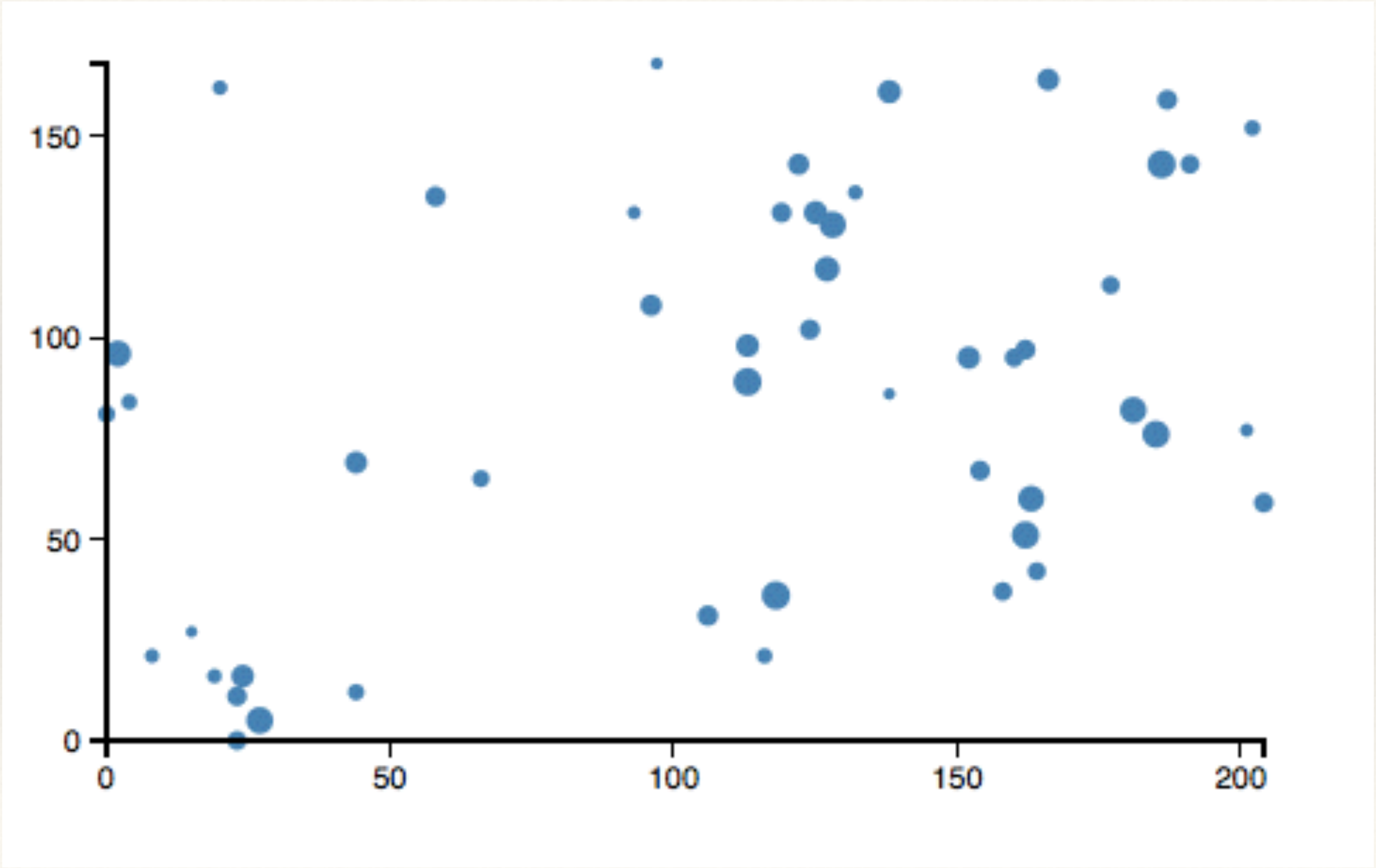
DataScience for Development and Social Change, 2015

D3

How to get started without getting scared…

D3 ("Data Driven Documents")



