ST346 广义线性模型回归与分类作业1 - 深度解析教程

前言

本教程旨在为本科生提供一个全面而深入的指导，帮助理解并完成ST346课程的第一次作业。我们将逐步深入每个问题，详细解释相关的统计学概念、R编程技巧，以及这些知识在实际问题中的应用。

数据准备

在开始分析之前，我们需要正确加载数据。这个看似简单的步骤实际上是整个分析过程的基础。

```
load("CourseworkData1.rda")
```

详细解释：

1. **文件格式**：`.rda`是R的数据文件格式，它可以存储多个R对象。这种格式比纯文本文件（如CSV）更高效，尤其是对于大型数据集。
2. **加载过程**：`load()`函数会将文件中的所有对象加载到当前的R环境中。这意味着，如果文件中有多个数据框或其他对象，它们都会被加载。
3. **工作目录**：确保`CourseworkData1.rda`文件在你的当前工作目录中。你可以使用`getwd()`查看当前工作目录，使用`setwd()`更改工作目录。
4. **数据检查**：加载后，应该立即检查数据。使用以下命令：

   ```
   str(cleaners)
   str(mushrooms)
   ```

   这会显示每个数据框的结构，包括变量名、类型和前几个值。

5. **潜在问题**：如果加载失败，可能的原因包括文件路径错误、文件损坏，或R版本不兼容。确保你使用的R版本与创建文件时的版本兼容。

为什么这一步很重要：

- **数据完整性**：确保你分析的是正确、完整的数据集。
- **变量理解**：了解每个变量的类型和结构，这对后续分析至关重要。
- **内存管理**：大型数据集可能需要考虑内存使用情况。
- **reproducibility**：记录数据加载步骤是实现研究可重复性的第一步。

问题1：加权回归

这个问题围绕一个清洁公司的案例展开，旨在探索团队规模与清洁效率之间的关系。这是一个典型的回归分析问题，但引入了加权回归的概念来处理可能存在的异方差性。

(a) 数据可视化

深入解析：

数据可视化是任何数据分析的第一步。它允许我们直观地理解数据的分布和变量之间的关系，often揭示可能被纯数字分析忽视的模式。

```
library(ggplot2)

plot <- ggplot(cleaners, aes(x = Staff, y = Offices)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "Relationship between Team Size and Offices Cl
eaned",
       x = "Number of Staff",
       y = "Number of Offices Cleaned") +
  theme_minimal()

print(plot)

correlation <- cor(cleaners$Staff, cleaners$Offices)
```

```
print(paste("Correlation coefficient:", round(correlation, 3)))
```

代码解析：

1. `library(ggplot2)`: 加载ggplot2包。ggplot2是一个强大的图形包，基于图形语法理论。
2. `ggplot(cleaners, aes(x = Staff, y = Offices))`:
   - 创建ggplot对象。
   - `cleaners`是数据源。
   - `aes()`定义美学映射，这里我们将Staff映射到x轴，Offices映射到y轴。
3. `geom_point()`: 添加散点图层。每个点代表一个观察值。
4. `geom_smooth(method = "lm", se = FALSE, color = "red")`:
   - 添加平滑层，这里使用线性回归（`method = "lm"`）。
   - `se = FALSE`不显示标准误差带。
   - `color = "red"`设置线条颜色为红色。
5. `labs()`: 添加图表标题和轴标签。
6. `theme_minimal()`: 应用简洁的主题样式。
7. `cor(cleaners$Staff, cleaners$Offices)`: 计算相关系数。

统计学原理：

1. **散点图**：
   - 用途：显示两个连续变量之间的关系。
   - 解释：点的分布模式可以揭示关系的性质（线性、非线性、强弱等）。
   - 注意：散点图不能证明因果关系，只能显示相关性。
2. **线性回归线**：
   - 原理：最小二乘法，最小化残差平方和。
   - 解释：线的斜率表示平均关系，截距表示当x=0时y的预测值。
   - 局限性：假设关系是线性的，可能掩盖非线性模式。

3. **相关系数**：
    - 范围：-1到1。
    - 解释：
        - 1表示完全正相关
        - 1表示完全负相关
        - 0表示无线性相关
    - 局限性：只衡量线性关系，对非线性关系不敏感。

