Response Surface Methodology, Stat 579 | Homework 04 | Name: Answer Key
Fall 2014 | | Prof. Erik B. Erhardt

Part I. (130 points) I recommend reading through all the parts of the HW (with my adjustments) before starting; this may save you some work.
MMA-RSM Chapter 4: 4.1, 4.4, 4.5, 4.7, 4.11, 4.20, 4.23.

- Whenever appropriate, use Table 4.11, p. 156, and take the main fraction when designing a $2(k-p)$.
- For 4.1, we want a one-half fraction of the 2^4 design (not 2^3).
- For 4.4, (i) show that in order to use the data from Table 4.5, p. 144, your design must at best alias some main effects, (ii) choose the design with second generator $I = ABD$ to analyze.
- For 4.23, the original design in Table 4.13 is a 2^{7-4}_{III}, not a 2^{7-3}_{III}, and the resulting design is a 2^{7-3}_{IV}, not a 2^{8-4}_{IV}.

General: Try to do all calculations in R. All R code for the assignment should be included with the part of the problem it addresses (for code and output use a fixed-width font, such as Courier). Code is used to calculate result; text is used to report and interpret results – do not report or interpret results in the code.

(30^{pts}) **1. 4.1** Suppose that in the chemical process development experiment described in Exercise 3.6, it was only possible to run a one-half fraction of the 2^3 design. Construct the design and perform the statistical analysis by selecting the relevant runs.
For 4.1, we want a one-half fraction of the 2^4 design (not 2^3).

Solution: Referring to Table 4.11 on p. 156 we will construct a 2^{4-1}_{IV} design with design generator $D = \pm ABC$. Thus we calculate where $d = abc$ in the data below and keep those 8 runs.
Read data.

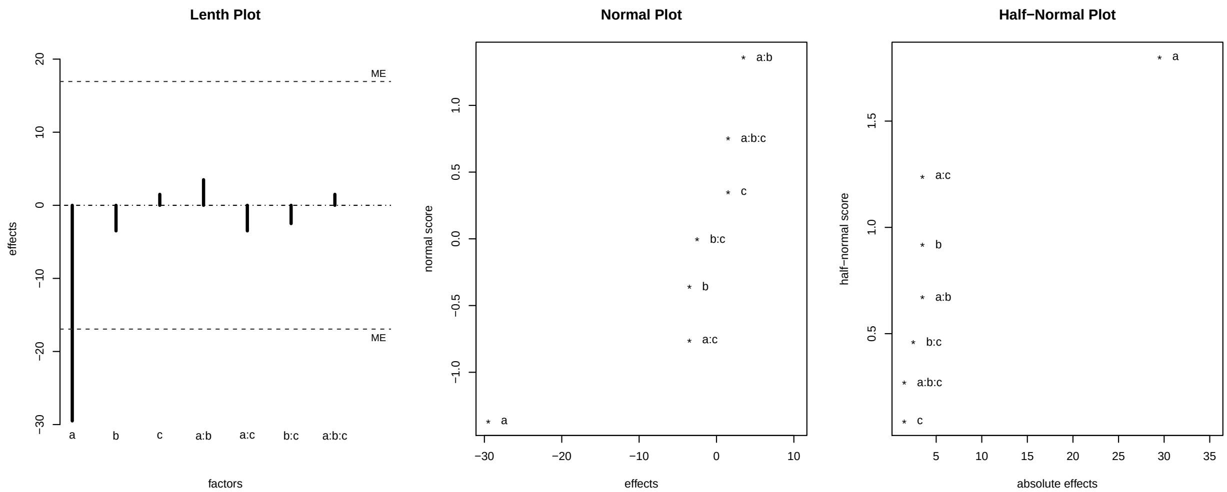
```
#### 4.1
fn.data <- "http://statacumen.com/teach/RSM/data/RSM_HW_03-06.txt"
df.4.1 <- read.table(fn.data, header=TRUE)
str(df.4.1)

## 'data.frame': 16 obs. of  5 variables:
##  $ a: int  -1 1 -1 1 -1 1 -1 1 -1 1 ...
##  $ b: int  -1 -1 1 1 -1 -1 1 1 -1 -1 ...
##  $ c: int  -1 -1 -1 -1 1 1 1 1 -1 -1 ...
##  $ d: int  -1 -1 -1 -1 -1 -1 -1 -1 1 1 ...
##  $ y: int  90 64 81 63 77 61 88 53 98 62 ...

# Find D=ABC and keep those runs
df.4.1$d_abc <- (df.4.1$d == (df.4.1$a * df.4.1$b * df.4.1$c))

df.4.1.use <- subset(df.4.1, d_abc == TRUE)
df.4.1.use

##     a  b  c  d  y d_abc
## 1  -1 -1 -1 -1 90  TRUE
## 4   1  1 -1 -1 63  TRUE
## 6   1 -1  1 -1 61  TRUE
## 7  -1  1  1 -1 88  TRUE
## 10  1 -1 -1  1 62  TRUE
## 11 -1  1 -1  1 87  TRUE
## 13 -1 -1  1  1 99  TRUE
## 16  1  1  1  1 60  TRUE
```

Fit first-order with three-way interaction linear model.

```
lm.4.1.y.3WIabc <- lm(y ~ (a + b + c)^3, data = df.4.1.use)
## externally Studentized residuals
#lm.4.1.y.3WIabc$studres <- rstudent(lm.4.1.y.3WIabc)
#summary(lm.4.1.y.3WIabc)
```

The Lenth plot below indicates that a is by far the only important effect. I project the design to a 2^1 replicated design in factor a.

```
# BsMD package has unreplicated factorial tests (Daniel plots (aka normal), and Lenth)
library(BsMD)
par(mfrow=c(1,3))
LenthPlot(lm.4.1.y.3WIabc, alpha = 0.05, main = "Lenth Plot") # , adj = 0.2

## alpha   PSE    ME   SME
##  0.05  4.50 16.94 40.54

DanielPlot(lm.4.1.y.3WIabc, main = "Normal Plot")
DanielPlot(lm.4.1.y.3WIabc, half = TRUE, main = "Half-Normal Plot")
```



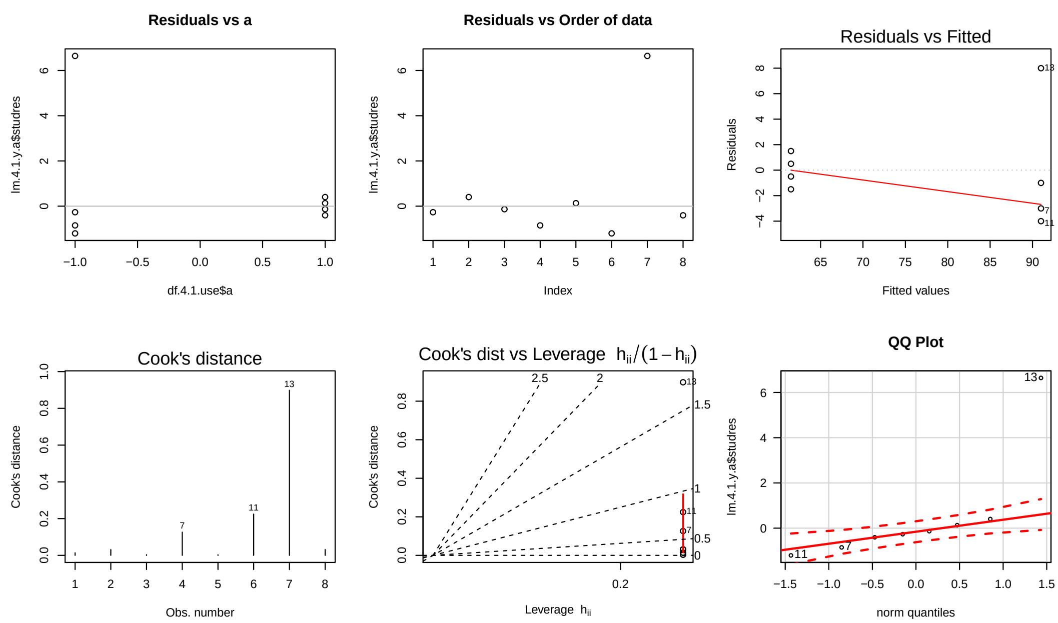
Fitting the model with just a is highly significant.
Fit first-order linear model of factor a.

```
library(rsm)
lm.4.1.y.a <- rsm(y ~ FO(a), data = df.4.1.use)
# externally Studentized residuals
lm.4.1.y.a$studres <- rstudent(lm.4.1.y.a)
summary(lm.4.1.y.a)

##
## Call:
## rsm(formula = y ~ FO(a), data = df.4.1.use)
##
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    76.25       1.41    54.2  2.6e-09 ***
## a             -14.75       1.41   -10.5  4.4e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Multiple R-squared:  0.948,Adjusted R-squared:  0.94
## F-statistic:  110 on 1 and 6 DF,  p-value: 4.42e-05
##
## Analysis of Variance Table
##
## Response: y
##             Df Sum Sq Mean Sq F value  Pr(>F)
## FO(a)        1   1740    1740     110 4.4e-05
## Residuals    6     95      16
## Lack of fit  0      0    -Inf
## Pure error   6     95      16
##
## Direction of steepest ascent (at radius 1):
##  a
## -1
##
## Corresponding increment in original units:
##  a
```

```
## -1
```

