

《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** | **实验二：统计图可视化** | | | | |
| **要 求** | **（1）按照实验报告模板完成实验报告，并提交到西电智课平台（占85分）。**  **（2）录制5分钟的带头像的视频，演示并介绍自己的作品（占15分）。** | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **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）： 2024年 月 日**

|  |
| --- |
| **一、实验目的及要求**  本实验的目的是通过使用 Python 编程语言及其数据可视化库（如 pyecharts），对给定的数据集进行深入分析和可视化展示。  1.掌握数据读取与处理的基本方法。  2.掌握使用 pyecharts 库进行 3D 数据可视化的技能。  3.分析和展示数据集中的趋势和特征。  4.完成实验报告，呈现实验过程、结果及分析。  **二、实验环境配置**  1.操作系统： Windows 10  2.编程语言： Python 3.8  3.集成开发环境（IDE）： Anaconda with Jupyter Notebook / Visual Studio Code  4.Python 库：  （1）pandas: 用于数据读取和处理  （2）pyecharts: 用于生成 3D 可视化图表  （3）datetime: 用于处理日期格式  **三、实验步骤**  1.数据读取与预处理  使用 pandas 库读取 CSV 文件，检查数据的结构和内容。  对数据进行清洗，包括处理缺失值和格式不一致的问题。  将日期列转换为日期对象，并提取年份和月份。  2.3D 可视化实现  使用 pyecharts 创建 3D 柱状图。  将处理后的数据映射到 X, Y, Z 轴。  生成并保存可视化图表。  3. 处理错误与调试  检查代码运行过程中产生的错误，尤其是日期解析和数据读取部分。  对数据进行过滤，排除无效数据行。  修复并重新运行代码，直到生成正确的可视化结果。  **四、实验结果**  成功生成了一个 3D 柱状图，展示了 1981-1990 年间澳大利亚墨尔本的每日最低气温数据。  图表准确地反映了不同年份和月份的温度变化趋势，验证了数据的有效性。  **五、源代码**   1. from pyecharts import options as opts 2. from pyecharts.charts import Bar3D 3. import pandas as pd 4. import datetime 5. *# 初始化3个列表以存储不同维度的数据* 6. x\_years = []  *# x轴 - 年份* 7. y\_month\_day = []  *# y轴 - 月日* 8. z\_temperature = []  *# z轴 - 温度* 9. *# 设置温度数据列名* 10. temperature\_col = 'Daily minimum temperatures in Melbourne, Australia, 1981-1990' 11. def load\_and\_process\_data(file\_path): 12. *# 读取CSV文件到DataFrame* 13. data\_frame = pd.read\_csv(file\_path) 14. *# 逐行处理数据，并跳过无效条目* 15. for \_, row in data\_frame.iterrows(): 16. *# 确保Date列内容为字符串且非空，同时Unnamed: 2列为空* 17. if isinstance(row['Date'], str) and row['Date'].strip() and pd.isna(row['Unnamed: 2']): 18. try: 19. *# 将日期字符串转换为日期对象* 20. date\_obj = datetime.datetime.strptime(row['Date'], '%Y/%m/%d') 21. x\_years.append(date\_obj.year)  *# 提取年份信息* 22. *# 格式化月日并添加到列表中* 23. y\_month\_day.append(f"{date\_obj.month:02}-{date\_obj.day:02}") 24. *# 存储日期和对应温度值* 25. z\_temperature.append([date\_obj, row[temperature\_col]]) 26. except ValueError as error: 27. print(f"日期格式错误: {row['Date']} - 错误详情: {error}") 28. else: 29. print(f"无效数据，跳过: {row['Date']}") 30. def generate\_3D\_bar\_chart(): 31. global x\_years 32. global y\_month\_day 33. *# 去重并排序* 34. x\_years = sorted(set(x\_years)) 35. y\_month\_day = sorted(set(y\_month\_day)) 36. *# 创建映射字典以映射年份和月日到索引* 37. year\_to\_index = {year: idx for idx, year in enumerate(x\_years)} 38. month\_day\_to\_index = {md: idx for idx, md in enumerate(y\_month\_day)} 39. *# 构建3D图表数据* 40. chart\_data = [] 41. for entry in z\_temperature: 42. x\_index = year\_to\_index[entry[0].year]  *# 获取年份索引* 43. y\_index = month\_day\_to\_index[f"{entry[0].month:02}-{entry[0].day:02}"]  *# 获取月日索引* 44. z\_value = entry[1]  *# 获取温度值* 45. chart\_data.append([x\_index, y\_index, z\_value])  *# 添加到数据集* 46. *# 配置并渲染3D柱状图* 47. bar\_chart = ( 48. Bar3D() 49. .add( 50. series\_name="", 51. data=chart\_data, 52. xaxis3d\_opts=opts.Axis3DOpts(x\_years, type\_="category"),  *# 设置x轴（年份）* 53. yaxis3d\_opts=opts.Axis3DOpts(y\_month\_day, type\_="category"),  *# 设置y轴（月日）* 54. zaxis3d\_opts=opts.Axis3DOpts(type\_="value"),  *# 设置z轴（温度）* 55. grid3d\_opts=opts.Grid3DOpts( 56. width=100, height=100, depth=100, 57. is\_rotate=True, 58. rotate\_speed=10, 59. rotate\_sensitivity=1, 60. ), 61. shading="color"  *# 着色模式* 62. ) 63. .set\_global\_opts( 64. title\_opts=opts.TitleOpts(title="Daily minimum temperatures in Melbourne, Australia, 1981-1990"), 65. visualmap\_opts=opts.VisualMapOpts(min\_=0, max\_=20),  *# 设置视觉映射* 66. ) 67. .render("temperature\_3D\_chart.html")  *# 输出html文件* 68. ) 69. if \_\_name\_\_ == '\_\_main\_\_': 70. csv\_file = "daily-minimum-temperatures-in-me.csv" 71. *# 加载并处理数据* 72. load\_and\_process\_data(csv\_file) 73. *# 生成3D柱状图* 74. generate\_3D\_bar\_chart() |