

《数据可视化实验》

报告

**学 号： 22009200894**

**姓 名： 王越洋**

**《数据可视化实验》报告评分表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **课程名称** | **数据可视化** | | | | |
| **学 号** | **22009200894** | **姓名** | **王越洋** | **成绩** |  |
| **题 目** | 实验三：空间数据可视化 | | | | |
| **要 求** | 在3D空间中绘制以下图形（建议使用Echarts完成）：  1、螺旋曲线（半径为5，每旋转一周z坐标增加π）  2、二元正态分布曲面  3、球面（半径为5） | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **评价项目** | **评价标准** | **满分** | **得分** |
| 方案合理性 | 能够根据题目的要求，对问题进行分析，利用了一定的基础理论和科学方法来解决实际工程问题，给出了完整的解决方案，报告中理论准备充分，有充分的分析论证过程。 | **15** |  |
| 结果正确性 | 报告数据与分析详实、正确、可信；  设计、实验及测试数据的充分性和可靠性；  理论推导或建模的严密性和完整性。 | **15** |  |
| 报告水平 | 全面了解本领域的动态，并能很好地评述研究背景（文献、市场、需求等）；  报告内容涉及较为深入的基础理论知识、专业技术知识、相关学科专业知识；  具有适当的技术难度，工作量较大。 | **25** |  |
| 写作能力 | 图表清晰，语言规范，符合实验报告要求；  设计、图表、软件的规范化和标准化；  文字表达的专业性、通顺性和概念的准确性；  论文构架的系统性、逻辑性和严谨性；  引用文献的真实性、贴切性和规范性。 | **15** |  |
| 演示视频 | 视频图像和声音清晰，能结合文档更好地展现作业的完成情况。 | **15** |  |
| 使用现代工具 | 能够针对复杂工程问题，开发、选择与使用恰当的技术、资源、现代工程工具和信息技术工具，包括对复杂工程问题的预测与模拟，并能够理解其局限性。 | **15** |  |
| **总 分** | | |  |

