CMPT 384 – Information Visualization

D3.js

Lab 3

Course Instructor: Debajyoti Mondal

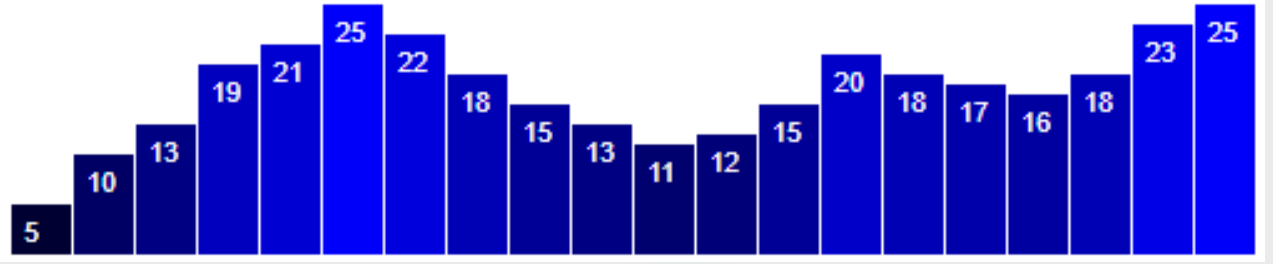
Lab Tutorial Instructor : Arman Heydari(arman.heydari@usask.ca)

Agenda

- ✓ D3 CSV Read
- ✓ D3 Circle Draw
- ✓ D3 max, min function
- ✓ D3 Scales – Linear scale
- ✓ D3 Axis
- ✓ D3 Transformation
- ✓ D3 Scatter Plot
- ✓ D3 Circle Draw
- ✓ D3 Text Element

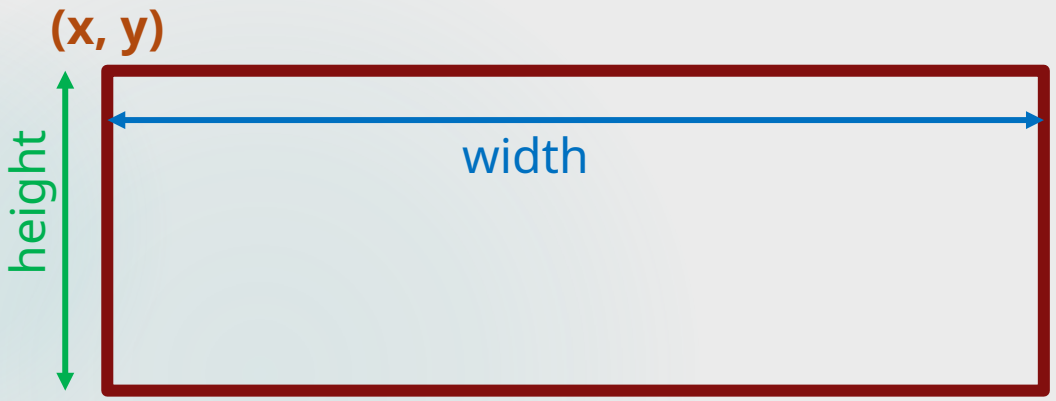
- D3 Rectangle Draw
- D3 Bar Chart with (hands-on)
  - Data labels
  - Color Scheme

```
var dataset = [ 5, 10, 13, 19, 21, 25, 22,
18, 15, 13, 11, 12, 15, 20, 18, 17, 16,
18, 23, 25 ];
```



