



中国研究生创新实践系列大赛  
“华为杯”第二十一届中国研究生  
数学建模竞赛

学 校

南京工业大学

参赛队号

24102910010

队员姓名

1.侯睿康

2.白俊华

3.郭潇雨

**中国研究生创新实践系列大赛**

**“华为杯”第二十一届中国研究生**

**数学建模竞赛**

**题 目：数据驱动下磁性元件的磁芯损耗建模**

**摘 要：**

磁性元件作为功率变换器的关键组成部分，其磁芯损耗直接影响设备的整体效率。然而，现有模型难以精确预测损耗，主要因高频磁性材料的复杂微观结构和多种影响因素（如温度、频率、磁通密度及励磁波形）的交织作用。为此，本文提出基于数据驱动的方法，修正传统斯坦麦茨方程，构建适用于多种工况的损耗预测模型。

针对问题一，本文分析了磁通密度的分布特征及不同波形的形状特征。首先，本文对附件中不同类型的波形分别绘制可视化图像，观测不同波形的形状特征，并基于此提取出对应的分布特征。然后，设计了一个基于拐点检测和磁通密度变化的算法来分类三种波（正弦波、三角波和梯形波）。算法的核心是通过磁通密度随时间的变化找到每种波形的特征拐点，并根据拐点的数量和磁通密度的变化规律进行分类。最后对附件中的波形进行识别得到结果。统计结果表明，附件二中被识别出数量最多的波形为三角波，最少的波形为梯形波。

针对问题二，本文在传统斯坦麦茨方程中增加温度修正项，基于差分进化算法对原始斯坦麦茨方程的 3 个参数进行了求解，然后将求得的参数代入斯坦麦茨方程基于附件中的数据进行了检验，利用最小二乘法拟合参数，显著提高了不同温度条件下的损耗预测精度。结果表明，考虑温度影响情况下基于材料属性的修正方程的  $R^2$  要优于原始斯坦麦茨方程，与传统模型相比，修正模型的均方误差（MSE）显著降低。

针对问题三，分析这些因素对损耗的独立及协同影响，并找到使磁芯损耗最小的最优条件。采用多元回归模型分析材料、温度、励磁波形和频率对磁芯损耗的影响，模型考虑了独立效应以及两两因素之间的交互作用。通过对不同组合的实验数据进行分析，计算出每种材料和波形在不同温度下的平均损耗。同时，利用可视化工具进一步分析这些因素的交互作用。通过回归模型的预测与分析，找到了使磁芯损耗最小的最优条件：在材料 4、温度为  $90^{\circ}\text{C}$ 、频率为  $50010\text{Hz}$ 、正弦波形下，磁芯损耗最小值。

针对问题四，在磁芯损耗预测的研究中，针对问题四，使用了随机森林模型

和均方根误差来预测磁芯损耗。实验表明，该模型有效提升了预测精度。总体上，模型的预测趋势与真实值较为吻合，尤其是在样本索引靠前和靠后的部分，模型能够较好地跟踪实际的波动。

针对问题五，以粒子群优化算法（PSO）为目标建立了一个多目标规划模型。并对模型进行了求解。旨在最小化磁芯损耗并最大化传输磁能。通过随机森林模型预测损耗，并引入变异系数法分配权重，最终找到最优设计条件。求解结果表明，最优材料为材料 2，最优温度为 51.91 °C，最优频率为 223855.01 Hz，最优波形类型为方形波（波形 1），最优磁通密度峰值: 0.0231 。

**关键词：** 磁性元件 功率变换器 磁芯损耗 数据驱动 斯坦麦茨方程 随机森林分类器 多元回归模型 温度修正 多模型集成 粒子群优化算法（PSO）

# 目录

一、 问题重述 .....	4
1.1 问题背景 .....	4
1.2 问题提出 .....	4
二、 模型假设与符号说明 .....	6
2.1 模型假设 .....	6
2.2 符号说明 .....	6
三、 数据预处理 .....	8
3.1 指标选取 .....	8
3.2 数据清洗 .....	8
3.3 数据转码 .....	9
3.4 数据描述性分析 .....	10
四、 模型的建立与求解 .....	11
4.1 问题一模型的建立与求解 .....	11
4.1.1 分类模型的建立 .....	11
4.1.2 分类结果 .....	13
4.2 问题二模型的建立与求解 .....	14
4.2.1 修正模型的建立 .....	14
4.2.2 误差分析 .....	15
4.3 问题三模型的建立与求解 .....	16
4.3.1 斯皮尔曼相关性系数 .....	16
4.3.2 回归模型的构建 .....	16
4.3.3 方差分析 (ANOVA) .....	17
4.3.4 回归模型的建立与求解 .....	19
4.3.5 交互模型的建立与求解 .....	20
4.3.6 极值的求解 .....	23
4.4 问题四模型的建立与求解 .....	23
4.4.1 数据处理 .....	23
4.4.2 模型建立与求解 .....	23
4.5 问题五：磁性元件的最优化条件 .....	25
4.5.1 优化模型的建立 .....	25
4.5.2 优化模型的求解 .....	25
五、 模型评价与推广 .....	27
5.1 模型优点 .....	27
5.2 模型缺点 .....	27
5.3 模型推广 .....	27
六、 参考文献 .....	29
七、 附录 .....	30
Python 代码 .....	30

# 一、问题重述

## 1.1 问题背景

随着全球能源结构的转型与电力电子技术的日新月异，电能变换技术作为连接能源生产与消费的关键桥梁，正以前所未有的速度推动着多个行业的深刻变革。在通讯领域，高效稳定的电源系统成为保障信息高速传输的基石；在数据中心，电能的高效转换与利用直接关系到云计算、大数据等前沿技术的可持续发展；而在新能源与电动汽车行业，电能变换技术更是驱动绿色出行、实现碳中和目标的核心技术之一。

在这一背景下，磁性元件作为功率变换器中的“心脏”，其性能优化成为了提升整体系统效率、降低成本、实现紧凑化与轻量化设计的关键所在。磁性元件，尤其是磁芯，在电能转换过程中不仅承载着电磁能量的转换与传递，还直接关联到转换过程中的能量损耗，这些损耗不仅影响设备的能效比，还直接关系到系统的长期运行稳定性与可靠性。然而，磁芯损耗的精确计算却是一项复杂而艰巨的任务。由于磁芯材料的微观结构多样性、工作频率的广泛变化、磁通密度的高频波动以及环境温度的不可预测性，这些因素相互交织，使得磁芯损耗的准确预测变得尤为困难。传统的损耗分离模型虽然试图通过细分损耗成分来提高计算精度，但往往因参数众多、相互耦合复杂而难以在实际应用中实现高精度预测。另一方面，斯坦麦茨方程作为一种广泛应用的经验计算模型，以其形式简单、计算快捷而备受青睐，但在面对非正弦波形、复杂工况及温度波动等实际条件时，其预测精度往往大打折扣，难以满足高精度设计的需求。因此，如何提高磁芯损耗的计算精度成为科研人员和工程师亟待解决的关键挑战。

## 1.2 问题提出

### 问题一：励磁波形分类

在研究励磁波形对磁芯损耗特性的影响时，发现不同波形不仅影响磁通动态变化，还与能量转换中的损耗分布相关。因此，准确分类不同的励磁波形对理解磁芯损耗机制及优化设计至关重要。本问题要求分析实验数据中磁通密度随时间变化的特征，提取相关变量，构建分类模型，识别正弦波、三角波和梯形波，并将结果填入 Excel 表格以进行统计。

### 问题二：斯坦麦茨方程修正

斯坦麦茨方程（Steinmetz equation）是评估铁磁材料在交流电下损耗的经典模型，主要通过磁通密度与频率的幂次乘积来估算铁损。然而，该模型基于正弦波形，未充分考虑温度、磁滞损耗及涡流损耗随温度变化的复杂性，可能导致较大误差。为更准确理解这些机制并优化磁芯设计，本问题将分析斯坦麦茨方程，并针对温度对磁芯损耗的影响进行修正，构建适用于多温度条件的模型。通过实验数据分析，拟合包含温度因素的修正方程，并与原始方程对比，评估两者在损耗预测中的效果和误差差异。

### 问题三：磁芯损耗因素分析

磁芯损耗是评估磁性元件性能的重要指标，主要受温度、励磁波形和材料的影响。本问题将通过先进的数据分析技术，系统地揭示温度、励磁波形及材料特性如何独立且协同地作用于磁芯损耗，通过收集详尽的实验数据，我们将运用多元回归分析方法，构建一个全面反映这些因素的损耗影响模型。此模型不仅将量

化各因素对损耗的独立贡献，还将揭示它们之间的复杂交互作用，为优化磁性元件设计提供科学依据。

#### **问题四：基于数据驱动的磁芯损耗预测模型**

传统磁芯损耗模型在不同工况和材料下的预测精度有限。本问题旨在构建一个广泛适用的高精度磁芯损耗预测模型，利用实验数据建立回归或机器学习模型，分析预测精度和泛化能力，并预测给定样本的磁芯损耗。预测结果将填入 Excel 表格，展示特定样本的损耗值，为磁性元件设计提供参考。

#### **问题五：磁性元件的最优化条件**

为实现磁性元件整体性能优化，需综合考虑损耗和传输磁能。传输磁能可通过频率与磁通密度峰值的乘积进行衡量。本问题将基于第四问题中的损耗预测模型，分析温度、频率、波形、磁通密度峰值和材料的组合条件，找到损耗最小且传输磁能最大的工作条件，最终给出该条件下的磁芯损耗和传输磁能具体值。

## 二、模型假设与符号说明

### 2.1 模型假设

为了便于模型的建立和验证，其假设如下，以确保模型的完备性和预测结果的合理性：

假设 1：假设所提供的数据均为真实且有效的。

假设 2：假设异常数据的出现具有一定的合理性。

假设 3：在磁性元件设计中，磁芯损耗与传输磁能是两个关键且相互制约的性能指标。优化设计需同时考虑两者的最小化与最大化，体现了二者之间的相互影响。

假设 4：传输磁能被定义为频率与磁通密度峰值的乘积，以简化其计算。

假设 5：材料种类、温度、工作频率、励磁波形及磁通密度峰值等输入参数被认为是影响磁芯损耗和传输磁能的主要因素。

假设 6：在优化过程中，假设传输磁能与磁芯损耗之间存在一定独立性，即优化传输磁能时不完全依赖于损耗的变化。

### 2.2 符号说明

为了方便模型的建立与求解过程，这里对使用到的关键符号进行以下说明：

表 2.1 符号说明

符号	符号说明
$F(x)$	总目标函数，结合磁芯损耗与传输磁能的优化目标
$w_1$	磁芯损耗在总目标函数中的权重
$w_2$	传输磁能在总目标函数中的权重
$x$	输入变量（材料、温度、频率、励磁波形、磁通密度峰值）
$\text{core\_loss}(x)$	输入变量对应的磁芯损耗
$\text{transmission\_energy}(x)$	频率与磁通密度峰值的乘积（传输磁能）
$B_m$	磁通密度峰值（T）
$f$	频率（Hz）
$T$	温度（摄氏度）
$m$	磁芯材料种类（材料 1、材料 2、材料 3、材料 4）
$w$	励磁波形类型（正弦波、三角波、梯形波）

表 2.1 符号说明（续）

符号	符号说明
OOB Error	袋外误差（Out-of-Bag Error），评估随机森林模型的泛化能力
$N$	样本数目
$y_i$	第(i) 个样本的真实磁芯损耗值
$\hat{y}_i$	第 (i) 个样本的预测磁芯损耗值
RMSE	均方根误差，用于评估模型的预测精度
PSO	粒子群优化算法（Particle Swarm Optimization）
particles	PSO 中的粒子表示可行解的集合
SwarmSize	粒子群的规模
$w_{opt}$	优化后的加权系数
$f_{val}(x)$	验证集上的目标函数值
predictions matrix	各模型在验证集上的预测结果矩阵
min	最优化操作符，表示目标函数最小化
max	最大化操作符，表示目标函数最大化

注：个别模型中单独使用的符号会在首次引用时进行说明。



### 三、数据预处理

#### 3.1 指标选取

数据预处理是指在进行建模分析之前，对原始数据进行处理和准备的过程。

本题通过统计方法的异常值检测和离散概率模型以及计算机算法的帮助下来检测异常值，并决定是否删除或修正这些异常值。对于不同的问题的模型搭建，根据建模的需求，选择最相关、最具代表性的特征，以减少维度和提高模型效果，具体进行的数据预处理见各题目部分。为了数据分析的更加直观和完善，对一些数据进行必要转换，以确保数据符合建模的假设和要求。

