

Chapter 10 Question 4.

Solution 1. we copy the data into a .csv file and read it:

```
#read the data
setwd("C:\\Akira\\R")
resp = read.csv("resp.csv", header = TRUE)
```

For part (a):

Our model here is the randomized complete block design:

$$Y_{hi} = \mu + \theta_h + \tau_i + \epsilon_{hi}$$

with $h = 1, \dots, 9; i = 1, 2, 3$.

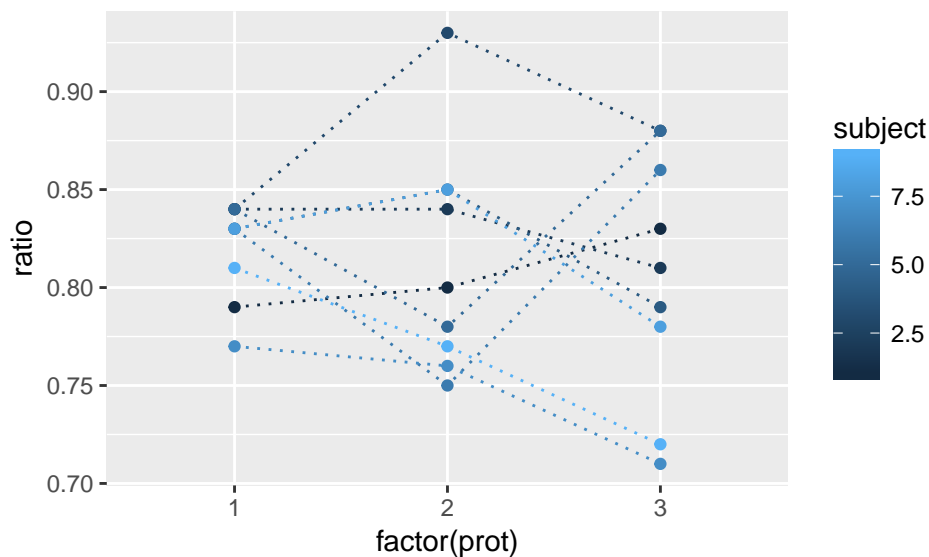
The model assumption is $\epsilon_{hi} \sim N(0, \sigma^2)$ i.i.d.

We fit the data into the model:

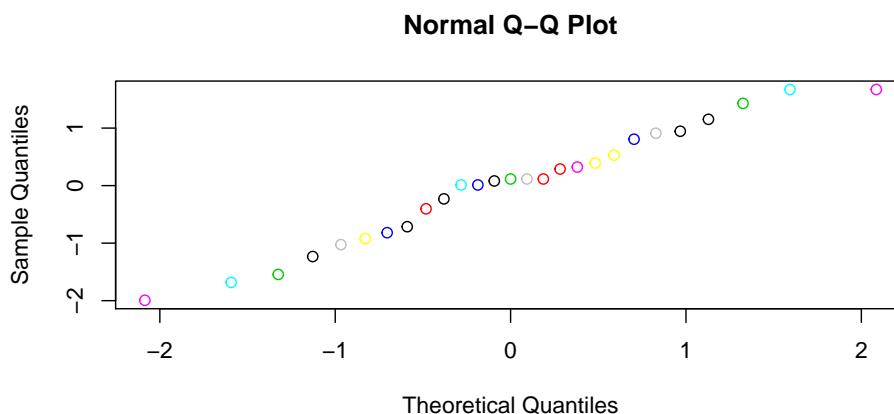
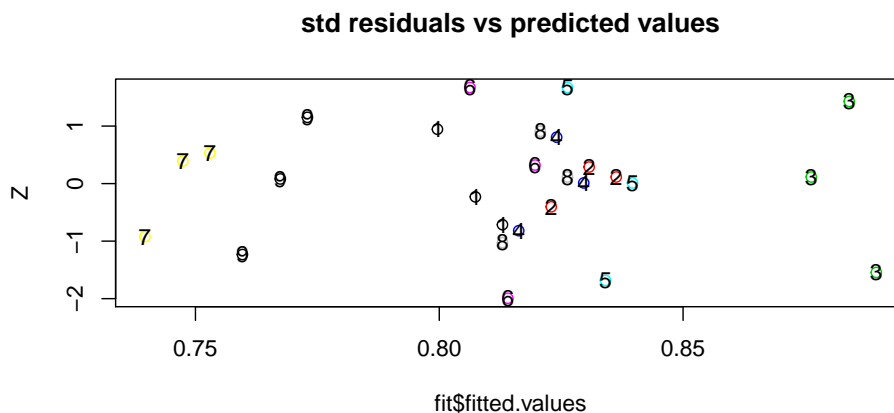
```
fit <- lm(ratio~ factor(subject) + factor(prot), data=resp)
#anova(fit)
SSE <- sum((resp$ratio - fit$fitted.values)^2)
Z = fit$residuals/sqrt(SSE/26)
```