D3 Rectangle Draw

- Rectangle is another primary shape



Required Attributes

- Top-Left Co-ordinate (x, y)
- width
- height

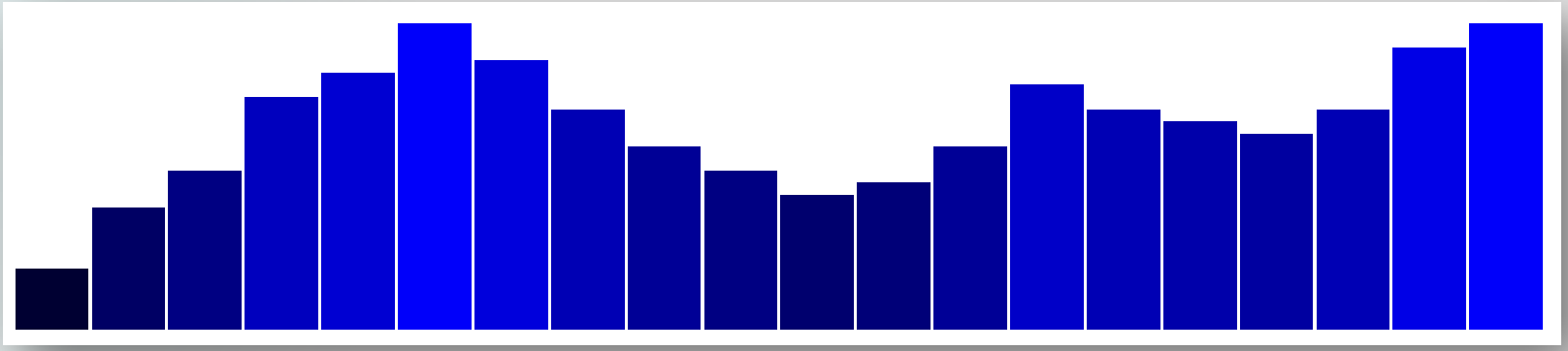
```
<rect x="50" y="20" width="150" height="150“></
rect>
```

Start the server

```
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\> python -m SimpleHTTPServer 8888
C:\Users\jyoti\AppData\Local\Programs\Python\Python37-32\python.exe: No
module named SimpleHTTPServer

C:\> python -m http.server 8888
Serving HTTP on 0.0.0.0 port 8888 (http://0.0.0.0:8888/) ...
```

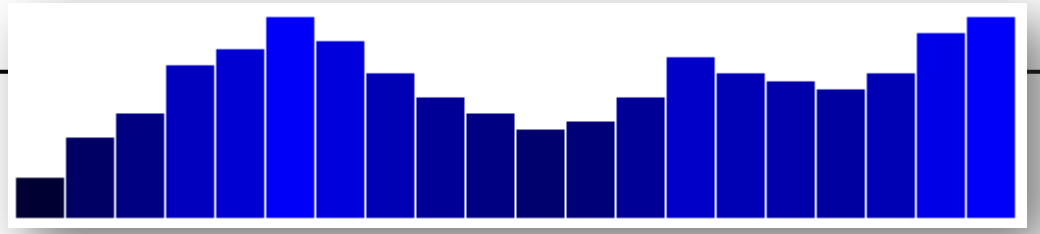
Display Static Data

```
var w = 500; var h = 100; var barPadding = 1;
var dataset = [ 5, 10, 13, 19, 21, 25, 22, 18, 15, 13, 11, 12, 15, 20,
18, 17, 16, 18, 23, 25 ];

//Create SVG element
var svg = d3.select("body") .append("svg")
            .attr("width", w).attr("height", h);
```



Bar Chart

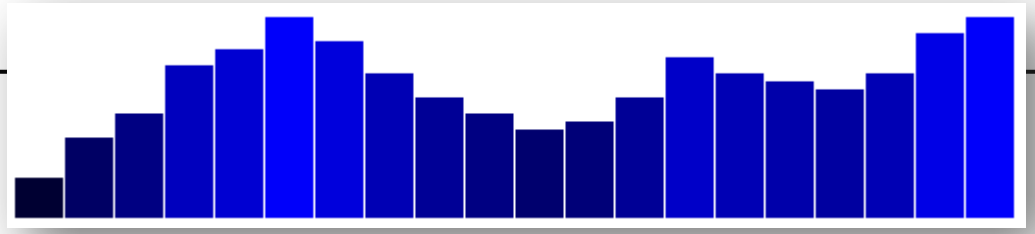
```
var w = 500; var h = 100; var barPadding = 1;
var dataset = [ 5, 10, 13, 19, 21, 25, 22, 18, 15, 13, 11, 12, 15, 20, 18, 17, 16, 18, 23, 25 ];

//Create SVG element
var svg = d3.select("body") .append("svg")
                    .attr("width", w).attr("height", h);
```



```
svg.selectAll("rect")
      .data(dataset)
      .enter()
      .append("rect")

      .attr("x", function(d, i) { return i*(w/dataset.length); })
      .attr("y", function(d) { return h-(d*4); })

      .attr("width", w/dataset.length - barPadding)
      .attr("height", function(d) { return d*4; })

      .attr("fill", function(d) {
                  return "rgb(0,0,"+Math.round(d*10) + ")"; });
```

