该文只使用d3.jsV4版本进行绘制，不关注V3版本，我们要与时俱进。

在绘制图表的过程中，直坐标系是绕不开的一个问题，直方图，折线图，散点图等等都需要使用到直坐标系。而其中最关键的是x轴的绘制，因为y轴基本上都是数值显示。如何用代码实现，x轴的表现形式是什么，这是本文主要讨论的问题。

1 连续性x坐标轴

1.1 从零开始的连续性x坐标轴

什么是从零开始，就是从绘制的坐标轴的最左端开始显示第一个刻度。**折线图**，**散点图**经常采用这种样式的x坐标轴。

**let** height = 400

**let** width = 600

**let** x = d3.scaleLinear().range([0, width])

**let** xScale = x.domain([0, 10])

// x轴

**let** xAxis = svg.append('g')

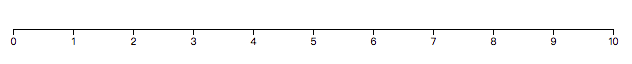
.attr('class', 'xAxis')

.attr('transform', `translate(0, ${height})`)

.call(d3.axisBottom(xScale))

使用d3.axisTop和d3.axisBottom()来控制刻度显示在坐标轴的上方或者下方。

表现形式：



1.2 不从零开始的连续性x坐标轴

情况比较少，基本不用，所以不作阐述。

1.3 时间型x坐标轴

时间轴也是线性的，所以将它归为此类。

**let** height = 400

**let** width = 600

**let** x = d3.scaleTime().range([0, width])

**let** xScale = x.domain([**new** Date(2017, 1), **new** Date(2017, 6)])

// x轴

**let** xAxis = svg.append('g')

.attr('class', 'xAxis')

.attr('transform', `translate(0, ${height})`)

.call(d3.axisBottom(xScale))

表现形式：

https://segmentfault.com/img/remote/1460000010910312

2 非连续性x坐标轴

2.1 不从零开始的非连续性x坐标轴

先从不从零开始说起，因为这个用法比较正常。关键是使用d3.scaleBand()。

**let** height = 400

**let** width = 600

**let** x = d3.scaleBand().range([0, width])

**let** xScale = x.domain(['北京', '上海', '广州', '深圳'])

// x轴

**let** xAxis = svg.append('g')

.attr('class', 'xAxis')

.attr('transform', `translate(0, ${height})`)

.call(d3.axisBottom(xScale))

表现形式：

https://segmentfault.com/img/remote/1460000010910313

基本上，柱状图都会采用这种x坐标轴。

2.2 从零开始的非连续性x坐标轴

使用d3.scaleOrdinal()

**let** height = 400

**let** width = 600

**let** x = d3.scaleOrdinal().range([150, 300, 450, 600])

**let** xScale = x.domain(['北京', '上海', '广州', '深圳'])

// x轴

**let** xAxis = svg.append('g')

.attr('class', 'xAxisis')

.attr('transform', `translate(0, ${height})`)

.call(d3.axisBottom(xScale))

表现形式：

https://segmentfault.com/img/remote/1460000010910314

在正常情况中，x轴的数据经常是非线性的。而绘制折线图，散点图等等又需要采用这种表现形式，所以这种方法是比较常用的。