Homework 6

I found this half-normal quantile plot written by professor TODO

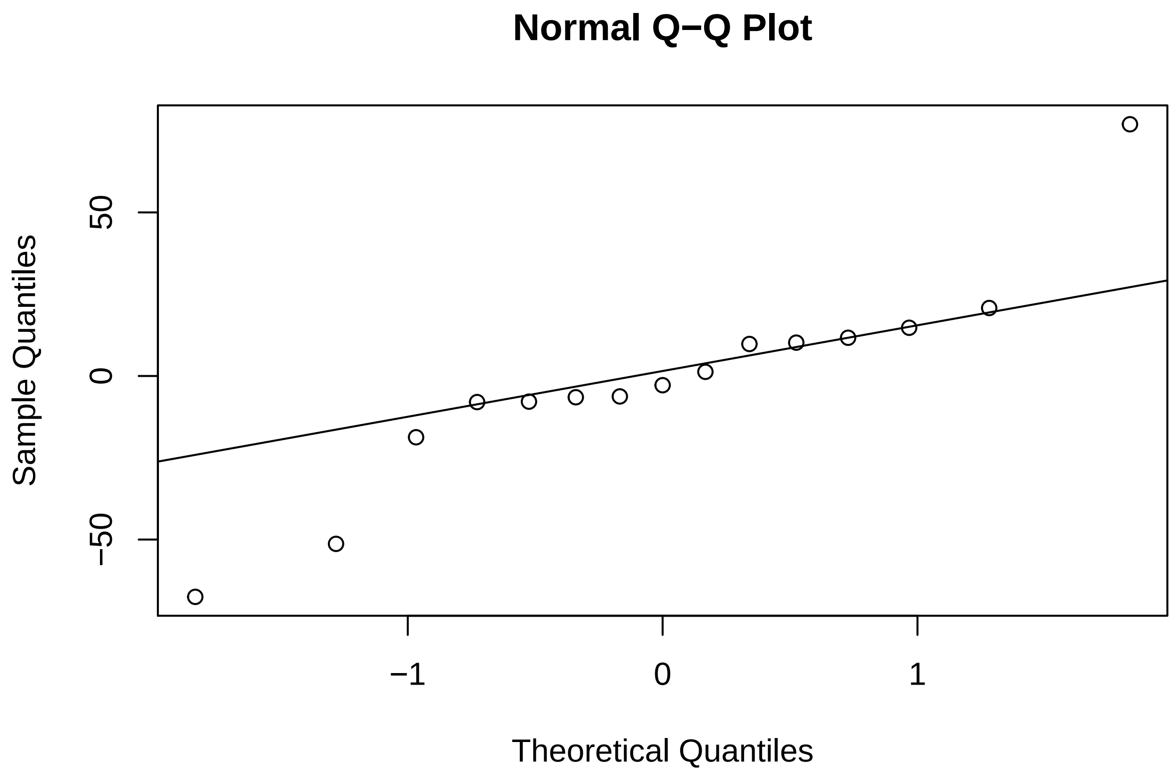
```
halfnormalplot <- function(y, label=F, n=length(y), fac.names=NULL,
                           xlim=c(-.1, 2.5), main="Half-Normal Plot",l_pos,
                           ...)
  { # label the most singificant n effects
    m <- length(y)
    x <- seq(0.5+0.25/m, 1.0-0.25/m, by=0.5/m)
    x <-  qnorm(x)
    y <-  sort(abs(y))
    qqplot(x, y, xlab="half-normal quantiles", ylab="absolute effects",
           xlim=xlim, main=main, ...)
    if(is.null(fac.names)) fac.names <-  names(y)
    else fac.names <-  rev( c(fac.names, rep("", length(y)-length(fac.names)) ) )
    if(label) for(i in (m-n+1):m) text(x[i]+.2,y[i], fac.names[i], pos=l_pos)
  }
```

6.17

```
effects <- c("A","B","C","D","AB","AC","AD","BC",
             "BD","CD","ABC","ABD","ACD","BCD","ABCD")
values <- c(76.95,-67.52,-7.84,-18.73,-51.32,
            11.69,9.78,20.78,14.74,1.27,-2.82,
            -6.5,10.2,-7.98,-6.25)
df1 <- data.frame(effects, values)
```

a

```
qqnorm(df1$values)
qqline(df1$values)
```



b

The effects A, B, and AB are significant according to the Normal QQ plot. Therefore a model could include these effects only.

6.24

```
A <- c("3rd","3rd","1st","1st","3rd","3rd","1st",
       "1st","3rd","3rd","1st","1st","3rd","3rd",
       "1st","1st")

B <- c("BW","BW","BW","BW","Color","Color","Color",
       "Color","BW","BW","BW","BW","Color","Color",
       "Color","Color")

C <- c(19.95,19.95,19.95,19.95,19.95,19.95,19.95,
       19.95,24.95,24.95,24.95,24.95,24.95,24.95,
       24.95,24.95)
C <- as.factor(C)

Number_of_Orders <- c(50,54,44,42,46,48,42,43,49,
                      46,48,45,47,48,56,54)
```

```r
df2 <- data.frame(A, B, C, Number_of_Orders)

options(contrasts=c("contr.sum","contr.poly"))
```

