**Basic Cursors**

# **Ask**

* Should be able to render svg image as cursor
* On click event, cursor should be rendered at the nearest data set value
* Should be able to click around the line chart as well
* Should be able to show data set value associated with the cursor
* Should display indicators on x-axis and y-axis which represents data set value, associated with the cursor.
* Should be able to enable/disable cursor display
* When multiple plots are rendered, should be able to show cursors associated with each plot
* Should be able to show cursor A and B associated with a single line chart and show the cursor difference

# **Approach**

### **Should be able to render svg image as cursor**

Should append image element, with image path as xlink:href attribute

 .append("image")

 .attr("xlink:href",'../../assets/ACursorActive.svg')

### **On click event, cursor should be rendered at the nearest data set value**

We can get x value associated with the click event, using

d3.pointer(event)[0]

Using d3.bisectCenter function, we can get i(index) of the data set value whose x value is closer to the x value associated with the click event. Post getting the index, x, y attributes associated with the cursor are modified

this.basic\_cursorA.attr("x",this.x(SpectrumData[i].x\_value)-10) .attr("y", this.y(SpectrumData[i].y\_value)-15)

### **Should be able to click around the line chart as well**

Added click event to the svg

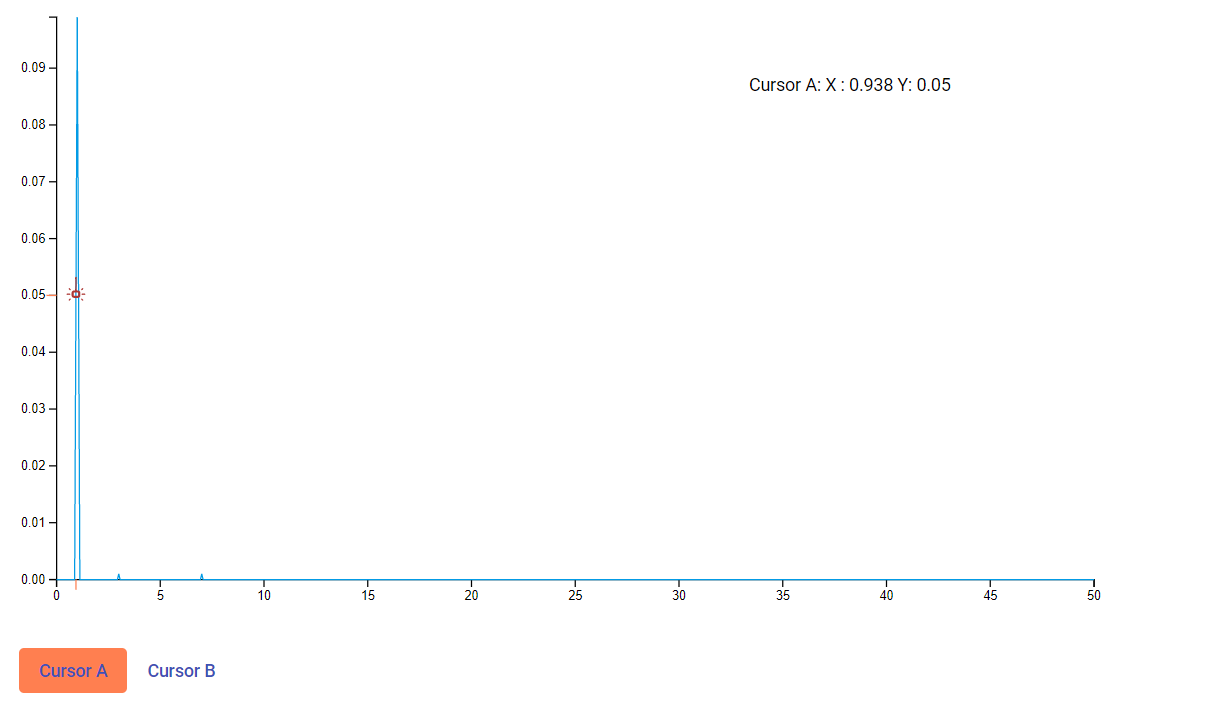
   .on("click", (event:any)=>this.mouseClick(event))