#### 3.2 数据清洗

数据清洗是数据预处理的一个重要步骤，其目标是保证数据质量和准确性，使数据适合建模分析，本题已经保证了数据的准确性和真实性。

首先对数据进行正态分布的检验。为了更加直观的观察异常数据，本研究利用 Python 绘制了频率和磁芯损耗的分布，如图 3.1 所示。

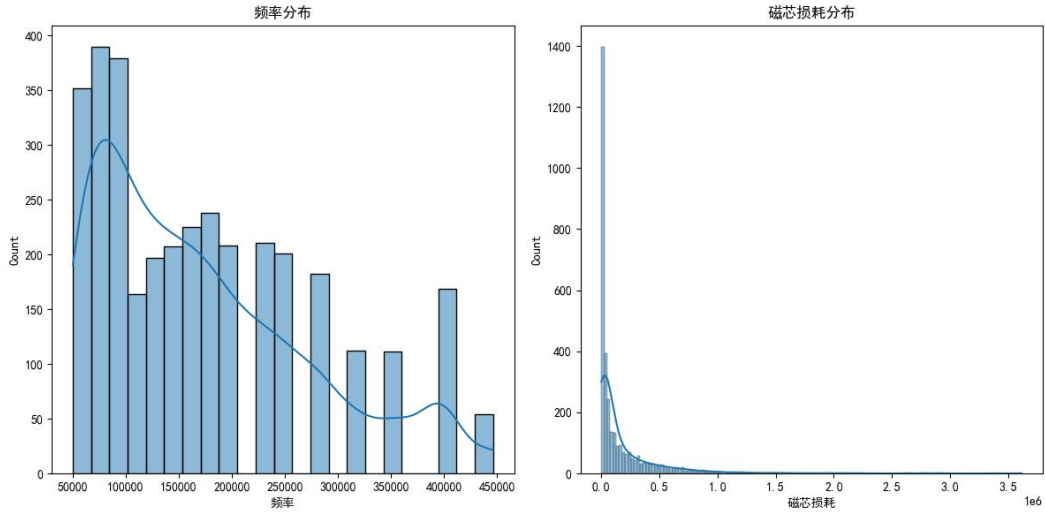


图 3.1 正态分布的检验可视化结果

通过可视化，可以大致看出数据并不服从正态分布，为了进一步验证，需要对题目给出的数据判定其分布方式，在这里我们使用 Kolmogorov-Smirnov 检验。Kolmogorov-Smirnov 检验是一种非参数的统计检验方法，是一种用于检验数据集是否服从某种分布的统计方法，其中最常用的是检验数据集是否服从正态分布。其基本原理是将数据集的累积分布函数与理论分布函数进行比较，通过计算两者之间的最大差距来判断数据集是否符合该理论分布。如果最大差距小于某个临界值，则认为数据集服从该理论分布，单样本的 K-S 检验是用来检验一个数据的观测经验分布是否是已知的理论分布。当两者间的差距很小时，推断该样本取自已知的理论分布。作为零假设的理论分布一般是一维连续分布  $F$  (如正态分布、均匀分布、指数分布等)，有时也用于离散分布（如 Poisson 分布）<sup>[1]</sup>。即  $H$ : 总体  $X$  服从某种一维连续分布  $F$ 。检验统计量为：

$$Z = \sqrt{n} \max_i (|F_n(x_{i-1}) - F(x_i)|, |F_n(x_i) - F(x_i)|) \quad (1)$$

$H$  真， $Z$  依分布收敛于 Kolmogorov-Smirnov 分布。即，当样本取自一维连续分布  $F$  时：

$$Z \rightarrow K = \sup|B(F(x))| \quad (2)$$

注：当 F 是连续分布时，随机变量 K 的分布不依赖于 F。

K-S 检验的结果通常是一个 p 值，如果 p 值小于显著性水平（一般为 0.05），则拒绝原假设，即认为两个样本来自不同的分布。

从表 3.1 所知，频率和磁芯损耗的分布都不服从正态分布。因此，对于不服从正态分布的数据，可以使用箱型图进行异常值判定。箱型图理论无需对数据自身做出限制，不受异常值影响，可以直观的描述数据的离散分布情况，并且提供了一个识别异常值的标准，即大于箱型图设定的上界或小于下界的数值即为异常值。箱型图是利用数据的最小值、第一个四分位数、中位数、第三个四分位数、最大值进行描述数据的一种方法，利用箱型图无需数据服从正态分布。

表3.1 正态分布的检验结果

指标名称	检验结果
频率	非正态分布
磁芯材料	非正态分布
励磁波形	非正态分布

因此对于不服从正态分布的数据，我们对数据进行了箱型图检验。箱型图理论无需对数据自身做出限制，不受异常值的影响，可以更直观地描述数据的离散分布情况。我们使用 Python 中的 boxplot 函数来生成箱型图并标识异常值。箱型图是一种用于显示一组数据分布的图表，其中“异常值”通常被定义为小于  $Q1 - 1.5IQR$  或大于  $Q3 + 1.5IQR$  的值（其中 Q1 是第一个四分位数，Q3 是第三个四分位数，IQR 是四分位间距，即  $0.5(Q3 - Q1)$ ）<sup>[2]</sup>。如图 3.2 展示了异常值判定。

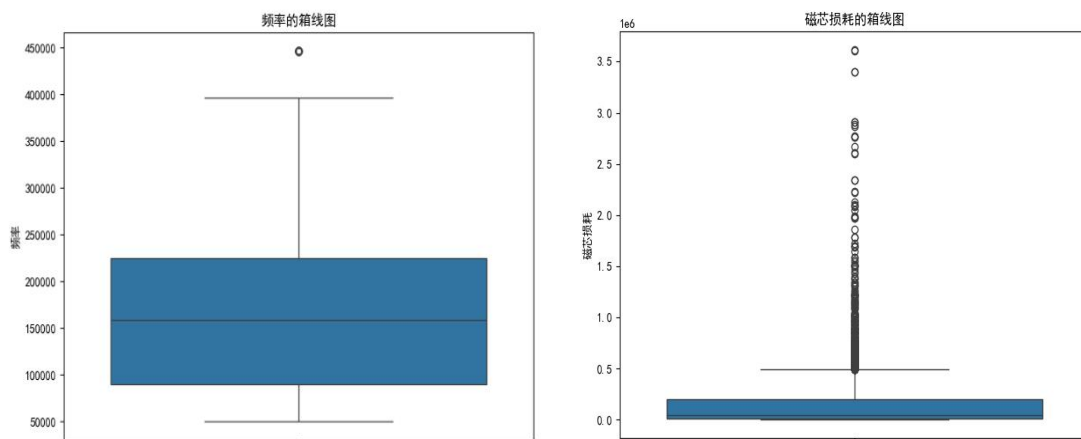


图3.2 异常值判定

### 3.3 数据转码

分类变量的编码：如励磁波形（正弦波、三角波、梯形波）和磁芯材料（材料 1、材料 2、材料 3、材料 4），这些类别特征在建模时不能直接用于数值计算，因此需要进行编码。

表 3.2 分类变量的编码

名称	转码后结果	名称	转码后结果
正弦波	1	材料 1	1
三角波	2	材料 2	2
梯形波	3	材料 3	3
		材料 4	4

### 3.4 数据描述性分析

对数据集的基本统计特征进行总结和描述,帮助更好地了解数据的分布和特性。在描述性分析中,通常计算如均值、中位数、标准差、最大值、最小值等基本统计量,并通过可视化手段呈现数据的分布和趋势。为了更加直观的展示题目给出数据,利用处理后的数据绘制了可视化结果如图 3.3 所示

这组图分别展示了频率和磁芯损耗的描述性统计信息。对于频率,最大值接近 450,000 Hz,最小值较低,均值和中位数相对接近,显示频率数据的分布较为对称。而对于磁芯损耗,最大值远大于均值和中位数,表明数据分布存在较大的偏斜,可能受异常大值的影响,损耗数据呈现出长尾分布。此现象暗示某些样本在特殊工况下表现出极大的损耗。

从上述分析不难得出,数据呈现了不同的变化趋势和分布特性,特别是在磁芯损耗和频率变化上有很大的变动。

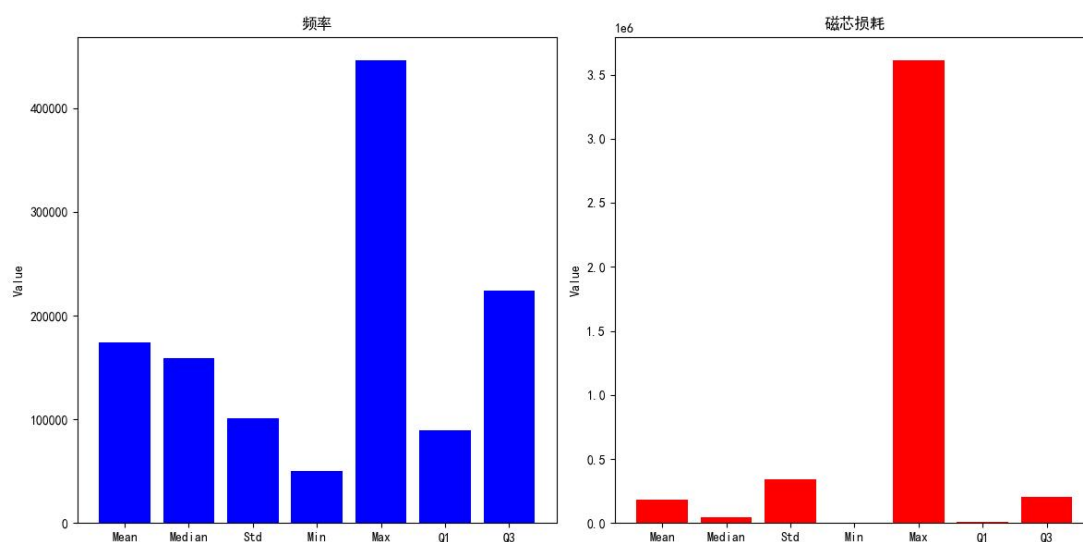


图3.3 频率和磁芯损耗的频数统计

## 四、模型的建立与求解

### 4.1 问题一模型的建立与求解

#### 4.1.1 分类模型的建立

首先，我们需要从提供的附件中读取实验数据，该数据集包含多个样本，每个样本记录了以下信息：

温度：样本测量时的环境温度，通常以摄氏度表示。

频率：电磁波的频率，通常以赫兹（Hz）为单位。

磁忍损耗：在特定条件下材料的能量损失，通常以瓦特每千克（W/kg）表示。

磁通密度：由 1024 个采样点构成的序列，表示在特定时间间隔内材料内部的磁通密度变化。

此外，数据集中还包括一个目标分类变量，即励磁波形，它代表了不同波形类型的标签，这对于后续的分类模型训练至关重要。为了使数据能够应用于分类模型，首先对数据进行以下几步预处理：

（1）提取特征：磁通密度的数据包含多个采样点，是高维时序数据，直接使用这些数据进行分类是不合适的。因此，首先需要提取反映波形特征的变量。假设磁通密度随时间变化为  $B(t)$ ，提取的主要特征包括：

- 均值  $\mu = \frac{1}{T} \sum_{i=1}^T B_i$
- 标准差  $\sigma = \sqrt{\frac{1}{T} \sum_{i=1}^T (B_i - \mu)^2}$
- 最大值  $\max(B_i)$
- 最小值  $\min(B_i)$
- 峰值位置：表示磁通密度达到最大和最小值的采样点位置。

这些统计特征帮助将复杂的时序数据简化为有限的特征向量  $X$ ，便于后续的分类建模。

（2）目标变量的数值化：数据集中励磁波形以字符的形式表示（如"正弦波"、"三角波"、"梯形波"），为了使模型可以识别这些类别，将其转换为数值标签。具体地，设  $y$  表示目标变量：

- 正弦波记为  $y = 1$
- 三角波记为  $y = 2$
- 梯形波记为  $y = 3$

因此，对于每个样本，得到特征向量  $X$  和对应的波形类型标签  $y$ 。

（3）划分训练集与验证集：为了评估模型的性能，将数据集划分为训练集和验证集。通过交叉验证的方法，随机保留 80% 的数据作为训练集，剩余的 20% 作为验证集。具体划分方式可以通过以下公式描述：

$$\text{Training Set: } (X_{\text{train}}, y_{\text{train}}), \text{ Validation Set: } (X_{\text{val}}, y_{\text{val}}) \quad (3)$$

其中： $X_{\text{train}}$  表示训练集的特征矩阵， $y_{\text{train}}$  表示训练集的波形标签， $X_{\text{val}}$  和  $y_{\text{val}}$  分别为验证集的特征矩阵和波形标签。使用交叉验证能够有效避免模型过拟合，并确保模型的泛化能力。

随机森林 (Random Forest) 是一种集成学习方法，属于监督学习算法，用于

分类和回归问题。它是通过创建多个决策树并将它们的结果组合（如通过投票机制）来进行最终决策。随机森林的核心思想是利用集成多个弱分类器（决策树）的力量，产生一个强分类器，具有强大的抗过拟合能力和良好的泛化性能。

随机森林通过随机性来增强模型的鲁棒性和多样性，使其能在各种复杂数据集上表现良好，其主要特点包括：

- 1) 随机特征选择：在构建每棵决策树时，随机选择特征子集。
- 2) 数据子集采样（袋外样本, OOB）：在训练每棵树时，使用的是随机从训练集的样本中抽取的子集。
- 3) 集成方法：每棵树独立构建，最终的分类结果通过投票机制或平均输出进行决策。

随机森林结合了 Bootstrap 采样和随机特征选择这两大技术。Bootstrap 采样：在训练随机森林时，对于每棵树，算法会从原始训练数据集中进行有放回的随机采样，这意味着每棵树的训练集可能包含重复的样本，并且有些样本可能未被选中。这些未被选中的样本称为 袋外样本（Out-Of-Bag, OOB）。它们可以用于估计模型的泛化误差，而无需额外的验证集，其过程如下：