The main effect a is significant.
The plot below indicates there is one potential outlier, but it is only 1 of 8 points, so it is unclear whether it is an outlier or reflecting variability. Confirmatory runs will help clarify whether homoscedasticity is an issue.

```
# plot diagnistics
par(mfrow=c(2,3))

plot(df.4.1.use$a, lm.4.1.y.a$studres, main="Residuals vs a")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

# residuals vs order of data
plot(lm.4.1.y.a$studres, main="Residuals vs Order of data")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

plot(lm.4.1.y.a, which = c(1,4,6))

# Normality of Residuals
library(car)
qqPlot(lm.4.1.y.a$studres, las = 1, id.n = 3, main="QQ Plot")

## 13 11  7
##  8  1  2

cooks.distance(lm.4.1.y.a)

##        1        4        6        7       10       11       13       16
## 0.014035 0.031579 0.003509 0.126316 0.003509 0.224561 0.898246 0.031579
```



(30pts) **2.** **4.4** Example 4.2 describes a process improvement study in the manufacture of an integrated circuit. Suppose that only eight runs could be made in this process. Set up an appropriate 2^{5-2} design and find the alias structure. Use the data from Example 4.2 as the observations in this design, and estimate the factor effects. What conclusions can you draw?
For 4.4, (i) show that in order to use the data from Table 4.5, p. 144, your design must at best alias some main effects, (ii) choose the design with second generator $I = ABD$ to analyze.

Solution: An ideal 1/4 fraction of a 2^5 design is a 2^{5-2}_{III} design with 8 runs.
The first 1/2 fraction is given by Table 4.5, with $I = ABCDE$, which is optimal for 16 runs. However, taking another 1/2 fraction (for a 1/4 fraction) is not optimal. Whether we choose I=(2-factor) or I=(3-factor), we will alias main effects. WLOG, let the two-factor be CE (equivalent to $I = ABD$). We find that $C * I = CCE = E$. Therefore, the resolution is less than III.
Using generators $I = ABCDE$ and $I = ABD$, we obtain aliases:

```
I =ABCDE = ABD = CE
A = BCDE = BD = ACE
B = ACDE = AD = BCE
C = ABDE = ABCD = E  <-- C = E main effects aliased
D = ABCE = AB = CDE
AC = BDE = BCD = AE
BC = ADE = ACD = BE
CD = ABE = ABC = DE
```

Take those treatments under the generating with the same signs (keeping +s).
Read data.

```
#### 4.4
fn.data <- "http://statacumen.com/teach/RSM/data/RSM_HW_04-04.txt"
df.4.4 <- read.table(fn.data, header=TRUE)
df.4.4$id <- 1:dim(df.4.4)[1]
df.4.4$e <- df.4.4$a * df.4.4$b * df.4.4$c * df.4.4$d

# Find E=ABCD and D=AB and keep those runs with all +1s
df.4.4$e_abcd <- (df.4.4$e == (df.4.4$a * df.4.4$b * df.4.4$c * df.4.4$d))
df.4.4$d_ab   <- (df.4.4$d == (df.4.4$a * df.4.4$b))
## reorder by aliasing groups
#df.4.4 <- df.4.4[order(df.4.4$d_ab, df.4.4$id),]

df.4.4$use <- ((df.4.4$e_abcd == TRUE) & (df.4.4$d_ab == TRUE))
df.4.4

##     a  b  c  d     y id  e e_abcd  d_ab   use
## 1  -1 -1 -1 -1  15.1  1  1   TRUE FALSE FALSE
## 2   1 -1 -1 -1  20.6  2 -1   TRUE  TRUE  TRUE
## 3  -1  1 -1 -1  68.7  3 -1   TRUE  TRUE  TRUE
## 4   1  1 -1 -1 101.0  4  1   TRUE FALSE FALSE
## 5  -1 -1  1 -1  32.9  5 -1   TRUE FALSE FALSE
## 6   1 -1  1 -1  46.1  6  1   TRUE  TRUE  TRUE
## 7  -1  1  1 -1  87.5  7  1   TRUE  TRUE  TRUE
## 8   1  1  1 -1 119.0  8 -1   TRUE FALSE FALSE
## 9  -1 -1 -1  1  11.3  9 -1   TRUE  TRUE  TRUE
## 10  1 -1 -1  1  19.6 10  1   TRUE FALSE FALSE
## 11 -1  1 -1  1  62.1 11  1   TRUE FALSE FALSE
## 12  1  1 -1  1 103.2 12 -1   TRUE  TRUE  TRUE
## 13 -1 -1  1  1  27.1 13  1   TRUE  TRUE  TRUE
## 14  1 -1  1  1  40.3 14 -1   TRUE FALSE FALSE
## 15 -1  1  1  1  87.7 15 -1   TRUE FALSE FALSE
## 16  1  1  1  1 128.3 16  1   TRUE  TRUE  TRUE

# subset the data
df.4.4.use <- subset(df.4.4, use)
```

```
    -orig. design- -generating--              -use-
Run A B C D E=ABCD I=ABCDE I=ABD treat Yield
  1 - - - - +       +       -     e      15.1
  2 + - - - -       +       +     a      20.6     *
  3 - + - - -       +       +     b      68.7     *
  4 + + - - +       +       -     abe   101.0
  5 - - + - -       +       -     c      32.9
  6 + - + - +       +       +     ace    46.1     *
```

```
 7 - + + - +      +       +     bce     87.5    *
 8 + + + - -      +       -     abc    119.0
 9 - - - + -      +       +     d       11.3    *
10 + - - + +      +       -     ade     19.6
11 - + - + +      +       -     bde     62.1
12 + + - + -      +       +     abd    103.2    *
13 - - + + +      +       +     cde     27.1    *
14 + - + + -      +       -     acd     40.3
15 - + + + -      +       -     bcd     87.7
16 + + + + +      +       +     abcde  128.3    *
```