Where you're going: a D3 File

```
var xScale = d3.scale.linear()
   .domain([0, d3.max(dataset, function(d) { return d[0]; })])
   .range([padding,w - padding *2]);

var yScale = d3.scale.linear()
   .domain([0, d3.max(dataset, function(d) { return d[1]; })])
   .range([h - padding,padding]);

var rScale = d3.scale.linear()
   .domain([0, d3.max(dataset, function(d) { return d[2];})])
   .range([2,5]);

var svg = d3.select("body")
   .append("svg")
   .attr("width", w)
   .attr("height", h);

svg.selectAll("circle")
   .data(dataset)
   .enter()
   .append("circle")
   .attr("fill", "steelblue")
   .attr("cx", function(d) {
      return xScale(d[0]);
   })
   .attr("cy", function(d) {
      return yScale(d[1]);
   })
   .attr("r", function(d) {
      return rScale(d[2]);
   });
```

We start with HTML

```
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>Hello! This is my title</title>
</head>
<body>
    <h1>Hello World!</h1>
</body>
</html>
```

DOM (Document Object Model)

- Open your html file in Chrome
- Click view->developer->javascript console
- type “document” in the new window

```
> document
<- ▼#document
    ▼<html lang="en">
      ▼<head>
          <title>Hello! This is my title</title>
        </head>
      ▼<body>
          <h1>Hello World!</h1>
        </body>
      </html>
>
```

Add some D3

```
<html lang="en">
<head>
    <title>this is where you put your title</title>
    <meta charset="utf-8">
    <script type="text/javascript" src="http://d3js.org/d3.v3.min.js"></script>
</head>
<body>
    <script type="text/javascript">
        d3.select("body").append("p").text("This is my new paragraph!");
    </script>
</body>
</html>
```

Type “document” in the javascript console

```
> document
<- ▼#document
    ▼<html lang="en">
      ▼<head>
          <title>Hello! This is my title</title>
        <meta charset="utf-8">
        <script type="text/javascript" src="http://d3js.org/d3.v3.min.js"></script>
      </head>
    ▼<body>
        <h1>Hello World!</h1>
        <script type="text/javascript">
                d3.select('body').append('p').text('This is my new paragraph!');
            </script>
        <p>This is my new paragraph!</p>
      </body>
    </html>
```

Why do we care about the DOM?

- The DOM is created when you open a webpage
- It contains the html code for that page
- And handles things like mouse clicks on the page
- D3 works by changing the DOM.
  - Just like the last example, where “ New paragraph! ” appeared in the DOM.

Adding D3 to your HTML

- If you have stable Internet:
  - use http://d3js.org/d3.v3.min.js
  - <script type="text/javascript" src="d3/d3.js"></script>

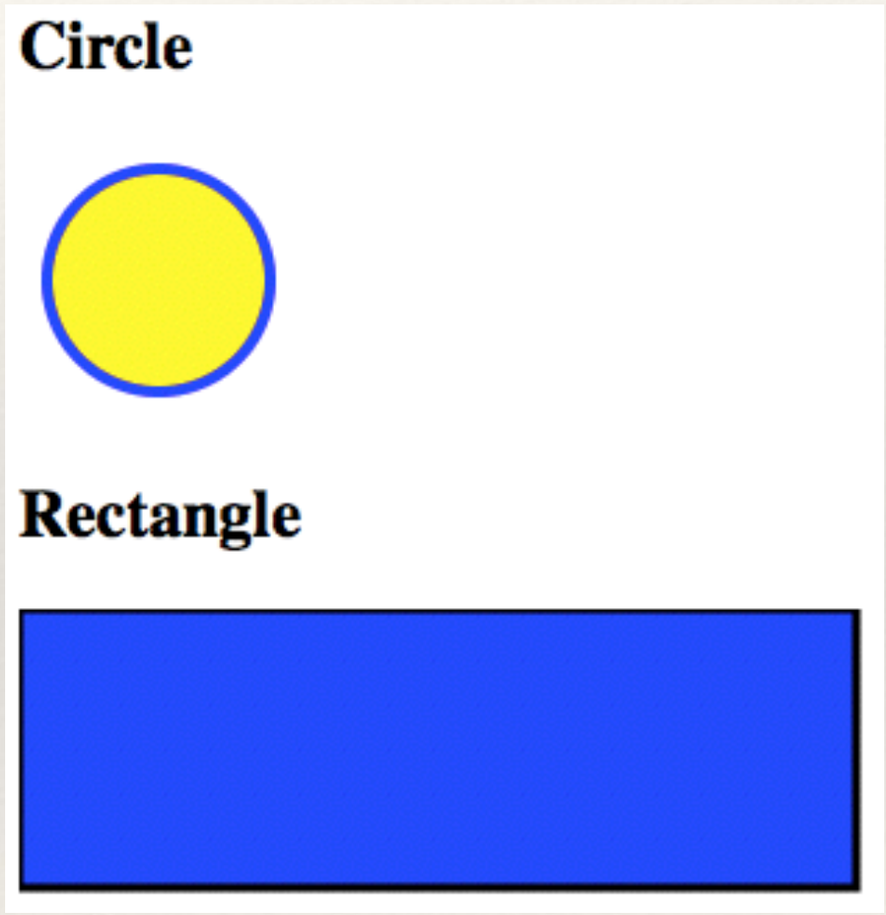
- If you have no Internet:
  - Download the D3 library from http://d3js.org/
  - use d3/d3.js
  - <script type="text/javascript" src="http://d3js.org/d3.v3.min.js"></script>

