

Ch-06 R Codes

Ping-Yang Chen

Textbook: Montgomery, D. C. (2012). *Design and analysis of experiments*, 8th Edition. John Wiley & Sons.
Online handouts: https://github.com/PingYangChen/ANOVA_Course_R_Code

Chapter 6

2²-Design: Chemical Process Example

Read the csv file 6_ChemicalRecovery.csv in R.

```
df5 <- read.csv(file.path("data", "6_ChemicalRecovery.csv"))
```

The estimation of factor effects are:

```
# Effect of factor A (reactant concentration)
mean(df5$Recovery[df5$ReactConc == 1] - df5$Recovery[df5$ReactConc == -1])
# or 2*mean(df5$Recovery*df5$ReactConc)

# Effect of factor A (catalyst amount)
mean(df5$Recovery[df5$CataAmo == 1] - df5$Recovery[df5$CataAmo == -1])

# Effect of two-factor interaction AB
twofi <- as.numeric(df5$ReactConc)*df5$CataAmo
mean(df5$Recovery[twofi == 1] - df5$Recovery[twofi == -1])

## Effect of factor A (reactant concentration): 8.3333
## Effect of factor A (catalyst amount): -5.0000
## Effect of two-factor interaction AB: 1.6667
```

Use `aov()` to fit the ANOVA model (model description in math is omitted). Here, if the columns of the factor in the `data.frame` are not of the `factor` type. We can also specify the type of the factors as `factor` in the R model formula.

```
fit5 <- aov(Recovery ~ factor(ReactConc)*factor(CataAmo), data = df5)
summary(fit5)

##                                     Df Sum Sq Mean Sq F value    Pr(>F)
## factor(ReactConc)                 1 208.33 208.33  53.191 8.44e-05 ***
##
```

```

## factor(CataAmo)           1   75.00    75.00  19.149  0.00236 **
## factor(ReactConc):factor(CataAmo) 1    8.33     8.33   2.128  0.18278
## Residuals                  8   31.33    31.33   3.92
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

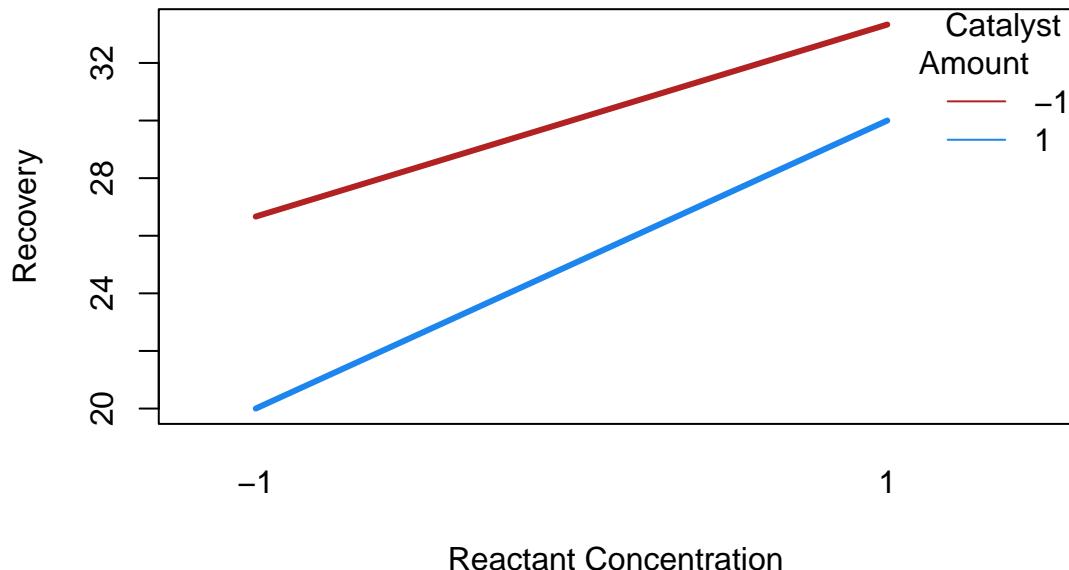
The two main effects are significant and their two-factor interaction may not exist.

To visualize the analysis result of the factorial experiment, interaction plot is commonly used tool.

```

interaction.plot(
  x.factor = df5$ReactConc, # x-axis variable
  trace.factor = df5$CataAmo, # variable for lines
  response = df5$Recovery, # y-axis variable
  ylab = "Recovery", xlab = "Reactant Concentration",
  col = c("firebrick", "dodgerblue2"),
  lty = 1, lwd = 3, trace.label = "Catalyst\\nAmount"
)

```



2^3 -Design: Plasma Etching Example

Read the csv file 6_PlasmaEtching_2^3.csv in R.