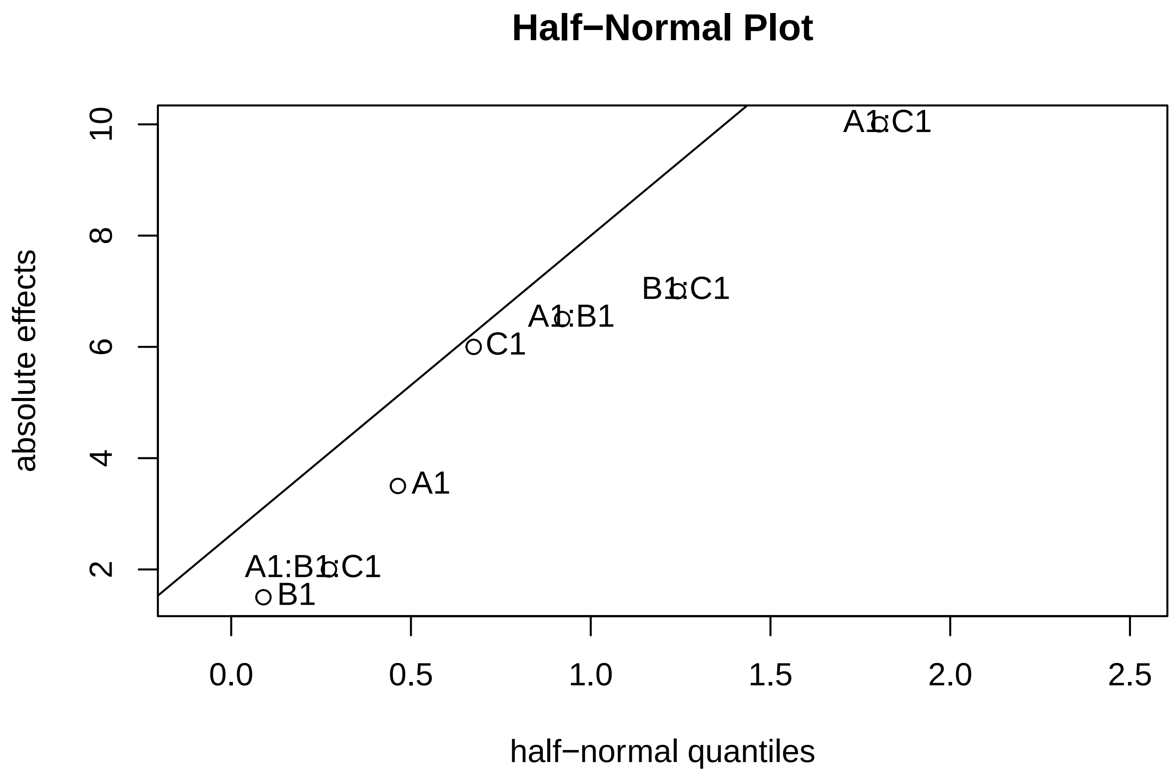
a

The factors which are significant are C, AB, AC, BC with p-values: 0.0085163, 0.0056019, 0.0004176, and 0.0037282 respectively.

```r
m <- lm(Number_of_Orders ~ A*B*C,df2)
anova(m)
```

```
## Analysis of Variance Table
##
## Response: Number_of_Orders
##           Df Sum Sq Mean Sq F value    Pr(>F)
## A          1  12.25   12.25  4.0833 0.0779708 .
## B          1   2.25    2.25  0.7500 0.4116944
## C          1  36.00   36.00 12.0000 0.0085163 **
## A:B        1  42.25   42.25 14.0833 0.0056019 **
## A:C        1 