Drawing Shapes in HTML

SVG graphics language: draws things like circles and rectangles…

```
<svg width="100" height="100">

  <circle cx="50" cy="50" r="40" stroke="blue" stroke-width="4"
fill="yellow" />

</svg>
```

```
<svg width="400" height="110">

  <rect width="300" height="100" style=“fill:rgb(0,0,255); stroke-
width:3; stroke:rgb(0,0,0)">

</svg>
```

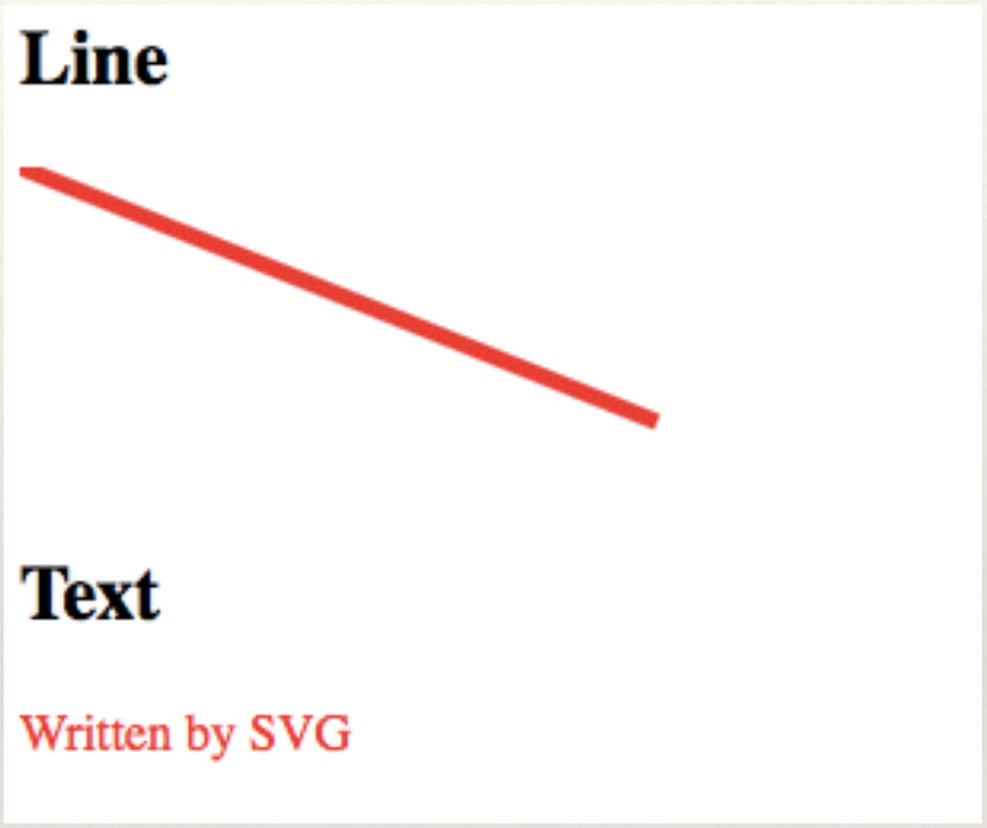


http://www.w3schools.com/svg/

Drawing Lines and Text with SVG

And lines and text

```
<svg height="100" width="500">
  <line x1="0" y1="0" x2="200" y2="80"
style="stroke:rgb(255,0,0);stroke-width:5" />
</svg>

<svg height="30" width="200">
  <text x="0" y="15" fill="red">Written by SVG</text>
</svg>
```



CSS (Cascading StyleSheets) makes HTML pretty

- HTML classes and ids:

```
<p class=“orange">This is an orange paragraph</p>
< p id=“mylabel”>This paragraph is labelled</p>
```

- CSS:

```
p {
  color: blue; /* Turn every paragraph blue */
}
#mylabel {
  font-style: bold; /* Make the mylabel paragraph text bold */
}
```

D3 is called from Javascript

A basic D3 visualization

- We need:
  - Styles (line colors, text fonts etc)
  - A drawing area
  - A dataset (x, y values)
  - x and y scales (e.g. a coordinate system)
  - Shapes