Bar Chart

```
var w = 500; var h = 100; var barPadding = 1;
var dataset = [ 5, 10, 13, 19, 21, 25, 22, 18, 15, 13, 11, 12, 15, 20, 18, 17, 16, 18, 23, 25 ];

//Create SVG element
var svg = d3.select("body") .append("svg")
                    .attr("width", w).attr("height", h);
```



```
svg.selectAll("rect")
      .data(dataset)
      .enter()
      .append("rect")

      .attr("x", function(d, i) { return i*(w/dataset.length); })
      .attr("y", function(d) { return h-(d*4); })

      .attr("width", w/dataset.length - barPadding)
      .attr("height", function(d) { return d*4; })

      .attr("fill", function(d) {
                  return "rgb(0,0,"+Math.round(d*10) + ")"; });
```

Bar Chart

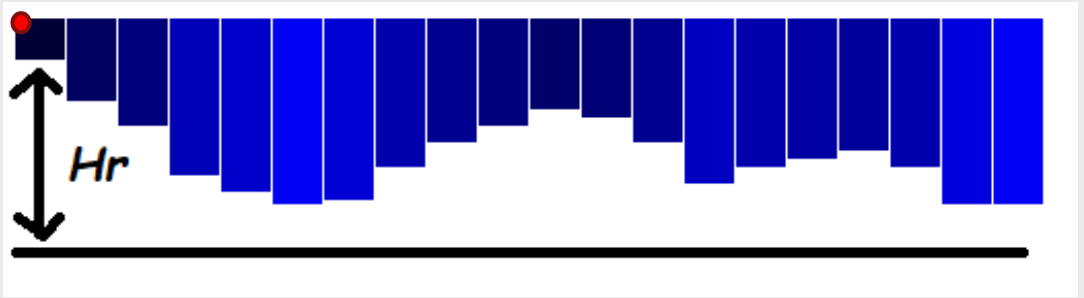


Figure 1

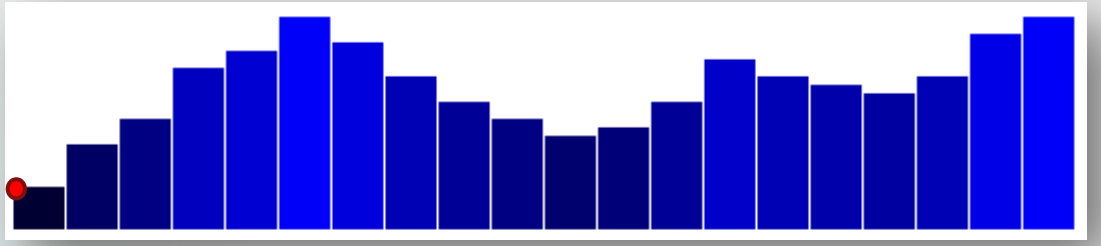


Figure 2

- The origin is in the top left corner in SVG by default so drawing a set of rectangles with different heights would give Figure 1 which is not what we want.
- So we change the Y values from 0 to a dynamic value.
- Then we invert the plot , we do this by pushing each rectangle downwards by Hr

$$Hr = h - (d*4)$$

Bar Chart

```
var w = 500; var h = 100; var barPadding = 1;
var dataset = [ 5, 10, 13, 19, 21, 25, 22, 18, 15, 13, 11, 12, 15, 20, 18, 17, 16, 18, 23, 25 ];

//Create SVG element
var svg = d3.select("body") .append("svg")
                    .attr("width", w).attr("height", h);
```

```
svg.selectAll("rect")
      .data(dataset)
      .enter()
      .append("rect")

      .attr("x", function(d, i) { return i*(w/dataset.length); })
      .attr("y", function(d) { return h-(d*4); })

      .attr("width", w/dataset.length - barPadding)
      .attr("height", function(d) { return d*4; })

      .attr("fill", function(d) {
                  return "rgb(0,0,"+Math.round(d*10) + ")"; });
```