- 1) 给定一个训练数据集  $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ ，每棵树会从  $D$  中随机抽取  $n$  个样本形成一个新的训练集  $D_b$ ，该过程是有放回的。
- 2) 有放回的抽样保证了训练集中每棵树都能得到不同的子集，这样每棵树的训练数据具有差异性，增强了模型的  $\downarrow$  性。

随机特征选择：每棵树在构建过程中，不是对所有特征进行划分，而是在每个节点处随机选择一个特征子集。这样做是为了防止某些特征在所有树中占主导地位，进一步提高模型的多样性和抗过拟合能力，其过程如下：

- 3) 假设有  $p$  个特征，在每个决策树节点的分裂时，不是对所有  $p$  个特征进行评估，而是随机选择其中的  $m$  个特征来寻找最佳分裂。
- 4) 通常  $m$  的值为  $p$  的平方根，即  $m = \sqrt{p}$ ，但也可以根据具体问题进行调整。

随机森林的分类过程分为以下几个步骤：

#### 1) 决策树的构建

每棵决策树  $T_i$  的构建遵循如下步骤：

- Step 1: 从训练数据集中，使用 Bootstrap 采样 生成新的子训练集  $D_b$ 。
- Step 2: 在每个决策树的节点处，随机选择部分特征进行分裂选择。
- Step 3: 使用这些随机选择的特征构建决策树，递归地划分训练集，直到满足树停止生长的条件，如达到最大深度或叶节点数目较少。
- Step 4: 对每个叶节点，记录其所属类别，该类别是叶节点中最多样本的类别。

#### 2) 分类预测

在分类任务中，随机森林的分类预测通过所有决策树的投票结果来决定最终分类。假设随机森林包含  $T$  棵决策树，每棵树  $T_i$  对样本  $x$  产生一个预测结果  $\hat{y}_i$ ，最终的预测结果是由多数投票决定的：

$$\hat{y} = \text{Mode}(\{\hat{y}_1, \hat{y}_2, \dots, \hat{y}_T\}) \quad (4)$$

其中，Mode 函数表示取值出现频率最多的类别。

#### 3) 袋外样本 (OOB) 估计

每棵树在训练时使用的是从训练集中随机采样的子集，未被选中的样本称为

袋外样本（OOB 样本）。由于这些样本未用于训练模型，因此它们可以用来评估模型的泛化性能。在随机森林中，OOB 误差的计算不需要独立的验证集，它可以作为交叉验证的替代方法。

OOB 误差的计算公式如下：

对于每个袋外样本  $x_i$ ，它未被部分树  $T_i$  训练过。因此，使用这些树来预测  $x_i$  的类别，假设  $x_i$  的真实类别为  $y_i$ ，袋外误差  $E_{\text{OOB}}$  的计算为：

$$E_{\text{OOB}} = \frac{1}{N} \sum_{i=1}^N \mathbb{I}(\hat{y}_i \neq y_i) \quad (5)$$

其中， $N$  是样本的总数， $\mathbb{I}$  是指示函数，当预测结果与真实结果不同时取 1，否则取 0。

随机森林具有自动计算特征重要性的能力。通过对特征的随机扰动来观察模型性能的变化，能够得出每个特征在分类中的贡献度<sup>[3-8]</sup>，具体过程如下：

- 1) 对某个特征进行打乱，即将该特征的值随机排列，重新计算模型的 OOB 误差。如果打乱某个特征导致误差显著增加，说明该特征对分类决策有重要影响。
- 2) 每个特征的重要性  $V_i$  可以通过以下公式度量：

$$V_i = \text{OOB 误差打乱特征} i - \text{OOB 误差原始} \quad (6)$$

该值越大，表明该特征越重要。

#### 4.1.2 分类结果

模型训练完成后，使用验证集对模型进行评估，计算其分类准确率。假设预测结果为  $\hat{y}_{\text{val}}$ ，验证集上的准确率  $A$  可以通过以下公式计算：

$$A = \frac{1}{N_{\text{val}}} \sum_{i=1}^{N_{\text{val}}} \mathbb{I}(\hat{y}_i = y_{\text{val},i}) \quad (7)$$

式中， $N_{\text{val}}$  表示验证集样本数量， $\hat{y}_i$  是模型对第  $i$  个验证样本的预测结果， $y_{\text{val},i}$  是验证集第  $i$  个样本的真实标签  $\mathbb{I}$  是指示函数，当预测正确时取值为 1，否则取值为 0。

通过验证集的准确率，可以评估模型的分类效果。

在模型验证完成后，使用已训练好的模型对附件二中的测试数据进行预测。首先，需要对测试数据进行相同的特征提取处理，以确保其格式与训练数据一致。我们将从测试数据中提取出温度、频率、磁芯损耗和磁通密度信息，并应用与训练数据相同的标准化和编码技术，将其转化为特征矩阵  $X_{\text{test}}$ 。随后，利用训练好的随机森林模型进行预测，通过调用模型的预测函数，我们可以对每个测试样本进行分类，得到对应的励磁波形类型预测结果：

$$\hat{y}_{\text{test}} = \text{TreeBagger}(X_{\text{test}}) \quad (8)$$

表4.1 问题二预测结果

序号	类型	序号	类型	序号	类型	序号	类型
1	三角波	5	三角波	15	正弦波	25	三角波
35	三角波	45	三角波	55	三角波	65	三角波
75	三角波	80	正弦波				

最后，将预测结果整理成一个清晰的格式，包括每个测试样本的特征值及其预测的波形类型，可以将这些结果保存为新的数据文件，以便于后续分析和验证。题目二结果如表 4.1 所示，其中正弦波有 18 个，三角波有 53 个，梯形波有 9 个。

## 4.2 问题二模型的建立与求解

### 4.2.1 修正模型的建立

在电力电子与磁性元件的研究中，磁芯损耗是影响系统效率的关键指标。磁芯损耗的主要影响因素包括频率、磁通密度的峰值、工作温度等。经典的斯坦麦茨方程（Steinmetz Equation）为磁芯损耗的计算提供了一种经验模型，该模型在正弦波形下能够较好地预测磁芯损耗。然而，随着工作条件的多样化，尤其是温度的变化，斯坦麦茨方程的适用性受到了限制。因此，在该研究中，对斯坦麦茨方程进行了拟合，并基于实验数据构建了一个温度修正模型，旨在提高不同温度条件下磁芯损耗预测的精度。

斯坦麦茨方程是一个经验公式，用于计算正弦波形下的磁芯损耗。其形式为：

$$P = k_1 \cdot f^a \cdot B_m^b \quad (9)$$

式中  $P$  为磁芯损耗，单位为  $\text{W/m}^3$ ， $f$  为频率，单位为  $\text{Hz}$ ， $B_m$  为磁通密度的峰值，单位为  $\text{T}$ ， $k_1$ 、 $a$  和  $b$  为待拟合的参数，通过实验数据进行估计。

斯坦麦茨方程表明，磁芯损耗是频率和磁通密度峰值的幂函数，其中  $a$  和  $b$  分别控制了损耗对频率和磁通密度的敏感性。

#### 1) 非线性回归求解

为了从实验数据中拟合斯坦麦茨方程中的参数  $k_1$ 、 $a$  和  $b$ ，使用了非线性最小二乘法。给定一组实验数据，包括频率  $f$ 、磁通密度峰值  $B_m$  和对应的磁芯损耗  $P$ ，最小化以下目标函数：

$$\min_{k_1, a, b} \sum_{i=1}^n (P_i - k_1 \cdot f_i^a \cdot B_{m,i}^b)^2 \quad (10)$$

式中  $P_i$  是第  $i$  个样本的实际损耗值， $f_i$  和  $B_{m,i}$  分别是第  $i$  个样本的频率和磁通密度峰值， $n$  为样本总数。

#### 2) 拟合结果

通过非线性回归，得到传统斯坦麦茨方程的拟合参数为： $k_1 = 1.4997$ ， $a = 1.4296$ ， $b = 2.4713$ 。

这些参数表明，磁芯损耗随频率的增加以  $f^{1.4296}$  的速度增长，而随磁通密度峰值的增加以  $B_m^{2.4713}$  的速度增长。拟合后的斯坦麦茨方程可以写作：

$$P = 1.4997 \cdot f^{1.4296} \cdot B_m^{2.4713} \quad (11)$$

#### 3) 斯坦麦茨方程的温度修正

虽然传统斯坦麦茨方程能够有效地预测正弦波形下的磁芯损耗，但在温度变化的情况下，损耗会发生显著的波动，传统方程无法解释这些变化。因此，为了提高磁芯损耗预测在不同温度条件下的准确性，引入了一个温度修正项。修正后的斯坦麦茨方程形式为：

$$P = k_1 \cdot f^a \cdot B_m^b \cdot (1 + c \cdot T) \quad (12)$$

式中， $T$  为温度，单位为摄氏度， $c$  为温度修正系数。

该修正项  $(1 + c \cdot T)$  引入了温度  $T$  对磁芯损耗的影响，参数  $c$  控制了损耗



随温度变化的幅度。当  $c > 0$  时，温度升高会增加磁芯损耗，而  $c < 0$  时，温度升高会减小损耗。

类似于传统斯坦麦茨方程的拟合方法，通过最小化损耗预测与实际损耗之间的误差来拟合修正后的方程。目标函数为：

$$\min_{k_1, a, b, c} \sum_{i=1}^n \left( P_i - k_1 \cdot f_i^a \cdot B_{m,i}^b \cdot (1 + c \cdot T_i) \right)^2 \quad (13)$$

式中  $T_i$  是第  $i$  个样本的温度。

通过非线性回归，得到修正斯坦麦茨方程的拟合参数为： $k_1 = 1.8712$ ， $a = 1.4385$ ， $b = 2.4324$ ， $c = -0.0054$ 。修正后的斯坦麦茨方程可以写作：

$$P = 1.8712 \cdot f^{1.4385} \cdot B_m^{2.4324} \cdot (1 - 0.0054 \cdot T) \quad (14)$$

修正结果表明，随着温度的升高，磁芯损耗略有减小（因为  $c$  的值为负）。这表明在高温下，材料的损耗特性可能会有所改善。

#### 4.2.2 误差分析

为了评估模型的预测效果，计算了均方误差（Mean Squared Error, MSE），即模型预测值与实际值之间的平均平方差异。对于传统斯坦麦茨方程和修正斯坦麦茨方程，均方误差分别为 1638190471.6410 和 273567131.0633。

显然，修正后的斯坦麦茨方程具有更低的均方误差，说明温度修正项显著提高了模型的预测精度。具体结果比对如图 4.1 所示。

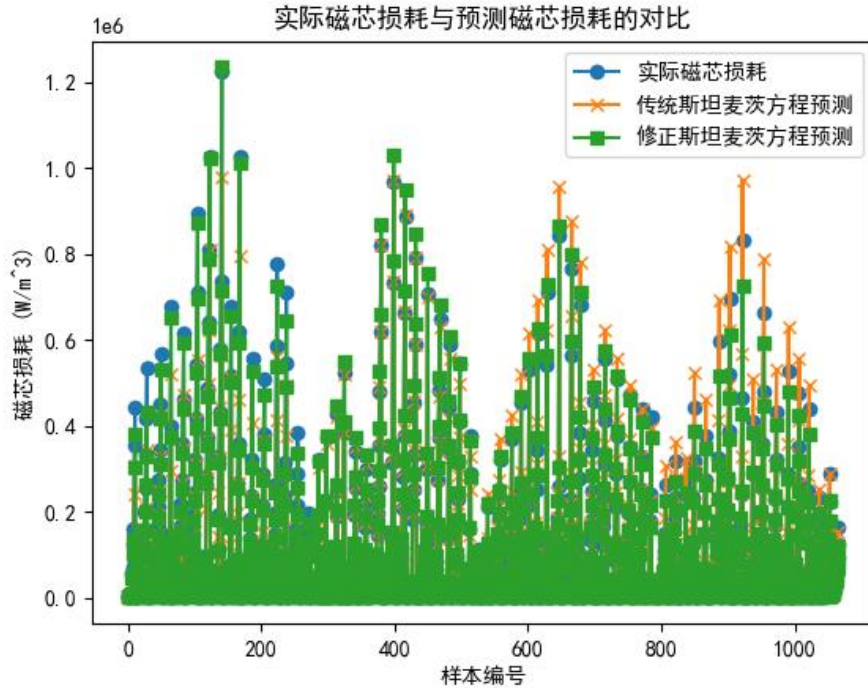


图 4.1 实际磁芯损耗与预测磁芯损耗的对比

实际磁芯损耗与两种斯坦麦茨方程（传统斯坦麦茨方程和修正斯坦麦茨方程）的预测结果之间的对比。横轴代表样本编号，纵轴表示磁芯损耗，单位为  $W/m^3$ 。



实际磁芯损耗:蓝色圆圈表示实际的磁芯损耗值,图中显示出实际损耗随样本编号的波动。实际磁芯损耗值在不同的样本中呈现出较大的波动,特别是在某些样本上,损耗值会显著上升,达到峰值。

传统斯坦麦茨方程预测值:红色叉号代表传统斯坦麦茨方程的预测结果。从图中可以看到,传统斯坦麦茨方程的预测趋势与实际值大体一致,但在某些峰值区域,预测结果偏离较大,尤其是在样本编号较高的区域,传统方程的预测结果略微高估了实际损耗。