如何回答这个问题：

1. 描述散点图的模式：
    - 点是否呈现明显的趋势？
    - 是否有离群点？
    - 数据分布是否均匀？
2. 解释回归线：
    - 线的斜率是正还是负？这意味着什么？
    - 线是否很好地拟合了数据点？
3. 解释相关系数：
    - 系数的大小表明关系强度如何？
    - 正负号表明什么？
4. 结合以上观察，讨论假设线性关系的合理性：
    - 数据是否显示明显的非线性模式？
    - 是否有其他因素可能影响这个关系？
5. 考虑实际意义：
    - 这种关系在清洁公司的实际运营中意味着什么？
    - 是否有其他因素（如办公室大小、清洁标准）可能影响这个关系？

实际应用示例：

假设散点图显示明显的正相关，相关系数为0.85。你的回答可能是：

"散点图显示了员工数量和清洁办公室数量之间的明显正相关关系。随着团队规模增加，清洁的办公室数量也总体上增加。线性回归线很好地拟合了大部分数据点，表明线性关系是一个合理的假设。相关系数为0.85，表明两个变量之间存在强烈的正相关。这意味着，总的来说，更大的清洁团队能清洁更多的办公室。"

(b) 线性模型拟合

深入解析：

线性回归是统计学中最基本也是最常用的模型之一。它试图通过一个线性方程来描述因变量（Y）和自变量（X）之间的关系。在这个问题中，我们用它来量化团队规模和清洁效率之间的关系。

```
model <- lm(Offices ~ Staff, data = cleaners)
summary(model)

slope <- coef(model)["Staff"]
print(paste("Slope coefficient:", round(slope, 3)))
```

代码解析：

1. `lm(Offices ~ Staff, data = cleaners)`：
   - `lm()`函数用于拟合线性模型。
   - `Offices ~ Staff`是公式，表示Offices是因变量，Staff是自变量。
   - `data = cleaners`指定数据源。
2. `summary(model)`：
   - 提供模型的详细统计信息，包括系数估计、标准误差、t值、p值、R平方值等。
3. `coef(model)["Staff"]`：
   - `coef()`函数提取模型系数。
   - `["Staff"]`选择Staff变量的系数，即斜率。

统计学原理：

1. **线性回归模型**：
   - 公式：$Y = \beta 0 + \beta 1X + \varepsilon$

- Y：因变量（这里是Offices）
- X：自变量（这里是Staff）
- β0：截距
- β1：斜率
- ε：误差项

2. **最小二乘法**：
   - 原理：最小化残差平方和。
   - 目的：找到最佳拟合线。
3. **模型解释**：
   - 截距（β0）：当X=0时Y的预测值。
   - 斜率（β1）：X每增加一个单位，Y平均增加的量。
4. **统计显著性**：
   - p值：系数等于0的概率。通常p < 0.05被认为是统计显著的。
   - t值：系数除以其标准误差，用于检验系数是否显著不同于0。
5. **模型拟合优度**：
   - R平方：解释的方差比例，范围0到1。
   - 调整后的R平方：考虑了模型复杂度的R平方。

如何回答这个问题：

1. 解释模型摘要：
   - 报告并解释截距和斜率系数。
   - 讨论这些系数的统计显著性（p值）。
   - 解释R平方值。
2. 重点解释斜率系数：
   - 给出具体的解释，例如"每增加一名员工，平均多清洁X个办公室"。
   - 讨论这个系数的实际意义。
3. 评估模型整体：

- 模型是否统计显著（F统计量和其p值）？
- R平方值表明模型解释了多少变异？

4. 考虑实际应用：
   - 这个模型对公司决策有何启示？
   - 模型的局限性是什么？

实际应用示例：

假设模型输出显示斜率为3.5，p值小于0.001，R平方为0.8。你的回答可能是：

"线性回归模型显示，员工数量（Staff）对清洁的办公室数量（Offices）有显著影响。斜率系数为3.5（$p < 0.001$），这意味着平均而言，每增加一名员工，预计每天能多清洁3.5个办公室。这个关系在统计上高度显著。

模型的R平方值为0.8，表明团队规模解释了80%的办公室清洁数量的变异。这个较高的R平方值表明模型具有良好的解释力。

这种回答不仅解释了统计结果，还讨论了其实际意义和局限性，展示了深入的理解和批判性思维。

(c) 残差分析

```
plot_data <- data.frame(
  Fitted = fitted(model),
  Residuals = residuals(model)
)

residual_plot <- ggplot(plot_data, aes(x = Fitted, y = Residu
als)) +
  geom_point() +
  geom_hline(yintercept = 0, linetype = "dashed", color = "re
d") +
  labs(title = "Residuals vs Fitted Values",
       x = "Fitted Values",
       y = "Residuals") +
  theme_minimal()
```

```
print(residual_plot)
```