**评阅人（签字）： 2023年 月 日**

|  |
| --- |
| **一、实验目的及要求**  掌握空间数据可视化的原理和方法。  实验内容：  在3D空间中绘制以下图形（建议使用Echarts完成）：  1、螺旋曲线（半径为5，每旋转一周z坐标增加π）  2、二元正态分布曲面  3、球面（半径为5）  **二、实验环境配置**  1.硬件要求：  操作系统：Windows, macOS, 或 Linux  内存：4GB 以上  CPU：现代多核处理器（如 Intel i5 以上）  2.软件要求：  浏览器：Chrome 或 Firefox (支持 WebGL)  Python 环境（可选）：用于安装前端开发环境与静态服务器（如 Flask 或 Http.server）  3.前端库：  引入 ECharts 和 ECharts-GL 库，以支持 3D 图形渲染。  引入 CDN 链接。  **三、实验步骤**  1.绘制螺旋曲线：  （1）初始化 HTML 页面结构，并创建一个用于展示图表的 div 容器。  （2）使用 ECharts 初始化 3D 图表实例。  （3）编写螺旋曲线函数，用于生成螺旋曲线的坐标数据，其中半径为5，每旋转一周，z 轴的坐标增加 π。  （4）将生成的数据添加至数组，构成 3D 螺旋线的坐标点。  （5）配置 ECharts 3D 图表的轴和数据系列，设置合适的 x、y 和 z 轴参数，使其能够容纳螺旋数据。  （6）通过 ECharts 的 setOption 方法应用配置，渲染螺旋曲线。  2.绘制二元正态分布曲面：  （1）在 HTML 页面上设置用于显示图表的容器。  （2）初始化 ECharts 3D 图表实例。  （3）定义一个二元正态分布函数，参数包括中心位置、幅度以及 x 和 y 方向的标准差。  （4）遍历 x 和 y 范围，计算每个点的 z 值，并将每个 (x, y, z) 坐标点保存到数组中，构成二元正态分布曲面。  （5）配置 3D 图表选项，包括 x、y 和 z 轴，以及可视化的颜色映射。  （6）使用 setOption 将选项应用到图表实例，渲染出 3D 的正态分布曲面。  3.绘制球面：  （1）创建用于展示球面的 HTML 容器。  （2）初始化 ECharts 3D 图表实例。  （3）配置球面所需的参数方程，通过 ECharts 的 parametricEquation 设置参数 u 和 v 的范围及步长。  （4）定义球面方程，通过正弦和余弦计算 x、y 和 z 的值，使球面具有半径为5的大小。  （5）配置球面的可视化颜色映射，选择适当的颜色以增强 3D 效果。  （6）使用 setOption 将设置应用到图表中，完成球面图形的渲染。  **四、实验结果**  1、螺旋曲线（半径为5，每旋转一周z坐标增加π）  2、二元正态分布曲面  3、球面（半径为5）  **五、源代码**  1、螺旋曲线（半径为5，每旋转一周z坐标增加π）   1. <!DOCTYPE html> 2. <html lang="en"> 3. <head> 4. <meta charset="UTF-8"> 5. <meta name="viewport" content="width=device-width, initial-scale=1.0"> 6. <title>螺旋曲线</title> 7. <style> 8. .container { width: 300px; height: 300px; background-color: rgb(188, 227, 236); } 9. </style> 10. </head> 11. <body> 12. *<!-- 创建用于绘制图表的容器 -->* 13. <div id="spiralChart" class="container"></div> 14. *<!-- 引入 ECharts 和 ECharts-GL -->* 15. <script src="https://cdn.jsdelivr.net/npm/echarts/dist/echarts.min.js"> 16. </script> 17. <script src="https://cdn.jsdelivr.net/npm/echarts-gl/dist/echarts-gl.min.js"> 18. </script> 19. <script> 20. *// 初始化 ECharts 实例* 21. const chart = echarts.init(document.getElementById('spiralChart')); 22. *// 螺旋函数，用于计算螺旋线的坐标* 23. function generateSpiral(radius, factor, angle) { 24. return [radius \* Math.cos(factor \* angle), radius \* 25. Math.sin(factor \* angle), factor \* angle]; 26. } 27. *// 生成螺旋曲线数据* 28. const spiralData = []; 29. for (let angle = 0; angle < 30 \* Math.PI; angle += 0.1) { 30. spiralData.push(generateSpiral(5, 0.5, angle)); 31. } 32. *// 配置 ECharts 3D 螺旋曲线图* 33. const options = { 34. grid3D: {}, 35. xAxis3D: {}, 36. yAxis3D: {}, 37. zAxis3D: { max: 60 }, 38. series: [{ 39. type: 'line3D', 40. data: spiralData, 41. }] 42. }; 43. *// 渲染图表* 44. chart.setOption(options); 45. </script> 46. </body> 47. </html>   2、二元正态分布曲面   1. <!DOCTYPE html> 2. <html lang="en"> 3. <head> 4. <meta charset="UTF-8"> 5. <meta name="viewport" content="width=device-width, initial-scale=1.0"> 6. <title>二元正态分布</title> 7. <style> 8. .container { width: 300px; height: 300px; background-color: 9. rgb(188, 227, 236); } 10. </style> 11. </head> 12. <body> 13. <div id="normalDistChart" class="container"></div> 14. <script src="https://cdn.jsdelivr.net/npm/echarts/dist/echarts.min.js"> 15. </script> 16. <script src="https://cdn.jsdelivr.net/npm/echarts-gl/dist/echarts-gl.min.js"> 17. </script> 18. <script> 19. const chart = echarts.init(document.getElementById('normalDistChart')); 20. function gaussian(amplitude, centerX, centerY, sigmaX, sigmaY) { 21. return function(x, y) { 22. const exponent = -( 23. (Math.pow(x - centerX, 2) / (2 \* Math.pow(sigmaX, 2))) + 24. (Math.pow(y - centerY, 2) / (2 \* Math.pow(sigmaY, 2))) 25. ); 26. return amplitude \* Math.exp(exponent); 27. }; 28. } 29. const gaussianDist = gaussian(50, 0, 0, 20, 20); 30. const data = []; 31. for (let y = -50; y <= 50; y++) { 32. for (let x = -50; x <= 50; x++) { 33. data.push([x, y, gaussianDist(x, y)]); 34. } 35. } 36. const options = { 37. grid3D: {}, 38. xAxis3D: {}, 39. yAxis3D: {}, 40. zAxis3D: { max: 60 }, 41. series: [{ 42. type: 'surface', 43. data: data 44. }] 45. }; 46. chart.setOption(options); 47. </script> 48. </body> 49. </html>   3、球面（半径为5）   1. <!DOCTYPE html> 2. <html lang="en"> 3. <head> 4. <meta charset="UTF-8"> 5. <meta name="viewport" content="width=device-width, initial-scale=1.0"> 6. <title>球面</title> 7. <style> 8. .container { width: 300px; height: 300px; background-color: 9. rgb(188, 227, 236); } 10. </style> 11. </head> 12. <body> 13. <div id="sphereChart" class="container"></div> 14. <script src="https://cdn.jsdelivr.net/npm/echarts/dist/echarts.min.js"> 15. </script> 16. <script src="https://cdn.jsdelivr.net/npm/echarts-gl/dist/echarts-gl.min.js"> 17. </script> 18. <script> 19. const chart = echarts.init(document.getElementById('sphereChart')); 20. const options = { 21. tooltip: {}, 22. visualMap: { show: false, dimension: 2, min: -1, max: 1, inRange: 23. { 24. color: ['#313695', '#4575b4', '#74add1', '#abd9e9', '#e0f3f8', 25. '#ffffbf', '#fee090', '#fdae61', '#f46d43', '#d73027', '#a50026'] 26. }}, 27. xAxis3D: {}, 28. yAxis3D: {}, 29. zAxis3D: {}, 30. grid3D: {}, 31. series: [{ 32. type: 'surface', 33. parametric: true, 34. parametricEquation: { 35. u: { min: -Math.PI, max: Math.PI, step: Math.PI / 20 }, 36. v: { min: 0, max: Math.PI, step: Math.PI / 20 }, 37. x: function (u, v) { return 5 \* Math.sin(v) \* Math.sin(u); }, 38. y: function (u, v) { return 5 \* Math.sin(v) \* Math.cos(u); }, 39. z: function (u, v) { return 5 \* Math.cos(v); } 40. } 41. }] 42. }; 43. chart.setOption(options); 44. </script> 45. </body> 46. </html> |