100.00  100.00 33.3333 0.0004176 ***
## B:C        1  49.00   49.00 16.3333 0.0037282 **
## A:B:C      1   4.00    4.00  1.3333 0.2815369
## Residuals  8  24.00    3.00
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

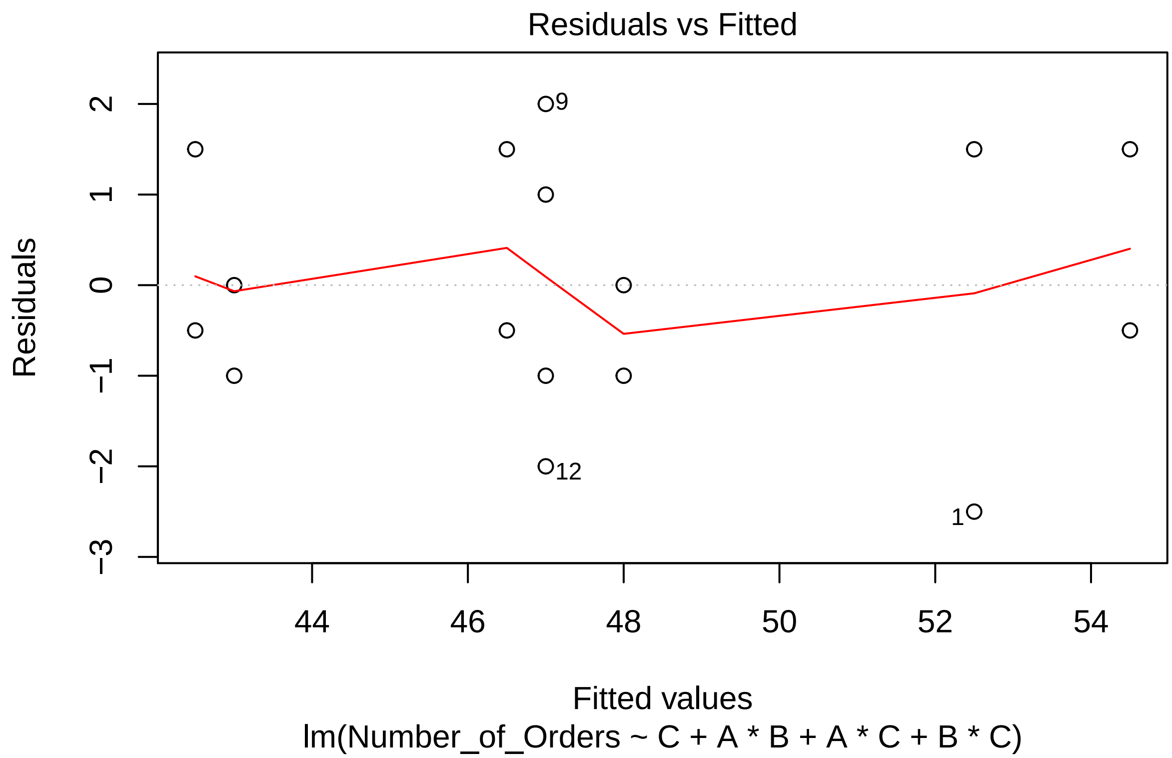
```r
#par(mfrow=c(2,2))
halfnormalplot(m$effects[2:8],label=T, l_pos=2)
qqline(m$effects[2:8])
```

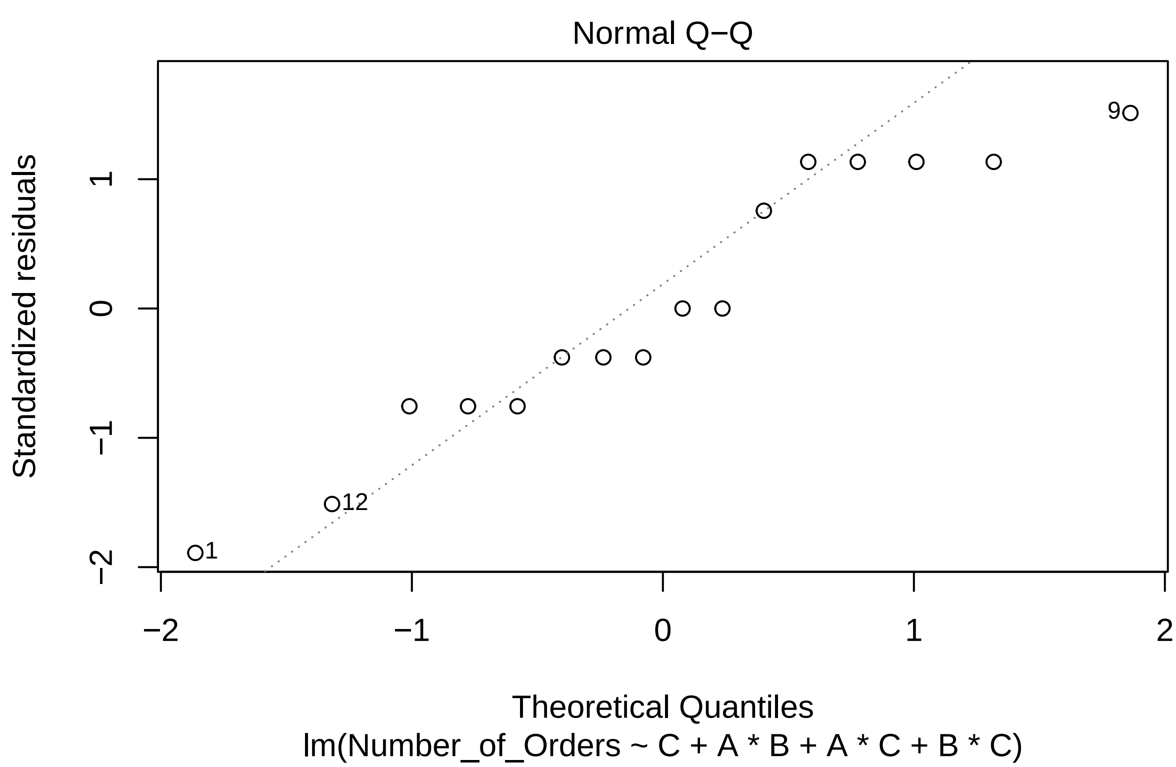


b

The Residual plot does not show any indication of non-constant variance. The normal Q-Q plot shows that the residuals are not following the normal distribution.

