

多元数据分析第三次作业

应数 2101 杨嘉昱 2216113458

2024 年 5 月 7 日

第二章作业

2.2

Solution.

1. 模型为

$$Y = X\beta + \varepsilon$$

最小二乘估计为

$$\hat{\beta} = (X^T X)^{-1} X^T Y = \frac{X^T Y}{X^T X} = \frac{\sum_{j=1}^n x_j y_j}{\sum_{j=1}^n x_j^2}.$$

由于

$$\hat{\beta} = \frac{X^T Y}{X^T X} = \frac{X^T (X\beta + \varepsilon)}{X^T X} = \beta + \frac{X^T \varepsilon}{X^T X}.$$

从而

$$\mathbb{E}\hat{\beta} = \beta + \mathbb{E}\left(\frac{X^T \varepsilon}{X^T X}\right) = \beta$$

即 $\hat{\beta}$ 为 β 的无偏估计。

2. 令

$$\hat{\sigma}^2 = \frac{1}{n-1} Y^T (I - H) Y, \quad H = \frac{X X^T}{X^T X}.$$

则

$$\begin{aligned} (n-1)\mathbb{E}\hat{\sigma}^2 &= \mathbb{E}\text{tr}(Y^T (I - H) Y) = \mathbb{E}\text{tr}(\varepsilon^T (I - H) \varepsilon) \\ &= \mathbb{E}\text{tr}((I - H) \varepsilon \varepsilon^T) = \sigma^2 \text{tr}\left(I - \frac{X X^T}{X^T X}\right) = (n-1)\sigma^2 \end{aligned}$$

从而

$$\hat{\sigma}^2 = \frac{1}{n-1} \frac{X^T X Y^T Y - (X^T Y)^2}{X^T X} = \frac{1}{n-1} \cdot \frac{\left(\sum_{j=1}^n x_j^2\right) \left(\sum_{j=1}^n y_j^2\right) - \left(\sum_{j=1}^n x_j y_j\right)^2}{\sum_{j=1}^n x_j^2}$$

是 σ^2 的无偏估计。

3. 全模型为

$$Y = X\beta + \varepsilon.$$

$$\text{SSE}(R) = Y^T (I - H) Y = \frac{X^T X Y^T Y - (X^T Y)^2}{X^T X}, \quad f_R = n - 1.$$

约简模型为

$$Y = \varepsilon$$

$$\text{SSE}(F) = \sum_{j=1}^n (y_j - 0)^2 = Y^T Y, \quad f_F = n.$$

从而统计量以及零分布为

$$F = \frac{(\text{SSE}(R) - \text{SSE}(F))/1}{\text{SSE}(F)/n} = n \cdot \frac{\text{SSE}(R) - \text{SSE}(F)}{\text{SSE}(F)} \sim F(1, n).$$

4. 由于

$$\hat{\beta} \sim \mathcal{N}\left(\beta, \sigma^2 (X^T X)^{-1}\right), \quad \frac{n-1}{\sigma^2} \hat{\sigma}^2 \sim \chi(n-1).$$

依假设 $\beta = 0$ 。从而 t 统计量集及零分布为：

$$\frac{\hat{\beta} \sqrt{X^T X}}{\hat{\sigma}} \sim t(n-1).$$

5. 由于

$$\hat{y}_0 = x_0^T \hat{\beta} \sim \mathcal{N}\left(x_0^T \beta, \sigma^2 x_0^T (X^T X)^{-1} x_0\right) \quad y_0 = x_0^T \beta + \varepsilon \sim \mathcal{N}\left(x_0^T \beta, \sigma^2\right)$$

故

$$\hat{y}_0 - y_0 \sim \mathcal{N}\left(0, \sigma^2 \left(1 + x_0^T (X^T X)^{-1} x_0\right)\right)$$

从而

$$\frac{\hat{y}_0 - y_0}{\sqrt{\text{MSE} \left(1 + x_0^T (X^T X)^{-1} x_0\right)}} \sim t(n-1).$$

从而置信度为 $1 - \alpha$ 的置信区间为

$$\begin{aligned} & \hat{y}_0 \pm t_{1-\alpha/2}(n-p) \sqrt{\text{MSE} \left(1 + x_0^T (X^T X)^{-1} x_0\right)} \\ &= x_0^T \frac{X^T Y}{X^T X} \pm t_{1-\alpha/2}(n-p) \sqrt{\frac{1}{n-1} \frac{X^T X Y^T Y - (X^T Y)^2}{X^T X} \cdot \frac{X^T X + x_0^T x_0}{X^T X}}. \end{aligned}$$

□

2.3

Solution. 记

$$X_i = (x_{1i}, \dots, x_{ni})^T, (i = 1, 2), \quad X_3 = (\sqrt{x_{13}}, \dots, \sqrt{x_{n3}})^T$$

那么全模型为

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2 + \beta_4 X_3 + \varepsilon.$$

此时自由度以及误差平方和为

$$f_R = 4, \quad \text{SSE}(R)$$

1. 约简模型为

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon$$

自由度以及误差平方和为

$$f_F = 2, \quad \text{SSE}(F)$$

统计量以及零分布为

$$\frac{(\text{SSE}(R) - \text{SSE}(F))/(f_R - f_F)}{\text{SSE}(F)/f_F} = \frac{\text{SSE}(R) - \text{SSE}(F)}{\text{SSE}(F)} \sim F(2, 2).$$

2. 约简模型为

$$Y = \beta_0 + \beta_1(X_1 + X_3) + \beta_3X_1X_2 + \beta_4X_3 + \varepsilon.$$

自由度以及误差平方和为

$$f_F = 3, \quad \text{SSE}(F)$$

统计量以及零分布为

$$\frac{(\text{SSE}(R) - \text{SSE}(F))/(f_R - f_F)}{\text{SSE}(F)/f_F} = 3 \cdot \frac{\text{SSE}(R) - \text{SSE}(F)}{\text{SSE}(F)} \sim F(1, 3).$$

3. 约简模型为

$$Y = \beta_0 + \beta_1X_1 + \beta_2X_2 + \beta_3X_1X_2 + X_3 + \varepsilon.$$

自由度以及误差平方和为

$$f_F = 3, \quad \text{SSE}(F)$$

统计量以及零分布为

$$\frac{(\text{SSE}(R) - \text{SSE}(F))/(f_R - f_F)}{\text{SSE}(F)/f_F} = 3 \cdot \frac{\text{SSE}(R) - \text{SSE}(F)}{\text{SSE}(F)} \sim F(1, 3).$$

□

2.4

Solution. 记

$$Y = (y_1, \dots, y_n)^T, \quad \varepsilon = (\varepsilon_1, \dots, \varepsilon_n)^T, \quad X_i = (x_{1i}, \dots, x_{ni})^T, i = 1, 2$$

其中 $n = 15$ 。又记

$$\beta = (\beta_0, \beta_1, \beta_2)^T, \quad X = (\mathbf{1}, X_1, X_2)$$

那么模型为

$$Y = \beta_0 + \beta_1X_1 + \beta_2X_2 + \varepsilon = X\beta + \varepsilon.$$

1. β 的最小二乘估计为

$$\hat{\beta} = (X^T X)^{-1} X^T Y$$

均方误差的估计为

$$\hat{\sigma}^2 = \frac{1}{14} Y^T (I - H) Y, \quad H = X (X^T X)^{-1} X^T.$$

带入数值求解得:

$$\hat{\beta} = (3.45, 4.96, 0.01)^T, \quad \hat{\sigma}^2 = 4.74.$$

2. 方差分析表为 由于 p 值很小, 拒绝假设 $H_0: \beta_1 = \beta_2 = 0$, 即认为 Y 与 X_1, X_2 的线性回归关系显

方差来源	自由度	平方和	均方	F 值	p 值
回归 (R)	2	53844.72	26922.36	5679.466	0
误差 (E)	12	56.88	4.74		
总和 (T)	14	53901.6			

