

学习 d3 过程中遇到的比例尺总结

作者: [zjhch123](#)

原文链接: <https://ld246.com/article/1591517347481>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

学习过程中用到的比例尺总结

在学习d3、写Demo时用到过一些比例尺，但是总是用了就忘，记不住它们的用处，分辨不出什么场合适合用哪种比例尺。于是整理一篇文章记录它们的用处与用法。

scaleLinear()

线性比例尺，最基础常见的比例尺。会将连续的 **domain**与连续的 **range**以线性的关系对应输出

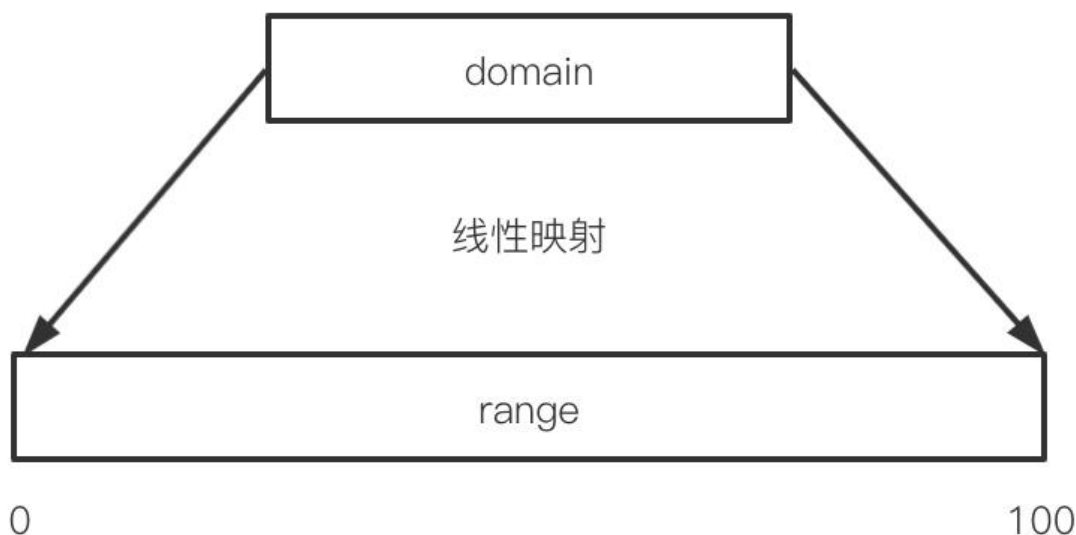
```
const scaleLinear = d3.scaleLinear().domain([0, 10]).range([0, 100])
```

```
scaleLinear(0) // 0  
scaleLinear(2) // 20  
scaleLinear(5) // 50
```

如果用普通 **javascript**代码来描述其行为的话，大致应该是

```
const getScaleLinear = (input, output) => {  
  const min = input[0]  
  const max = input.slice(-1)[0]  
  const minOutput = output[0]  
  const maxOutput = output.slice(-1)[0]  
  
  return (v) => (v - min) / (max - min) * (maxOutput - minOutput)  
}
```

其映射关系图如下



scaleBand()

不连续的 **domain**与连续的 **range**进行映射。会将 **range**依据 **domain**的个数进行分段。不知道应该何最标准的翻译，直接上Demo吧。

```
const scaleBand = d3.scaleBand().domain([0, 1, 2, 3]).range([0, 100]);
```

```
scaleBand(0); // 0
scaleBand(1); // 25
scaleBand(2); // 50
scaleBand(3); // 75
```

```
scaleBand(4); // undefined <----
scaleBand(8); // undefined
```

