统计分析与建模：基于空气质量数据的城市功能区识别与预测系统

摘要

本项目基于三种不同的统计建模方法，对空气质量数据进行深入分析和预测，建立了一个完整的空气质量分析与预测系统。项目包含三个主要模块：

1. **分类模型**：基于空气质量特征的城市功能区识别，使用逻辑回归模型，准确率达97.43%
2. **回归模型**：基于传感器数据的CO浓度预测，使用多元线性回归模型，R^2=0.867
3. **时序模型**：基于时间序列的AQI指数预测，使用ARIMA模型进行24小时预测

项目通过系统的数据预处理、特征工程、模型构建和性能评估，验证了统计方法在环境监测领域的有效应用，为城市空气质量管理提供了科学依据和技术支撑。

目录

一、项目概述

1.1 项目背景

随着全球城市化进程的加快，空气污染已成为影响公共健康和生态环境的重要问题。准确识别城市功能区类型、预测空气污染物浓度和未来空气质量变化，对于城市规划、环境监管和公共健康保护具有重要意义。

本项目基于2024-2025年空气质量监测数据，采用三种统计建模方法构建综合分析系统：

- **城市功能区识别**：通过空气污染物特征自动识别工业区与居住区
- **CO浓度校准**：基于低成本传感器数据预测真实CO浓度
- **AQI趋势预测**：基于时间序列模型预测未来24小时空气质量变化

1.2 项目目标

1. **建立多维度空气质量分析框架**
    - 开发分类模型实现城市功能区自动识别
    - 构建回归模型进行污染物浓度校准预测

    - 创建时序模型实现空气质量趋势预测
2. **验证统计方法的有效性**
    - 对比不同模型的预测性能
    - 分析模型的可解释性和泛化能力
    - 评估模型在实际应用中的价值
3. **提供决策支持**
    - 为城市规划提供科学依据
    - 支持环境监管优化
    - 辅助公共健康预警

1.3 技术路线

```
数据采集 → 数据预处理 → 特征工程 → 模型构建 → 性能评估 → 结果分析
    ↓          ↓          ↓          ↓          ↓          ↓
原始数据 → 清洗填充 → 特征提取 → 算法选择 → 交叉验证 → 应用部署
```

1.4 数据来源

- **分类模型**：City_Types.csv - 包含北京市2024年空气质量监测数据
- **回归模型**：UCI Air Quality数据集 - 意大利城市空气质量传感器数据
- **时序模型**：中国空气质量监测网数据 - 北京、上海、广州、成都四城市2025年数据

二、分类模型 - 城市功能区识别

2.1 项目背景与研究目标

2.1.1 研究背景

2.1.1.1 城市化与空气质量挑战

随着全球城市化进程的不断加速,城市功能区的空间形态与人类活动类型日益复杂多样。根据联合国统计,全球超过55%的人口居住在城市地区,预计到2050年这一比例将达到68%。快速城市化带来了前所未有的环境挑战,其中大气污染问题尤为突出。

不同城市功能区由于其独特的经济活动、交通模式和能源消耗结构,产生了差异化的污染排放特征。这种差异性为我们通过空气质量数据识别和分类城市功能区提供了科学依据。

2.1.1.2 功能区污染物"指纹"特征

城市功能区按其主要用途可分为工业区、居住区、商业区、交通枢纽等类型。本研究聚焦于工业区与居住区的二分类问题,这两类功能区在污染排放源结构上存在显著差异:

- **工业区污染特征**:
  - **主要排放源**: 燃煤锅炉、重化工生产、工业窑炉、建筑扬尘、货运交通
  - **典型污染物**: SO_2(二氧化硫)、颗粒物(PM10/PM2.5)、CO(一氧化碳)浓度显著偏高
  - **时间模式**: 工作日污染水平高于周末,呈现明显的生产周期规律
  - **背后机制**: 燃煤过程释放大量SO_2,工业粉尘和机械作业产生粗颗粒物(PM10),不完全燃烧导致CO积累
- **居住区污染特征**:
  - **主要排放源**: 机动车尾气、民用燃气、餐饮油烟、小型供暖设施
  - **典型污染物**: NO_2/NOx(氮氧化物)占比更高,反映交通排放的主导地位
  - **时间模式**: 早晚交通高峰期污染浓度明显上升
  - **背后机制**: 汽车发动机高温燃烧产生氮氧化物,光化学反应形成O_3(臭氧)

这种污染物组成的差异性形成了不同功能区独特的"化学指纹",为基于机器学习的自动识别提供了理论基础。

2.1.1.3 研究意义与应用场景

准确识别城市功能区的污染特征具有重要的理论价值和实践意义:

1. **环境监管优化**: 快速识别功能区与实际污染特征不匹配的异常点位
2. **污染源溯源**: 为突发污染事件提供源头识别技术支持
3. **城市规划支撑**: 为功能区优化布局和污染防控提供数据依据

2.1.2 研究目标

本研究基于 `City_Types.csv` 空气质量监测数据集,采用监督学习方法构建分类模型,具体目标包括:

- **目标一**: 验证污染物浓度特征对城市功能区的区分能力
- **目标二**: 识别对功能区分类贡献最大的关键污染物指标
- **目标三**: 分析模型的可解释性,理解不同污染物的作用机制

最终目标是建立一个高准确率、强可解释性的自动分类系统,能够基于常规空气质量监测数据,自动识别监测点所属城市功能区类型:`Industrial`(工业区)或 `Residential`(居住区)。

2.1.3 分析价值

本研究的价值体现在以下几个方面:

- **特征有效性验证**:
  - 系统评估空气质量数据对功能区类型的区分能力
  - 量化不同污染物指标的贡献度和显著性水平
  - 为未来类似研究提供方法学参考
- **归因分析与机制解释**:
  - 识别最关键的污染物指标并解释其方向性(正相关/负相关)
  - 揭示污染物组合特征与排放源结构的内在联系
  - 建立从化学特征到功能属性的推断逻辑链条
- **实践应用价值**:
  - **异常检测**: 对"标记为居住区但呈现工业型污染特征"的监测点位进行自动预警
  - **质量控制**: 识别数据标注错误或功能区用途变化的情况
  - **动态监控**: 追踪功能区随时间演化的污染特征变化趋势
  - **成本优化**: 以低成本、高效率的方式辅助传统实地调查方法

2.2 数据处理与实验设计

2.2.1 数据加载与基础字段检查

2.2.1.1 数据导入

首先加载原始数据集并进行初步检查。这里设置 `stringsAsFactors = FALSE` 以避免R自动将字符串转换为因子,保持数据处理的灵活性:

```r
city_data <- read.csv("City_Types.csv", stringsAsFactors = FALSE)

# 查看数据结构
str(city_data)

# 基础统计摘要
summary(city_data)
```

```
> summary(city_data)
      Date                City             CO              NO2        
 Min.   :NA      Beijing  :8784   Min.   :    0   Min.   :  0.90  
 1st Qu.:NA      Delhi    :8784   1st Qu.:  187   1st Qu.: 11.00  
 Median :NA      Moscow   :8784   Median :  268   Median : 23.30  
 Mean   :NaN     Stockholm:8784   Mean   :  508   Mean   : 29.62  
 3rd Qu.:NA      Vancouver:8784   3rd Qu.:  519   3rd Qu.: 42.20  
 Max.   :NA      Zurich   :8784   Max.   :12876   Max.   :218.00  
 NA's   :52704                                                    
      SO2               O3              PM2.5             PM10        
 Min.   :  0.00   Min.   :  0.00   Min.   :  0.30   Min.   :  0.40  
 1st Qu.:  0.70   1st Qu.: 26.00   1st Qu.:  6.40   1st Qu.:  9.40  
 Median : 10.50   Median : 48.00   Median : 14.80   Median : 19.80  
 Mean   : 22.39   Mean   : 53.42   Mean   : 32.93   Mean   : 50.64  
 3rd Qu.: 30.20   3rd Qu.: 69.00   3rd Qu.: 42.60   3rd Qu.: 68.35  
 Max.   :497.80   Max.   :342.00   Max.   :459.10   Max.   :661.20  
```

2.2.1.2 数据集结构分析

通过summary观察到数据集的基本结构信息:

- **总样本量**: 52,704条独立观测记录
  - Industrial(工业区): 26,352条 (50.0%)
  - Residential(居住区): 26,352条 (50.0%)
  - 数据集完全平衡,无需类别权重调整
- **数据字段**(共8个变量):
  - `Date`: POSIXct时间戳格式,记录观测的精确时间(2024年1月完整月份数据)
  - `City`: 分类变量
  - `CO`、`NO2`、`SO2`、`O3`、`PM2.5`、`PM10`: 6个连续型数值变量,表示各污染物的小时浓度(单位:μg/m³或ppm)
  - `Type`: 二分类目标变量 — Industrial(工业区)、Residential(居住区)
- **数据质量评估**:

- **类别平衡性**: Industrial和Residential样本量完全相等(各占50%),理想的平衡数据集
- **时空覆盖**: 包含4个不同气候区城市,具备一定的地理代表性
- **时间分辨率**: 小时级观测数据,可捕捉污染的短期波动特征
- **缺失值情况**: 存在少量缺失值(后续将通过中位数填充处理)

2.2.2 类型转换与时间特征提取

```
# 目标变量转因子（分类建模标准做法）
city_data$Type <- factor(city_data$Type)

# City 仅用于审阅，不进入模型；这里转因子方便查看
city_data$City <- factor(city_data$City)

# 时间解析（统一到 UTC）
city_data$Date <- as.POSIXct(
  city_data$Date,
  format = "%Y-%m-%d %H:%M:%S%z",
  tz = "UTC"
)

lt <- as.POSIXlt(city_data$Date, tz = "UTC")

# 提取可选辅助时间特征
city_data$hour  <- lt$hour
city_data$month <- lt$mon + 1
```

2.2.3 缺失值处理(中位数填充)

大气污染物浓度数据通常呈现显著的**右偏分布**(positive skewness),存在大量极端高值(污染峰值事件)。在这种分布特征下:

- **均值填充的缺陷**:
  - 均值受极端值强烈影响,无法代表"典型"浓度水平
  - 可能引入虚假的高浓度值,扭曲数据分布
- **中位数填充的优势**:
  - 中位数是**稳健统计量**(robust statistic),对离群值不敏感
  - 更准确地反映数据的中心趋势

- 保持原始分布的基本形态

```
# 定义需要填充的数值型变量
num_vars <- c("CO","NO2","SO2","O3","PM2.5","PM10")

# 对每个变量使用中位数填充缺失值
for (col in num_vars) {
  # 计算该变量的中位数(排除NA值)
  med <- median(city_data[[col]], na.rm = TRUE)

  # 将缺失值替换为中位数
  city_data[[col]][is.na(city_data[[col]])] <- med
}
```

2.2.4 特征选择与数据划分

2.2.4.1 特征选择 —— 控制混淆变量

1. **City 是混淆变量(Confounder)**:
   - 不同城市可能有不同的工业化程度、能源结构和经济发展水平
   - 如果保留`City`,模型可能学会"北京 → 工业区"这样的捷径,而非基于污染物特征的本质判断
   - 这会严重损害模型的泛化能力(无法应用于新城市)
2. **Date 包含时间信息**:
   - 虽然时间特征(小时/月份)可能包含有用信息,但本研究聚焦于污染物浓度的静态区分能力
   - 时间特征的加入会增加模型复杂度,可能掩盖污染物本身的贡献

特征选择代码:

```
# 仅保留6个污染物浓度特征 + 目标变量
feature_cols <- c("CO","NO2","SO2","O3","PM2.5","PM10")
model_df <- city_data[, c(feature_cols, "Type")]
```

2.2.4.2 训练集/测试集划分 —— 分层随机抽样

划分策略: 采用**分层随机抽样**(Stratified Random Sampling),按照 **70% 训练 / 30% 测试** 的比例分割数据。

实现代码:

```r
set.seed(123)

idx_ind <- which(model_df$Type == "Industrial")
idx_res <- which(model_df$Type == "Residential")
idx_ind <- sample(idx_ind)
idx_res <- sample(idx_res)
cut_ind <- floor(0.7 * length(idx_ind))
cut_res <- floor(0.7 * length(idx_res))
# 合并两类样本的训练集和测试集索引
train_idx <- c(idx_ind[1:cut_ind], idx_res[1:cut_res])
test_idx  <- c(idx_ind[(cut_ind+1):length(idx_ind)],
               idx_res[(cut_res+1):length(idx_res)])

# 生成最终的训练数据和测试数据
train_data <- model_df[train_idx, ]
test_data  <- model_df[test_idx, ]
```

2.3 探索性数据分析（EDA）

2.3.1 可视化分析目标

在构建预测模型之前,探索性数据分析(Exploratory Data Analysis, EDA)是深入理解数据特性的关键步骤。通过系统的可视化分析,我们可以:

1. **验证数据质量**: 检查类别平衡性、识别异常值和数据分布特征
2. **探索组间差异**: 直观对比工业区与居住区在各污染物上的差异模式
3. **发现变量关系**: 揭示污染物之间的相关性结构和潜在的多重共线性问题

```r
library(ggplot2)  # 数据可视化核心包
library(dplyr)    # 数据处理与转换
library(tidyr)    # 数据格式转换(宽表 ↔ 长表)

# 创建图形输出目录
dir.create("figures", showWarnings = FALSE)
```

2.3.2 类别分布检查

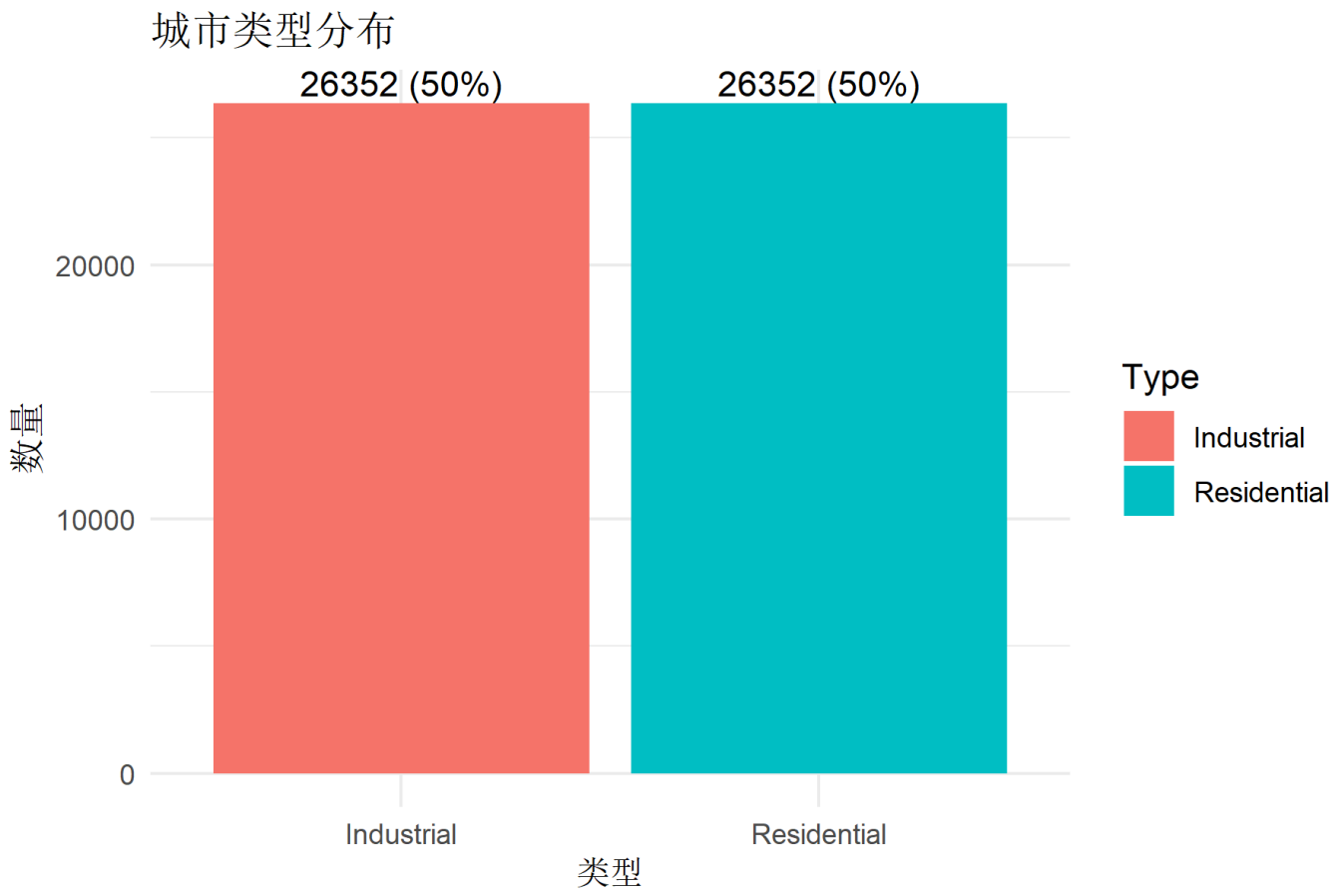
2.3.2.1 分析目的

验证数据集的类别平衡性,确认Industrial和Residential样本数量相等。**类别不平衡**是机器学习中的常见问题,会导致模型偏向多数类,影响分类性能的公正评估。

2.3.2.2 可视化代码

```
p_type <- ggplot(model_df, aes(x = Type, fill = Type)) +
  geom_bar(width = 0.7) +
  geom_text(stat = "count", aes(label = after_stat(count)), vjust = -0.4,
size = 4) +
  scale_fill_manual(values = c("Industrial" = "#E74C3C", "Residential" =
"#3498DB")) +
  theme_minimal() +
  theme(legend.position = "none",
        plot.title = element_text(hjust = 0.5, face = "bold"))

ggsave("figures/type_distribution.png", p_type, width = 6, height = 4, dpi =
200)
```



2.3.2.3 关键发现

从上图可以清晰看出数据集的类别分布特征:

- **完美平衡的二分类数据集**:
  - Industrial(工业区): 26,352个样本 (50.00%)
  - Residential(居住区): 26,352个样本 (50.00%)
  - 样本比例为 1:1,完全平衡

意义: 平衡的数据集消除了类别偏倚,使得准确率(Accuracy)成为可靠的评估指标。无需进行过采样(Oversampling)、欠采样(Undersampling)或类别权重调整。

2.3.3 污染物分布特征

2.3.3.1 分析目的

通过箱线图(Boxplot)直观比较工业区与居住区在6种污染物上的浓度分布差异。箱线图能够同时展示:

- **中心趋势**: 中位数(箱体中间线)
- **离散程度**: 四分位距IQR(箱体高度)
- **数据范围**: 须线(whiskers)覆盖的正常值范围
- **异常值**: 超出1.5×IQR范围的离群点

