Lecture Exercise 5 (2/9)

Submit your team number

Question *Submitted Feb 9th 2023 at 8:31:11 pm*

Please enter your team number.

50

1. Draw a scatterplot using a CSV file

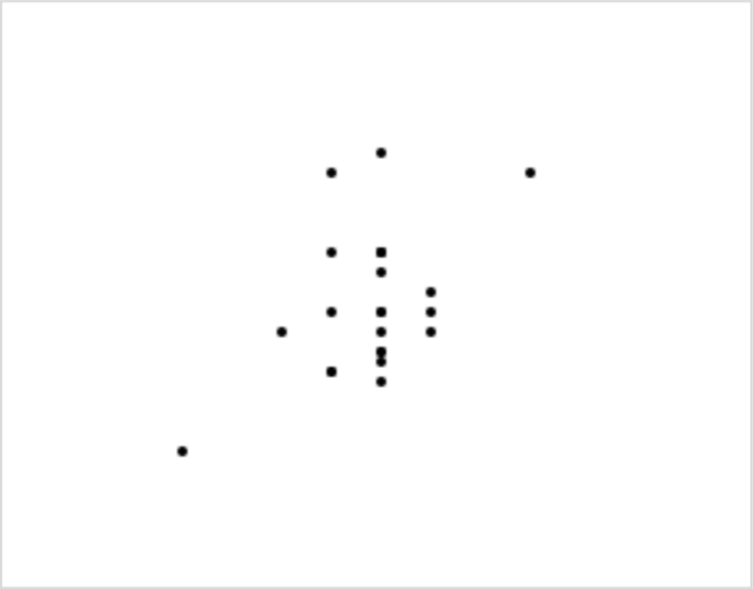
1. Create **exercise5.html**. Make sure to import d3.min.js as well.

2. Load/process the dataset that we used for Lab 3 (cereals.csv). You can refer the code from Lab 3 too.

3. Copy/paste the following code and fill in your code to generate a scatterplot between Calories and Carbo. The x- and y- axis indicate Calories and Carbo respectively.

```
/* Draw scatterplot with Calories and Carbo */
const cereals = d3.csv("cereals.csv", rowConverter);
cereals.then(function(data) {
    let svgwidth = 300;
    let svgheight = 300;

    let svg = d3.select("body").append("svg")
        .attr("width", svgwidth)
        .attr("height", svgheight);

    // Question 2-1: add your code below to draw a scatterplot

});
```

The output scatter plot should look something like below.



Question *Submitted Feb 9th 2023 at 8:47:46 pm*

Copy and paste your code for the question.

```
svg.selectAll(".circlea1").data(data).enter().append("circle").attr("cx", d => 2*d.Calories).attr("cy",d
=> (svgheight - 10*d.Carbo)).attr("r","1px");
```

2. Draw a Scatterplot with Calories and Carbo

Now we want to encode another attribute, Manufacturer, to the current scatterplot. Recall the channel rankings framework we learned in the last lecture: Manufacturer is an unordered attribute, so we are going to use the **color-hue** channel.

Encode Manufacturer by 1) setting different color hues for each manufacturer and 2) set the opacity to be 0.5 (in case some dots are overlaid on top of others).

Question *Submitted Feb 9th 2023 at 8:55:27 pm*

Copy and paste your code that selects different colors for different manufacturers.

```
let ordinalScale = d3.scaleOrdinal()

 .domain(d3.map(data, function(d){return d.Manufacturer;})

 .keys()).range(d3.schemePaired);

 svg.selectAll(".circlea1").data(data).enter().append("circle").attr("cx", d => 2*d.Calories).attr("cy",d => (svgheight - 10*d.Carbo)).attr("r","1px").attr("opacity","0.5").attr("fill", d => ordinalScale(d.Manufacturer));
```

3. Add dynamic scaling to a scatterplot

Define xScale using `d3.scaleLinear()` which will be used for the x-axis, Calories. You may find these two D3 methods useful when defining the domain and range.

- d3.min() extracts all Calories values from the data then finds the minimum
- d3.max() uses the same scheme as d3.min() to find the maximum

Question 1 *Submitted Feb 9th 2023 at 9:05:29 pm*

Copy and paste your definition of xScale.

```
let scaleLinearX = d3.scaleLinear()

 .domain([d3.min(data, d => d.Calories),d3.max(data, d => d.Calories)])

 .range([paddingtonBear, svgwidth-paddingtonBear]);
```

Question 2 *Submitted Feb 9th 2023 at 9:05:26 pm*

You might have noticed that nothing changes on the resulting html. This is because you defined the new scale but have not adapted it to your chart yet. Update the “cx” value of each circle element properly so that it uses xScale.