### **Should be able to show data set value associated with the cursor**

Rendered a div element, which is used to display the cursor value. On click, the text associated with the div is updated with the data set value of index i, which we get from bisectCenter function.

const text = d3.select('.readoutA');

 text.text(` Cursor A: X : ${SpectrumData[i].x\_value} Y: ${SpectrumData[i].y\_value} `);



### **Should display indicators on x-axis and y-axis which represents data set value, associated with the cursor**

Appended g elements with data set values as x,y attributes. Initially we are considering data set value associated with 80 index

this.svg.select(".xAxis").selectAll(".AX")

 .data([1])

 .enter()

.append('g').attr('class', 'AX')

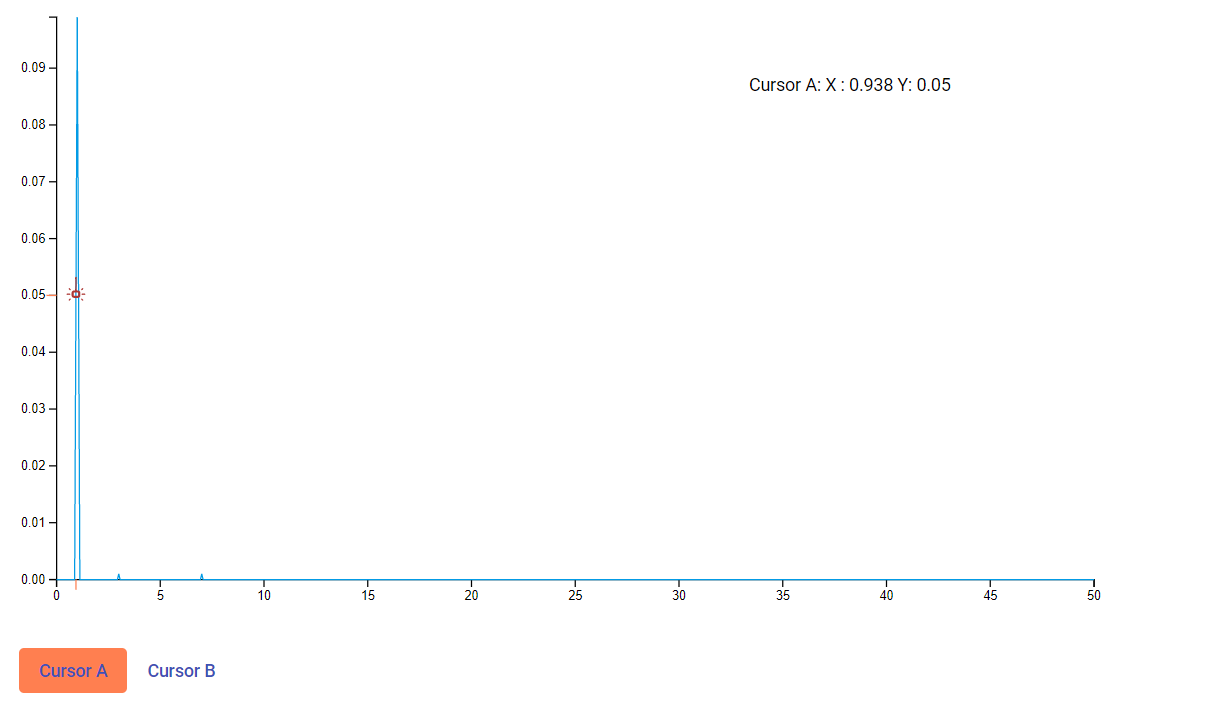
. attr("transform", "translate(" + this.x(SpectrumData[80].x\_value) + "," + 0 + ")").append('line').attr("stroke", "coral").attr("y2",8)