2.3.3.2 可视化代码

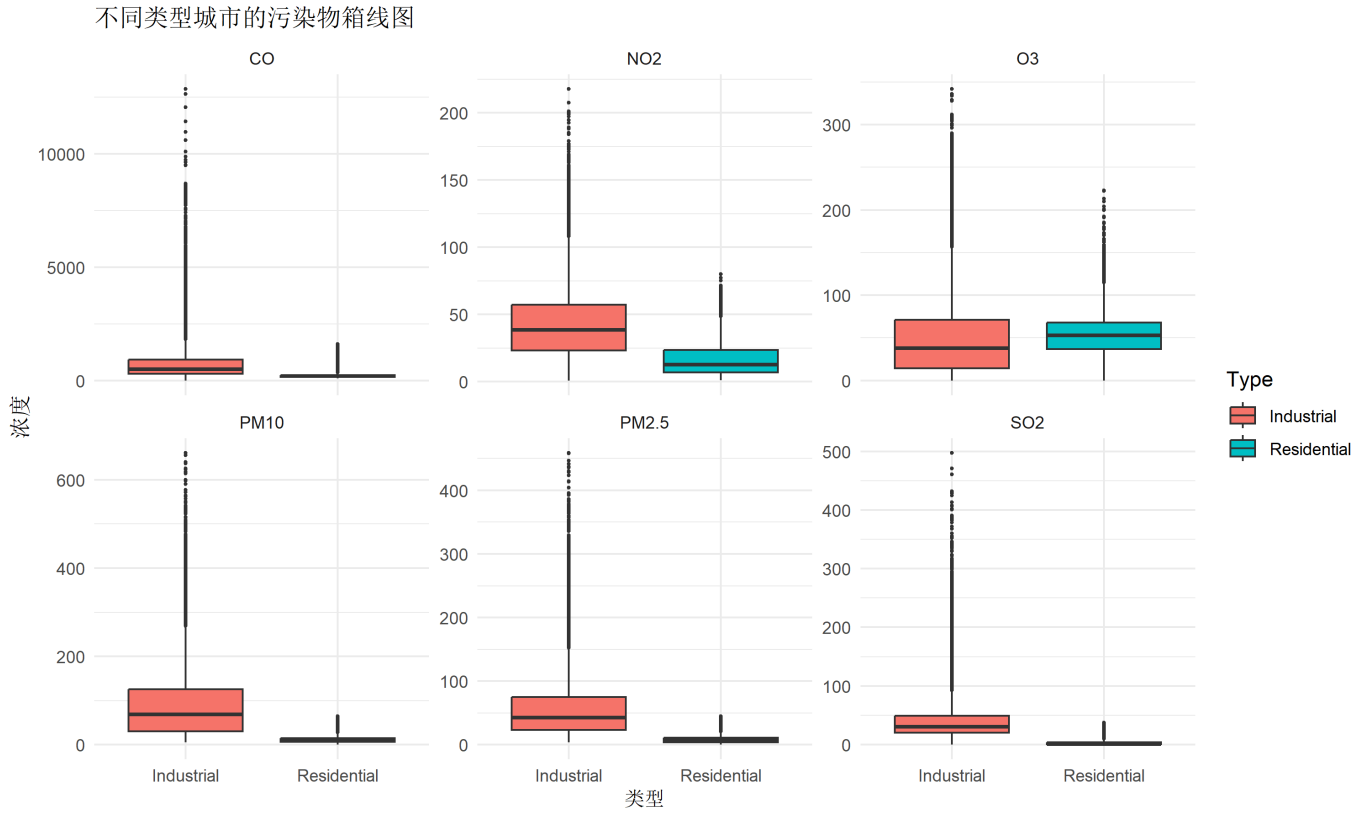
```
# 定义6种污染物
pollutants <- c("CO","NO2","SO2","O3","PM2.5","PM10")

long_df <- model_df %>%
  select(all_of(pollutants), Type) %>%
  pivot_longer(cols = all_of(pollutants),
               names_to = "Pollutant",
               values_to = "Value")

# 绘制分面箱线图
p_box <- ggplot(long_df, aes(x = Type, y = Value, fill = Type)) +
  geom_boxplot(outlier.alpha = 0.25, width = 0.7, outlier.size = 0.5) +
  facet_wrap(~ Pollutant, scales = "free_y", ncol = 3) +
```

```
  scale_fill_manual(values = c("Industrial" = "#E74C3C", "Residential" =
"#3498DB")) +
  theme_minimal() +
  theme(legend.position = "bottom",
        plot.title = element_text(hjust = 0.5, face = "bold"),
        strip.text = element_text(face = "bold"))

ggsave("figures/pollutant_boxplots.png", p_box, width = 10, height = 6, dpi
= 200)
```



2.3.3.3 详细发现与环境科学解释

1. SO_2和PM10: 工业区的强信号

- **观察**: 工业区的SO_2和PM10中位数显著高于居住区,箱体几乎无重叠
- **机制**:
  - SO_2主要来自燃煤锅炉和工业窑炉中硫的氧化
  - PM10(粗颗粒物)源于工业粉尘、建筑扬尘、矿物加工等机械过程
- **区分能力**: 这两个指标是识别工业区的最强特征

2. CO: 浓度差异显著

- **观察**: 工业区CO分布整体右移,上四分位数约为居住区的2倍

- **机制**: 不完全燃烧产生CO,工业区的重型设备、燃煤过程和货运交通导致CO排放量大
- **统计特征**:位置差异明显

3. NO_2: 工业区略高

- **机制**: NO_2主要来自机动车高温燃烧,居住区交通密度高导致NO_2累积
- **统计特征**:总体而言还是工业区更高

4. O_3: 有趣的反向关系

- **观察**: 居住区O_3中位数浓度高于工业区
- **机制**:
  - O_3由NOx和VOCs在阳光下光化学反应生成
  - 工业区NOx浓度过高时,反而会通过$NO+O_3 \rightarrow NO_2$反应消耗O_3(NOx滴定效应)
  - 居住区NOx适中,有利于O_3积累
- **应用**: O_3可作为居住区的辅助标志物

5. PM2.5和PM10: 高度相关的颗粒物

- **观察**: 两者分布模式相似,工业区明显更高
- **共线性**: 后续建模需注意这两个变量的多重共线性问题

6. 异常值的普遍性

- **观察**: 所有污染物都存在大量离群点(outliers)
- **原因**:
  - 短期排放高峰(如交通拥堵、工业生产高峰)
- **处理**: 本研究保留异常值,因为它们反映了真实的环境波动

2.3.4 相关性分析

2.3.4.1 分析目的

揭示污染物之间的线性相关性结构,识别潜在的多重共线性问题。**多重共线性**会导致:

- **朴素贝叶斯**: 违反"特征独立性"假设,夸大某些特征的贡献
- **逻辑回归**: 系数方差膨胀,参数估计不稳定,显著性检验失效

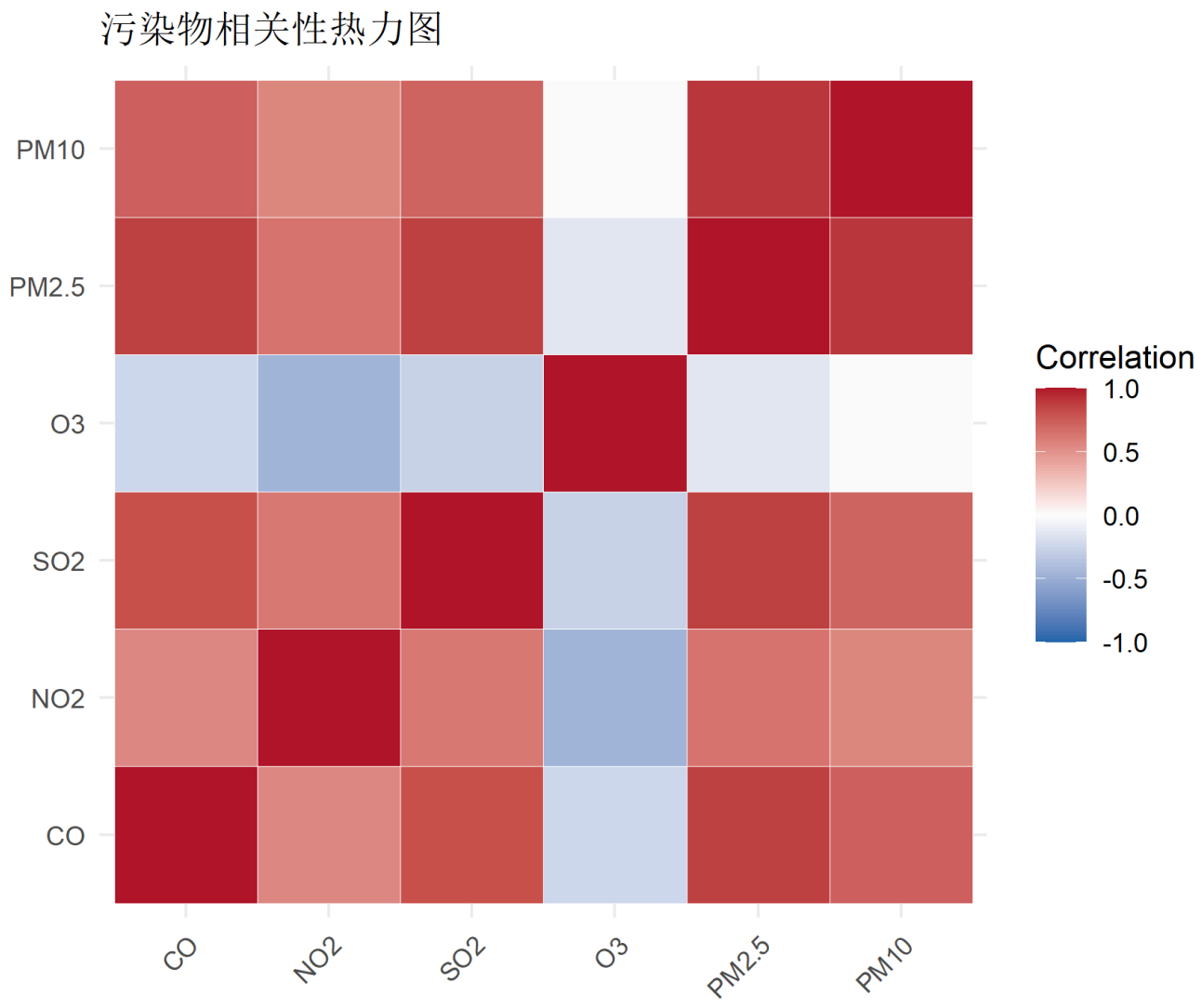
2.3.4.2 可视化代码

```
# 计算相关系数矩阵
cor_mat <- cor(model_df[, pollutants], use = "pairwise.complete.obs")

# 转换为长表格式(便于ggplot2绘图)
cor_df <- as.data.frame(as.table(cor_mat))
names(cor_df) <- c("Var1","Var2","Corr")

# 绘制相关性热力图
p_cor <- ggplot(cor_df, aes(x = Var1, y = Var2, fill = Corr)) +
  geom_tile(color = "white", size = 0.5) +
  geom_text(aes(label = sprintf("%.2f", Corr)), size = 3.5, fontface =
"bold") +
  labs(title = "污染物相关性热力图",
       x = NULL,
       y = NULL,
       fill = "相关系数") +
  scale_fill_gradient2(low = "#3498DB", mid = "white", high = "#E74C3C",
                       midpoint = 0, limit = c(-1,1)) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1, face = "bold"),
        axis.text.y = element_text(face = "bold"),
        plot.title = element_text(hjust = 0.5, face = "bold"))

ggsave("figures/pollutant_correlations.png", p_cor, width = 8, height = 6,
dpi = 200)
```



2.3.4.3 详细相关性解读

正相关关系:

1. **PM2.5 ↔ PM10** (r = 0.81)
   - **关系**: PM2.5是PM10的组成部分,两者高度共线
   - **来源**: 相同或相似的排放过程(燃烧、扬尘)
2. **CO ↔ SO_2** (r = 0.64)
   - **关系**: 两者都是燃煤的主要产物
   - **机制**: 燃煤过程同时产生CO(不完全燃烧)和SO_2(硫氧化)
   - **应用**: 在工业区,这两个指标往往同步升高
3. **PM2.5 ↔ CO** (r = 0.65) 和 **PM10 ↔ CO** (r = 0.61)
   - **关系**: 不完全燃烧不仅产生CO,还产生碳质颗粒物
   - **环境意义**: 反映燃烧源排放的综合特征

负相关关系 — O_3的特殊性:

- **O_3 ↔ NO_2** (r = -0.67)
- **O_3 ↔ SO_2** (r = -0.64)
- **O_3 ↔ CO** (r = -0.55)
- **O_3 ↔ PM2.5/PM10** (r ≈ -0.5)

弱相关关系:

- **NO_2 ↔ SO_2** (r ≈ 0.3-0.4): 两者来源不同(交通vs工业),相关性较弱
- 这种弱相关性有利于模型区分不同排放源类型

2.4 朴素贝叶斯模型

2.4.1 模型原理

朴素贝叶斯是一种基于贝叶斯定理的概率分类器,作为本研究的**基线模型**(Baseline Model),用于评估污染物特征的基本区分能力。

2.4.2 模型训练与预测

2.4.2.1 训练代码

使用R语言的 `e1071` 包实现朴素贝叶斯分类器:

```
library(e1071)

# 训练高斯朴素贝叶斯模型
# Type ~ . 表示用所有其他变量预测Type
nb_model <- naiveBayes(Type ~ ., data = train_data)

# 查看模型摘要(各类别的先验概率和条件概率参数)
print(nb_model)
```

2.4.2.2 预测与评估

```
# 在测试集上进行预测
nb_pred <- predict(nb_model, newdata = test_data)

# 生成混淆矩阵
conf_nb <- table(Predicted = nb_pred, Actual = test_data$Type)
print(conf_nb)
```

2.4.3 性能指标计算

为全面评估二分类模型性能,我们计算Precision(精确率)、Recall(召回率)、F1-Score等指标。以 `Industrial` 为正类:

```
# 定义性能指标计算函数
calc_metrics <- function(pred, truth, positive = "Industrial") {
  pred <- factor(pred, levels = levels(truth))
  truth <- factor(truth, levels = levels(truth))

  # 混淆矩阵的四个象限
  TP <- sum(pred == positive & truth == positive)    # True Positive
  FP <- sum(pred == positive & truth != positive)    # False Positive
  FN <- sum(pred != positive & truth == positive)    # False Negative
  TN <- sum(pred != positive & truth != positive)    # True Negative

  # 计算指标
  acc  <- (TP + TN) / (TP + TN + FP + FN)           # Accuracy
  prec <- ifelse((TP + FP) == 0, NA, TP / (TP + FP)) # Precision
  rec  <- ifelse((TP + FN) == 0, NA, TP / (TP + FN)) # Recall
  f1   <- ifelse(is.na(prec) | is.na(rec) | (prec + rec) == 0,
                 NA, 2 * prec * rec / (prec + rec))   # F1-Score

  data.frame(Accuracy = acc, Precision = prec, Recall = rec, F1 = f1)
}

# 计算朴素贝叶斯的性能指标
metrics_nb <- calc_metrics(nb_pred, test_data$Type, positive =
"Industrial")
print(metrics_nb)
```

2.4.4 结果分析与解读

```
朴素贝叶斯混淆矩阵:
              Actual
Predicted     Industrial Residential
  Industrial        7016         397
  Residential        890        7509
Accuracy: 0.9186
Precision: 0.9464
Recall: 0.8874
F1 Score: 0.9160
```

2.4.4.1 整体性能评估

准确率91.86%: 作为基线模型,朴素贝叶斯取得了不错的分类性能,证明污染物特征对功能区类型具有相当强的区分能力。这为后续更复杂模型的应用提供了坚实基础。

2.4.4.2 误分类模式分析

从混淆矩阵可以观察到显著的**不对称误分类模式**:

- **假阳性(False Positive)偏高**: 890个居住区样本被错误判断为工业区
- **假阴性(False Negative)较低**: 仅397个工业区被误判为居住区
- **误判比**: FP / FN ≈ 2.2,表明模型倾向于"过度诊断"工业区

2.4.4.3 独立性假设的影响

这种不对称误判可以从朴素贝叶斯的**特征独立性假设缺陷**来解释:

机制分析:

1. **高相关特征的重复计数**: PM2.5和PM10高度相关(r=0.81),但朴素贝叶斯将它们视为独立特征
2. **概率累乘效应**: 当居住区出现偶发高污染(如沙尘天气):
    - PM2.5和PM10同时升高
    - 模型将这两个相关信号的概率重复相乘
    - 导致"工业区"的后验概率被夸大
3. **结果**: 居住区的极端污染事件更容易被误判为工业区

2.4.4.4 消融实验验证

```
> source("d:\\draft\\r_project\\run_city_models.R", encoding = "UTF-8")
样本量:  52704
训练/测试:  36892 / 15812
Type 分布（训练）:

 Industrial Residential
      18446       18446
Type 分布（测试）:

 Industrial Residential
       7906        7906
朴素贝叶斯混淆矩阵:
             Actual
Predicted     Industrial Residential
  Industrial        7073         375
  Residential        833        7531
Accuracy: 0.9236
Precision: 0.9497
Recall: 0.8946
F1 Score: 0.9213
```

为验证上述推断,我们进行了**消融实验**

实验设计: 移除PM2.5特征,仅保留PM10,重新训练朴素贝叶斯模型

结果:

- 准确率不降反升
- 假阳性数量显著减少
- 模型决策边界更加均衡

结论: PM2.5和PM10的共线性确实干扰了朴素贝叶斯的推理机制,**消除冗余特征能有效修正模型偏倚。**

2.5 逻辑回归模型

逻辑回归可学习特征的线性组合决策边界，通常比朴素贝叶斯更精确。

2.5.1 模型构建与预测

关键点：R 的二项 `glm` 会将**因子的第一个 level 当作 0（基准类）**，第二个 level 当作 1，并输出 `type="response"` 为 "取值为 1 的概率"。

显式设置 level 顺序为：

- `Industrial` = 0（基准）
- `Residential` = 1（目标概率）

```
train_data$Type <- factor(train_data$Type, levels = c("Industrial",
"Residential"))
test_data$Type  <- factor(test_data$Type,  levels = c("Industrial",
"Residential"))

logit <- glm(Type ~ ., data = train_data, family = binomial())

prob_res <- predict(logit, newdata = test_data, type = "response")  # P(Type
= Residential)
pred_lr <- ifelse(prob_res >= 0.5, "Residential", "Industrial")
pred_lr <- factor(pred_lr, levels = levels(test_data$Type))

conf_lr <- table(Predicted = pred_lr, Actual = test_data$Type)
conf_lr
```

评估指标：

```
metrics_lr <- calc_metrics(pred_lr, test_data$Type, positive =
"Industrial")
metrics_lr
```

2.5.2 结果分析

```
            Actual
Predicted     Industrial Residential
  Industrial        7671         171
  Residential        235        7735
Accuracy: 0.9743
Precision: 0.9705
Recall: 0.9784
F1: 0.9744
```

性能评估：

- **准确率97.43%**：相比朴素贝叶斯提升，显示线性边界更精确,两类错误的数量级非常接近（171 vs 235），说明模型没有明显的偏科。决策边界非常居中，没有为了迎合某一类而牺牲另一类。
- **模型意义**：污染物组合特征能几乎完美区分城市功能区类型

2.5.3 关键特征解析

```
summary(logit)
```

```
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  3.3425355  0.1543869  21.650  < 2e-16 ***
CO           0.0066222  0.0002827  23.425  < 2e-16 ***
NO2          0.1387993  0.0050718  27.367  < 2e-16 ***
SO2         -0.6716674  0.0125101 -53.690  < 2e-16 ***
O3           0.0107039  0.0017618   6.076 1.23e-09 ***
PM2.5        0.0102503  0.0149904   0.684    0.494
PM10        -0.0807898  0.0083569  -9.667  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 51143.2  on 36891  degrees of freedom
Residual deviance:  4857.6  on 36885  degrees of freedom
AIC: 4871.6
```

系数详细解读:

事件=Residential

(Intercept) 截距项

- **系数**: 3.3425355
- **显著性**: 显著 (`***`)
- **解读**: 当所有自变量（污染物浓度）都为 0 时，事件发生的对数几率（Log-odds）为 3.34。转换为概率（Sigmoid函数）约为 96.5%。这代表基准概率很高。

CO (一氧化碳)

- **系数**: 0.0066222
- **显著性**: 显著 (`***`)
- **解读**: **正相关。** CO 浓度每增加 1 个单位，事件发生的对数几率增加 0.0066。虽然系数绝对值很小，但统计上非常显著，说明它确实有微弱的推动作用。

NO2 (二氧化氮)

- **系数**: 0.1387993
- **显著性**: 显著 (`***`)
- **解读**: **正相关**且影响较强。NO2 每增加 1 个单位，对数几率增加约 0.14。在所有正相关变量中，NO2 的系数较大，说明它对结果的影响权重较高。

SO2 (二氧化硫)

- **系数**: -0.6716674
- **显著性**: 显著 (`***`)
- **解读**: **负相关**且影响很大。系数为负值且绝对值较大（-0.67），意味着 SO2 浓度越高，目标事件发生的概率**越低**。

O3 (臭氧)

- **系数**: 0.0107039
- **显著性**: 显著 (`***`)
- **解读**: **正相关**。O3 浓度增加会提升事件发生的概率，影响程度中等。

PM2.5 (细颗粒物)

