

HW6: Blocking and Fractional Factorial Designs

MATH/STAT 571B

DUE: 4/19/2024 11:59pm

Homework Guidelines

Please submit your answers on Gradescope as a PDF with pages matched to question answers.

One way to prepare your solutions to this homework is with R Markdown, which provides a way to include mathematical notation, text, code, and figures in a single document. A template .Rmd file is available through D2L.

Make sure all solutions are clearly labeled, and please utilize the question pairing tool on Gradescope. You are encouraged to work together, but your solutions, code, plots, and wording should always be your own. Come and see me during office hours or schedule an appointment when you get stuck and can't get unstuck.

I. Mathematical Foundations [19 pts]

- (1) [4 pts] A collaborator sends you a file with the following tabular data for a full factorial experiment with three factors, A , B , and C :

A	B	C	y
–	–	–	y_1
+	–	–	y_2
–	+	–	y_3
+	+	–	y_4
–	–	+	y_5
+	–	+	y_6
–	+	+	y_7
+	+	+	y_8

He also tells you that the experiment was run in two blocks, but he forgot to record which observation came from which block. Looking back at his notes, he recites that "... blocking was chosen to confound blocks with the interaction of factors B and C ." Use this information to create a column called 'block' that indicates with 1s and 2s which observations belong to each of the two blocks.

```
levels <- c(-1, 1)
df <- expand.grid(A = levels, B = levels, C = levels)
df
```

```
##      A  B  C
## 1 -1 -1 -1
## 2  1 -1 -1
```

```
## 3 -1 1 -1
## 4 1 1 -1
## 5 -1 -1 1
## 6 1 -1 1
## 7 -1 1 1
## 8 1 1 1
```

```
df <- df %>%
  mutate(Block= ifelse(A*B*C == -1, 1,
                        ifelse(A*B*C == 1, 2, NA)))
df
```

```
##      A  B  C Block
## 1 -1 -1 -1      1
## 2 1 -1 -1      2
## 3 -1 1 -1      2
## 4 1 1 -1      1
## 5 -1 -1 1      2
## 6 1 -1 1      1
## 7 -1 1 1      1
## 8 1 1 1      2
```

- (2) [9 pts] A sequential experiment was carried out for analyzing the effects of varying four factors, A , B , C , and D on a response, y . In the first phase of the experiment, one quarter of the full factorial design was carried out according to the defining relation $I = ABC = BCD$. In the second phase of the experiment, another quarter of the full factorial design was carried out according to the defining relation $I = -ABC = BCD$.

- (a) [3 pts] What is the resolution of the phase 1 design? Explain your reasoning. **Phase 1: $I = ABC = BCD = AD$, AD is shortest so Resolution 2.**
- (b) [3 pts] What is the resolution of the design after combining phases 1 and 2? Explain your reasoning. **Phase 1: $I = ABC = BCD = AD$; Phase 2: $I = -ABC = BCD = -AD$; When added, you get $I = BCD$ so resolution 3.**

In the third and final phase, one more quarter of the full factorial design was carried out according to the defining relation $I = ABC = -BCD$. A model with all first and second-order factors was fit to the resulting 12 runs that produced the following R output.

- (c) [3 pts] Explain how we can be sure from the output that the resolution of the combined three phase design is at least IV. **The output shows that both 1st and 2nd order coefficients are not being confounded. Therefore, at least 2nd order coefficients (or higher) are being aliased.**
- (3) [6 pts] A full fold-over two-phase experiment was carried out for analyzing the effects of varying five factors, A , B , C , D , and E on a response, y . In the first phase of the experiment, one quarter of the full factorial design was carried out according to the defining relation $I = ABC = CDE$. In the second phase of the experiment, another quarter of the full factorial design was carried out to create a full fold-over design.
- (a) [3 pts] What is the resolution of the phase 1 design? Explain your reasoning. **Phase 1: $I = ABC = CDE = ABDE$; ABC and CDE are the shortest, so the resolution is III**
- (b) [3 pts] What is the resolution of the design after combining phases 1 and 2? Explain your reasoning. **Phase 1: $I = ABC = CDE = ABDE$; Phase 2: $I = -ABC = -CDE = ABDE$; Together: $I = ABDE$ so Resolution IV.**