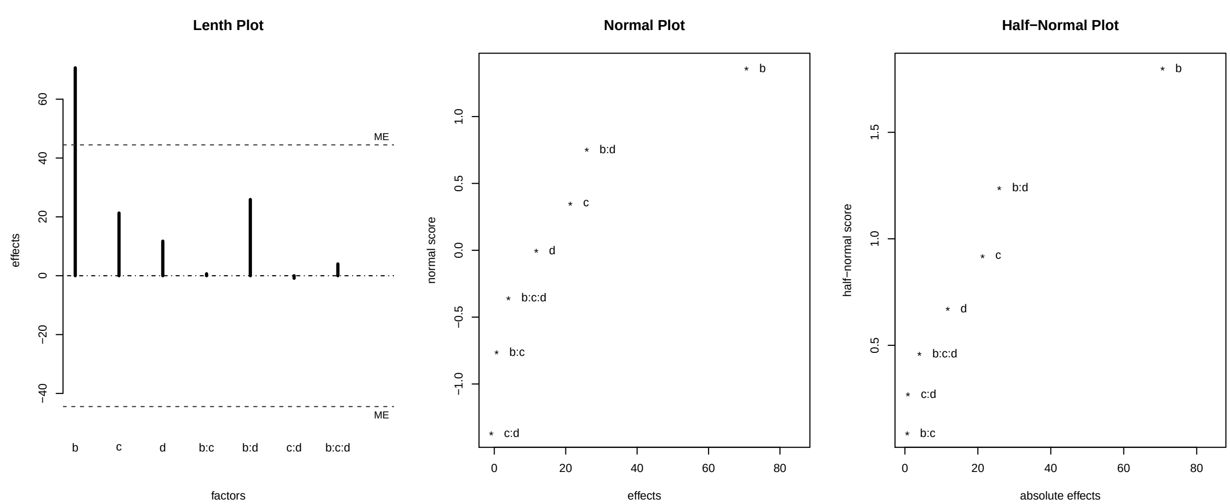
With respect to our aliasing structure, these are the runs that will be performed. Note that only 4 of the 8 alias groups are run, however all of the 8 confounded effects can be estimated.

```
runs        how many  Aliases
abd,abcde   2         I =ABCDE = ABD = CE
a,ace       2         A = BCDE = BD = ACE
b,bce       2         B = ACDE = AD = BCE
            x         C = ABDE = ABCD = E <-- no C = E runs
d,cde       2         D = ABCE = AB = CDE
            x         AC = BDE = BCD = AE
            x         BC = ADE = ACD = BE
            x         CD = ABE = ABC = DE
```

The Lenth plot below indicates that B is the only important effect (not quite) passing the SMOE bounds.

```
# Fit first-order with three-way interaction linear model in b, c, d.
lm.4.4.y.3WIbcd <- lm(y ~ (b + c + d)^3, data = df.4.4.use)
## externally Studentized residuals
#lm.4.4.y.3WIbcd£studres <- rstudent(lm.4.4.y.3WIbcd)
#summary(lm.4.4.y.3WIbcd)
```

```
# BsMD package has unreplicated factorial tests (Daniel plots (aka normal), and Lenth)
library(BsMD)
par(mfrow=c(1,3))
LenthPlot(lm.4.4.y.3WIbcd, alpha = 0.05, main = "Lenth Plot") # , adj = 0.2

##  alpha    PSE     ME    SME
##   0.05  11.81  44.46 106.41

DanielPlot(lm.4.4.y.3WIbcd, main = "Normal Plot")
DanielPlot(lm.4.4.y.3WIbcd, half = TRUE, main = "Half-Normal Plot")
```



(10pts) **3.** **4.5** Continuation of Exercise 4.4. Suppose you have made the eight runs in the 2^{5-2} design in Exercise 4.4. What additional runs would be required to identify the factor effects that are of interest? What are the alias relationships in the combined design?

Solution: Given the 8 runs from the previous experiment, we should run 8 additional runs in such a way to unalias C and E. The alias structure from 4.4 is $I = ABCDE = ABD = CE$. We have run the 1/4 fraction associated with $I = ABD$. We should now run the 1/4 fraction associated with $I = -ABD$. This will reduce the alias structure to I=ABCDE, which is the original 2^{5-1}_{V} design.

(25pts) **4.** **4.7** An article in the Journal of Quality Technology (Vol. 17, 1985, pp. 198–206) describes the use of a replicated fractional factorial to investigate the effect of five factors on the free height of leaf springs used in a automotive application. The factors are A = furnace temperature, B = heating time, C = transfer time, D = hold-down time, and E = quench oil temperature. The data are shown in Table E4.1

(a) (5 pts) Write out the alias structure for this design. What is the resolution of this design?

Solution: Inspecting the runs we find that $D = ABC$, therefore, $I = ABCD$ is the generator. In the alias structure below we see that effects with E are aliased with other effects with E. This is not optimal. Because of the aliasing, this is a 2^{5-1}_{IV} design (it could have been resolution V with $I = ABCDE$).

```
I=ABCD    AB=CD   AE=BCDE   ABE=CDE
A=BCD     AC=BD   BE=ACDE   ACE=BDE
B=ACD     BC=AD   CE=ABDE   ADE=BCE
C=ABD             DE=ABCE
D=ABC
E=ABCDE
```

(b) (5 pts) Analyze the data. What factors influence the mean free height?

Solution: Read data.

```
#### 4.7
fn.data <- "http://statacumen.com/teach/RSM/data/RSM_HW_04-07.txt"
df.4.7 <- read.table(fn.data, header=TRUE)

# reshape data into long format
library(reshape2)
df.4.7 <- melt(df.4.7, id.vars = c("a", "b", "c", "d", "e"), variable.name = "block", value.name = "y")
str(df.4.7)

## 'data.frame': 48 obs. of  7 variables:
##  $ a    : int  -1 1 -1 1 -1 1 -1 1 -1 1 ...
##  $ b    : int  -1 -1 1 1 -1 -1 1 1 -1 -1 ...
##  $ c    : int  -1 -1 -1 -1 1 1 1 1 -1 -1 ...
##  $ d    : int  -1 1 1 -1 1 -1 -1 1 -1 1 ...
##  $ e    : int  -1 -1 -1 -1 -1 -1 -1 -1 1 1 ...
##  $ block: Factor w/ 3 levels "y1","y2","y3": 1 1 1 1 1 1 1 1 1 1 ...
##  $ y    : num  7.78 8.15 7.5 7.59 7.54 7.69 7.56 7.56 7.5 7.88 ...
```

Fit first-order with four-way interaction linear model in a, b, c, e.

```
lm.4.7.y.4WIabce <- lm(y ~ (a + b + c + e)^4, data = df.4.7)
# externally Studentized residuals
lm.4.7.y.4WIabce$studres <- rstudent(lm.4.7.y.4WIabce)
summary(lm.4.7.y.4WIabce)

##
## Call:
## lm(formula = y ~ (a + b + c + e)^4, data = df.4.7)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.29333 -0.04833 -0.00667  0.07500  0.21000
##
## Coefficients:
```

```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.625625   0.020205  377.41  < 2e-16 ***
## a            0.121042   0.020205    5.99  1.1e-06 ***
## b           -0.081875   0.020205   -4.05  0.00030 ***
## c           -0.024792   0.020205   -1.23  0.22877
## e           -0.119375   0.020205   -5.91  1.4e-06 ***
## a:b         -0.014792   0.020205   -0.73  0.46945
## a:c          0.000625   0.020205    0.03  0.97552
## a:e          0.031875   0.020205    1.58  0.12450
## b:c         -0.011458   0.020205   -0.57  0.57460
## b:e          0.076458   0.020205    3.78  0.00064 ***
## c:e         -0.016458   0.020205   -0.81  0.42134
## a:b:c        0.045625   0.020205    2.26  0.03089 *
## a:b:e        0.001042   0.020205    0.05  0.95920
## a:c:e        0.009792   0.020205    0.48  0.63125
## b:c:e       -0.029792   0.020205   -1.47  0.15013
## a:b:c:e      0.019792   0.020205    0.98  0.33466
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.14 on 32 degrees of freedom
## Multiple R-squared:  0.783,Adjusted R-squared:  0.681
## F-statistic:  7.7 on 15 and 32 DF,  p-value: 7.49e-07
```

The ANOVA is significant. By the parameter estimate table, we find that A, B, E, $D = ABC$, and BE are significant.