```
df6 <- read.csv(file.path("data", "6_PlasmaEtching_2^3.csv"))
```

The estimation of factor effects are:

```

# Effect of main effects
2*mean(df6$EachRate*df6$Gap)
2*mean(df6$EachRate*df6$GasFlow)
2*mean(df6$EachRate*df6$Power)

# Effect of two-factor interactions

```

```

GapGasFlow <- df6$Gap * df6$GasFlow
GapPower <- df6$Gap * df6$Power
GasFlowPower <- df6$GasFlow * df6$Power
2*mean(df6$EachRate*GapGasFlow)
2*mean(df6$EachRate*GapPower)
2*mean(df6$EachRate*GasFlowPower)

# Effect of three-factor interaction
GapGasFlowPower <- df6$Gap * df6$GasFlow * df6$Power
2*mean(df6$EachRate*GapGasFlowPower)

```

```

##   Factor Est.Effect
## 1      A   -101.625
## 2      B     7.375
## 3      C   306.125
## 4     AB   -24.875
## 5     AC  -153.625
## 6     BC    -2.125
## 7    ABC     5.625

```

Use `aov()` to fit the ANOVA model (model description in math is omitted). Here, if the columns of the factor in the `data.frame` are not of the `factor` type. We can also specify the type of the factors as `factor` in the R model formula.

```

fit6 <- aov(
  EachRate ~ factor(Gap) * factor(GasFlow) * factor(Power),
  data = df6
)
summary(fit6)

```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)						
## factor(Gap)	1	41311	41311	18.339	0.002679						
## factor(GasFlow)	1	218	218	0.097	0.763911						
## factor(Power)	1	374850	374850	166.411	1.23e-06						
## factor(Gap):factor(GasFlow)	1	2475	2475	1.099	0.325168						
## factor(Gap):factor(Power)	1	94403	94403	41.909	0.000193						
## factor(GasFlow):factor(Power)	1	18	18	0.008	0.930849						
## factor(Gap):factor(GasFlow):factor(Power)	1	127	127	0.056	0.818586						
## Residuals	8	18020	2253								
##											
## factor(Gap)				**							
## factor(GasFlow)				***							
## factor(Power)				***							
## factor(Gap):factor(GasFlow)				***							
## factor(Gap):factor(Power)				***							
## factor(GasFlow):factor(Power)				***							
## factor(Gap):factor(GasFlow):factor(Power)				***							
## Residuals				---							
## ---											
## Signif. codes:	0	'***'	0.001	'**'	0.01	'*'	0.05	'.'	0.1	','	1

The significant effects are main effects `Gap` and `Power`, and, the two-factor interaction `Gap:Power`.

We can also fit a response surface model and obtain the same conclusion.

```
ols6_full <- lm(EachRate ~ Gap * GasFlow * Power, data = df6)
summary(ols6_full)
```

```
##
## Call:
## lm(formula = EachRate ~ Gap * GasFlow * Power, data = df6)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -65.50 -11.12    0.00  11.12  65.50
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 776.062   11.865  65.406 3.32e-12 ***
## Gap         -50.812   11.865  -4.282 0.002679 **  
## GasFlow       3.688   11.865   0.311 0.763911    
## Power        153.062   11.865  12.900 1.23e-06 ***
## Gap:GasFlow  -12.438   11.865  -1.048 0.325168    
## Gap:Power    -76.812   11.865  -6.474 0.000193 *** 
## GasFlow:Power -1.062   11.865  -0.090 0.930849    
## Gap:GasFlow:Power  2.813   11.865   0.237 0.818586  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 47.46 on 8 degrees of freedom
## Multiple R-squared:  0.9661, Adjusted R-squared:  0.9364 
## F-statistic: 32.56 on 7 and 8 DF,  p-value: 2.896e-05
```

The final RSM is the reduced model that those insignificant terms are removed from the full model. Now we can use the real value of the factor levels as the predictors' values from this reduced model for better interpretation.