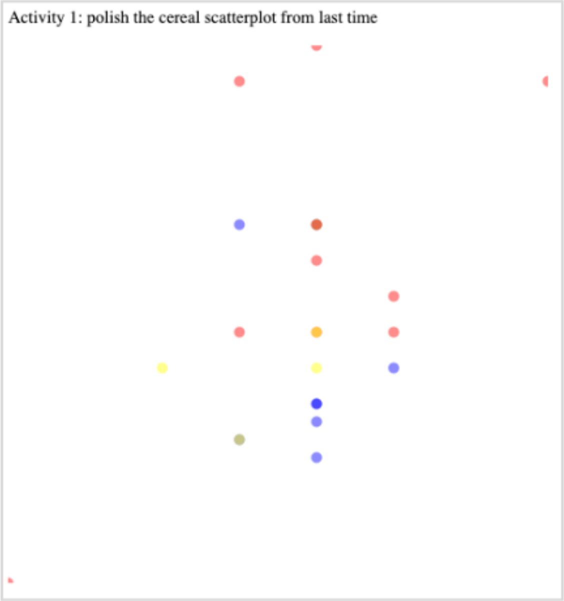
Copy and paste your code line with updated .attr(“cx”, ...)

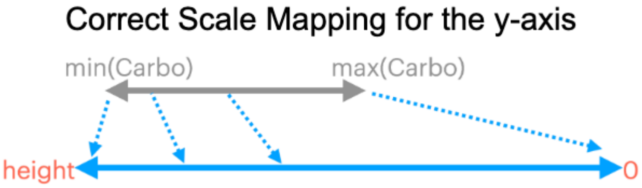
```
svg.selectAll(".circlea1").data(data).enter().append("circle").attr("cx", d =>
scaleLinearX(d.Calories)).attr("cy",d =>
scaleLinearY(d.Carbo)).attr("r","3px").attr("opacity","0.5").attr("fill", d =>
ordinalScale(d.Manufacturer));
```

Question 3 *Submitted Feb 9th 2023 at 9:05:53 pm*

Apply dynamic scaling along the y-axis too, by defining the yScale variable. After applying dynamic scaling to both x- and y-axis, the distribution of the point marks should look exactly as below. If different, likely your scaling is not done properly.

(Hint) the y-axis grows downward, not upward. In other words, the bottom dots have high y-coordinates on the <svg> canvas. The visualization of the correct mapping below would help.





Copy and paste your corresponding code below.

```
let scaleLinearY = d3.scaleLinear()

 .domain([d3.max(data, d => d.Carbo),d3.min(data, d => d.Carbo)])

 .range([paddingtonBear, svgheight-paddingtonBear]);
```

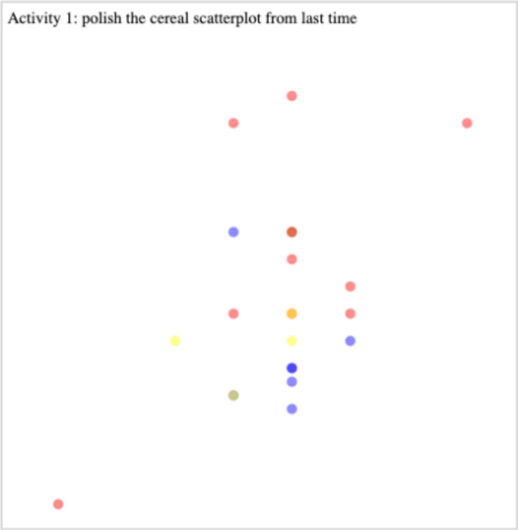
Question 4 *Submitted Feb 9th 2023 at 9:06:23 pm*

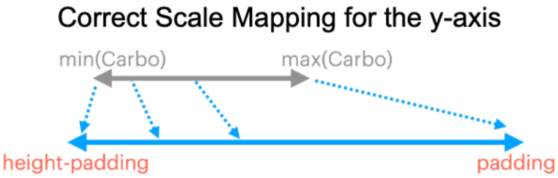
Notice that on the current scatterplot some circles around the edges are only partially shown. The reason is that the min(or max) cx/cy values are mapped to the smallest (or largest) pixel in the canvas, which hides some part of the circles. You can fix that too by adding some buffer space, called padding, around the edges.

Define a variable **padding**, and use it to update range() in xScale and yScale.

The new output should look as below. No points are hidden!