(c) (5 pts) Calculate the range and standard deviation of the free height for each run. Is there any indication that any of these factors affects variability in the free height?

Solution: The ranges and standard deviations of each run are below.

```
library(plyr)
df.4.7.var <- ddply(df.4.7, .(a,b,c,e)
                    , function (.X) {
                        data.frame(
                          range = diff(range(.X$y))
                        , sd = sd(.X$y)
                        )
                      }
                    )
df.4.7.var
```

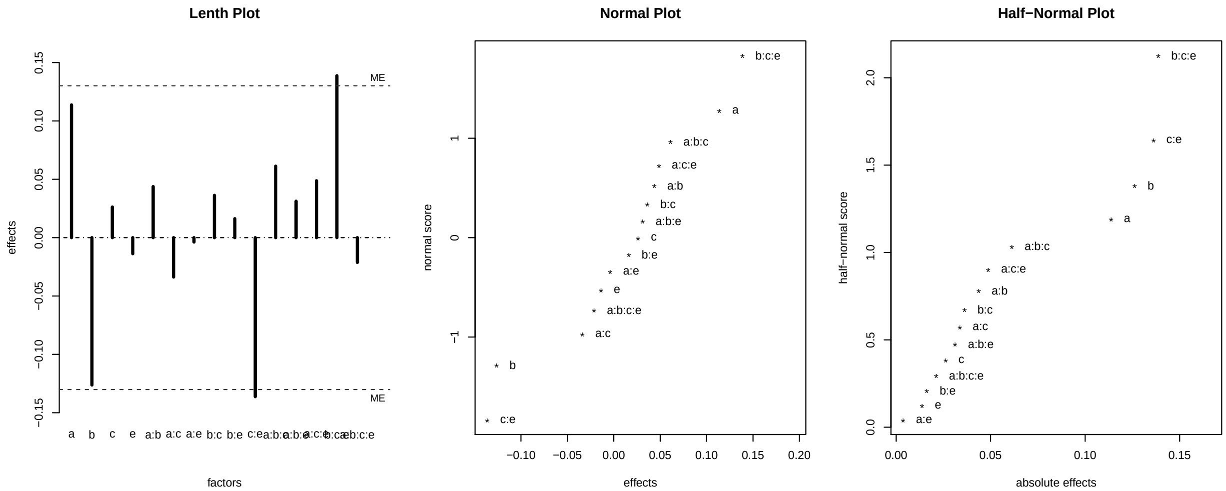
```
##     a  b  c  e range      sd
## 1  -1 -1 -1 -1  0.03 0.01732
## 2  -1 -1 -1  1  0.38 0.19313
## 3  -1 -1  1 -1  0.46 0.23861
## 4  -1 -1  1  1  0.12 0.06928
## 5  -1  1 -1 -1  0.06 0.03464
## 6  -1  1 -1  1  0.06 0.03464
## 7  -1  1  1 -1  0.12 0.06110
## 8  -1  1  1  1  0.07 0.04041
## 9   1 -1 -1 -1  0.30 0.16523
## 10  1 -1 -1  1  0.44 0.25403
## 11  1 -1  1 -1  0.40 0.22279
## 12  1 -1  1  1  0.13 0.06506
## 13  1  1 -1 -1  0.19 0.10214
## 14  1  1 -1  1  0.19 0.09609
## 15  1  1  1 -1  0.25 0.12503
## 16  1  1  1  1  0.31 0.15948
```

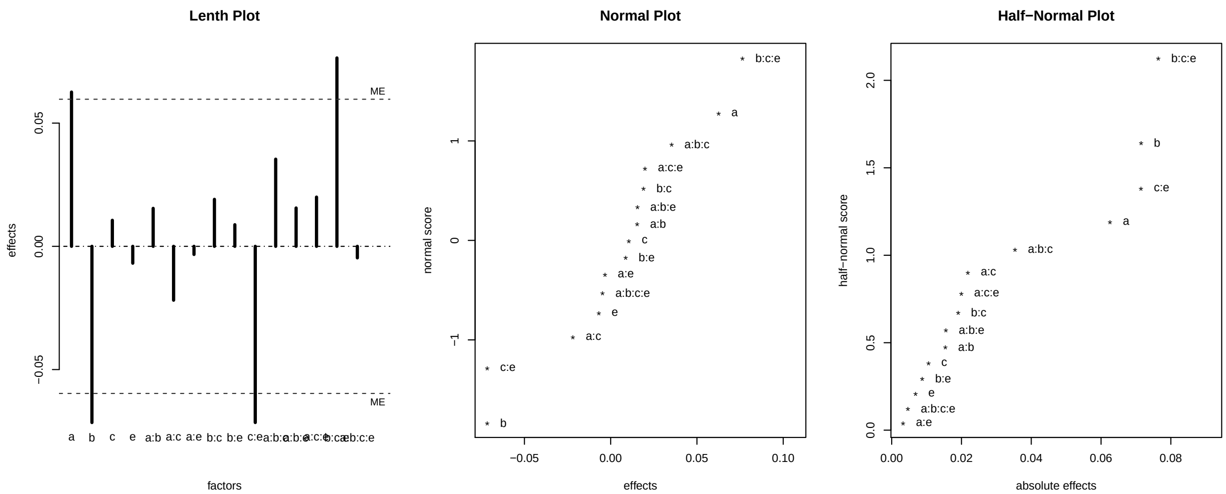
The plots below indicates there are range (top row) and sd (bottom row) patterns with respect to factors a and b, and interactions ce and bce.

When b is at its low level, the range and sd are both larger than when b is at its high level. When a is at its high level, the range and sd are both larger than when a is at its low level.

```
# Fit first-order with three-way interaction linear model in b, c, d.
lm.4.7.range.4WIabce <- lm(range ~ (a + b + c + e)^4, data = df.4.7.var)
## externally Studentized residuals
#lm.4.7.range.4WIabce$studres <- rstudent(lm.4.7.range.4WIabce)
#summary(lm.4.7.range.4WIabce)

lm.4.7.sd.4WIabce <- lm(sd ~ (a + b + c + e)^4, data = df.4.7.var)
## externally Studentized residuals
#lm.4.7.sd.4WIabce$studres <- rstudent(lm.4.7.sd.4WIabce)
#summary(lm.4.7.sd.4WIabce)
```

```
# BsMD package has unreplicated factorial tests (Daniel plots (aka normal), and Lenth)
library(BsMD)
par(mfrow=c(2,3))
LenthPlot(lm.4.7.range.4WIabce, alpha = 0.05, main = "Lenth Plot") # , adj = 0.2

##   alpha     PSE      ME     SME
## 0.05000 0.05063 0.13014 0.26419

DanielPlot(lm.4.7.range.4WIabce, main = "Normal Plot")
DanielPlot(lm.4.7.range.4WIabce, half = TRUE, main = "Half-Normal Plot")

# BsMD package has unreplicated factorial tests (Daniel plots (aka normal), and Lenth)
library(BsMD)
#par(mfrow=c(1,3))
LenthPlot(lm.4.7.sd.4WIabce, alpha = 0.05, main = "Lenth Plot") # , adj = 0.2

##   alpha     PSE      ME     SME
## 0.05000 0.02322 0.05968 0.12117

DanielPlot(lm.4.7.sd.4WIabce, main = "Normal Plot")
DanielPlot(lm.4.7.sd.4WIabce, half = TRUE, main = "Half-Normal Plot")
```