II. Applications [21 pts]

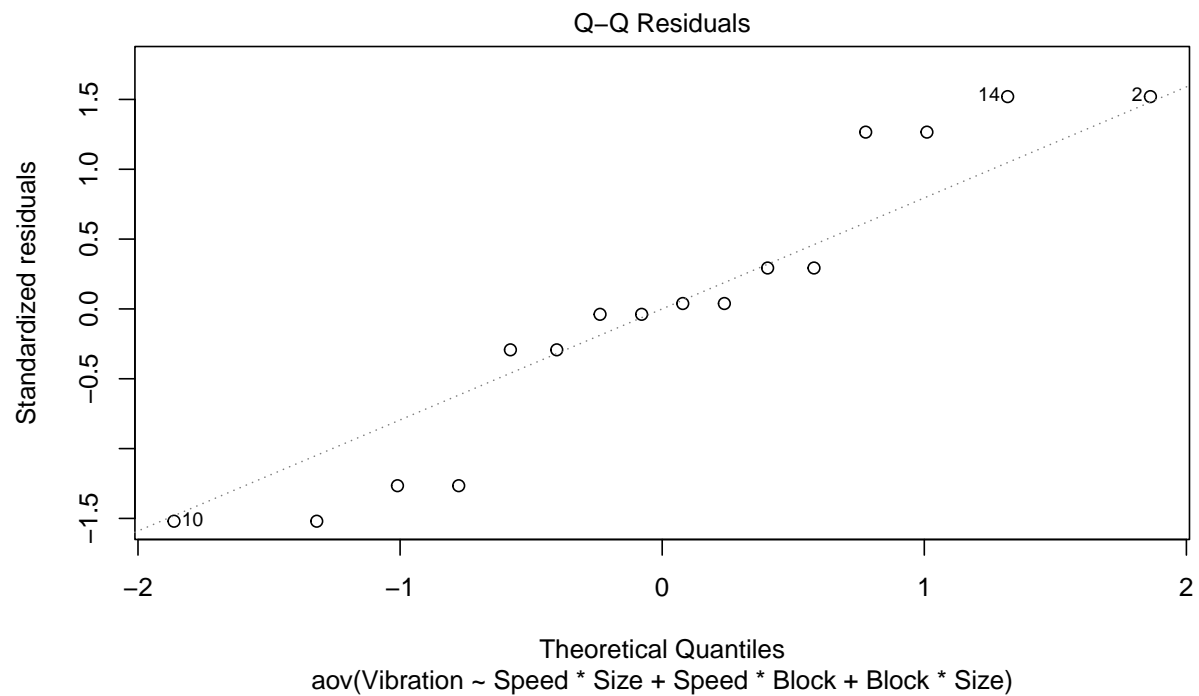
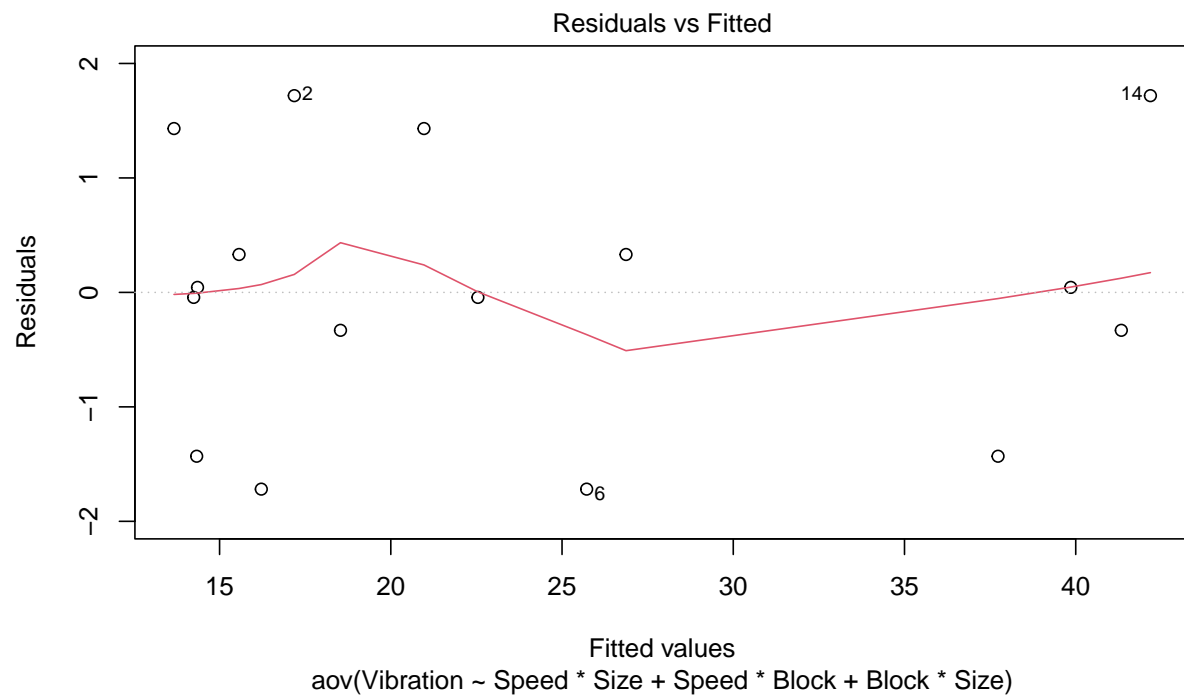
- (4) [7 pts] (DAE 7.2) *Hint: You will need to create your own variable indicating which replicate each observation is associated with.* I removed the Size:Speed:Block interaction to free up degrees of freedom for prediction. This test shows that Size, Speed, and Size:Speed appear to be significant at the $\alpha=0.05$ level. However, the blocks do not appear to be significant. The diagnostics show normal residuals and constant variance for the fitted values. However, there is some difference in the residuals against the blocks.

```
vibration_blk <- read.csv("../Datasets/Q6-5.csv")
vibration_blk <- vibration_blk %>%
  mutate(Block=rep(1:4,4)) %>%
  mutate_at(vars(-Vibration), as.factor)

