101B - HW5

```
library(phia)
```

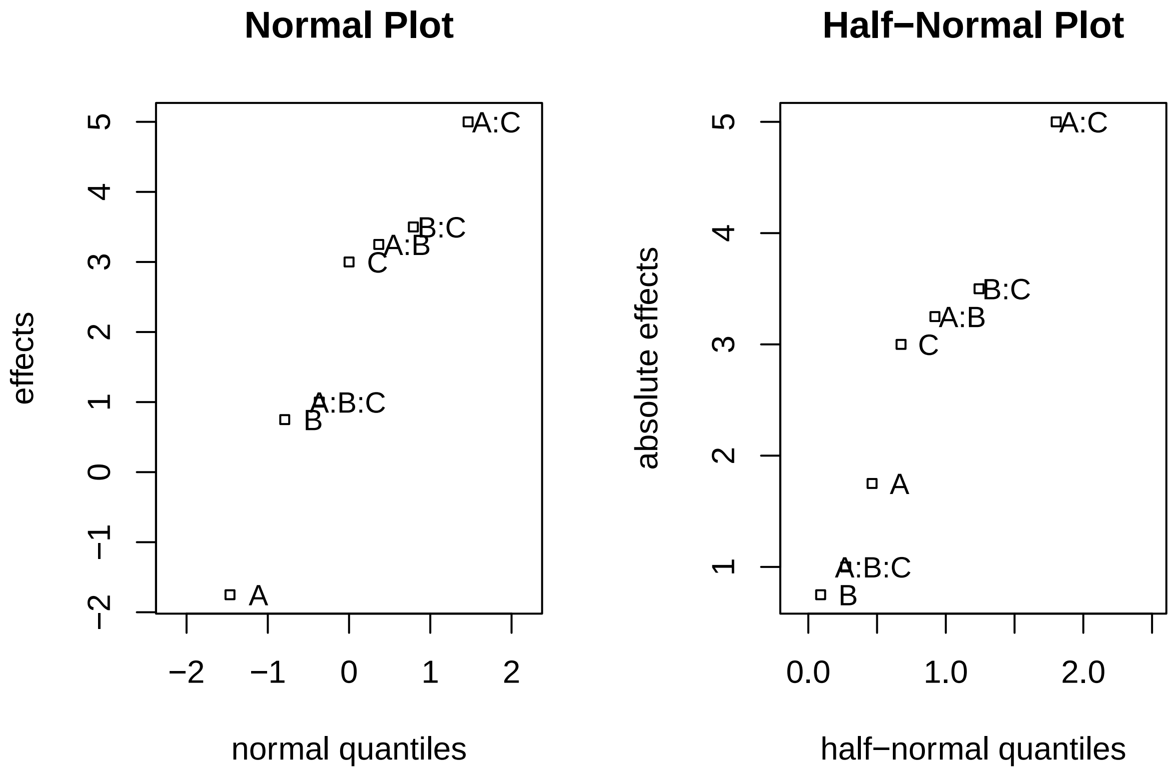
```
## Loading required package: car

## Warning: package 'car' was built under R version 4.1.2

## Loading required package: carData

## Warning: package 'carData' was built under R version 4.1.2
```

```
A <- c(-1,1,-1,1,-1,1,-1,1)
B <- c(-1,-1,1,1,-1,-1,1,1)
C <- c(-1,-1,-1,-1,1,1,1,1)

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


data <- data.frame("A" = A, "B" = B, "C" = C, "Y" = response)


dat = data.frame(response=response,A=rep(A,each=2),B=rep(B,each=2),C=rep(C,each=2))

normalplot <- function(y, label=F, n=length(y), fac.names=NULL, xlim=c(-2.2, 2.2), main="Normal Plot",
{ # label the most singificant n effects
m <- length(y)
x <- seq(0.5/m, 1.0-0.5/m, by=1/m)
x <- qnorm(x)
y <- sort(y)
qqplot(x, y,
xlab="normal quantiles", ylab="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)) ) )
ord=order(abs(y))
if(label) for(i in ord[(m-n+1):m]) text(x[i]+.35,y[i], fac.names[i],cex=0.9)
}
```

```r
halfnormalplot <- function(y, label=F, n=length(y), fac.names=NULL, xlim=c(-.1, 2.5), main="Half-Normal
{ # 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],cex=0.9)
}

res.lm = lm(response~A*B*C, data = dat)
summary(res.lm)
```

```
##
## Call:
## lm(formula = response ~ A * B * C, data = dat)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##     -2     -1      0      1      2
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   47.625      0.433 109.985 5.22e-14 ***