```
#par(mfrow=c(2,2))
m <- lm(Number_of_Orders ~ C + A*B + A*C + B*C,df2)
plot(m,1:2)
```

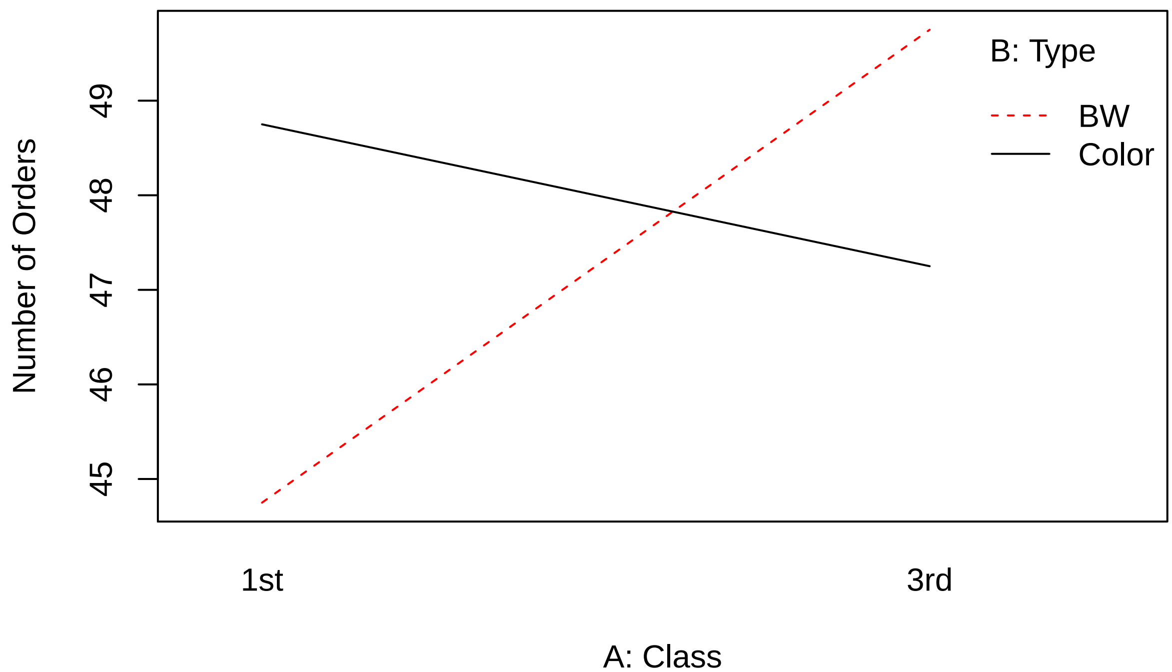

Residuals vs Fitted
Residuals
Fitted values
lm(Number_of_Orders ~ C + A * B + A * C + B * C)
9
12
1


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(Number_of_Orders ~ C + A * B + A * C + B * C)
9
12
1

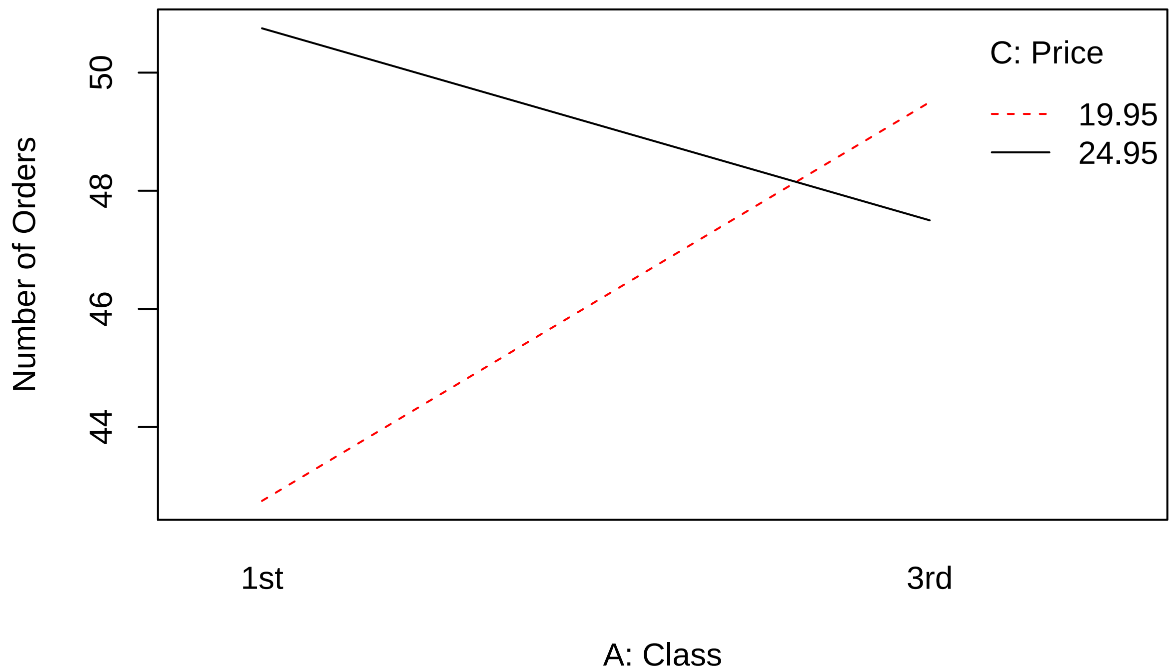
c

According to the interaction plots, I recommend 3^{rd} class mail with black and white brochures, and a price of \$19.95 this would create the highest number of orders.

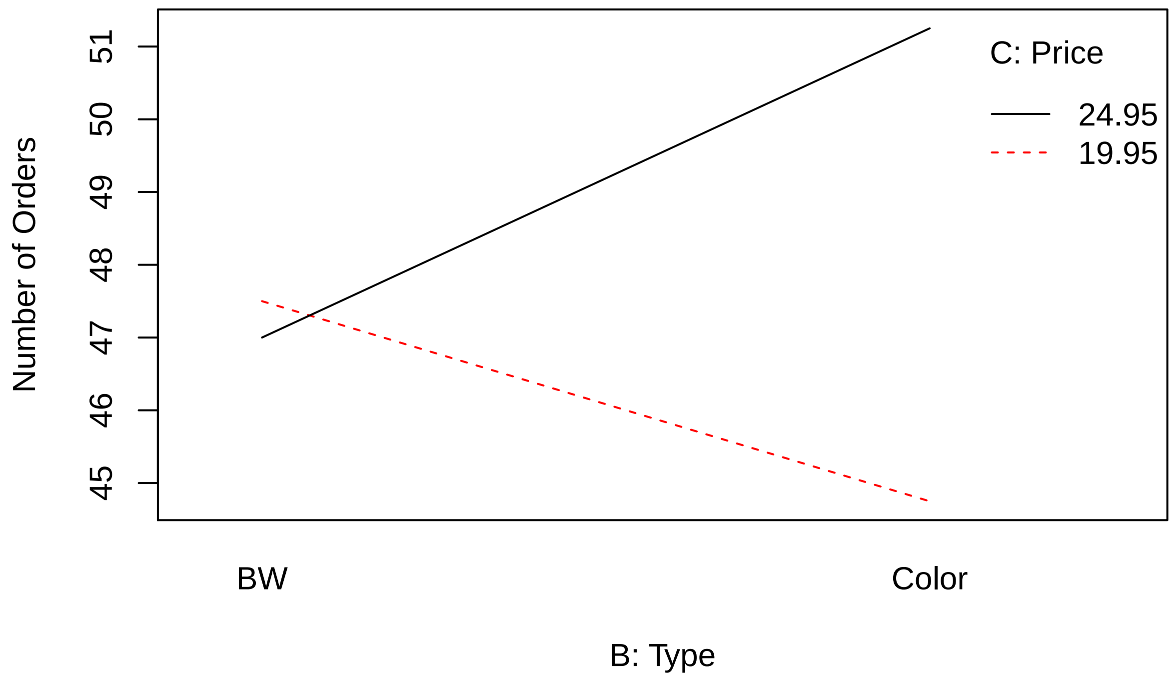
```
#par(mfrow=c(2,2))
interaction.plot(df2$A,df2$B,df2$Number_of_Orders,
                 xlab = "A: Class", ylab="Number of Orders",
                 trace.label = "B: Type",col=c("red", "black"))
```