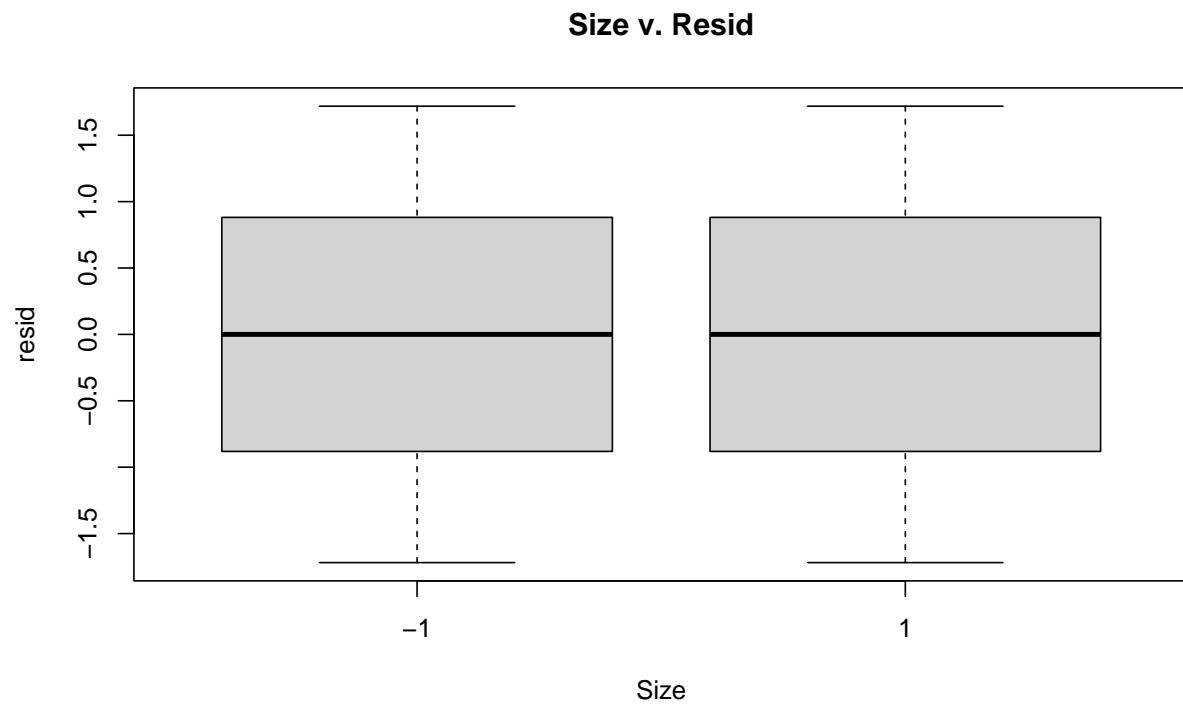
# use recommended blocking
vibration_aov_blk <- aov(Vibration ~ Speed*Size + Speed*Block + Block*Size, data = vibration_blk)
anova(vibration_aov_blk)

## Analysis of Variance Table
##
## Response: Vibration
##          Df Sum Sq Mean Sq F value    Pr(>F)
## Speed      1  227.26   227.26  33.3270 0.010333 *
## Size       1 1107.23  1107.23 162.3746 0.001043 **
## Block      3   44.36    14.79   2.1686 0.270653
## Speed:Size  1  303.63   303.63  44.5274 0.006863 **
## Speed:Block 3    4.63    1.54   0.2264 0.873056
## Size:Block  3    2.27    0.76   0.1111 0.947990
## Residuals  3   20.46    6.82
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

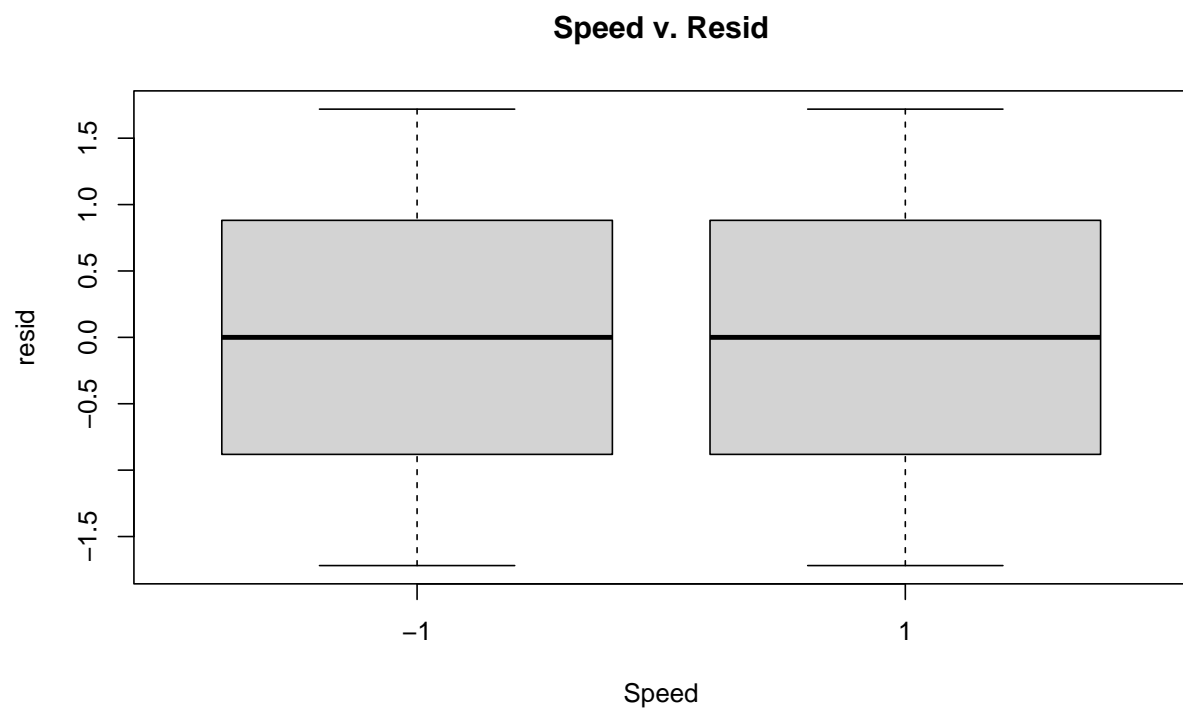
#QQ, residuals v. fitted
plot(vibration_aov_blk, which = 1:2)
```