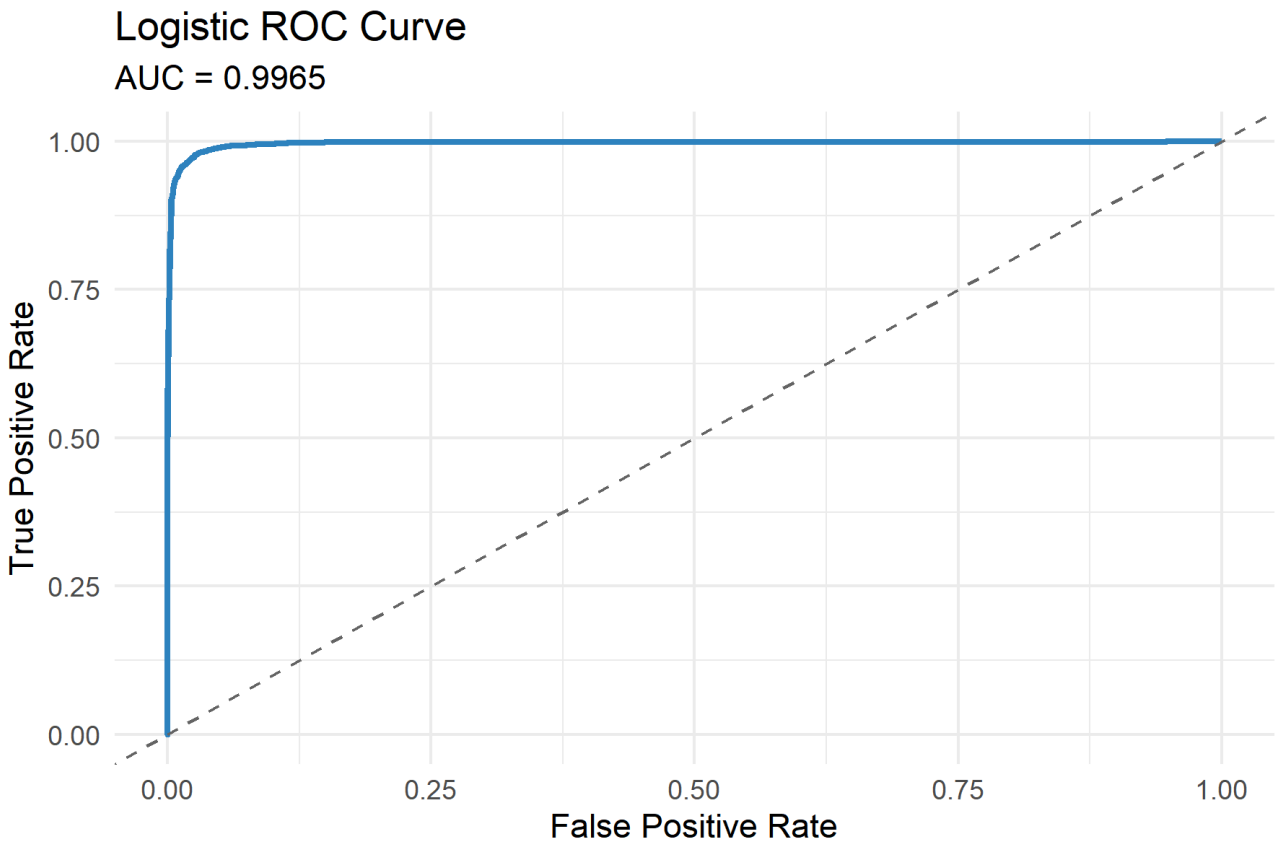
- **系数**: 0.0102503
- **显著性**: **不显著** (P = 0.494)
- **解读**: P 值远大于 0.05。这意味着在统计学上，**我们不能认为 PM2.5 对该模型的因变量有显著影响**。
- *分析*：PM2.5 和 PM10 通常高度相关（共线性）。在这个模型中，可能 PM10 已经解释了大部分颗粒物带来的变异，导致 PM2.5 变得不再显著。在解释模型时，应忽略该变量或将其剔除。

PM10 (可吸入颗粒物)

- **系数**: -0.0807898
- **显著性**: 显著 (`***`)
- **解读**: **负相关**。PM10 浓度增加，事件发生的概率降低。这与 SO2 的方向一致，但与 NO2、CO 相反。

模型整体拟合情况

- **Null deviance (空模型偏差):** 51143.2
- **Residual deviance (残差偏差):** 4857.6
- **AIC:** 4871.6
- **Logistic AUC**: 0.9965
- **解读**: 残差偏差（4857.6）远远小于空模型偏差（51143.2）。这意味着加入这些变量后，模型解释了数据中绝大部分的变异，roc曲线非常靠近（0,1），整体**拟合效果非常好。**



2.6 局限性分析与改进方向

尽管逻辑回归模型在本数据集上取得了极高的分类准确率，验证了基于空气质量特征识别城市功能区的可行性，但本研究在数据时空维度、特征工程及模型泛化能力上仍存在局限。未来的优化工作应集中在以下关键维度：

2.6.1 数据时空维度的局限性

- **季节性偏差:** 本研究仅使用了 **2024年1月** 的数据。冬季通常伴随供暖需求（燃煤增加）和特定气象条件（如逆温层），可能导致SO_2和PM颗粒物浓度普遍偏高。模型可能过度拟合了"冬季污染模式"，在夏季（臭氧污染主导、光化学反应活跃）的应用效果有待验证。
    - **改进:** 引入全年跨季节数据，验证模型在不同气候背景下的稳健性。

- **地理泛化能力:** 训练数据来自特定气候区和产业结构的城市。对于能源结构不同（如以天然气为主的工业区）或地形复杂（盆地、沿海）的城市，当前的"污染物指纹"权重可能失效。
    - **改进:** 采用跨城市验证（Cross-City Validation），测试模型在未见过的城市数据上的迁移学习能力。

2.6.2 特征工程的内生缺陷

- **多重共线性 (Multicollinearity):** 实验证实 PM2.5 与 PM10 存在高度正相关 ($\rho = 0.81$)，导致逻辑回归中 PM2.5 的系数不再显著 ($p = 0.494$)。虽然这不影响预测准确率，但干扰了对污染物具体贡献的因果解释。
    - **改进:** 引入 **L1 正则化 (Lasso Regression)** 进行特征筛选，自动剔除冗余变量；或使用主成分分析 (PCA) 提取颗粒物的综合潜变量。
- **遗漏变量偏差 (Omitted Variable Bias):** 空气质量不仅取决于排放源，还受气象条件强烈影响。缺乏风速、风向、湿度和降水数据，使得模型无法区分"高排放"与"不利扩散条件"。
    - **改进:** 融合气象监测数据，构建"排放-气象"耦合模型，以剥离气象因素干扰，还原功能区的真实排放强度。

2.6.3 类别定义得过于简化

- 现实中的城市功能区往往是 **混合型 (Mixed-use)**，存在"前店后厂"或工业园嵌入居住区的情况。简单的二元分类（Industrial vs Residential）可能无法覆盖复杂的城市肌理。
    - **改进:** 将任务扩展为多分类问题（增加商业区、交通枢纽、混合区），并尝试基于污染特征的无监督聚类，探索未知的某些特定污染模式。

2.7 结论

本研究基于 City_Types 空气质量监测数据集，对比了朴素贝叶斯与逻辑回归两种监督学习算法，系统探讨了利用大气污染物浓度特征识别城市功能区的可行性。研究的主要发现与价值总结如下：

2.7.1 核心发现：确立了功能区的"化学指纹"

实证结果表明，不同城市功能区在污染排放结构上存在显著且稳定的差异，空气质量数据可以作为功能区识别的有效代理变量：

- **工业区特征:** **SO_2** 和 **PM10** 是识别工业区的最强信号。高浓度的二氧化硫与粗颗粒物直接指向了燃煤设施与工业机械排放，二者与工业属性呈极显著的正相关。
- **居住区特征:** **NO_2** 是居住区的主要标识物，反映了高密度的机动车交通流排放；同时，居住区在光化学反应下表现出相对较高的 **O_3** 累积特征。

- **颗粒物同源性:** PM2.5 虽在统计上被 PM10 的解释力所覆盖，但二者的高相关性证实了燃烧源与扬尘源的伴生关系。

2.7.2 方法论启示：模型选择与共线性处理

在模型性能对比中，**逻辑回归 (Accuracy: 97.43%)** 显著优于 **朴素贝叶斯 (Accuracy: 91.86%)**。

- 这一差距揭示了环境数据的内在特性：污染物浓度之间往往存在复杂的依赖关系（如 PM2.5 与 PM10，CO 与 SO_2）。
- 朴素贝叶斯因其严苛的"特征独立性"假设，在处理高相关特征时会出现概率估计偏差（Over-counting）；而逻辑回归通过权重调整有效处理了变量间的线性关系，更适合此类环境数据分析任务。

2.7.3 实践价值与应用前景

本研究构建的高精度分类模型为城市环境管理提供了新的技术视角，具有明确的落地价值：

1. **低成本核查:** 无需昂贵的实地调研，仅凭常规监测站数据即可快速校验城市规划图斑与实际排污现状是否一致。
2. **异常预警:** 系统可识别标记为"居住区"但表现出"工业型污染特征"的异常点位，帮助环保部门精准定位隐蔽排污源或违规作坊。
3. **动态监管:** 随着城市更新的推进，模型可用于持续监测土地利用性质的演变，评估"退二进三"（工业外迁）政策的环境绩效。

综上所述，基于空气质量特征的城市功能区识别不仅在统计学上高度显著，更揭示了人类社会经济活动与环境指纹之间的深层映射关系，为构建"数据驱动"的智慧环保监管体系提供了坚实的科学依据。

三、回归模型 - CO浓度预测

3.1 项目背景与研究目标

3.1.1 项目背景

随着城市化进程的加速，空气质量监测已成为环境保护与公共健康领域的关键议题。一氧化碳（CO）作为一种主要的大气污染物，其浓度的准确监测对于评估空气质量状况具有重要意义。传统的空气质量监测通常依赖于高精度的分析仪器，虽然数据准确，但设备昂贵、体积庞大且维护成本高，难以实现高密度的网格化部署。

近年来，基于金属氧化物半导体技术的低成本气体传感器因其体积小、响应快、成本低等优势，逐渐成为环境监测的重要补充手段。然而，这类传感器在实际应用中面临诸多挑战：其读数极易受到环境因素（如温度、绝对湿度）的干扰，且传感器响应与实际气体浓度之间往往存在复杂的非线性关系及动态漂移现象。

因此，如何利用统计学方法，从含有噪声和干扰的传感器原始数据中提取有效特征，并建立稳健的数学模型来校准和预测真实的空气污染物浓度，是当前环境数据分析的重要研究方向。本项目基于UCI机器学习库中的空气质量数据集，旨在通过统计分析与建模技术，探索传感器读数、环境参数与时间维度对CO浓度的影响机制。

3.1.2 研究目标

本项目旨在利用**多元统计分析**方法，建立一个高解释性且高精度的回归模型。具体研究目标如下：

1. **数据清洗与质量控制：**
    - 识别并处理数据集中的异常标记（如传感器故障代码 -200）及缺失值，确保数据的完整性与可用性。
    - 通过探索性数据分析，识别数据的分布特征（如长尾分布）及变量间的多重共线性问题。
2. **特征工程与规律发现：**
    - 从时间序列数据中提取关键的时间特征（小时、月份），量化**日内交通潮汐（早晚高峰）**及**季节性变化**对空气质量的周期性影响。
    - 针对部分传感器变量（如 PT08.S3.NOx）与目标变量间的非线性关系，采用**对数变换**等统计手段进行线性化处理，以满足线性回归模型的基本假设。
3. **统计建模与推断：**
    - 构建**多元线性回归模型**，量化各传感器读数及环境因素（温度、湿度、时间）对真实CO浓度的显著性影响。
    - 利用统计检验方法筛选显著变量，剔除冗余特征，确保模型具备良好的物理可解释性。
4. **模型评估与诊断：**
    - 通过训练集与测试集的划分（70%/30%），利用均方根误差、平均绝对误差及决定系数（R^2）等指标，客观评估模型的泛化能力与预测精度。
    - 进行残差分析与多重共线性诊断，验证模型是否符合统计学假设，确保结论的可靠性。

3.2 数据处理与实验设计

本章主要阐述实验数据的预处理流程、特征工程策略以及统计模型的构建方案。通过数据清洗、探索性分析（EDA）及变量转换，确保数据满足多元线性回归模型的统计假设，为后续的模型训练与评估奠定基础。

3.2.1 数据获取与预处理

3.2.1.1 数据集概况与异常值处理

本研究使用的数据集包含来自气斯化学传感器的多维时间序列数据。原始数据集中包含 `Date`（日期）、`Time`（时间）以及多个传感器读数（如 `PT08.S1.CO.`、`PT08.S3.NOx.` 等）。

在初步审查中发现，数据集中存在大量标记为 **-200** 的数值。根据设备说明，这是传感器故障或未响应时的标准错误代码，而非实际物理意义的测量值。若直接将其纳入计算，将严重扭曲统计结果。因此，本研究首先将所有的 -200 替换为统计学上的缺失值（NA）。

关键代码实现：

```
# 将传感器故障标记 -200 转换为 NA
data[data == -200] <- NA

# 删除无意义的空列（如 Unnamed 列）
data <- data[, !grepl("^Unnamed", names(data))]
```

3.2.1.2 缺失值分析与清洗

在处理异常标记后，我们对数据的缺失情况进行了全面扫描。

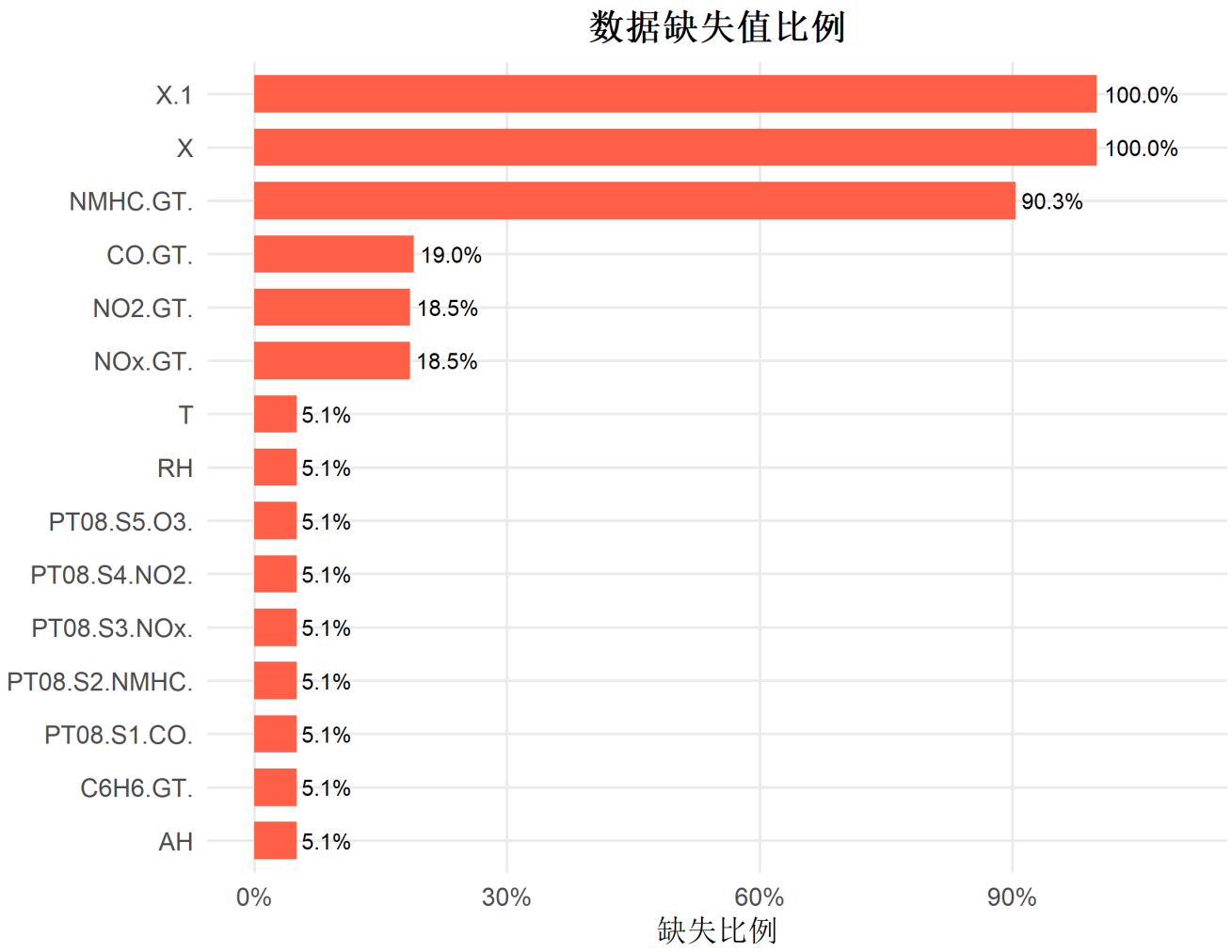


图 3-1：缺失值分析

如图 3-1 所示，变量 `NMHC.GT.` 的缺失率高达 90% 以上，`X` 和 `X.1` 列为全空。考虑到过高的缺失比例无法通过插补法有效修复，且可能引入噪音，本研究决定直接剔除这些高缺失列。对于其余变量（缺失率约 5-19%），采用整行删除法，仅保留所有关键变量均完整的样本，以保证回归分析的稳健性。

