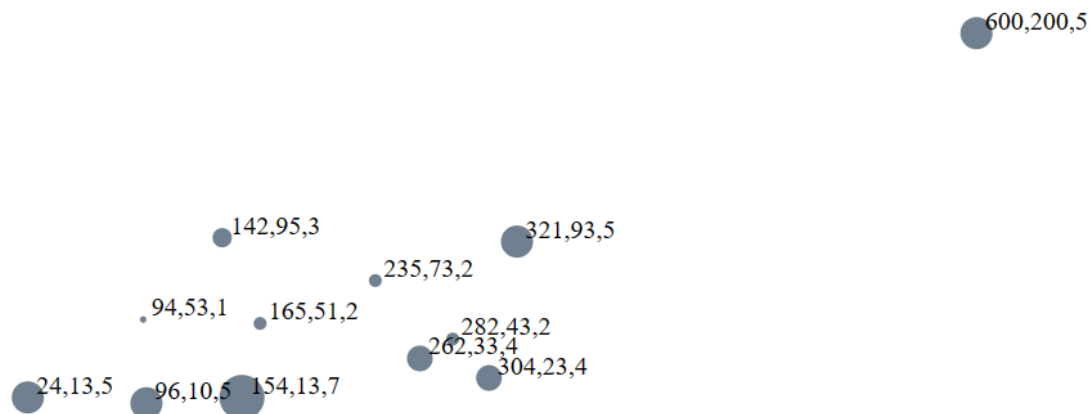


Task 3.1

The D3 Journey Start Here



COS30045 Data Visualisation
Toan Nguyen

Fig1. Output

```
1 <!DOCTYPE html>
2 <html lang = "en">
3 <head>
4   <meta charset = "utf-8" />
5   <meta name = "description" content = "Data Visualisation"/>
6   <meta name = "keywords" content = "HTML,CSS,D3" />
7   <meta name = "author" content = "Toan Nguyen" />
8
9   <title> Task 3.1 Drawing with data</title>
10
11   <script src = "https://d3js.org/d3.v7.min.js" ></script>
12
13 </head>
14
15 <body>
16   <h1> The D3 Journey Start Here </h1>
17   <script src="scripts.js"></script>
18   <br>
19   <bf>
20     <footer style = "color:grey"> COS30045 Data Visualisation<br>
21     Toan Nguyen</footer>
22 </body>
23
24 </html>
```

Fig2. 3.1 HTML code

```

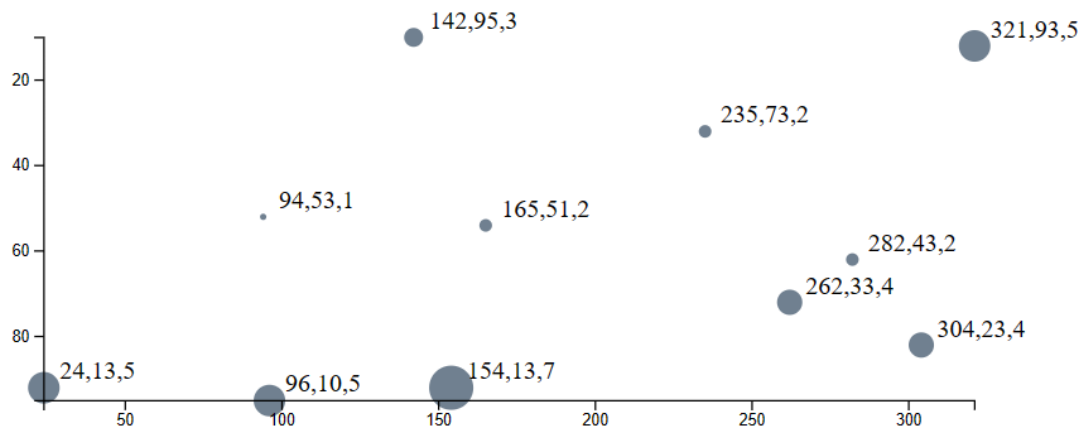
1  var w = 700;
2  var h = 300;
3  var padding = 35;
4
5  var dataset = [
6    [142,95,3],
7    [262,33,4],
8    [94,53,1],
9    [282,43,2],
10   [165,51,2],
11   [96,10,5],
12   [24,13,5],
13   [304,23,4],
14   [154,13,7],
15   [321,93,5],
16   [235,73,2],
17   [600,200,5]
18 ];
19
20 var xScale = d3.scaleLinear()
21   .domain([d3.min(dataset, function (d) {
22     return d[0];
23   }),
24     d3.max(dataset, function (d){
25       return d[0];
26     })])
27   .range([padding, w - (padding + 40)]);
28
29 var yScale = d3.scaleLinear()
30   .domain([d3.min(dataset, function (d) {
31     return d[1];
32   }),
33     d3.max(dataset, function (d){
34       return d[1];
35     })])
36   .range([padding, h - padding]);
37
38 var svg = d3.select("body")      //Select the body of the document
39   .append("svg")                //append the svg to the the element "body"
40   .attr("height",h)              //svg's height
41   .attr("width", w);             //svg's width
42
43 svg.selectAll("circle")          //select all rectangles
44   .data(dataset)                 //count and prepare dataset
45   .enter()                       //create the space holder for the dataset
46   .append("circle")
47   .attr("cx", function(d,i) {
48     return xScale(d[0]);
49   })
50   .attr("cy", function(d,i) {
51     return h - yScale(d[1]);
52   })
53   .attr("r", function(d) {
54     return d[2]*2;
55   })
56   .attr("fill", "slategray");
57 ;
58
59 svg.selectAll("text")
60   .data(dataset)
61   .enter()
62   .append("text")
63   .text(function(d){
64     return d[0] + "," + d[1] + "," + d[2];
65   })
66   .attr("x", function(d){
67     return xScale(d[0]+5);
68   })
69   .attr("y", function(d) {
70     return h - yScale(d[1]+2);
71   });