## A             -0.875      0.433  -2.021 0.077971 .
## B              0.375      0.433   0.866 0.411694
## C              1.500      0.433   3.464 0.008516 **
## A:B            1.625      0.433   3.753 0.005602 **
## A:C            2.500      0.433   5.774 0.000418 ***
## B:C            1.750      0.433   4.041 0.003728 **
## A:B:C          0.500      0.433   1.155 0.281537
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.732 on 8 degrees of freedom
## Multiple R-squared:  0.911,  Adjusted R-squared:  0.8332
## F-statistic:  11.7 on 7 and 8 DF,  p-value: 0.001223
```

```r
par(mfrow=c(1,2))
effects = 2*coef(res.lm)[-1]
normalplot(effects,label=T,type="p",pch=22,cex=0.7)
halfnormalplot(effects,label=T,type="p",pch=22,cex=0.7)
```



```
#Looking at the half normal probability plot, we can conclude the main effects are A and C, and the int

res.anova = anova(res.lm)
res.anova
```

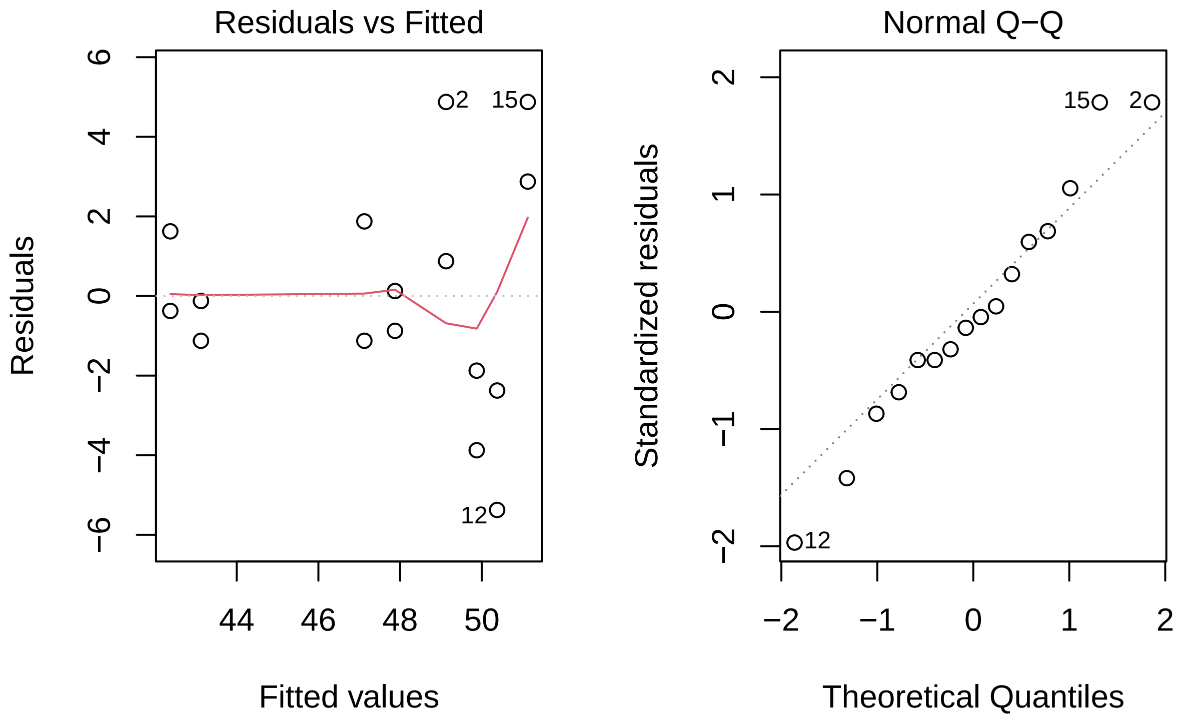
```
## Analysis of Variance Table
##
## Response: response
##           Df Sum Sq Mean Sq F value    Pr(>F)
## A          1  12.25   12.25  4.0833 0.0779708 .
## B          1   2.25    2.25  0.7500 0.4116944
## C          1  36.00   36.00 12.0000 0.0085163 **
## A:B        1  42.25   42.25 14.0833 0.0056019 **
## 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
```

```
res.lm2 = lm(response~A+B+C+A:C, data = dat)
summary(res.lm2)
```

```
##
## Call:
```

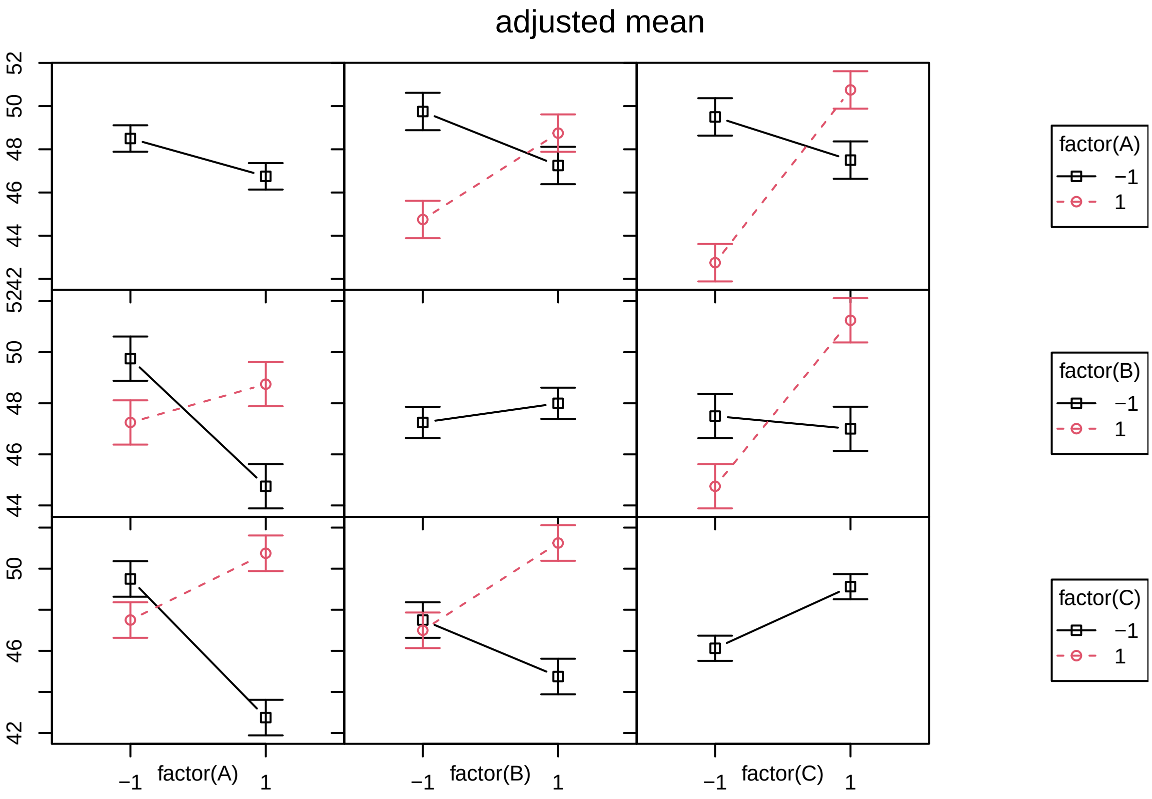
```
## lm(formula = response ~ A + B + C + A:C, data = dat)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -5.375 -1.312 -0.250  1.688  4.875
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  47.6250     0.8231  57.858 5.08e-15 ***