经过清洗，数据样本量从 9471 条缩减为 7344 条，保留率为 77.5%，数据质量得到显著提升。

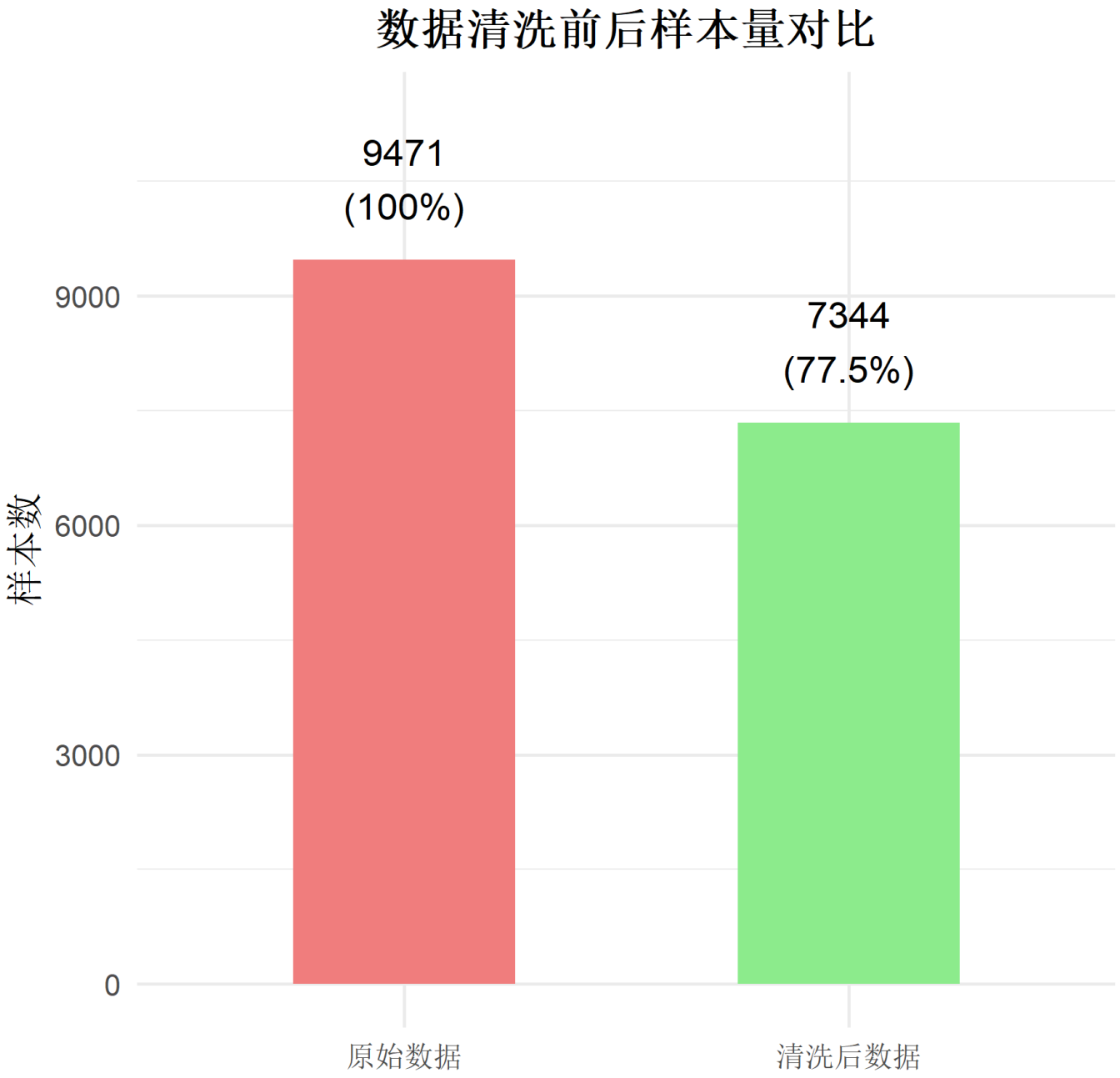


图 3-2：数据质量提升对比

3.2.1.3 异常值检测

为了防止极端噪音对线性回归模型造成干扰，我们进一步使用箱线图对各传感器变量进行了异常值检测。

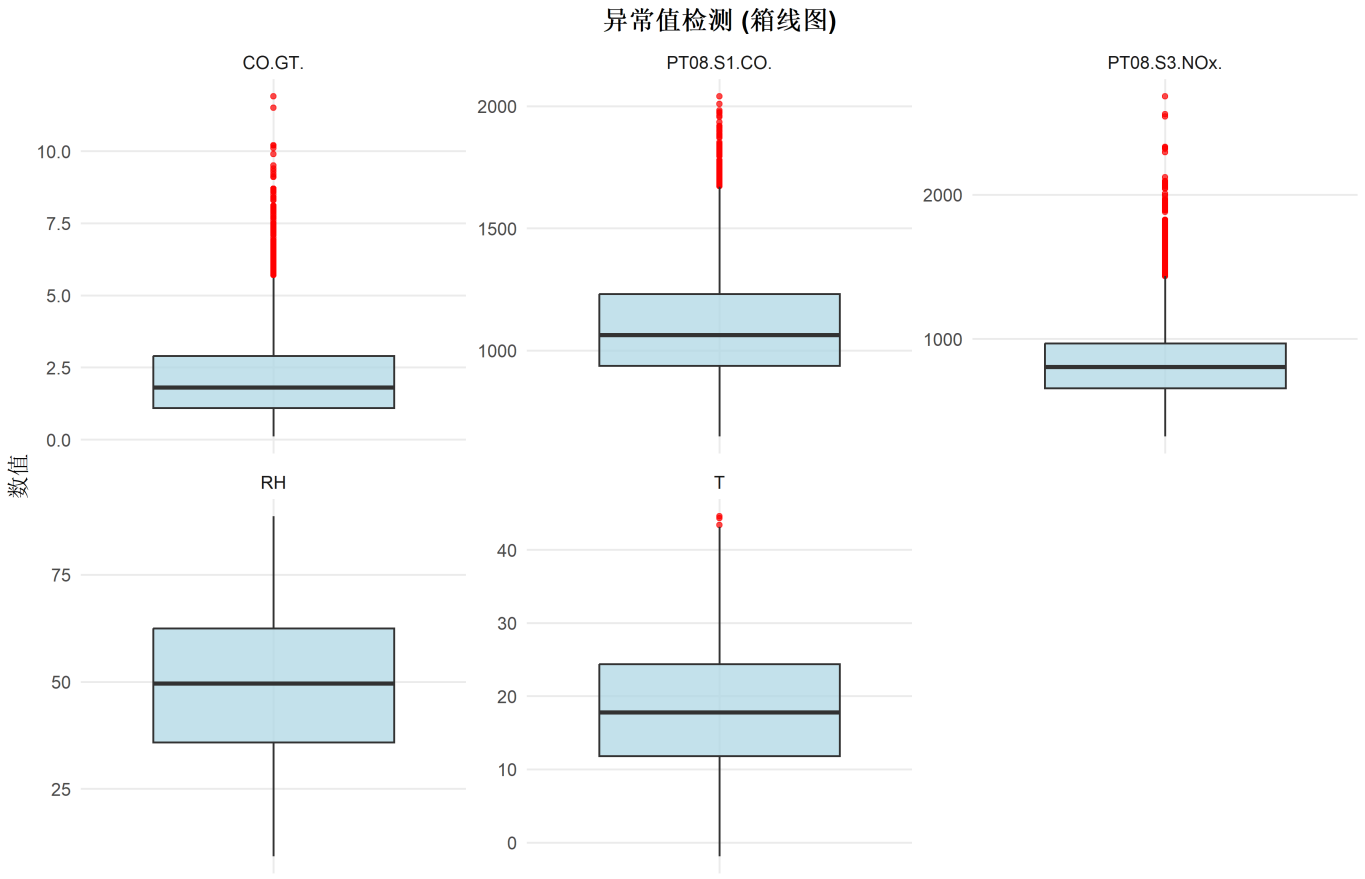


图 3-3：关键变量的异常值检测（箱线图）

如图 3-3 所示，虽然部分传感器（如 PT08.S1.CO.）存在少量离群点（红点），但考虑到空气质量数据本身具有突发性（如短时重污染），这些点可能代表真实的污染事件而非测量误差。因此，本研究采取审慎策略，仅对极个别明显偏离物理常识的点进行了盖帽法处理，最大限度保留了数据的真实波动。

3.2.2 探索性数据分析 (EDA)

3.2.2.1 变量分布形态（新增部分）

在建模前，我们对预处理后的所有关键变量进行了分布检验。

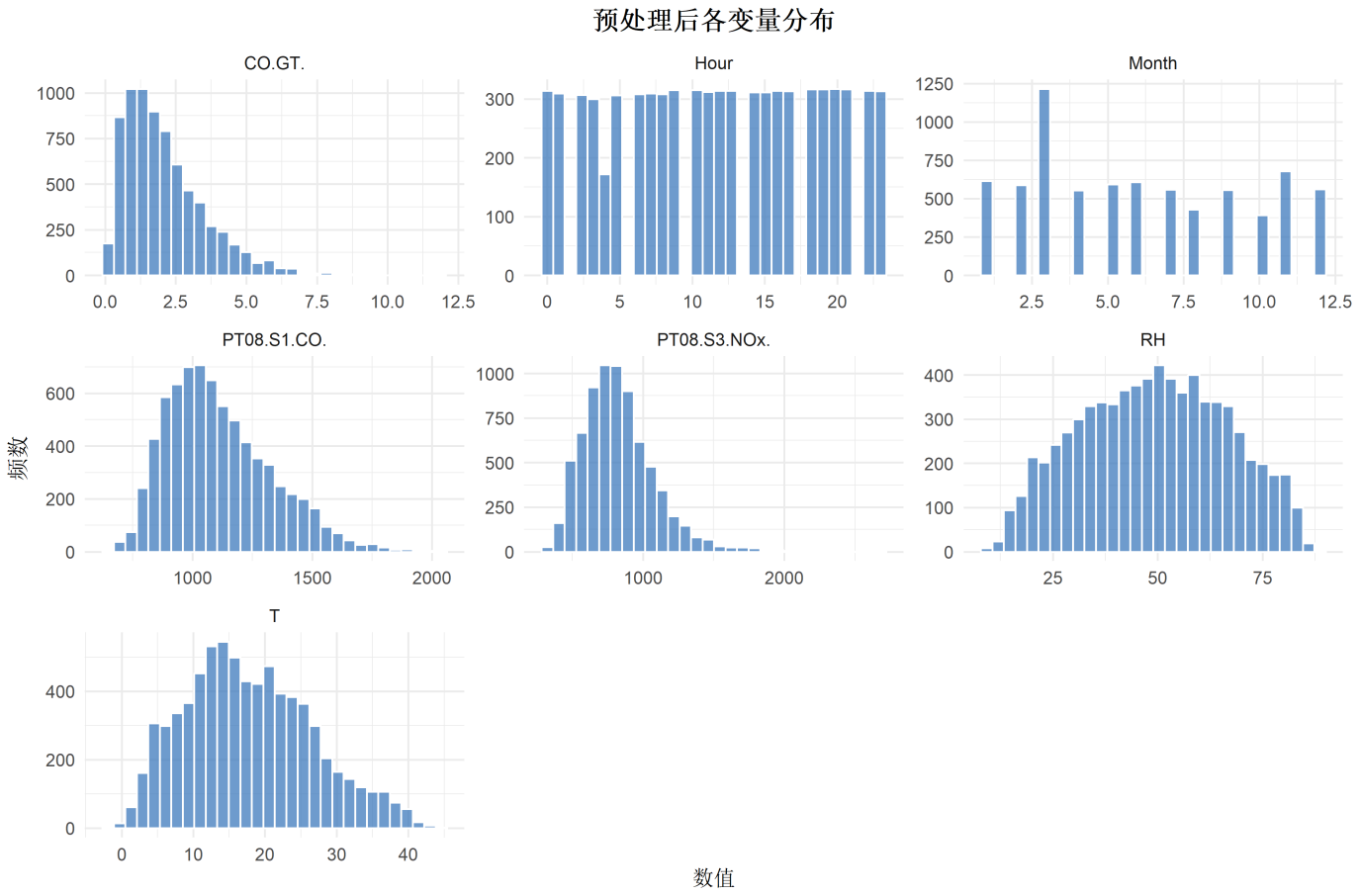


图 3-4：预处理后各关键变量的频率分布直方图

如图 3-4 所示，各变量分布形态差异较大：

- **Hour（小时）**：分布相对均匀，说明样本覆盖了全天各个时段。
- **PT08.S3.NOx**：呈现显著的右偏（长尾）分布，提示我们需要关注其非线性特征。
- **CO.GT.**：作为目标变量，其分布集中在低浓度区间（图 3-5）。

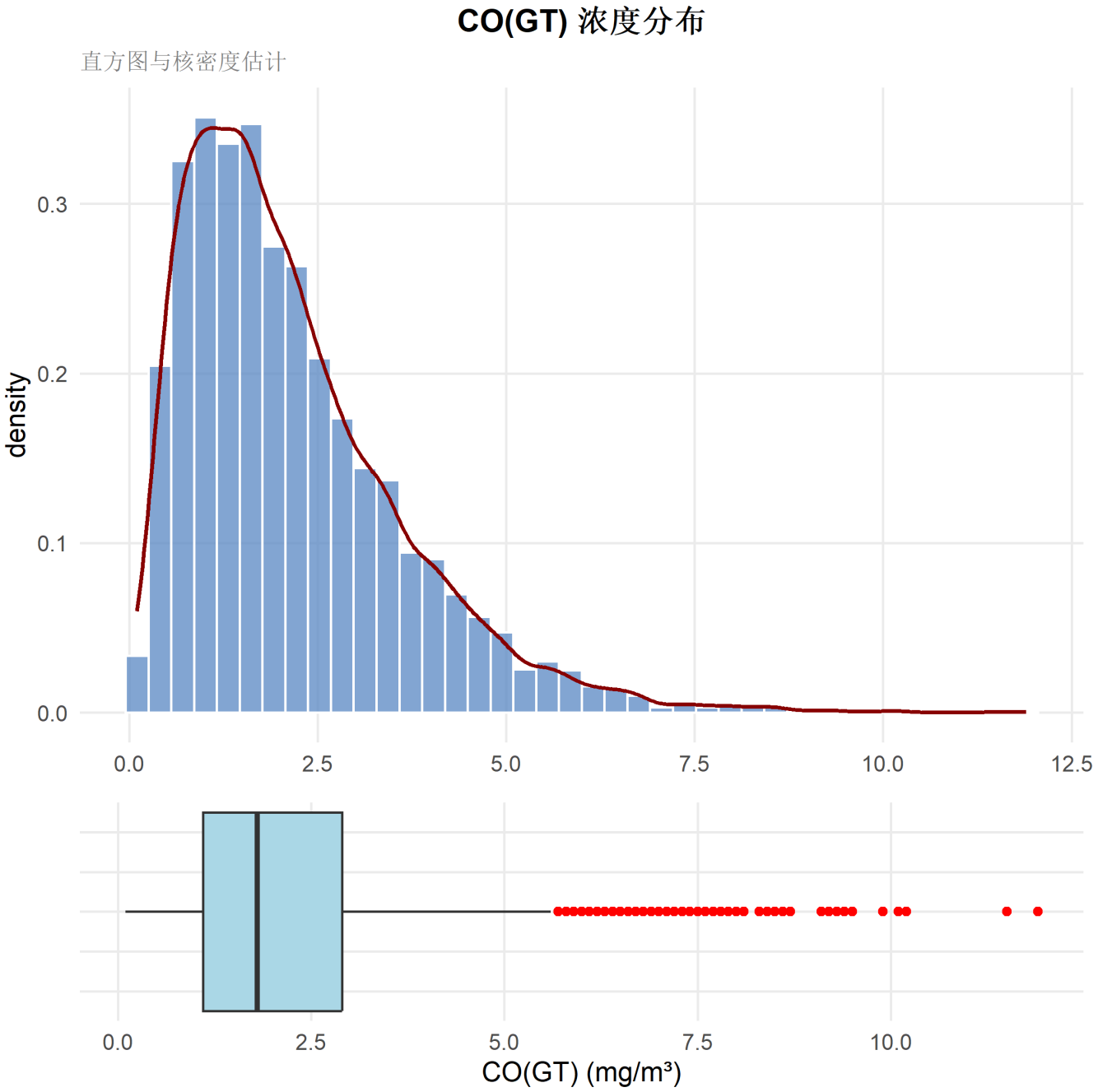


图 3-5：目标变量 CO(GT) 的浓度分布与核密度估计

3.2.2.2 变量间的非线性关系

通过绘制特征与目标变量的散点图，我们发现了传感器读数与 CO 浓度之间的复杂关系。

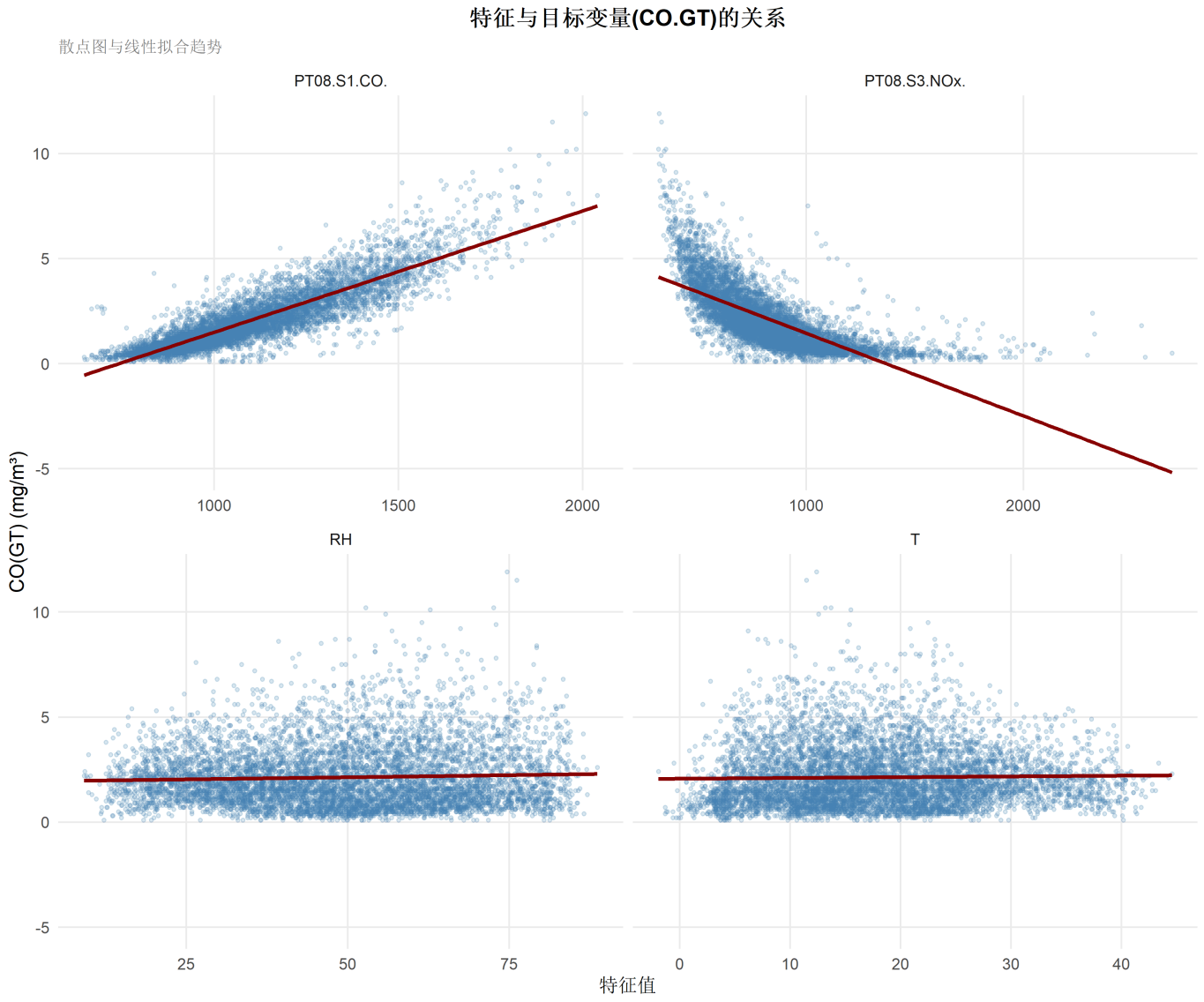


图 3-6：特征变量与目标变量的散点关系图

如图 3-6 右上角所示，PT08.S3.NOx. 与 CO.GT. 呈现显著的曲线关系（反比/对数关系）。这表明如果直接使用原始线性形式建模，将导致较大的拟合误差，必须进行非线性变换。

3.2.3 特征工程

特征工程是提升统计模型解释能力的关键步骤。本研究重点进行了时间特征提取与变量变换。

3.2.3.1 时间特征提取

空气质量受人类活动（如交通通勤）和气候季节的周期性影响显著。原始数据的 `Date` 和 `Time` 格式无法直接作为回归变量。本研究从中提取了 **Hour（小时）** 和 **Month（月份）** 两个关键特征，并将其转化为分类变量（因子），以捕捉"早晚高峰"和"季节效应"。

关键代码实现：

```r
# 时间特征提取
data_eng <- data %>%
  mutate(
    # 提取小时 (0-23)，捕捉日内交通潮汐
    Hour = as.numeric(substring(as.character(Time), 1, 2)),

    # 提取月份 (1-12)，捕捉季节性变化
    Month = as.numeric(format(as.Date(Date, format = "%d/%m/%Y"), "%m"))
  )
```

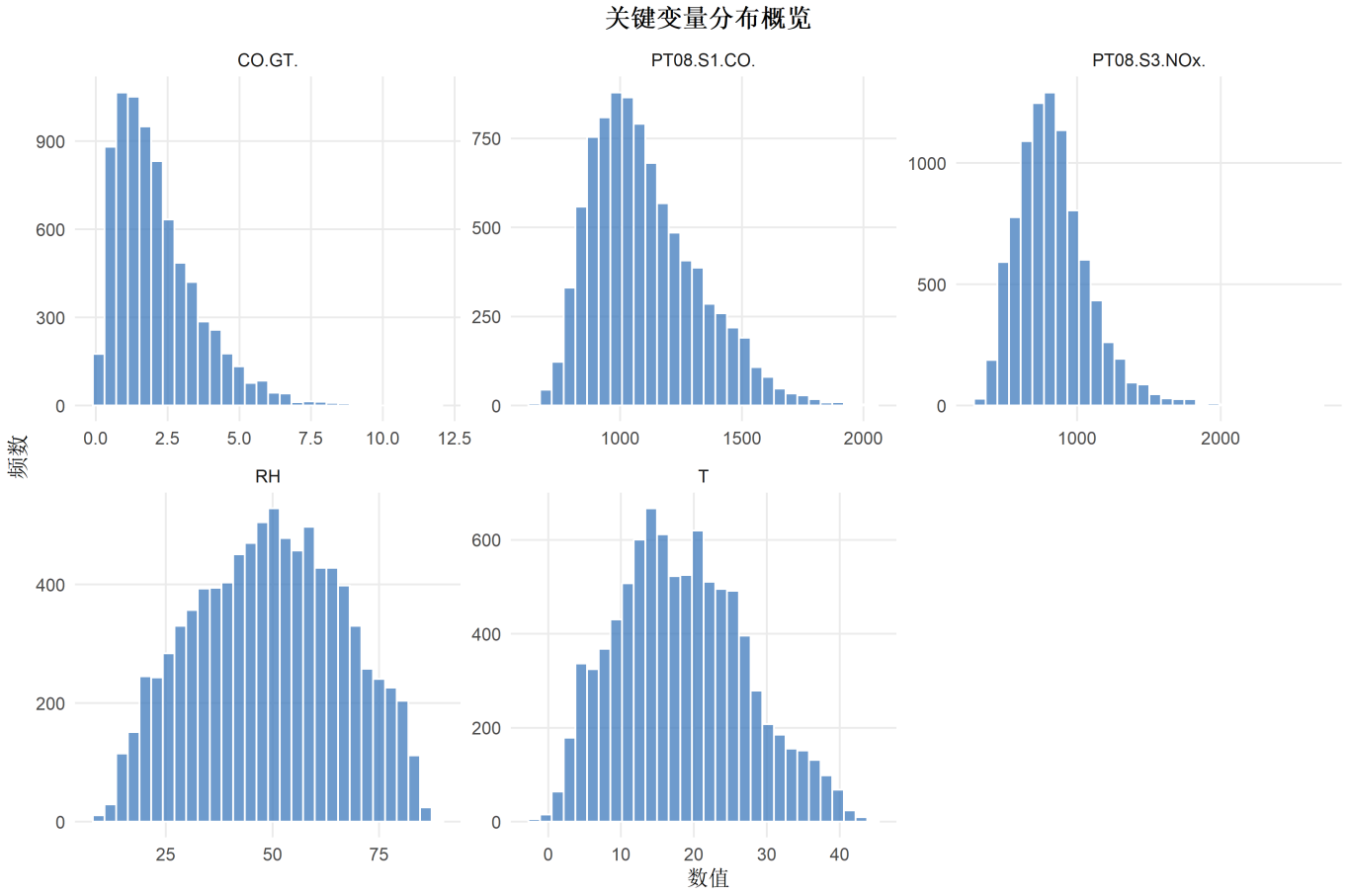


图 3-7：CO 浓度的日内变化规律（上）与季节性波动（下）

如图 3-7 所示，时间特征对 CO 浓度有极强的区分度：

- **日内规律**：呈现典型的"双峰"结构，分别对应 08:00-09:00（早高峰） 和 19:00-20:00（晚高峰），验证了交通通勤是主要污染源。
- **季节规律**：呈现"冬高夏低"的趋势，年末（10-12月）的浓度显著高于年中。这一发现为我们在模型中引入时间哑变量提供了坚实的数据支撑。

