

大数据分析实践实验报告

实验三：电子表格数据可视化实践

一、实验内容

本次实验主要学习如何使用电子表格进行数据可视化。我们使用了一个开源的电子表格组件，在网页中创建了一个可以编辑的数据表格，然后通过编程的方式将表格中的数据用图表的形式展示出来。

实验包含两个主要部分：

1. 运行给出的示例代码
2. 将表格中的数据用两种不同的折线图进行可视化展示

二、实验步骤

第一步：搭建基础环境

首先在HTML页面中引入必要的库文件，包括电子表格组件和图表绘制库。

第二步：创建电子表格

在网页上创建一个可以编辑的电子表格区域，设置表格的行列数和样式。

第三步：准备示例数据

在表格中预先填入一些示例数据，包括不同年份计算机学院和法学院的招生人数。

第四步：实现数据可视化功能

编写JavaScript代码，实现以下功能：

- 读取表格中的数据
- 处理数据格式，使其适合绘制图表
- 创建两种折线图来展示数据

第五步：添加交互功能

实现复选框功能，让用户可以控制图表的显示和隐藏。

三、关键代码说明

1. 引入必要的库文件

```
<link rel="stylesheet" href="https://unpkg.com/x-data-spreadsheet@1.1.5/dist/xspreadsheet.css" />
<script src="https://unpkg.com/x-data-spreadsheet@1.1.5/dist/xspreadsheet.js">
</script>
<script src="https://d3js.org/d3.v6.js"></script>
```

2. 创建电子表格

```

var xs = x_spreadsheet("#xspreadsheet", {
  mode: 'edit',
  showToolBar: true,
  showGrid: true,
  row: {
    len: 15,
    height: 25,
  },
  col: {
    len: 8,
    width: 100,
  }
})

```

3. 设置初始数据

```

xs.cellText(0, 1, "计算机").cellText(0, 2, "法学").reRender();
xs.cellText(1, 0, "2017").cellText(1, 1, "23").cellText(1, 2, "15").reRender();
xs.cellText(2, 0, "2018").cellText(2, 1, "36").cellText(2, 2, "26").reRender();

```

4. 读取表格数据并绘制图表的核心函数

```

function update() {
  // 读取表格数据
  var data = [];
  var ytitle = []; // 年份
  var xtitle = []; // 学院名称

  for (var i = 1; i < 20; i++) {
    if (xs.cell(i, 0) === null || xs.cell(i, 0).text === "") {
      break;
    }
    data.push([]);
    ytitle.push(xs.cell(i, 0).text);
  }

  // 处理数据并创建折线图
  var lineData1 = []; // 各学院数据
  var lineData2 = []; // 总人数数据

  for (var i = 0; i < ytitle.length; i++) {
    var year = ytitle[i];
    var computerValue = data[i][0];
    var lawValue = data[i][1];
    var totalValue = computerValue + lawValue;

    lineData1.push({year: year, computer: computerValue, law: lawValue});
    lineData2.push({year: year, total: totalValue});
  }

  // 绘制折线图的代码...
}

```

5. 创建折线图的代码片段

```
// 创建折线生成器
const lineComputer = d3.line()
  .x(d => xScale1(d.year) + xScale1.bandwidth() / 2)
  .y(d => yScale1(d.computer));

// 绘制计算机学院折线
chart1.append("path")
  .datum(lineData1)
  .attr("class", "line")
  .attr("d", lineComputer)
  .style("stroke", getColor(0));
```