```
interaction.plot(df2$A,df2$C,df2$Number_of_Orders,
                 xlab = "A: Class", ylab="Number of Orders",
                 trace.label = "C: Price",col=c("red", "black"))
```




```r
interaction.plot(df2$B,df2$C,df2$Number_of_Orders,
                 xlab = "B: Type", ylab="Number of Orders",
                 trace.label = "C: Price",col=c("red", "black"))
```



6.30

```
library(readxl)
df3 <-  read_excel("/Users/Earle/Downloads/Test_book_prob.xlsx")
names(df3) <- c("A", "B", "C", "Scrumptiousness")
df3$A <- as.factor(df3$A)
df3$B <- as.factor(df3$B)
df3$C <- as.factor(df3$C)
```

a)

The ANOVA indicates that the most significant factor is the pan matrial. Creating model with this we see that a glass pan plays the significant role in scrumptiousness.

```
m <- lm(Scrumptiousness ~ A*B*C, df3)
anova(m)
```

```
## Analysis of Variance Table
##
## Response: Scrumptiousness
##           Df Sum Sq Mean Sq F value   Pr(>F)
## A          1  72.25  72.250 11.9527 0.001049 **
## B          1  18.06  18.062  2.9882 0.089385 .
```

```
## C          1   0.06   0.062  0.0103 0.919370
## A:B        1   0.06   0.062  0.0103 0.919370
## A:C        1   1.56   1.562  0.2585 0.613154
## B:C        1   1.00   1.000  0.1654 0.685751
## A:B:C      1   0.25   0.250  0.0414 0.839584
## Residuals 56 338.50   6.045
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
m <- lm(Scrumptiousness ~ A, df3)
summary(m)
```

```
##
## Call:
## lm(formula = Scrumptiousness ~ A, data = df3)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.625 -1.500  0.375  1.406  6.500
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   11.562      0.301   38.41  < 2e-16 ***
## A1             1.062      0.301    3.53  0.00079 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.408 on 62 degrees of freedom
## Multiple R-squared:  0.1673, Adjusted R-squared:  0.1539
## F-statistic: 12.46 on 1 and 62 DF,  p-value: 0.0007901
```

b)

c)

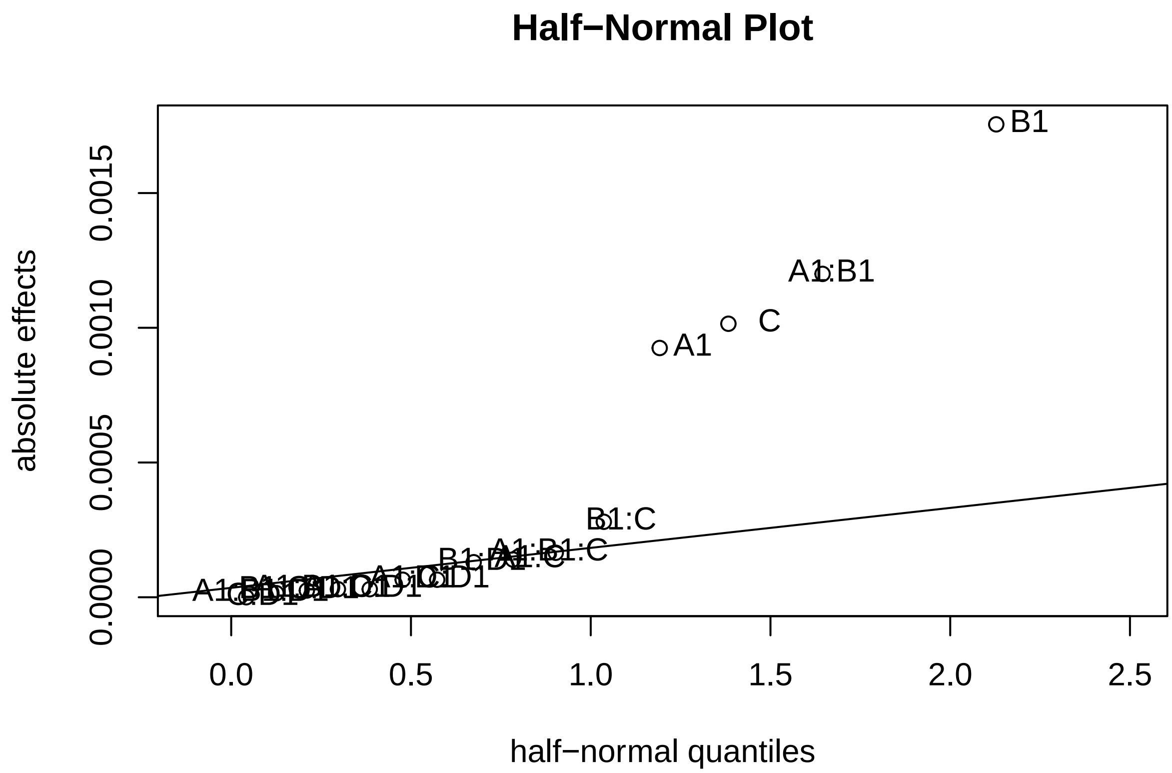
6.35

```r
df4 <- read_excel("/Users/Earle/Downloads/Chapter6_p35.xlsx")
names(df4) <- c("A","B","C","D","Surface_Roughness")
df4$A <- as.factor(df4$A)
df4$B <- as.factor(df4$B)
df4$c <- as.factor(df4$C)
df4$D <- as.factor(df4$D)
```

a

The half normal probability indicates that A, B, C, and AB significant.