可以观察到，输入的值必须在 **domain** 中被定义才能得到输出

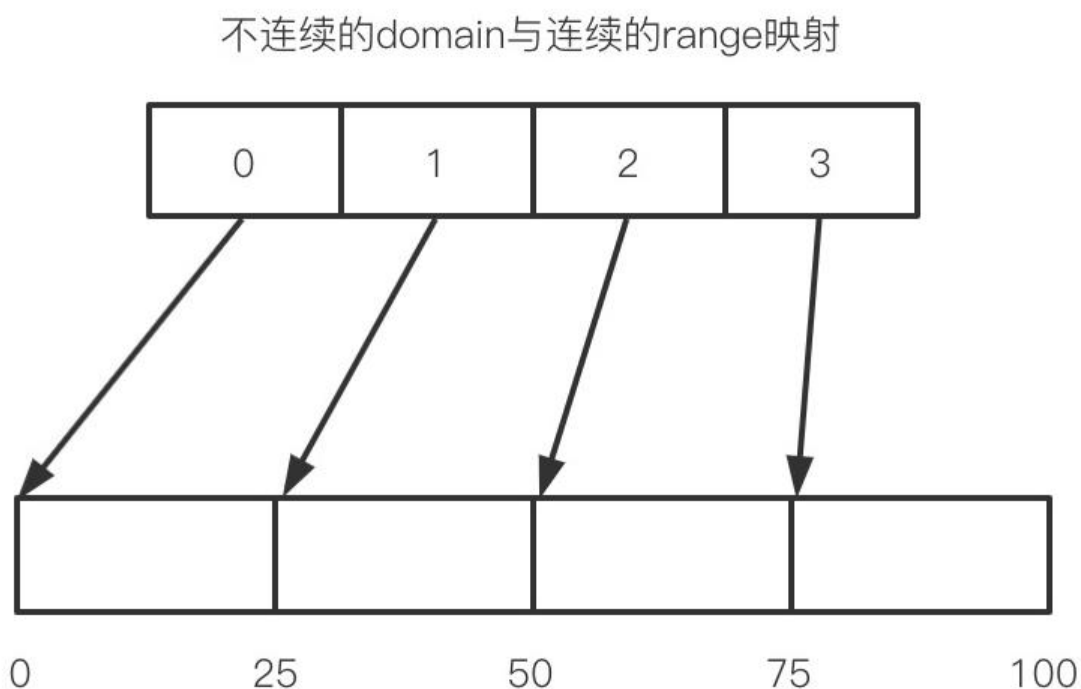
1. 0, 1, 2, 3 均被定义过，所以输出不是 **undefined**
2. 4, 8 未被定义，所以输出是 **undefined**。

翻译成 **javascript** 代码应该是

```
const getScaleBand = (input, output) => {
  const count = input.length
  const minOutput = output[0]
  const maxOutput = output.slice(-1)[0]
  const step = (maxOutput - minOutput) / count

  return (v) => {
    const index = input.indexOf(v)
    return index === -1 ? undefined : index * step + minOutput
  }
}
```

其映射关系图如下



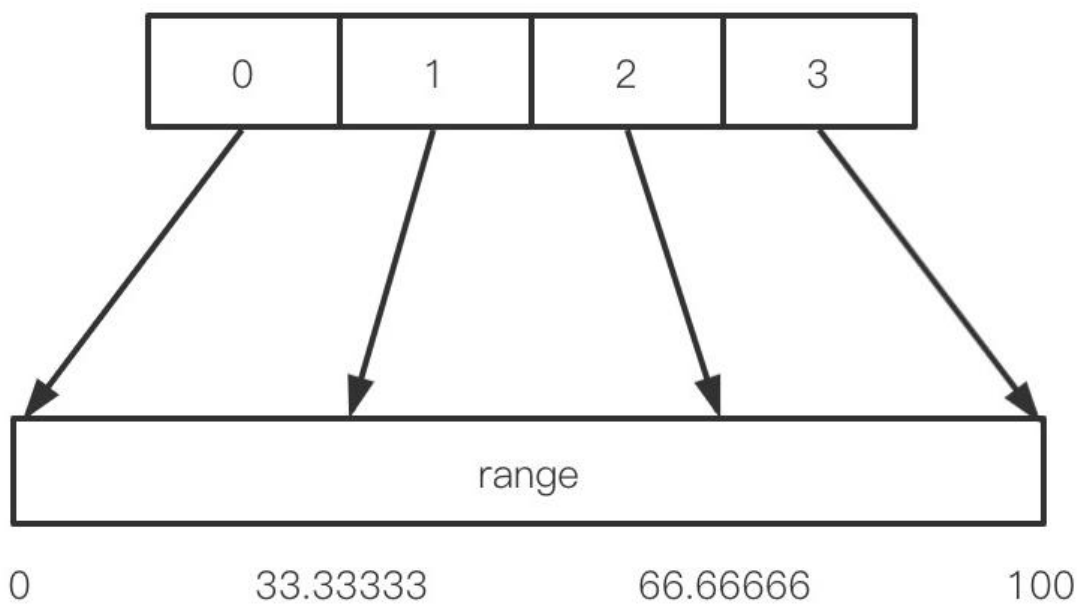
scalePoint()