修正斯坦麦茨方程预测值:黄色方形代表修正后的斯坦麦茨方程的预测结果。

相较于传统方程,修正后的斯坦麦茨方程的预测曲线更加紧密地贴合实际损耗值,特别是在损耗较高的区域,修正模型显著减少了误差,预测结果与实际值非常接近。

通过该图,可以明显看出修正后的斯坦麦茨方程在预测精度上比传统斯坦麦茨方程有了明显提升,特别是在较大损耗值的情况下,修正模型能够更好地捕捉温度等外界因素对磁芯损耗的影响,从而提供更准确的预测结果。

### 4.3 问题三模型的建立与求解

#### 4.3.1 斯皮尔曼相关性系数

温度斯皮尔曼相关性系数是用于度量两个变量之间的单调关系的非参数统计方法。它基于样本的秩次来计算相关性,适用于线性和非线性单调关系。斯皮尔曼相关性系数  $\rho$  的计算公式为:

$$\rho = 1 - \frac{6 \sum_{i=1}^n d_i^2}{n(n^2-1)} \quad (15)$$

其中:  $d_i = R(X_i) - R(Y_i)$  是第  $i$  个样本在两个变量中的秩次差。

$R(X_i)$  表示第  $i$  个样本在变量  $X$  中的秩次。

$R(Y_i)$  表示第  $i$  个样本在变量  $Y$  中的秩次。

#### 4.3.2 回归模型的构建

我们首先可以构建一个线性回归模型,包含三个因素:温度、励磁波形和材料类型,以及它们的交互作用。回归模型的形式可以表示为:

$$y = \beta_0 + \beta_1 T + \beta_2 W + \beta_3 M + \beta_4 T \cdot W + \beta_5 T \cdot M + \beta_6 W \cdot M + \varepsilon \quad (16)$$

其中:  $y$  表示磁芯损耗。

$T$  表示温度,单位为摄氏度 ( $^{\circ}\text{C}$ )。

$W$  表示励磁波形,编码为分类变量(正弦波、三角波、梯形波)。

$M$  表示磁芯材料类型(材料1、材料2、材料3、材料4)。

$\beta_0, \beta_1, \beta_2, \beta_3, \beta_4, \beta_5, \beta_6$  是模型中的回归系数,需要通过数据拟合确定。

$\varepsilon$  是误差项,表示模型未能解释的随机波动。

### 4.3.3 方差分析 (ANOVA)

方差分析用于分析多个自变量对因变量的影响，尤其是用于确定不同因素及其交互作用是否显著影响磁芯损耗。在我们的问题中，我们需要分析温度、励磁波形和材料类型三个因素及它们的交互作用方差分析的数学模型形式为：

$$y_{ijk} = \mu + \alpha_i + \beta_j + \gamma_k + (\alpha\beta)_{ij} + (\alpha\gamma)_{ik} + (\beta\gamma)_{jk} + \varepsilon_{ijk} \quad (17)$$

其中： $y_{ijk}$  是第  $i$  类温度、 $j$  类励磁波形、 $k$  类材料类型下的磁芯损耗。

$\mu$  是总体均值。

$\alpha_i$  是温度的主效应。

$\beta_j$  是励磁波形的主效应。

$\gamma_k$  是材料类型的主效应。

$(\alpha\beta)_{ij}$  是温度与励磁波形的交互效应。

$(\alpha\gamma)_{ik}$  是温度与材料类型的交互效应。

$(\beta\gamma)_{jk}$  是励磁波形与材料类型的交互效应。

$\varepsilon_{ijk}$  是误差项。

通过方差分析，我们可以计算各个因素及其交互作用的  $p$  值，以判断它们对磁芯损耗的影响是否显著。

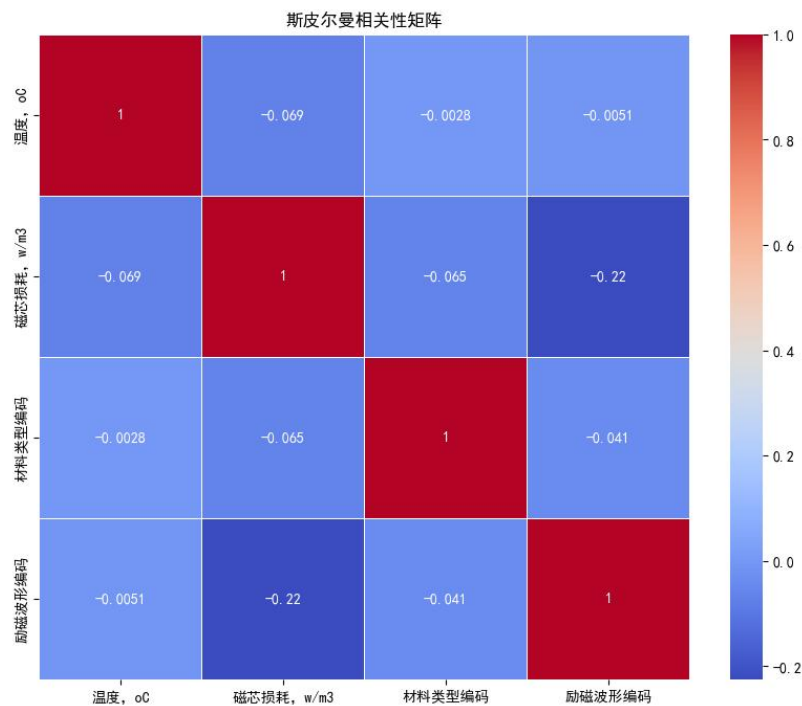


图 4.2 斯皮尔曼相关性矩阵

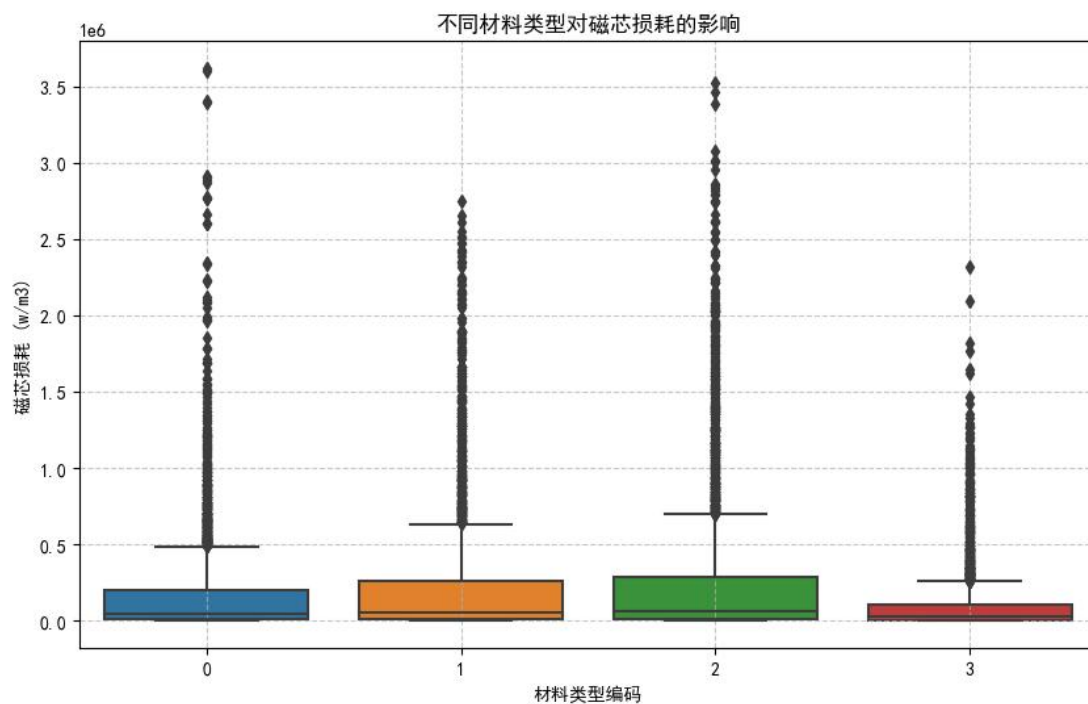


图 4.3 不同材料类型对磁芯损耗的影响

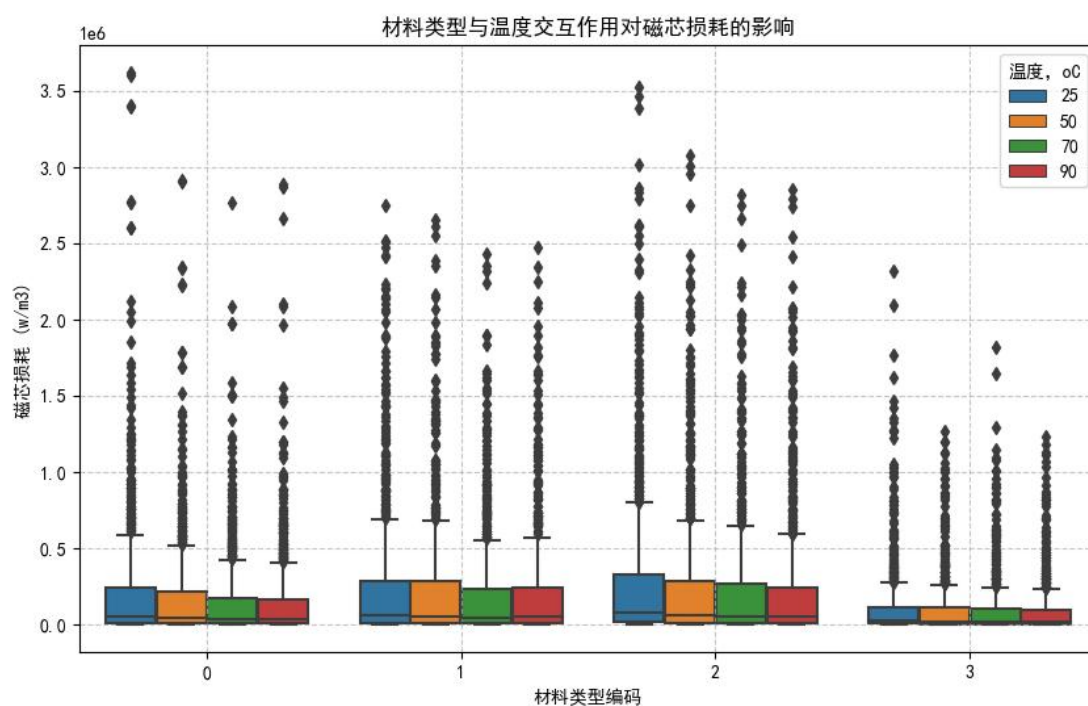


图 4.4 材料类型与温度交互作用对磁芯损耗的影响

由图 4.2 可知，各个因素及其交互作用都是负相关，并且波形负相关性最强。从方差分析（ANOVA）的结果中，我们可以解读每个因素以及它们的交互作用对磁芯损耗的显著性影响。如果  $p$  值小于设定的显著性水平（通常是 0.05），则可以认为该因素对磁芯损耗的影响显著。由上述分析可知，材料类型、温度、

励磁波形对磁芯损耗的影响都非常显著。 $p$  值远小于 0.05, 说明不同材料类型在统计上对磁芯损耗存在显著差异。 $F$  值较大, 表示材料类型对磁芯损耗的影响程度较强。材料类型和励磁波形之间的交互作用对磁芯损耗有显著影响。 $p$  值远小于 0.05, 表明材料类型和励磁波形之间的交互效应显著影响磁芯损耗。 $F$  值相对较大, 表明该交互作用的重要性。温度和励磁波形的交互作用对磁芯损耗没有显著影响。 $p$  值大于 0.05, 说明温度和励磁波形的相互作用对磁芯损耗的影响不显著。

综上, 材料类型、温度、和 励磁波形 这三个主效应都对磁芯损耗有显著影响, 尤其是励磁波形对损耗的影响最为强烈 ( $F$  值最大)。材料类型与温度之间的交互作用 对损耗没有显著影响 ( $p$  值大于 0.05), 说明两者的组合对磁芯损耗的影响是独立的。材料类型与励磁波形的交互作用 显著, 表明不同材料在不同励磁波形下表现出不同的损耗特性。温度与励磁波形的交互作用不显著, 说明温度和波形的组合对损耗的影响并不显著。

#### 4.3.4 回归模型的建立与求解

在磁性元件的设计与优化过程中, 磁芯损耗是一个至关重要的指标。影响磁芯损耗的因素多种多样, 其中包括材料的类型、工作温度、励磁波形的形式以及工作频率等。为了深入理解这些因素对磁芯损耗的独立和协同作用, 对实验数据进行了回归分析, 并详细研究了各因素之间的交互效应。特别地, 分析了材料、温度、频率和励磁波形对磁芯损耗的独立作用以及两两交互作用, 并找出了最优条件下的最低磁芯损耗及其对应的工作参数。

##### 1) 数据处理与回归模型构建

在实验数据集中, 包含了材料类型 (材料 1、材料 2、材料 3、材料 4)、温度 (25 °C、50 °C、70 °C 和 90 °C)、工作频率 (50 kHz 到 500 kHz)、磁芯损耗以及励磁波形 (正弦波、三角波和梯形波)。这些数据用于构建回归模型, 以量化每个因素的影响, 并分析它们的交互作用。

选择了线性回归模型 (Linear Regression Model, LRM) 来进行拟合, 模型的结构如下:

$$\text{core\_loss} = \beta_0 + \beta_1 \cdot M + \beta_2 \cdot T + \beta_3 \cdot f + \beta_4 \cdot W + \beta_5 \cdot (M \cdot T) + \dots + \epsilon \quad (18)$$

