

实 验 报 告

（ 2023 / 2024学年 第2学期）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 课程名称 | Python数据分析与数据挖掘 | | | | | |
| 实验名称 | 数据预处理 | | | | | |
| 实验时间 | 2024 | 年 | 03 | 月 | 15 | 日 |
| 指导单位 | 武汉纺织大学 | | | | | |
| 指导教师 | 王兆静 | | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 学生姓名 | 江成 | 学号 | 2104240713 |
| 学院(系) | 计算机与人工智能学院 | 专业 | 数据科学与大数据技术 |

|  |  |
| --- | --- |
| 1. **实验目标**   1.了解数据预处理的意义  2.掌握数据清洗、数据变换和数据规约的典型方法   1. **实验内容与要求**   **任务1: 缺失值处理**  **题目要求：**自己设计一个数据集（可以选择以“height\_weight.xlsx” 为基础），利用Pandas完成对数据的导入，分别实现对缺失值所在行的删除和插补。最后，截图展示原始数据和处理后的数据。  **任务2: 离群点处理**  **题目要求：**自己设计一个数据集（可以选择以“height\_weight.xlsx” 为基础），采用  3准则判断离群点，并画图展示结果。  选做：箱型图判断离群点。  **任务3：数据变换与规约**  **题目要求：对**“height\_weight.xlsx”数据集按照列进行归一化处理，并画图展示处理前和处理后数据的分布情况；  选做：对“height\_weight.xlsx”数据集中的样本进行有重复和无重复的采样。 | |
| **三、主要操作步骤及实验结果记录（对实验过程中的主要操作步骤进行描述，并随时记录实验过程中观察到的结果，必要时可辅助截图）**  3.1.算法基本原理  **任务1: 缺失值处理**  1. 数据读取与初步观察  Pandas 是一个强大的Python数据分析工具库，它提供了快速、灵活和表达力强的数据结构，旨在使数据操作和分析更加简单易行。.read\_excel() 函数用于读取Excel文件，而 .info() 和 .describe() 方法提供了数据的基本信息和统计描述，包括每列的数据类型、非空值的数量、平均值、标准差等。  2. 数据预处理  在这一步，我向数据框中添加了新的列。insert() 方法用于在指定的位置插入新的列，而 np.linspace() 函数用于生成在一定范围内的等差数列。  3. 缺失值检测  在数据处理过程中，缺失值是一个常见的问题。Pandas 提供了多种方法来检测和处理缺失值。.isnull() 方法用于检测数据中的缺失值，返回一个布尔类型的DataFrame，其中True表示缺失值。np.any() 函数用于检查一个数组中是否有任何元素为True。  4 .缺失值处理方法  处理缺失值有多种方法，每种方法都有其优缺点。以下是我尝试的几种方法的基本原理：  a. 删除缺失值  删除缺失值是最简单的方法，但可能会导致信息的丢失。.dropna() 方法用于删除包含缺失值的行或列。  b. 替换缺失值  在某些情况下，缺失值可能被标记为特定的字符串（如 ‘nul’）。在这种情况下，首先需要将这些字符串替换为Pandas能够识别的缺失值表示（np.nan）。  c. 填充缺失值  填充缺失值是一种常用的方法，它允许我们保留所有数据，同时为缺失值提供合理的估计。以下是尝试的四种填充方法的基本原理：  使用常量值填充：这种方法为所有缺失值分配一个常数值（在这个例子中是100）。这是一种简单的方法，但可能不会为数据提供准确的表示。  前向填充（ffill）：这种方法使用前一个非缺失值来填充缺失值。这对于时间序列数据或有序数据是有意义的，因为它假设缺失值与前一个值相似。  后向填充（bfill）：与ffill相反，bfill使用后一个非缺失值来填充缺失值。这种方法假设缺失值与后一个值相似。  使用平均值填充：这种方法使用每列的平均值来填充缺失值。这是一种常用的方法，因为它考虑了数据的统计特性，但可能不适用于所有情况（例如，当平均值不是“典型”值时）。  **任务2: 离群点处理**  1. 离群点检测  离群点检测是数据预处理的一个重要步骤，特别是在进行统计分析和机器学习建模之前。离群点可能是由于测量错误、数据输入错误或其他异常情况造成的。  3σ准则 是一种常用的统计方法，用于识别数据集中的异常值。它基于正态分布的性质，其中约68%的数据值位于平均值的1个标准差内，约95%的数据值位于2个标准差内，而约99.7%的数据值位于3个标准差内。