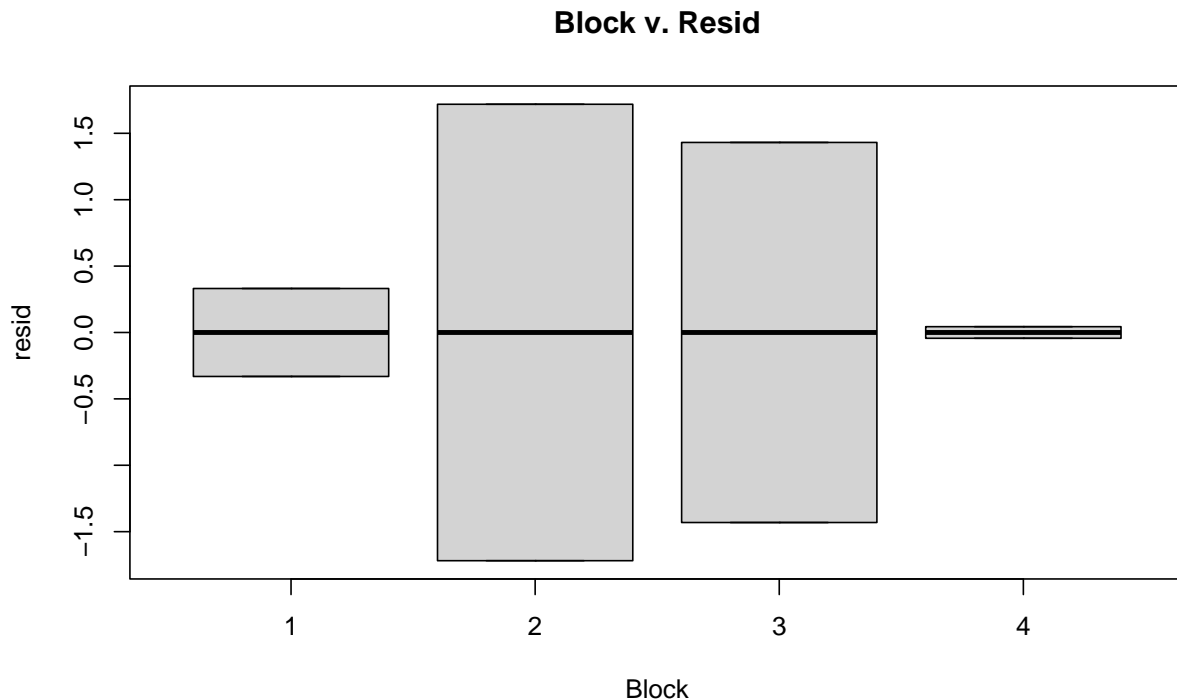
```
#Residuals v. predictors
resids <- resid(vibration_aov_blk)
plot(vibration_blk$Size, resids, xlab="Size", ylab="resid", main = "Size v. Resid")
```



```
plot(vibration_blk$Speed, resid, xlab="Speed", ylab="resid", main = "Speed v. Resid")
```



```
plot(vibration_blk$Block, resid, xlab="Block", ylab="resid", main = "Block v. Resid")
```



```
# see why residuals are so different
resid_check <- matrix(resids, byrow = T, nrow = 4)
rownames(resid_check) <- c(1:4)
colnames(resid_check) <- c("(1)", "a", "b", "ab")
resid_check
```

```
##      (1)      a      b      ab
## 1 -0.33125  1.71875 -1.43125  0.04375
## 2  0.33125 -1.71875  1.43125 -0.04375
## 3  0.33125 -1.71875  1.43125 -0.04375
## 4 -0.33125  1.71875 -1.43125  0.04375
```