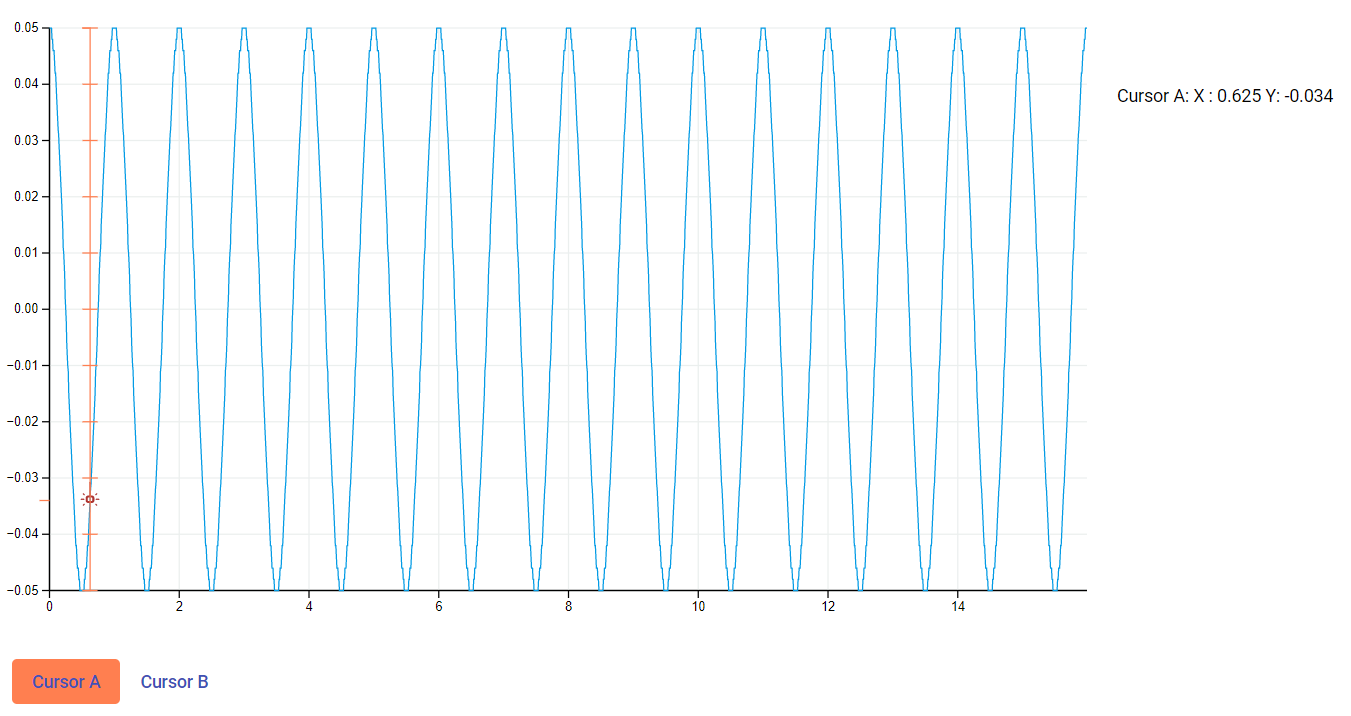
 this.svg.select(".yAxis").selectAll(".AY")

 .data([1])

 .enter().append('g').attr('class', 'AY'). attr("transform", "translate(" + 0 + "," + this.y(SpectrumData[80].y\_value) + ")").append('line').attr("stroke", "coral").attr("y2",8) .attr('transform', 'rotate(90)')

On click event, we are modifying these x, y values based on the index, which we get from bisectCenter function.