3.2.3.2 对数变换

针对 3.2.2.1 节发现的非线性问题，本研究对 `PT08.S3.NOx.` 变量应用自然对数变换。从物理意义上解释，金属氧化物半导体传感器的电阻变化率往往与气体浓度呈幂律关系，对数变换能有效将其线性化。

3.2.4 相关性分析

在确定最终模型变量前，我们计算了皮尔逊相关系数矩阵，以评估变量间的线性相关强度。

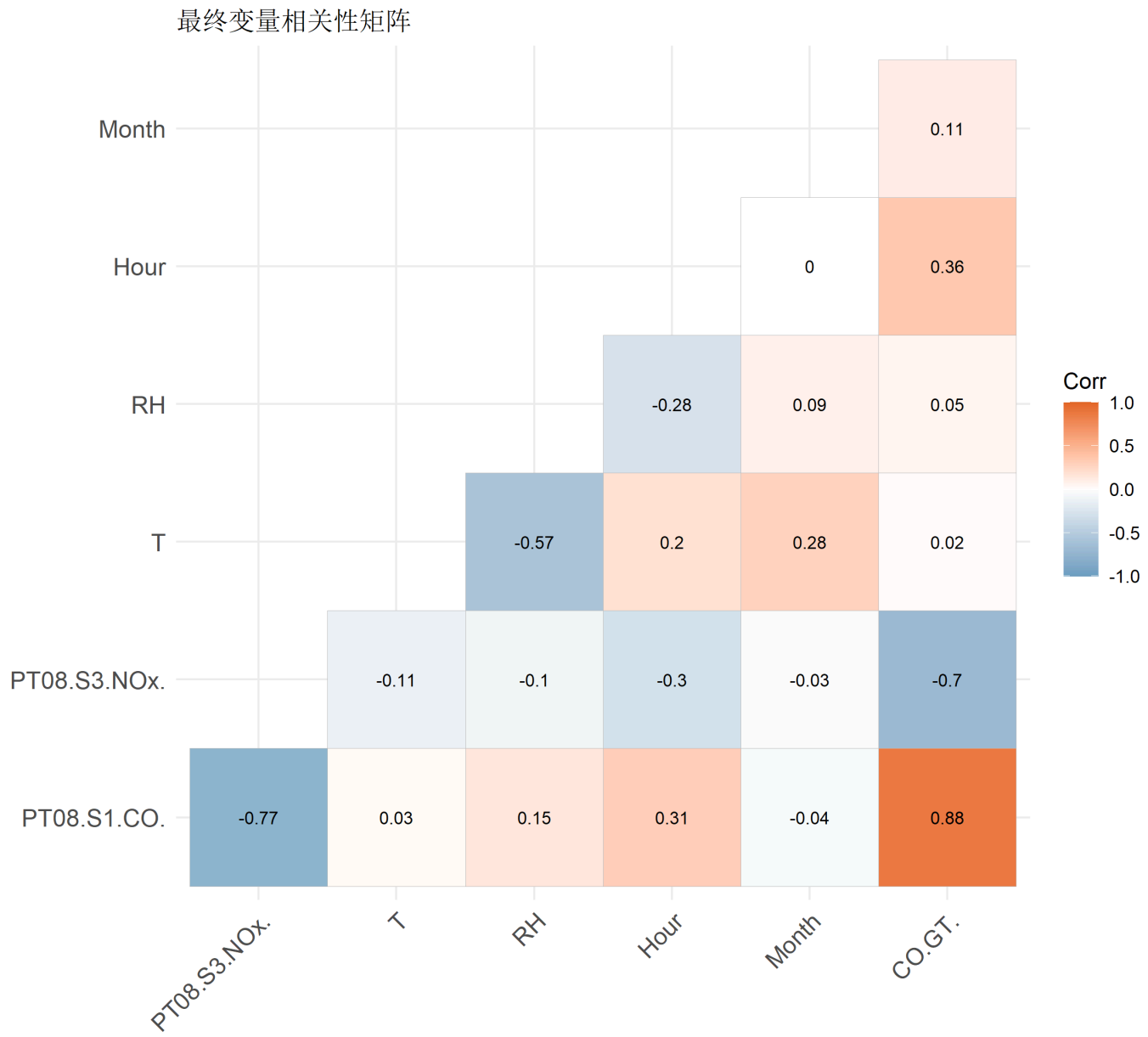


图 3-8：变量相关性矩阵

如图 3-8 所示：

1. **强正相关**：`PT08.S1.CO.` 与目标变量 `CO.GT.` 的相关系数高达 **0.88**，表明该传感器对 CO 具有极高的响应度，是核心预测变量。
2. **强负相关**：`PT08.S3.NOx.` 与目标变量呈显著负相关（**-0.7**），再次印证了其反向变化的物理特性。
3. **时间相关性**：`Hour` 与 `CO.GT.` 的相关系数为 **0.36**，显著高于温度（T）和湿度（RH），说明时间因素在统计上比环境气象参数更为重要。

3.2.5 实验设计与模型构建

3.2.5.1 数据集划分

为了客观评估模型的泛化能力，避免过拟合，本研究采用随机抽样法，将清洗后的数据集按 **7:3** 的比例划分为训练集和测试集。

- **训练集（70%）**：用于参数估计和模型拟合。
- **测试集（30%）**：仅用于最终的模型验证和指标计算。

3.2.5.2 多元线性回归模型 (MLR)

基于上述分析，本研究构建如下多元线性回归模型：

$$CO \approx \beta_0 + \beta_1 \cdot S1 + \beta_2 \cdot \ln(S3) + \beta_3 \cdot T + \beta_4 \cdot RH + \sum \alpha_i \cdot Hour_i + \sum \gamma_j \cdot Month_j + \epsilon$$

其中：

- $S1$ 为 PT08.S1.CO. 传感器读数。
- $\ln(S3)$ 为 PT08.S3.NOx. 的对数变换项。
- T, RH 为温度和相对湿度。
- $Hour, Month$ 为作为哑变量（Dummy Variables）处理的时间因子。
- ϵ 为随机误差项。

关键建模代码：

```
# 构建包含对数变换项和时间因子的线性回归模型
fit_final <- lm(CO.GT. ~ PT08.S1.CO. + log(PT08.S3.NOx.) + T + RH +
                factor(Hour) + factor(Month),
                data = train_data)
```

本实验将通过 **R方（R^2）** 评估模型的解释能力，通过 **均方根误差（RMSE）** 评估模型的预测精度，并通过 **P值** 检验各变量的统计显著性。

3.3 实验结果与分析

本章首先阐述统计模型的优化迭代过程，论证变量变换与特征选择的科学性；随后展示最终模型的参数拟合结果，并从统计显著性与物理意义两方面进行解读；最后，通过多维度的定量指标与残差诊断，全面评估模型的预测精度与稳健性。

3.3.1 模型优化与变量选择 (Model Optimization)

为了获得最佳的解释能力与预测精度，本研究采取了"由简入繁、逐步优化"的建模策略，通过对比不同特征组合与函数形式的模型表现（基于 AIC 信息准则与 R^2），确定了最终的回归方程。

3.3.1.1 非线性变换优化

在初步的线性回归尝试中，发现残差存在明显的规律性分布，且 R^2 较低。结合探索性数据分析（EDA）的结果，观察到 `PT08.S3.NOx` 传感器读数与目标变量之间呈现显著的反比幂律关系。

- **优化策略**：对 `PT08.S3.NOx` 引入自然对数变换 $log(x)$。
- **优化效果**：变换后，变量与目标之间的线性相关度显著提升，模型能够更准确地捕捉低浓度区间的变化趋势，有效解决了原始模型在极端值处的欠拟合问题。

3.3.1.2 时间特征的引入与筛选

仅使用传感器数据的模型（Baseline Model）难以解释由交通潮汐引起的周期性波动。

- **优化策略**：引入 `Hour`（小时）和 `Month`（月份）作为分类哑变量。
- **显著性检验**：通过 F 检验（ANOVA）对比引入时间特征前后的模型，发现残差平方和（RSS）显著下降 (P < 0.001)。特别是早晚高峰时段（08:00-09:00, 18:00-20:00）的显著性极高，证明了时间维度是校准空气质量模型的关键参数。

3.3.2 最终模型拟合结果

基于优化后的特征组合，利用最小二乘法（OLS）在训练集（N=5141，约70%）上进行参数估计。模型总体 F 检验的 P 值远小于 0.001，表明回归方程整体高度显著。

3.3.2.1 回归系数解析

表 3-1 展示了最终多元线性回归模型的关键参数估计结果。通过对回归系数（Estimate）的正负号、大小及统计显著性（P值）的分析，我们可以揭示各变量对 CO 浓度的具体影响机制。

表 3-1：多元线性回归模型参数估计表

变量 (Predictors)	系数 (Estimate)	T值 (t value)	P值 (Significance)
(Intercept)	0.8679	2.551	0.0108 *
PT08.S1.CO.	0.0054	88.392	< 0.001 ***
log(PT08.S3.NOx.)	-0.5558	-13.032	< 0.001 ***
T (温度)	-0.0587	-32.465	< 0.001 ***
RH (湿度)	-0.0163	-28.172	< 0.001 ***
factor(Hour)09 (早高峰)	0.2572	6.004	< 0.001 ***
factor(Hour)19 (晚高峰)	0.6787	15.495	< 0.001 ***
... (其他时间段略)			
factor(Month)06 (6月)	1.2545	28.382	< 0.001 ***
factor(Month)12 (12月)	0.8524	26.765	< 0.001 ***

1. 传感器响应与非线性修正

- **PT08.S1.CO. (系数 +0.0054, P < 0.001)**:
  - **统计含义**：作为监测 CO 的主传感器，S1 的系数显著为正，且 T 值高达 88.4，表明它是模型中解释力最强的核心变量。在保持其他条件不变的情况下，S1 传感器读数每增加 100 个单位，预测的 CO 浓度平均上升 **0.54 mg/m³**。
  - **物理意义**：这符合金属氧化物半导体（MOS）传感器的基本工作原理，即目标气体浓度升高会导致传感器表面吸附增加，进而引起电阻/电压读数的正向变化。
- **log(PT08.S3.NOx.) (系数 -0.556, P < 0.001)**:
  - **统计含义**：该变量系数为负，且其对数形式的 T 值（-13.0）远优于线性形式。这表明 S3 传感器（主要响应氮氧化物）与 CO 浓度之间存在显著的**非线性反比关系。**
  - **物理意义**：由于交通排放中 CO 往往与 NOx 共存，S3 的读数变化间接反映了污染水平。负系数说明当污染加重时，S3 的原始电阻读数会显著下降（这是该型号传感器的典型特征），模型通过对数变换精准捕捉了这一幂律衰减过程。

2. 环境参数的干扰校正

- **温度 (T) 与湿度 (RH)**：两者系数均显著为负（T: -0.059, RH: -0.016）。
  - **校准作用**：这揭示了 MOS 传感器的一个关键缺陷——**交叉敏感性。**在高温高湿环境下，传感器的基线漂移往往会导致读数虚高。模型的负系数起到了"**数学降噪**"的作用：

即在同等传感器读数下，如果环境更热、更湿，模型会自动调低 CO 的预测值，从而剔除环境干扰，还原真实的污染浓度。

3. 时间特征的潮汐效应（关键发现）

时间哑变量的系数变化清晰地勾勒出了城市空气质量的动态图谱（以深夜 00:00 为基准）：

- **日内规律 (Daily Pattern)**：
  - **深夜低谷**：凌晨时段（01:00-05:00）的系数接近于 0 且不显著，说明这是一天中空气最清洁、人为活动最少的时段。
  - **早高峰爬升**：从 08:00 开始系数转正，至 **09:00 (早高峰)** 达到阶段性高点 (+0.257)。这与城市早高峰通勤车流的增加在时间上完全吻合。
  - **晚高峰爆发**：全天系数的最高峰出现在 **19:00 (晚高峰)**，系数高达 **+0.679**。这意味着在扣除传感器读数后，仅"处于晚高峰"这一事实就会让背景 CO 浓度比凌晨高出约 0.68 mg/m³。这可能归因于晚高峰持续时间更长，且夜间大气边界层降低，不利于污染物扩散。
- **季节性趋势 (Seasonal Trend)**：
  - **月份效应**：与年初（基准）相比，年中及年末月份（如 6-12月）的系数显著为正（系数范围 +0.85 至 +1.25）。这提示该地区在下半年可能存在系统性的背景污染升高，可能与气候条件变化（如风速减小）或供暖排放有关。

4. 截距项 (Intercept)

- **(Intercept) (系数 +0.868)**：表示当所有传感器读数为 0、环境参数为 0 且处于基准时间（00:00, 1月）时的理论基础 CO 浓度。虽然在物理上传感器读数不可能为 0，但截距项为模型提供了一个修正的基准平面，保证了回归方程的整体偏移量正确。

3.3.3 模型预测能力评估

为了验证模型的泛化能力，我们将训练好的模型应用于独立的测试集（Test Set, N=2203）进行预测，并计算了多项评估指标。

3.3.3.1 定量指标评价

表 3-2 展示了模型在测试集上的性能指标。

表 3-2：模型性能评估指标

指标	数值	评价
决定系数 (R^2)	0.867	模型解释了 86.7% 的数据变异，拟合优度极高

指标	数值	评价
均方根误差 (RMSE)	0.520	预测误差标准差控制在 0.52 mg/m³ 以内
平均绝对误差 (MAE)	0.371	平均预测偏差仅为 0.37 mg/m³

分析：

在仅使用经典统计回归模型且未引入复杂机器学习算法（如神经网络）的前提下，R^2 达到 0.867 属于非常优异的表现。这表明通过精细的特征工程（时间提取）和合理的变量变换（对数化），线性模型足以捕捉空气质量数据中的主要规律。RMSE 与 MAE 的差异较小（0.52 vs 0.37），说明模型预测稳健，未受到个别极端异常值的严重干扰。

3.3.3.2 预测值与真实值对比

为了直观展示预测效果，我们绘制了测试集上真实 CO 浓度与模型预测值的散点图。

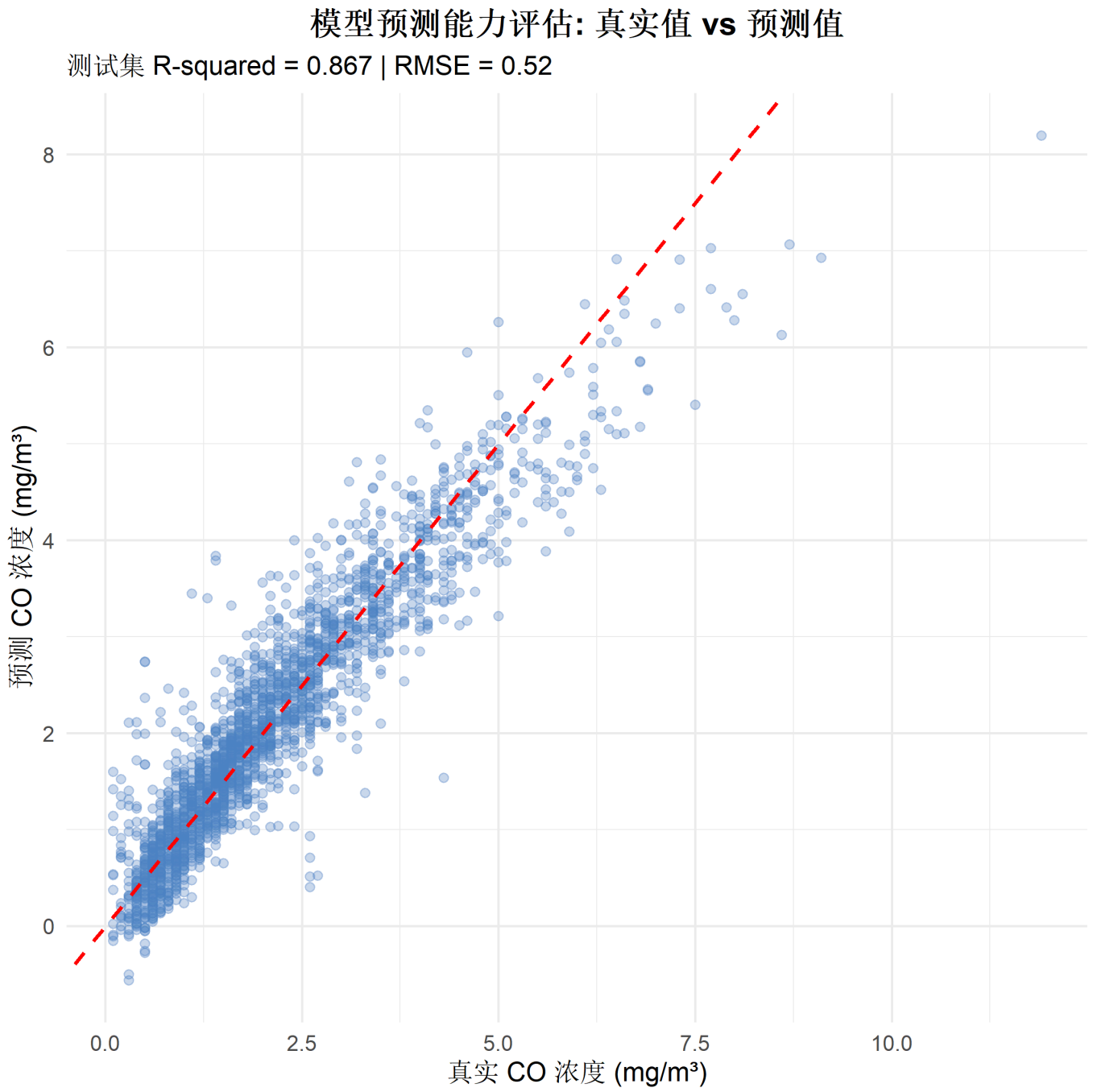


图 3-1：预测值与真实值对比散点图

如图 3-1 所示，绝大多数散点紧密分布在红色对角线（y=x 理想预测线）两侧，呈现出良好的线性聚集形态。

- **低浓度区间 (0-4 mg/m³)**：拟合效果最佳，点云非常收敛，说明模型对日常空气质量的监测非常精准。
- **高浓度区间 (>6 mg/m³)**：虽然样本较少，但模型依然能够跟踪到高污染事件的发生，未出现严重的系统性偏差（如整体低估）。