## A            -0.8750     0.8231  -1.063   0.3106
## B             0.3750     0.8231   0.456   0.6576
## C             1.5000     0.8231   1.822   0.0957 .
## A:C           2.5000     0.8231   3.037   0.0113 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.293 on 11 degrees of freedom
## Multiple R-squared:  0.5579, Adjusted R-squared:  0.3972
## F-statistic: 3.471 on 4 and 11 DF,  p-value: 0.04567
```

```r
par(mfrow=c(1,2))
plot(res.lm2,1:2)
```



```r
#Nothing unusual about residuals
```

```
aov <- aov(response ~ factor(A) * factor(B) * factor(C), data = dat)

factorial <- interactionMeans(aov, factors = c("factor(A)", "factor(B)" ,"factor(C)"))

plot(factorial)
```



```
A1 <- c(-1,1,-1,1,-1,1,-1,1,-1,1,-1,1,-1,1,-1,1)
B1 <- c(-1,-1,1,1,-1,-1,1,1,-1,-1,1,1,-1,-1,1,1)
C1 <- c(-1,-1,-1,-1,1,1,1,1,-1,-1,-1,-1,1,1,1,1)
D1 <- c(-1,-1,-1,-1,-1,-1,-1,-1,1,1,1,1,1,1,1,1)

response1 <- c(0.00340,
              0.00362,
              0.00301,
              0.00182,
              0.00280,
              0.00290,
              0.00252,
              0.00160,
              0.00336,
              0.00344,
              0.00308,
              0.00184,
              0.00269,
              0.00284,
              0.00253,
              0.00163)
data <- data.frame("A" = A1, "B" = B1, "C" = C1, "D" = D1, "response" = response)


m <- lm(response1 ~ A*B*C*D, data = data)
summary(m)
```

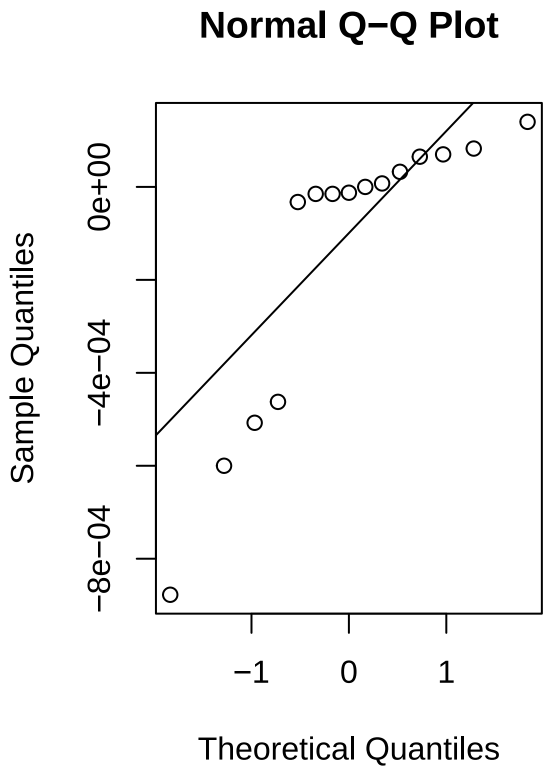
```
##
## Call:
## lm(formula = response1 ~ A * B * C * D, data = data)
##
## Residuals:
## ALL 16 residuals are 0: no residual degrees of freedom!
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.692e-03        NaN     NaN      NaN
## A           -2.312e-04        NaN     NaN      NaN
## B           -4.387e-04        NaN     NaN      NaN
## C           -2.537e-04        NaN     NaN      NaN
## D           -1.625e-05        NaN     NaN      NaN
## A:B         -3.000e-04        NaN     NaN      NaN
## A:C          3.500e-05        NaN     NaN      NaN
## B:C          7.000e-05        NaN     NaN      NaN
## A:D         -7.500e-06        NaN     NaN      NaN
## B:D          3.250e-05        NaN     NaN      NaN
## C:D         -8.868e-20        NaN     NaN      NaN
## A:B:C        4.125e-05        NaN     NaN      NaN
## A:B:D        3.750e-06        NaN     NaN      NaN
## A:C:D        1.625e-05        NaN     NaN      NaN
## B:C:D       -6.250e-06        NaN     NaN      NaN
## A:B:C:D     -7.500e-06        NaN     NaN      NaN
##
## Residual standard error: NaN on 0 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:    NaN
## F-statistic:   NaN on 15 and 0 DF,  p-value: NA
```

