

CMPT 384 – Information Visualization

D3.js

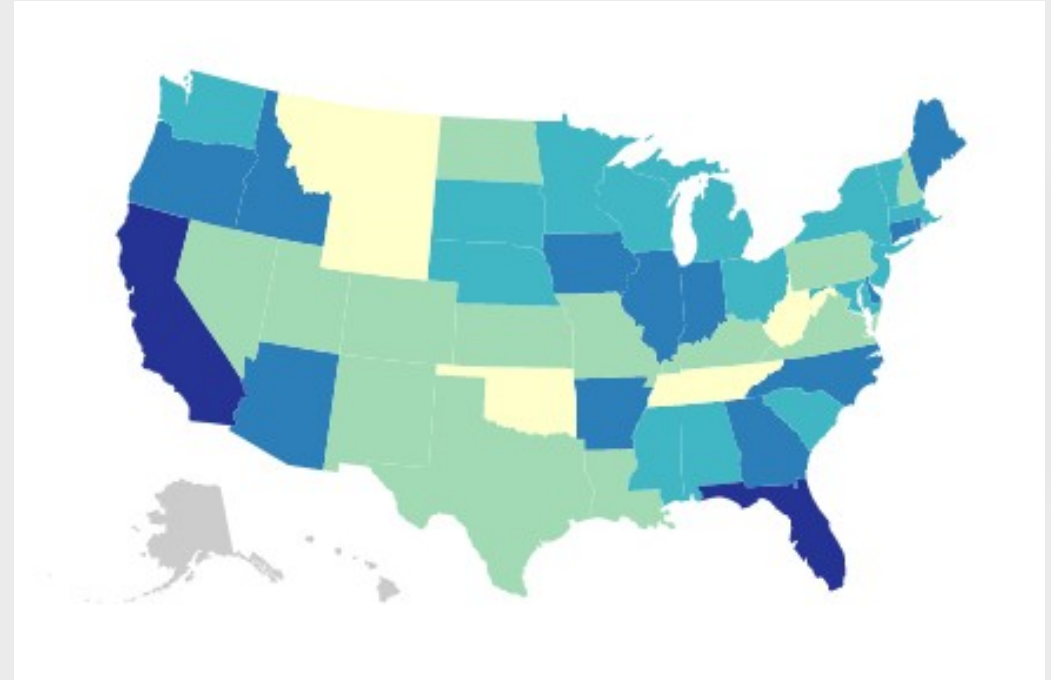
Lab 5

Course Instructor: Debajyoti Mondal

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

Agenda

- D3 Path
- D3 Generator (line)
- D3 Scale (Quantize)
- D3 Geographic Maps
 - D3 Geo Projection
 - Path Generator



D3 Path

- Path tag and d attribute

Command	Parameters	Repeatable	Explanation
M (m)	x, y	Yes	moveto Move the pen to a new location. No line is drawn. All path data must begin with a 'moveto' command.
L (l)	x, y	Yes	lineto Draw a line from the current point to the point (x,y).
C (c)	x1 y1 x2 y2 x y	Yes	curveto Draw a cubic Bézier curve from the current point to the point (x,y) using (x1,y1) as the control point at the beginning of the curve and (x2,y2) as the control point at the end of the curve.
Q (q)	x1 y1 x y	Yes	quadratic Bézier curveto Draw a quadratic Bézier curve from the current point to (x,y) using (x1,y1) as the control point.
Z (z)	none	No	closepath Closes the path. A line is drawn from the last point to the first point drawn.

```
<path d="M 10 25 L 10 75 L 60 75 L 10 25"></path>
```

