to get our data into a form we can use. I know this is your favorite part!
 - We need to minimize the data so that we can plot all of it. We will first try to minimize the data of interest by averaging the data for each day for each stream. You will need to use `d3.nest` with *two* key values. Set your keys as the `location_id` and the `date`. An object in your final array will be in the form:

```
Key: "1"
values: [{key: "2016-01028",
         value: {discharge: 0.31, temperature: 48.53, conductance: 0},
         {key: ...
```

- There are cases where you have data values with no entries, for example in conductance, you can turn these values to null using the following conditional (ternary) operator¹:

```
function(d){ return (d === "") ? null : +d.conductance;}
```

This code is equivalent to an if-else statement:

```
if(d === "")
    return null;
else
    return +d.conductance;
```

and a version exists in many languages. Take a look at this Wikipedia page:

<https://en.wikipedia.org/wiki/\%3F>:

- To make your life easier, you may want to un-nest your data and put each piece of data on the same level. `d3.merge` will merge together multiple arrays into one larger array. The following code will let you merge together all the data that you need:

```
dataIwant = d3.merge(nestedData.map(function(d){
    return d.values.map(function(v){
        return {
            //define all of your variables here

        };
    })
}));
```

- define the `traits` that you care about in your data using the starter code as a guide. You will want to eliminate the headers that are not of interest.
- make sure that your "circles" are using the correct data.

¹https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Conditional_Operator

- Finally, make some changes to your `.css` file to adjust the appearance of the hidden circles. There is so much data that it is hard to really see what is going on. In `circle.hidden` make `fill none`, set `stroke` to a light gray color, and set `stroke-opacity` (range is from 0 - transparent, to 1 - opaque) close to transparent.

Scatterplot matrix II

Once you have a scatterplot matrix, create a second version where you filter or calculate the data in a different way. Here are some options, but don't let this limit you:

- One season
- Values in the top 75%
- Logarithmic scales instead of linear
- Dates since Hurricane Harvey

Submission

Add your completed lab 3 pages to your Lab 3 page on your Davidson Domain.