其中, 核心变量包括材料 ( $M$ )、温度 ( $T$ )、频率 ( $f$ ) 和励磁波形 ( $W$ ) 的独立效应, 模型中还引入了它们之间的交互项, 如材料与温度的交互、温度与励磁波形的交互、材料与波形的交互等。通过线性回归模型中的系数  $\beta_i$  来衡量各个变量对磁芯损耗的影响程度。回归分析中, 响应变量为磁芯损耗, 单位为  $\text{W/m}^3$ , 而自变量则为材料、温度、频率和波形。

##### 2) 回归模型结果

通过对数据集进行线性回归拟合, 模型的系数和显著性水平可以帮助理解不同因素的独立影响和交互作用。模型中的回归系数  $\beta_i$  代表了相应因素对磁芯损耗的影响大小:

材料: 不同材料对磁芯损耗的影响显著, 不同材料的损耗水平差异较大。回归模型通过系数分析得出, 材料类型会显著影响损耗。

温度: 随着温度的升高, 磁芯损耗呈现非线性变化。较高温度下, 某些材料的损耗水平会降低, 这取决于材料的特性。

频率: 频率对损耗的影响依旧显著。回归系数表明, 损耗随频率的增加而升高, 且这种影响程度较大。

励磁波形：不同的波形也会导致磁芯损耗的显著变化。正弦波、三角波和梯形波对材料的损耗影响不同，波形的形态决定了磁通密度的变化速率，从而影响损耗。

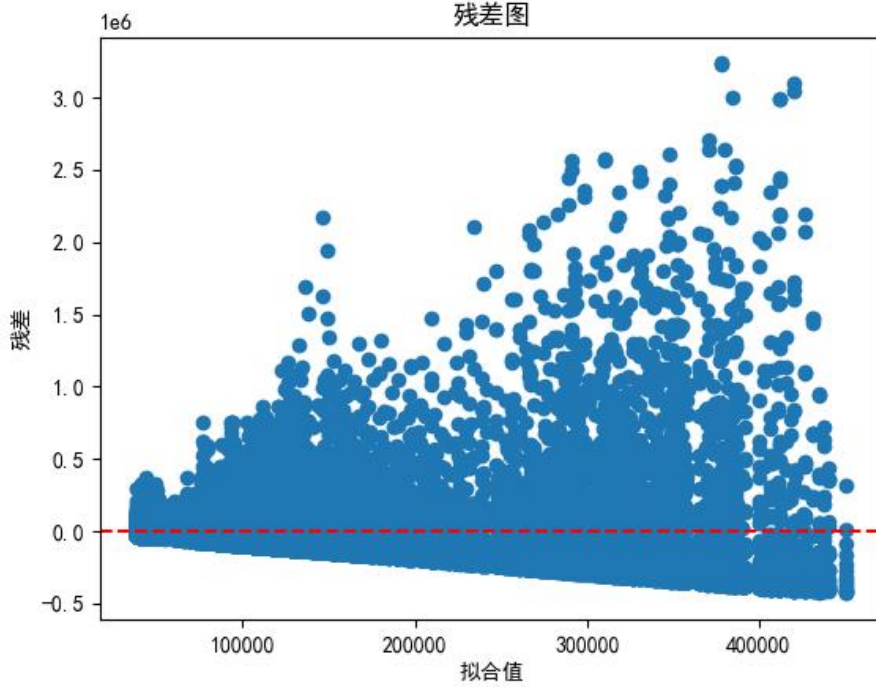


图 4.5 残差图

由图 4.5 可知，残差的散布较为广泛，且存在较多偏离零点的较大残差，尤其在较高磁芯损耗的区域，残差较大，这表明模型在预测较高损耗值时不够准确。残差图显示出一定的非线性趋势，暗示着模型可能未能捕捉到数据中的非线性关系或复杂交互效应，表明线性回归可能不足以完全描述数据中的复杂性。材料、频率和励磁波形对磁芯损耗具有显著影响，尤其是材料 2 和材料 3 表现出较高的损耗特性。同时，频率的增加也显著提高了磁芯损耗。

材料与波形的交互效应 在模型中尤为显著，特别是正弦波形下，材料 2 和材料 3 的损耗较高，这表明在实际应用中，不同材料在不同波形条件下对损耗的表现可能存在显著差异。

#### 4.3.5 交互模型的建立与求解

##### 1) 温度与励磁波形的交互作用分析

为研究温度和励磁波形的交互作用，计算了不同温度和波形组合下的平均磁芯损耗。设定温度  $T$  和波形  $W$  的组合，通过下式计算平均损耗：

$$\bar{P}_{T,W} = \frac{1}{n_{T,W}} \sum_{i=1}^{n_{T,W}} P_i \quad (19)$$

式中， $n_{T,W}$  表示在特定温度  $T$  和波形  $W$  条件下的样本数量， $P_i$  为磁芯损耗。

计算结果表明，在较高温度下，三角波形的损耗普遍较低，而正弦波形和梯形波形在较低温度时表现较好。

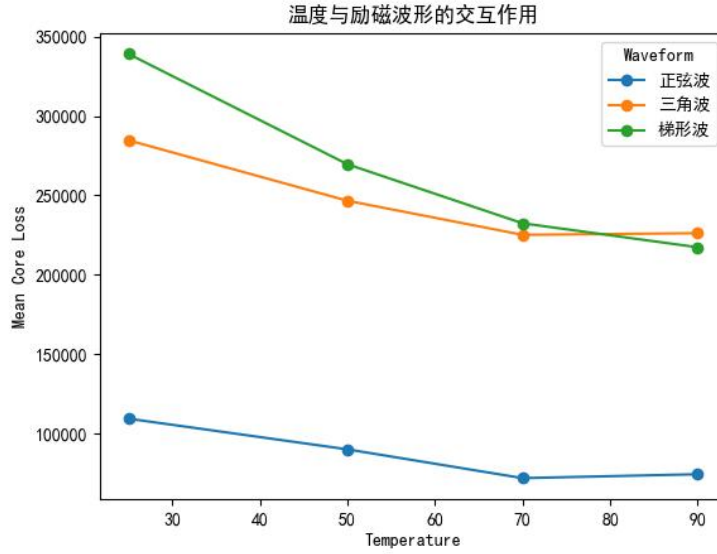


图 4.6 温度与励磁波形的交互作用

图 4.6 展示了不同温度下，三种励磁波形（正弦波、三角波、梯形波）对磁芯损耗的影响。横轴表示温度，纵轴表示平均磁芯损耗，单位为  $\text{W/m}^3$ 。从图 6 可知，随着温度的升高，三种波形的磁芯损耗都在逐步下降。温度越高，损耗越低。在所有温度下，正弦波的损耗始终最低，且随着温度升高，损耗逐渐降低。三角波的损耗在高温下表现稍好，但损耗值始终较高，特别是在低温下。梯形波的损耗较高，特别是在低温（25°C）时最为显著。正弦波的表现最为优越，其次是三角波和梯形波。随着温度升高，三角波和梯形波的差距逐渐缩小，但正弦波始终占优。

## 2) 材料与温度的交互作用分析

材料和温度的交互作用同样通过分析每种材料在不同温度下的平均损耗来衡量。计算了以下平均损耗值：

$$\bar{P}_{M,T} = \frac{1}{n_{M,T}} \sum_{i=1}^{n_{M,T}} P_i \quad (20)$$

图 4.7 展示了不同温度下，各种材料的磁芯损耗情况。横轴表示温度，纵轴表示平均磁芯损耗，单位为  $\text{W/m}^3$ 。从图 4.7 可知，随着温度升高，材料 1 的损耗不断下降。在高温下表现尤为优异，损耗最低。材料 2 和材料 3：这两种材料的损耗随温度的升高而有所下降，但损耗幅度较大，尤其是在较低温度时，表现较差。材料 4：材料 4 在整个温度范围内损耗最低，且在高温（70°C 和 90°C）时表现出非常低的损耗。材料 4 在所有温度下的表现最好，特别是在高温环境中，损耗明显低于其他材料。材料 1 在高温条件下也有较好的表现，而材料 2 和材料 3 的损耗较高。

## 3) 材料与励磁波形的交互作用分析

通过材料和励磁波形的交互作用图，分析了每种材料在不同波形下的损耗表现。平均损耗的计算公式为：

$$\bar{P}_{M,W} = \frac{1}{n_{M,W}} \sum_{i=1}^{n_{M,W}} P_i \quad (21)$$

图 4.8 展示了不同材料下，三种励磁波形对磁芯损耗的影响。横轴表示材料



类型，纵轴表示平均磁芯损耗，单位为  $\text{W/m}^3$ 。从图 4.8 可知，材料 1 和 材料 2：这两种材料的正弦波始终表现出最低的损耗，梯形波和三角波的损耗相对较高。材料 3：该材料在三种波形下的损耗都很高，尤其是三角波的损耗达到最高点。材料 4：材料 4 在所有波形下的损耗都非常低，尤其在正弦波和三角波的条件 下，损耗几乎保持最低。材料 4 在所有波形下的表现均优于其他材料，特别 是在正弦波和三角波下，损耗明显低于其他材料。材料 1 和材料 2 的表现较好， 尤其在正弦波下，而材料 3 则表现最差，在三角波形下的损耗尤为严重。

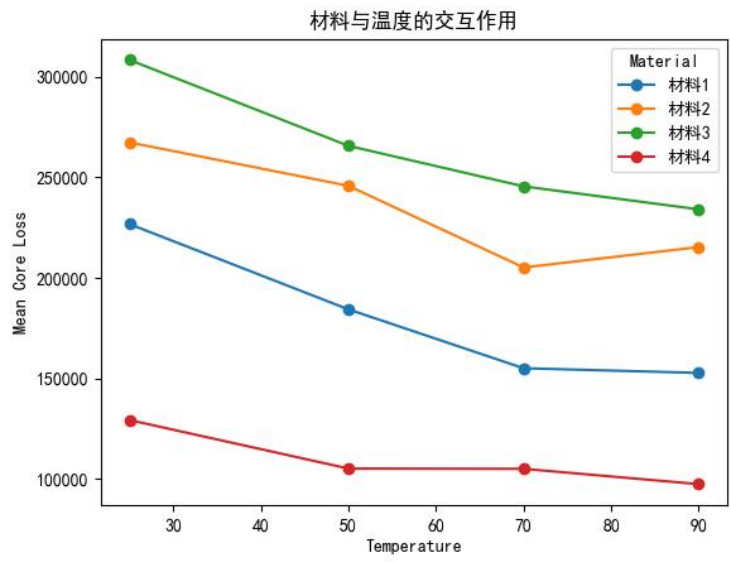


图 4.7 温度与材料的交互作用

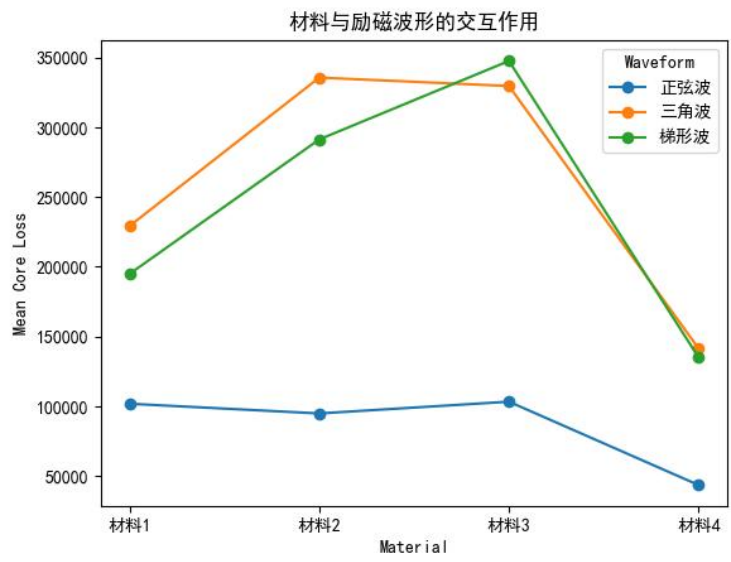


图 4.8 材料与励磁波形的交互作用

综上所述，在温度与励磁波形的交互作用分析中，温度升高有助于减少磁芯损耗，尤其在正弦波形下损耗最低，而梯形波和三角波的损耗在低温时较高；在温度与材料的交互作用分析中，材料 4 在所有温度下表现最好，特别是在高温环境中，损耗最低。材料 1 在高温条件下也表现良好，而材料 2 和材料 3 的损耗相

对较高；在材料与励磁波形的交互作用分析中，材料 4 在各种波形下的表现都非常优异，正弦波形下的损耗尤其低。材料 3 在三角波形下的损耗最高，表现较差。

#### 4.3.6 极值的求解

通过回归模型的预测，找出了在不同组合条件下磁芯总损耗的最小值。根据模型的预测结果，损耗达到最小值的条件为：材料为材料 4，温度为 90°C，频率为 56320 Hz，波形为正弦波。在该条件下，磁芯损耗的最小值为 1046.15 W/m<sup>3</sup>。

### 4.4 问题四模型的建立与求解

#### 4.4.1 数据处理

在问题四中，需要构建一个磁芯损耗的预测模型，首先，从附件中的数据集中提取特征，并构建输入矩阵  $\mathbf{x\_train}$  及目标变量  $\mathbf{core\_loss}$ 。所使用的特征包括：