  - x and y axes

Style

```
<style>
  .axis path,
  .axis line {
    fill: none;
    stroke: black;
    shape-rendering: crispEdges;
  }
  .axis text {
    font-family: sans-serif;
    font-size: 11px;
  }
</style>
```

Drawing area

```
var vizwidth=500;
var vizheight=300;
var padding=30;

var svg = d3.select("body")
    .append("svg")
    .attr("width", vizwidth)
    .attr("height", vizheight);
```

Data

```
var dataset = [];
 var numDataPoints = 50;
 var xRange = Math.random() * 1000;
var yRange = Math.random() * 1000;
var rRange = Math.random() * 1000;
for (var i =0;i<numDataPoints;i++) {
  var newNumber1 = Math.floor(Math.random() * xRange);
  var newNumber2 = Math.floor(Math.random() * yRange);
  var newNumber3 = Math.floor(Math.random() * rRange);
  dataset.push([newNumber1, newNumber2, newNumber3]);
}
```

x, y and radius Scales

```
var xScale = d3.scale.linear()
   .domain([0, d3.max(dataset, function(d) { return d[0]; })])
   .range([padding,vizwidth - padding *2]);

var yScale = d3.scale.linear()
   .domain([0, d3.max(dataset, function(d) { return d[1]; })])
   .range([vizheight - padding,padding]);

var rScale = d3.scale.linear()
   .domain([0, d3.max(dataset, function(d) { return d[2];})])
   .range([2,5]);
```