著。另一方面, 计算得

$$R = 0.9994722.$$

说明 Y 与 X_1, X_2 线性关系显著。

3. 置信区间分别为

$$[4.83, 5.09] \quad [0.007, 0.01]$$

4. 全模型为

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2 + \varepsilon$$

约简模型为

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon$$

检验假设为

$$H_0 : \beta_3 = 0.$$

统计量观测值与 p 值为

$$F = 0.03156013, \quad p = 0$$

从而拒绝原假设，认为有必要引入交叉项。

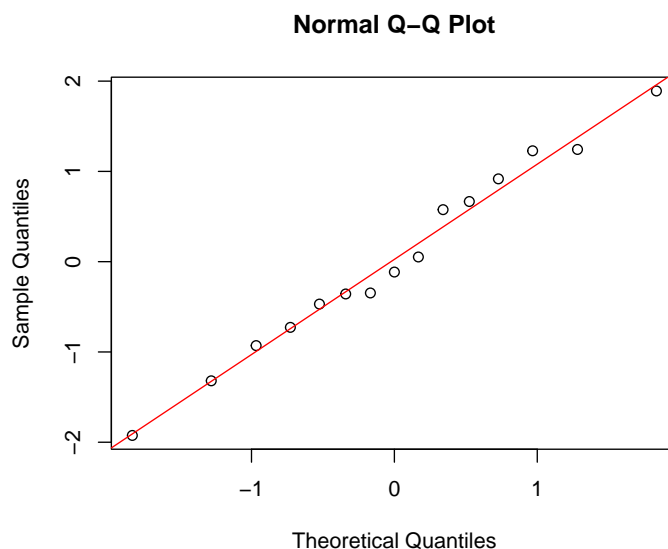
5. 预测值与置信区间分别为

$$1113.568, \quad [1086.16, 1140.975].$$

6. 拟合值，残差与学生化残差分别为

$$\begin{aligned} \hat{y} &= (161.90, 122.67, 224.43, 131.24, 67.70, 169.68, 79.73, 189.67, \\ &\quad 119.83, 53.29, 253.72, 228.69, 144.98, 100.53, 210.94)^T \\ \hat{\varepsilon} &= (0.10, -2.66, -1.43, -0.24, -0.70, -0.68, 1.27, \\ &\quad 2.32, -3.83, 1.71, -1.72, 3.31, -0.98, 2.46, 1.06)^T \\ r &= (0.05, -1.32, -0.73, -0.11, -0.36, -0.35, 0.67, \\ &\quad 1.23, -1.92, 0.92, -0.93, 1.89, -0.47, 1.24, 0.58)^T \end{aligned}$$

QQ 图如下



由于散点图近似线性，从而可以认为误差项是正态的。并且求的相关系数为

$$\hat{\rho} = 0.9933947.$$

本题所用代码如下：

```

1 D <- read.table('exercise2_4.txt')
2 colnames(D) <- c('y', 'X1', 'X2')
3 lm.obj <- lm(y~., data = D)
4 summary(lm.obj)
5
6 sse <- sum (( fitted (lm.obj) - D$y)^2)
7 ssr <- sum (( fitted (lm.obj) - mean (D$y))^2)
8 sst = sse + ssr
9 msr = ssr / 2
10 mse = sse / 12
11 f_value = msr / mse
12 p_value <- 1 - pf(f_value, 2, 12)
13 R_value <- sqrt(ssr/sst)
14 ssr
15 sse
16 sst
17 msr
18 mse
19 f_value
20 p_value
21 R_value
22
23 A <- summary(lm.obj)$coefficients
24 alpha <- 0.05
25 df <- lm.obj$df.residual
26 A[,1]
27 left <- A[,1] - A[,2] * qt(1 - alpha / 2, df)
28 right <- A[,1] + A[,2] * qt(1 - alpha / 2, df)
29 left
30 right
31
32 new.data <- D
33 new.data[,4] <- D[,2] * D[,3]
34 fit <- lm(y~., data = new.data)
35 f_0_num <- (sum(lm.obj$residuals^2) - sum(fit$residuals^2)) / (lm.obj$df.residual - fit$df.
    residual)
36 f_0_den <- sum(fit$residuals^2) / fit$df.residual
37 f_0 <- f_0_num / f_0_den
38 p.value <- 1 - pf(f_0, lm.obj$df.residual - fit$df.residual, fit$df.residual)
39 f_0
40 p_value
41
42 predict(lm.obj, data.frame(X1 = 220, X2 = 2055), interval = 'prediction', type = 'response')
43
44 stand.residuals <- rstandard(lm.obj)
45 qqnorm(stand.residuals)
46 x <- seq(from = -2.5, to = 2.5, by = 0.1)
47 y <- sd(stand.residuals) * x + mean(stand.residuals)
48 abline(x, y, coef = c(mean(stand.residuals), sd(stand.residuals)), col = 'red')
49
50 correlation <- cor(qqnorm(stand.residuals)$x, qqnorm(stand.residuals)$y)
51 fitted_value <- lm.obj$fitted.values
52 residual_value <- lm.obj$residuals
53 fitted_value
54 residual_value
55 stand.residuals
56 correlation
57

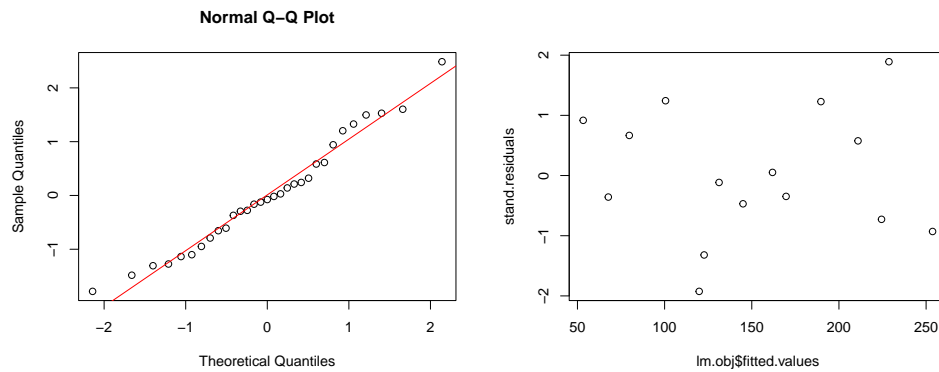
```

□

2.6

Solution.