- (5) [7 pts] (DAE 7.5) *Hint: Similar to question (1), you will need to create an indicator variable for block. Verify your block averages for Yield are 85 and 81.75. At an $\alpha = 0.05$, there appears to be evidence of significant factors A, D, AD, ABC, ABD to on yield. These interactions also imply the lower terms AC and AB to be significant. The diagnostic plots show normal residuals and constant variance across fitted values and predictors (except possibly in the B factor).*

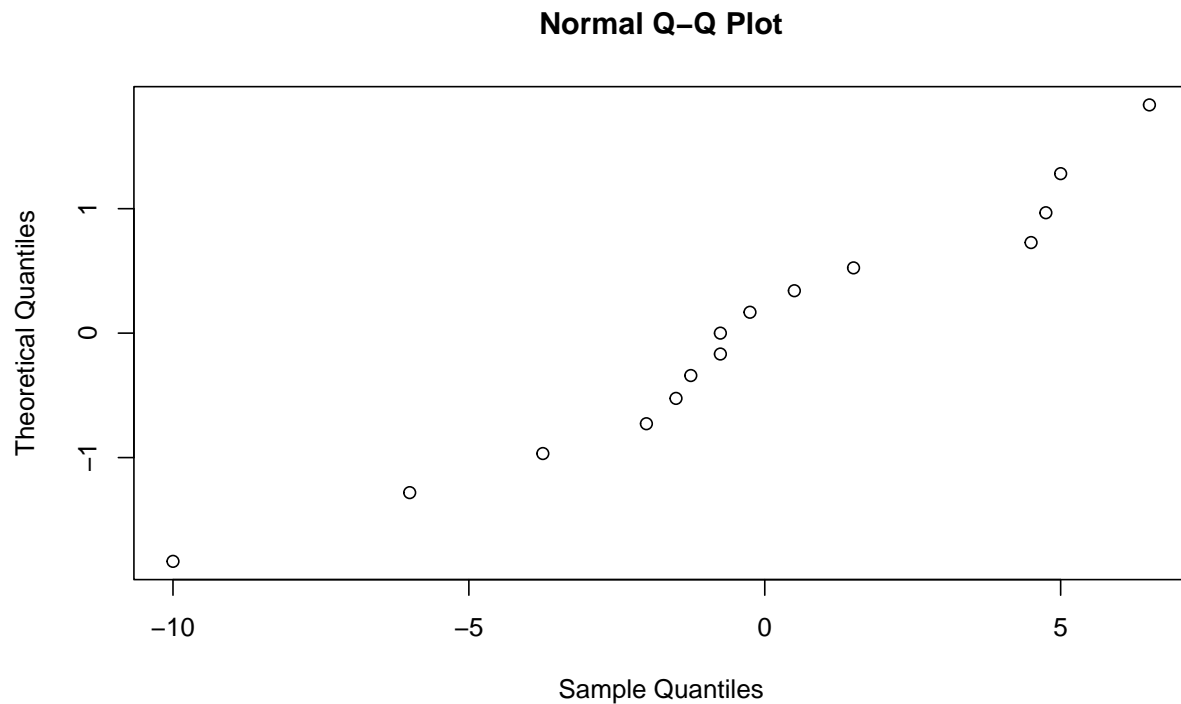
```
yield_blk <- read.csv("./Datasets/Q6-7.csv")
yield_blk <- yield_blk[yield_blk$replicate=="I",] %>%
  mutate(ABCD=A*B*C*D) %>%
  mutate(Block= ifelse(ABCD == -1, 1,
                       ifelse(ABCD == 1, 2, NA))) %>%
  mutate_at(vars(-Yield), as.factor)
```

```
tapply(yield_blk$Yield, yield_blk$Block, mean)
```

```
##      1      2
## 81.75 85.00
```

```
# anova
yield_aov_blk <- aov(Yield ~ (A+B+C+D)^3 + Block, data = yield_blk,
  contrasts = list(A="contr.helmert",
    B="contr.helmert",
    C="contr.helmert",
    D="contr.helmert"))
```

```
# normal plot to remove some interactions
coefs <- coef(yield_aov_blk)
qqnorm(coefs[-1] * 2, datax = T)
```



```
# likely significant coeffs: A, ABC, AB, ABD, D, Block
# required lower order coeffs: BC, AC, AD, BD, B, C
# likely not significant and removable: BCD, ACD, CD
sort(coefs[-1]) # REMOVE THE VALUES THAT HAVE THE LOWEST ABSOLUTE VALUE
```

```
##      A1 A1:B1:C1      A1:D1 B1:C1:D1      B1:D1      B1:C1      B1      C1
## -5.000 -3.000 -1.875 -1.000 -0.750 -0.625 -0.375 -0.375
## A1:C1:D1 A1:C1      C1:D1      A1:B1 A1:B1:D1      D1      Block2
## -0.125  0.250  0.750  2.250  2.375  2.500  3.250
```

```
# redo anova with removed interactions
yield_aov_blk2 <- aov(Yield ~ A + A:B:C + A:B + A:B:D + D + Block + B:C + A:C + A:D + B:D + B + C, data = data)
contrasts = list(A="contr.helmert",
                 B="contr.helmert",
                 C="contr.helmert",
                 D="contr.helmert")
summary(lm(yield_aov_blk2))
```

```
##
## Call:
## lm(formula = yield_aov_blk2)
##
## Residuals:
```