```r
options(contrasts=c("contr.sum","contr.poly"))
m <- lm(Surface_Roughness ~ A*B*C*D,df4)
halfnormalplot(m$effects[2:16],label=T,l_pos=2)
qqline(m$effects[2:16])
```



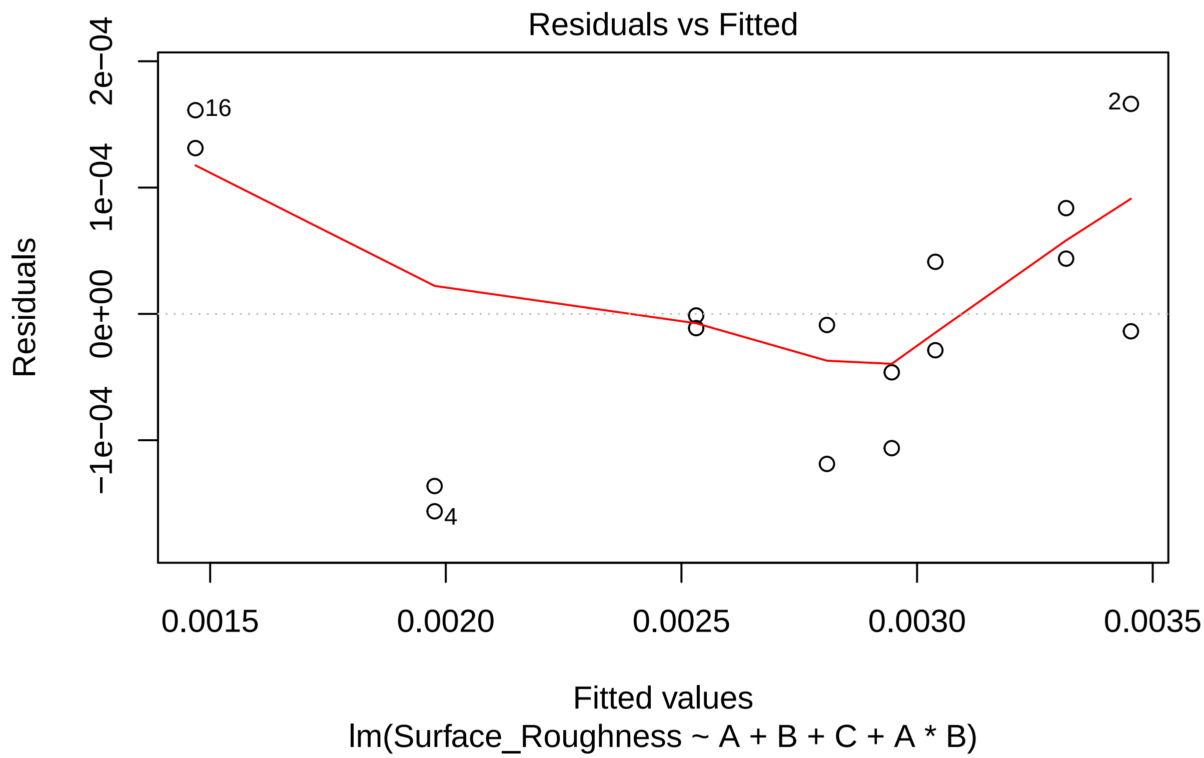
b

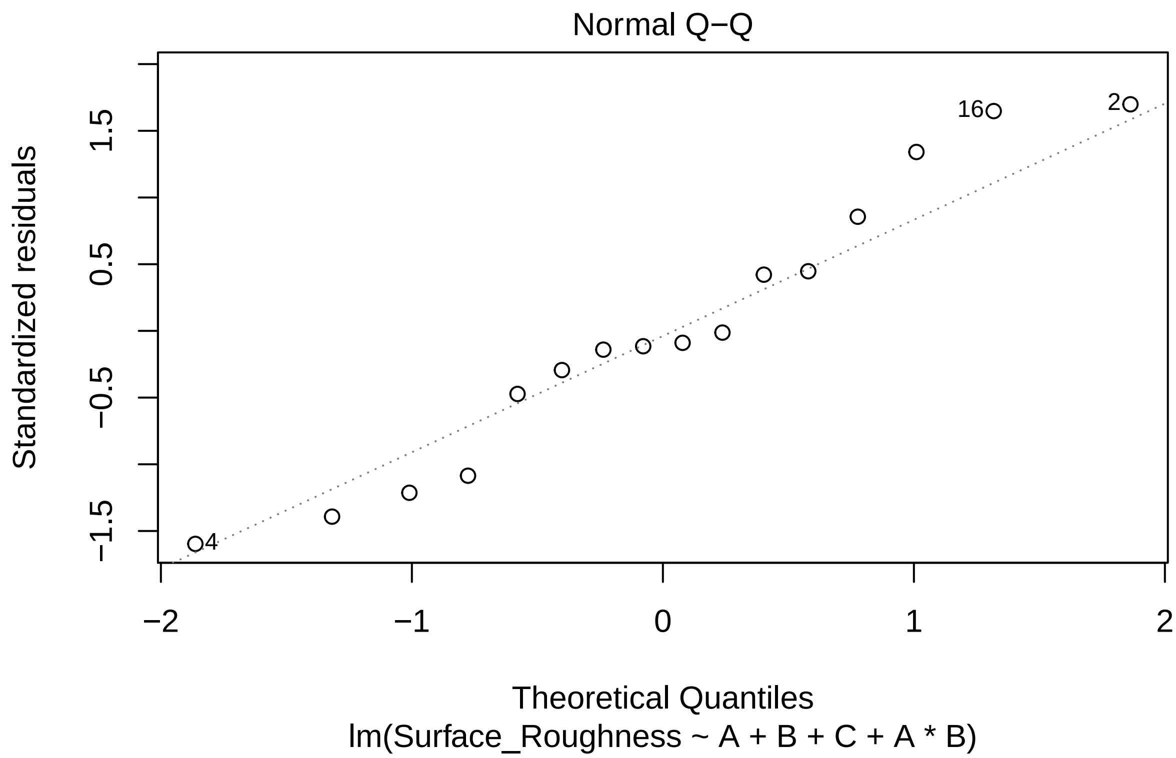
The residual plot show a trend line, but there is no indication of non-constant variance. The normal QQ plot shows that the resiuals are not following the normal distribution.