与 `scaleBand()` 一样，也是不连续的 `domain` 与连续的 `range` 之间的映射。与 `scaleBand()` 不同的是，`calePoint()` 中 `domain` 的最大/最小值是与 `range` 中的最大/最小值对应的。`domain` 中剩余的值会与 `range` 中相应的值进行对应。

```
const scalePoint = d3.scalePoint().domain([0, 1, 2, 3]).range([0, 100]);
```

```
scalePoint(0); // 0
scalePoint(1); // 33.33333
scalePoint(2); // 66.66666
scalePoint(3); // 100
```

其映射关系图如下



scaleOrdinal

`scaleOrdinal` 比 `scaleBand` 更进一步，是不连续的 `domain` 与不连续的 `range` 之间的对应关系。

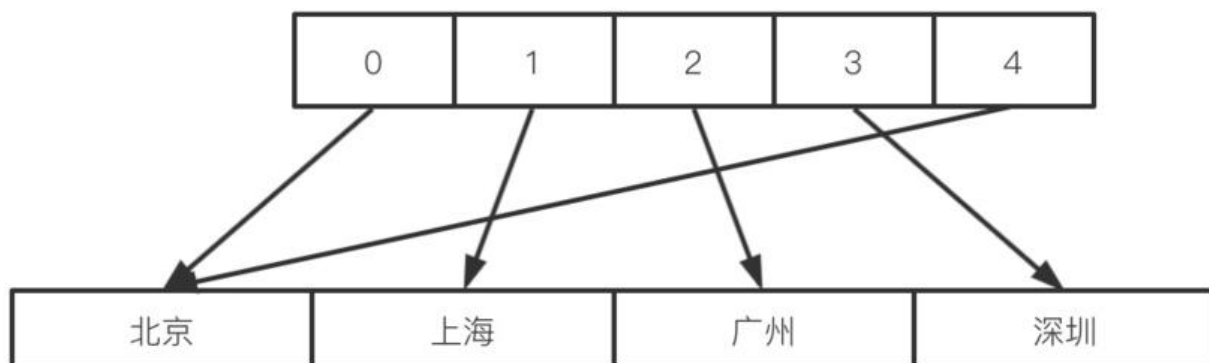
```
const scaleOrdinal = d3.scaleOrdinal().domain([0, 1, 2, 3]).range(['北京', '上海', '广州', '深圳'])
```

```
scaleOrdinal(0); // 北京
scaleOrdinal(1); // 上海
scaleOrdinal(2); // 广州
scaleOrdinal(3); // 深圳
```

```
scaleOrdinal(4); // 北京 <---
```

可以观察到，当输入 `domain` 中不存在的值时，`scaleOrdinal` 会自动寻找对应的值

其映射关系图如下



未完待续

目前我只接触到了上述4种比例尺，如果以后遇到了新的比例尺也会继续在这里更新。

我的d3学习仓库，包含了一些常见图表的Demo