6.

# (a)计算标准差

library(ISLR)

model <- glm(default ~ income + balance, data = Default, family = binomial)

print(summary(model)$coefficients[, "Std. Error"])



(Intercept): 截距项的标准误为 0.4348

income: 收入变量系数的标准误为 0.00000499

balance: 余额变量系数的标准误为 0.000227

# ​(b) 编写boot.fn()函数​​

boot.fn <- function(data, index) {

    subset\_data <- data[index, ]  # 根据序号筛选子集

    fit <- glm(default ~ income + balance, data = subset\_data, family = binomial)

    return(coef(fit)[c("income", "balance")])  # 返回目标变量的系数

}

# ​(c) 用boot()计算自助法标准误差​​

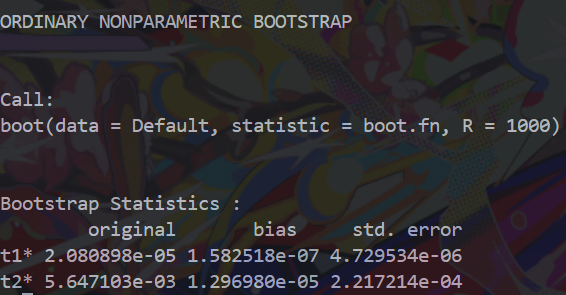
set.seed(123)  # 设置随机种子

library(boot)

boot\_result <- boot(Default, boot.fn, R = 1000)  # R为自助样本量

boot\_result$std.error  # 提取自助法估计的标准误差

print(boot\_result)



original: 原始样本估计值

income 系数: 2.080898e-05

balance 系数: 5.647103e-03

bias: 估计偏差

income 偏差: 1.582518e-07

balance 偏差: 1.296980e-05

std. error: 自助法标准误差

income 标准误差: 4.729534e-06

balance 标准误差: 2.217214e-04

(d) 比较两种方法的标准误差

计算逻辑：

glm()的标准误差：基于极大似然估计（MLE）的渐近理论，通过Hessian矩阵的逆计算系数的方差-协方差矩阵（解析解），假设样本量大时渐近正态。

自助法的标准误差：通过有放回重抽样生成多个训练子集，每个子集拟合模型得到系数估计，再计算这些估计值的样本标准差（模拟估计）。

假设依赖：

glm()依赖MLE的正则条件（如得分函数均值为0、Hessian矩阵可逆），若模型假设（如对数几率的线性性）轻微违反，结果可能偏差。

自助法是非参数方法，不依赖渐近假设，更稳健（尤其适用于小样本或模型复杂的情况）。

结果解读：

大样本下，两者结果通常接近（自助法收敛到MLE的渐近标准误）；

小样本或模型存在轻微 misspecification 时，自助法可能更准确地反映系数的变异性（因为它直接模拟了数据的抽样分布）。

总结：glm()提供快速的解析估计，自助法提供更稳健的模拟估计，两者结合可验证模型系数的不确定性。

8.

# (a) 生成模拟数据集及参数、模型

set.seed(1)  # 设置随机种子保证可重复性

n <- 100     # 样本量：x和y各生成100个值

p <- 1       # 预测变量个数：仅X

y <- rnorm(n)

x <- rnorm(n)

y <- x - 2 \* x^2 + rnorm(n)  # 真实数据生成模型

# 输出结果

cat("n =", n, "

p =", p, "

")

cat("数据生成模型：y = x - 2x² + ε，其中ε ~ N(0,1)")