```r
m <- lm(Surface_Roughness ~ A + B + C + A*B,df4)
anova(m)
```

```
## Analysis of Variance Table
##
## Response: Surface_Roughness
##           Df     Sum Sq    Mean Sq F value    Pr(>F)
## A          1 8.5562e-07 8.5562e-07  61.425 7.936e-06 ***
## B          1 3.0800e-06 3.0800e-06 221.114 1.249e-08 ***
## C          1 1.0302e-06 1.0302e-06  73.960 3.263e-06 ***
## A:B        1 1.4400e-06 1.4400e-06 103.377 6.261e-07 ***
## Residuals 11 1.5322e-07 1.3930e-08
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
plot(m,1:2)
```

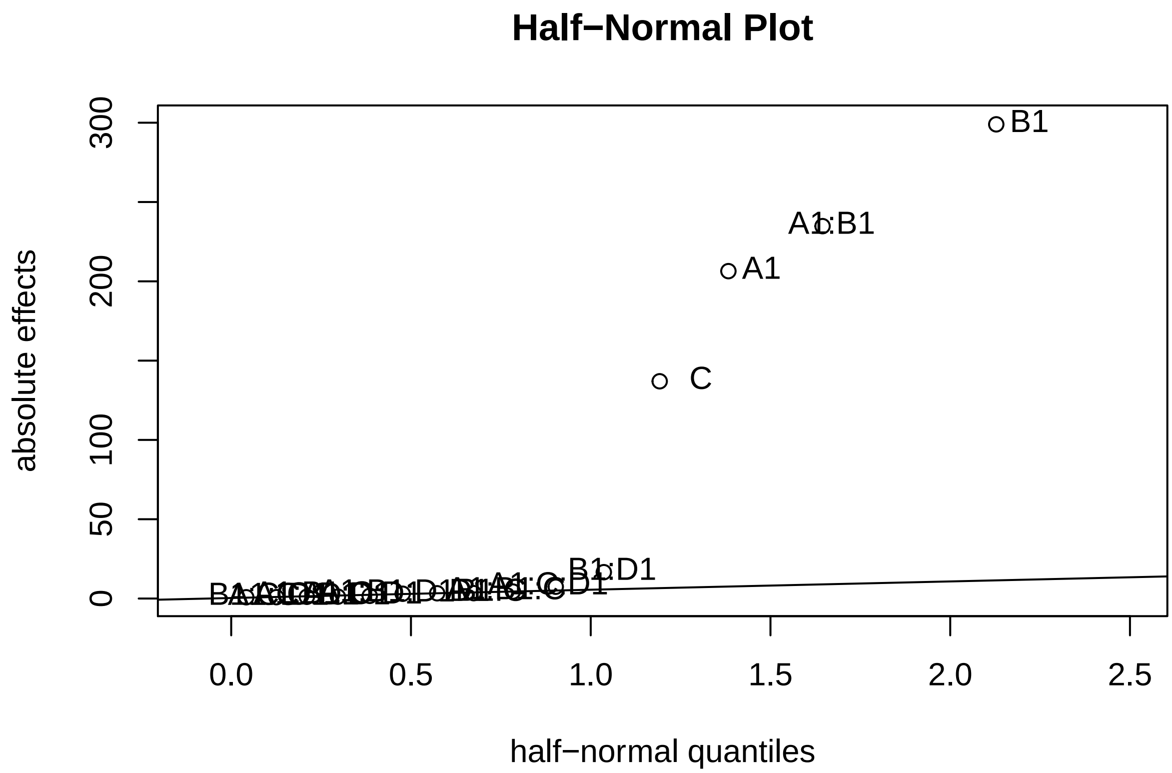

Residuals vs Fitted
Residuals
2e−04
1e−04
0e+00
−1e−04
16
2
4
0.0015
0.0020
0.0025
0.0030
0.0035
Fitted values
lm(Surface_Roughness ~ A + B + C + A * B)



c

The transformation eliminated the trend line that was present in the earlier untransformed analysis.

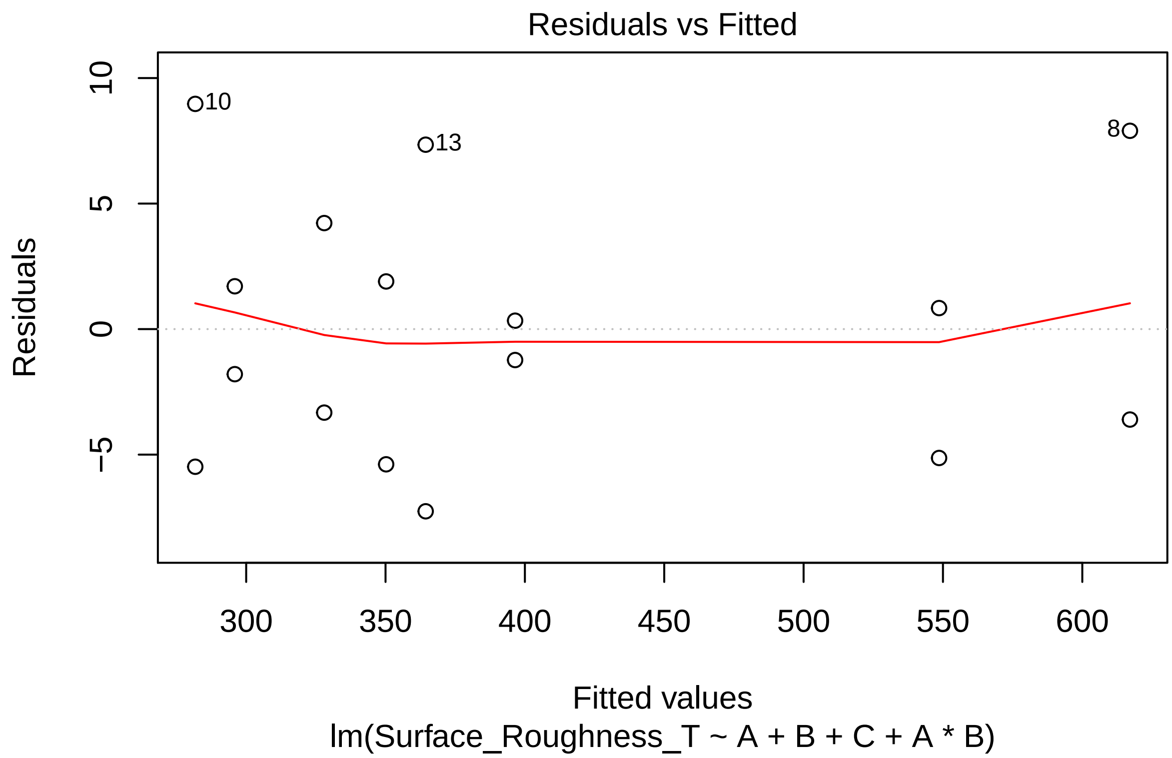
```
df4$Surface_Roughness_T <- 1/df4$Surface_Roughness