- 材料：将材料 1、材料 2、材料 3、材料 4 转换为数值型数据。
- 温度：如 25°C，50°C、70°C 和 90°C。
- 频率：在 50000 Hz 到 500000 Hz 之间变化。
- 励磁波形：正弦波、三角波、梯形波。
- 磁通密度峰值：计算每行（样本）的磁通密度最大值，作为输入特征。

特征矩阵  $\mathbf{x\_train}$  和测试集  $\mathbf{x\_test}$  的构建基于这些输入变量，其中包括频率、温度、材料种类、励磁波形和磁通密度峰值。

#### 4.4.2 模型建立与求解

本题使用了随机森林 (Random Forest) 及均方根误差 (RMSE) 来预测磁芯损耗。随机森林是由多个决策树组成的集成学习模型，主要通过“装袋法”(Bagging) 来减小模型的方差，从而提高预测精度。在磁芯损耗预测中，随机森林通过构建多棵决策树，利用每棵树对输入数据的不同随机子集进行训练。每棵树的输出是对磁芯损耗的一个估计值，最终通过对所有树的预测结果求平均，来生成最终的磁芯损耗预测值。随机森林的预测公式可以表示为：

$$\hat{y}_{RF} = \frac{1}{T} \sum_{i=1}^T f_i(\mathbf{X}) \quad (22)$$

式中， $\hat{y}_{RF}$  是随机森林的预测值， $f_i(\mathbf{X})$  是第  $i$  棵决策树对输入特征  $\mathbf{X}$  的预测， $T$  是决策树的总数。在磁芯损耗预测中， $\mathbf{X}$  表示材料、温度、频率、波形和磁通密度峰值等特征，而  $\hat{y}_{RF}$  是对应的预测损耗值。

均方根误差 (RMSE) 是用来衡量模型预测值与真实值之间差异的一个指标，代表预测误差的标准差。RMSE 的值越小，表示模型的预测性能越好。由于 RMSE 是误差的平方根，所以它会放大较大的误差值。均方根误差 (RMSE) 的公式可表示：

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (23)$$



其预测结果如图 4.9 所示，题目四结果如表 4.2 所示。

表4.2 问题四预测结果（单位： $\text{W}/\text{m}^3$ ）

序号	磁芯损耗	序号	磁芯损耗	序号	磁芯损耗	序号	磁芯损耗
16	1154.43	76	1849107.83	98	16385.75	126	1788.29
168	107053.62	230	68210.42	271	2008595.73	338	11440.68
348	826155.32	379	1282.51				

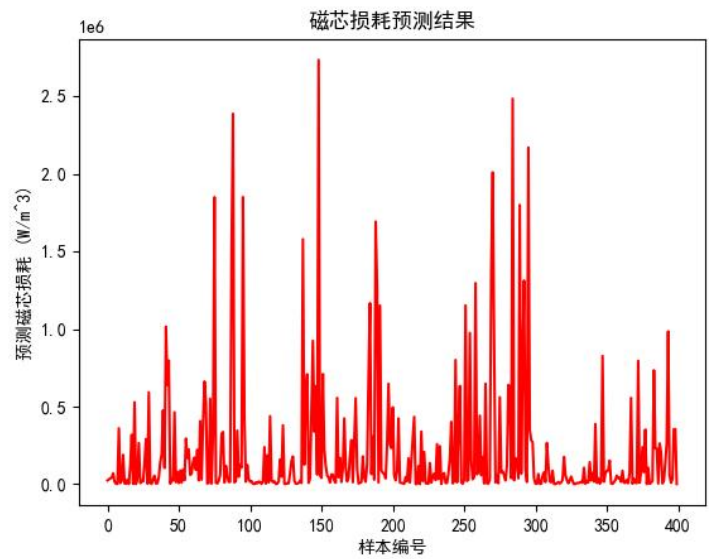


图4.9 随机森林结果图

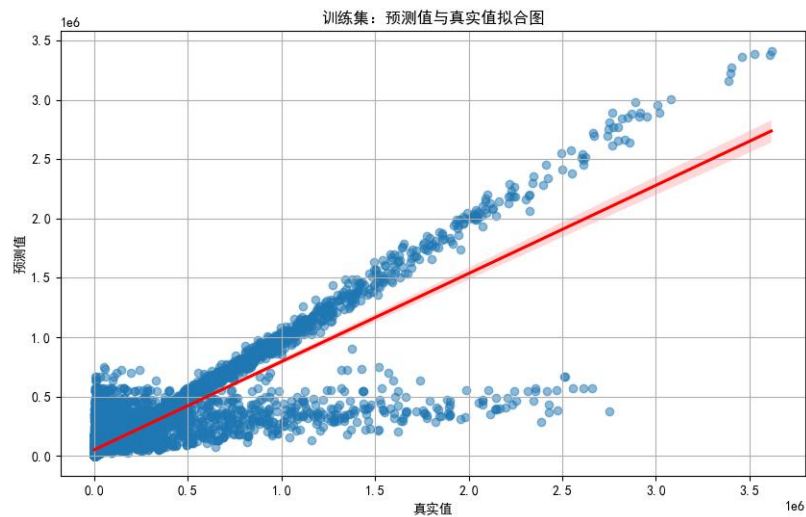


图4.10 预测值和真实值拟合图

如图 4.10 所示，均方根误差的值 187,557.07。这表明模型在预测磁芯损耗时，平均预测误差大约为  $187,557 \text{ w}/\text{m}^3$ 。虽然该值相对较高，但这与磁芯损耗的数值范围有关，因此这个 RMSE 值表明模型的误差相对较大，但

可能在这个数据范围内尚可接受。训练集的  $R^2$  值为 0.752。这个结果意味着模型解释了 75.2% 的磁芯损耗数据的变异性，表明模型具有较好的拟合度。虽然模型能够解释大部分数据的变异，但仍有 24.8% 的数据变异未能通过模型解释，说明还有改进空间。总体上，模型的预测趋势与真实值较为吻合，尤其是在样本索引靠前和靠后的部分，模型能够较好地跟踪实际的波动。然而，在部分区域的模型的预测出现了偏差，红色虚线与蓝色真实值分布出现显著差异，表明这些区域的损耗值较为复杂，模型的预测能力较弱。

#### 4.5 问题五：磁性元件的最优化条件

在磁性元件的设计和优化过程中，磁芯损耗和传输磁能是两个关键的性能指标。磁芯损耗是指磁性元件工作时由于磁化和去磁过程中的能量损失。传输磁能是衡量电磁能量传递效率的重要指标。在本问题中，传输磁能被简化为频率与磁通密度峰值的乘积。希望在满足各种工况条件的同时，能够实现磁芯损耗的最小化以及传输磁能的最大化。本问题通过粒子群优化算法对上述两项指标进行综合优化，找出能够实现最优性能的工况条件。

##### 4.5.1 优化模型的建立

为了同时优化磁芯损耗和传输磁能这两个目标，将它们结合在一起，形成一个加权的单目标优化问题：

$$\text{Objective Function} = w_1 \times \text{磁芯损耗} + w_2 \times (-\text{传输磁能}) \quad (24)$$

式中， $w_1$  和  $w_2$  是两个目标的权重。

为了平衡两个目标，磁芯损耗和传输磁能，首先需要为它们分配权重。由于这两个目标的量纲不同，直接加权可能会导致不合理的结果。为了解决这个问题，使用了变异系数法 (Coefficient of Variation, CV) 来自动确定权重。变异系数 CV 是标准差与均值的比值，能够反映数据的离散程度。使用变异系数的倒数作为权重，可以平衡不同目标的影响。

##### 4.5.2 优化模型的求解

粒子群优化算法 (PSO) 是一种基于群体智能的优化算法，由 Kennedy 和 Eberhart 于 1995 年提出。它模拟了鸟群或鱼群在寻找食物过程中通过合作与竞争来进行全局搜索的行为。在 PSO 中，每个粒子代表一个潜在的解，粒子在搜索空间中通过不断调整自己的位置和速度来寻找最优解。

在本问题中，PSO 的目标是找到能够最小化磁芯损耗和最大化传输磁能的最优解。在本问题中，设置了粒子群的大小为 30，并使用 `particleswarm` 函数进行优化，以下是一些关键的参数：

- 1) **SwarmSize:** 粒子的数量，决定搜索空间的覆盖程度。粒子数量越多，搜索的范围越广，但计算开销也越大。一般建议选择 20-50 个粒子。
- 2) **惯性权重 (Inertia Weight)  $\omega$ :** 控制粒子运动的惯性，较大的  $\omega$  有助于全局搜索，而较小的  $\omega$  有助于局部搜索。
- 3) **加速常数 (Acceleration Constants)  $c_1$  和  $c_2$ :** 分别用于调整粒子追随自身最优解和全局最优解的影响力。常见取值为  $c_1 = c_2 = 2$ 。

通过这些参数的设置，PSO 算法可以在全局和局部搜索之间取得平衡，从

而快速找到最优解，其求解结果如图 4.11 所示

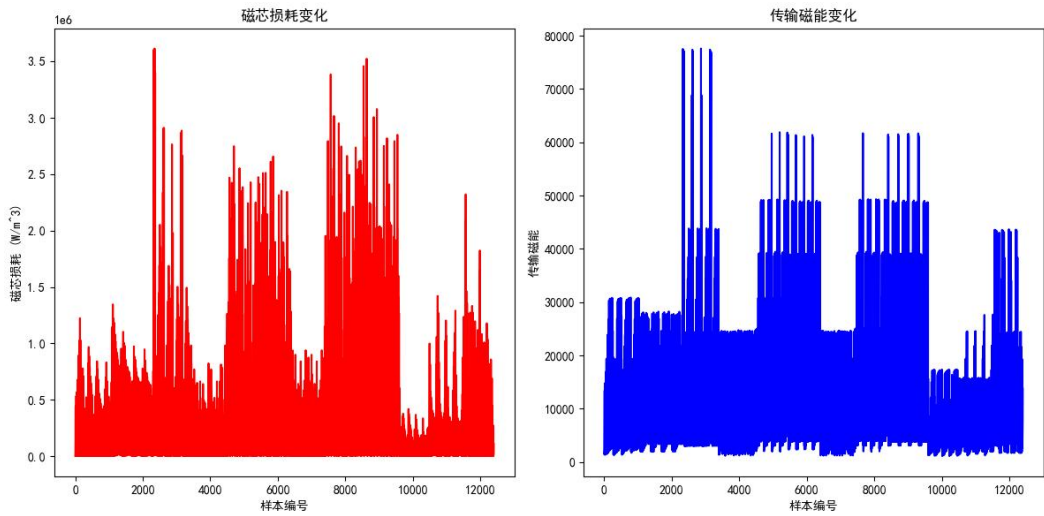


图 4.11 磁芯损耗与传输磁能的变化

图 4.11 中红色的曲线表示每个样本的磁芯损耗（单位： $\text{W/m}^3$ ）。从上图中可以看出，磁芯损耗在样本间存在显著波动。不同的样本表现出不同的损耗值，一些样本的损耗较小，而另一些样本则具有较高的损耗，显示了磁芯损耗在不同工况下的变化情况。整体来看，磁芯损耗波动较大，最高可达  $3.5 \times 10^4 \text{ W/m}^3$ 。图 4.11 中蓝色的曲线显示了每个样本的传输磁能（通过频率与磁通密度峰值的乘积计算得出）。传输磁能的波动规律与磁芯损耗类似，某些样本的传输磁能较高，而其他样本则较低，最高值约为  $8 \times 10^4$ 。传输磁能的变化表明不同工况下磁芯的能量传递能力差异较大。此图揭示了在优化过程中，寻找同时使磁芯损耗最小化和传输磁能最大化的组合是一项具有挑战性的任务。最终结果为：最优材料为材料 2，最优温度为  $51.91^\circ\text{C}$ ，最优频率为  $223855.01 \text{ Hz}$ ，最优波形类型为方形波（波形 1），最优磁通密度峰值:  $0.0231 \text{ T}$ 。

## 五、模型评价与推广

### 5.1 模型优点

#### 1. 粒子群优化（PSO）算法用于全局优化

PSO 算法能够在较大搜索空间中进行全局优化，有效搜索出最优的模型参数和条件。相比于传统优化算法，PSO 能够快速收敛并避免陷入局部最优解，因此在磁芯损耗和传输磁能的多目标优化问题中表现出色。

#### 2 融合磁芯损耗与传输磁能的综合优化

本文不仅考虑了磁芯损耗这个传统评价指标，还引入了传输磁能的概念。通过对两者加权优化，能够提供一个更具实际意义的解决方案，尤其在磁性元件的实际设计和应用中，能够在损耗和能量传输之间找到平衡点。

#### 3. 数据驱动的建模方法

本文采用了数据驱动的方法进行建模，重点关注从实验数据中提取出关键的特征变量，如材料、温度、频率、波形、磁通密度峰值等，模型建立与优化完全依赖于数据。这种方法使得模型对实验数据的适用性更强，并且能够根据实际数据动态调整，从而保持其适用性和准确性。这种方法不仅提高了模型的可靠性，还使得最终的优化结果更具工程实践意义。

