

STA305 FP RMD

Yiliu Cao

2023-04-06

```
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.2 --
## v ggplot2 3.3.6      v purrr  0.3.4
## v tibble  3.1.8      v dplyr  1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.2      v forcats 0.5.2
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
library(knitr)
```

Data

We firstly need to generate the data in this experiment, it includes the design matrix and the outcome.
For the design matrix:

```
# Set up the factorial design with complete blocks
# The default number of block is 2 and the number of factor is 3
n <- 50
x1 <- rep(c(-1, 1), 2^3*n)
x2 <- rep(c(-1, -1, 1, 1), 2^3*n/2)
x3 <- rep(c(rep(-1, 4), rep(1, 4)), 2^3*n/4)

# Distinguish the blocks for each observation
blk <- (c(rep(1, 2^3*n), rep(2, 2^3*n)))
run <- c(1:(2^3*n*2))
data <- data.frame(Run = run,
                   M = x1,
                   O = x2,
                   P = x3,
                   Block = blk)
```

Generate the outcome:

```
# Set up the outcome
set.seed(305)
# Different factor has different effects on the outcome
rating <- c()
for (i in 1:nrow(data)){
  basic <- 50
  if (data[i,2] == 1){
    basic <- basic + rnorm(1, 15, 5)
  }
}
```

```

}
if (data[i,3] == 1){
  basic <- basic + rnorm(1, 15, 10)
}
if (data[i,4] == 1){
  basic <- basic + rnorm(1, 10, 5)
}
if (basic > 100){
  basic <- 100
}
else if (basic < 0){
  basic <- 0
}
rating[i] <- round(basic, 0)
}
data$Rating <- rating
# You can choose to export the data set
# write.csv(data, "data_FP.csv", row.names = FALSE)

```

Methods

We firstly can pre-check the interactions of all paired factor by the interaction plots.

```

interaction.plot(
data$M,
data$O,
data$Rating,
trace.label = "Onions",
xlab = "Mushrooms",
ylab = "Rating",
main = "Figure 1: The interaction between Mushrooms and Onions"
)

```

Figure 1: The interaction between Mushrooms and Onions



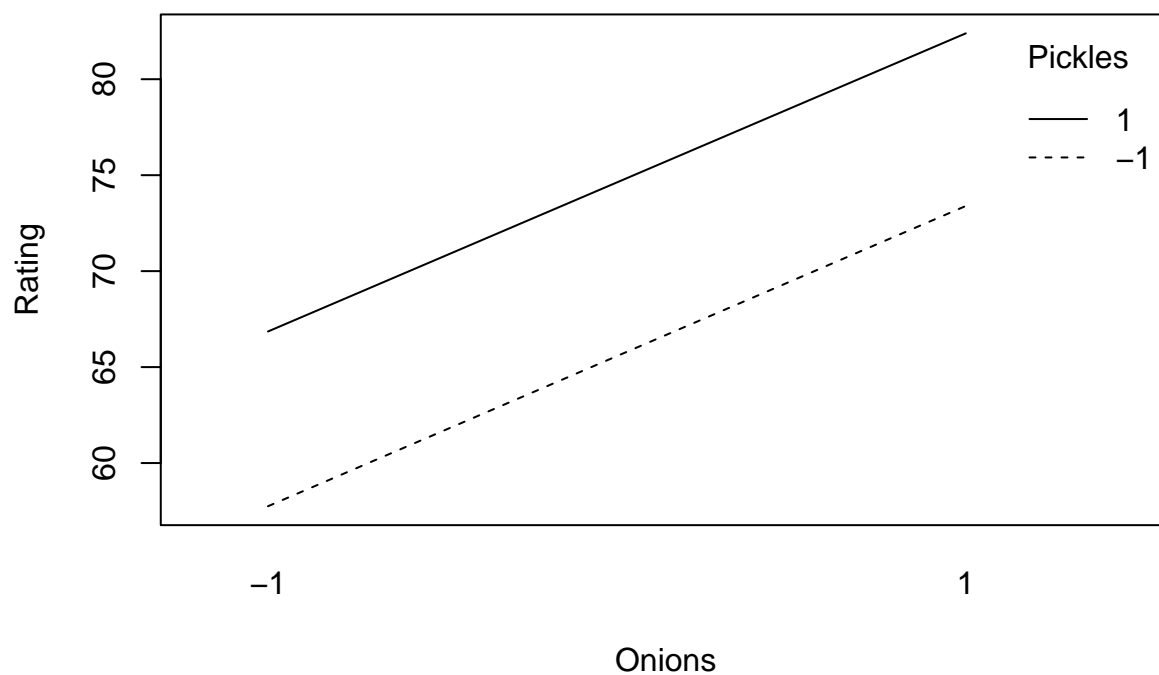
```
interaction.plot(  
  data$M,  
  data$P,  
  data$Rating,  
  trace.label = "Pickles",  
  xlab = "Mushrooms",  
  ylab = "Rating",  
  main = "Figure 2: The interaction between Mushrooms and Pickles"  
)
```