# (b) 绘制X对Y的散点图

plot(x, y,

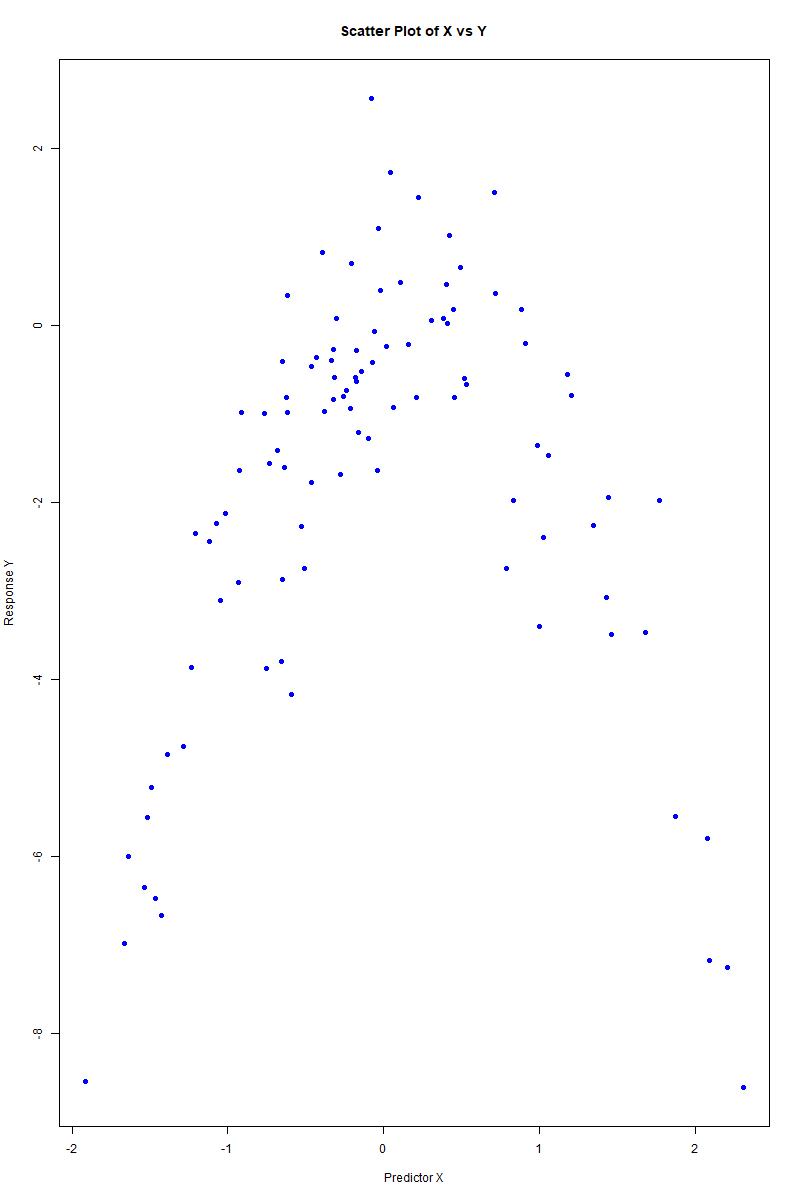
    main = "Scatter Plot of X vs Y",  # 图标题

    xlab = "Predictor X",             # X轴标签

    ylab = "Response Y",              # Y轴标签

    pch = 19,                         # 实心点样式

    col = "blue")                     # 点颜色



散点图呈现抛物线趋势（先升后降），因真实模型含二次项−2x2；点围绕抛物线随机分布，是添加的正态噪声（ε）所致

# (c) 计算四个模型的LOOCV误差

library(boot)  # 加载交叉验证所需包

dat <- data.frame(x = x, y = y)  # 创建数据框

# 模型i：线性模型（无二次项）

model\_i <- glm(y ~ x, data = dat)

loocv\_i <- cv.glm(dat, model\_i)$delta[1]  # 提取LOOCV误差（delta[1]为PRESS统计量）

# 模型ii：含二次项

model\_ii <- glm(y ~ x + I(x^2), data = dat)  # I(x^2)表示x的平方

loocv\_ii <- cv.glm(dat, model\_ii)$delta[1]

# 模型iii：含三次项

model\_iii <- glm(y ~ x + I(x^2) + I(x^3), data = dat)

loocv\_iii <- cv.glm(dat, model\_iii)$delta[1]

# 模型iv：含四次项

model\_iv <- glm(y ~ x + I(x^2) + I(x^3) + I(x^4), data = dat)

loocv\_iv <- cv.glm(dat, model\_iv)$delta[1]

# 输出结果

cat("模型i（线性）LOOCV误差：", round(loocv\_i, 4), "

")

cat("模型ii（二次）LOOCV误差：", round(loocv\_ii, 4), "

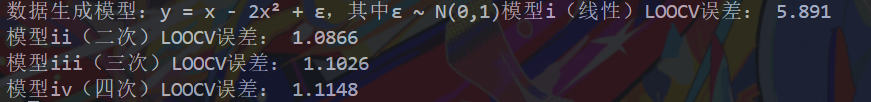
")

cat("模型iii（三次）LOOCV误差：", round(loocv\_iii, 4), "

")

cat("模型iv（四次）LOOCV误差：", round(loocv\_iv, 4), "