代码解析：

1. `fitted(model)` 和 `residuals(model)`：
   - `fitted()` 函数返回模型的预测值。
   - `residuals()` 函数返回残差（观察值减去预测值）。
2. `data.frame()`：创建一个新的数据框，包含拟合值和残差。
3. `ggplot()` 函数创建残差图：
   - `aes(x = Fitted, y = Residuals)`：将拟合值映射到x轴，残差映射到y轴。
   - `geom_point()`：添加散点。
   - `geom_hline(yintercept = 0)`：添加y=0的参考线。
   - `labs()`：添加图表标题和轴标签。
   - `theme_minimal()`：应用简洁的主题样式。

统计学原理：

1. **残差定义**：
   残差 = 观察值 - 预测值
   它代表了模型未能解释的变异。
2. **残差图的目的**：
   - 检查线性性假设
   - 检查同方差性假设
   - 识别异常值和高影响点
   - 检测模型的系统性偏差
3. **理想的残差图特征**：
   - 残差随机分布在y=0线周围
   - 没有明显的模式或趋势
   - 残差的分散程度大致相同（同方差性）

4. **常见问题及其表现**：
    - 非线性关系：残差图呈现曲线模式
    - 异方差性：残差的分散程度随拟合值变化（如漏斗形）
    - 异常值：极端的残差点
    - 系统性偏差：残差一致地高估或低估某些范围的值

如何回答这个问题：

1. 描述残差的整体分布：
    - 残差是否大致对称分布在y=0线周围？
    - 是否存在明显的模式或趋势？
2. 评估同方差性假设：
    - 残差的分散程度是否随拟合值变化？
    - 如果存在异方差性，描述其模式（如漏斗形）。
3. 识别潜在的异常值或高影响点：
    - 是否有极端的残差点？
    - 这些点对应的观察值有什么特殊性？
4. 讨论线性假设：
    - 残差图是否暗示了非线性关系？
    - 如果是，描述你观察到的非线性模式。
5. 总结模型适当性：
    - 基于残差分析，评估模型的整体适当性。
    - 提出可能的改进建议（如需要）。

实际应用示例：

假设残差图显示了一个轻微的漏斗形模式，残差的分散程度随拟合值增加而增加。你的回答可能是：

"残差图显示了几个重要的特征：

1. 整体分布：残差大致对称分布在y=0线周围，这是一个好迹象，表明模型没有系统性地高估或低估预测值。
2. 异方差性：然而，我们观察到一个轻微的漏斗形模式。随着拟合值增加，残差的分散程度也增加。这表明存在异方差性，违反了普通最小二乘法的同方差性假设。这意味着模型对较大团队的预测可能不如对较小团队的预测那么精确。
3. 线性性：没有观察到明显的曲线模式，这支持了线性关系的假设。
4. 异常值：在图的右上角有几个点的残差相对较大。这些可能代表特殊情况，如特别高效的大型团队，值得进一步调查。
5. 模型适当性：尽管存在异方差性，但整体而言，残差没有显示强烈的系统性模式，这表明线性模型仍然是一个合理的选择。

(d) 方差估计

深入解析：

方差估计是理解数据变异性的关键步骤，特别是在考虑加权回归时。这一步骤旨在量化不同团队规模下办公室清洁数量的方差，为后续的加权回归分析做准备。

```
variance_table <- aggregate(Offices ~ Staff, data = cleaners,
FUN = var)
names(variance_table) <- c("Team_size", "Variance")
variance_table$Variance <- round(variance_table$Variance, 2)
print(variance_table)

# 验证结果
given_values <- data.frame(
  Team_size = c(2, 4, 6, 8, 10, 12, 16),
  Given_Variance = c(9.00, 24.67, 22.00, 44.12, 62.84, 53.14,
144.01)
)
comparison <- merge(variance_table, given_values, by = "Team_
size", all = TRUE)
comparison$Difference <- round(comparison$Variance - comparis
on$Given_Variance, 2)
print(comparison)
```