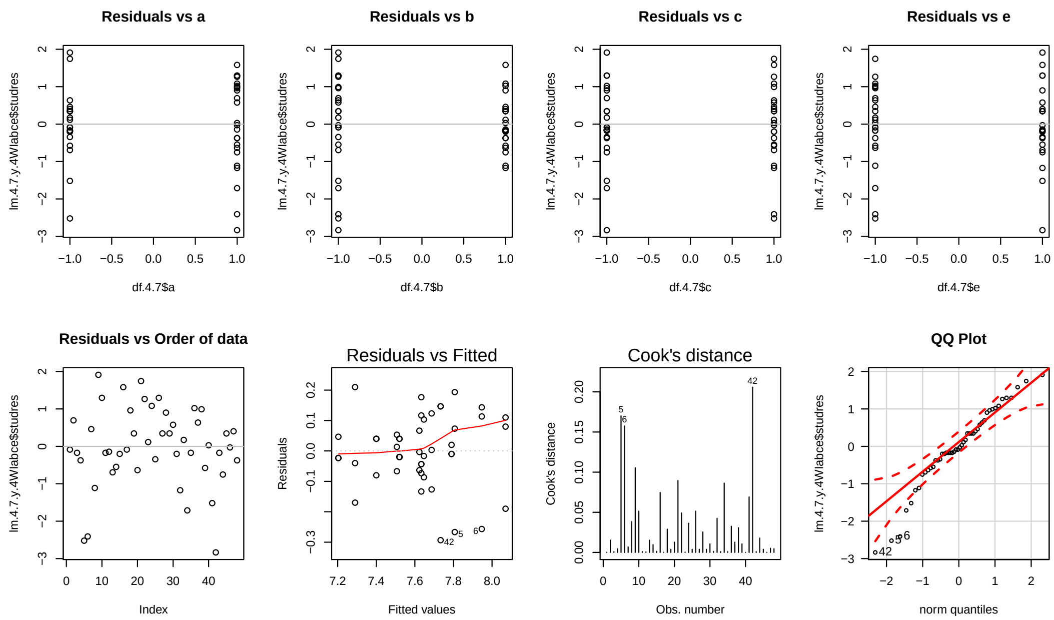


(d) (5 pts) Analyze the residuals from this experiment, and comment on your findings.

Solution: The plot below indicates there are three low outliers which are each influential. The difference in variability for a and b low and high can be seen in the residual plots.

```
# plot diagnistics
par(mfrow=c(2,4))

plot(df.4.7$a, lm.4.7.y.4WIabce$studres, main="Residuals vs a")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

plot(df.4.7$b, lm.4.7.y.4WIabce$studres, main="Residuals vs b")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

plot(df.4.7$c, lm.4.7.y.4WIabce$studres, main="Residuals vs c")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

plot(df.4.7$e, lm.4.7.y.4WIabce$studres, main="Residuals vs e")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

# residuals vs order of data
```

```
plot(lm.4.7.y.4WIabce$studres, main="Residuals vs Order of data")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

plot(lm.4.7.y.4WIabce, which = c(1,4)) # c(1,4,6)

# Normality of Residuals
library(car)
qqPlot(lm.4.7.y.4WIabce$studres, las = 1, id.n = 3, main="QQ Plot")

## 42  5  6
##  1  2  3

cooks.distance(lm.4.7.y.4WIabce)

##         1         2         3         4         5         6         7         8         9
## 2.392e-04 1.531e-02 9.568e-04 4.492e-03 1.701e-01 1.576e-01 6.804e-03 3.838e-02 1.055e-01
##        10        11        12        13        14        15        16        17        18
## 5.146e-02 9.568e-04 6.645e-04 1.531e-02 9.595e-03 1.302e-03 7.466e-02 2.392e-04 2.894e-02
##        19        20        21        22        23        24        25        26        27
## 3.827e-03 1.286e-02 8.941e-02 4.914e-02 4.253e-04 3.639e-02 3.827e-03 5.146e-02 3.827e-03
##        28        29        30        31        32        33        34        35        36
## 2.554e-02 3.827e-03 1.063e-02 1.302e-03 4.253e-02 9.568e-04 8.635e-02 9.568e-04 3.256e-02
##        37        38        39        40        41        42        43        44        45
## 1.286e-02 3.073e-02 1.063e-02 2.658e-05 6.913e-02 2.058e-01 9.568e-04 1.797e-02 3.827e-03
##        46        47        48
## 2.658e-05 5.209e-03 4.492e-03
```



(e) (5 pts) Is this the best possible design for five factors in 16 runs? Specifically, can you find a fractional design for five factors in 16 runs with a higher resolution than this one?

Solution: Using $I = ABCDE$ as a generator would increase this design from a resolution IV to V.

(15[pts]) **5. 4.11** An industrial engineer is conducting an experiment using a Monte Carlo simulation model of an inventory system. The independent variables in her model are the order quantity (A), the reorder point (B), the setup cost (C), the backorder cost (D), and the carrying cost rate (E). The response variable is average annual cost. To conserve computer time, she decides to investigate these factors using a 2^{5-2}_{III} design with $I = ABD$ and $I = BCE$. The results she obtains are $de = 95$, $ae = 134$, $b = 158$, $abd = 190$, $cd = 92$, $ac = 187$, $bce = 155$, and $abcde = 185$.

(a) (5 pts) Verify that the treatment combinations given are correct. Estimate the effects, assuming three-factor and higher interaction are negligible.

Solution: The generators $I = ABD$ and $I = BCE$ are what is recommended by Table 4.11, but with A and B swapped. Below I have generated the complete alias structure, and have indicated the three fractions that were run (though the second and third are not both run in the problem – one or the other).

```
                                          ------- fraction --------
                                            first   second    third
I-3  = ABD-1 = BCE-1  = ACDE-3   190 155              98 135
A    = BD-2  = ABCE-2 = CDE               153 191
B-1  = AD-3  = CE-3   = ABCDE-1  158 185             137  96
C-2  = ABCD  = BE     = ADE-2              99 136
D    = AB-2  = BCDE-2 = ACE               187 150
E-2  = ABDE  = BC     = ACD-2              93 139
AC-1 = BCD-3 = ABE-3  = DE-1      92  95             154 193
CD-1 = ABC-3 = BDE-3  = AE-1      92 134             189 152
```

```
#### 4.11
fn.data <- "http://statacumen.com/teach/RSM/data/RSM_HW_04-11.txt"
df.4.11 <- read.table(fn.data, header=TRUE)
df.4.11$fraction <- factor(df.4.11$fraction)
str(df.4.11)

## 'data.frame': 24 obs. of  7 variables:
##  $ a       : int  -1 1 -1 1 -1 1 -1 1 1 -1 ...
##  $ b       : int  -1 -1 1 1 -1 -1 1 1 -1 -1 ...
##  $ c       : int  -1 -1 -1 -1 1 1 1 1 -1 -1 ...
##  $ d       : int  1 -1 -1 1 1 -1 -1 1 1 -1 ...
##  $ e       : int  1 1 -1 -1 -1 -1 1 1 1 1 ...
##  $ y       : int  95 134 158 190 92 187 155 185 136 93 ...
##  $ fraction: Factor w/ 3 levels "1","2","3": 1 1 1 1 1 1 1 1 2 2 ...

df.4.11a <- subset(df.4.11, (fraction == "1"))
df.4.11b <- subset(df.4.11, ((fraction == "1") | (fraction == "2")))
df.4.11c <- subset(df.4.11, ((fraction == "1") | (fraction == "3")))
```

