山东工商学院
Shandong Technology and Business University

数学建模中的数据处理与数据分析 (上)

邓富文

计算机科学与技术学院

July 27, 2025

目录

模块引言 I

数据：数学建模的基石

在当今数据爆炸的时代，数据已成为驱动科学研究、技术创新和商业决策的核心动力。对于数学建模而言，数据更是其不可或缺的基石。一个优秀的数学模型，无论其理论基础多么深厚，如果缺乏高质量的数据支撑，或者未能对数据进行恰当的处理与分析，其结果的可靠性和实用性都将大打折扣。

数据处理与分析的重要性

数据处理与数据分析是连接原始数据与有效模型的桥梁，它能将看似杂乱无章的数字转化为有价值的信息和洞察，为模型的构建、验证、优化乃至最终的决策提供坚实的基础。

模块引言 II

竞赛中的数据分析题目

在近几年的全国数学建模竞赛中，有很多重点关注数据分析的题目，例如：

- 2020 全国大学生数学建模竞赛 C 题：中小微企业的信贷策略
- 2022 全国大学生数学建模竞赛 C 题：古代玻璃制品的成分分析与鉴别
- 2023 全国大学生数学建模竞赛 C 题：蔬菜类商品自动定价与补货决策

对于其他题目，即使题目本身并不以数据和统计建模为中心，但仍需要掌握基本的数据处理技巧，以顺利完成建模过程或结果数据管理、数据可视化等任务。

本模块核心内容与日程

核心内容

本模块中，我们将从数据的基本概念出发，深入探讨数据清洗、数据集成、数据转换等预处理的关键环节，并通过数据探索性分析揭示数据内在的规律和特征。同时，我们还将通过编程实例，学习数据处理与分析、数据可视化的通用流程和实用技巧，从而将所学知识灵活应用于各类数学建模问题中。

本模块培训日程

- 7 月 26 日上午：MATLAB 数据分析与可视化实践（上）
- 7 月 26 日下午：MATLAB 数据分析与可视化实践（下）
- 7 月 27 日上午：Python 数据分析与可视化基础
- 7 月 27 日下午：大模型驱动的数据处理与数据分析

数据的基本概念 I

Definition

数据是关于事实、概念或指令的一种表示形式，它以正式化的方式进行交流、解释或处理。在数学建模中，我们所接触的数据通常是现实世界中各种现象的量化或分类描述。

数据的基本概念 II

数据的类型

- **定量数据 (Quantitative Data)**：指可以用数值表示的数据，可以进行算术运算。
  - **连续型数据 (Continuous Data)**：在给定区间内可以取任意值的数值数据。例如，身高、体重、温度。
  - **离散型数据 (Discrete Data)**：只能取有限个或可数无限个值的数值数据。例如，班级人数、掷骰子的点数。
- **定性数据 (Qualitative Data)**：指不能用数值表示，或者数值不具有数学意义的数据。
  - **名义型数据 (Nominal Data)**：数据之间没有顺序关系。例如，性别、血型。
  - **有序型数据 (Ordinal Data)**：数据之间存在顺序或等级关系。例如，学历、满意度。
- **时间序列数据 (Time Series Data)**：按时间顺序记录的数据点序列。
- **面板数据 (Panel Data)**：结合了时间序列数据和截面数据的特点。
- **非结构化数据**：文档、图像、视频、声音等。

数据的生命周期

数据从产生到最终被利用并可能被归档或销毁，经历了一系列阶段，这被称为数据的生命周期：

1. **数据采集 (Data Collection)**：根据研究目的，通过各种方法（实验、调查、爬虫等）获取原始数据。
2. **数据存储 (Data Storage)**：将采集到的数据以适当的格式（如 CSV、Excel、数据库）进行保存。
3. **数据处理 (Data Processing)**：对原始数据进行清洗、转换、集成等操作，使其适合后续的分析。
4. **数据分析 (Data Analysis)**：运用统计学、机器学习等方法，从处理后的数据中提取有价值的信息。
5. **数据可视化 (Data Visualization)**：将数据分析结果以图表、图像等直观形式展现出来。
6. **数据应用 (Data Application)**：将数据分析的洞察应用于实际问题中，支持决策、优化流程、预测未来。