1. 正态 QQ 图和学生化残差 - \hat{y} 图如下

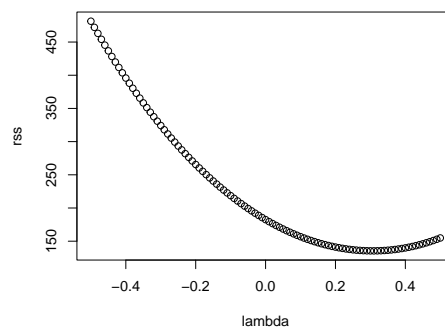


相关系数为

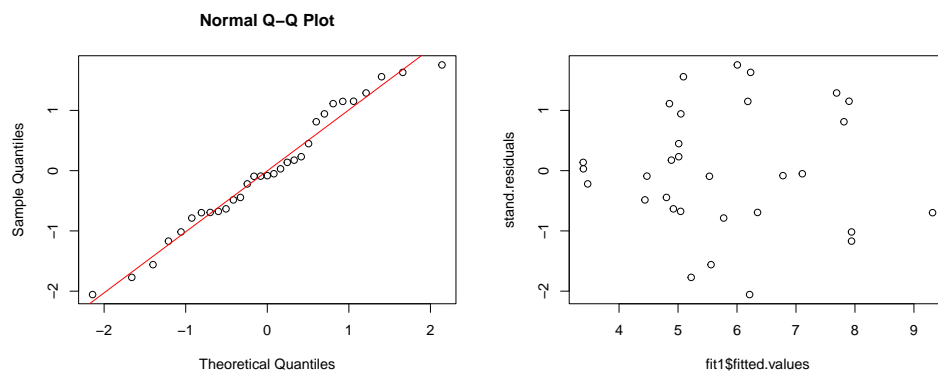
$$\hat{\rho} = 0.9883331.$$

由于散点图近似线形，因此可认为模型是合理的。

2. 计算得变换参数的值为 $\lambda = 0.66$



正态 QQ 图和学生化残差 - \hat{y} 图如下



相关系数为

$$\hat{\rho} = 0.9889423.$$

说明拟合程度较线性更加成功。

本题所用代码如下：

```

1 D <- read.table('exercise2_6.txt')
2 colnames(D) <- c('X1', 'X2', 'y')
3 lm.obj <- lm(y~., data = D)
4 summary(lm.obj)
5
6 stand.residuals <- rstandard(lm.obj)
7 qqnorm(stand.residuals)
8 correlation <- cor(qqnorm(stand.residuals)$x, qqnorm(stand.residuals)$y)
9 x <- seq(from = -2.5, to = 2.5, by = 0.1)
10 y <- sd(stand.residuals) * x + mean(stand.residuals)
11 abline(x, y, coef = c(mean(stand.residuals), sd(stand.residuals)), col = 'red')
12 correlation
13
14 plot(lm.obj$fitted.values, stand.residuals)
15
16 n <- dim(D)[1]
17 X <- D[,1:2]
18 y <- D[,3]
19 X <- as.matrix(cbind(matrix(1,n,1), X))
20 H <- X %%% solve(t(X) %%% X) %%% t(X)
21 lambda <- seq(from = -0.5, to = 0.5, by = 0.01)
22 K <- length(lambda)
23 rss <- rep(0, K)
24 for (k in 1:K) {
25   if (lambda[k] != 0) {
26     Z <- (1 / lambda[k]) * (y^(lambda[k]) - 1) / (prod(y)) ^ ((lambda[k] - 1) / n)
27   } else {
28     Z <- log(y) * (prod(y))^(1 / n)
29   }
30   rss[k] <- t(Z) %%% (diag(n) - H) %%% Z
31 }
32 opt.lamb <- lambda[which.min(rss)]
33 opt.lamb
34
35 new.y <- (y ^ opt.lamb - 1) / (opt.lamb)
36 new.data <- D
37 new.data[,3] <- new.y
38 fit1 <- lm(y~., data = new.data)
39 summary(fit1)
40
41 stand.residuals <- rstandard(fit1)
42 qqnorm(stand.residuals)
43 correlation <- cor(qqnorm(stand.residuals)$x, qqnorm(stand.residuals)$y)
44 x <- seq(from = -2.5, to = 2.5, by = 0.1)
45 y <- sd(stand.residuals) * x + mean(stand.residuals)
46 abline(x, y, coef = c(mean(stand.residuals), sd(stand.residuals)), col = 'red')
47 correlation
48
49 plot(fit1$fitted.values, stand.residuals)
50

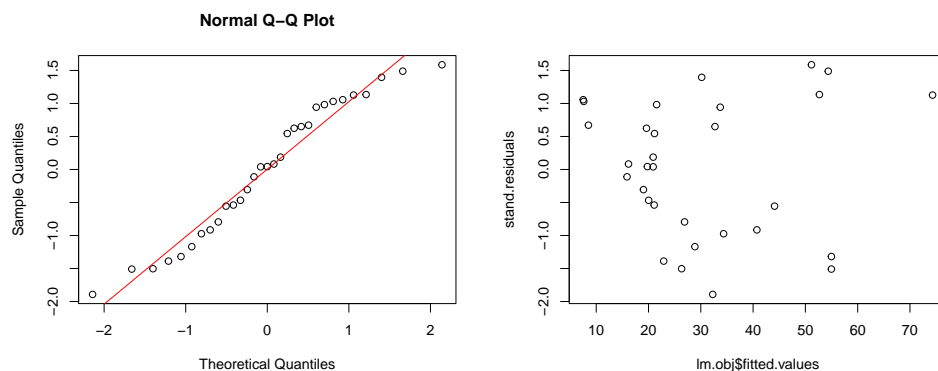
```

□

2.7

Solution.

1. 正态 QQ 图和学生化残差 - \hat{y} 图如下

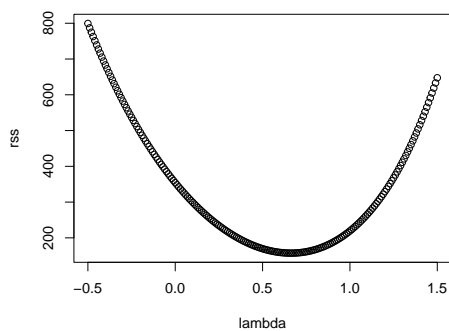


相关系数为

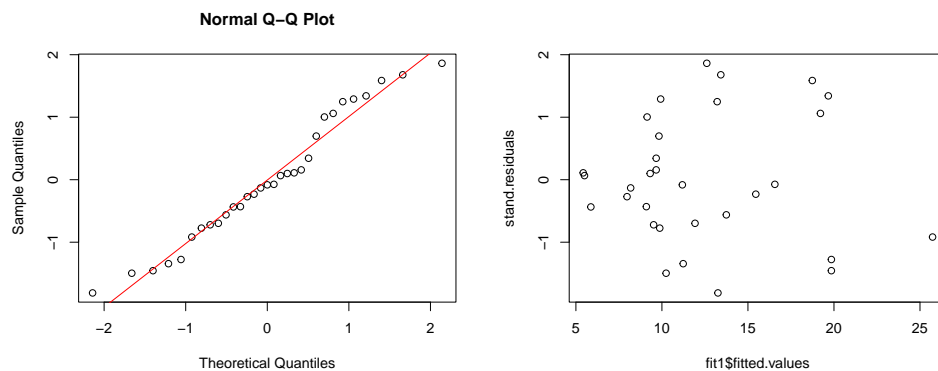
$$\hat{\rho} = 0.9791288.$$

由于散点图不是近似线形，因此可认为模型是不合理的。

2. 计算得变换参数的值为 $\lambda = 0.66$



正态 QQ 图和学生化残差 - \hat{y} 图如下



QQ 图近似线性，相关系数为

$$\hat{\rho} = 0.9889423.$$

说明拟合程度较线性更加成功。

所用代码如下

```

1 D_r <- read.table('exercise2_6.txt')
2 D <- D_r
3 D[,1] <- D_r[,1] * D_r[,1]
4 colnames(D) <- c('X1', 'X2', 'y')
5 lm.obj <- lm(y~., data = D)
6 summary(lm.obj)
7
8 stand.residuals <- rstandard(lm.obj)
9 qqnorm(stand.residuals)
10 correlation <- cor(qqnorm(stand.residuals)$x, qqnorm(stand.residuals)$y)
11 x <- seq(from = -2.5, to = 2.5, by = 0.1)
12 y <- sd(stand.residuals) * x + mean(stand.residuals)
13 abline(x, y, coef = c(mean(stand.residuals), sd(stand.residuals)), col = 'red')
14 correlation
15
16 plot(lm.obj$fitted.values, stand.residuals)
17
18 n <- dim(D)[1]
19 X <- D[,1:2]
20 y <- D[,3]
21 X <- as.matrix(cbind(matrix(1,n,1), X))
22 H <- X %*% solve(t(X) %*% X) %*% t(X)
23 lambda <- seq(from = -0.5, to = 1.5, by = 0.01)
24 K <- length(lambda)
25 rss <- rep(0, K)
26 for (k in 1:K) {
27   if (lambda[k] != 0) {
28     Z <- (1 / lambda[k]) * (y^(lambda[k]) - 1) / (prod(y)) ^ ((lambda[k] - 1) / n)
29   } else {
30     Z <- log(y) * (prod(y))^(1 / n)
31   }
32   rss[k] <- t(Z) %*% (diag(n) - H) %*% Z
33 }
34 plot(lambda,rss)
35 opt.lamb <- lambda[which.min(rss)]
36 opt.lamb
37
38 new.y <- (y ^ opt.lamb - 1) / (opt.lamb)
39 new.data <- D
40 new.data[,3] <- new.y
41 fit1 <- lm(y~., data = new.data)
42 summary(fit1)
43
44 stand.residuals <- rstandard(fit1)
45 qqnorm(stand.residuals)
46 correlation <- cor(qqnorm(stand.residuals)$x, qqnorm(stand.residuals)$y)
47 x <- seq(from = -2.5, to = 2.5, by = 0.1)
48 y <- sd(stand.residuals) * x + mean(stand.residuals)
49 abline(x, y, coef = c(mean(stand.residuals), sd(stand.residuals)), col = 'red')
50 correlation
51
52 plot(fit1$fitted.values, stand.residuals)
53

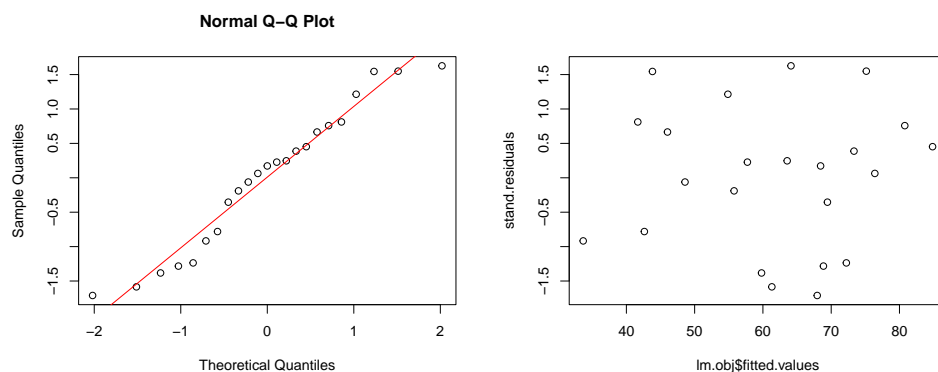
```

□

2.9

Solution.