代码解析：

1. `aggregate(Offices ~ Staff, data = cleaners, FUN = var)`：
   - `aggregate()` 函数用于按组计算统计量。
   - `Offices ~ Staff` 指定按Staff分组计算Offices的统计量。
   - `FUN = var` 指定计算方差。
2. `names(variance_table) <- c("Team_size", "Variance")`：
   重命名列，使输出更易读。
3. `round(variance_table$Variance, 2)`：
   将方差四舍五入到两位小数。
4. `merge(variance_table, given_values, by = "Team_size", all = TRUE)`：
   合并计算的方差和给定的方差，用于比较。

统计学原理：

1. **方差**：
   - 定义：衡量数据点与均值偏离程度的平均值。
   - 公式：Var(X) = E[(X - μ)^2]，其中μ是均值。
   - 意义：方差越大，数据的离散程度越高。
2. **分组方差**：
   - 目的：了解不同组（这里是不同团队规模）内部的变异性。
   - 意义：可以揭示异方差性，即变异性随自变量（团队规模）变化的情况。
3. **异方差性**：
   - 定义：误差项的方差不恒定，而是随预测变量变化。
   - 影响：违反了普通最小二乘法的假设，可能导致估计效率降低和推断不准确。
   - 解决方法：加权最小二乘法是常用的处理方法之一。

如何回答这个问题：

1. 解释计算的方差表：
   - 描述不同团队规模的方差。

- 讨论是否观察到方差随团队规模变化的模式。

2. 比较计算结果和给定值：
   - 验证计算是否准确。
   - 如果有差异，讨论可能的原因（如舍入误差）。
3. 讨论观察到的异方差性模式：
   - 方差是否随团队规模增加而增加？
   - 这种模式对回归分析有何影响？
4. 考虑实际含义：
   - 不同团队规模的方差差异意味着什么？
   - 这如何影响公司的决策和资源分配？

实际应用示例：

假设计算结果与给定值完全匹配，你的回答可能是：

"方差分析结果揭示了几个重要的发现：

1. 方差随团队规模变化：
   观察到方差generally随着团队规模的增加而增加。例如，2人团队的方差为9.00，而16人团队的方差高达144.01。这明显表明存在异方差性。
2. 非线性增长：
   方差的增长并不完全线性。例如，从10人团队（方差62.84）到12人团队（方差53.14）时，方差略有下降，但之后又显著增加。
3. 计算验证：
   我们的计算结果与给定值完全匹配，差异均为0，证实了计算的准确性。

(e) 权重计算

深入解析：

权重计算是实施加权回归的关键步骤。这个过程旨在处理我们在前面步骤中观察到的异方差性问题。通过适当的加权，我们可以提高回归模型的效率和准确性。

```
base_variance <- variance_table$Variance[variance_table$Team_
size == 2]
```

```
variance_table$Weight <- base_variance / variance_table$Varia
nce
print(variance_table)

assign_weight <- function(staff_size) {
  weight <- variance_table$Weight[variance_table$Team_size ==
staff_size]
  return(weight)
}

cleaners$Weight <- sapply(cleaners$Staff, assign_weight)
head(cleaners)
```

代码解析：

1. `base_variance <- variance_table$Variance[variance_table$Team_size == 2]`:
   选择团队规模为2的方差作为基准。
2. `variance_table$Weight <- base_variance / variance_table$Variance`:
   计算每个团队规模的权重，基准方差除以各自的方差。
3. `assign_weight` 函数:
   - 这是一个自定义函数，用于为每个观察分配权重。
   - 它根据团队规模查找对应的权重。
4. `sapply(cleaners$Staff, assign_weight)`:
   使用
   `sapply()` 函数将 `assign_weight` 函数应用于每个 Staff 值。

统计学原理：

1. **加权最小二乘法（WLS）的原理**:
   - 目的：处理异方差性问题。
   - 原理：给予方差小的观察更大的权重，方差大的观察较小的权重。
   - 数学表达：最小化 Σ w_i (y_i - (β_0 + β_1 x_i))^2，其中 w_i 是权重。
2. **权重的选择**:

- 理论上最优的权重是误差方差的倒数。
- 在实践中，我们通常不知道真实的误差方差，所以使用估计值。
- 这里我们使用观察到的组内方差的倒数作为权重。

3. **为什么使用方差的倒数**:
   - 直觉解释：方差大的观察值包含的"噪音"更多，因此应该设置较小的权重。
   - 数学解释：这种加权使得加权后的误差项具有常数方差（同方差性）。
4. **标准化权重**:
   - 选择一个基准方差（这里是团队规模为2的方差）。
   - 将所有权重标准化，使得基准组的权重为1。
   - 这种标准化不影响估计结果，但使得权重更易解释。