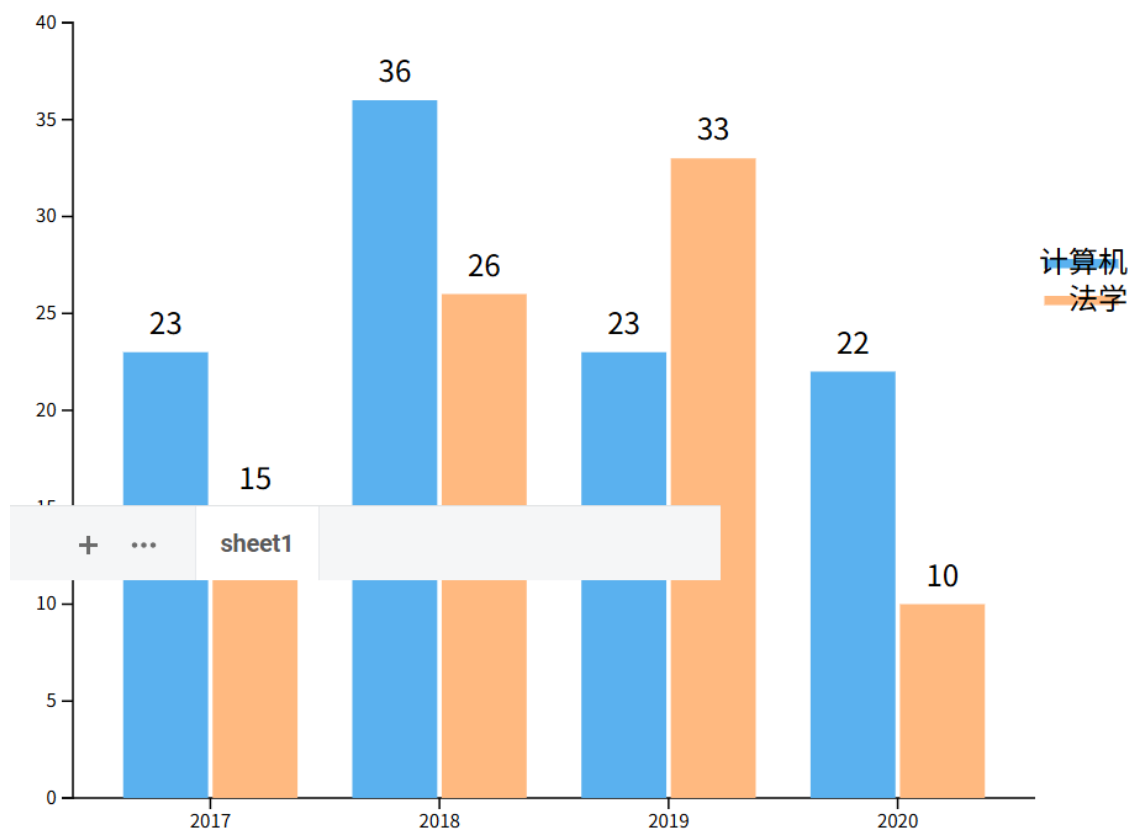
四、结果展示

实验成功实现了以下功能：

1. 交互式电子表格：

- 创建了一个可以自由编辑的电子表格
- 用户可以修改任何单元格的数据
- 表格具有基本的格式化功能

运行示例结果如下：

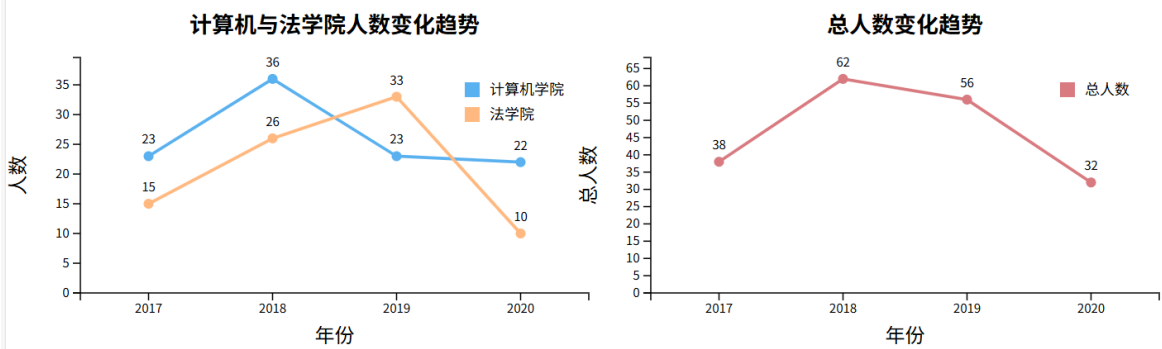


1. 双折线图展示：

- **左图：**展示了计算机学院和法学院从2017年到2020年的招生人数变化趋势
 - 红色折线表示计算机学院人数变化
 - 蓝色折线表示法学院人数变化
- **右图：**展示了两个学院总人数的变化趋势
 - 绿色折线表示总人数变化