The treatment combinations for the first fraction are correct, as Yate's Order is preserved as defined in terms of a, b, and c with $D = AB$ and $E = BC$.

```
df.4.11a

##    a  b  c  d  e   y fraction
## 1 -1 -1 -1  1  1  95        1
## 2  1 -1 -1 -1  1 134        1
## 3 -1  1 -1 -1 -1 158        1
## 4  1  1 -1  1 -1 190        1
## 5 -1 -1  1  1 -1  92        1
## 6  1 -1  1 -1 -1 187        1
## 7 -1  1  1 -1  1 155        1
## 8  1  1  1  1  1 185        1
```

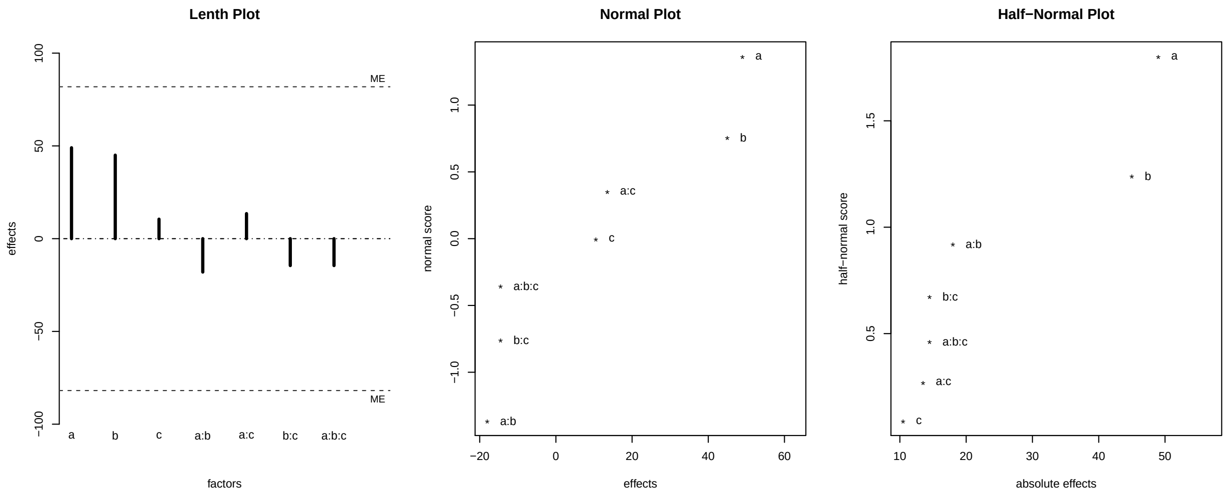
The Lenth plot below indicates no large effects. Results are so far inconclusive.
Fit first-order with three-way interaction linear model.

```
lm.4.11.y.3WIabc <- lm(y ~ (a + b + c)^3, data = df.4.11a)
## externally Studentized residuals
#lm.4.11.y.3WIabc£studres <- rstudent(lm.4.11.y.3WIabc)
#summary(lm.4.11.y.3WIabc)
```

The Lenth plot below indicates no large effects. Results are so far inconclusive.

```
# BsMD package has unreplicated factorial tests (Daniel plots (aka normal), and Lenth)
library(BsMD)
par(mfrow=c(1,3))
LenthPlot(lm.4.11.y.3WIabc, alpha = 0.05, main = "Lenth Plot") # , adj = 0.2

##  alpha    PSE     ME    SME
```

```
##    0.05  21.75  81.87 195.93
DanielPlot(lm.4.11.y.3WIabc, main = "Normal Plot")
DanielPlot(lm.4.11.y.3WIabc, half = TRUE, main = "Half-Normal Plot")
```



(b) (5 pts) Suppose that a second fraction is added to the first. The runs in this new design are $ade = 136$, $e = 93$, $ab = 187$, $bd = 153$, $acd = 139$, $c = 99$, $abce = 191$, and $bcde = 150$. How was this second fraction obtained? Add these runs to the original fraction, and estimate the effects.

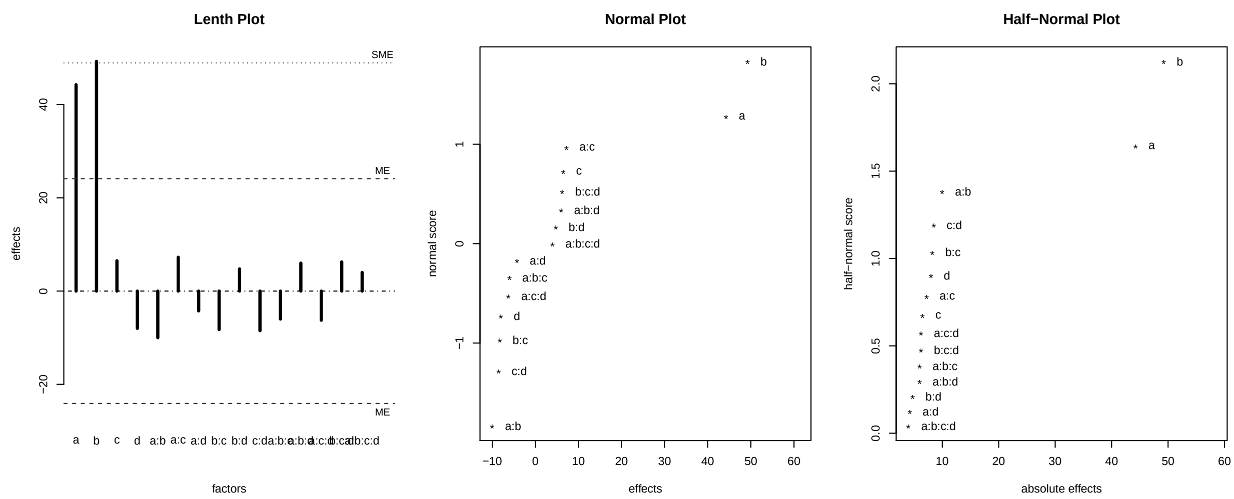
Solution: The second fraction was obtained by doing a single-factor fold-over in a of the first fraction. This allows us to estimate A and two-factor interactions involving A.
By folding over on a, because of the $D = AB$ relationship, d can be used in the Lenth procedure. Fit first-order with four-way interaction linear model.

```
df.4.11b
##     a  b  c  d  e   y fraction
## 1  -1 -1 -1  1  1  95        1
## 2   1 -1 -1 -1  1 134        1
## 3  -1  1 -1 -1 -1 158        1
## 4   1  1 -1  1 -1 190        1
## 5  -1 -1  1  1 -1  92        1
## 6   1 -1  1 -1 -1 187        1
## 7  -1  1  1 -1  1 155        1
## 8   1  1  1  1  1 185        1
## 9   1 -1 -1  1  1 136        2
## 10 -1 -1 -1 -1  1  93        2
## 11  1  1 -1 -1 -1 187        2
## 12 -1  1 -1  1 -1 153        2
## 13  1 -1  1  1 -1 139        2
## 14 -1 -1  1 -1 -1  99        2
## 15  1  1  1 -1  1 191        2
## 16 -1  1  1  1  1 150        2
lm.4.11.y.3WIabcd <- lm(y ~ (a + b + c + d)^4, data = df.4.11b)
## externally Studentized residuals
#lm.4.11.y.3WIabcd$studres <- rstudent(lm.4.11.y.3WIabcd)
#summary(lm.4.11.y.3WIabcd)
```

The Lenth plot below indicates two marginal effects, A and B.

```
# BsMD package has unreplicated factorial tests (Daniel plots (aka normal), and Lenth)
library(BsMD)
par(mfrow=c(1,3))
LenthPlot(lm.4.11.y.3WIabcd, alpha = 0.05, main = "Lenth Plot") # , adj = 0.2
##  alpha    PSE     ME    SME
```

```
## 0.050 9.375 24.099 48.925
DanielPlot(lm.4.11.y.3WIabcd, main = "Normal Plot")
DanielPlot(lm.4.11.y.3WIabcd, half = TRUE, main = "Half-Normal Plot")
```



(c) (5 pts) Suppose that the fraction $abc = 189$, $ce = 96$, $bed = 154$, $acde = 135$, $abe = 193$, $bde = 152$, $ad = 137$, and $(1) = 98$ was run. How was this fraction obtained? Add these data to the original fraction, and estimate the effects.