1. 正态 QQ 图和学生化残差 - \hat{y} 图如下



相关系数为

$$\hat{\rho} = 0.9816644.$$

由于散点图近似线形，因此可认为模型是合理的。

2. (a) $R_a^2(p)$ 准则：计算得

$$X_1 \leftrightarrow 0.5794702, \quad X_1, X_2 \leftrightarrow 0.6305423 \quad X_1, X_2, X_3 \leftrightarrow 0.6209731$$

故选取 X_1, X_2 作为变量。

(b) C_p 准则：计算得

$$X_1 \leftrightarrow 4.299472, \quad X_1, X_2 \leftrightarrow 2.495063 \quad X_1, X_2, X_3 \leftrightarrow 4.000000$$

故选取 X_1, X_2 作为变量。

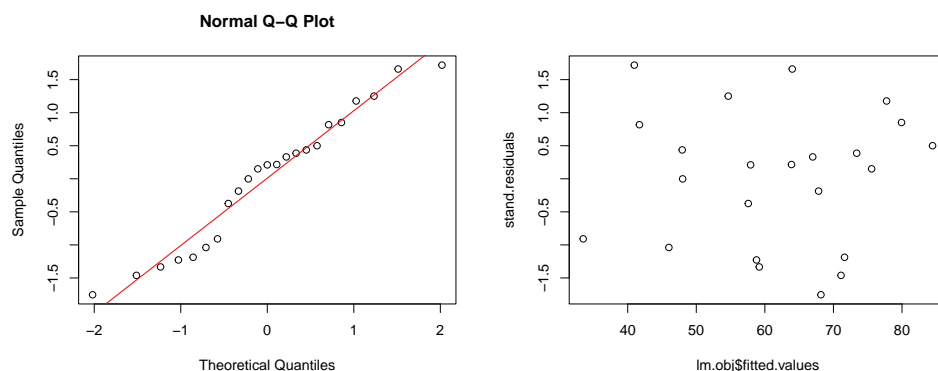
(c) $PRESS_p$ 准则：计算得

$$X_1 \leftrightarrow 0.5985851, \quad X_1, X_2 \leftrightarrow 0.6641294 \quad X_1, X_2, X_3 \leftrightarrow 0.6726586$$

故选取 X_1 作为变量。

3. 逐步回归法得：选取 X_1, X_2 作为变量。基本与 2 中结果一致。

4. 正态 QQ 图和学生化残差 - \hat{y} 图如下



相关系数为

$$\hat{\rho} = 0.982029.$$

由于散点图近似线形，因此可认为模型是合理的。

所用代码如下

```

1 D <- read.table('exercise2_9.txt')
2 colnames(D) <- c('X1', 'X2', 'X3', 'y')
3 lm.obj <- lm(y~., data = D)
4 summary(lm.obj)
5
6 stand.residuals <- rstandard(lm.obj)
7 qqnorm(stand.residuals)
8 correlation <- cor(qqnorm(stand.residuals)$x, qqnorm(stand.residuals)$y)
9 x <- seq(from = -2.5, to = 2.5, by = 0.1)
10 y <- sd(stand.residuals) * x + mean(stand.residuals)
11 abline(x, y, coef = c(mean(stand.residuals), sd(stand.residuals)), col = 'red')
12 correlation
13
14 plot(lm.obj$fitted.values, stand.residuals)
15
16 library(leaps)
17 regfit.full = regsubsets(y~., data = D)
18 summary(regfit.full)
19 summary(regfit.full)$adjr2
20 summary(regfit.full)$cp
21 summary(regfit.full)$rsq
22
23 model <- lm(y ~ ., data = D)
24 step_model <- step(model, direction = "both")
25 summary(step_model)
26
27 new.data <- D[,1:2]
28 new.data[,3] <- D[,4]
29 colnames(new.data) <- c('X1', 'X2', 'y')
30 new.data
31 lm.obj <- lm(y~., data = new.data)
32 summary(lm.obj)
33
34 stand.residuals <- rstandard(lm.obj)
35 qqnorm(stand.residuals)
36 correlation <- cor(qqnorm(stand.residuals)$x, qqnorm(stand.residuals)$y)
37 x <- seq(from = -2.5, to = 2.5, by = 0.1)
38 y <- sd(stand.residuals) * x + mean(stand.residuals)
39 abline(x, y, coef = c(mean(stand.residuals), sd(stand.residuals)), col = 'red')
40 correlation
41
42 plot(lm.obj$fitted.values, stand.residuals)
43

```

□

2.10

Solution.

1. 带入数值求解得

$$\beta = (-20.00, 0.61, 0.18, 8.98)^T$$

检验 p 值分别为

$$0.542, \quad 0.526 \quad 0.153 \quad 0.510$$

从而

$$P(Y = 1) = \frac{\exp(-20.00 + 0.61X_1 + 0.18X_2 + 8.98X_3)}{1 + \exp(-20.00 + 0.61X_1 + 0.18X_2 + 8.98X_3)}$$

并且只有 X_2 对是否破产有重要影响。

2. 检验假设为

$$H_0: \beta_3 = 0.$$

计算得 p 值为 $0.510 > 0.05$, 可认为无显著影响。

建立仅含 X_1, X_2 的模型为

$$P(Y = 1) = \frac{\exp(-0.55037 + 0.15737X_1 + 0.19475X_2)}{1 + \exp(-0.55037 + 0.15737X_1 + 0.19475X_2)}$$

p 值分别为

$$0.5628 \quad 0.0357 \quad 0.1117$$

可认为 X_1 对 $P(Y = 1)$ 有显著影响。预测值如下:

7.930776e-13 3.290103e-01 2.220446e-16 1.224829e-04 1.665939e-05 6.684758e-10
 7.972692e-04 2.220446e-16 8.682391e-01 3.739876e-10 1.118967e-06 1.118674e-13
 2.220446e-16 2.148712e-02 6.072085e-12 2.220446e-16 5.694457e-05 2.379878e-02
 1.100124e-03 8.784556e-06 2.211979e-05 9.335252e-03 2.013984e-09 5.541148e-11
 1.396926e-02 1.247900e-04 4.289639e-06 2.377275e-03 1.987758e-03 7.295078e-05
 2.395350e-02 2.019661e-05 1.963000e-01 9.999181e-01 9.999528e-01 4.278330e-01
 9.998776e-01 9.999041e-01 9.942752e-01 9.999028e-01 9.962502e-01 9.999976e-01
 9.999927e-01 9.999982e-01 9.999931e-01 9.415830e-01 9.999936e-01 9.970490e-01
 9.999937e-01 9.928044e-01 9.994180e-01 5.071745e-01 8.717798e-01 9.990596e-01
 9.999562e-01 9.995826e-01 9.904502e-01 9.999816e-01 9.999931e-01 9.999643e-01
 9.999907e-01 9.998978e-01 9.997753e-01 9.999619e-01 9.974965e-01 7.933926e-01

可见大部分预测比较成功。

3. 计算得概率分别为

$$P(Y = 1) = 1, \quad P(Y = 1) = 0.9938452$$

可见概率相差不大

所用代码如下

```
1 D <- read.table('exercise2_10.txt')
2 colnames(D) <- c('X1', 'X2', 'X3', 'y')
3 log.obj <- glm(y~., family = binomial(link = 'logit'), data = D)
4 model_summary <- summary(log.obj)
5 wald_values <- model_summary$coefficients[, "Pr(>|z|)"]
6 p_values <- model_summary$coefficients[, "Pr(>|z|)"]
7 wald_values
8 p_values
9
10 new.obj <- glm(y~ X1 + X2, family = binomial(link = 'logit'), data = D)
11 summary(new.obj)
12
```

```

13 predict.glm(new.obj, newdata = D[,c(1,2)], type = 'response')
14 new_data <- data.frame(X1 = 48.8, X2 = -10.5, X3 = 1.8)
15 predict(log.obj, newdata = new_data, type = "response")
16 predict(new.obj, newdata = new_data[,c(1,2)], type = "response")
17

```

□

第三章作业

3.1

Proof. 由于

$$\alpha_i = \mu_{i\cdot} - \mu = \frac{1}{b} \sum_{j=1}^b \mu_{ij} - \frac{1}{ab} \sum_{i=1}^a \sum_{j=1}^b \mu_{ij} = \frac{1}{b} \sum_{j=1}^b \left(\mu_{ij} - \frac{1}{a} \sum_{i=1}^a \mu_{ij} \right)$$

故

$$\sum_{i=1}^a \alpha_i = \sum_{i=1}^a \frac{1}{b} \sum_{j=1}^b \left(\mu_{ij} - \frac{1}{a} \sum_{i=1}^a \mu_{ij} \right) = \frac{1}{b} \sum_{j=1}^b \left(\sum_{i=1}^a \mu_{ij} - a \cdot \frac{1}{a} \sum_{i=1}^a \mu_{ij} \right) = 0.$$

同理，

$$\sum_{j=1}^b \beta_j = 0.$$

另一方面，由于

$$\gamma_{ij} = (\mu_{ij} - \mu) - (\alpha_i + \beta_j), \quad \beta_j = \mu_{\cdot j} - \mu, \quad \mu_{\cdot j} = \frac{1}{a} \sum_{i=1}^a \mu_{ij},$$

故

$$\sum_{i=1}^a \gamma_{ij} = \sum_{i=1}^a \mu_{ij} - a\mu - \sum_{i=1}^a \alpha_i - a\beta_j = \sum_{i=1}^a \mu_{ij} - a\mu - a(\mu_{\cdot j} - \mu) = \sum_{i=1}^a \mu_{ij} - a\mu_{\cdot j} = 0.$$