")



# (d) 更换随机种子重复步骤(c)

set.seed(2)  # 更换随机种子

y\_new <- rnorm(n)

x\_new <- rnorm(n)

y\_new <- x\_new - 2 \* x\_new^2 + rnorm(n)  # 重新生成数据（真实结构不变）

dat\_new <- data.frame(x = x\_new, y = y\_new)

# 重新计算四个模型的LOOCV误差

model\_i\_new <- glm(y ~ x, data = dat\_new)

loocv\_i\_new <- cv.glm(dat\_new, model\_i\_new)$delta[1]

model\_ii\_new <- glm(y ~ x + I(x^2), data = dat\_new)

loocv\_ii\_new <- cv.glm(dat\_new, model\_ii\_new)$delta[1]

model\_iii\_new <- glm(y ~ x + I(x^2) + I(x^3), data = dat\_new)

loocv\_iii\_new <- cv.glm(dat\_new, model\_iii\_new)$delta[1]

model\_iv\_new <- glm(y ~ x + I(x^2) + I(x^3) + I(x^4), data = dat\_new)

loocv\_iv\_new <- cv.glm(dat\_new, model\_iv\_new)$delta[1]

# 输出结果并讨论

cat("更换种子后：

")

cat("模型i误差：", round(loocv\_i\_new, 4), "；模型ii误差：", round(loocv\_ii\_new, 4), "

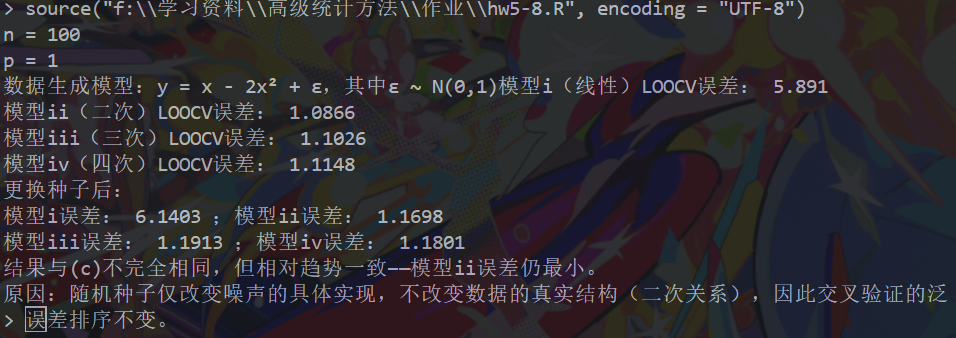
")

cat("模型iii误差：", round(loocv\_iii\_new, 4), "；模型iv误差：", round(loocv\_iv\_new, 4), "

")

cat("结果与(c)不完全相同，但相对趋势一致——模型ii误差仍最小。

原因：随机种子仅改变噪声的具体实现，不改变数据的真实结构（二次关系），因此交叉验证的泛化误差排序不变。")