数据在数学建模中的作用 I

模型构建的基础

- **数据驱动的建模思想**：许多数学模型并非凭空构建，而是基于对数据的观察和分析。数据可以帮助我们识别变量之间的关系、确定模型的结构形式。
- **参数估计**：在许多数学模型中，模型的具体形式可能已知，但其中的参数需要从数据中学习得到。数据的质量和数量直接影响参数估计的准确性和稳定性。
- **模型验证与校准**：模型构建完成后，需要利用独立的数据集来验证模型的有效性和泛化能力。

问题识别与定义

- **通过数据发现问题**：通过对现有数据的初步探索性分析，我们可以发现数据中的异常、趋势或模式，从而更清晰地定义问题。
- **明确建模目标**：数据分析可以帮助我们量化问题，并设定具体的建模目标。

数据在数学建模中的作用 II

决策支持与结果解释

- **基于数据分析的决策制定**：数学模型的结果往往以数据分析的形式呈现。决策者需要理解这些数据分析结果，才能做出明智的决策。
- **模型结果的合理性评估**：数据分析不仅用于模型构建和验证，也用于解释和评估模型输出的合理性。

数学建模竞赛中常见的数据形式

数学建模竞赛提供的数据通常以文件形式存在，最常见的是以下几种：

- 表格数据：CSV (Comma Separated Values) 文件、Excel (XLSX, XLS) 文件
- 纯文本文件：TXT 文件
- 图像数据 (JPEG, PNG 等)
- 其他数据形式（JSON、XML 等）

加载 CSV/TXT 文件

在 MATLAB 中，加载 CSV 或纯文本文件最推荐的方式是使用 `readtable` 函数，它可以智能地识别数据类型并将其存储为 `table` 数据类型，这在后续的数据处理中非常方便。

代码参考

具体实现请查阅代码文件：`load_csv.m`

加载 Excel 文件

加载 Excel 文件同样推荐使用 `readtable` 函数，它能很好地处理 Excel 文件的多工作表、不同数据类型等特性。对于旧版 MATLAB 或特定需求，也可以使用 `xlsread`。

代码参考

具体实现请查阅代码文件：`load_excel.m`

加载图像文件

对于图像数据，MATLAB 提供了 `imread` 函数来读取图像文件，并将其存储为数值矩阵。图像数据通常是 uint8 类型，像素值范围 0-255。对于灰度图像，是二维矩阵；对于彩色图像（RGB），是三维数组。

代码参考

具体实现请查阅代码文件：`load_image.m`

从剪贴板加载数据

在某些场景下，数据可能直接以表格形式提供在网页上，或者需要从 Excel 等其他应用程序复制粘贴。MATLAB 可以通过 `clipboard` 函数从系统剪贴板读取文本数据。

代码参考

具体实现请查阅代码文件：`load_from_clipboard.m`

数据初步查看与概览

数据加载完成后，立即对数据进行初步的查看和概览是非常重要的步骤。这有助于我们快速了解数据的结构、大小、变量名称以及是否存在明显的问题。

代码参考

具体实现请查阅代码文件：`data_viewing.m`

数据导出

在完成数据的加载、清洗、处理和分析之后，一个常见且重要的步骤是将处理后的结果或中间数据保存到文件中。数据导出有多种目的：

- 结果存档
- 与他人共享
- 跨软件协作
- 报告撰写

在数学建模竞赛中，将数据导出到 Excel 文件非常普遍。我们可以用 `writetable` 函数完成此任务。

MATLAB Table 数据类型简介

什么是 Table?