同理

$$\sum_{j=1}^b \gamma_{ij} = 0$$

□

3.2

Proof. 由于

$$\begin{aligned}
SS_A &= bc \sum_i (\alpha_i + \bar{\varepsilon}_{i\cdot} - \bar{\varepsilon})^2 = bc \sum_i \alpha_i^2 + 2bc \sum_i \alpha_i (\bar{\varepsilon}_{i\cdot} - \bar{\varepsilon}) + bc \sum_i (\bar{\varepsilon}_{i\cdot} - \bar{\varepsilon})^2 \\
&= bc \sum_i \alpha_i^2 + 2bc \sum_i \alpha_i (\bar{\varepsilon}_{i\cdot} - \bar{\varepsilon}) + bc \sum_i \bar{\varepsilon}_{i\cdot}^2 - 2bc \sum_i \bar{\varepsilon}_{i\cdot} \bar{\varepsilon} + bc \sum_i \bar{\varepsilon}^2 \\
&= bc \sum_i \alpha_i^2 + 2bc \sum_i \alpha_i (\bar{\varepsilon}_{i\cdot} - \bar{\varepsilon}) + bc \sum_i \bar{\varepsilon}_{i\cdot}^2 - abc \bar{\varepsilon}^2
\end{aligned}$$

其中用到了

$$\sum_i \bar{\varepsilon}_{i\cdot} = \frac{1}{bc} \sum_{i,j,k} \varepsilon_{ijk} = \frac{1}{a} \bar{\varepsilon}.$$

而

$$\bar{\varepsilon}_{i\cdot} \sim \mathcal{N}\left(0, \frac{\sigma^2}{bc}\right), \quad \bar{\varepsilon} \sim \mathcal{N}\left(0, \frac{\sigma^2}{abc}\right)$$

故

$$\begin{aligned}\mathbb{E}SS_A &= bc \sum_i \alpha_i^2 + bc \sum_i \mathbb{E}\bar{\epsilon}_{i..}^2 - abc \mathbb{E}\bar{\epsilon}^2 = bc \sum_i \alpha_i^2 + abc \text{Var } \bar{\epsilon}_{i..} - abc \text{Var } \bar{\epsilon} \\ &= bc \sum_i \alpha_i^2 + (a-1)\sigma^2.\end{aligned}$$

同理有

$$\mathbb{E}SS_B = (b-1)\sigma^2 + ac \sum_{j=1}^b \beta_j^2.$$

$$\begin{aligned}SS_{AB} &= c \sum_{i,j} (\gamma_{ij} + \bar{\epsilon}_{ij.} - \bar{\epsilon}_{i..} - \bar{\epsilon}_{.j.} + \bar{\epsilon})^2 \\ &= c \sum_{i,j} \gamma_{ij}^2 + c \sum_{i,j} \gamma_{ij} (\bar{\epsilon}_{ij.} - \bar{\epsilon}_{i..} - \bar{\epsilon}_{.j.} + \bar{\epsilon}) + c \sum_{i,j} (\bar{\epsilon}_{ij.} - \bar{\epsilon}_{i..} - \bar{\epsilon}_{.j.} + \bar{\epsilon})^2\end{aligned}$$

而

$$\gamma_{ij} \sim \mathcal{N}(0, \sigma^2) \text{ i.i.d} \quad \implies \quad \mathbb{E} \left(c \sum_{i,j} \gamma_{ij}^2 + c \sum_{i,j} \gamma_{ij} (\bar{\epsilon}_{ij.} - \bar{\epsilon}_{i..} - \bar{\epsilon}_{.j.} + \bar{\epsilon}) \right) = c \sum_{i,j} \gamma_{ij}^2$$

故只需要计算 SS_{AB} 的最后一项即可。事实上

$$\sum_{i,j} (\bar{\epsilon}_{ij.} - \bar{\epsilon}_{i..} - \bar{\epsilon}_{.j.} + \bar{\epsilon})^2 = \sum_{i,j} (\bar{\epsilon}_{ij.}^2 - \bar{\epsilon}_{i..}^2 - \bar{\epsilon}_{.j.}^2 + \bar{\epsilon}^2) + 2 \sum_{i,j} (-\bar{\epsilon}_{ij.}\bar{\epsilon}_{i..} - \bar{\epsilon}_{ij.}\bar{\epsilon}_{.j.} + \bar{\epsilon}_{ij.}\bar{\epsilon} + \bar{\epsilon}_{i..}\bar{\epsilon}_{.j.} - \bar{\epsilon}_{i..}\bar{\epsilon} - \bar{\epsilon}_{.j.}\bar{\epsilon})$$

而

$$\begin{aligned}\sum_j \bar{\epsilon}_{ij.} &= \frac{1}{c} \sum_{j,k} \epsilon_{ijk} = b\bar{\epsilon}_{i..}, & \sum_i \bar{\epsilon}_{ij.} &= \frac{1}{c} \sum_{i,k} \epsilon_{ijk} = a\bar{\epsilon}_{.j.}, & \sum_{i,j} \bar{\epsilon}_{ij.} &= \frac{1}{c} \sum_{i,j,k} \epsilon_{ijk} = ab\bar{\epsilon} \\ \sum_i \bar{\epsilon}_{i..} &= \frac{1}{bc} \sum_{i,j,k} \epsilon_{ijk} = a\bar{\epsilon} & \sum_j \bar{\epsilon}_{.j.} &= \frac{1}{ac} \sum_{i,j,k} \epsilon_{ijk} = b\bar{\epsilon}\end{aligned}$$

故有

$$\begin{aligned}& \sum_{i,j} (-\bar{\epsilon}_{ij.}\bar{\epsilon}_{i..} - \bar{\epsilon}_{ij.}\bar{\epsilon}_{.j.} + \bar{\epsilon}_{ij.}\bar{\epsilon} + \bar{\epsilon}_{i..}\bar{\epsilon}_{.j.} - \bar{\epsilon}_{i..}\bar{\epsilon} - \bar{\epsilon}_{.j.}\bar{\epsilon}) \\ &= -\sum_i \bar{\epsilon}_{i..} - \sum_j \bar{\epsilon}_{.j.} + ab\bar{\epsilon}^2 + ab\bar{\epsilon}^2 - ab\bar{\epsilon}^2 - ab\bar{\epsilon}^2 = -\sum_i \bar{\epsilon}_{i..} - \sum_j \bar{\epsilon}_{.j.}.\end{aligned}$$

故

$$\begin{aligned}\mathbb{E}SS_{AB} &= c \sum_{i,j} \gamma_{ij}^2 + c \sum_{i,j} (\mathbb{E}\bar{\epsilon}_{ij.}^2 - \mathbb{E}\bar{\epsilon}_{i..}^2 - \mathbb{E}\bar{\epsilon}_{.j.}^2 + \mathbb{E}\bar{\epsilon}^2) - 2c \left(\sum_i \mathbb{E}\bar{\epsilon}_{i..} + \sum_j \mathbb{E}\bar{\epsilon}_{.j.} \right) \\ &= c \sum_{i,j} \gamma_{ij}^2 + abc\sigma^2 \left(\frac{1}{c} - \frac{1}{bc} - \frac{1}{ac} + \frac{1}{abc} \right) + 0 \\ &= c \sum_{i,j} \gamma_{ij}^2 + \sigma^2(a-1)(b-1).\end{aligned}$$

最后

$$\begin{aligned}SS_E &= \sum_{i,j,k} (\epsilon_{ijk} - \bar{\epsilon}_{ij.})^2 = \sum_{i,j,k} \epsilon_{ijk}^2 - 2 \sum_{i,j,k} \epsilon_{ijk} \bar{\epsilon}_{ij.} + \sum_{i,j,k} \bar{\epsilon}_{ij.}^2 = \sum_{i,j,k} \epsilon_{ijk}^2 + \sum_{i,j,k} \bar{\epsilon}_{ij.}^2 - 2 \sum_{i,j} \bar{\epsilon}_{ij.} \sum_k \epsilon_{ijk} \\ &= \sum_{i,j,k} \epsilon_{ijk}^2 + \sum_{i,j,k} \bar{\epsilon}_{ij.}^2 - 2c \sum_{i,j} \bar{\epsilon}_{ij.}^2.\end{aligned}$$

从而

$$\mathbb{E}SS_E = abc \frac{\sigma^2}{1} + abc \frac{\sigma^2}{c} - 2abc \cdot \frac{\sigma^2}{c} = ab(c-1)\sigma^2.$$

□

3.5

Solution.