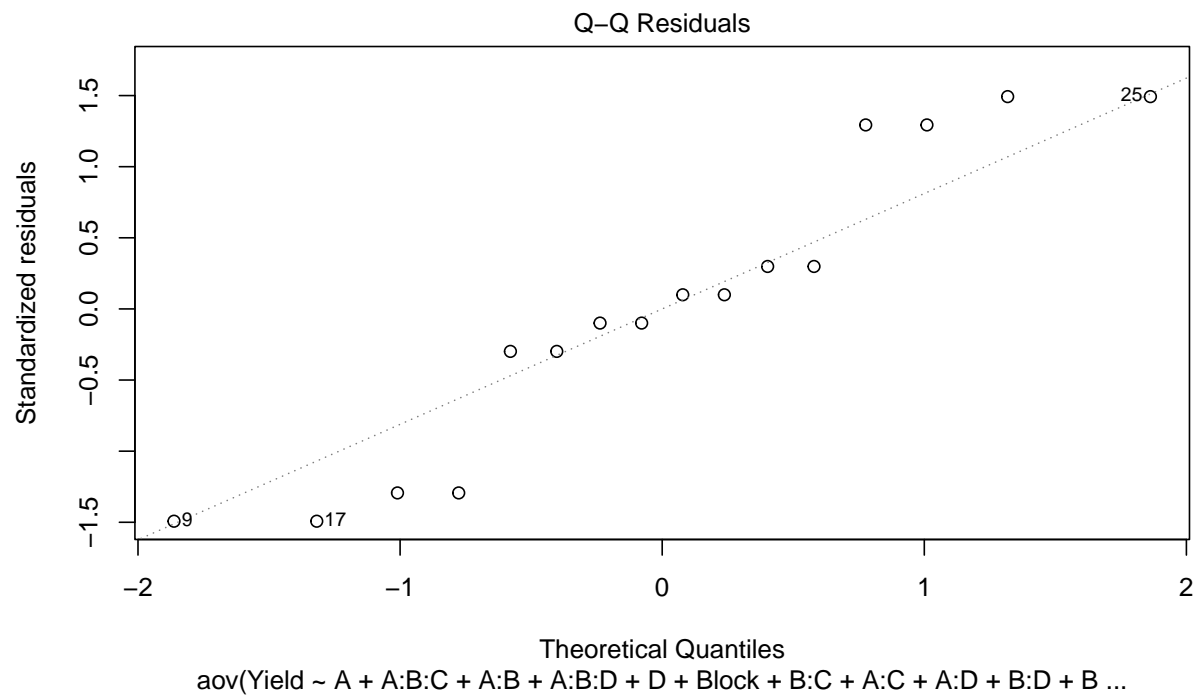
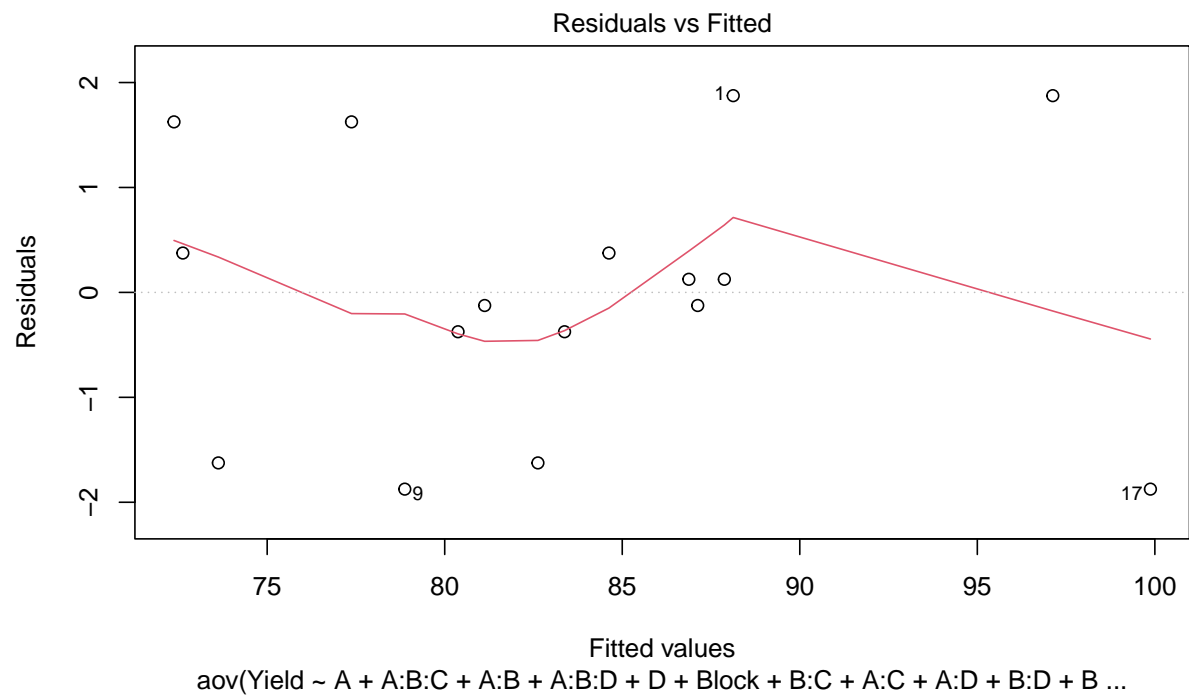
##	1	3	5	7	9	11	13	15	17	19	21
##	1.875	1.625	-0.125	-0.375	-1.875	-1.625	0.125	0.375	-1.875	-1.625	0.125
##	23	25	27	29	31						
##	0.375	1.875	1.625	-0.125	-0.375						

```
##
## Coefficients:
```