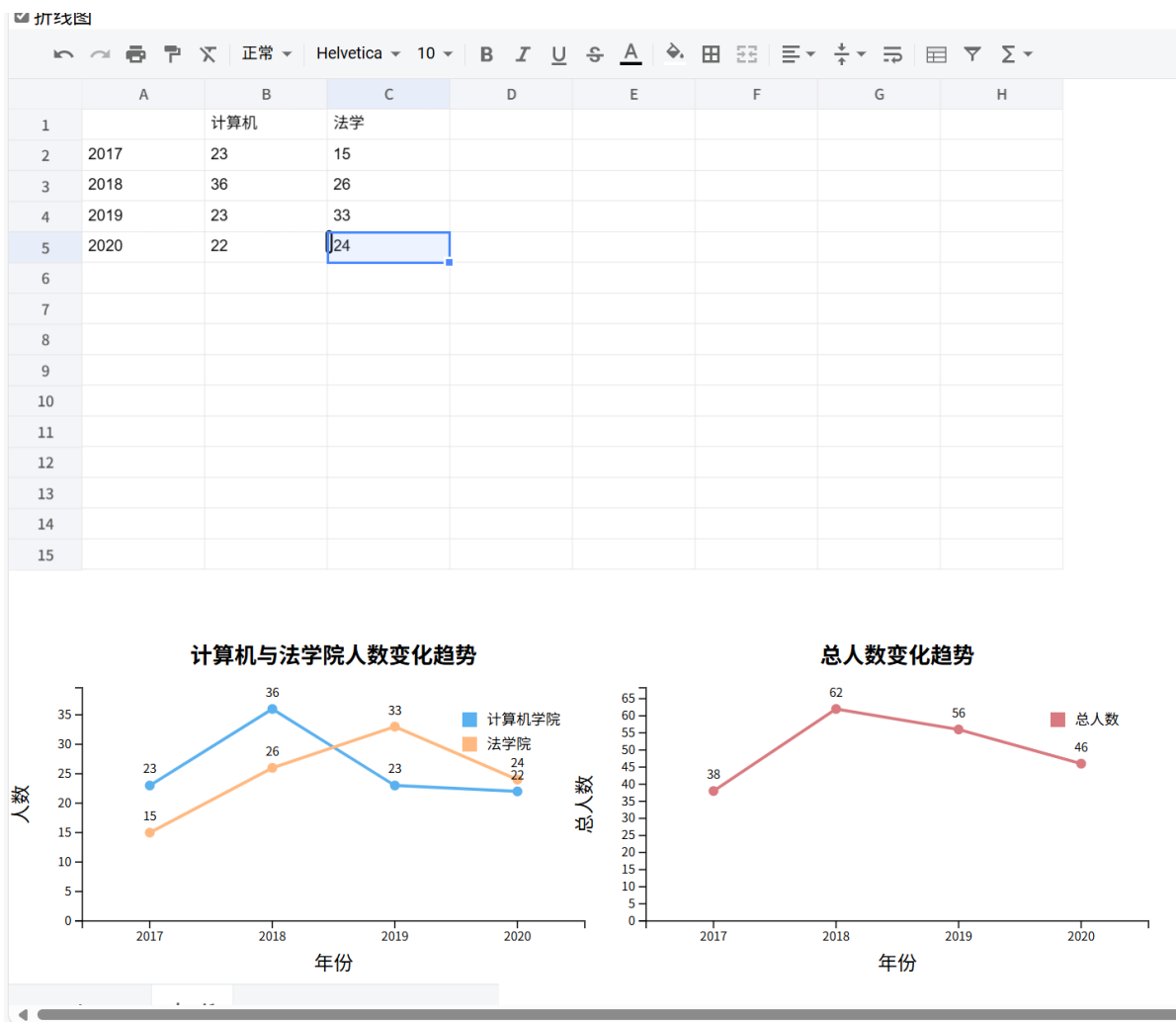
折线图

	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								



1. 实时更新功能:

- 当用户在表格中修改数据时，图表会自动更新
- 通过复选框可以控制图表的显示和隐藏



1. 数据标签和图例：

- 每个数据点都显示了具体数值
- 添加了图例说明，方便理解图表内容

五、实验结论

通过本次实验，我们成功实现了电子表格数据的可视化展示，主要收获如下：

- 技术掌握：**学会了如何使用JavaScript库创建交互式电子表格和数据可视化图表。
- 数据处理：**掌握了从表格中读取数据、处理数据格式、并将数据转换为图表所需格式的方法。
- 可视化效果：**实现了清晰直观的折线图，能够很好地展示数据的变化趋势。
- 交互体验：**实现了数据修改后图表的实时更新，提供了良好的用户体验。
- 实际应用：**这种技术可以应用于各种需要数据分析和可视化的场景，如成绩统计、销售数据分析、科研数据处理等。

实验证明，将电子表格与数据可视化相结合，能够更直观地理解数据背后的规律和趋势，为数据分析提供了有力的工具。