1. 带入数据得检验的 p 值为 4.33×10^{-5} 远远小于 $\alpha = 0.05$ 。从而拒绝原假设认为有显著影响。
2. 置信区间分别为

$$[6.25239, 7.503166], \quad [7.652239, 8.614427] \quad [8.289951, 10.11005]$$

同时置信区间分别为

$$[-3.793829, 1.282718] \quad [-4.431541, -0.2129032], \quad [-3.031692, 0.8983584]$$

可以看出投入经费越高，生产能力改善月显著。

所用代码如下

```

1 group1 <- c(7.6, 8.2, 6.8, 5.8, 6.9, 6.6, 6.3, 7.7, 6.0)
2 group2 <- c(6.7, 8.1, 9.4, 8.6, 7.8, 7.7, 8.9, 7.9, 8.3, 8.7, 7.1, 8.4)
3 group3 <- c(8.5, 9.7, 10.1, 7.8, 9.6, 9.5)
4
5 # 进行方差分析
6 model <- aov(c(group1, group2, group3) ~ rep(c("L", "M", "H"), c(length(group1), length(group2), length(group3))))
7
8 # 查看方差分析结果
9 summary(model)
10
11 # 设定置信水平为95%
12 alpha <- 0.05
13
14 # 计算每个组均值的置信区间
15 mean_group1 <- mean(group1)
16 mean_group2 <- mean(group2)
17 mean_group3 <- mean(group3)
18
19 se_group1 <- sd(group1) / sqrt(length(group1))
20 se_group2 <- sd(group2) / sqrt(length(group2))
21 se_group3 <- sd(group3) / sqrt(length(group3))
22
23 ci_group1 <- c(mean_group1 - qt(1 - alpha / 2, df = length(group1) - 1) * se_group1,
24               mean_group1 + qt(1 - alpha / 2, df = length(group1) - 1) * se_group1)
25 ci_group2 <- c(mean_group2 - qt(1 - alpha / 2, df = length(group2) - 1) * se_group2,
26               mean_group2 + qt(1 - alpha / 2, df = length(group2) - 1) * se_group2)
27 ci_group3 <- c(mean_group3 - qt(1 - alpha / 2, df = length(group3) - 1) * se_group3,
28               mean_group3 + qt(1 - alpha / 2, df = length(group3) - 1) * se_group3)
29
30 # 计算差值的Bonferroni校正置信区间
31 diff_ci_group1_group2 <- ci_group1 - ci_group2
32 diff_ci_group1_group3 <- ci_group1 - ci_group3
33 diff_ci_group2_group3 <- ci_group2 - ci_group3
34
35 # 进行Bonferroni校正
36 alpha_bonferroni <- alpha / 3 # 对每个置信区间进行调整
37
38 ci_diff_group1_group2_bonferroni <- diff_ci_group1_group2 + c(-1, 1) * qt(1 - alpha_bonferroni / 2, df = Inf)
39 ci_diff_group1_group3_bonferroni <- diff_ci_group1_group3 + c(-1, 1) * qt(1 - alpha_bonferroni / 2, df = Inf)

```

```

40 ci_diff_group2_group3_bonferroni <- diff_ci_group2_group3 + c(-1, 1) * qt(1 - alpha_bonferroni
    / 2, df = Inf)
41
42 # 输出结果
43 cat("Group 1 mean CI (95%):", ci_group1, "\n")
44 cat("Group 2 mean CI (95%):", ci_group2, "\n")
45 cat("Group 3 mean CI (95%):", ci_group3, "\n")
46 cat("Group 1 - Group 2 difference CI (Bonferroni adjusted, 95%):",
    ci_diff_group1_group2_bonferroni, "\n")
47 cat("Group 1 - Group 3 difference CI (Bonferroni adjusted, 95%):",
    ci_diff_group1_group3_bonferroni, "\n")
48 cat("Group 2 - Group 3 difference CI (Bonferroni adjusted, 95%):",
    ci_diff_group2_group3_bonferroni, "\n")
49

```

□

3.6

Solution.

1. 样本均值和标准差如下:

	Fe^{2+}	Fe^{2+}		Fe^{2+}	Fe^{2+}
高剂量	3.698889	5.936667	高剂量	2.030870	2.806778
中剂量	8.203889	9.632222	中剂量	5.447386	6.691215
低剂量	11.750000	12.639444	低剂量	7.028150	6.082089

表 1: 样本均值与标准差

标准差较为明显, 可见假定误差的等发性不合理

2. 变换后的样本均值和标准差如下:

	Fe^{2+}	Fe^{2+}		Fe^{2+}	Fe^{2+}
高剂量	1.160924	1.680129	高剂量	0.5854773	0.4645464
中剂量	1.901225	2.090045	中剂量	0.6585116	0.5736511
低剂量	2.279981	2.403389	低剂量	0.6563113	0.5693701

表 2: 变换后的样本均值与标准差

此时标准差趋于一致。

3. 方差分析表如下

	Sum Sq	Mean Sq	F value	Pr(>F)
Fe^{2+}	15.59	7.794	22.524	7.91e-09 ***
Fe^{3+}	2.07	2.074	5.993	0.0161 *
$Fe^{2+} : Fe^{3+}$	0.81	0.405	1.171	0.3143

表 3: 方差分析表

由于 $0.3143 > \alpha$, 可见交互效应不显著, 而各自影响较为显著。

4. 关于剂量的置信区间与同时置信区间分别为：(顺序为：高 + 中，高 + 低，中 + 低)

$[-0.7996574, -0.3505595]$ $[0.4978426, 0.9469405]$ $[1.0729510, 1.5220490]$

$[-0.8506702, -0.2995467]$ $[0.4468298, 0.9979533]$ $[1.0219382, 1.5730618]$

关于铁离子种类的置信区间与同时置信区间分别为

$[-0.5521593, -0.002128917]$, $[-0.592573, 0.03828476]$

可见剂量对存留量的影响差异比较显著，而铁离子种类对存留量的影响差异不大。

所用代码如下

```
1 Pre_data <- data.frame(X=c
2   (0.71,1.66,2.01,2.16,2.42,2.42,2.56,2.60,3.31,3.64,3.74,3.74,4.39,4.50,5.07,5.26,8.15,
3   8.24,2.20,2.93,3.08,3.49,4.11,4.95,5.16,5.54,5.68,6.25,7.25,7.90,8.85,11.96,15.54,15.89,
4   18.30,18.59,2.25,3.93,5.08,5.82,5.84,6.89,8.50,8.56,9.44,10.52,13.46,13.57,14.76,16.41,
5   16.96,17.56,22.82,29.13,2.20,2.69,3.54,3.75,3.83,4.08,4.27,4.53,5.32,6.18,6.22,6.33,6.97,
6   6.97,7.52,8.36,11.65,12.45,4.04,4.16,4.42,4.93,5.49,5.77,5.86,6.28,6.97,7.06,7.78,
7   9.23,9.34,9.91,13.46,18.40,23.89,26.39,2.71,5.43,6.38,6.38,8.32,9.04,9.56,10.01,10.08,
8   10.62,13.80,15.99,17.90,18.25,19.32,19.87,21.60,22.25),
9   A = gl(3,18,108),
10  B = gl(2,54,108))
11
12 model <- aov(X ~ A + B + A:B, data = Pre_data)
13 summary(model)
14 means <- aggregate(X ~ A + B, data = Pre_data, FUN = mean)
15 std_devs <- aggregate(X ~ A + B, data = Pre_data, FUN = sd)
16
17 print(means)
18 print(std_devs)
19
20 Pre_data$X <- log(Pre_data$X)
21 means <- aggregate(X ~ A + B, data = Pre_data, FUN = mean)
22 std_devs <- aggregate(X ~ A + B, data = Pre_data, FUN = sd)
23
24 print(means)
25 print(std_devs)
26
27 model <- aov(X ~ A + B + A:B, data = Pre_data)
28 summary(model)
29
30 attach(Pre_data)
31 mu.A = c(mean(X[A == 1]), mean(X[A == 2]), mean(X[A == 3]))
32 mu.B = c(mean(X[B == 1]), mean(X[B == 2]))
33 a <- 3
34 b <- 2
35 c <- 18
36 alpha = 0.05
37 mse <- sum(model$residuals^2) / (a * b * (c - 1))
38
39 index.martix <- matrix(c(1,1,2,2,3,3), nrow = 3, ncol = 2)
40 mean.diff.cfi <- matrix(0,3,2)
41 mean.diff.cfi
42 adj.alpha <- alpha / 3
43 index.martix[1,]
44 # A 置信区间
45 t_quan <- qt(1 - 0.5 * alpha, a * b * (c - 1))
46 for (i in 1:3) {
```

```

46 ind <- index.martix[i,]
47 mean.diff <- mu.A[ind[1]] - mu.A[ind[2]]
48 fac <- t_quan * sqrt(2 * mse / a / c)
49 mean.diff.cfi[i,1] <- mean.diff - fac
50 mean.diff.cfi[i,2] <- mean.diff + fac
51 }
52 mean.diff.cfi
53 # A 同时置信区间
54 t_quan <- qt(1 - 0.5 * adj.alpha, a * b * (c - 1))
55 for (i in 1:3) {
56 ind <- index.martix[i,]
57 mean.diff <- mu.A[ind[1]] - mu.A[ind[2]]
58 fac <- t_quan * sqrt(2 * mse / a / c)
59 mean.diff.cfi[i,1] <- mean.diff - fac
60 mean.diff.cfi[i,2] <- mean.diff + fac
61 }
62 mean.diff.cfi
63
64 # B 置信区间
65 t_quan <- qt(1 - 0.5 * alpha, a * b * (c - 1))
66 mu.B[1] - mu.B[2] - t_quan * sqrt(2 * mse / b / c)
67 mu.B[1] - mu.B[2] + t_quan * sqrt(2 * mse / b / c)
68 # B 同时置信区间
69 t_quan <- qt(1 - 0.5 * alpha / 2, a * b * (c - 1))
70 mu.B[1] - mu.B[2] - t_quan * sqrt(2 * mse / b / c)
71 mu.B[1] - mu.B[2] + t_quan * sqrt(2 * mse / b / c)
72

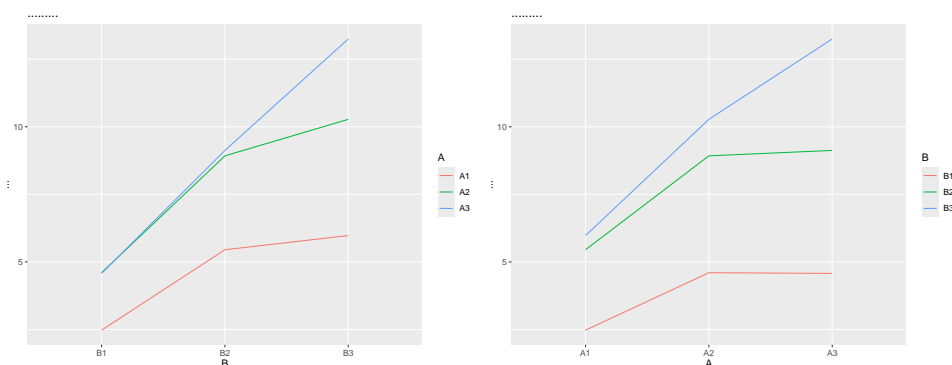
```

