

山东大学计算机科学与技术学院

大数据分析实践课程实验报告

学号：202300130166

姓名：朱亚宁

班级：23 数据

实验题目：电子表格

实验学时：2

实验日期：2025. 10. 17

实验目标：添加基于开源电子表格的新可视化函数

实验步骤：

一、代码复原实现

结果：

生成可视化图表

柱状图 (分组)

	A	B	C	D	E	F	G	H
1		计算机	法学					
2	2017	23	15					
3	2018	36	26					
4	2019	23	33					
5	2020	22	10					
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								

人数

40

35

30

25

20

15

10

5

0

年份

2017

2018

2019

2020

计算机

法学

二、添加新的可视化函数（折线图）

①函数首先从传入的结构化数据 visData 中提取原始数值 rawData、类别标签 categories、系列标签 series 和最大值 maxValue，接着定义图表的边距、宽度和高度，其中宽度和高度是扣除边距后的实际绘制区域大小。

②清空容器中已有的 SVG 元素，避免重复绘制，再创建一个 SVG 容器并设置其宽高，通过 translate 属性将绘图区域偏移边距，为坐标轴和标签预留空间。

③定义 X 轴为序号轴，以年份为域名，范围覆盖绘图区域宽度，将其绘制在图表底部并添加“年份”标签；Y 轴为线性轴，域名从 0 到最大值的 1.1 倍，范围从底部到顶部，绘制在左侧并添加“人数”标签。

④处理数据，将每个专业对应的年份和数值转换为折线图所需的格式，每个专业生成一条线的数据。然后定义线生成器，设置 x 坐标为年份分组的中间位置，y 坐标映射数值，使用平滑曲线过渡。通过线生成器绘制每条折线，用不同颜色区分专业，线条无填充、宽度为 2。再为每个数据点绘制圆形标记，位置与折线的点对应，颜色与所属专业一致。

⑤在每个数据点上方添加数值标签，居中显示具体人数。最后创建图例，每个专业对应一个色块和名称，放置在图表右侧，通过颜色对应帮助区分不同专业的折线。

核心代码：

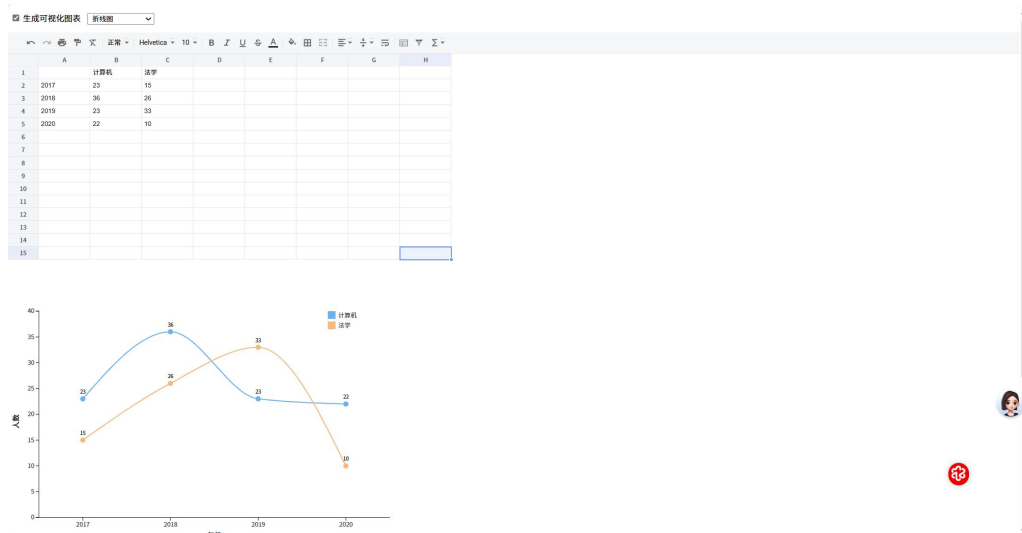
```
// 处理数据：转为折线图需要的格式（每个系列一条线）
const lineData = series.map((s, sIdx) => {
  return categories.map((c, cIdx) => ({
    category: c,
    series: s,
    value: rawData[cIdx][sIdx]
  }));
});

// 定义线生成器
const lineGenerator = d3.line()
  .x(d => xAxis(d.category) + xAxis.bandwidth() / 2) // 线点在年份中间
  .y(d => yAxis(d.value))
  .curve(d3.curveMonotoneX); // 平滑曲线

// 绘制每条折线
svg.append("g")
  .selectAll("path")
  .data(lineData)
  .join("path")
  .attr("d", d => lineGenerator(d))
  .attr("fill", "none")
  .attr("stroke", (d, idx) => getColor(idx))
  .attr("stroke-width", 2);

// 绘制数据点
svg.append("g")
  .selectAll("circle")
  .data(lineData.flat())
  .join("circle")
  .attr("cx", d => xAxis(d.category) + xAxis.bandwidth() / 2)
  .attr("cy", d => yAxis(d.value))
  .attr("r", 5)
  .attr("fill", (d, idx) => getColor(series.indexOf(d.series)));
```

结果：



结论分析与体会：

本实验通过整合开源电子表格与可视化库，实现了数据录入、处理与可视化的闭环流程。其核心价值在于验证了 “低代码” 模式下数据可视化的可行性 。