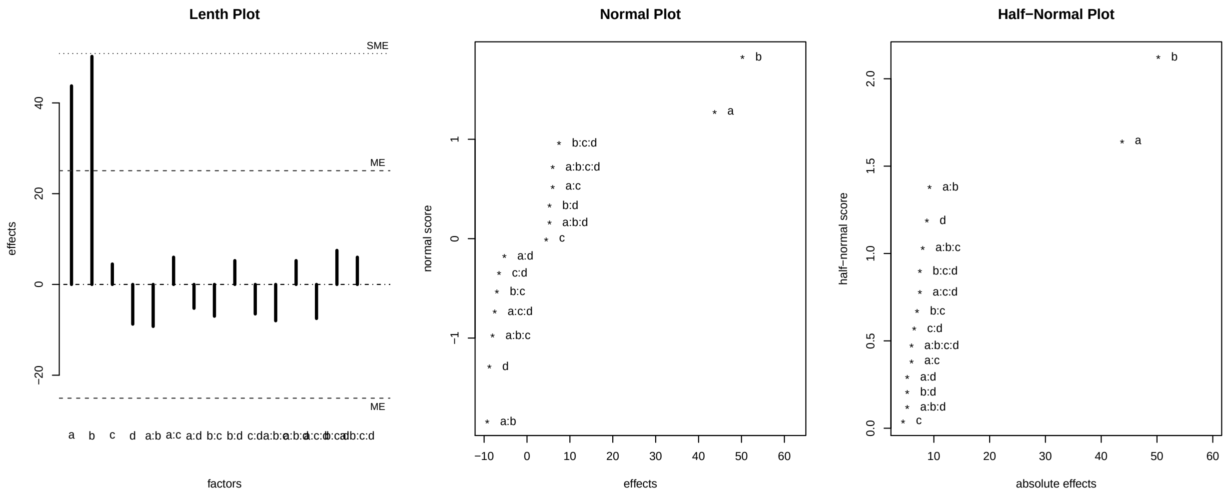
Solution: The third fraction a full fold-over of the first fraction. The single generator for the first and third fractions is $I = ABD \times BCD = ACDE$, a resolution IV design. We can now use a, b, c, d in the Lenth procedure.
Fit first-order with four-way interaction linear model.

```
df.4.11c
##     a  b  c  d  e   y fraction
## 1  -1 -1 -1  1  1  95        1
## 2   1 -1 -1 -1  1 134        1
## 3  -1  1 -1 -1 -1 158        1
## 4   1  1 -1  1 -1 190        1
## 5  -1 -1  1  1 -1  92        1
## 6   1 -1  1 -1 -1 187        1
## 7  -1  1  1 -1  1 155        1
## 8   1  1  1  1  1 185        1
## 17  1  1  1 -1 -1 189        3
## 18 -1 -1  1 -1  1  96        3
## 19 -1  1  1  1 -1 154        3
## 20  1 -1  1  1  1 135        3
## 21  1  1 -1 -1  1 193        3
## 22 -1  1 -1  1  1 152        3
## 23  1 -1 -1  1 -1 137        3
## 24 -1 -1 -1 -1 -1  98        3
lm.4.11.y.3WIabcd <- lm(y ~ (a + b + c + d)^4, data = df.4.11c)
## externally Studentized residuals
#lm.4.11.y.3WIabcd$studres <- rstudent(lm.4.11.y.3WIabcd)
#summary(lm.4.11.y.3WIabcd)
```

The Lenth plot below now indicates A and B are most important.

```
# BsMD package has unreplicated factorial tests (Daniel plots (aka normal), and Lenth)
library(BsMD)
par(mfrow=c(1,3))
LenthPlot(lm.4.11.y.3WIabcd, alpha = 0.05, main = "Lenth Plot") # , adj = 0.2
## alpha   PSE    ME   SME
```

```
##  0.05  9.75 25.06 50.88
DanielPlot(lm.4.11.y.3WIabcd, main = "Normal Plot")
DanielPlot(lm.4.11.y.3WIabcd, half = TRUE, main = "Half-Normal Plot")
```



(10pts) **6. 4.20** Project the 2^{4-1}_{IV} design in Example 4.1 into two replicates of a 2^2 design in the factors A and B. Analyze the data and draw conclusions.

Solution: Read data.

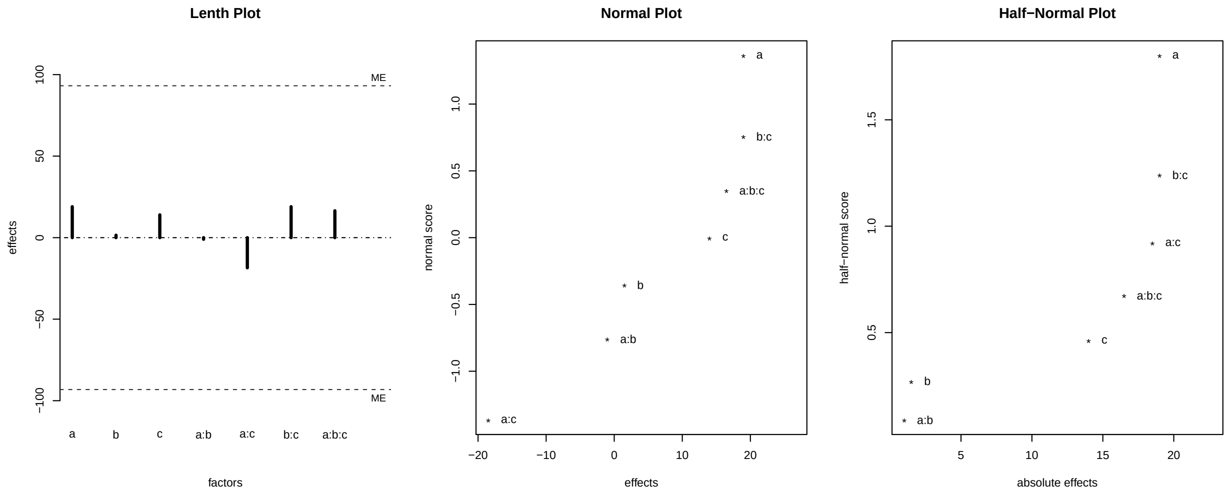
```
#### 4.20
fn.data <- "http://statacumen.com/teach/RSM/data/RSM_HW_04-20.txt"
df.4.20 <- read.table(fn.data, header=TRUE)
str(df.4.20)

## 'data.frame': 8 obs. of  4 variables:
##  $ a: int  -1 1 -1 1 -1 1 -1 1
##  $ b: int  -1 -1 1 1 -1 -1 1 1
##  $ c: int  -1 -1 -1 -1 1 1 1 1
##  $ y: int  45 100 45 65 75 60 80 96
```

Fit first-order with three-way interaction linear model.

```
lm.4.20.y.3WIabc <- lm(y ~ (a + b + c)^3, data = df.4.20)
## externally Studentized residuals
#lm.4.20.y.3WIabc$studres <- rstudent(lm.4.20.y.3WIabc)
#summary(lm.4.20.y.3WIabc)
```