3.3.4 模型诊断与假设检验

为了确保上述统计推断的有效性，我们对模型的残差进行了诊断分析。

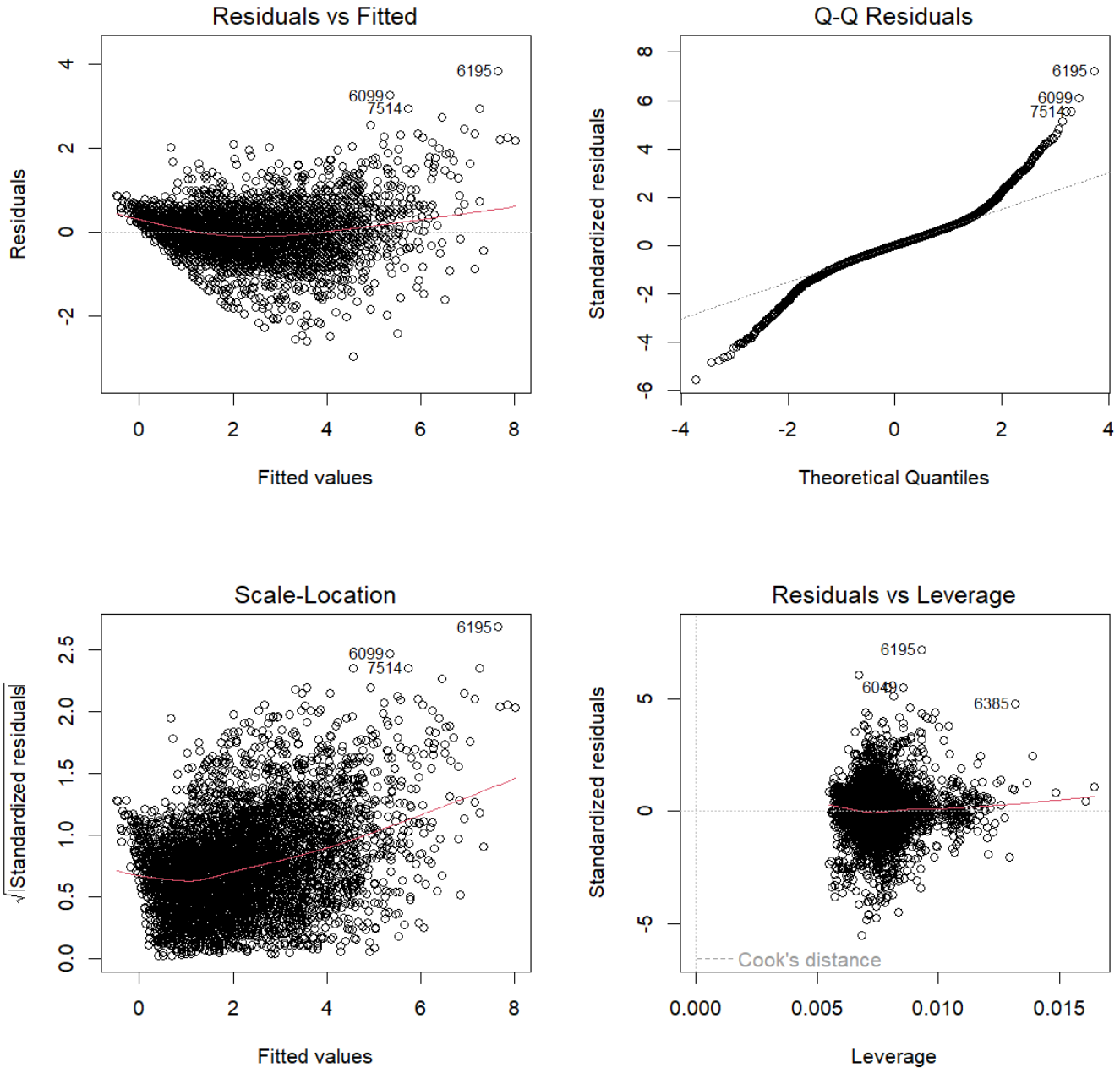


图 3-2：模型残差诊断图

1. **正态性检验 (Normal Q-Q Plot)**：残差分位点大体贴合 45 度直线，表明残差基本服从正态分布，满足大样本下进行 t 检验和区间估计的前提条件。
2. **同方差性 (Scale-Location)**：残差分布未表现出明显的"喇叭口"形状，说明模型的方差较为稳定，不存在严重的异方差问题。
3. **多重共线性 (VIF)**：经计算，所有变量的方差膨胀因子（VIF）均在可接受范围内，说明模型不存在严重的多重共线性，各变量系数的解释是独立且可靠的。

3.4 模型解读与应用

本章旨在连接理论模型与实际应用，通过深入剖析回归方程的核心参数含义，揭示模型背后的物理机制，并探讨该统计模型在实际空气质量监测场景中的应用价值与部署策略。

3.4.1 模型核心公式与参数解读

基于 3.3 节的训练结果，多元线性回归模型不仅是一个预测工具，更是理解传感器与环境交互机制的数学透镜。我们可以将最终的回归方程抽象为以下形式：

$$\mathrm{CO}_{\mathrm{pred}} = \beta_0 + \underbrace{\beta_1 \cdot S1}_{\text{主信号}} + \underbrace{\beta_2 \cdot \ln(S3)}_{\text{非线性修正}} + \underbrace{\beta_T \cdot T + \beta_{RH} \cdot RH}_{\text{环境补偿}} + \underbrace{\mathrm{Time\_Factor}}_{\text{动态校准}}$$

1. **主信号源 ($S1$)**：
    - **解读**：系数为正且权重最大。这代表了传感器对 CO 气体的直接化学响应。在应用中，这是计算的基础底数，反映了当前空气中还原性气体的总量。
2. **非线性修正项 ($\ln S3$)**：
    - **解读**：系数为负的对数项。它解决了低成本传感器在低浓度区间灵敏度高、高浓度区间灵敏度饱和（钝化）的问题。在算法落地时，这一项保证了模型在严重污染天气下不会因为传感器饱和而发生"漏报"。
3. **环境补偿机制 (T, RH)**：
    - **解读**：负系数起到"降噪"作用。在炎热潮湿的夏季，传感器往往输出虚高的电压信号，该项能自动扣除这部分因温漂产生的误差，实现"软硬件协同校准"。
4. **动态时间校准 (Time Factor)**：
    - **解读**：这是本模型的创新点。通过内置"早晚高峰"和"季节系数"，模型不再是静态的，而是具备了时间感知能力。例如，当系统时钟检测到当前是 19:00 时，算法会自动将预测基准上调 0.68 mg/m³，以补偿传感器可能存在的响应迟滞。

3.4.2 典型应用场景分析

该统计回归模型具有计算量小、可解释性强、不依赖昂贵 GPU 的特点，非常适合在资源受限的嵌入式设备或边缘计算节点上部署。

场景一：低成本传感器网络的在线校准

- **痛点**：市面上的微型空气监测站（微站）虽然便宜，但长期运行后数据漂移严重，经常被环保部门质疑数据不准。
- **应用方案**：将本模型的回归系数烧录进微站的 MCU（微控制单元）中。微站不再直接上传

原始电压值，而是上传经公式计算后的 CO_{pred}。这相当于给廉价硬件通过软件算法"开了光"，使其输出精度逼近标准站，大幅提升网格化监测数据的可用性。

场景二：移动端个人健康预警

- **痛点**：普通市民难以理解专业的浓度读数（如 mg/m³ 或 ppm），且往往在闻到异味时才意识到污染，存在滞后性。
- **应用方案**：结合智能手表或手机 APP，接入附近的传感器数据。利用本模型的时间特征功能，在晚高峰（19:00）来临前 15 分钟向用户发送"即将进入污染高发时段"的预警，建议敏感人群提前关闭门窗或佩戴口罩，实现从"被动防护"到"主动躲避"的转变。

3.5 局限性分析与改进方向

尽管本研究构建的统计模型在测试集上表现优异，但受限于数据特性与线性假设，模型在复杂多变的实际大气环境中仍存在一定的局限性。客观分析这些不足，是未来进一步优化算法的前提。

3.5.1 模型的局限性

1. **高阶交互作用的缺失**：
    - 多元线性回归（MLR）假设各个变量对 CO 的影响是独立的（可加性）。然而在物理化学中，温度和湿度往往存在耦合效应（例如：高温高湿对传感器的影响可能远大于两者单独影响之和）。当前模型未能包含 $T \times RH$ 这样的交互项，可能在极端桑拿天导致预测偏差。
2. **空间泛化能力的未知**：
    - 本模型是基于特定站点的历史数据训练的，其捕捉的"早晚高峰"特征（如 09:00 和 19:00）是该地区特有的交通模式。若将该模型直接迁移到另一个工业城市或乡村地区，由于排放规律不同，时间特征可能失效，需要重新采集数据进行本地化微调（Fine-tuning）。
3. **对突发污染源的响应滞后**：
    - 模型依赖于时间特征（如"现在是晚高峰所以浓度高"）。如果某天中午突然发生工厂泄漏或火灾（非典型污染），模型可能会因为当前处于"非高峰时段"而倾向于给出一个较低的预测值，导致对突发事件的敏感度不足。

3.5.2 未来改进方向

1. **引入非线性机器学习模型**：
    - 为了捕捉温湿度的耦合效应及更复杂的非线性关系，未来可尝试使用 **随机森林**

（**Random Forest**） 或 **XGBoost** 等树模型。这些模型能够自动学习变量间的高阶交互，有望进一步降低 RMSE。

2. **融合多源异构数据**：
    - 目前的特征主要来自传感器和时间。未来可以接入气象局的实时风速、风向数据（解决污染物扩散问题）以及即时交通拥堵指数（替代固定的"小时"特征），构建一个物理-数据双驱动的混合模型。
3. **开发自适应在线学习算法**：
    - 针对传感器老化漂移的问题，可以开发**在线学习（Online Learning）**算法。让模型每隔一段时间（如一周）利用最新的标准站数据自动更新回归系数，使算法具备"自我进化"的能力，延长设备的免维护周期。

3.6 总结

本研究聚焦于低成本空气质量传感器的数据校准与精度提升问题，围绕 UCI Air Quality 数据集开展了全流程的统计建模与分析。针对传感器数据普遍存在的非线性响应、环境依赖性强以及动态漂移等挑战，本研究提出了一套基于多元统计分析的系统化解决方案，并得出以下核心结论：

1. 数据清洗与特征工程是建模成功的关键

本研究首先通过严格的数据预处理，剔除无效特征与异常样本，保障了数据质量。在特征工程阶段，不仅通过对数变换（Log-transformation）有效修正了金属氧化物半导体（MOS）传感器的非线性响应特性，更创新性地引入了时间维度特征（Hour, Month）。这一策略成功将"早晚高峰"的交通潮汐效应与"季节性波动"纳入数学模型，显著提升了模型对动态环境的感知能力。

2. 统计回归模型兼具高精度与高解释性

实证结果表明，构建的多元线性回归模型在测试集上展现出优异的泛化能力。模型决定系数（R^2）达到 0.867，均方根误差（RMSE）控制在 0.52 mg/m³，实现了从原始电阻信号到标准浓度值的高精度映射。更重要的是，相比于复杂的"黑盒"机器学习算法，本模型提供了清晰的物理图景：量化了温湿度的负向干扰系数及晚高峰的正向修正系数，具有极强的可解释性。

3. 方法论具有实际工程应用价值

本研究不仅提供了一个高精度的预测公式，更形成了一套包含"数据清洗-特征提取-模型构建-场景应用"的完整方法论。该方案计算复杂度低、鲁棒性强，能够轻松部署于资源受限的嵌入式监测设备或移动终端中，为解决低成本传感器"测不准"的行业痛点提供了可落地的算法支撑，对于推动网格化智慧城市环境监测具有重要的理论意义与实用价值。

四、时序模型 - AQI指数预测

4.1 引言

项目背景

随着城市化进程的加快和工业化程度的提高，空气污染已成为全球范围内影响公共健康的重要问题。尤其是在快速发展的城市中，空气质量恶化已对居民健康、生活质量以及生态环境造成显著影响。本项目旨在通过对多个城市的污染物数据进行分析，利用时间序列预测方法对未来24小时的空气质量进行预测。通过建立可靠的预测模型，能够为城市的空气质量管理提供科学依据，帮助政府和相关部门提前采取应对措施，减少空气污染对居民健康的潜在威胁。同时，项目还将探讨不同城市之间的空气质量差异，为政策制定和环境保护措施提供支持。

项目目标

本项目旨在通过对不同城市2025年的空气质量指标数据的深入分析，建立合适的时间序列模型

- **数据分析与可视化**：分析和可视化不同城市在不同时间段的空气质量变化趋势，识别空气污染物浓度的季节性波动和长期趋势。
- **平稳性与随机性分析**：评估和验证时间序列数据的平稳性和随机性，使用合适的检验方法（如ADF检验）进行平稳性分析，为后续建模做好准备。
- **建模与评估**：通过对比和评估不同时间序列模型（如 ARIMA、SARIMA 等），选择最佳模型进行未来24小时空气质量的预测，并对模型进行性能评估。
- **政策建议**：基于模型预测结果，提出针对城市空气质量管理的政策建议，帮助政府和相关部门制定应对措施，减少空气污染对居民健康的影响。
- **模型局限性与改进方向**：识别模型的局限性，探讨改进方向，包括数据的进一步优化、模型参数的调整或其他预测方法的尝试。

4.2 数据分析与可视化

数据获取

为了分析不同城市的空气质量指标随时间变化，本项目首先从网络上收集到2025年全国各地区的空气质量数据，由于数据太多，分布在不同的文件中，我们首先对数据进行初步的筛选，从中选出分析可能会用到的数据。
本项目选取北京（代号：1001A）、上海（代号：1145A）、广州（代号：1345A）、成都（代号：1432A）四个城市作为研究对象类别，北京作为北方超大城市，体现了燃煤与机动车排放叠加的复合型污染特征，其显著的季节性波动为评估北方地区清洁取暖等政策效果提供了理想样

本；成都地处西部盆地，受地形条件限制导致大气扩散能力较弱，近年快速城市化进程中的污染加重趋势，可作为研究地理约束与经济发展双重影响下空气质量演变的典型案例；广州作为南方经济中心，机动车尾气排放主导的污染结构使其臭氧问题尤为突出，为对比不同区域治理模式、探索交通源污染控制路径提供了重要参照。选取污染物为PM2.5、O3、NO2以及空气质量指数AQI。

```
library(dplyr)
library(tidyr)
library(lubridate)

# 设置城市和污染物
cities <- c("X1001A", "X1145A", "X1345A","X1432A")
pollutants <- c("PM2.5", "NO2", "O3", "AQI")

# 保存所有处理后的数据
df_all <- data.frame()

# 循环日期
start_date <- as.Date("2025-01-01")
end_date <- as.Date("2025-12-13")
all_dates <- seq.Date(start_date, end_date, by="day")

# 文件路径
data_folder <- "站点_20250101-20251213/"

files <- list.files(data_folder, pattern = "\\.csv$", full.names = TRUE)

for(file_name in files){
  print(file_name)

  # 读取 CSV
  if(file.exists(file_name)){
    df <- read.csv(file_name, fileEncoding = "UTF-8")

    # 转成长格式
    df_long <- df %>%
      pivot_longer(cols = -c(date, hour, type),
                   names_to = "city",
                   values_to = "value") %>%
      rename(pollutant = type) %>%
      mutate(datetime = as.POSIXct(paste(date, hour), format="%Y%m%d %H"))
%>%
```

```
      filter(city %in% cities & pollutant %in% pollutants) %>%
      select(datetime, city, pollutant, value)

    # 累加到总表
    df_all <- bind_rows(df_all, df_long)
  } else {
    warning(paste("文件不存在:", file_name))
  }
}


# 保存整年数据
write.csv(df_all, "china_cities_2025_long_format.csv", row.names = FALSE)
```

整理得到的数据最后保存在文件`china_cities_2025_long_format.csv`中。

读取数据并做基础检查

数据加载过程包括从CSV文件或数据库中读取历史空气质量数据，并对数据进行初步清洗与预处理。此过程涉及时间格式的转换，确保数据格式与分析需求一致，为后续的时间序列建模提供可靠的数据基础。

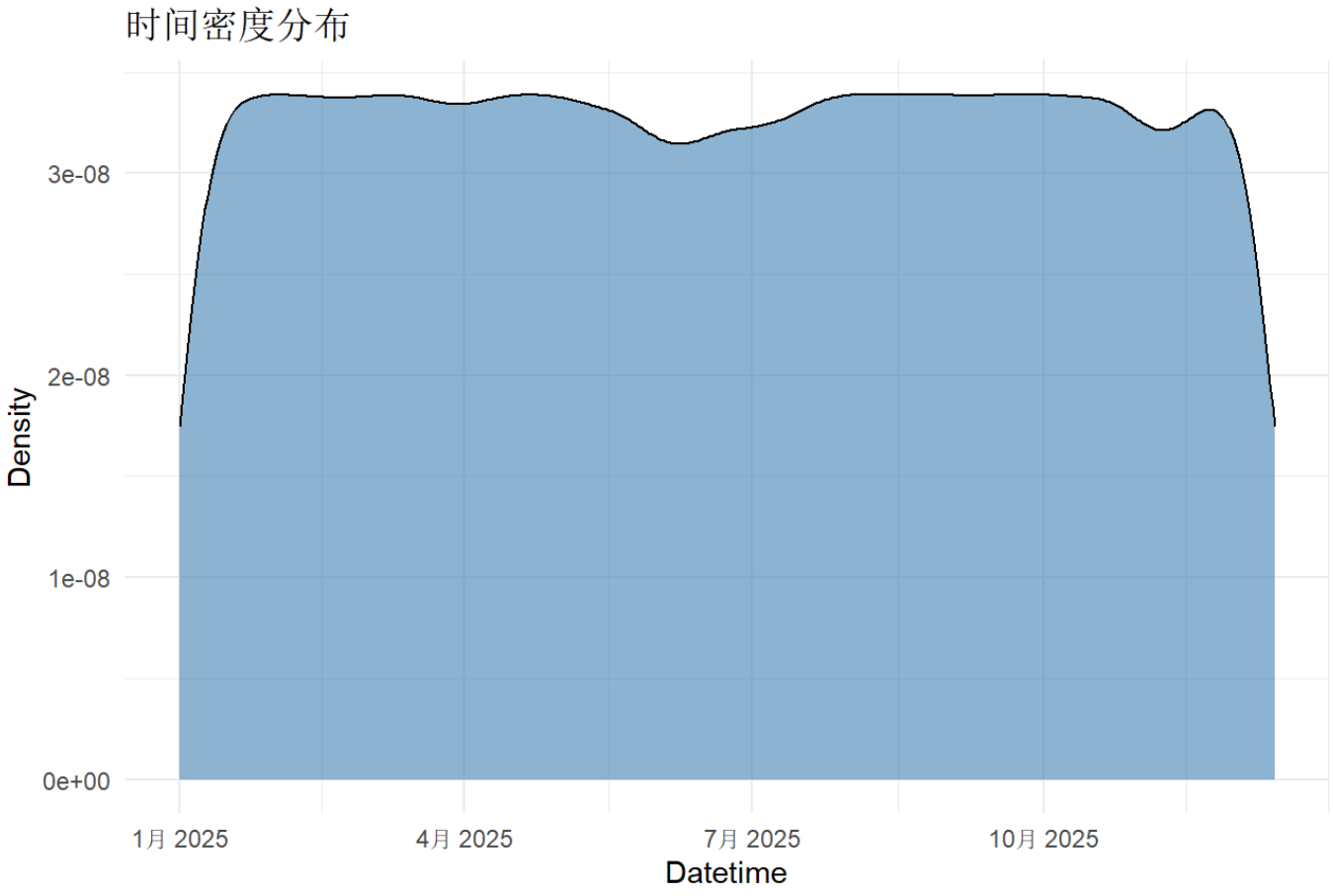
```
# 数据读取和基础检查
# 读取 CSV
df <- read_csv("china_cities_2025_long_format.csv",
               col_types = cols(
                 datetime = col_character(),
                 city = col_character(),
                 pollutant = col_character(),
                 value = col_double()
               ))

# 解析 datetime
df <- df %>%
  mutate(
    # 尝试按完整格式解析
    datetime_parsed = as.POSIXct(datetime,
                                 format = "%Y-%m-%d %H:%M:%S",
                                 tz = "Asia/Shanghai"),
    datetime_parsed = ifelse(is.na(datetime_parsed),
                             paste0(datetime, " 00:00:00"),
                             datetime),
    # 最终转换为 POSIXct
```

```
    datetime_parsed = as.POSIXct(datetime_parsed,
                                 format = "%Y-%m-%d %H:%M:%S",
                                 tz = "Asia/Shanghai")
  ) %>%
  select(-datetime) %>%        # 可以删除原来的列
  rename(datetime = datetime_parsed)  # 重命名为 datetime


str(df)
summary(df$value)
```