```
par(mfrow=c(1,2))
effects = 2*coef(m)[-1]
qqnorm(effects);qqline(effects)

#normalplot <- function(y, label=F, n=length(y), fac.names=NULL, xlim=c(-2.2, 2.2), main="Normal Plot",
#{ # label the most singificant n effects
#m <- length(y)
#x <- seq(0.5/m, 1.0-0.5/m, by=1/m)
#x <- qnorm(x)
#y <- sort(y)
#qqplot(x, y,
#xlab="normal quantiles", ylab="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)) ) )
```

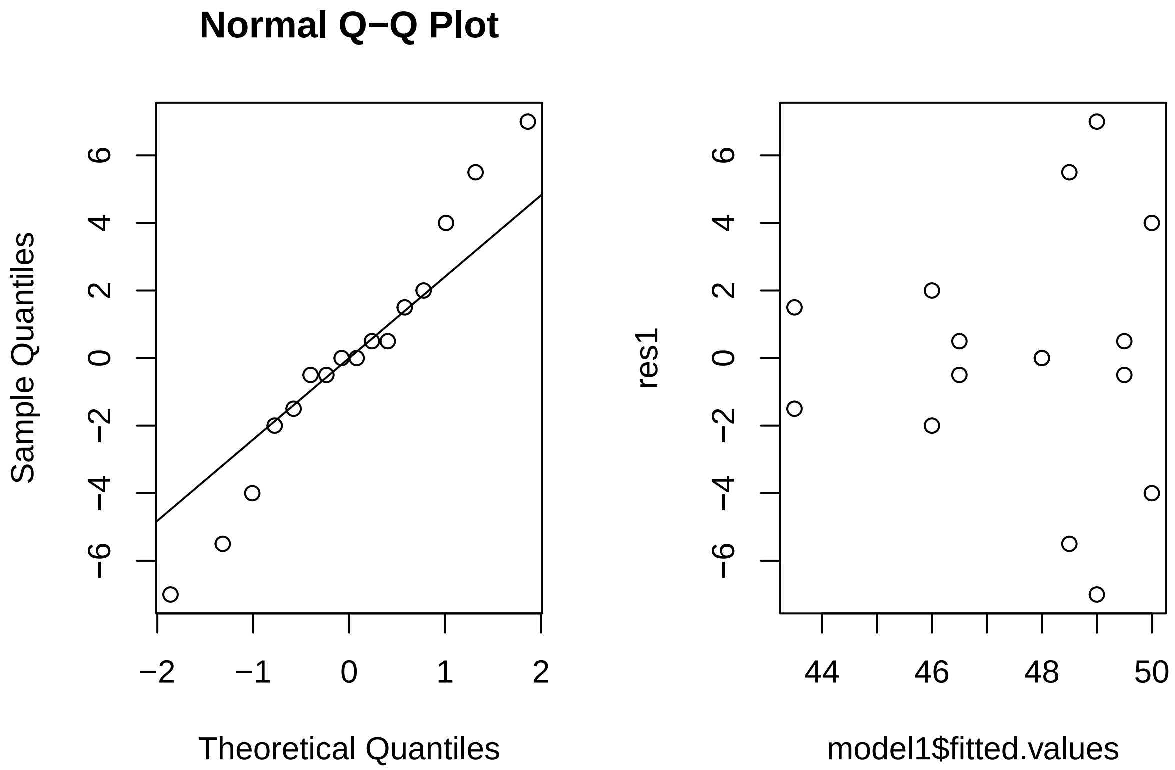
```r
#ord=order(abs(y))
#if(label) for(i in ord[(m-n+1):m]) text(x[i]+.35,y[i], fac.names[i],cex=0.9)
#}

model1 <- lm(response ~ A*B*C, data)
summary(model1)
```

```
##
## Call:
## lm(formula = response ~ A * B * C, data = data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -7.000 -1.625  0.000  1.625  7.000
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   47.625      1.262  37.725 2.68e-10 ***
## A             -0.125      1.262  -0.099    0.924
## B             -0.875      1.262  -0.693    0.508
## C              0.375      1.262   0.297    0.774
## A:B           -0.625      1.262  -0.495    0.634
## A:C            0.375      1.262   0.297    0.774
## B:C            1.625      1.262   1.287    0.234
## A:B:C          0.125      1.262   0.099    0.924
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.05 on 8 degrees of freedom
## Multiple R-squared:  0.2437, Adjusted R-squared:  -0.418
## F-statistic: 0.3683 on 7 and 8 DF,  p-value: 0.8971
```

```r
res1 <- model1$residuals
par(mfrow=c(1,2))
```



```
qqnorm(res1); qqline(res1)
plot(model1$fitted.values, res1)
```



```
dat$new <- 1/dat$response
#model2 <- lm(new ~ A*B*C*D, dat)
#summary(model2)
```

```
#Q3
#There can be 16 effects in a 2^4 full factorial model
#i.e.
x <- choose(4,0) + choose(4,1) + choose(4,2) + choose(4,3) + choose(4,4)
x
```

```
## [1] 16
```

```
#Q4
#2^3 replicated twice - (2^k)(n-1)
(2^3)*(2-1)
```

```
## [1] 8
```

```
#Q5
#2^3 replicated twice:
n <- (2^3)*2 #Total number of replicates
n
```

```
## [1] 16
```

```
#3 Factors
d <- 3
#residual df
f <- 1

#number of residual degrees of freedom for reduced
total = n - d - f
total
```

```
## [1] 12
```