First note that the full 2^{4-1}_{IV} design, by the Lenth plot below, has no important effects.

```
# BsMD package has unreplicated factorial tests (Daniel plots (aka normal), and Lenth)
library(BsMD)
par(mfrow=c(1,3))
LenthPlot(lm.4.20.y.3WIabc, alpha = 0.05, main = "Lenth Plot") # , adj = 0.2

##  alpha    PSE     ME    SME
##   0.05  24.75  93.16 222.96

DanielPlot(lm.4.20.y.3WIabc, main = "Normal Plot")
DanielPlot(lm.4.20.y.3WIabc, half = TRUE, main = "Half-Normal Plot")
```

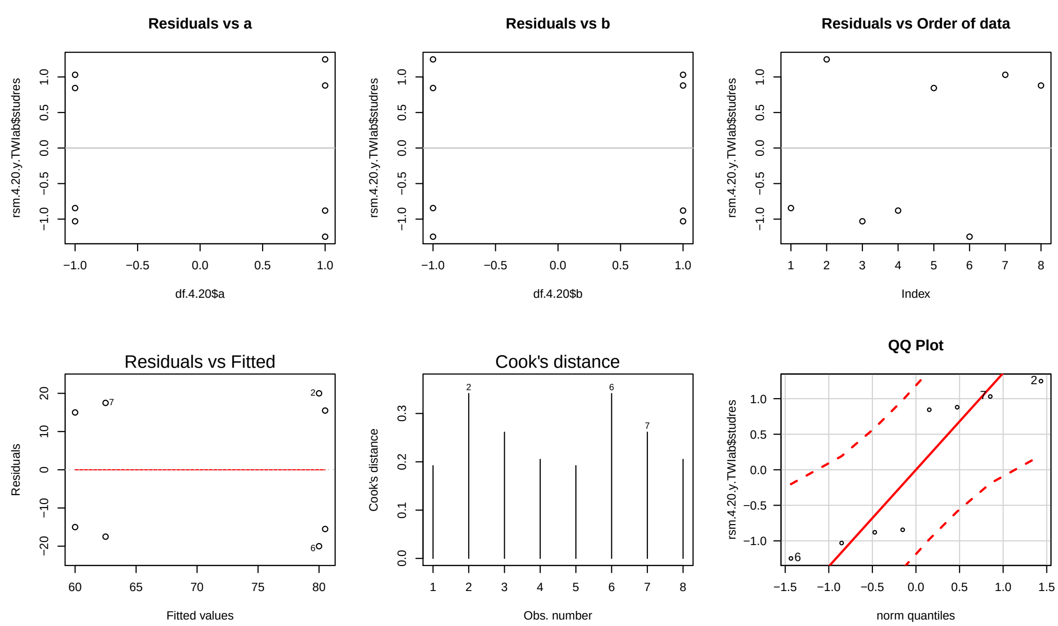


Projected into 2 replicates of a 2^2 in A and B, we have no significant factors (the ANOVA is not significant).
Fit first-order with two-way interaction linear model of factors a and b.

```
library(rsm)
rsm.4.20.y.TWIab <- rsm(y ~ FO(a, b) + TWI(a, b), data = df.4.20)
# externally Studentized residuals
rsm.4.20.y.TWIab$studres <- rstudent(rsm.4.20.y.TWIab)
summary(rsm.4.20.y.TWIab)

##
## Call:
## rsm(formula = y ~ FO(a, b) + TWI(a, b), data = df.4.20)
##
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    70.75       8.56    8.27   0.0012 **
## a               9.50       8.56    1.11   0.3291
## b               0.75       8.56    0.09   0.9344
## a:b            -0.50       8.56   -0.06   0.9562
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Multiple R-squared:  0.237,Adjusted R-squared:  -0.335
## F-statistic: 0.415 on 3 and 4 DF,  p-value: 0.752
##
## Analysis of Variance Table
##
## Response: y
##             Df Sum Sq Mean Sq F value Pr(>F)
## FO(a, b)     2    727     363    0.62   0.58
## TWI(a, b)    1      2       2    0.00   0.96
## Residuals    4   2343     586
## Lack of fit  0      0
## Pure error   4   2343     586
##
## Stationary point of response surface:
##    a    b
##  1.5 19.0
##
## Eigenanalysis:
## $values
## [1]  0.25 -0.25
##
## $vectors
##      [,1]    [,2]
## a -0.7071 -0.7071
## b  0.7071 -0.7071
```

Projected into 2 replicates of a 2^2 in A and B, we have no significant factors (the ANOVA is not significant).
The residual plots below suggests that there may be a pair of factors that can explain the interesting pattern in the residuals. But checking all pairs (a and c, and b and c), no important factors are revealed. More experiments will be necessary.

```
# plot diagnostics
par(mfrow=c(2,3))

plot(df.4.20$a, rsm.4.20.y.TWIab$studres, main="Residuals vs a")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

plot(df.4.20$b, rsm.4.20.y.TWIab$studres, main="Residuals vs b")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

# residuals vs order of data
plot(rsm.4.20.y.TWIab$studres, main="Residuals vs Order of data")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

plot(rsm.4.20.y.TWIab, which = c(1,4))

# Normality of Residuals
library(car)
qqPlot(rsm.4.20.y.TWIab$studres, las = 1, id.n = 3, main="QQ Plot")

## 6 2 7
## 1 8 7

cooks.distance(rsm.4.20.y.TWIab)

##      1      2      3      4      5      6      7      8
## 0.1921 0.3414 0.2614 0.2051 0.1921 0.3414 0.2614 0.2051
```



(10pts) **7. 4.23** Fold over the 2^{7-3}_{III} design in Table 4.13. Verify that the resulting design is a 2^{8-4}_{IV} design. Is this a minimal design?
For 4.23, the original design in Table 4.13 is a 2^{7-4}_{III}, not a 2^{7-3}_{III}, and the resulting design is a 2^{7-3}_{IV}, not a 2^{8-4}_{IV}.
Solution: In the first fraction the generators are:

```
I=ABD I=ACE I=BCF I=ABCG
```

In the second fraction the generators are:

```
I=-ABD  I=-ACE  I=-BCF  I=ABCG
```

To contruct the design (p. 169) the like sign word I=ABCG will be one generator. The other generators are created by combining generators where the signs changed.

```
I=(ABD)(ACE)=BCDE
I=(ABD)(BCF)=ACDF
I=(ACE)(BCF)=ABEF
```

Therefore, the complete defining relation for the combined design is:

```
I=ABCG=BCDE=ACDF=ABEF=ADEG=BDFG=CEFG
```

Because the defining relation for the combined design contains only four-letter words (four-term interactions), the combinded design is of resolution IV.
The resulting 16 runs are given here:

	basic			--------aliases--------				
Run	A	B	C	D=AB	E=AC	F=BC	G=ABC	treat
1	-	-	-	+	+	+	-	def
2	+	-	-	-	-	+	+	afg
3	-	+	-	-	+	-	+	beg
4	+	+	-	+	-	-	-	abd
5	-	-	+	+	-	-	+	cdg
6	+	-	+	-	+	-	-	ace
7	-	+	+	-	-	+	-	cdg
8	+	+	+	+	+	+	+	abcdefg

	fold-over design							
Run	A	B	C	D=-AB	E=-AC	F=-BC	G=ABC	treat
9	+	+	+	-	-	-	+	abcg
10	-	+	+	+	+	-	-	bcde
11	+	-	+	+	-	+	-	acdf
12	-	-	+	-	+	+	+	cefg
13	+	+	-	-	+	+	-	abef
14	-	+	-	+	-	+	+	bdfg
15	+	-	-	+	+	-	+	adeg
16	-	-	-	-	-	-	-	(1)

A resolution IV design that contains exactly $2k$ runs is a *minimal design*. In our case our $k-p$ design is a $7-3$. Since it contains 16 runs, not $2 \times 7 = 14$ runs, it is not minimal.