Figure 2: The interaction between Mushrooms and Pickles



```
interaction.plot(  
  data$O,  
  data$P,  
  data$Rating,  
  trace.label = "Pickles",  
  xlab = "Onions",  
  ylab = "Rating",  
  main = "Figure 3: The interaction between Onions and Pickles"  
)
```

Figure 3: The interaction between Onions and Pickles



Then we can implement the statistical analysis by starting from constructing the full model fit which contains all the effects and the block.

```
model_full <- lm(Rating ~ M*O*P + as.factor(Block), data = data)
```

Then we can build tables using the function `kable()` for both the summary table and ANOVA table. For the summary table:

```
model_f_lm <- summary(model_full)
tibble(source = c("Intercept", "Mushrooms", "Onions", "Pickles",
                  "Mushrooms:Onions", "Mushrooms:Pickles", "Onions:Pickles",
                  "Mushrooms:Onions:Pickles", "Blocks"),
       estimate = round(model_f_lm$coefficients[, "Estimate"][c(1:4, 6:9, 5)], 4),
       se = round(model_f_lm$coefficients[, "Std. Error"][c(1:4, 6:9, 5)], 4),
       t = round(model_f_lm$coefficients[, "t value"][c(1:4, 6:9, 5)], 3),
       p = round(model_f_lm$coefficients[, "Pr(>|t|)"][c(1:4, 6:9, 5)], 5)) %>%
  rename(`Effects` = source,
         `Estimate` = estimate,
         `Standard Error` = se,
         `T value` = t,
         `P values` = p) %>%
  kable(caption = "The summary of the full model")
```

Table 1: The summary of the full model

Effects	Estimate	Standard Error	T value	P values
Intercept	70.7200	0.4199	168.412	0.00000
Mushrooms	7.0675	0.2969	23.802	0.00000
Onions	7.7950	0.2969	26.252	0.00000
Pickles	4.5250	0.2969	15.239	0.00000
Mushrooms:Onions	-0.8000	0.2969	-2.694	0.00720
Mushrooms:Pickles	-0.1450	0.2969	-0.488	0.62545
Onions:Pickles	-0.0275	0.2969	-0.093	0.92623
Mushrooms:Onions:Pickles	-0.2625	0.2969	-0.884	0.37694
Blocks	-1.2450	0.5939	-2.096	0.03636