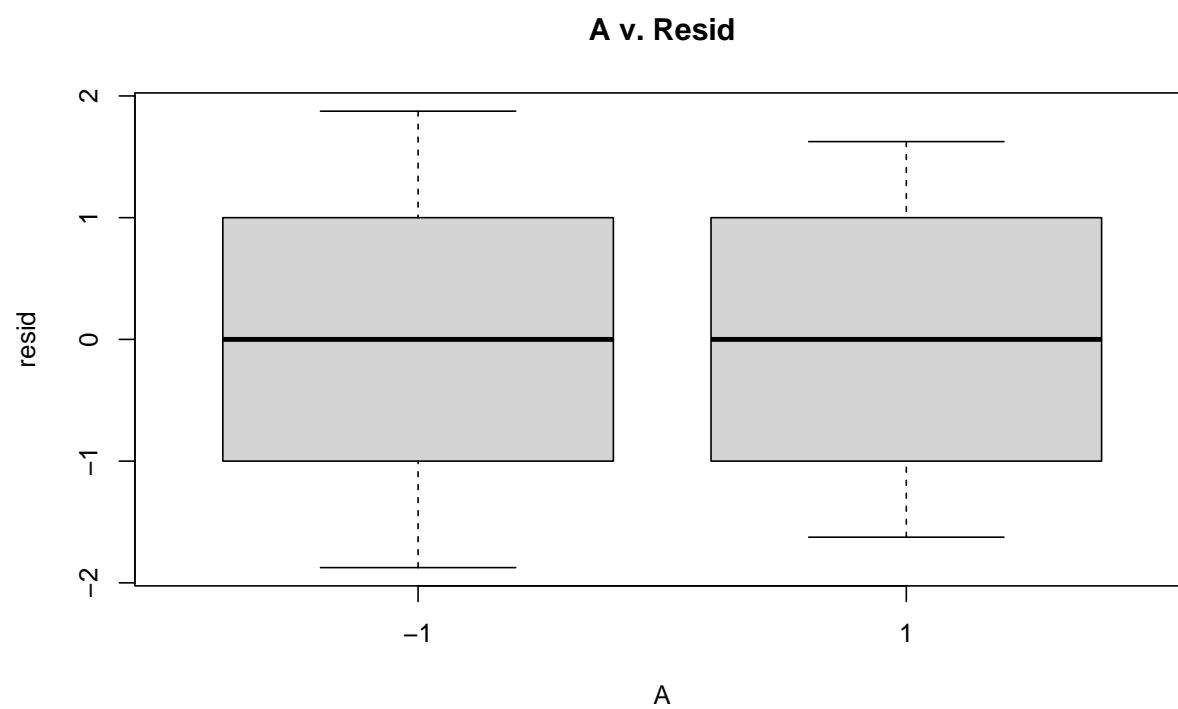
##		Estimate	Std. Error	t value	Pr(> t)
##	(Intercept)	84.875	2.615	32.456	6.43e-05 ***
##	A1	-12.500	3.553	-3.518	0.0390 *
##	D1	15.000	2.901	5.170	0.0140 *
##	Block2	3.250	1.451	2.240	0.1109
##	B1	-3.750	3.553	-1.055	0.3687
##	C1	-6.000	2.901	-2.068	0.1305
##	A1:B1	11.500	5.025	2.289	0.1061
##	B1:C1	9.500	4.103	2.315	0.1035
##	A1:C1	13.000	4.103	3.169	0.0505 .
##	A1:D1	-17.000	4.103	-4.143	0.0255 *
##	B1:D1	-12.500	4.103	-3.047	0.0556 .
##	A1:B1:C1	-24.000	5.802	-4.136	0.0256 *
##	A1:B1:D1	19.000	5.802	3.275	0.0466 *