(Hint) The visualization below would help you find the answer for yScale. And you can apply the same strategy for xScale.




Correct Scale Mapping for the y-axis

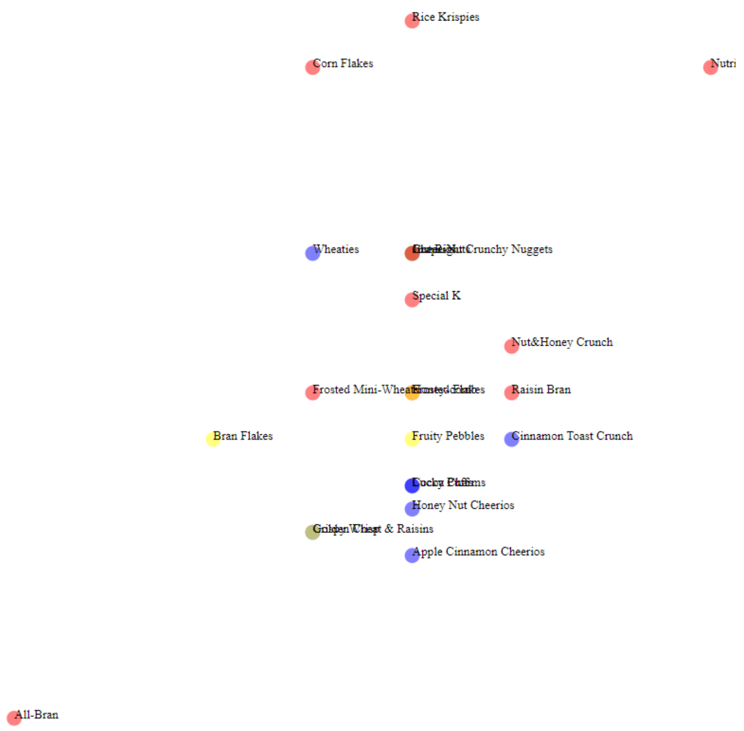
Copy and paste the new definition of xScale and yScale.

```
var paddingtonBear = 5;



 let scaleLinearX = d3.scaleLinear()

 .domain([d3.min(data, d => d.Calories),d3.max(data, d => d.Calories)])

 .range([paddingtonBear, svgwidth-paddingtonBear]);

 let scaleLinearY = d3.scaleLinear()

 .domain([d3.max(data, d => d.Carbo),d3.min(data, d => d.Carbo)])

 .range([paddingtonBear, svgheight-paddingtonBear]);
```

4. Add a Label to the Scatterplot

Add labels of cereal names, on the bottom-right with font size "5px" of each data point.

Your output should be similar to:



Question *Submitted Feb 9th 2023 at 9:13:50 pm*

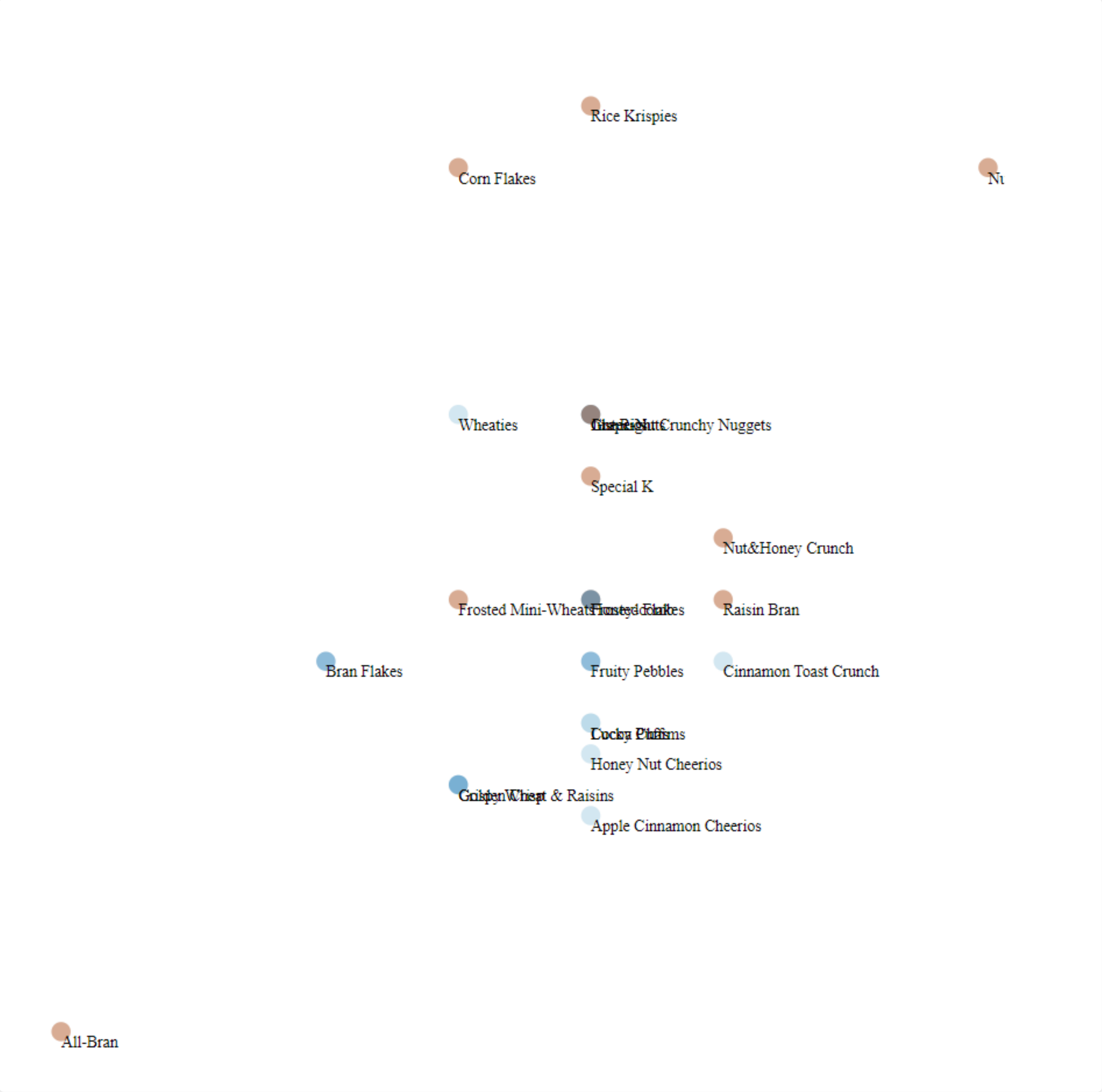
Copy and paste the part of your code which adds labels of cereal names.

```
svg.selectAll(".dumb").data(data)

 .enter()

 .append("text")

 .attr("class", "dumb")

 .attr("x", d => scaleLinearX(d.Calories))

 .attr("y", d => scaleLinearY(d.Carbo)+paddingtonBear)

 .text(d => d.Name)

 .attr("font-size","5px")
```

Upload Your Files

Question 1 *Submitted Feb 9th 2023 at 9:15:43 pm*

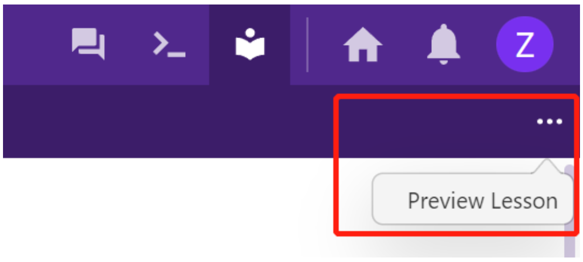
Upload the screenshot of your resulting webpage. You will need to click the "clip" button to upload a file into the Answer box.



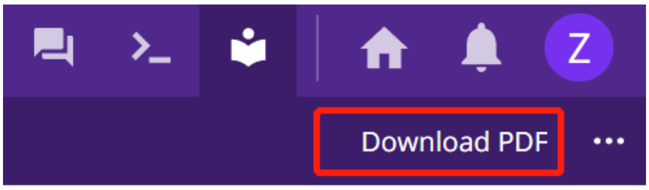
Question 2 *Submitted Feb 9th 2023 at 9:15:51 pm*

You need to download the PDF of lecture exercise 3 and upload it with other files to the Gradescope. Follow the instructions on how to download PDF file:

1. Click on the ellipsis button and the Preview Lesson.



2. After that, click on the Download PDF button.



● PDF downloaded!

○ Haven't done yet!

Question 3 *Submitted Feb 9th 2023 at 9:15:50 pm*

Upload the following files to Gradescope. You need to make **a group submission,adding all present members in your team**, so that the present members get the participation credit.

Files to upload:

- exercise5.html
- PDF you downloaded as Q2

● Our team uploaded the the files on gradescope!

○ Oops, our team did not upload the files on gradescope!

Feedback

Question *Submitted Feb 9th 2023 at 9:16:01 pm*

Was the activity today clear? If not, please share how the course can improve it. Your comments will help us design future lab content (and also future students).

hehe