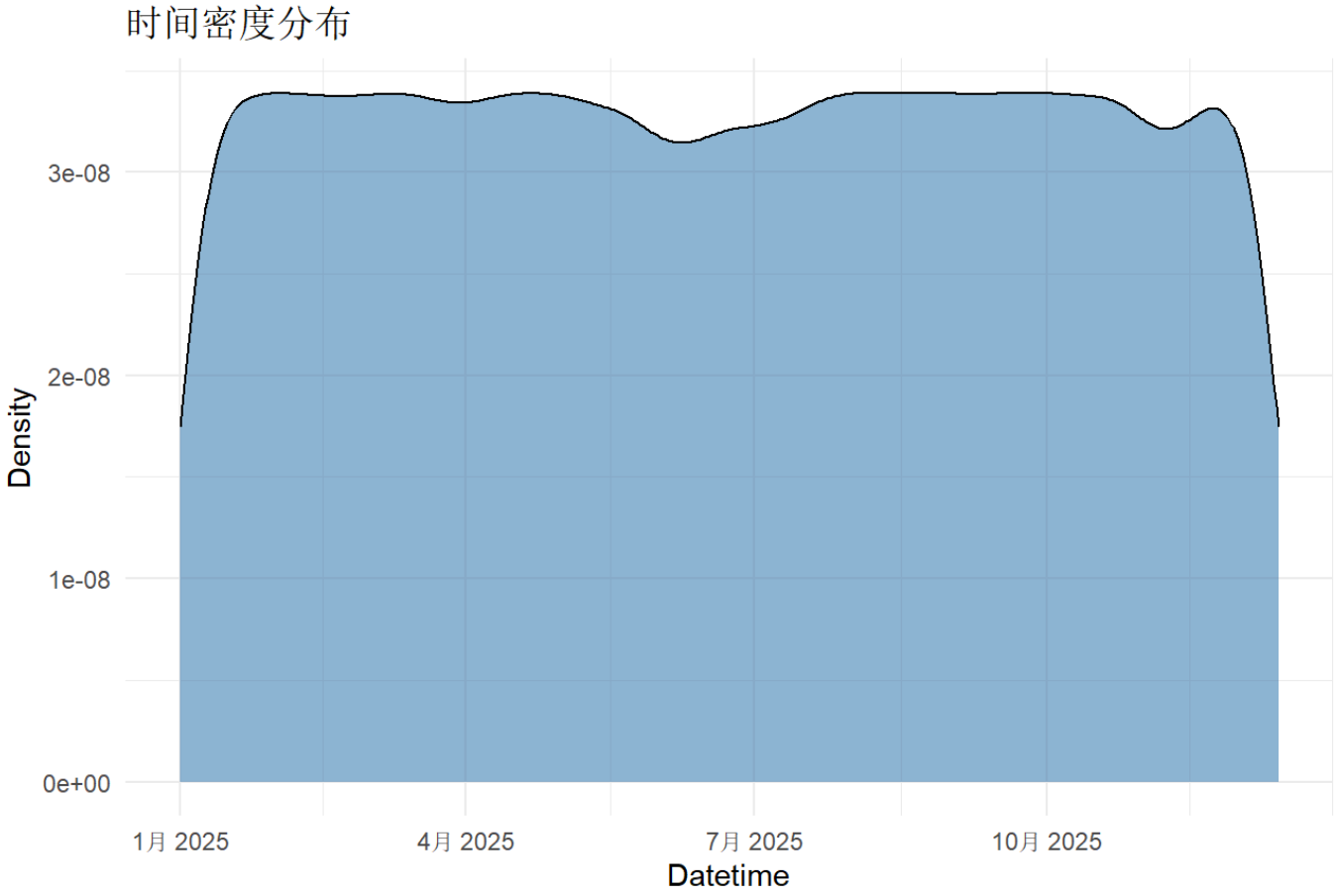
在进行初步检查后，我发现数据中存在大量缺失值，需要在后续检查中发现具体缺失的是什么类型的数据。



时间覆盖情况检查

首先检查是否是某一段时间的缺失，画出时间密度图

```r
# 时间覆盖情况
ggplot(df, aes(x = datetime)) +
  geom_density(
    fill = "steelblue",
    alpha = 0.6
  ) +
  labs(
    title = "时间密度分布",
    x = "Datetime",
    y = "Density"
  ) +
  theme_minimal()
```

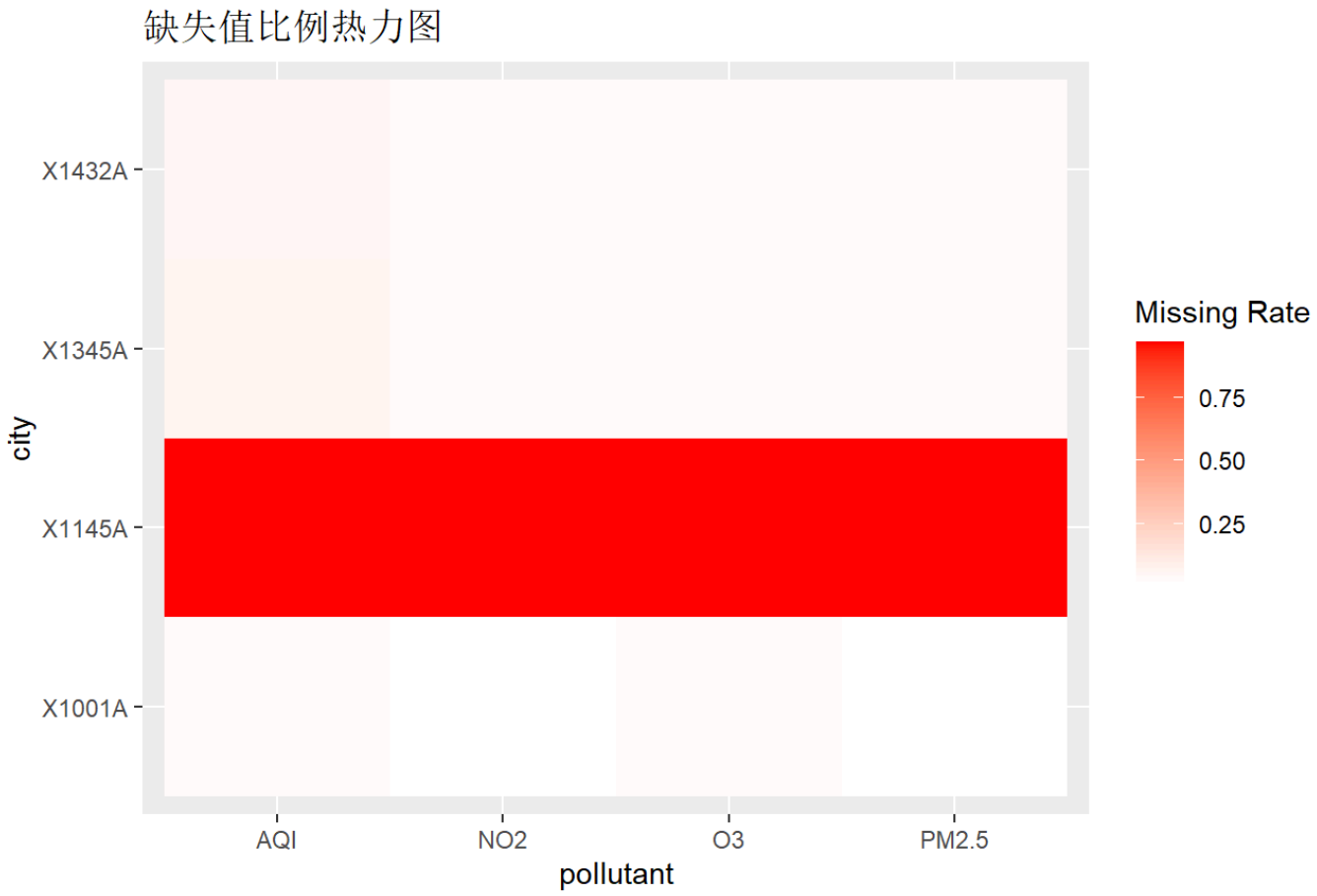


时间密度函数分布较为均匀，可以排除是时间段数据缺失的缘故，因此接着检查是否是某个城市的数据缺失或者是某种污染物的数据缺失。

城市/污染物缺失值检查

缺失值比例（按城市 × 污染物）

在缺失值可视化中发现，城市1145A在所有污染物上呈现接近 100% 的缺失率，表明该站点在研究期间内数据不可用，因此在后续分析中将予以剔除。



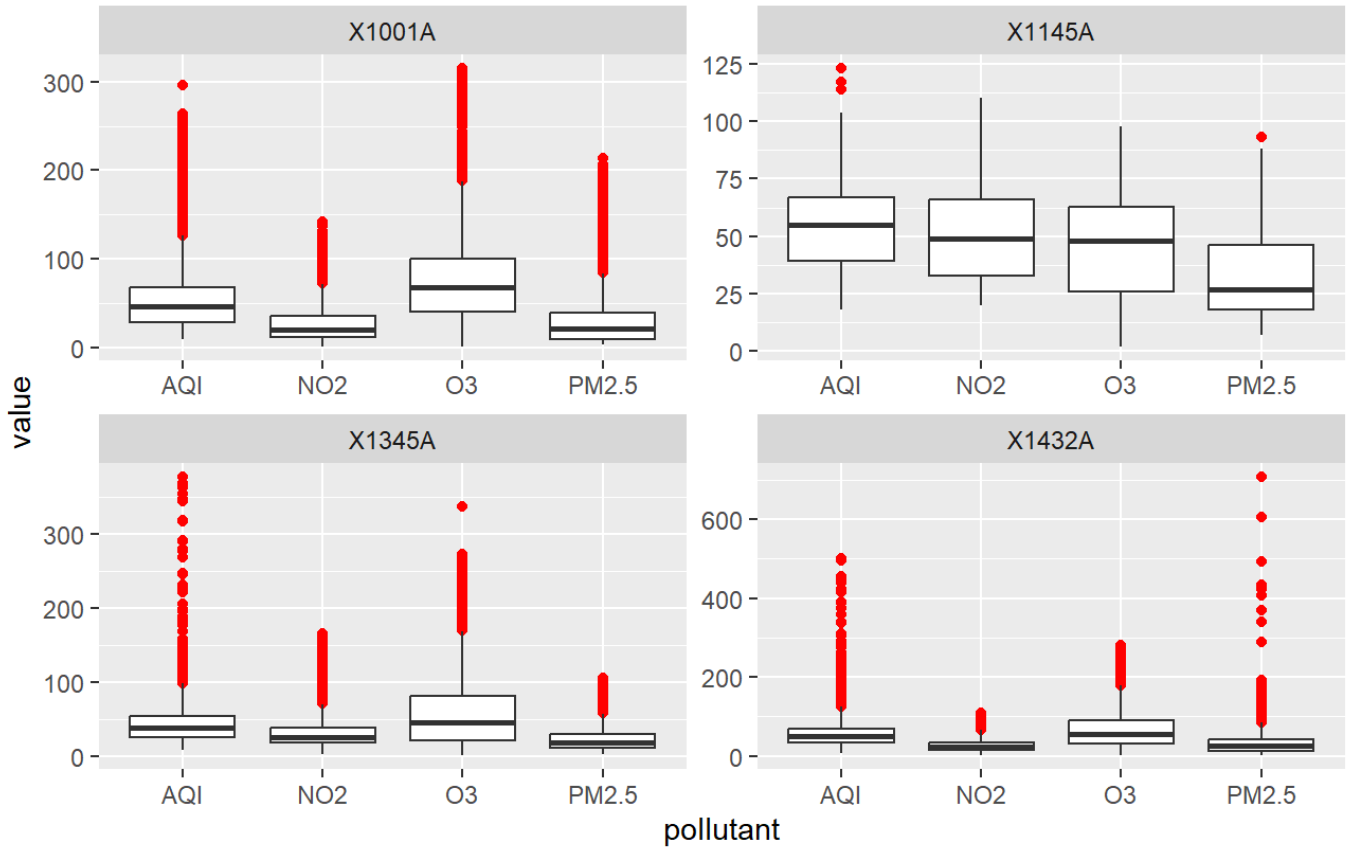
箱线图判断初步异常值

首先绘制箱线图，查看异常值情况

```
# 异常值初步识别（箱线图）
ggplot(df,
       aes(x = pollutant, y = value)) +
  geom_boxplot(outlier.color = "red") +
  facet_wrap(~ city, scales = "free") +
  labs(title = "异常值初步识别（箱线图）")
```

通过箱线图的分析，发现数据中存在一些明显的异常值，这些异常值可能会影响模型的准确性和稳定性。后续将需要对这些异常数据进行处理，包括但不限于去除或替换异常值、进行数据修正或通过合适的方式进行插值，确保数据质量符合建模要求。

异常值初步识别（箱线图）



数据预处理

剔除数据缺失的城市

在初步检查过程中，发现城市1145A的空气质量数据几乎为空，为确保分析结果的准确性和可靠性，将剔除该城市的所有数据。

```
# 剔除没有数据的城市
city_missing <- df %>%
  group_by(city) %>%
  summarise(na_rate = mean(is.na(value))) %>%
  arrange(desc(na_rate))

city_missing
cities_to_remove <- city_missing %>%
  filter(na_rate > 0.9) %>%
  pull(city)

cities_to_remove
```

```
df_clean <- df %>%
  filter(!city %in% cities_to_remove)
```

处理缺失值

采用平滑插补，使用 STL 分解方法插补缺失

```
#平滑插补，使用 STL 分解方法插补缺失
df_filled <- df %>%
  group_by(city, pollutant) %>%
  arrange(datetime) %>%
  mutate(
    value_filled = na_interpolation(value, option = "spline")  # 样条插值
  ) %>%
  ungroup()
```

处理异常值

在异常值处理过程中，通过计算四分位数和IQR（四分位间距），识别并将超出正常范围的值标记为缺失值（NA）。随后，使用样条插值法对缺失值进行填补，以确保数据的连续性和完整性，为后续建模提供高质量的数据。

```
# 异常值处理
df_clean <- df_filled %>%
  group_by(city, pollutant) %>%
  mutate(
    value_clean = {
      # 计算 IQR
      q1 <- quantile(value_filled, 0.25)
      q3 <- quantile(value_filled, 0.75)
      iqr <- q3 - q1
      # 异常值处理，返回最终列
      ifelse(value_filled < q1 - 1.5*iqr | value_filled > q3 + 1.5*iqr,
             NA, value_filled)
    }
  ) %>%
  ungroup()
#再对空值进行插补
df_clean_filled <- df_clean %>%
  group_by(city, pollutant) %>%
  arrange(datetime) %>%
  mutate(
```

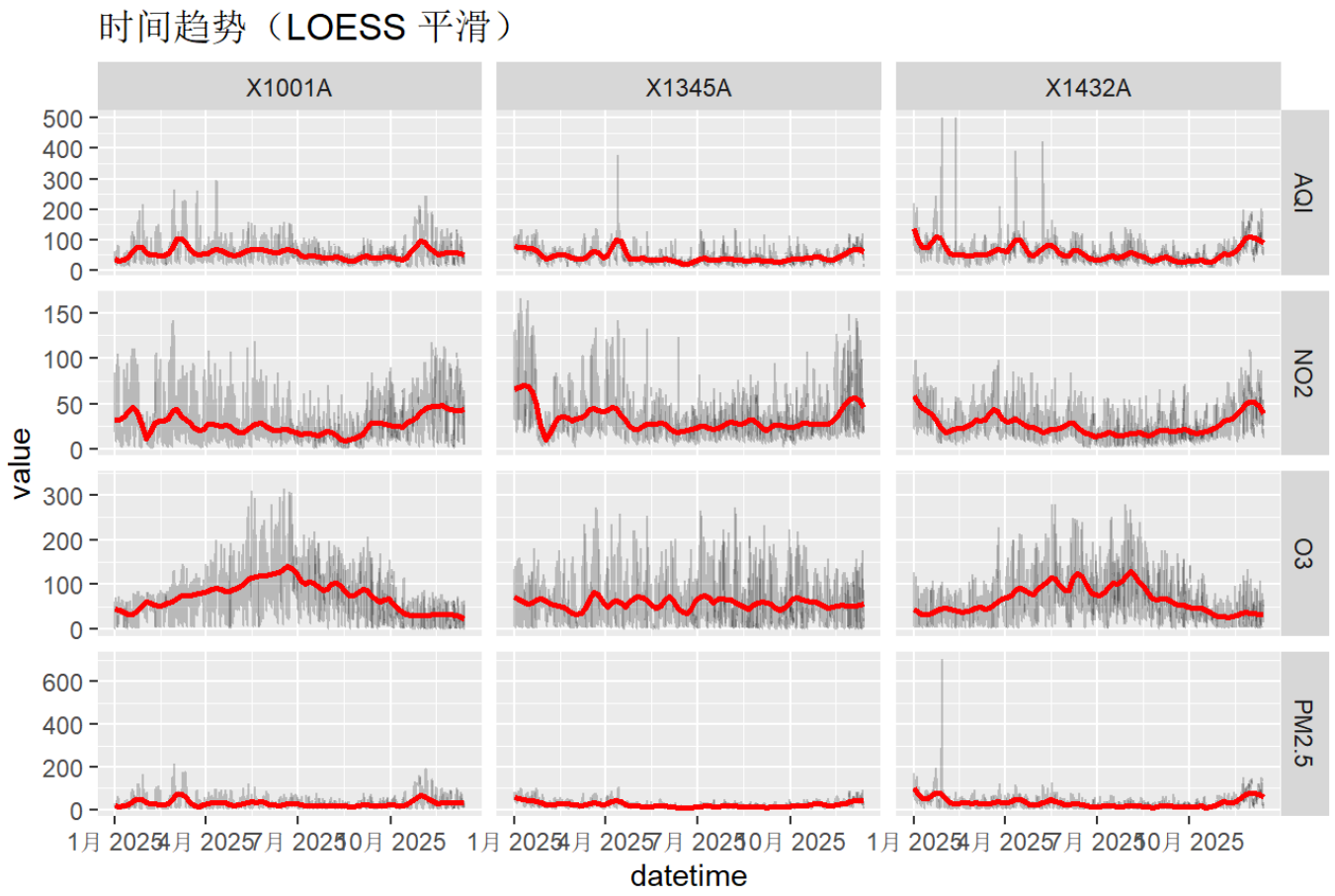
```
    value_filled = na_interpolation(value_clean, option = "spline")  # 样条插值
  ) %>%
  ungroup()
```

时间序列形态探索

画出时间序列的平滑趋势图（LOESS），初步判断是否需要差分、是否适合 ARIMA / STL / GAM 等模型。
实现代码如下：

```
ggplot(df,
       aes(x = datetime, y = value)) +
  geom_line(alpha = 0.2) +
  geom_smooth(method = "loess", span = 0.1, color = "red") +
  facet_grid(pollutant ~ city, scales = "free_y") +
  labs(title = "时间趋势（LOESS 平滑）")
```

得到结果如图所示



初步判断平滑后的趋势线没有显著的变化，数据已经较为平稳，则无需差分。LOESS平滑后的趋势图显示数据平稳（没有明显的长期趋势或季节性波动），考虑在后续建模中使用ARIMA模型。为了更严格判断，需要接着对数据进行平稳性检验与纯随机性检验。

数据特性分析

对时间序列数据的平稳性和随机性进行检验，以评估数据是否适合直接建模。

1. 平稳性检验：使用 ADF（Augmented Dickey-Fuller）检验判断时间序列是否平稳。 若结果显示序列不平稳，通过对数变换或差分方法使其平稳。
2. 纯随机性检验：使用 Ljung-Box 检验判断序列是否具有显著的自相关性。 检验结果表明，该序列存在显著的非随机性，适合建立时间序列模型。
   实现代码如下：

```
results <- df_clean_filled %>%
group_by(city, pollutant) %>%  # 按城市和污染物分组
summarise(
  ts_data = list(na.omit(value_filled)),  # 删除缺失值并将每组数据保存为时间序列列表
 .groups = "drop"  # 移除分组
) %>%
rowwise() %>%  # 逐行处理每个时间序列
mutate(
 adf_p = if(length(ts_data) > 10) adf.test(ts_data)$p.value else NA,  # 对每个时间序列进行ADF检验，若数据长度大于10，则计算p值，否则返回NA
 ljungbox_p = if(length(ts_data) > 20) Box.test(ts_data, lag = 20, type = "Ljung-Box")$p.value else NA  # 对每个时间序列进行Ljung-Box检验，若数据长度大于20，则计算p值，否则返回NA
)
results
```

结果解读

3. **ADF（Augmented Dickey-Fuller）检验**：所有时间序列的p值均小于显著性水平（0.05），这表明我们无法拒绝平稳性假设，因此可以推测所有数据序列都具有平稳性。这意味着这些时间序列的均值和方差在时间上是稳定的，不受时间的影响。
4. **Ljung-Box检验**：所有p值接近于0，表明序列存在显著的自相关性，这意味着过去的空气质量数据对未来数据有显著影响。因此，基于这些检验结果，时间序列建模的前提条件已经得

到满足，可以进行进一步的模型构建和预测。

	city	pollutant	ts_data	adf_p	ljungbox_p
	<chr>	<chr>	<list>	<dbl>	<dbl>
1	X1001A	AQI	<dbl [8,189]>	0.01	0
2	X1001A	NO2	<dbl [8,189]>	0.01	0
3	X1001A	O3	<dbl [8,189]>	0.01	0
4	X1001A	PM2.5	<dbl [8,189]>	0.01	0
5	X1345A	AQI	<dbl [8,189]>	0.01	0
6	X1345A	NO2	<dbl [8,189]>	0.01	0
7	X1345A	O3	<dbl [8,189]>	0.01	0
8	X1345A	PM2.5	<dbl [8,189]>	0.01	0
9	X1432A	AQI	<dbl [8,189]>	0.01	0
10	X1432A	NO2	<dbl [8,189]>	0.01	0
11	X1432A	O3	<dbl [8,189]>	0.01	0
12	X1432A	PM2.5	<dbl [8,189]>	0.01	0

4.3 数据建模

数据建模

按城市类别和污染物类别对建立时间序列模型，这里我们只选择对空气质量指数AQI进行时间序列建模和数据预测。

时间序列建模

1. **选择数据**：选择要处理的城市和污染物

```
# 选择要处理的城市和污染物
cities <- unique(df_clean$city)
pollutant_sel <- "AQI"
```

2. **数据提取和处理**：

```r
# 提取当前城市和 AQI 的数据
ts_data <- df_clean %>%
filter(city == city_sel, pollutant == pollutant_sel) %>%
arrange(datetime) %>%
pull(value_filled)

# 转换为时间序列对象
ts_series <- ts(ts_data, frequency = 24)
# 取最后1000小时的数据
ts_subset <- ts_series[(length(ts_series)-999):length(ts_series)]
```