```
ols6_reduced <- lm(EachRate ~ NumGap * NumPower, data = df6)
summary(ols6_reduced)
```

```
##
## Call:
## lm(formula = EachRate ~ NumGap * NumPower, data = df6)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -72.50 -15.44    2.50  18.69  66.50
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -2461.188   349.513  -7.042 1.35e-05 ***
## NumGap       2242.344   342.725   6.543 2.76e-05 ***
## NumPower      10.743    1.063  10.107 3.19e-07 ***
## NumGap:NumPower -7.681    1.042  -7.370 8.62e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```

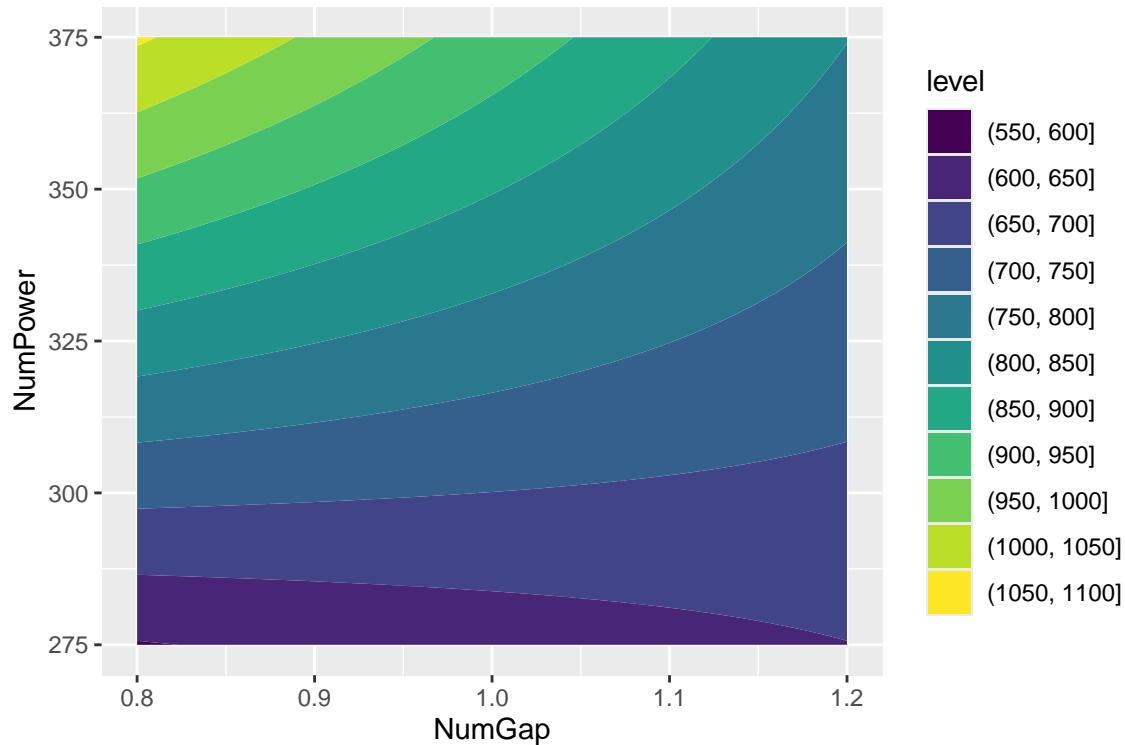
## 
## Residual standard error: 41.69 on 12 degrees of freedom
## Multiple R-squared:  0.9608, Adjusted R-squared:  0.9509
## F-statistic: 97.91 on 3 and 12 DF,  p-value: 1.054e-08

x1_grid <- seq(min(df6$NumGap), max(df6$NumGap), length = 100)
x2_grid <- seq(min(df6$NumPower), max(df6$NumPower), length = 100)
newx <- data.frame(
  NumGap = rep(x1_grid, each = 100),
  NumPower = rep(x2_grid, time = 100)
)

rsplot_data6 <- data.frame(newx, rs = predict(ols6_reduced, newx))

library(ggplot2)
#- Add color to the contour plot
ggplot(rsplot_data6) +
  geom_contour(aes(NumGap, NumPower, z = rs), colour = "white") +
  geom_contour_filled(aes(NumGap, NumPower, z = rs))

```



Unreplicated 2^k Factorial Designs: The Resin Plant Experiment

Read the csv file `6_PilotPlant.csv` in R.