D3 Generator

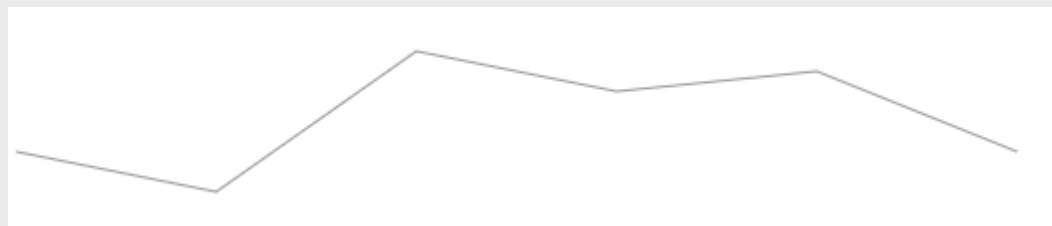
<u>line</u>	Generates path data for a multi-segment line (typically for line charts)
<u>area</u>	Generates path data for an area (typically for stacked line charts and streamgraphs)
<u>stack</u>	Generates stack data from multi-series data
<u>arc</u>	Generates path data for an arc (typically for pie charts)
<u>pie</u>	Generates pie angle data from array of data
<u>symbol</u>	Generates path data for symbols such as plus, star, diamond

```
var lineGenerator = d3.line();
```

```
var points = [  
  [0, 80],  
  [100, 100],  
  [200, 30],  
  [300, 50],  
  [400, 40],  
  [500, 80]  
];
```

```
var pathData = lineGenerator(points);  
// pathData is "M0,80L100,100L200,30L300,50L400,40L500,80"
```

```
d3.select('path')  
  .attr('d', pathData);
```



D3 Path – example 1

Line generator

```
<svg width='500' height='300'>
  <path d="M 10 25 L 10 75 L 60 75 L 10 25"></path>
</svg>
```

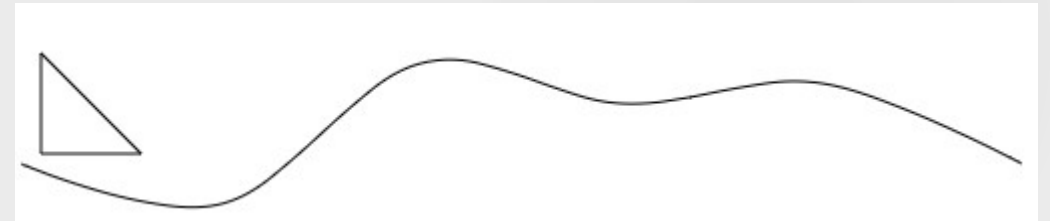
```
// Specify the type of curve
// and create a line path generator

var lineGenerator = d3.line()
  .curve(d3.curveCardinal);

// Array of points x,y
var points = [
  [0, 80],
  [100, 100],
  [200, 30],
  [300, 50],
  [400, 40],
  [500, 80]
];

// Convert data points into path
var pathData = lineGenerator(points);

// Add a new path element and set the path data as attribute
d3.select('svg').append("path")
  .attr('d', pathData);
```



- d3.curveCardinal
- d3.curveNatural
- d3.curveBasis
- d3.curveStep

D3 Scales – Quantize Scale

```
var bucket = d3.scaleQuantize()  
    .domain([0, 1])  
    .range(['A', 'B', 'C'])
```