For the ANOVA table:

```
anova_f <- anova(model_full)
tibble(source = c("Mushrooms", "Onions", "Pickles", "Mushrooms:Onions",
                  "Mushrooms:Pickles", "Onions:Pickles",
                  "Mushrooms:Onions:Pickles", "Blocks", "Residuals"),
       df = anova_f$Df[c(1:3, 5:8, 4, 9)],
       ss = round(anova_f$`Sum Sq`[c(1:3, 5:8, 4, 9)], 0),
       ms = round(anova_f$`Mean Sq`[c(1:3, 5:8, 4, 9)], 0),
       f = round(anova_f$`F value`[c(1:3, 5:8, 4, 9)], 4),
       p = round(anova_f$`Pr(>F)`[c(1:3, 5:8, 4, 9)], 5)) %>%
  rename(`Effects` = source,
         `Degree of Freedom` = df,
         `Sum of Squares` = ss,
         `Mean Squares` = ms,
         `F value` = f,
         `P value` = p) %>%
  kable(caption = "The ANOVA table of the full model")
```

Table 2: The ANOVA table of the full model

Effects	Degree of Freedom	Sum of Squares	Mean Squares	F value	P value
Mushrooms	1	39960	39960	566.5335	0.00000
Onions	1	48610	48610	689.1697	0.00000
Pickles	1	16381	16381	232.2368	0.00000
Mushrooms:Onions	1	512	512	7.2590	0.00720
Mushrooms:Pickles	1	17	17	0.2385	0.62545
Onions:Pickles	1	1	1	0.0086	0.92623
Mushrooms:Onions:Pickles	1	55	55	0.7815	0.37694
Blocks	1	310	310	4.3951	0.03636
Residuals	791	55792	71	NA	NA

From the above linear model, using the significance level 0.05, the significant effects are M, P, O and M:O. We can refit a new model containing only these effects, and repeat the same process. Construct the reduced model:

```
model_reduced <- lm(Rating ~ M + O + P + M:O + as.factor(Block), data = data)
```

Construct the summary table using kable()

```
model_r_lm <- summary(model_reduced)
tibble(source = c("Intercept", "Mushrooms", "Onions", "Pickles", "Mushrooms:Onions",
                  "Blocks"),
       estimate = round(model_r_lm$coefficients[, "Estimate"][c(1:4, 6, 5)], 4),
       se = round(model_r_lm$coefficients[, "Std. Error"][c(1:4, 6, 5)], 4),
       t = round(model_r_lm$coefficients[, "t value"][c(1:4, 6, 5)], 3),
       p = round(model_r_lm$coefficients[, "Pr(>|t|)"][c(1:4, 6, 5)], 5)) %>%
  rename(`Effects` = source,
         `Estimate` = estimate,
         `Standard Error` = se,
         `T value` = t,
         `P values` = p) %>%
  kable(caption = "The summary of the reduced model")
```

Table 3: The summary of the reduced model

Effects	Estimate	Standard Error	T value	P values
Intercept	70.7200	0.4194	168.622	0.00000
Mushrooms	7.0675	0.2966	23.832	0.00000
Onions	7.7950	0.2966	26.285	0.00000
Pickles	4.5250	0.2966	15.258	0.00000
Mushrooms:Onions	-0.8000	0.2966	-2.698	0.00713
Blocks	-1.2450	0.5931	-2.099	0.03613

Construct the ANOVA table using kable()

```
anova_r <- anova(model_reduced)
tibble(source = c("Mushrooms", "Onions", "Pickles", "Mushrooms:Onions",
                  "Blocks", "Residuals"),
       df = anova_r$Df[c(1:3, 5, 4, 6)],
       ss = anova_r$`Sum Sq`[c(1:3, 5, 4, 6)],
       ms = anova_r$`Mean Sq`[c(1:3, 5, 4, 6)],
```

```

f = anova_r$`F value`[c(1:3, 5, 4, 6)],
p = anova_r$`Pr(>F)`[c(1:3, 5, 4, 6))] %>%
rename(`Source` = source,
       `Degree of Freedom` = df,
       `Sum of Squares` = ss,
       `Mean Squares` = ms,
       `F value` = f,
       `P value` = p) %>%
kable(caption = "The ANOVA table of the reduced model")