m <- lm(Surface_Roughness_T ~ A*B*C*D, df4)
halfnormalplot(m$effects[2:16],label=T,l_pos=2)
qqline(m$effects[2:16])
```

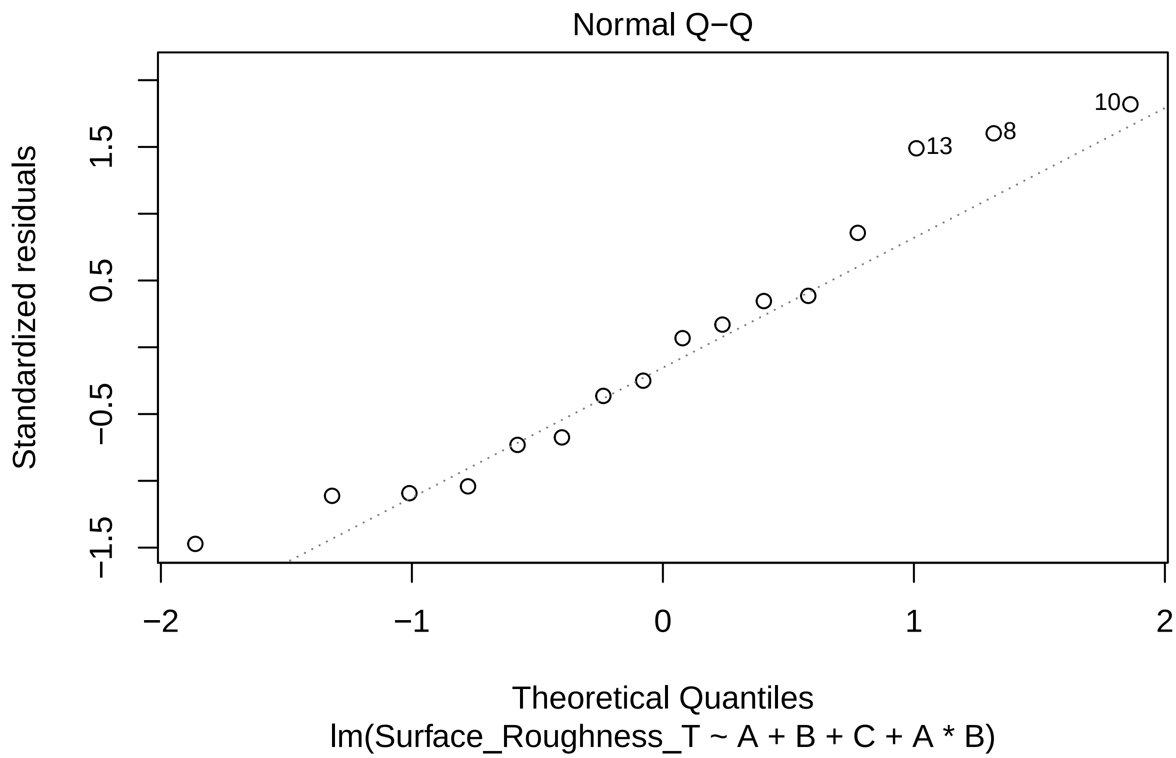



```
m <- lm(Surface_Roughness_T ~ A+B+C+A*B, df4)
anova(m)
```

```
## Analysis of Variance Table
##
## Response: Surface_Roughness_T
##           Df Sum Sq Mean Sq F value    Pr(>F)
## A          1  42611   42611 1205.11 1.359e-12 ***
## B          1  89386   89386 2527.99 2.367e-14 ***
## C          1  18762   18762  530.63 1.168e-10 ***
## A:B        1  55130   55130 1559.16 3.332e-13 ***
## Residuals 11    389      35
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
plot(m,1:2)
```


Residuals vs Fitted
Residuals
10
5
0
−5
10
13
8
300
350
400
450
500
550
600
Fitted values
lm(Surface_Roughness_T ~ A + B + C + A * B)


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(Surface_Roughness_T ~ A + B + C + A * B)
−1.5
−0.5
0.5
1.5
−2
−1
0
1
2
10
8
13

d

6.45

7.14