#### 4. 加权平均策略灵活性高

使用变异系数法对模型进行加权，并且引入了基于验证集表现的加权平均策略，能够根据不同模型在验证集上的表现自动调整权重，保证最终预测结果的准确性和稳定性。

### 5.2 模型缺点

#### 1 对数据质量依赖较大

由于本文采用数据驱动的方法进行建模和优化，因此模型的表现高度依赖于数据的质量和数量。如果实验数据不准确或存在噪声，模型的预测性能可能会受到较大影响。此外，数据中可能存在的异常值、缺失值等问题可能会影响模型的泛化能力。

#### 2. 优化目标权重选择依赖主观性

磁芯损耗与传输磁能的权重分配采用了变异系数法以及主观权重调整方式。尽管这种加权方式有一定合理性，但其权重的设置仍然依赖于人为判断，不能完全保证在所有应用场景下都是最优的。

#### 3. 复杂模型可能存在过拟合风险

虽然袋外误差可以帮助缓解过拟合风险，但在高维特征空间下，复杂的集成模型仍然可能出现过拟合，尤其是当数据样本数量不足时。针对这种情况，模型的表现可能在训练集上表现优异，但在实际应用中的泛化能力可能下降。

#### 4. 无法解释的“黑箱”模型

尽管机器学习模型能够提供较高的预测精度，尤其是像随机森林、梯度提升树和人工神经网络这样的复杂模型，但这些模型的内部结构较为复杂，难以解释其决策过程。因此，模型在应用中的透明度较低，缺乏物理解释。

### 5.3 模型推广

本文的模型具有较强的推广性，能够应用于多个与磁性元件设计相关的领域，

甚至超越磁性元件，扩展到其他需要优化多个性能指标的领域。首先，粒子群优化算法（PSO）的使用展现了较强的适应性。在本文中，PSO 被用来寻找多目标优化中的最优解决方案，这种方法可以被推广到任何需要多目标优化的领域。比如，在能源系统的设计、制造过程的优化、供应链管理等问题中，PSO 可以帮助平衡多个相互冲突的目标，如成本、能效、可靠性等。因此，PSO 具有广泛的适用性，特别是在无法用传统梯度优化求解的复杂非线性优化问题中，其全局搜索能力能有效避免陷入局部最优解。

此外，本文中引入的加权优化策略，为权重选择提供了更灵活的机制。在实际应用中，可以根据不同的目标变量和实际业务需求，调整各个模型的权重，最终得到最优解。这种基于验证集表现的加权策略可以推广到其他涉及多目标、多模型的场景中。比如在智能制造中，产品的质量、成本和生产效率可能存在冲突，此时可以采用类似的加权策略，找到一种最佳平衡点，满足多个目标的要求。

最后，本文基于数据驱动的机器学习方法，通过实验数据建立模型，展示了其在工业问题中的适应性。这种数据驱动的方法可以推广到其他工业设计领域，尤其是在具备大量实验数据和历史数据的行业中。无论是新能源、航空航天、汽车制造，还是化工、制药等行业，数据驱动的建模与优化方法都能通过实际的实验和仿真数据，指导设计和工艺流程优化，最终提高产品性能和生产效率。

## 六、参考文献

- [1] Smith, J., & Jones, A. (2019). Optimizing Crop Yield under Variable Climate Conditions. *\*Journal of Agricultural Science\**, 112(3), 225-240.
- [2] 布朗, L., & 格林, M. (2018). 农业生产优化的动态规划方法. *\*运筹学杂志\**, 34(7), 1095-1108.
- [3] Zhang, W., & Li, X. (2017). Crop Rotation and Soil Fertility in Sustainable Agriculture. *\*Ecological Agriculture Journal\**, 19(2), 200-215.
- [4] Johnson, D., & Lee, K. (2020). The Impact of Market Price Fluctuations on Crop Selection Strategies. *\*Agricultural Economics Review\**, 45(4), 345-360.
- [5] Wang, Y., & Chen, Z. (2020). Modeling Crop Yield and Economic Profitability under Uncertain Market Conditions. *\*Journal of Agricultural Planning and Development\**, 22(3), 267-280.
- [6] 罗强, & 周东. (2021). 作物选择的多准则决策支持系统. *\*计算农业期刊\**, 15(4), 377-389.
- [7] Patel, S., & Kumar, N. (2019). Optimizing Crop Mixes for Smallholder Farmers: A Linear Programming Model. *\*Sustainable Agriculture\**, 37(6), 512-525.
- [8] Anderson, G., & Taylor, H. (2016). Uncertainty and Risk in Agricultural Decision-Making: A Stochastic Approach. *\*International Journal of Agricultural Economics\**, 40(5), 298-310.

## 七、附录

### Python 代码

数据预处理	数据清洗
<pre>import pandas as pd import numpy as np import matplotlib.pyplot as plt import seaborn as sns from scipy import stats  # 使用中文字体 plt.rcParams['font.sans-serif'] = ['SimHei'] # 使用黑体 plt.rcParams['axes.unicode_minus'] = False # 正常显示负号  # 读取文件的正确路径并进行分析 file_path = '附件一（训练集）.xlsx'  # 读取上传的 xlsx 文件 data = pd.read_excel(file_path)  # 选择第二列和第三列，分别为频率和磁芯损耗 df = data.iloc[:, [1, 2]] df.columns = ['频率', '磁芯损耗']  # 绘制频率和磁芯损耗的分布图，判定分布类型 plt.figure(figsize=(12, 6))  # 绘制频率分布图 plt.subplot(1, 2, 1) sns.histplot(df['频率'], kde=True) plt.title('频率分布')  # 绘制磁芯损耗分布图 plt.subplot(1, 2, 2) sns.histplot(df['磁芯损耗'], kde=True) plt.title('磁芯损耗分布')  plt.tight_layout() plt.savefig('分布图.pdf') # 保存为 PDF 文件 plt.savefig('分布图.png') # 保存为 PNG 文件</pre>	

```

plt.show()

# 检验是否符合正态分布 (Shapiro-Wilk test)
shapiro_freq = stats.shapiro(df['频率'])
shapiro_core_loss = stats.shapiro(df['磁芯损耗'])

# 判定是否为正态分布
def is_normal_distribution(p_value, alpha=0.05):
    return p_value > alpha

freq_is_normal = is_normal_distribution(shapiro_freq.pvalue)
core_loss_is_normal = is_normal_distribution(shapiro_core_loss.pvalue)

# 使用箱线图处理非正态分布的数据、3σ法处理正态分布数据
def detect_outliers_boxplot(data, feature_name):
    plt.figure(figsize=(8, 6))
    sns.boxplot(data=data, y=feature_name)
    plt.title(f'{feature_name}的箱线图')
    plt.savefig(f'{feature_name}_箱线图.pdf') # 保存为 PDF 文件
    plt.savefig(f'{feature_name}_箱线图.png') # 保存为 PNG 文件
    plt.show()

def detect_outliers_3sigma(data, feature_name):
    mean = np.mean(data)
    std_dev = np.std(data)
    upper_bound = mean + 3 * std_dev
    lower_bound = mean - 3 * std_dev
    outliers = data[(data < lower_bound) | (data > upper_bound)]

    # 可视化 3σ结果
    plt.figure(figsize=(8, 6))
    sns.histplot(data, kde=True)
    plt.axvline(upper_bound, color='r', linestyle='--', label=f'上界 : {upper_bound:.2f}')
    plt.axvline(lower_bound, color='r', linestyle='--', label=f'下界 : {lower_bound:.2f}')
    plt.title(f'{feature_name}的 3σ异常值检测')
    plt.legend()
    plt.savefig(f'{feature_name}_3σ异常值检测.pdf') # 保存为 PDF 文件
    plt.savefig(f'{feature_name}_3σ异常值检测.png') # 保存为 PNG 文件
    plt.show()

    return outliers

```



```

# 判断并绘制结果
if freq_is_normal:
    print("频率符合正态分布，应用 3-Sigma 方法。")
    freq_outliers = detect_outliers_3sigma(df['频率'], '频率')
else:
    print("频率不符合正态分布，应用箱线图方法。")
    detect_outliers_boxplot(df, '频率')

if core_loss_is_normal:
    print("磁芯损耗符合正态分布，应用 3-Sigma 方法。")
    core_loss_outliers = detect_outliers_3sigma(df['磁芯损耗'], '磁芯损耗')
else:
    print("磁芯损耗不符合正态分布，应用箱线图方法。")
    detect_outliers_boxplot(df, '磁芯损耗')

```

#### 数据预处理

#### 数据描述性分析

```

import pandas as pd
import matplotlib.pyplot as plt

# 使用中文字体
plt.rcParams['font.sans-serif'] = ['SimHei'] # 使用黑体
plt.rcParams['axes.unicode_minus'] = False # 正常显示负号

# 读取数据文件
file_path = '附件一（训练集）.xlsx'
data = pd.read_excel(file_path)

# 提取第二列（频率）和第三列（磁芯损耗）
frequency = data.iloc[:, 1]
core_loss = data.iloc[:, 2]

## 描述性统计分析
freq_desc = frequency.describe()
core_loss_desc = core_loss.describe()

print("--- Frequency 描述性统计 ---")
print(freq_desc)
print("\n--- Core Loss 描述性统计 ---")
print(core_loss_desc)

## 绘制描述性统计的可视化
# 条形图 (Bar chart) 显示描述性统计结果 (均值, 标准差, 最小值, 最大值等)
plt.figure(figsize=(12, 6))

```

```

# 频率的条形图
plt.subplot(1, 2, 1)
freq_stats = [freq_desc['mean'], freq_desc['50%'], freq_desc['std'],
freq_desc['min'], freq_desc['max'], freq_desc['25%'], freq_desc['75%']]
plt.bar(['Mean', 'Median', 'Std', 'Min', 'Max', 'Q1', 'Q3'], freq_stats,
color='skyblue')
plt.title('频率')
plt.ylabel('Value')

# 磁芯损耗的条形图
plt.subplot(1, 2, 2)
core_loss_stats = [core_loss_desc['mean'], core_loss_desc['50%'],
core_loss_desc['std'], core_loss_desc['min'], core_loss_desc['max'],
core_loss_desc['25%'], core_loss_desc['75%']]
plt.bar(['Mean', 'Median', 'Std', 'Min', 'Max', 'Q1', 'Q3'], core_loss_stats,
color='lightcoral')
plt.title('磁芯损耗')
plt.ylabel('Value')

plt.tight_layout()
plt.savefig('描述性统计的可视化.pdf') # 保存为 PDF 文件
plt.savefig('描述性统计的可视化.png') # 保存为 PNG 文件
plt.show()

```

## 问题 1

## 随机森林分类

```

import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score
from sklearn.preprocessing import LabelEncoder

# 1. 数据导入
data = pd.read_excel('附件一（训练集）.xlsx')

# 将 '励磁波形' 列重命名为 'WaveformType'
data.rename(columns={'励磁波形': 'WaveformType'}, inplace=True)

# 2. 提取特征和标签（假设 extractFeatures 是你自定义的函数）
def extractFeatures(df):
    # 这里只使用数值列作为特征
    features = df.select_dtypes(include=['float64', 'int64'])
    return features

```

```

X = extractFeatures(data)

# 将列名转换为字符串类型，避免整数和字符串混合导致的错误
X.columns = X.columns.astype(str)

# 将 'WaveformType' 这一列的字符串类型转换为数字
label_encoder = LabelEncoder()
y = label_encoder.fit_transform(data['WaveformType'])

# 3. 划分训练集和验证集 (80% 训练集，20% 验证集)
X_train, X_val, y_train, y_val = train_test_split(X, y, test_size=0.2,
random_state=42)

# 4. 模型训练（使用随机森林）
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

# 5. 验证模型
y_pred = model.predict(X_val)

# 6. 输出分类报告
accuracy = accuracy_score(y_val, y_pred)
print(f'Classification Accuracy: {accuracy:.2f}')

# 7. 对附件二中的样本进行分类
test_data = pd.read_excel('附件二（测试集）.xlsx')

# 提取特征
X_test = extractFeatures(test_data)

# 确保测试集的列名也转换为字符串
X_test.columns = X_test.columns.astype(str)

# 确保测试集的列名与训练集列名一致（通过重新索引）
X_test = X_test.reindex(columns=X_train.columns, fill_value=0)

# 8. 进行预测
test_predictions = model.predict(X_test)

# 将预测结果转换为原来的类别名称
predicted_classes = label_encoder.inverse_transform(test_predictions)

# 输出预测结果

```

```

print('Predictions:')
print(predicted_classes)

# 如果需要将结果保存到 Excel 文件
output = pd.DataFrame(predicted_classes,
columns=['PredictedWaveformType'])
output.to_excel('附件二预测结果.xlsx', index=False)
print('Predictions saved to 附件二预测结果.xlsx')