A

0-0.33

B

0.33-0.66

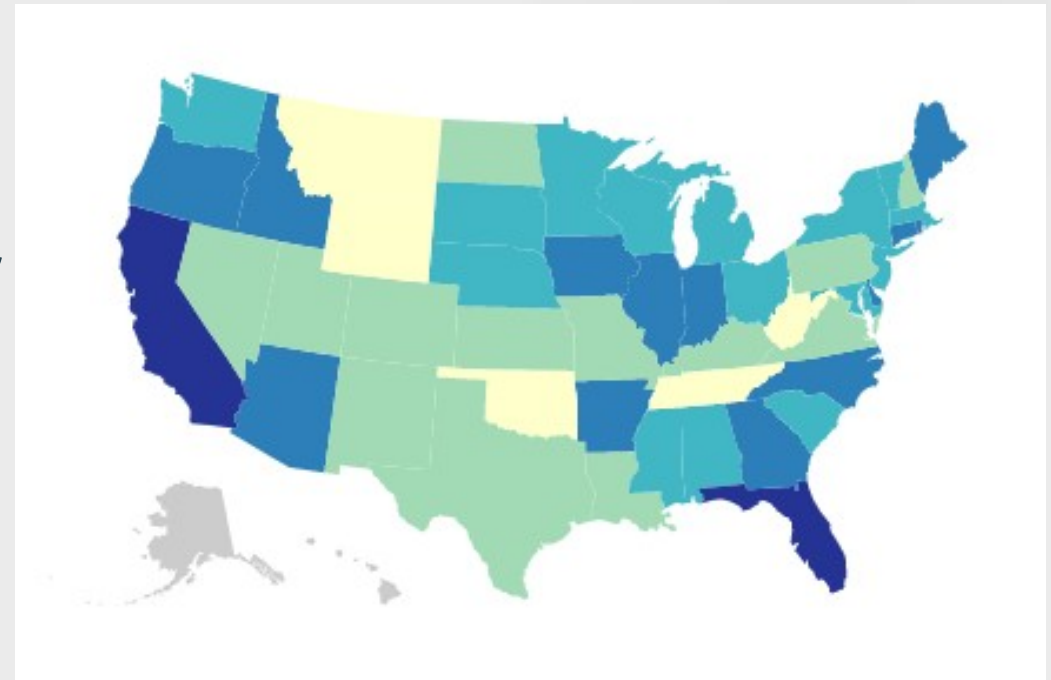
C

0.66 – 1.00

```
>> bucket(0.25)  
>> 'A'  
>> bucket(0.75)  
>> 'C'
```

D3 Geographic Maps

- Declare a map projection (<https://github.com/d3/d3-geo-projection>)
- Define geo-path generator
- Read geoJSON file
- Map data with geoJSON
- Set geo-path in 'd' attribute of 'path'



US Agri Productivity

<https://www.data.gov/>

Finding the Right GEOJSON

LIST OF MOST NORTH AMERICAN CITIES AND PROVINCES -

[https://github.com/codeforamerica/click that hood/tree/master/public/data](https://github.com/codeforamerica/click_that_hood/tree/master/public/data)

CONVERT PUBLICLY AVAILABLE SHAPE FILES INTO GEOJSON

-

<http://geojson.io/#map=2/20.0/0.0>

Finding the Right Data

OPEN-DATA Websites -

- <http://www.opendatask.ca/data/> - Curated List of all data on saskatoon available for free
- <http://opendata-saskatoon.cloudapp.net/> - Catalogue of Saskatoon open data
- <https://data.calgary.ca/> - Calgary open data

Start the server

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

```
C:\> python -m SimpleHTTPServer 8888
```

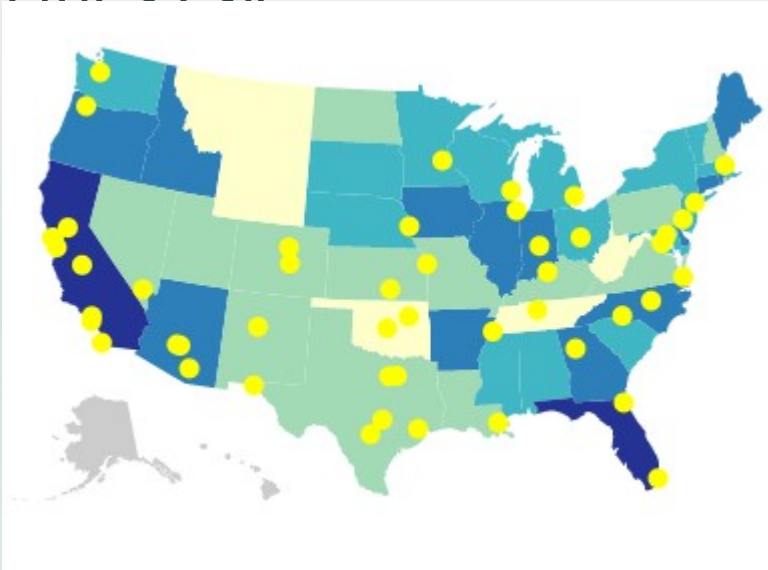
```
C:\Users\jyoti\AppData\Local\Programs\Python\Python37-32\python.exe: No  
module named SimpleHTTPServer
```

```
C:\> python -m http.server 8888
```

```
Serving HTTP on 0.0.0.0 port 8888 (http://0.0.0.0:8888/) ...
```

Try this -

Try to create a visualization like the below with the help of `us-cities.csv`



Hint : You can get the pixel position for the cities by passing latitude and longitude to the projection

Answer is here: [example_4_cities.html](#)