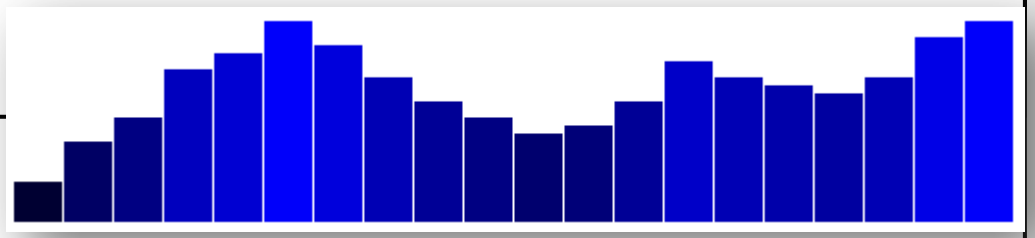
Bar Chart

```
var w = 500; var h = 100; var barPadding = 1;
var dataset = [ 5, 10, 13, 19, 21, 25, 22, 18, 15, 13, 11, 12, 15, 20, 18, 17, 16, 18, 23, 25 ];

//Create SVG element
var svg = d3.select("body") .append("svg")
                    .attr("width", w).attr("height", h);
```

```
svg.selectAll("rect")
      .data(dataset)
      .enter()
      .append("rect")

      .attr("x", function(d, i) { return i*(w/dataset.length); })
      .attr("y", function(d) { return h-(d*4); })

      .attr("width", w/dataset.length - barPadding)
      .attr("height", function(d) { return d*4; })

      .attr("fill", function(d) {
                  return "rgb(0,0,"+ d*10 + ")"; });
```

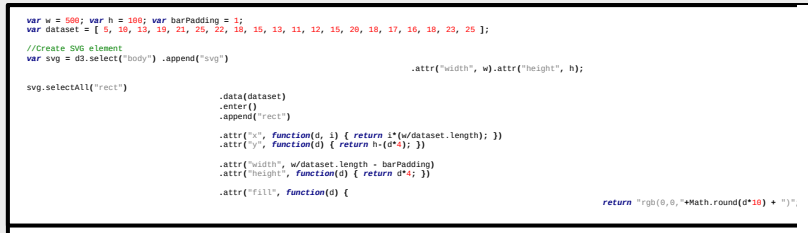
Bar Chart

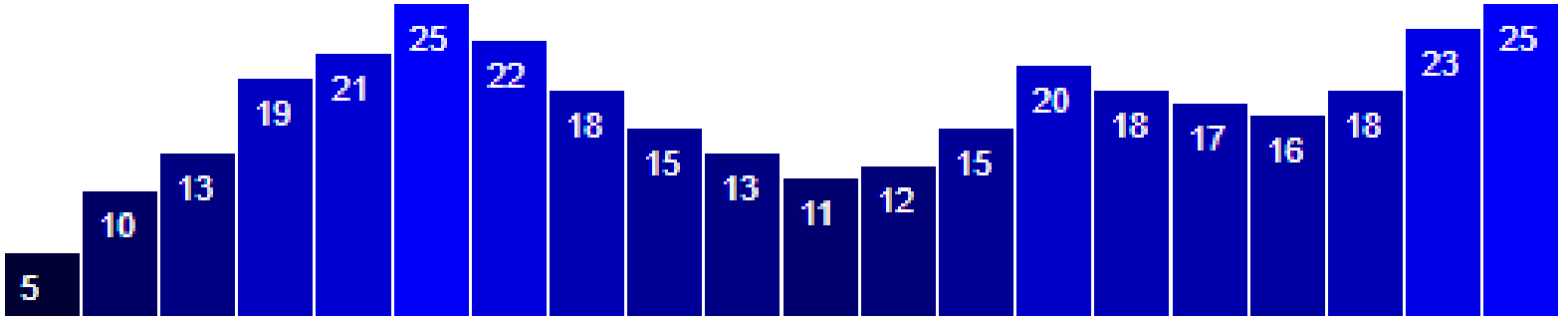
```
var w = 500; var h = 100; var barPadding = 1;
var dataset = [ 5, 10, 13, 19, 21, 25, 22, 18, 15, 13, 11, 12, 15, 20, 18, 17, 16, 18, 23, 25 ];

//Create SVG element
var svg = d3.select("body") .append("svg")
                    .attr("width", w).attr("height", h);
```