```

Fig 3: Task 3.1 JS code

Task 3.2

The D3 Journey Start Here



COS30045 Data Visualisation
Toan Nguyen

Fig 4: Task 3.2 Output

```
1 <!DOCTYPE html>
2 <html lang = "en">
3 <head>
4   <meta charset = "utf-8" />
5   <meta name = "description" content = "Data Visualisation"/>
6   <meta name = "keywords" content = "HTML,CSS,D3" />
7   <meta name = "author" content = "Toan Nguyen" />
8
9   <title> Task 3.2 Drawing with data</title>
10
11   <script src = "https://d3js.org/d3.v7.min.js" ></script>
12
13 </head>
14
15 <body>
16   <h1> The D3 Journey Start Here </h1>
17
18   <script src="scripts.js"></script>
19
20   <br>
21   <bf>
22   <footer style = "color:grey"> COS30045 Data Visualisation<br>
23   Toan Nguyen</footer>
24 </body>
25
26 </html>
```

Fig 5: Task 3.2 HTML code

```

1 var w = 700;
2 var h = 300;
3 var padding = 35;
4
5 var dataset = [
6   [142,95,3],
7   [262,33,4],
8   [94,53,1],
9   [282,43,2],
10  [165,51,2],
11  [96,10,5],
12  [24,13,5],
13  [304,23,4],
14  [154,13,7],
15  [321,93,5],
16  [235,73,2],
17  [600,200,5]
18 ];
19
20 var xScale = d3.scaleLinear()
21   .domain([d3.min(dataset, function (d) {
22     return d[0];
23   }),
24   d3.max(dataset, function (d){
25     return d[0];
26   })])
27   .range([padding, w - (padding + 40)]);
28
29 var yScale = d3.scaleLinear()
30   .domain([d3.min(dataset, function (d) {
31     return d[1];
32   }),
33   d3.max(dataset, function (d){
34     return d[1];
35   })])
36   .range([padding, h - padding]);
37
38 var xAxis = d3.axisBottom()
39   .ticks(5)
40   .scale(xScale);
41 var yAxis = d3.axisLeft()
42   .ticks(5)
43   .scale(yScale);
44
45
46 var svg = d3.select("body")           //Select the body of the document
47   .append("svg")                     //append the svg to the the element "body"
48   .attr("height",h)                 //svg's height
49   .attr("width", w);                //svg's width
50
51 svg.selectAll("circle")              //select all rectangles
52   .data(dataset)                    //count and prepare dataset
53   .enter()                          //create the space holder for the dataset
54   .append("circle")
55   .attr("cx", function(d,i) {
56     return xScale(d[0]);
57   })
58   .attr("cy", function(d,i) {
59     return h - yScale(d[1]);
60   })
61   .attr("r", function(d) {
62     return d[2]*2;
63   })
64   .attr("fill", "slategray");
65
66
67
68 svg.selectAll("text")
69   .data(dataset)
70   .enter()
71   .append("text")
72   .text(function(d){
73     return d[0] + ", " + d[1] + ", " + d[2];
74   })
75   .attr("x", function(d){
76     return xScale(d[0]+5);
77   })
78   .attr("y", function(d) {
79     return h - yScale(d[1]+2);
80   });
81
82 svg.append("g")
83   .attr("transform", "translate(0, " + (h - padding)+ ")")
84   .call(xAxis);
85
86 svg.append("g")
87   .attr("transform", "translate(" + padding + ", 0 " + ")")
88   .call(yAxis);

```

Fig 6: Task 3.2 JS code