MATLAB 的 `table` 是一种功能强大的数据容器，专门用于存储列向的、异构类型的数据（即每列可以是不同的数据类型，如数值、文本、分类等），并且可以为行和列指定名称。

Table 的优势

与传统的数值矩阵相比，`table` 的优势在于其结构化和可读性，它非常类似于电子表格或数据库中的表。

访问表格数据 I

MATLAB 提供了多种灵活的方式来访问 `table` 中的数据。

点表示法

这是访问单个列最直接、最常用的方法。它将指定的列数据提取为一个标准的 MATLAB 数组。

括号表示法

使用圆括号进行索引，其结果仍然是一个 `table`。这种方法非常适合用于选择原始表格的一个子集（选择特定的行和列），同时保持表格的结构。

花括号表示法

使用花括号进行索引，用于从表格的单元格中提取原始数据。提取出的数据类型取决于该列原始的数据类型（如数值、cell 等），而不是 `table`。

访问表格数据 II

代码参考

具体实现请查阅代码文件：`data_accessing.m`

修改表格

修改表格内容同样非常灵活，包括增加、删除、重命名列以及修改单元格数据。

代码参考

具体实现请查阅代码文件：`data_editing.m`

表格排序

使用 `sortrows` 函数可以方便地根据一列或多列对表格进行排序。

代码参考

具体实现请查阅代码文件：**`data_sorting.m`**

分组与聚合

核心概念

分组与聚合是 `table` 数据类型最强大的功能之一，类似于 Excel 中的分类汇总和 SQL 中的 `GROUP BY` 操作。通过对数据进行分组，我们可以计算每个组的汇总统计信息。

主要函数

- `groupcounts`: 用于快速统计某个分类变量中每个类别的出现次数。
- `groupsummary`: 功能更为强大，可以按一个或多个分组变量，对一个或多个数据变量计算多种统计量（如均值、总和、标准差等）。

代码参考

具体实现请查阅代码文件：**`data_aggregation.m`**

缺失值处理 (1/2): 概念与方法

什么是缺失值 (Missing Values)?

是指数据集中某些变量的值为空、未知或未记录的情况。它们通常以 `NaN` 在数值数据中表示。缺失值的存在可能导致模型偏差、算法失效、信息损失。

处理方法分类

- **删除法**
  - 删除含有缺失值的观测（行）
  - 删除含有缺失值的变量（列）
- **填充法**
  - 均值/中位数/众数填充
  - 回归填充
  - 插值法（线性插值、多项式插值等）

缺失值处理 (2/2): MATLAB 实现

代码参考

具体实现请查阅代码文件：`missing_value_handling.m`

异常值检测与处理 (1/2): 概念与方法 I

什么是异常值 (Outliers)?

也称离群点，是指数据集中那些显著偏离其他观测值的点。它们可能是由于测量误差、数据录入错误或数据本身固有的稀有事件造成的。

识别方法

- **基于统计的方法**
  - **Z-score 法**: $Z = \frac{x-\mu}{\sigma}$，适用于正态分布。
  - **IQR 倍数法**: 基于 $Q_1 - 1.5 \times IQR$ 和 $Q_3 + 1.5 \times IQR$ 判断。
  - 3σ **原则**: 基于 $(\mu - 3\sigma, \mu + 3\sigma)$ 范围判断。
- **可视化方法**: 箱线图、散点图等。
- **基于模型的方法**: 局部异常因子 (LOF)、隔离森林 (Isolation Forest) 等。

异常值检测与处理 (1/2): 概念与方法 II

处理方法

删除、替换（均值、中位数、截断）或修正（数据变换）。

异常值检测与处理 (2/2): MATLAB 实现

代码参考

具体实现请查阅代码文件：`outlier_handling.m`

重复值处理

什么是重复值 (Duplicate Values)?

是指数据集中存在完全相同或高度相似的记录。重复值可能导致统计偏差、模型过拟合和资源浪费。通常，我们关注的是整行记录的重复。

MATLAB 实现

在 MATLAB 中，可以使用 `unique` 函数来识别并获取唯一行。

代码参考

具体实现请查阅代码文件：**`duplicate_value_handling.m`**

数据集成 (1/2): 异构数据源整合 I

什么是数据集成 (Data Integration)?