□

3.7

Solution.

1. 作出图像如下



2. 方差分析表如下

	Sum Sq	Mean Sq	F value	Pr(>F)
A	220.02	110.01	1827.9	<2e-16 ***
B	123.66	61.83	1027.3	<2e-16 ***
A:B	29.43	7.36	122.2	<2e-16 ***

可见交互效应显著。

3. B 各区间如下

	[,1]	[,2]		[,1]	[,2]		[,1]	[,2]
[1,]	-3.2806901	-2.6693099	[1,]	-4.63069	-4.01931	[1,]	-4.85569	-4.24431
[2,]	-3.8056901	-3.1943099	[2,]	-5.98069	-5.36931	[2,]	-8.98069	-8.36931
[3,]	-0.8306901	-0.2193099	[3,]	-1.65569	-1.04431	[3,]	-4.43069	-3.81931

图 1: B 各水平置信区间

	[,1]	[,2]		[,1]	[,2]		[,1]	[,2]
[1,]	-3.2806901	-2.6693099	[1,]	-4.63069	-4.01931	[1,]	-4.85569	-4.24431
[2,]	-3.8056901	-3.1943099	[2,]	-5.98069	-5.36931	[2,]	-8.98069	-8.36931
[3,]	-0.8306901	-0.2193099	[3,]	-1.65569	-1.04431	[3,]	-4.43069	-3.81931

图 2: B 各水平同时置信区间

A 各区间如下

	[,1]	[,2]		[,1]	[,2]		[,1]	[,2]
[1,]	-2.4306901	-1.8193099	[1,]	-3.7806901	-3.1693099	[1,]	-4.60569	-3.99431
[2,]	-2.4056901	-1.7943099	[2,]	-3.9806901	-3.3693099	[2,]	-7.58069	-6.96931
[3,]	-0.2806901	0.3306901	[3,]	-0.5056901	0.1056901	[3,]	-3.28069	-2.66931

图 3: A 各水平置信区间

	[,1]	[,2]		[,1]	[,2]		[,1]	[,2]
[1,]	-2.4306901	-1.8193099	[1,]	-3.7806901	-3.1693099	[1,]	-4.60569	-3.99431
[2,]	-2.4056901	-1.7943099	[2,]	-3.9806901	-3.3693099	[2,]	-7.58069	-6.96931
[3,]	-0.2806901	0.3306901	[3,]	-0.5056901	0.1056901	[3,]	-3.28069	-2.66931

图 4: A 各水平同时置信区间

所用代码如下

```

1 data1 <- data.frame(X_data = c(2.4,2.7,2.3,2.5, 4.6,4.2,4.9,4.7, 4.8,4.5,4.4,4.6,
2   5.8,5.2,5.5,5.3, 8.9,9.1,8.7,9.0, 9.1,9.3,8.7,9.4,
3   6.1,5.7,5.9,6.2, 9.9,10.5,10.6,10.1, 13.5,13.0,13.3,13.2),
4   A_con = gl(3,12,36),
5   B_con = gl(3,4,36))
6
7 attach(data1)
8 mu = matrix(0,3,3)
9 for (i in 1:3) {
10  for (j in 1:3){
11    mu[i,j] = mean(X_data[A_con == i & B_con == j])
12  }
13 }
14 mu
15
16 library(ggplot2)
17
18 # 创建示例数据
19 data <- data.frame(
20   B = c("B1", "B2", "B3"),
21   A1 = c(2.475, 5.450, 5.975),

```

```

22 A2 = c(4.600, 8.925, 10.275),
23 A3 = c(4.575, 9.125, 13.250)
24 )
25
26 # 将数据从宽格式转换为长格式
27 data_long <- tidyr::gather(data, key = "A", value = "value", -B)
28
29 # 绘制折线图
30 ggplot(data_long, aes(x = B, y = value, color = A, group = A)) +
31   geom_line() +
32   labs(title = "折线图", x = "B", y = "值")
33
34
35 data <- data.frame(
36   A = c("A1", "A2", "A3"),
37   B1 = c(2.475, 4.600, 4.575),
38   B2 = c(5.450, 8.925, 9.125),
39   B3 = c(5.975, 10.275, 13.250)
40 )
41
42 # 将数据从宽格式转换为长格式
43 data_long <- tidyr::gather(data, key = "B", value = "value", -A)
44
45 # 绘制折线图
46 ggplot(data_long, aes(x = A, y = value, color = B, group = B)) +
47   geom_line() +
48   labs(title = "折线图", x = "A", y = "值")
49
50 model <- aov(X_data ~ A_con + B_con + A_con:B_con, data = data1)
51 summary(model)
52
53 # adj-----
54 a <- 3
55 b <- 3
56 c <- 5
57 mse <- sum(model$residuals ^ 2) / a / b / (c - 1)
58 m <- 3
59 adj.alpha = 0.05 / m
60 t_tuan <- qt(1 - 0.5 * adj.alpha, a * b * (c - 1))
61 index.martix <- matrix(c(1, 1, 2, 2, 3, 3), nrow = 3, ncol = 2)
62 confi.int.B1 <- matrix(0, 3, 2)
63 mu.B <- mu[,1]
64
65 for (i in 1:3) {
66   ind <- index.martix[i,]
67   mean.diff <- mu.B[ind[1]] - mu.B[ind[2]]
68   fac <- t_tuan * sqrt(2 * mse / c)
69   confi.int.B1[i,1] <- mean.diff - fac
70   confi.int.B1[i,2] <- mean.diff + fac
71 }
72
73 confi.int.B1
74
75 a <- 3
76 b <- 3
77 c <- 5
78 mse <- sum(model$residuals ^ 2) / a / b / (c - 1)
79 m <- 3
80 adj.alpha = 0.05 / m