```
df7 <- read.csv(file.path("data", "6_PilotPlant.csv"))
```

The estimation of factor effects are:

```

# Compute the model matrix of all effect terms without intercept
mmat7 <- model.matrix(~ Temperature*Pressure*CH2OConc*StirRate - 1, data = df7)
# Calculate the effect sizes using the +/- signs of the model matrix
eff7 <- numeric(ncol(mmat7))
for (i in 1:ncol(mmat7)) {
  eff7[i] <- 2*mean(df7$FiltrationRate*mmat7[,i])
}
names(eff7) <- colnames(mmat7)

```

	Factor	Est.Effect
## 1	Temperature	21.625
## 2	Pressure	3.125
## 3	CH2OConc	9.875
## 4	StirRate	14.625
## 5	Temperature:Pressure	0.125
## 6	Temperature:CH2OConc	-18.125
## 7	Pressure:CH2OConc	2.375
## 8	Temperature:StirRate	16.625
## 9	Pressure:StirRate	-0.375
## 10	CH2OConc:StirRate	-1.125
## 11	Temperature:Pressure:CH2OConc	1.875
## 12	Temperature:Pressure:StirRate	4.125
## 13	Temperature:CH2OConc:StirRate	-1.625
## 14	Pressure:CH2OConc:StirRate	-2.625
## 15	Temperature:Pressure:CH2OConc:StirRate	1.375

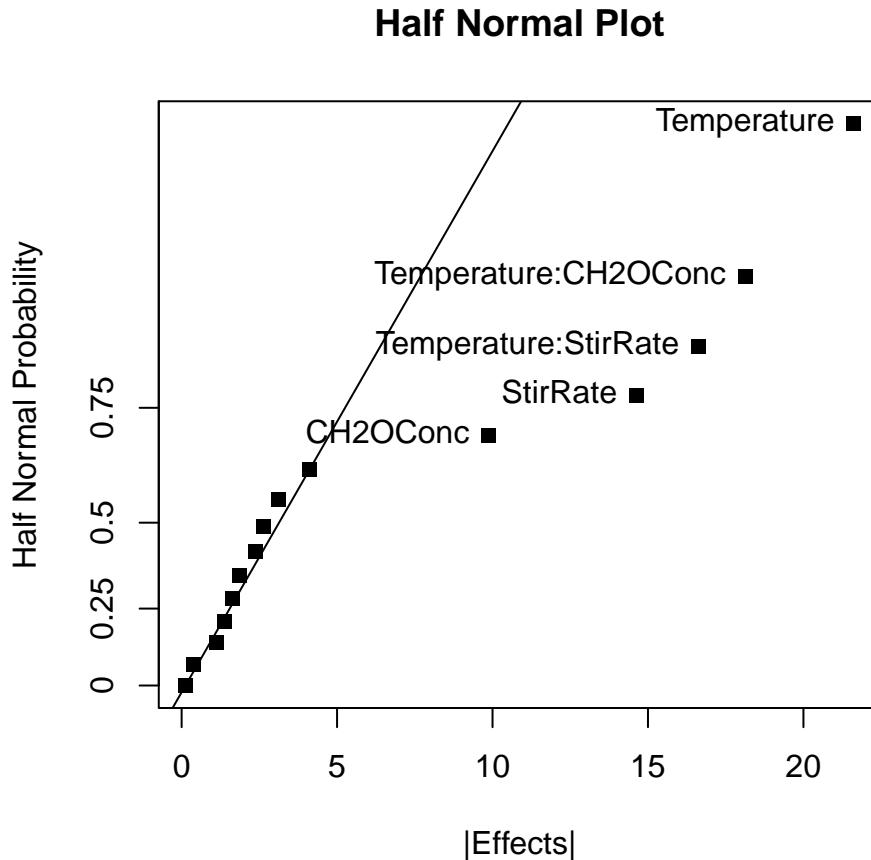
Because there is no replicates in each treatment combination, the estimate of the random error σ^2 does not exist and the ANOVA table is not available. Of all effect terms, we try of eliminate some of them before analyzing the data. According to the model assumption, the effects that are negligible should be similar to random error which is normally distributed with zero mean and constant variance. Therefore, a QQ-plot or half-Normal plot is helpful to identify effective effects. Below are the codes of half-Normal plot.

```

# Half Normal Plot
halfqqnorm <- function(input, tol = 0.5) {
  y <- sort(abs(input))
  nq <- qnorm(seq(0.5, 0.99, length = length(y)))
  plot(y, nq, yaxt = "n", pch = 15,
    xlab = "|Effects|", ylab = "Half Normal Probability")
  title("Half Normal Plot")
  # choose anchor point to draw a straight line
  s <- min(which(diff(y)/diff(range(y)) > 1/(length(y)-1)))
  abline(a = -y[1]*(nq[s]-nq[1])/(y[s]-y[1]), b = (nq[s]-nq[1])/(y[s]-y[1]))
  axis(2, at = qnorm(seq(0.5, 0.9999, length = 5)),
    labels = round(seq(0, 1, length = 5),2))
  loc <- sqrt((nq - (y - y[1])*(nq[s]-nq[1])/(y[s]-y[1]))^2) > tol
  if (is.null(names(y))) {
    text(y[loc], nq[loc], order(abs(input))[loc], pos = 2)
  } else {
    text(y[loc], nq[loc], names(abs(input))[order(abs(input))[loc]], pos = 2)
  }
}

```