如何回答这个问题：

1. 解释权重计算的方法：
   - 描述如何选择基准方差。
   - 解释权重计算公式。
2. 分析计算出的权重：
   - 描述权重如何随团队规模变化。
   - 讨论这种变化的含义。
3. 解释权重分配的过程：
   - 描述如何使用自定义函数分配权重。
   - 解释为什么需要这种方法。
4. 讨论这种加权方法的优缺点：
   - 优点：处理异方差性，提高估计效率。
   - 潜在缺点：如果方差估计不准确，可能引入新的偏差。
5. 考虑实际应用：
   - 这种加权如何影响不同规模团队在分析中的重要性。
   - 讨论这对公司决策可能产生的影响。

实际应用示例：

"基于前面的方差分析，我们计算了每个团队规模的权重，以准备进行加权回归分析。这个过程涉及几个关键步骤：

1. 权重计算方法：
   我们选择团队规模为2的方差（9.00）作为基准。每个团队规模的权重计算为基准方差除以该规模的方差。例如，对于16人团队：
   权重 = 9.00 / 144.01 ≈ 0.0625
2. 权重分析：
   - 2人团队：权重为1（基准）
   - 4人团队：权重约为0.3648
   - ...
   - 16人团队：权重约为0.0625

   我们观察到权重随团队规模增加而减小，这反映了较大团队的清洁效率变异性更大。
3. 权重分配过程：
   我们创建了一个自定义函数 `assign_weight`，然后使用 `sapply` 将其应用于每个观察值。这确保了每个数据点都получ得到与其团队规模相应的权重。

(f) 加权线性模型

深入解析：

加权线性回归是处理异方差性的有效方法。通过给予不同观测不同的权重，我们可以提高模型估计的效率和准确性。

```
weighted_model <- lm(Offices ~ Staff, data = cleaners, weight
s = Weight)
summary(weighted_model)

plot_data <- data.frame(
  Fitted = fitted(weighted_model),
  Weighted_Residuals = residuals(weighted_model, type = "resp
onse") * sqrt(cleaners$Weight)
)
```

```
residual_plot <- ggplot(plot_data, aes(x = Fitted, y = Weighted_Residuals)) +
  geom_point() +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  labs(title = "Weighted Residuals vs Fitted Values",
       x = "Fitted Values",
       y = "Weighted Residuals") +
  theme_minimal()

print(residual_plot)
```

代码解析：

1. `lm(Offices ~ Staff, data = cleaners, weights = Weight)`:
   - 使用 `lm()` 函数拟合加权线性模型。
   - `weights = Weight` 参数指定了每个观察的权重。
2. `residuals(weighted_model, type = "response") * sqrt(cleaners$Weight)`:
   - 计算加权残差。
   - 乘以权重的平方根是为了standardize残差。
3. `ggplot()` 用于创建加权残差图。

统计学原理：

1. **加权最小二乘法（WLS）**:
   - 目标函数：最小化 Σ w_i (y_i - (β_0 + β_1 x_i))^2
   - w_i 是第i个观察的权重
   - 这lead到加权正规方程
2. **加权估计的性质**:
   - 无偏性：如果原始模型是正确的，加权估计仍然是无偏的。
   - 效率：在异方差情况下，WLS估计比OLS估计更有效（方差更小）。

- 一致性：在一般条件下，WLS估计是一致的。

3. **加权残差**:
   - 原始残差：e_i = y_i - ŷ_i
   - 加权残差：e_i * √w_i
   - 目的：standardize残差，使其可比
4. **诊断图的解释**:
   - 理想情况下，加权残差应随机分布在y=0线周围。
   - 任何系统性模式都可能indicate模型specification的问题。

如何回答这个问题：

1. 比较加权模型与原始模型：
   - 对比系数估计和标准误。
   - 比较R平方值和F统计量。
2. 解释加权模型的系数：
   - 给出系数的实际解释。
   - 讨论统计显著性。
3. 分析加权残差图：
   - 描述残差的分布模式。
   - 比较与原始残差图的差异。
   - 讨论是否解决了异方差性问题。
4. 评估模型的整体适当性：
   - 基于统计结果和残差分析，判断模型是否适当。
   - 讨论可能的局限性或改进空间。
5. 考虑实际应用：
   - 讨论这个加权模型如何改变我们对清洁效率的理解。
   - 提出基于这个模型的实际建议。