### **Should be able to enable/disable cursor display**

To achieve this, we have used a button

<button mat-button color="primary" (click)="handleCursorAButtonClick()" id="cursorA">Cursor A</button>

For enabling:

To enable, we have to render a cursor

his.svg.append('g')

            .selectAll(".cursorA")

            .data(this.dataSet)

            .enter()

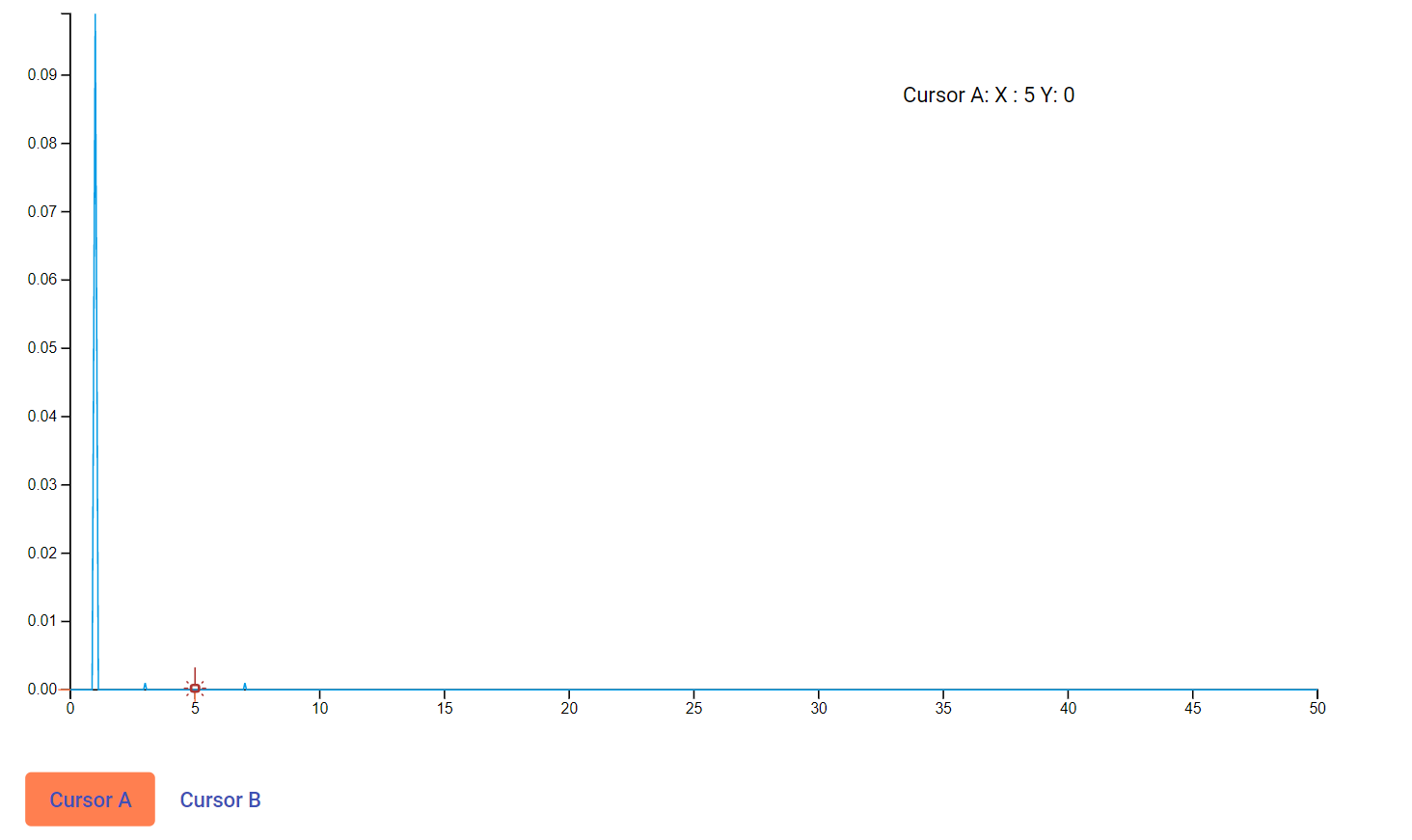
            .append("image")

            .attr('class','.cursorA')