因此，超出平均值3个标准差的数据点通常被视为离群点。  在代码中，我定义了一个函数 three\_sigma，它计算数据集的平均值和标准差，并确定了基于3σ准则的上下界。然后，遍历数据集中的每个数据点，检查它是否超出了这些界限。如果是，将它的索引添加到一个列表中，以便后续处理。  2. 数据可视化  数据可视化是一种强大的工具，可以帮助我们更好地理解和解释数据。Matplotlib 是一个Python绘图库，它提供了丰富的绘图功能，用于创建高质量的图表。  在代码中，首先使用散点图绘制所有的数据点，其中正常的点用绿色表示，离群点用红色表示。这样可以帮助我们直观地看到离群点的位置和数量。  删除了离群点后，再次绘制剩余的数据点，这次所有点都是蓝色的。这样可以帮助我们确认离群点已经被成功删除，并且数据集现在更加“干净”。  **箱型图判断离群点**  箱型图（Box Plot）是一种用于显示一组数据分布的统计图表，它通过绘制数据的中位数、四分位数和离群点来描述数据的分散情况。箱型图可以帮助我们快速识别数据中的异常值。  在箱型图中，数据被分为三个部分：   1. 下四分位数（Q1）：数据中的25%位于这个值以下。 2. 中位数（Q2或Median）：数据中的50%位于这个值以下。 3. 上四分位数（Q3）：数据中的75%位于这个值以下。   离群点的判断标准通常使用四分位距（IQR），即上四分位数与下四分位数之间的差值。一般来说，任何低于 Q1 - 1.5 \* IQR 或高于 Q3 + 1.5 \* IQR 的数据点都被视为离群点。  **任务3：数据变换与规约**  1. 最大最小规范化（Min-Max Scaling）  最大最小规范化是一种将数据缩放到固定范围（通常是0到1）的方法。它通过减去数据的最小值并除以数据的范围（最大值减去最小值）来实现。这种方法的优点是简单易行，但缺点是它对异常值敏感，且当数据集的范围变化时，缩放的结果可能会受到影响。  公式：  2. Z-score标准化（Standardization）  Z-score标准化是一种将数据转换为具有零均值和单位方差的方法。它通过从每个数据点中减去均值并除以标准差来实现。这种方法的优点是不受数据集的范围影响，且能够保留数据的原始分布形状。  公式：  **对“height\_weight.xlsx”数据集中的样本进行有重复和无重复的采样**  无重复采样：从数据集中随机选择固定数量的样本，每个样本被选中的概率相等。这种采样方法可能会导致某些样本重复出现，而其他样本可能根本不被选中。  有重复采样：从数据集中随机选择固定数量的样本，并且每个样本都有可能被多次选中。这种方法可以确保每个样本都有相同的机会被选中。  **过采样**  过采样：是一种用于处理分类数据不平衡问题的技术，其中少数类的样本数量通过复制（即过采样）增加到与多数类相似的数量。  SMOTE（Synthetic Minority Over-sampling Technique）：是一种过采样方法，它为少数类创建新的合成样本。SMOTE通过在少数类的样本之间创建人工边界来生成新的样本，从而增加少数类的样本数量。  3.2.步骤与结果展示  **任务1: 缺失值处理**  1. 数据读取与初步观察  首先，读取Excel文件height\_weight\_none.xlsx，并对其进行了初步的观察。通过调用.info()和.describe()方法，可以得到数据的基本信息和统计描述。    2. 数据预处理  为了更好地理解和处理数据，我添加了两个新的列：年龄和时间。其中，年龄列是通过从20到30的范围内随机选择10个数字生成的，而时间列是通过使用NumPy的linspace函数生成的，范围从1到10。    随后，我删除了时间列，在本任务中不需要使用它。    3. 缺失值检测  使用.isnull()方法来检测数据中的缺失值。此外，使用np.any()函数来确定数据框中是否存在任何缺失值。    4. 缺失值处理方法  我尝试了以下几种方法来处理缺失值：  a. 删除缺失值  首先尝试了简单地删除含有缺失值的行。这是通过使用.dropna()方法实现的。    b. 替换缺失值  使用.replace()方法将字符串'nul'替换为NumPy的np.nan对象，以便更方便地处理缺失值。    c. 填充缺失值  尝试了四种不同的方法来填充缺失值：  使用常量值100填充所有缺失值。    使用前向填充（ffill）方法，沿着行的方向用前一个非缺失值填充。    使用后向填充（bfill）方法，沿着列的方向用后一个非缺失值填充。    使用每列的平均值来填充缺失值。    **任务2: 离群点处理**  1. 数据读取  首先，使用Pandas库读取Excel文件height\_weight\_outlier.xlsx，并将其存储为一个DataFrame对象。