```
halfqqnorm(eff7)
```



By the half Normal plot, we find out that the main effects **Temperature**, **CH2OConc** and **StirRate** and interactions **Temperature:CH2OConc**, **Temperature:StirRate** appear to be large.

Based on the observation from the half Normal plot, now the ANOVA model is

$$y_{ijkl} = \mu + \tau_i + \beta_j + \gamma_k + (\tau\beta)_{ij} + (\tau\gamma)_{ik} + \varepsilon_{ijkl} \quad (1)$$

- τ_i is the effect of the i th **Temperature** level, $i = 1, 2$.
- β_j is the effect of the j th **CH2OConc** level, $j = 1, 2$.
- γ_k is the effect of the k th **StirRate** level, $j = 1, 2$.
- $(\tau\beta)_{ij}$ is the interaction effect of the i th **Temperature** level and the j th **CH2OConc** level.
- $(\tau\gamma)_{ik}$ is the interaction effect of the i th **Temperature** level and the k th **StirRate** level.
- ε_{ijkl} is the random error, $l = 1, 2$, satisfying

$$\varepsilon_{ijkl} \stackrel{i.i.d.}{\sim} N(0, \sigma^2) \text{ where } \sigma^2 \text{ is the constant variance.}$$

Use **aov()** to fit the ANOVA model.

```
fit7 <- aov(
  FiltrationRate ~ factor(Temperature) * (factor(CH2OConc) + factor(StirRate)),
  data = df7
)
summary(fit7)
```

```

##                               Df Sum Sq Mean Sq F value    Pr(>F)
## factor(Temperature)          1 1870.6 1870.6  95.86 1.93e-06 ***
## factor(CH2OConc)            1  390.1  390.1   19.99  0.0012 **
## factor(StirRate)            1  855.6  855.6   43.85 5.92e-05 ***
## factor(Temperature):factor(CH2OConc) 1 1314.1 1314.1   67.34 9.41e-06 ***
## factor(Temperature):factor(StirRate) 1 1105.6 1105.6   56.66 2.00e-05 ***
## Residuals                   10  195.1   19.5
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Here, we left the interpretation of the response surface model for practice.

Addition of Center Points to a 2^k Designs

Recall the Resin Plant Experiment. Read the csv file `6_PilotPlant.csv` in R.

```
df7 <- read.csv(file.path("data", "6_PilotPlant.csv"))
```

```

df7_C <- data.frame(
  Temperature = rep(0, 4),
  Pressure = rep(0, 4),
  CH2OConc = rep(0, 4),
  StirRate = rep(0, 4),
  FiltrationRate = c(73, 75, 66, 69)
)

```

The calculation of $SS_{Purequadratic}$ with degree of freedom 1.

```

Yf_bar <- mean(df7$FiltrationRate)
Yc_bar <- mean(df7_C$FiltrationRate)
nf <- nrow(df7)
nc <- nrow(df7_C)
SS_pureQuad <- nf*nc*(Yf_bar - Yc_bar)^2/(nf + nc)

```

The calculation of SS_E with degree of freedom $n_c - 1$.

```
SS_E <- sum((df7_C$FiltrationRate - mean(df7_C$FiltrationRate))^2)
```

To test the significance of the Curvature, we compute the ratio of $MS_{Purequadratic}$ and MS_E as the F -statistic which follows the F distribution with degrees of freedom 1 and $n_c - 1$.

```

MS_pureQuad <- SS_pureQuad/1
MS_E <- SS_E/(nc - 1)
fval <- MS_pureQuad/MS_E
pval <- 1 - pf(fval, 1, nc - 1) # p-value

```

The part of the testing the significance of the Curvature of the ANOVA table.

```
print(data.frame(
  Source = c("Curvature", "Residual"),
  SS = c(SS_pureQuad, SS_E),
  DF = c(1, nc - 1),
  MS = c(MS_pureQuad, MS_E),
  "F" = c(sprintf("%.3f", fval), ""),
  "Pr(>F)" = c(sprintf("%.3f", pval), ""))
))

##      Source     SS   DF     MS      F Pr..F.
## 1 Curvature 1.5125  1 1.5125 0.093  0.780
## 2 Residual 48.7500  3 16.2500
```