是将来自不同来源、不同格式的数据合并到一个统一的数据集中的过程。

MATLAB 中的合并函数

- `join()`, `innerjoin()`, `outerjoin()`: 基于一个或多个共同的键（变量）将两个表格合并。
- `vertcat()`: 垂直拼接两个表格（即增加行），要求两个表格具有相同的列名和数据类型。
- `horzcat()`: 水平拼接两个表格（即增加列），要求两个表格具有相同的行数。

数据集成 (1/2): 异构数据源整合 II

代码参考：Join 与 Concatenate

`data_joining.m`

`data_concatenate.m`

数据集成 (2/2): 数据冲突解决

什么是数据冲突 (Data Conflicts)?

发生在数据集成过程中，当来自不同来源的数据对同一实体或属性提供不一致的值时。常见的冲突包括:

- 命名不一致
- 数据类型不一致
- 值不一致

解决方法

使用 `renamevars()`, `string()`, `double()` 等函数进行统一。

代码参考

`data_conflicts_handling.m`

数据转换：平滑与规范化

数据平滑 (Data Smoothing)

去除数据中噪声或随机波动，揭示其潜在趋势或模式的过程。常用方法包括移动平均、指数平滑等。

代码参考: **data_smoothing.m**

数据规范化/标准化

消除数据量纲对模型训练的影响，使不同特征具有可比性。

- Min-max 规范化: $x_{\text{norm}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$，缩放到 [0,1]。
- Z-score 标准化: $x_{\text{std}} = \frac{x - \mu}{\sigma}$，转换为均值为 0，标准差为 1。

代码参考: **data_normalization.m**

数据转换：离散化与特征工程 I

数据离散化/分箱

将连续型数值数据划分为有限个区间或类别（箱子）的过程。常用方法有等宽分箱、等频分箱、聚类分箱。

代码参考: **data_binning.m**

数据转换：离散化与特征工程 II

特征工程

从原始数据中创造新的特征，以提高模型性能的过程。常见方法包括：

- 从日期/时间中提取特征
- 组合特征
- 多项式特征
- 交互特征
- 聚合特征

代码参考: **feature_engineering.m**

练习 1.1：多种数据源的读取与概览 I

背景

一家市场研究公司提供了一批关于新产品销售情况的数据，这些数据分散在不同的文件中，请你将这些数据全部加载到 MATLAB 工作空间中，并进行初步的查看。

数据文件

- `product_sales.csv`
- `customer_feedback.txt` (Tab 分隔)
- `regional_quota.xlsx`

练习 1.1：多种数据源的读取与概览 II

要求

1. 读取 `product_sales.csv` 到变量 `salesData`。
2. 读取 `customer_feedback.txt` 到变量 `feedbackData`。
3. 读取 `regional_quota.xlsx` 到变量 `quotaData`。
4. 对每个变量，使用 `head` 和 `summary` 函数查看其信息。

练习 1.2：缺失值处理综合应用 I

背景

现有一份环境监测站收集的每日空气质量数据 `air_quality.csv`，其中部分传感器在某些天可能出现故障，导致数据缺失 (`NaN`)。请应用不同的策略来处理这些缺失值。

练习 1.2：缺失值处理综合应用 II

要求

1. 读取 `air_quality.csv` 到变量 `airQualityData`。
2. 找出所有 `NaN` 值的位置。
3. **方法 A（删除法）**：删除所有包含缺失值的行，生成新变量 `data_A`。
4. **方法 B（均值填充）**：对原始数据，使用每列的均值来填充该列的缺失值，生成新变量 `data_B`。
5. **方法 C（线性插值法）**：对原始数据，使用线性插值法来填充缺失值，生成新变量 `data_C`。
6. 打印并比较 `data_A`, `data_B`, `data_C` 的结果。

练习 1.3：数据规范化与标准化应用 I

背景

假设你正在处理一个客户数据集 `customer_data.csv`，其中包含“年龄”和“月收入”两个特征。由于这两个特征的数值范围（量纲）差异巨大，需要对数据进行规范化和标准化处理。

要求

1. 读取 `customer_data.csv` 到变量 `customerData`。
2. 对 `customerData` 进行 Min-Max 规范化，结果存为 `data_normalized`。
3. 对 `customerData` 进行 Z-score 标准化，结果存为 `data_standardized`。
4. 将原始数据、规范化后数据、标准化后数据并排显示，以作对比。

感谢观看！