DataFrame是一种二维数据结构，它包含行和列，可以方便地进行数据操作和分析。  2. 离群点检测  为了检测离群点，此处使0用了3σ准则。首先，将DataFrame对象转换为NumPy数组，以便进行计算。  然后，我定义了一个函数three\_sigma，它接受一个数据数组作为输入，并返回基于3σ准则的上下界。具体来说，它计算数据集的平均值和标准差，并确定了平均值加减2个标准差的上下界。  接下来，遍历数据集中的每个数据点，检查它是否超出了这些界限。如果是，将它的索引添加到一个列表中。  3. 数据可视化  为了更好地理解离群点，我使用matplotlib库进行了数据可视化。首先，我绘制了所有的数据点，其中正常的点用绿色表示，离群点用红色表示。    最后，删除离群点，并再次绘制了剩余的数据点，这次所有点都是蓝色的。    **箱型图判断离群点**  首先，需要计算数据集的下四分位数（Q1）和上四分位数（Q3），以及它们之间的差值，即IQR。然后，根据IQR计算离群点的范围，即下界和上界。接下来，遍历数据集中的每个数据点，检查它是否超出了离群点的范围。如果是，将它的索引添加到一个列表中。最后，使用matplotlib库绘制箱型图，以便直观地看到数据的分布情况，包括离群点的位置。    **任务3：数据变换与规约**  1. 绘制身高和体重的散点图  首先，定义一个函数plot\_height\_weight，它接受一个包含身高和体重数据的二维数组作为输入，并绘制身高和体重的散点图。    2. 最大最小规范化  接下来，使用最大最小规范化方法对数据进行缩放。首先计算数据的最小值和最大值，然后通过减去最小值并除以范围来缩放数据。    3. Z-score标准化  然后，我们使用Z-score标准化方法对数据进行转换。我们首先计算数据的均值和标准差，然后通过从每个数据点中减去均值并除以标准差来进行标准化。    **对“height\_weight.xlsx”数据集中的样本进行有重复和无重复的采样**  使用Python的random模块的sample函数进行无重复采样。sample函数从指定的范围内随机选择指定数量的元素，且每个元素被选中的概率相等。无重复采样    使用randint函数和循环进行有重复采样。randint函数用于生成一个随机整数，范围从low（包含）到high（不包含）。    **过采样**  首先创建一个包含两个特征（x1和x2）和一个目标值（y）的列表。  使用Pandas库将列表转换为DataFrame。    使用Python的Counter类计算目标值（y）的分布，然后绘制散点图。    使用SMOTE类进行过采样。fit\_resample方法接受特征和目标值，并返回过采样后的特征和目标值。计算过采样后的类别分布后绘制出过采样后的散点图。 | |
| **四、实验小结（结合所学知识对实验过程中观察到的实验结果进行分析，心得体会等）**  4.1.遇到的问题，如何解决的  本次实验主要遇到的问题就是 matplotlib 无法显示中文，查询资料后，重新设置字体成功解决。  4.2.总结与收获  ***总结***  数据处理的重要性：在本次实验中，我处理了多个数据集，包括原始数据集、包含缺失值的数据集、包含离群点的数据集等。通过这个过程，我深刻理解了数据预处理的重要性，尤其是在进行数据分析和建模之前。  缺失值处理方法：我学习了多种处理缺失值的方法，包括删除缺失值、填充缺失值等。每种方法都有其优缺点，需要根据具体的数据情况来选择合适的处理方法。  离群点检测与处理：我们使用箱型图和3σ准则来检测离群点，并尝试了删除和填充离群点的方法。这让我明白了离群点对数据分析的影响，以及如何有效地处理这些点。  数据归一化：我们学习了最大最小规范化（Min-Max Scaling）和Z-score标准化（Standardization）两种数据归一化方法。这两种方法都能够将数据缩放到特定的范围，但它们的应用场景和效果不同。  数据可视化：使用matplotlib库绘制了散点图、箱型图等，这让我更加明白了数据可视化在数据分析中的重要性，以及如何使用可视化工具来更好地理解数据。  ***收获***  解决问题的能力：在实验过程中，我遇到了各种问题，如matplotlib无法显示中文、数据处理和分析的困难等。通过查阅资料和尝试不同的方法，我提高了解决问题的能力。  数据预处理技巧：通过处理不同的数据集，我学习到了数据清洗、缺失值处理、离群点检测和处理等数据预处理技巧。  对数据分析的理解：通过本次实验，我对数据分析有了更深入的理解，掌握了数据分析的基本步骤和方法。  编程技能的提升：在实验过程中，我使用了Pandas、NumPy、matplotlib等Python库，一定程度上提高了我的编程技能。 |
| 说明：  实验报告提交到超星学习通 |