```

Table 4: The ANOVA table of the reduced model

Source	Degree of Freedom	Sum of Squares	Mean Squares	F value	P value
Mushrooms	1	39959.645	39959.64500	567.943634	0.0000000
Onions	1	48609.620	48609.62000	690.885122	0.0000000
Pickles	1	16380.500	16380.50000	232.814898	0.0000000
Mushrooms:Onions	1	512.000	512.00000	7.277020	0.0071324
Blocks	1	310.005	310.00500	4.406079	0.0361263
Residuals	794	55864.625	70.35847	NA	NA

Then we need to check the assumptions of ANOVA table:

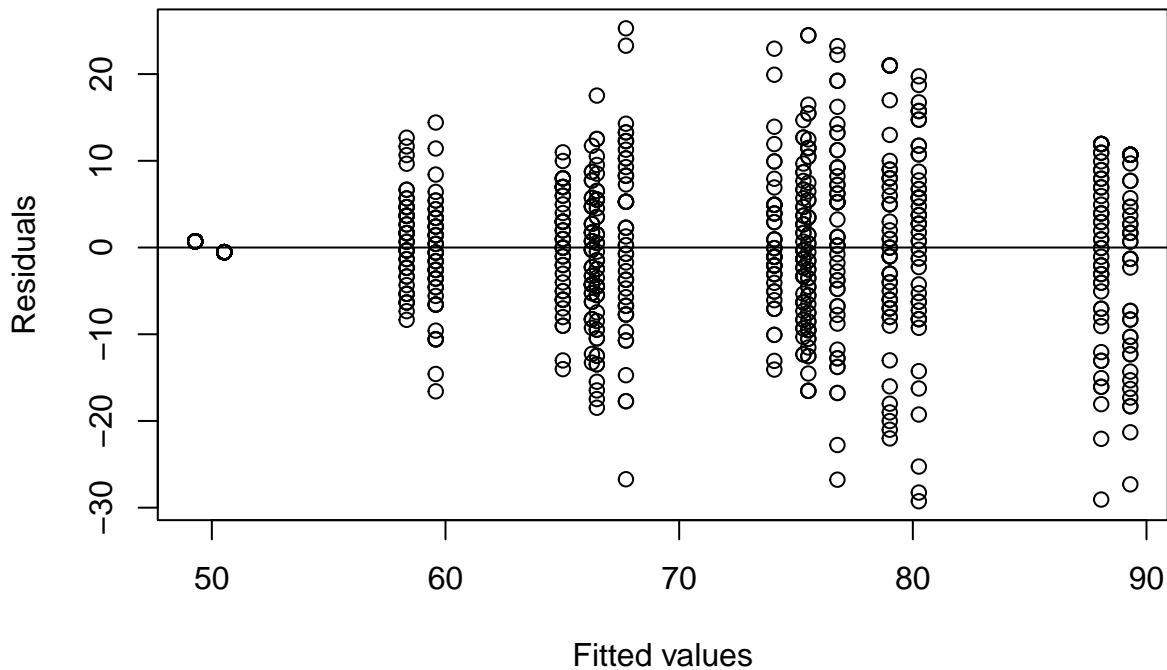
Checking the assumption of constancy of error variance for the reduced model:

```

plot(model_reduced$fitted.values,
     model_reduced$residuals,
     xlab = "Fitted values",
     ylab = "Residuals",
     main = "Figure 4: The distribution of residuals across all fitted values")
abline(h = 0)