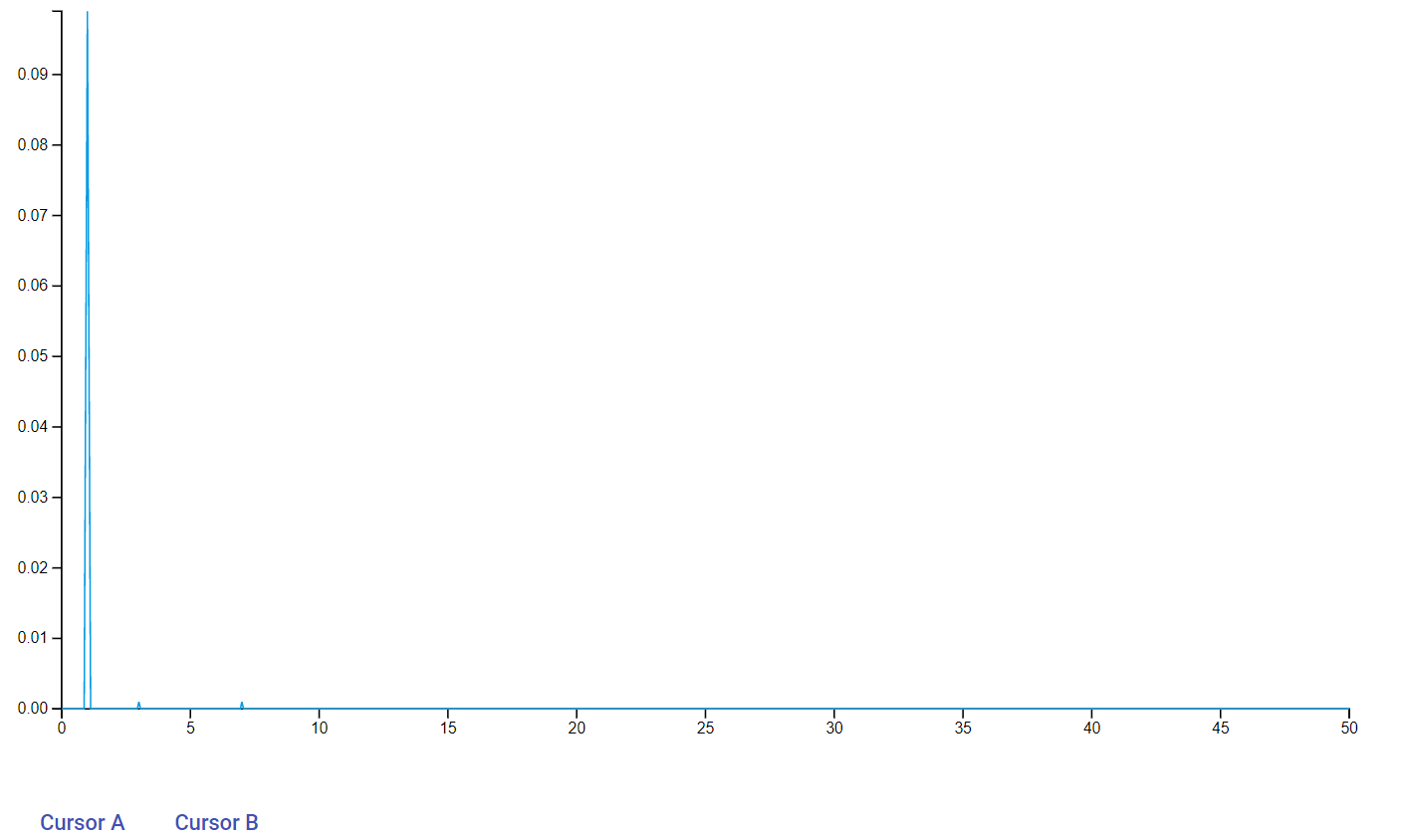
            .attr("xlink:href",'../../assets/cursorA.svg')

            .style("width","20px")

            .style("height","20px")



For disabling: We can use .remove()



### **When multiple plots are rendered, should be able to show cursors associated with each plot**

this.basic\_cursorA=  this.svg.append('g')

            .selectAll(".cursorA")

            .data(this.dataSet)

            .enter()

            .append("image")

            .attr('class','.cursorA')

            .attr("xlink:href",'../../assets/cursorA.svg')