实际应用示例：

"加权线性回归模型的结果展示了几个重要发现：

1. 模型比较：
   加权模型的R平方值从原模型的0.857增加到0.900，indicate更好的拟合。F统计量也有所提高，suggest模型的整体显著性增强。
2. 系数解释：
   - 截距：0.809 (p = 0.471)，统计上不显著。
   - Staff系数：3.826 (p < 2e-16)，高度显著。
     这意味着，平均而言，每增加一名员工，预期清洁的办公室数量增加3.826个。
     与原模型相比，这个估计略高（原为3.701），suggest未加权模型可能低估了员工数量的影响。
3. 加权残差分析：
   加权残差图显示残差更均匀地分布在y=0线周围，没有明显的漏斗形状。这indicate加权过程有效地处理了异方差性问题。
4. 模型适当性：
   overall，加权模型似乎更适合这个数据集。它不仅提高了拟合优度，还解决了之前观察到的异方差性问题。然而，仍有一些离群点值得进一步调查。

(g) 转换模型

深入解析：

转换模型提供了另一种处理异方差性的方法，它通过变量转换，实现了与加权回归在数学上等价的效果。这种方法不仅验证了我们之前的分析，还提供了一个新的视角来理解模型。

```
cleaners$Y_star <- cleaners$Offices * sqrt(cleaners$Weight)
cleaners$x_star <- cleaners$Staff * sqrt(cleaners$Weight)
cleaners$intercept_star <- sqrt(cleaners$Weight)
transformed_model <- lm(Y_star ~ 0 + intercept_star + x_star,
data = cleaners)
summary(transformed_model)
```

代码解析：

1. 创建转换变量：

- `Y_star`: 将因变量乘以权重的平方根
- `x_star`: 将自变量乘以权重的平方根
- `intercept_star`: 权重的平方根，用于转换后的截距项

2. `lm(Y_star ~ 0 + intercept_star + x_star, data = cleaners)`:
   - `0` 移除默认截距
   - `intercept_star` 作为新的截距项
   - `x_star` 作为转换后的自变量

统计学原理：

1. **变量转换的原理**:
   - 目标：将加权最小二乘问题转化为普通最小二乘问题
   - 方法：将所有变量乘以权重的平方根
2. **数学等价性**:
   - 原加权模型：min Σ w_i (Y_i - (β_0 + β_1 X_i))^2
   - 转换后：min Σ (√w_i Y_i - (β_0 √w_i + β_1 √w_i X_i))^2
   - 这两个优化问题在数学上是等价的
3. **转换模型的解释**:
   - 系数的解释与原始尺度相同
   - R平方值可能会改变，因为它现在基于转换后的变量
4. **优势**:
   - 允许使用标准的OLS诊断工具
   - 可以直观地看到加权如何影响各个观察
5. 解释转换模型的结果：
   - 给出系数的实际解释
   - 讨论统计显著性
   - 解释这些结果如何与原始问题联系
6. 讨论这种方法的优缺点：

- 优点：可以使用标准OLS诊断工具
- 缺点：可能使模型解释变得复杂

7. 考虑实际应用：
   - 讨论这种转换如何影响我们对清洁效率的理解
   - 考虑这种方法在其他类似问题中的应用

实际应用示例：

"通过变量转换，我们创建了一个等价于加权回归的模型。这个过程和结果揭示了几个重要的洞察：

1. 变量转换过程：
   我们创建了三个新变量：Y_star（转换后的Offices），x_star（转换后的Staff），和intercept_star（用于新截距）。每个变量都乘以权重的平方根，这effectively将加权最小二乘问题转化为普通最小二乘问题。
2. 模型比较：
   转换模型的结果与加权模型几乎完全一致：
   - 截距（通过intercept_star）：0.8095 (p = 0.471)
   - Staff系数（通过x_star）：3.8255 (p < 2e-16)
     这验证了两种方法的数学等价性，增强了我们对结果的信心。
3. 结果解释：
   - Staff系数（3.8255）表示，在考虑异方差性后，平均每增加一名员工，预期清洁的办公室数量增加约3.83个。
   - 这个估计比原始未加权模型（3.7009）略高，suggest未加权模型可能低估了员工数量的影响。
   - 截距不显著，indicate在理论上没有员工时，预期清洁的办公室数量接近于零，这在实际中是合理的。
4. 方法的优缺点：
   优点：
   - 允许使用标准的OLS诊断工具，如常规的残差图和影响力分析。
   - 提供了另一种视角来理解加权如何影响我们的估计。
     缺点：