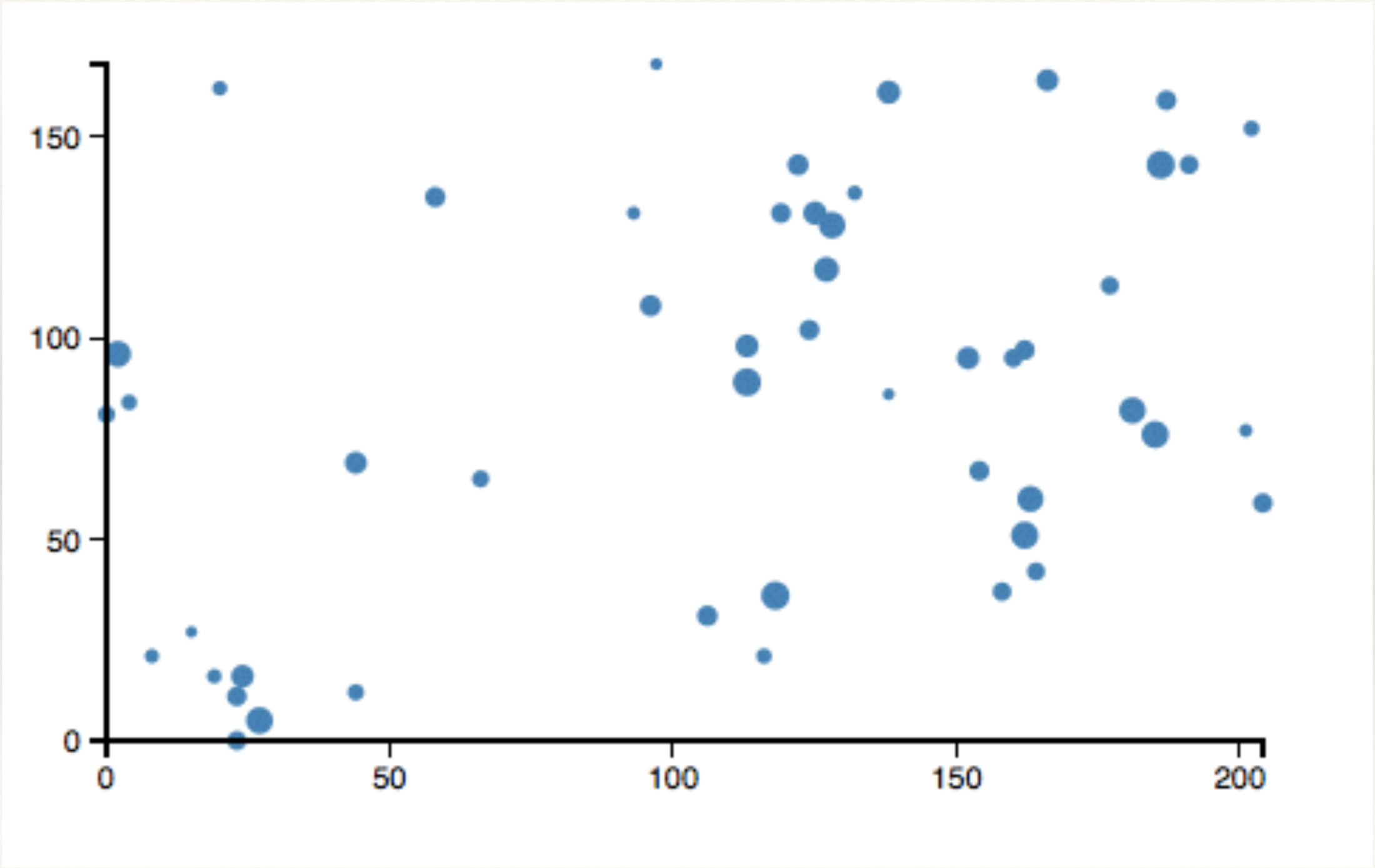
Shapes

```
svg.selectAll("circle")
   .data(dataset)
   .enter()
   .append("circle")
   .attr("fill", "steelblue")
   .attr("cx", function(d) {
      return xScale(d[0]);
   })
   .attr("cy", function(d) {
      return yScale(d[1]);
   })
   .attr("r", function(d) {
      return rScale(d[2]);
   });
```

x and y Axes

```
var xAxis = d3.svg.axis()
  .scale(xScale)
  .orient("bottom")
  .ticks(5);

svg.append("g")
  .attr("class", "axis")
  .attr("transform", "translate(0,"+(vizheight-padding)+")")
  .call(xAxis);

var yAxis = d3.svg.axis()
  .scale(yScale)
  .orient("left")
  .ticks(5);

svg.append("g")
  .attr("class", "axis")
  .attr("transform", "translate("+padding+",0)")
  .call(yAxis);
```

The end product:



A D3 Choropleth

- We need:
  - Styles (line colors, text fonts etc)
  - A set of map shapes (e.g. admin boundaries)
  - A dataset (shapename, value)
  - A color palette (value, color)
  - A legend
  - Tooltips

Reading data from a CSV file

Connecting D3 and Python

- Welcome to Flask…

Making D3 Easier

- D3 helpers:
  - NVD3
  - xCharts
  - dimple
  - Vega
- D3 libraries:
  - rCharts
  - d3py

http://schoolofdata.org/2013/08/12/climbing-the-d3-js-visualisation-stack/

Continuing your D3 journey