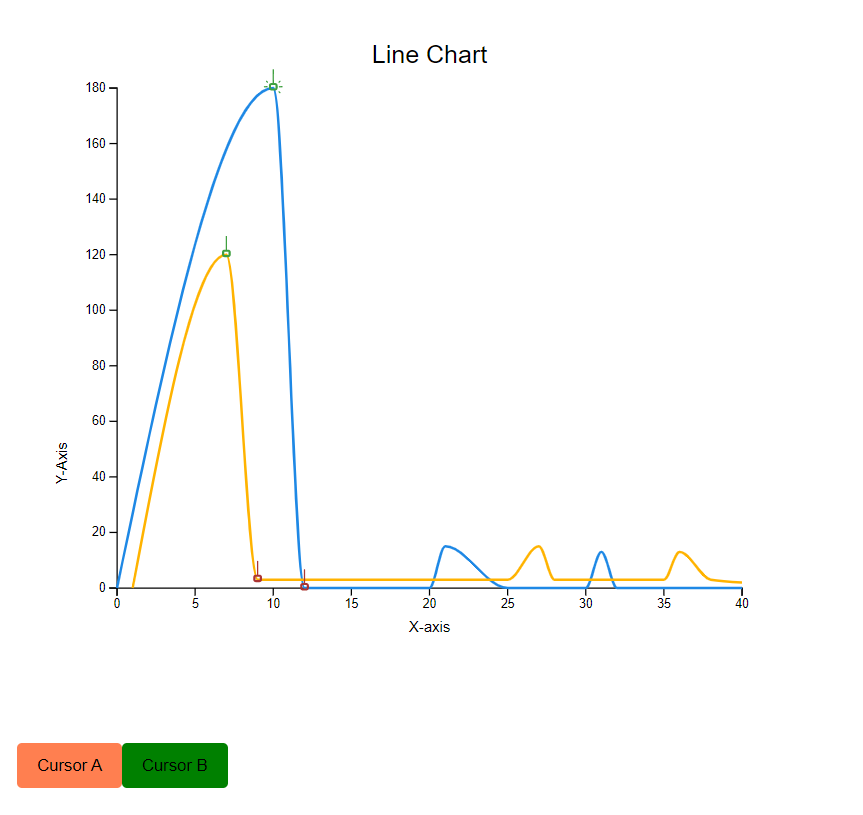
            .style("width","20px")

            .style("height","20px")

             .attr('x',function (d) { return xScale(d[0][0])-10; })

            .attr('y',function (d) { return yScale(d[0][1])-15; })

We have used function to set x,y values of cursors with respect to the line chart they are associated with



### **Should be able to show cursor A and B associated with a single line chart and show the cursor difference**

<button mat-button color="primary" (click)="handleCursorBButtonClick()" id="cursorB">Cursor B</button>

We have a button associated with cursor B, similar to the one we had for enabling disabling cursor A.

For enabling:

  this.basic\_cursorB= this.svg

    .selectAll(".cursorB")

    .data([1])

    .enter()

    .append('g')

    .append("image")

    .attr("xlink:href",'../../assets/BCursorActive.svg')

    .style("cursor", "pointer")

    .on("click", (event:any)=>this.cursorBClick(event))

    .style("width","20px")

    .style("height","20px")

    .attr('x',this.x(SpectrumData[85].x\_value)-10)

    .attr('y',this.y(SpectrumData[85].y\_value)-15)

For disabling we can use .remove()