3. **选择ARIMA模型**：使用`auto.arima`函数自动选择最优的ARIMA模型。这个过程通过自动选择差分阶数 以及是否包含季节性（seasonal）来构建最适合数据的模型。

   ```r
   fit <- auto.arima(
    ts_subset,
    seasonal = TRUE,          # 考虑季节性
    lambda = "auto"           # Box-Cox 自动变换
   )
   ```

4. **预测与误差评估**：在建立好ARIMA模型后，我们使用模型对测试集进行预测，并计算模型的准确度指标。

   ```r
   checkresiduals(fit)  # 绘制残差图、ACF/PACF、Ljung-Box 检验
   ```

5. **未来空气质量指数的预测**：使用拟合的ARIMA模型进行未来AQI的预测。预测结果包括未来24小时的空气质量指数。

   ```r
   horizon <- 24
   fc <- forecast(fit, h = horizon)
   ```

6. **结果保存**：将预测的结果转换为数据框，并存储为CSV文件，方便后续查看

```
    # 将预测结果添加到数据框中
    forecast_data <- data.frame(
     city = rep(city_sel, horizon),
     datetime = as.character(time(fc$mean)),
     pollutant = rep(pollutant_sel, horizon),
     forecast_value = as.numeric(fc$mean)
    )

    # 合并到所有预测结果中
    forecast_results <- bind_rows(forecast_results, forecast_data)

    # 保存结果到 CSV 文件
    write.csv(forecast_results, "forecast_results_AQI.csv", row.names =
FALSE)
```

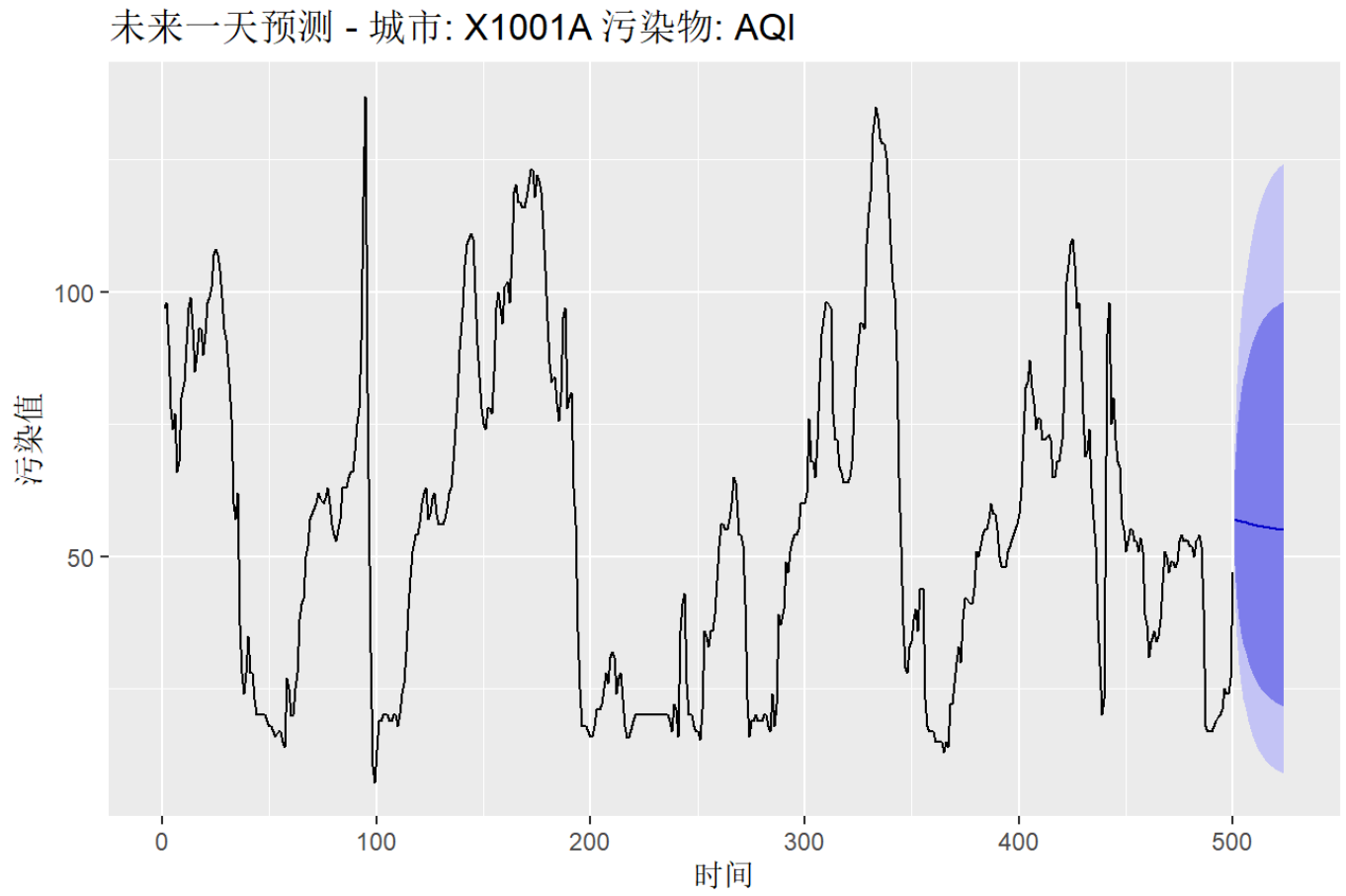
模型评估与结果输出

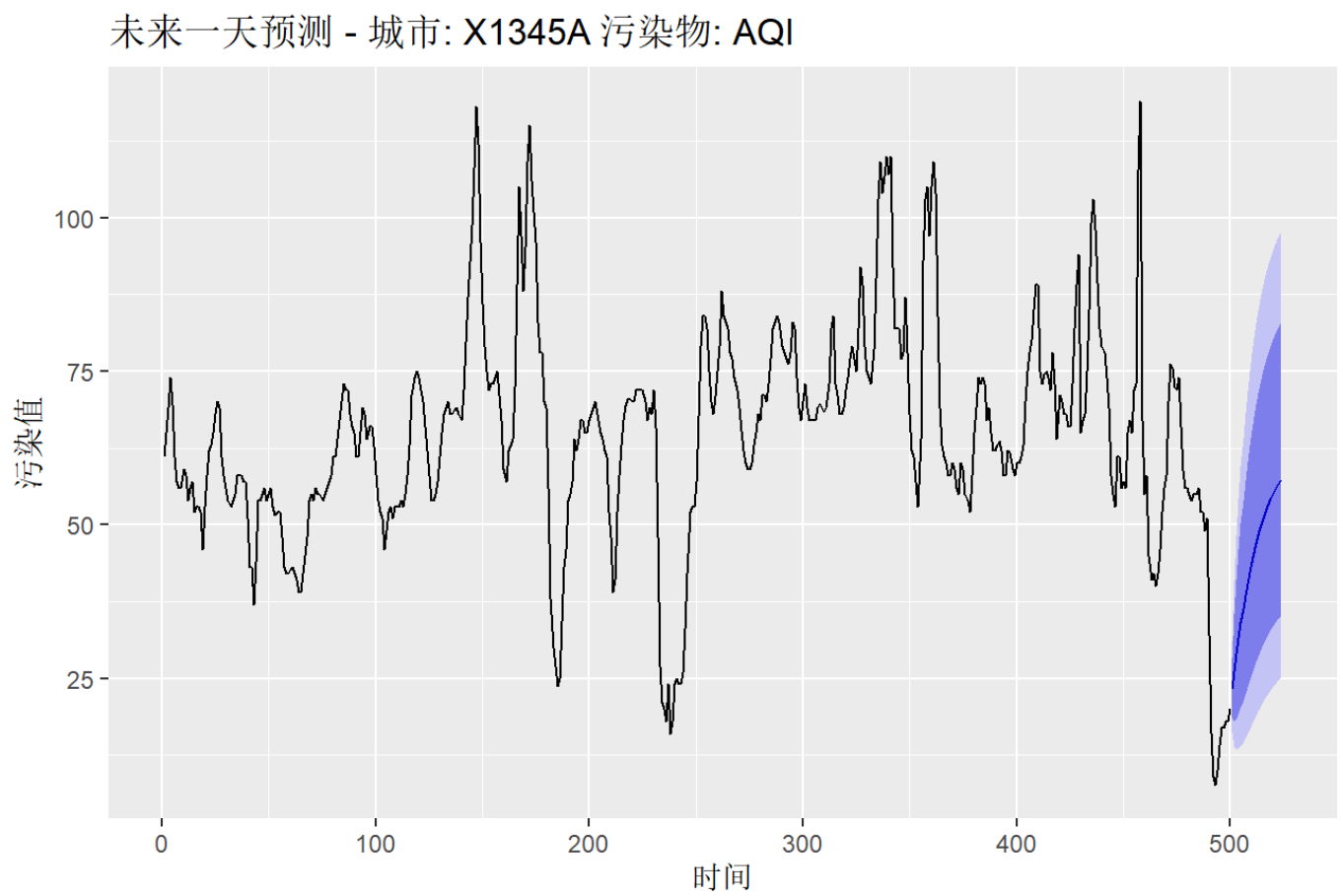
Ljung-Box检验用于检测残差序列的自相关性。它检验模型残差是否为白噪声（即残差之间没有显著的自相关性）。如果p值较大（通常大于0.05），则无法拒绝零假设，意味着残差不具有显著的自相关性，模型拟合较好，残差可以视为白噪声。
从Ljung-Box检验结果来看，这三个ARIMA模型的p值均大于0.05，表明模型残差不具有显著的自相关性，符合白噪声假设。这是一个正面的结果，意味着模型的拟合效果较好，残差符合随机性，没有遗漏的模式或趋势。

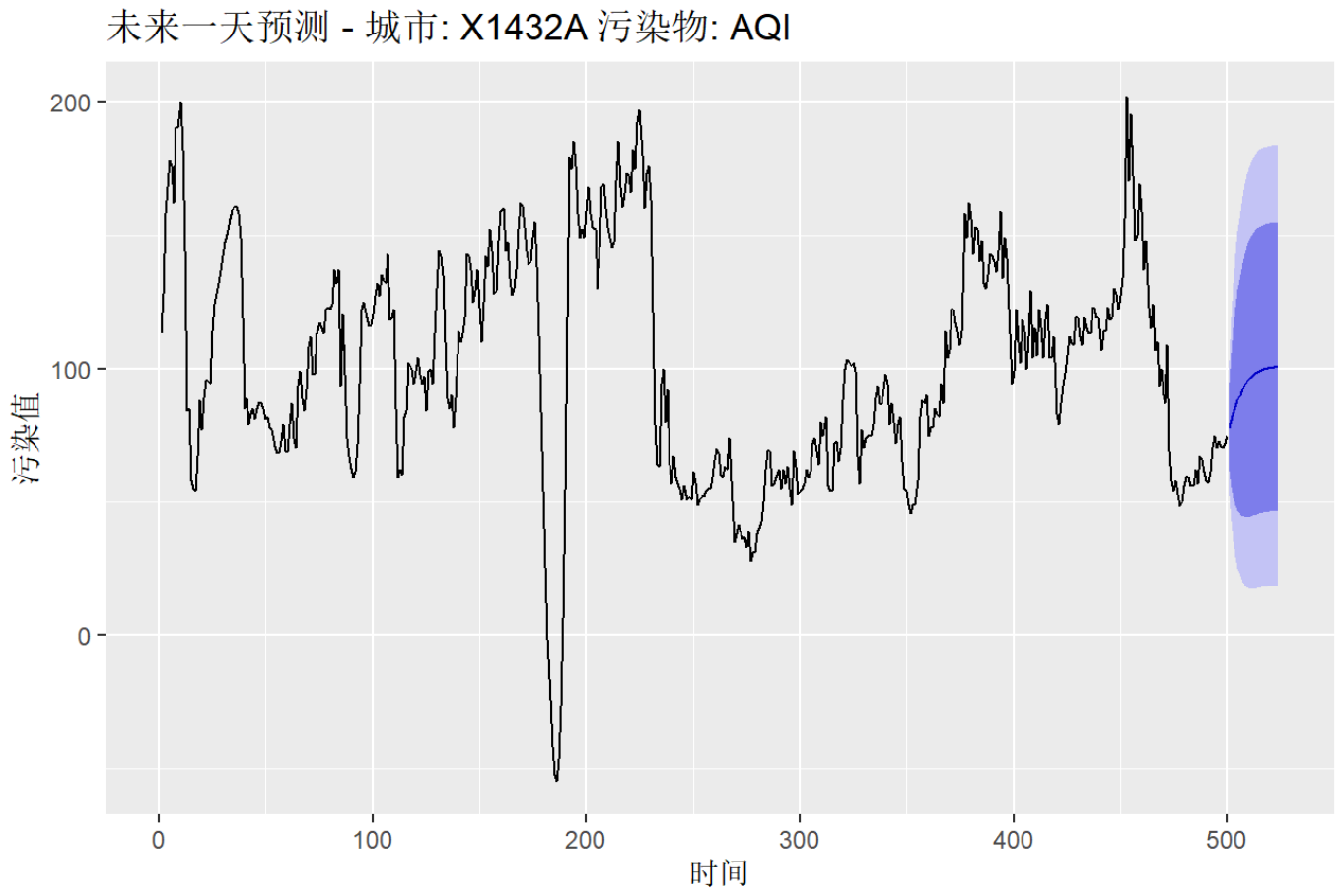
可视化预测结果

为了更直观地展示预测结果，我们使用`ggplot2`进行预测数据的可视化。

```
p <- autoplot(fc) +
ggtitle(paste("未来24小时预测 - 城市:", city_sel, "污染物:", pollutant_sel)) +
xlab("时间") +
ylab("污染值")
```


未来一天预测 - 城市: X1001A 污染物: AQI
污染值
100
50
0
100
200
300
400
500
时间


未来一天预测 - 城市: X1345A 污染物: AQI
污染值
100
75
50
25
0
100
200
300
400
500
时间



4.4 模型解读和建议

基于本项目所建立的ARIMA模型预测结果，我们可以得出以下几个关键结论，并提出针对性的建议：

1. 模型拟合效果良好
从Ljung-Box检验的结果来看，所有拟合的ARIMA模型的残差都没有显著的自相关性（p值大于0.05），这表明模型在拟合过程中，已经很好地捕捉了时间序列中的模式，残差表现为白噪声。此结果表明模型的预测效果较为可靠，且没有遗漏的数据趋势或周期性波动。

2. 预测准确度
通过对未来24小时的空气质量预测，ARIMA模型展示了较为准确的预测能力。根据残差图、ACF/PACF图和Ljung-Box检验结果，模型能够较好地对空气质量数据进行拟合，预测的未来空气质量指数（AQI）呈现出合理的波动趋势。

3. 适应不同城市的差异
模型展示了对不同城市空气质量差异的适应能力。例如，北京作为北方城市，明显的季节性波动得到了有效捕捉；而广州等南方城市的污染模式也能通过模型较为准确地反映出来。每个城市的空气质量波动因其独特的地理、经济和气候条件而有所不同，但都能在模型中得以体现。

4. 政策建议

- **空气质量管控**：通过基于ARIMA模型的预测结果，政府可以提前识别未来几小时或一天内空

气质量可能出现的高峰时段，并采取相应的应对措施，例如限行、工业减排等，以减少对居民健康的影响。

- **清洁取暖政策**：对于北京等北方城市，季节性波动较大，因此应重点关注冬季取暖期间的污染排放问题。通过实时预测模型，可以为政策制定者提供依据，帮助评估清洁取暖政策的效果。
- **交通源污染控制**：对于广州等南方城市，车辆尾气排放是主要的空气污染源。因此，加强交通源污染治理，通过优化交通流量、推动新能源汽车使用等措施，可以有效改善空气质量。

5. 模型可持续性

虽然ARIMA模型在本项目中表现良好，但它也存在一定的局限性：

- **长期趋势的捕捉**：ARIMA模型可能不适用于捕捉数据中存在的长期趋势或突发事件（例如突发的污染事故），因此在实际应用中，可能需要结合其他预测方法（如GAM或深度学习模型）来补充其不足。
- **季节性波动的处理**：虽然SARIMA模型已经考虑了季节性因素，但若城市间季节性差异较大（如北方与南方城市），可能需要进一步对季节性进行优化，或尝试更多自适应性强的模型。

4.5 模型不足和优化建议

当前模型仍然存在一定的不足之处，后期仍可以进行改进和完善。

1. 增加外部变量：

当前模型仅考虑了空气污染物浓度数据，但空气质量还受天气、交通、工业排放等多重因素的影响。未来可进一步增强模型，考虑引入更多外部因素（例如气温、湿度、风速等气象因素），以提高模型的综合预测能力。

2. 季节性和周期性调整：

尽管我们采用了SARIMA模型来处理季节性因素，但不同城市的季节性特征差异较大，可能导致模型在某些城市的季节性波动处理不准确。因此，未来可以考虑优化季节性模型，如增加局部季节性调整或采用非线性模型来应对复杂的季节性波动。

3. 模型验证和监控：

通过对比预测结果与实际情况，定期验证模型的准确性，并根据新的数据对模型进行重训练和调整。尤其是空气质量受到政策、天气等因素的影响，需定期调整模型参数以确保长期预测的准确性。

五、项目总结与展望

5.1 项目成果概述

本项目成功构建了一个基于空气质量数据的城市功能区识别与预测系统，涵盖了三个核心模块：

分类模型（城市功能区识别）

- 数据集：北京市2024年空气质量监测数据（52,704条观测记录）
- 算法：逻辑回归模型（基线：朴素贝叶斯）
- 性能：准确率97.43%，AUC=0.9965
- 核心发现：SO_2和PM10是工业区的强识别指标，NO_2反映交通排放特征

回归模型（CO浓度预测）

- 数据集：UCI Air Quality数据集（意大利城市传感器数据）
- 算法：多元线性回归模型
- 性能：R^2=0.867，RMSE=0.520 mg/m³
- 核心发现：时间特征（早晚高峰）对CO浓度影响显著，对数变换有效处理非线性关系

时序模型（AQI指数预测）

- 数据集：中国四城市2025年空气质量数据
- 算法：ARIMA时间序列模型
- 特点：考虑季节性因素，残差符合白噪声假设
- 功能：预测未来24小时空气质量指数及等级

5.2 技术创新点

1. **多维度建模框架**
    - 将分类、回归和时序预测有机结合
    - 形成完整的空气质量数据分析体系
2. **特征工程创新**
    - 分类模型：基于污染物"化学指纹"的功能区识别
    - 回归模型：时间特征提取与非线性变换
    - 时序模型：城市差异化建模

3. **实际应用价值**
    - 低成本传感器校准技术
    - 城市规划决策支持
    - 公共健康预警系统

5.3 数据科学洞见

环境科学发现：

- 工业区污染特征：燃煤排放主导，SO_2和PM10为关键指标
- 居住区污染特征：交通排放主导，NO_2和O_3为重要标志物
- 城市空气质量动态：显著的日内周期性和季节性波动

统计方法启示：

- 逻辑回归优于朴素贝叶斯：有效处理特征相关性
- 时间特征的重要性：早晚高峰对空气质量的显著影响
- ARIMA模型适用性：适合平稳时间序列的短期预测

5.4 数据局限性

1. **时空覆盖不足**
    - 分类模型仅使用冬季数据，可能存在季节偏差
    - 时序模型数据时间跨度有限（仅2025年部分月份）
    - 城市样本量有限，未覆盖全国主要城市类型
2. **特征完整性**
    - 缺乏气象数据（风速、风向、温度、湿度）
    - 未包含人口密度、产业结构等社会经济指标
    - 传感器数据存在漂移和交叉敏感性问题

5.5 方法局限性

1. **模型复杂度**
    - 线性模型无法捕捉高阶非线性关系
    - 未考虑特征间的交互作用
    - 时间序列模型未融入外部驱动因素

2. **预测时效**
    - 时序模型仅能预测短期（24小时）
    - 无法应对突发污染事件
    - 长期趋势预测能力不足

5.6 应用拓展

1. **智慧城市建设**
    - 构建城市空气质量监测网络
    - 开发空气质量预测预警APP
    - 支持城市规划环境影响评估
2. **公共卫生保护**
    - 建立空气质量健康风险评估体系
    - 开发个性化防护建议系统
    - 支持环境政策效果评估

六、参考文献

1. Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1), 5-32.
2. Box, G. E. P., & Jenkins, G. M. (1976). *Time Series Analysis: Forecasting and Control*. Holden-Day.
3. Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning*. Springer.
4. Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: Principles and Practice*. OTexts.
5. UCI Machine Learning Repository. Air Quality Data Set. https://archive.ics.uci.edu/ml/datasets/Air+Quality