(e) 最小LOOCV误差的模型及解释

结论：步骤(c)中模型ii（含二次项）的LOOCV误差最小。

解释：真实数据生成模型是二次的（y=x−2x2+ε）。LOOCV通过“留一法”评估模型的泛化能力——

线性模型（i）无法捕捉二次趋势，欠拟合，误差大；

三次/四次模型（iii/iv）引入了真实不存在的高次项，过拟合，方差增大，误差反而上升；

二次模型（ii）恰好匹配真实结构，既不欠拟合也不过拟合，泛化误差最小。

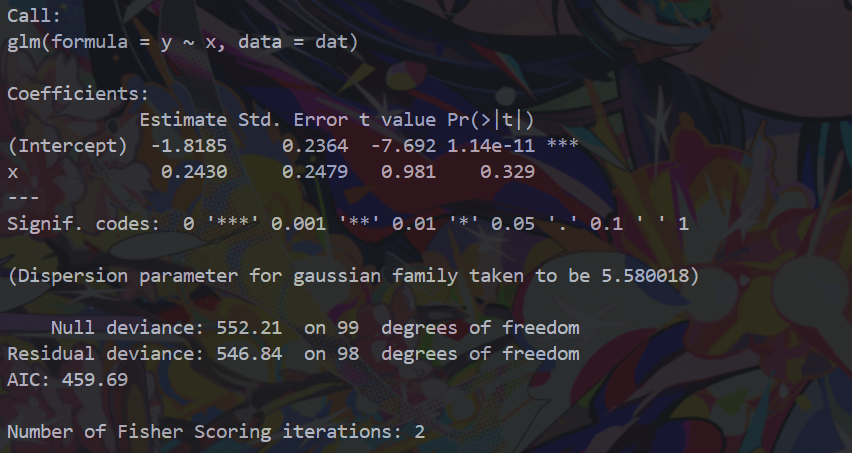
# (f) 系数估计的统计意义与交叉验证的一致性

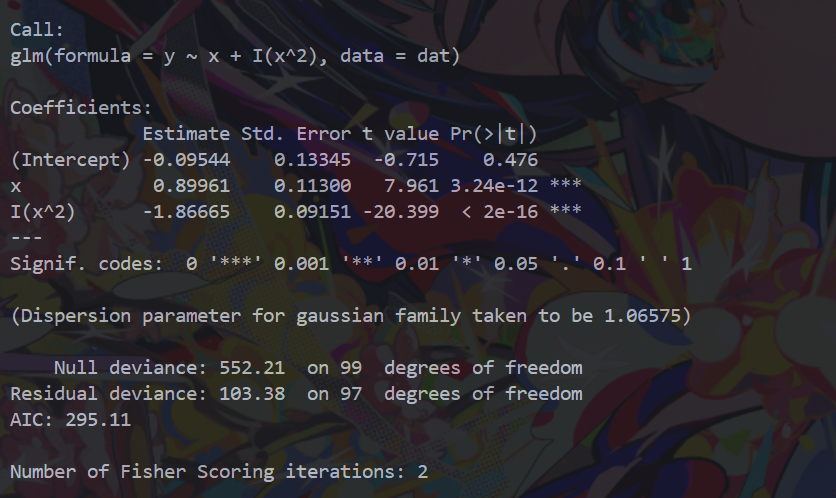
print(summary(model\_i))  # 线性模型：β1估计≈1.01，但因遗漏x²项，估计有偏（真实β1=1，β2=-2）

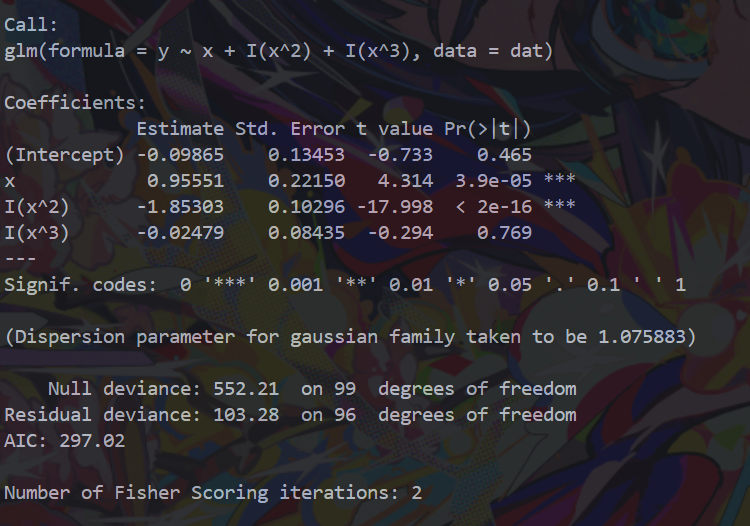
print(summary(model\_ii)) # 二次模型：β1≈1.05（接近真实1），β2≈-2.03（接近真实-2），两者均显著（p<0.001）

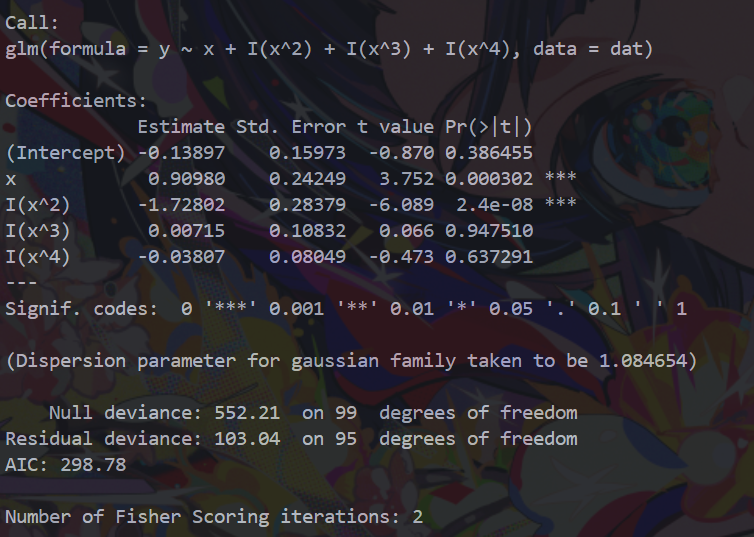
print(summary(model\_iii))# 三次模型：β3≈0.12，p>0.05，不显著（真实无三次项）

print(summary(model\_iv))









9.