```

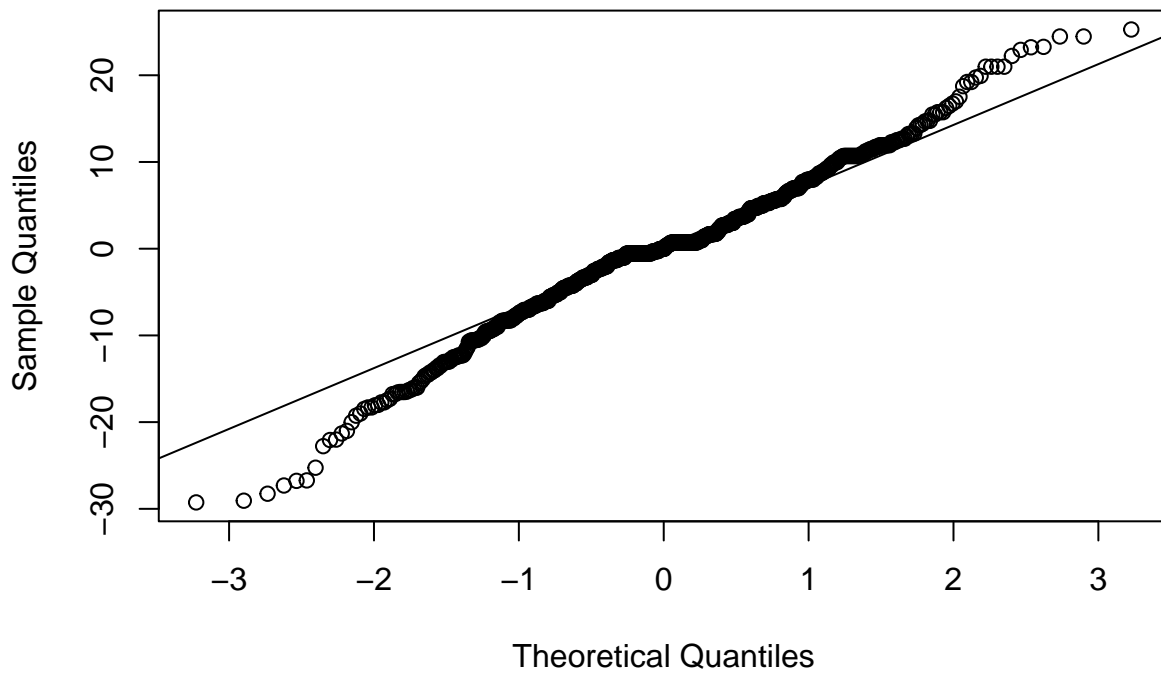
Figure 4: The distribution of residuals across all fitted values



Checking the normality of residuals:

```
qqnorm(model_reduced$residuals,  
       main = "Figure 5: The normal QQ plot of residuals")  
qqline(model_reduced$residuals)
```

Figure 5: The normal QQ plot of residuals



Lastly, we can find the main/interaction effects, their estimated standard error and the confidence interval.

```
effect = 2*round(model_r_lm$coefficients[, "Estimate"][c(2:4, 6)], 4)
sd <- 2*round(model_r_lm$coefficients[, "Std. Error"][c(2:4, 6)], 4)
var <- sd^2
df <- anova_r$Df[6]
lower <- round(effect-qt(0.975, df)*sd, 3)
upper <- round(effect+qt(0.975, df)*sd, 3)
ci <- mapply(function(a, b) paste0("(", a, ", ", b, ")"), lower, upper)
tibble(name = c("Mushrooms", "Onions", "Pickles", "Mushrooms:Onions"),
       effect = effect,
       var = var,
       ci = ci) %>%
  rename(`Effects` = name,
         `Factorial Effects` = effect,
         `Estimated Variance` = var,
         `95%Confidence Interval` = ci) %>%
  kable(caption = "The summary of the reduced model")
```

Table 5: The summary of the reduced model

Effects	Factorial Effects	Estimated Variance	95%Confidence Interval
Mushrooms	14.135	0.3518862	(12.971, 15.299)
Onions	15.590	0.3518862	(14.426, 16.754)
Pickles	9.050	0.3518862	(7.886, 10.214)
Mushrooms:Onions	-1.600	0.3518862	(-2.764, -0.436)