- 转换后的变量可能难以直观解释。
- 需要额外的数据处理步骤。

(h) 残差比较

深入解析：

这最后一步旨在通过数值比较来验证加权模型和转换模型的等价性。这不仅是一个技术验证，还能加深我们对这两种方法本质的理解。

```
weighted_residuals <- residuals(weighted_model, type = "respo
nse") * sqrt(cleaners$Weight)
transformed_residuals <- residuals(transformed_model)
residual_difference <- weighted_residuals - transformed_resid
uals
print(summary(residual_difference))

all_same <- all(abs(residual_difference) < 1e-10)
cat("\\nAre all residuals essentially the same? ", all_same)

plot(weighted_residuals, transformed_residuals,
     main = "Weighted vs Transformed Residuals",
     xlab = "Weighted Residuals", ylab = "Transformed Residua
ls")
abline(0, 1, col = "red")
```

代码解析：

1. `residuals(weighted_model, type = "response") * sqrt(cleaners$Weight)`:
   计算加权模型的加权残差。
2. `residuals(transformed_model)`:
   提取转换模型的残差。
3. `abs(residual_difference) < 1e-10`:
   使用一个小的阈值来判断残差是否实质上相等。
4. `plot()` 和 `abline()`:
   创建残差对比散点图，并添加y=x参考线。

统计学原理：

1. **残差等价性**:
   - 理论上，加权模型的加权残差应与转换模型的残差完全一致。
   - 任何差异通常都是由于计算机浮点数运算的精度限制造成的。
2. **数值精度**:
   - 在计算机科学中，浮点数运算可能lead到微小的舍入误差。
   - 使用小阈值（如1e-10）来判断"实质相等"是一种常见做法。
3. **模型验证**:
   - 残差的一致性验证了两种方法在数学上的等价性。
   - 这增强了我们对两种方法都正确实现的信心。

如何回答这个问题：

1. 报告残差差异的摘要统计：
   - 描述差异的magnitude。
   - 解释为什么会存在微小差异（如果有的话）。
2. 解释残差实质上是否相同：
   - 基于设定的阈值，给出结论。
   - 讨论这个结论的含义。
3. 分析残差对比图：
   - 描述点的分布。
   - 解释这个图如何支持你的结论。
4. 讨论这个验证的重要性：
   - 在统计学方法论上的意义。
   - 对实际问题解决的影响。
5. 考虑更广泛的应用：
   - 讨论这种验证方法在其他统计分析中的潜在用途。

实际应用示例：

"通过比较加权模型的加权残差和转换模型的残差，我们得到了以下重要发现：

1. 残差差异摘要：
   差异的摘要统计显示：
   最小值: -2.220e-16 最大值: 4.441e-16
   中位数: 0.000e+00 平均值: 8.903e-18
   这表明差异极其微小，基本上可以忽略不计。
2. 残差等价性：
   使用1e-10作为阈值，我们发现所有残差对实质上都是相同的。这strong支持了两种方法的数学等价性。
3. 残差对比图分析：
   散点图显示所有点都精确地落在y=x线上，进一步证实了残差的一致性。
4. 验证的重要性：
   - 方法论意义：这个验证增强了我们对both加权回归和变量转换方法的信心。它展示了how不同的统计技术可以lead到相同的结果。
   - 实际问题解决：这个结果表明，我们可以灵活选择使用加权回归或变量转换方法，基于具体情况和个人偏好，而不会影响结果的准确性。

总结：

这个深度解析教程涵盖了ST346课程作业的全部内容，从数据加载到复杂的统计分析。通过这个过程，我们不仅学习了如何使用R进行数据分析，还深入理解了每个统计概念背后的原理。关键点包括：

1. 数据可视化和探索性数据分析的重要性。
2. 线性回归模型的拟合、解释和诊断。
3. 处理异方差性的方法，包括加权回归和变量转换。
4. 不同统计方法之间的数学等价性及其验证。