```

```

81 t_tuan <- qt (1 - 0.5 * adj.alpha, a * b * (c - 1))
82 index.martix <- matrix(c(1, 1, 2, 2, 3, 3), nrow = 3, ncol = 2)
83 confi.int.B1 <- matrix(0, 3, 2)
84 mu.B <- mu[,2]
85
86 for (i in 1:3) {
87   ind <- index.martix[i,]
88   mean.diff <- mu.B[ind[1]] - mu.B[ind[2]]
89   fac <- t_tuan * sqrt(2 * mse / c)
90   confi.int.B1[i,1] <- mean.diff - fac
91   confi.int.B1[i,2] <- mean.diff + fac
92 }
93
94 confi.int.B1
95
96 a <- 3
97 b <- 3
98 c <- 5
99 mse <- sum(model$residuals ^ 2) / a / b / (c - 1)
100 m <- 3
101 adj.alpha = 0.05 / m
102 t_tuan <- qt (1 - 0.5 * adj.alpha, a * b * (c - 1))
103 index.martix <- matrix(c(1, 1, 2, 2, 3, 3), nrow = 3, ncol = 2)
104 confi.int.B1 <- matrix(0, 3, 2)
105 mu.B <- mu[,3]
106
107 for (i in 1:3) {
108   ind <- index.martix[i,]
109   mean.diff <- mu.B[ind[1]] - mu.B[ind[2]]
110   fac <- t_tuan * sqrt(2 * mse / c)
111   confi.int.B1[i,1] <- mean.diff - fac
112   confi.int.B1[i,2] <- mean.diff + fac
113 }
114
115 confi.int.B1
116
117
118 # inter-----
119 a <- 3
120 b <- 3
121 c <- 5
122 mse <- sum(model$residuals ^ 2) / a / b / (c - 1)
123
124 m <- 3
125 adj.alpha = 0.05
126 t_tuan <- qt (1 - 0.5 * adj.alpha, a * b * (c - 1))
127 index.martix <- matrix(c(1, 1, 2, 2, 3, 3), nrow = 3, ncol = 2)
128 confi.int.B1 <- matrix(0, 3, 2)
129 mu.B <- mu[,1]
130
131 for (i in 1:3) {
132   ind <- index.martix[i,]
133   mean.diff <- mu.B[ind[1]] - mu.B[ind[2]]
134   fac <- t_tuan * sqrt(2 * mse / c)
135   confi.int.B1[i,1] <- mean.diff - fac
136   confi.int.B1[i,2] <- mean.diff + fac
137 }
138
139 confi.int.B1

```

```

140
141 a <- 3
142 b <- 3
143 c <- 5
144 mse <- sum(model$residuals ^ 2) / a / b / (c - 1)
145 m <- 3
146 adj.alpha = 0.05
147 t_tuan <- qt (1 - 0.5 * adj.alpha, a * b * (c - 1))
148 index.martix <- matrix(c(1, 1, 2, 2, 3, 3), nrow = 3, ncol = 2)
149 confi.int.B1 <- matrix(0, 3, 2)
150 mu.B <- mu[,2]
151
152 for (i in 1:3) {
153 ind <- index.martix[i,]
154 mean.diff <- mu.B[ind[1]] - mu.B[ind[2]]
155 fac <- t_tuan * sqrt(2 * mse / c)
156 confi.int.B1[i,1] <- mean.diff - fac
157 confi.int.B1[i,2] <- mean.diff + fac
158
159 }
160 confi.int.B1
161
162 a <- 3
163 b <- 3
164 c <- 5
165 mse <- sum(model$residuals ^ 2) / a / b / (c - 1)
166 m <- 3
167 adj.alpha = 0.05
168 t_tuan <- qt (1 - 0.5 * adj.alpha, a * b * (c - 1))
169 index.martix <- matrix(c(1, 1, 2, 2, 3, 3), nrow = 3, ncol = 2)
170 confi.int.B1 <- matrix(0, 3, 2)
171 mu.B <- mu[,3]
172
173 for (i in 1:3) {
174 ind <- index.martix[i,]
175 mean.diff <- mu.B[ind[1]] - mu.B[ind[2]]
176 fac <- t_tuan * sqrt(2 * mse / c)
177 confi.int.B1[i,1] <- mean.diff - fac
178 confi.int.B1[i,2] <- mean.diff + fac
179
180 }
181 confi.int.B1
182
183
184 mu <- t(mu)
185 # adj-----
186 a <- 3
187 b <- 3
188 c <- 5
189 mse <- sum(model$residuals ^ 2) / a / b / (c - 1)
190 m <- 3
191 adj.alpha = 0.05 / m
192 t_tuan <- qt (1 - 0.5 * adj.alpha, a * b * (c - 1))
193 index.martix <- matrix(c(1, 1, 2, 2, 3, 3), nrow = 3, ncol = 2)
194 confi.int.B1 <- matrix(0, 3, 2)
195 mu.B <- mu[,1]
196
197 for (i in 1:3) {
198 ind <- index.martix[i,]

```

```

199 mean.diff <- mu.B[ind[1]] - mu.B[ind[2]]
200 fac <- t_quan * sqrt(2 * mse / c)
201 confi.int.B1[i,1] <- mean.diff - fac
202 confi.int.B1[i,2] <- mean.diff + fac
203
204 }
205 confi.int.B1
206
207 a <- 3
208 b <- 3
209 c <- 5
210 mse <- sum(model$residuals ^ 2) / a / b / (c - 1)
211 m <- 3
212 adj.alpha = 0.05 / m
213 t_tuan <- qt (1 - 0.5 * adj.alpha, a * b * (c - 1))
214 index.martix <- matrix(c(1, 1, 2, 2, 3, 3), nrow = 3, ncol = 2)
215 confi.int.B1 <- matrix(0, 3, 2)
216 mu.B <- mu[,2]
217
218 for (i in 1:3) {
219 ind <- index.martix[i,]
220 mean.diff <- mu.B[ind[1]] - mu.B[ind[2]]
221 fac <- t_quan * sqrt(2 * mse / c)
222 confi.int.B1[i,1] <- mean.diff - fac
223 confi.int.B1[i,2] <- mean.diff + fac
224
225 }
226 confi.int.B1
227
228 a <- 3
229 b <- 3
230 c <- 5
231 mse <- sum(model$residuals ^ 2) / a / b / (c - 1)
232 m <- 3
233 adj.alpha = 0.05 / m
234 t_tuan <- qt (1 - 0.5 * adj.alpha, a * b * (c - 1))
235 index.martix <- matrix(c(1, 1, 2, 2, 3, 3), nrow = 3, ncol = 2)
236 confi.int.B1 <- matrix(0, 3, 2)
237 mu.B <- mu[,3]
238
239 for (i in 1:3) {
240 ind <- index.martix[i,]
241 mean.diff <- mu.B[ind[1]] - mu.B[ind[2]]
242 fac <- t_quan * sqrt(2 * mse / c)
243 confi.int.B1[i,1] <- mean.diff - fac
244 confi.int.B1[i,2] <- mean.diff + fac
245
246 }
247 confi.int.B1
248
249
250 # inter-----
251 a <- 3
252 b <- 3
253 c <- 5
254 mse <- sum(model$residuals ^ 2) / a / b / (c - 1)
255 m <- 3
256 adj.alpha = 0.05
257 t_tuan <- qt (1 - 0.5 * adj.alpha, a * b * (c - 1))

```

```

258 index.martix <- matrix(c(1, 1, 2, 2, 3, 3), nrow = 3, ncol = 2)
259 confi.int.B1 <- matrix(0, 3, 2)
260 mu.B <- mu[,1]
261
262 for (i in 1:3) {
263   ind <- index.martix[i,]
264   mean.diff <- mu.B[ind[1]] - mu.B[ind[2]]
265   fac <- t_quan * sqrt(2 * mse / c)
266   confi.int.B1[i,1] <- mean.diff - fac
267   confi.int.B1[i,2] <- mean.diff + fac
268 }
269 confi.int.B1
270
271 a <- 3
272 b <- 3
273 c <- 5
274
275 mse <- sum(model$residuals ^ 2) / a / b / (c - 1)
276 m <- 3
277 adj.alpha = 0.05
278 t_tuan <- qt (1 - 0.5 * adj.alpha, a * b * (c - 1))
279 index.martix <- matrix(c(1, 1, 2, 2, 3, 3), nrow = 3, ncol = 2)
280 confi.int.B1 <- matrix(0, 3, 2)
281 mu.B <- mu[,2]
282
283 for (i in 1:3) {
284   ind <- index.martix[i,]
285   mean.diff <- mu.B[ind[1]] - mu.B[ind[2]]
286   fac <- t_quan * sqrt(2 * mse / c)
287   confi.int.B1[i,1] <- mean.diff - fac
288   confi.int.B1[i,2] <- mean.diff + fac
289 }
290 confi.int.B1
291
292 a <- 3
293 b <- 3
294 c <- 5
295
296 mse <- sum(model$residuals ^ 2) / a / b / (c - 1)
297 m <- 3
298 adj.alpha = 0.05
299 t_tuan <- qt (1 - 0.5 * adj.alpha, a * b * (c - 1))
300 index.martix <- matrix(c(1, 1, 2, 2, 3, 3), nrow = 3, ncol = 2)
301 confi.int.B1 <- matrix(0, 3, 2)
302 mu.B <- mu[,3]
303
304 for (i in 1:3) {
305   ind <- index.martix[i,]
306   mean.diff <- mu.B[ind[1]] - mu.B[ind[2]]
307   fac <- t_quan * sqrt(2 * mse / c)
308   confi.int.B1[i,1] <- mean.diff - fac
309   confi.int.B1[i,2] <- mean.diff + fac
310 }
311 confi.int.B1
312

```

□