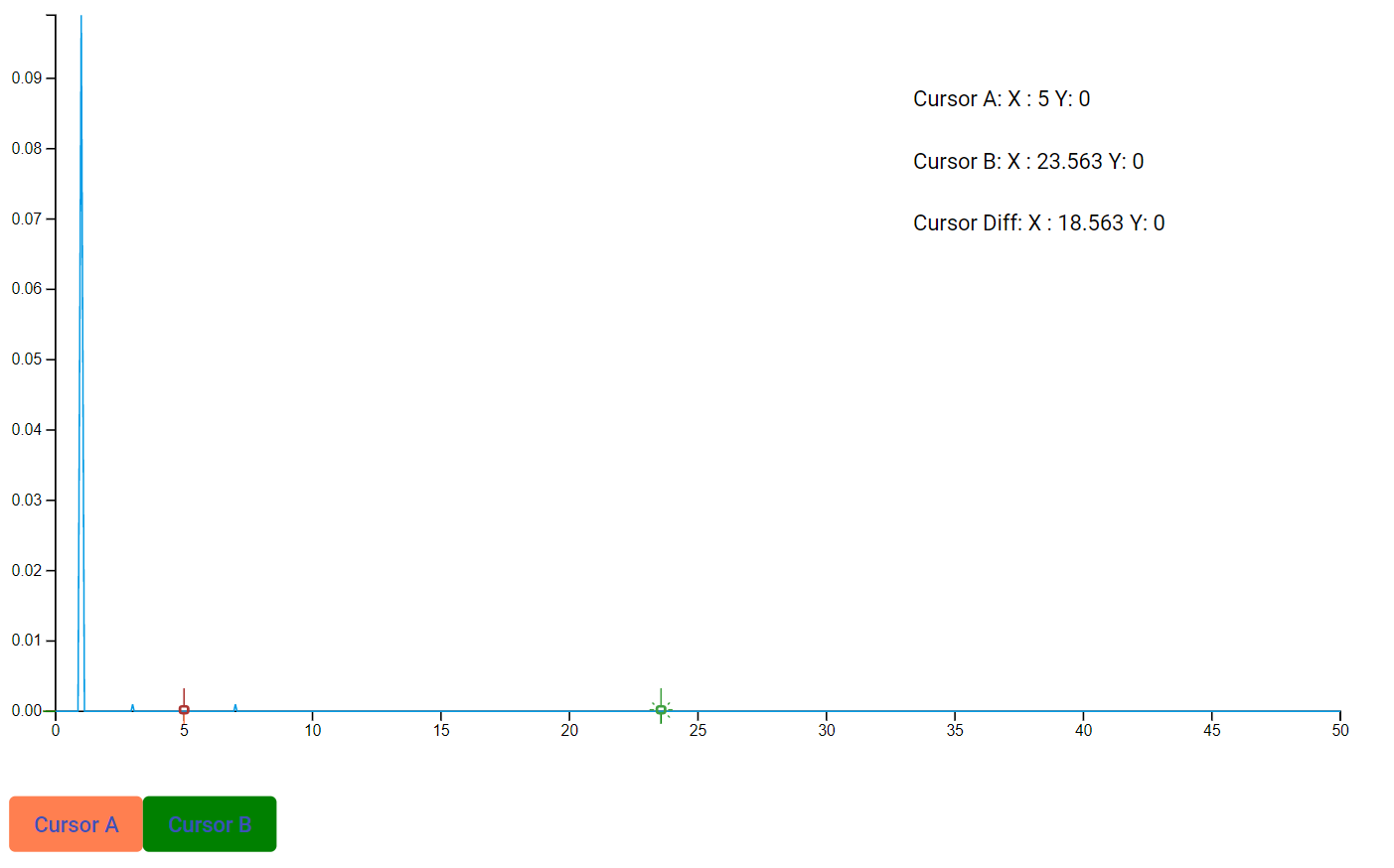
For cursor difference:

We have a div, to show this value

<div class="cursorDiff"></div>

We are modifying the text associated with this div, whenever required

 d3.selectAll(".cursorDiff").text(` Cursor Diff: X : ${Math.abs(this.cursorA\_value.x-this.cursorB\_value.x)} Y: ${Math.abs(this.cursorA\_value.y-this.cursorB\_value.y)} `);



# **Conclusion**

* Using d3 js, we can achieve all the goals specified with respect to basic cursors spike.

# **References**

* <https://www.geeksforgeeks.org/d3-js-bisectcenter-method/>
* <https://www.educative.io/edpresso/how-to-create-a-line-chart-using-d3>
* https://github.com/BentlyNevada-bh/spike-plots-d3js/tree/basic-cursors