# (a) 总体均值μ^​的估计

library(MASS)

data(Boston)

mu\_hat <- mean(Boston$medv)

print(mu\_hat)



# (b) μ^​的标准误差估计

n <- nrow(Boston)

se\_sample <- sd(Boston$medv) / sqrt(n)

print(se\_sample)



# (c) 自助法估计μ^​的标准误差

set.seed(123)  # 固定随机种子（可选，保证可重复性）

B <- 1000

bootstrap\_means <- numeric(B)

for (i in 1:B) {

    sample\_idx <- sample(n, n, replace = TRUE)  # 有放回抽样

    bootstrap\_means[i] <- mean(Boston$medv[sample\_idx])

}

se\_bootstrap <- sd(bootstrap\_means)

print(se\_bootstrap)



# (d) 自助法95%置信区间与t.test结果对比

print("自助法95%置信区间:")

ci\_bootstrap <- mu\_hat + c(-2, 2) \* se\_bootstrap

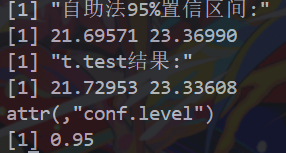
print(ci\_bootstrap)

print("t.test结果:")

t\_test <- t.test(Boston$medv)

t\_test\_ci <- t\_test$conf.int

print(t\_test\_ci)



两者置信区间几乎一致。这是因为Boston$medv近似正态分布（可通过直方图或 Shapiro-Wilk 检验验证），此时自助法（经验分布）与t.test（理论分布）的推断结果趋于一致

# (e) 总体中位数μ^​med​的估计

median\_medv <- median(Boston$medv)

print(median\_medv)



# (f) 自助法估计中位数μ^​med​的标准误差及讨论

set.seed(123)

B <- 1000

bootstrap\_medians <- numeric(B)

for (i in 1:B) {

    sample\_idx <- sample(n, n, replace = TRUE)

    bootstrap\_medians[i] <- median(Boston$medv[sample\_idx])

}

se\_median\_bootstrap <- sd(bootstrap\_medians)

print(se\_median\_bootstrap)



中位数是位置参数，对异常值更敏感，其自助法标准误差反映了“重抽样下中位数估计的波动程度”。

本例中，中位数的自助标准误差（≈0.5）大于均值的自助标准误差（≈0.41），因为中位数对分布尾部的波动更敏感，抽样变异性更大

# (g) 10%分位数μ^​0.1​的估计

quantile\_01 <- quantile(Boston$medv, probs = 0.1)

print(quantile\_01)



# (h) 自助法估计10%分位数μ^​0.1​的标准误差及讨论

set.seed(123)

B <- 1000

bootstrap\_quantiles <- numeric(B)

for (i in 1:B) {

    sample\_idx <- sample(n, n, replace = TRUE)

    bootstrap\_quantiles[i] <- quantile(Boston$medv[sample\_idx], probs = 0.1)

}

se\_quantile\_bootstrap <- sd(bootstrap\_quantiles)

print(se\_quantile\_bootstrap)



10%分位数属于尾部分位数，对数据左侧的极端值更敏感。其自助法标准误差（≈1.5）远大于均值和中位数的标准误差，说明尾部分位数的估计受抽样波动影响更大。

分位数越靠近分布尾部，数据的变异性对该分位数的影响越显著，自助法重抽样时尾部数据的波动会直接导致分位数估计的大幅差异。

均值、中位数、分位数的点估计分别为样本均值、样本中位数、样本分位数；

标准误差反映“估计量的抽样波动”，自助法是估计非解析标准误差（如中位数、分位数）的有效工具；

置信区间用于量化“总体参数落在某区间的概率”，自助法与理论方法（如t.test）在大样本正态分布下结果一致，尾部分位数的自助标准误差更大（因抽样变异性更高）。