To evaluate the assumptions for the block-treatment model above, we give following plots:

```
#par(mfrow = c(2, 2))
library(ggplot2)
ggplot(data=resp, aes(x=factor(prot), y = ratio, color = subject))+
  geom_line(aes(group=subject), linetype = "dotted") +geom_point()
```



```
par(mfrow = c(2, 1))
cols = factor(resp$subject)
plot(fit$fitted.values, Z, col=as.character(cols))
title(main = "std residuals vs predicted values")
text(fit$fitted.values, Z, labels=resp$subject)
qqnorm(Z, col=as.character(cols))
```



From $\text{ratio}(\text{response})$ versus $\text{protocol}(\text{treatment})$ for each $\text{block}(\text{subject})$, since the relationships are not parallel, this indicates a possible interaction between block and treatment factors.

We do not have info for the run order, so we can not test on independence.

From the standardized residuals against predicted value, we do not see much pattern, this shows support for equal variance. However due to small data here, this judgement may not be really valid.

The qq norm plot or standardized residuals almost forms a straightline, which shows support for normal assumption.

For part (b):

construct the ANOVA table:

```
anova(fit)

## Analysis of Variance Table
##
## Response: ratio
##           Df    Sum Sq   Mean Sq F value   Pr(>F)
## factor(subject)  8 0.037163 0.0046454  2.7672 0.03958 *
## factor(prot)     2 0.000807 0.0004037  0.2405 0.78904
## Residuals       16 0.026859 0.0016787
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The p value for testing equal effects of protocols is 0.78904 here. If we follow this we will fail to reject H_0 . But we need to point out that just as example 10.4.1, a p value this large is unusual. Notice we have a really small ratio of $msT/msE = 0.24$, which says the the average variability of measurement from one treatment to another is about four times smaller than the measurement error variability. However the former should include the latter and hence the ratio should be larger than 1. Also a p value of 0.79 says that there is only about 21% chance we get a ratio small like this under the null. This raise the doubt of model fitness. Maybe there is interaction between block and treatment.

For part (c):

To evaluate the usefulness of blocking, since it is not randomly assigned to experiment units, so rather than testing the equality of block effects, we only compare the block mean square $ms\theta$ with the error mean square msE . Our ratio here is 2.7672, and we consider it large enough to show the usefulness of blocking.

For part (d):

our confidence interval for any contrast under the randomized complete block design is:

$$\sum c_i \tau_i \in \left(\sum c_i \bar{y}_{.i} \pm w_S \sqrt{msE \sum c_i^2 / b} \right)$$

The critical coefficient for Scheffe method is:

$$w_S = \sqrt{(g-1)F_{g-1, bg-b-g+1, \alpha}}$$

In our case, we have $g = 3, b = 9, \alpha = 0.01$, so

$$w_S = \sqrt{2F_{2,16,0.01}} = 3.528806$$

```
w_S <- sqrt(2*qf(0.01, 2, 16, lower.tail = FALSE))
w_S

## [1] 3.528806
```

we compute the confidence interval as follows:

```
a<-c(1, -1, 0)
b<-c(1, 0, -1)
c<-c(0, 1, -1)
d<-c(1, -0.5, -0.5)
contrast <- list(a, b, c, d)
n <- length(contrast)
msE <- anova(fit)[[3]][[3]]
msE

## [1] 0.001678704

y <- c()
for (i in 1:3){
y[i] <- mean(resp[resp$prot==i,]$ratio)
}
for (i in 1:n){
#compute least square estimate
lse <- sum(contrast[[i]]*y)
#compute msd
msd <- w_S*sqrt(msE*sum(contrast[[i]]^2)/9)
#compute lower and upper bound for CI
ci_lower<- lse-msd
ci_upper<- lse+msd
#print the result of CI
cat("The CI for (", contrast[[i]], ") is (",
    ci_lower<- lse-msd, ", ", ci_upper<- lse+msd, ")", "\n")
}

## The CI for ( 1 -1 0 ) is ( -0.06260114 , 0.07371225 )
## The CI for ( 1 0 -1 ) is ( -0.05482336 , 0.08149003 )
## The CI for ( 0 1 -1 ) is ( -0.06037892 , 0.07593447 )
## The CI for ( 1 -0.5 -0.5 ) is ( -0.04958099 , 0.06846987 )
```