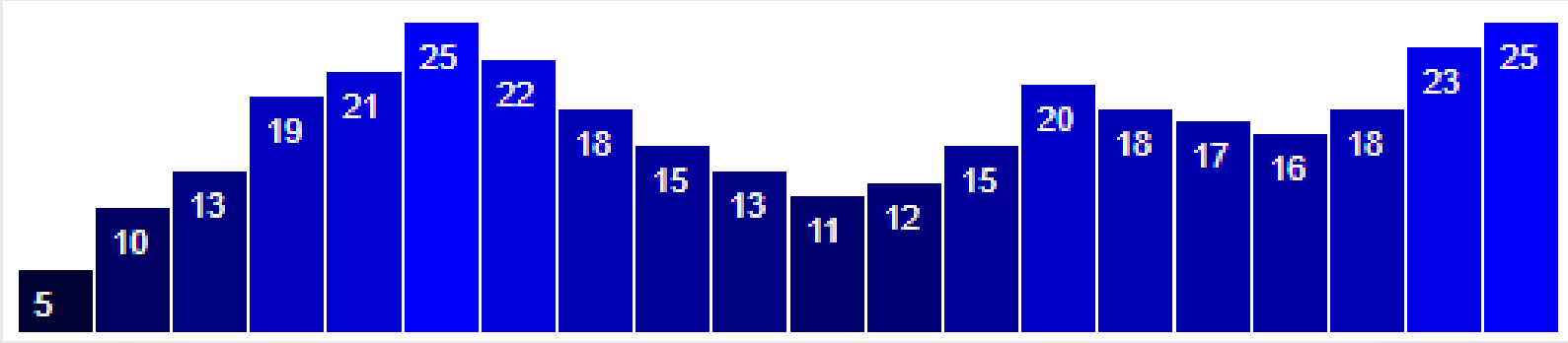
```
svg.selectAll("rect")
      .data(dataset)
      .enter()
      .append("rect")

      .attr("x", function(d, i) { return i*(w/dataset.length); })
      .attr("y", function(d) { return h-(d*4); })

      .attr("width", w/dataset.length - barPadding)
      .attr("height", function(d) { return d*4; })

      .attr("fill", function(d) {
                  return "rgb(0,0,"+ d*10 + ")"; });
```

Bar Chart

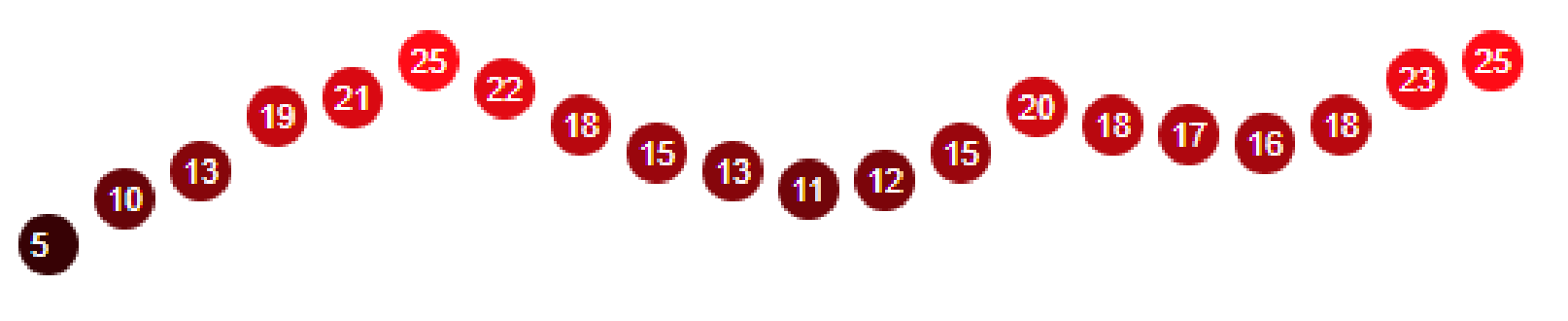





```
svg.selectAll("text")
       .data(dataset)
       .enter()
       .append("text")
       .text(function(d) { return d; })

       .attr("x", function(d,i) {return i*(w/dataset.length)+5; })
       .attr("y", function(d) { return h-(d*4)+15; })

       .attr("font-family", "sans-serif")
       .attr("font-size", "11px")
       .attr("fill", "white");
```



Try to modify the same code to get the graph below



If you cannot figure out the solution yourself , look at example_challenge.html for help.