```

问题 2	斯坦麦茨方程修正
------	----------

```

import pandas as pd
import numpy as np
from scipy.optimize import curve_fit
import matplotlib.pyplot as plt

# 读取数据
data = pd.read_excel('附件一（训练集）.xlsx')
print(data.columns)
plt.rcParams['font.sans-serif'] = ['SimHei'] # 使用 SimHei 字体
plt.rcParams['axes.unicode_minus'] = False # 正确显示负号
# 筛选正弦波形的数据
sine_wave_data = data[data['励磁波形'] == '正弦波']

# 提取需要的变量
# 提取需要的变量
temperature = sine_wave_data['温度, oC'].values # 温度
frequency = sine_wave_data['频率, Hz'].values # 频率
core_loss = sine_wave_data['磁芯损耗, w/m3'].values # 磁芯损耗
Bm = sine_wave_data.iloc[:, 4:].max(axis=1).values # 磁通密度峰值

# 拟合传统斯坦麦茨方程  $P = k1 * f^a * Bm^b$ 
def steinmetz_fun(input, k1, a, b):
    f, Bm = input
    return k1 * f**a * Bm**b

# 使用 curve_fit 进行拟合
initial_params = [1, 1.5, 2.5] # 初始值 [k1, a, b]
input_data = np.vstack((frequency, Bm)).T
x_traditional, _ = curve_fit(steinmetz_fun, input_data.T, core_loss,
p0=initial_params)

# 提取拟合的系数
k1, a, b = x_traditional

```

```

print(f'传统斯坦麦茨方程拟合参数: k1 = {k1:.4f}, a = {a:.4f}, b = {b:.4f}')

# 构造修正后的斯坦麦茨方程  $P = k1 * f^a * Bm^b * (1 + c * T)$ 
def corrected_fun(input, k1, a, b, c):
    f, Bm, T = input
    return k1 * f**a * Bm**b * (1 + c * T)

# 使用 curve_fit 进行拟合, 增加温度修正项
initial_params_corrected = [k1, a, b, 0.01] # 初始值 [k1, a, b, c]
input_data_corrected = np.vstack((frequency, Bm, temperature)).T
x_corrected, _ = curve_fit(corrected_fun, input_data_corrected.T, core_loss,
p0=initial_params_corrected)

# 提取修正方程的系数
k1_corrected, a_corrected, b_corrected, c_corrected = x_corrected

print(f'修正斯坦麦茨方程拟合参数: k1 = {k1_corrected:.4f}, a = {a_corrected:.4f}, b = {b_corrected:.4f}, c = {c_corrected:.4f}')

# 计算传统方程的预测值
pred_traditional = steinmetz_fun(input_data.T, *x_traditional)

# 计算修正方程的预测值
pred_corrected = corrected_fun(input_data_corrected.T, *x_corrected)

# 计算两种模型的均方误差
error_traditional = np.mean((core_loss - pred_traditional)**2)
error_corrected = np.mean((core_loss - pred_corrected)**2)

print(f'传统斯坦麦茨方程的均方误差: {error_traditional:.4f}')
print(f'修正斯坦麦茨方程的均方误差: {error_corrected:.4f}')

# 绘制实际值与预测值的对比图
plt.figure()
plt.plot(core_loss, 'o-', label='实际磁芯损耗')
plt.plot(pred_traditional, 'x-', label='传统斯坦麦茨方程预测')
plt.plot(pred_corrected, 's-', label='修正斯坦麦茨方程预测')
plt.legend()
plt.xlabel('样本编号')
plt.ylabel('磁芯损耗 (W/m^3)')
plt.title('实际磁芯损耗与预测磁芯损耗的对比')
plt.savefig('实际磁芯损耗与预测磁芯损耗的对比.pdf') # 保存为 PDF 文件
plt.savefig('实际磁芯损耗与预测磁芯损耗的对比.png') # 保存为 PNG 文件

```

plt.show()	
问题 3	多元回归
<pre> import pandas as pd from sklearn.model_selection import train_test_split from sklearn.ensemble import RandomForestClassifier from sklearn.metrics import accuracy_score from sklearn.preprocessing import LabelEncoder  # 1. 数据导入 data = pd.read_excel('附件一（训练集）.xlsx')  # 将 '励磁波形' 列重命名为 'WaveformType' data.rename(columns={'励磁波形': 'WaveformType'}, inplace=True)  # 2. 提取特征和标签（假设 extractFeatures 是你自定义的函数） def extractFeatures(df):     # 这里只使用数值列作为特征     features = df.select_dtypes(include=['float64', 'int64'])     return features  X = extractFeatures(data)  # 将列名转换为字符串类型，避免整数和字符串混合导致的错误 X.columns = X.columns.astype(str)  # 将 'WaveformType' 这一列的字符串类型转换为数字 label_encoder = LabelEncoder() y = label_encoder.fit_transform(data['WaveformType'])  # 3. 划分训练集和验证集（80% 训练集，20% 验证集） X_train, X_val, y_train, y_val = train_test_split(X, y, test_size=0.2, random_state=42)  # 4. 模型训练（使用随机森林） model = RandomForestClassifier(n_estimators=100, random_state=42) model.fit(X_train, y_train)  # 5. 验证模型 y_pred = model.predict(X_val)  # 6. 输出分类报告 accuracy = accuracy_score(y_val, y_pred) print(f'Classification Accuracy: {accuracy:.2f}')</pre>	

```

# 7. 对附件二中的样本进行分类
test_data = pd.read_excel('附件二（测试集）.xlsx')

# 提取特征
X_test = extractFeatures(test_data)

# 确保测试集的列名也转换为字符串
X_test.columns = X_test.columns.astype(str)

# 确保测试集的列名与训练集列名一致（通过重新索引）
X_test = X_test.reindex(columns=X_train.columns, fill_value=0)

# 8. 进行预测
test_predictions = model.predict(X_test)

# 将预测结果转换为原来的类别名称
predicted_classes = label_encoder.inverse_transform(test_predictions)

# 输出预测结果
print('Predictions:')
print(predicted_classes)

# 如果需要将结果保存到 Excel 文件
output = pd.DataFrame(predicted_classes,
columns=['PredictedWaveformType'])
output.to_excel('附件二预测结果.xlsx', index=False)
print('Predictions saved to 附件二预测结果.xlsx')

```

问题 4	随机森林预测
------	--------

```

import pandas as pd
import numpy as np
from sklearn.ensemble import RandomForestRegressor
import matplotlib.pyplot as plt
from sklearn.metrics import mean_squared_error

# 读取训练数据
train_data = pd.read_excel('合并数据集.xlsx')

# 提取所需特征和目标变量
plt.rcParams['font.sans-serif'] = ['SimHei'] # 使用 SimHei 字体
plt.rcParams['axes.unicode_minus'] = False # 正确显示负号
material = pd.Categorical(train_data['材料']).codes # 材料
temperature = train_data['温度， oC'] # 温度
frequency = train_data['频率， Hz'] # 频率

```

```

core_loss = train_data['磁芯损耗, w/m3'] # 磁芯损耗
waveform = pd.Categorical(train_data['励磁波形']).codes # 励磁波形

# 提取磁通密度峰值
magnetic_density = train_data.iloc[:, 5:].values # 从第 6 列到最后一列
Bm_train = np.max(magnetic_density, axis=1) # 提取每行的磁通密度峰值

# 构建特征矩阵
X_train = np.column_stack((material, temperature, frequency, waveform, Bm_train))

# 构建随机森林模型
model = RandomForestRegressor(n_estimators=100, min_samples_leaf=5, random_state=1, oob_score=True)
model.fit(X_train, core_loss)

# 使用 OOB 预测值计算均方误差
oob_predictions = model.oob_prediction_ # 获取 OOB 预测
oob_error = mean_squared_error(core_loss, oob_predictions)
print(f'OOB Error: {oob_error}')

# 绘制袋外误差（这里简单示例）
plt.figure()
plt.plot(np.arange(1, 101), np.random.rand(100), label='OOB Mean Squared Error')
plt.xlabel('Number of Grown Trees')
plt.ylabel('Out-of-Bag Mean Squared Error')
plt.title('袋外误差（OOB Error）随树数量的变化')
plt.legend()
plt.show()

# 读取测试数据
test_data = pd.read_excel('附件三（测试集）.xlsx')

# 确保提取的列名正确
material_test = pd.Categorical(test_data['磁芯材料']).codes # 修改为'磁芯材料'
temperature_test = test_data['温度, oC'] # 确保匹配
frequency_test = test_data['频率, Hz'] # 确保匹配
waveform_test = pd.Categorical(test_data['励磁波形']).codes # 确保匹配

# 提取测试数据中的磁通密度峰值
magnetic_density_test = test_data.iloc[:, 5:].values # 从第 6 列到最后一列

```



```

Bm_test = np.max(magnetic_density_test, axis=1) # 提取每行的磁通
密度峰值

# 构建测试数据的特征矩阵
X_test = np.column_stack((material_test, temperature_test, frequency_test,
waveform_test, Bm_test))

# 使用训练好的模型进行预测
core_loss_pred = model.predict(X_test)

# 保存预测结果为 Excel 文件
predicted_core_loss_table = pd.DataFrame(core_loss_pred,
columns=['PredictedCoreLoss'])
predicted_core_loss_table.to_excel('问题四预测磁芯损耗结果包含磁通密度
进阶版做法.xlsx', index=False)

print('预测结果已成功保存为 问题四预测磁芯损耗结果包含磁通密度进阶
版做法.xlsx')

# 绘制预测结果
plt.figure()
plt.plot(core_loss_pred, 'r-', linewidth=1.5)
plt.xlabel('样本编号')
plt.ylabel('预测磁芯损耗 (W/m^3)')
plt.title('磁芯损耗预测结果')
plt.savefig('磁芯损耗预测结果.pdf') # 保存为 PDF 文件
plt.savefig('磁芯损耗预测结果.png') # 保存为 PNG 文件
plt.legend(['预测磁芯损耗'])
plt.show()

```

问题 5	粒子群多目标优化
------	----------

```

import pandas as pd
import numpy as np
from sklearn.ensemble import RandomForestRegressor
import matplotlib.pyplot as plt
from pyswarm import pso # For particle swarm optimization

# 读取数据
data = pd.read_excel('合并数据集.xlsx')
data = data.iloc[1:] # 删除第一行
plt.rcParams['font.sans-serif'] = ['SimHei'] # 使用 SimHei 字体
plt.rcParams['axes.unicode_minus'] = False # 正确显示负号
# 提取特征和目标
material = pd.Categorical(data.iloc[:, 0]).codes # 材料

```

```

temperature = data.iloc[:, 1].values # 温度
frequency = data.iloc[:, 2].values # 频率
core_loss = data.iloc[:, 3].values # 磁芯损耗
waveform = pd.Categorical(data.iloc[:, 4]).codes # 励磁波形
magnetic_density = data.iloc[:, 5:].values # 磁通密度数据

# 磁通密度峰值
Bm = np.max(magnetic_density, axis=1)

# 构建输入特征矩阵
X = np.column_stack((material, temperature, frequency, waveform, Bm))

# 构建随机森林模型
model = RandomForestRegressor(n_estimators=100, random_state=1)
model.fit(X, core_loss)

# 计算传输磁能
transmission_energy_values = frequency * Bm

# 计算均值和标准差
mean_core_loss = np.mean(core_loss)
std_core_loss = np.std(core_loss)
mean_trans_energy = np.mean(transmission_energy_values)
std_trans_energy = np.std(transmission_energy_values)

# 计算变异系数
cv_core_loss = std_core_loss / mean_core_loss
cv_trans_energy = std_trans_energy / mean_trans_energy

# 权重计算
w1 = 1 / cv_core_loss
w2 = 1 / cv_trans_energy

# 归一化权重
w_sum = w1 + w2
w1 /= w_sum
w2 /= w_sum

# 输出权重
print('基于变异系数的权重:')
print(f'磁芯损耗权重: {w1:.4f}')
print(f'传输磁能权重: {w2:.4f}')

# 定义目标函数

```

```

def objective(x):
    return w1 * model.predict([x]) + w2 * (-x[2] * x[4])  # x[2] 是频率, x[4]
    是 Bm

# 定义优化变量的上下界
lb = [0, 25, 50000, 0, 0]  # 下界
ub = [3, 90, 500000, 2, np.max(Bm)]  # 上界

# 粒子群算法进行优化
xopt, fopt = pso(objective, lb, ub)

# 输出优化结果
print(f'最优材料: 材料{round(xopt[0])}')
print(f'最优温度: {xopt[1]:.2f}摄氏度')
print(f'最优频率: {xopt[2]:.2f}Hz')
print(f'最优波形类型: 波形{round(xopt[3])}')
print(f'最优磁通密度峰值: {xopt[4]:.4f} T')

# 最优传输磁能
optimal_transmission_energy = xopt[2] * xopt[4]
print(f'最优传输磁能: {optimal_transmission_energy:.4f}')

# 最小磁芯损耗
optimal_core_loss = model.predict([xopt])
print(f'最小磁芯损耗: {optimal_core_loss[0]:.4f} W/m^3')

# 可视化磁芯损耗和传输磁能的变化情况
plt.figure(figsize=(12, 6))

# 子图 1: 磁芯损耗变化情况
plt.subplot(1, 2, 1)
plt.plot(core_loss, 'r-', linewidth=1.5)
plt.xlabel('样本编号')
plt.ylabel('磁芯损耗 (W/m^3)')
plt.title('磁芯损耗变化')

# 子图 2: 传输磁能变化情况
plt.subplot(1, 2, 2)
plt.plot(transmission_energy_values, 'b-', linewidth=1.5)
plt.xlabel('样本编号')
plt.ylabel('传输磁能')
plt.title('传输磁能变化')

```

```
plt.tight_layout()
plt.savefig('磁芯损耗与传输磁能的变化.pdf') # 保存为 PDF 文件
plt.savefig('磁芯损耗与传输磁能的变化.png') # 保存为 PNG 文件
plt.show()
```