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.901 on 3 degrees of freedom
## Multiple R-squared:  0.9737, Adjusted R-squared:  0.8685
## F-statistic: 9.252 on 12 and 3 DF,  p-value: 0.04625
```

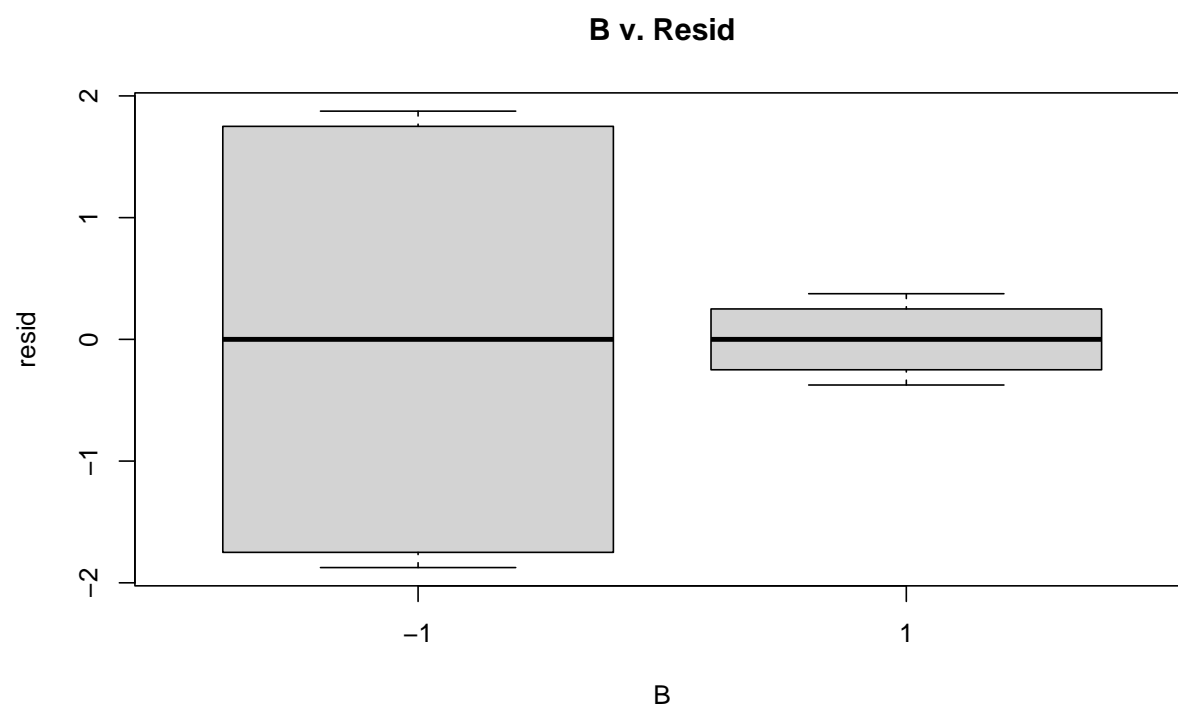
```
#QQ, residuals v. fitted
plot(yield_aov_blk2, which = 1:2)
```

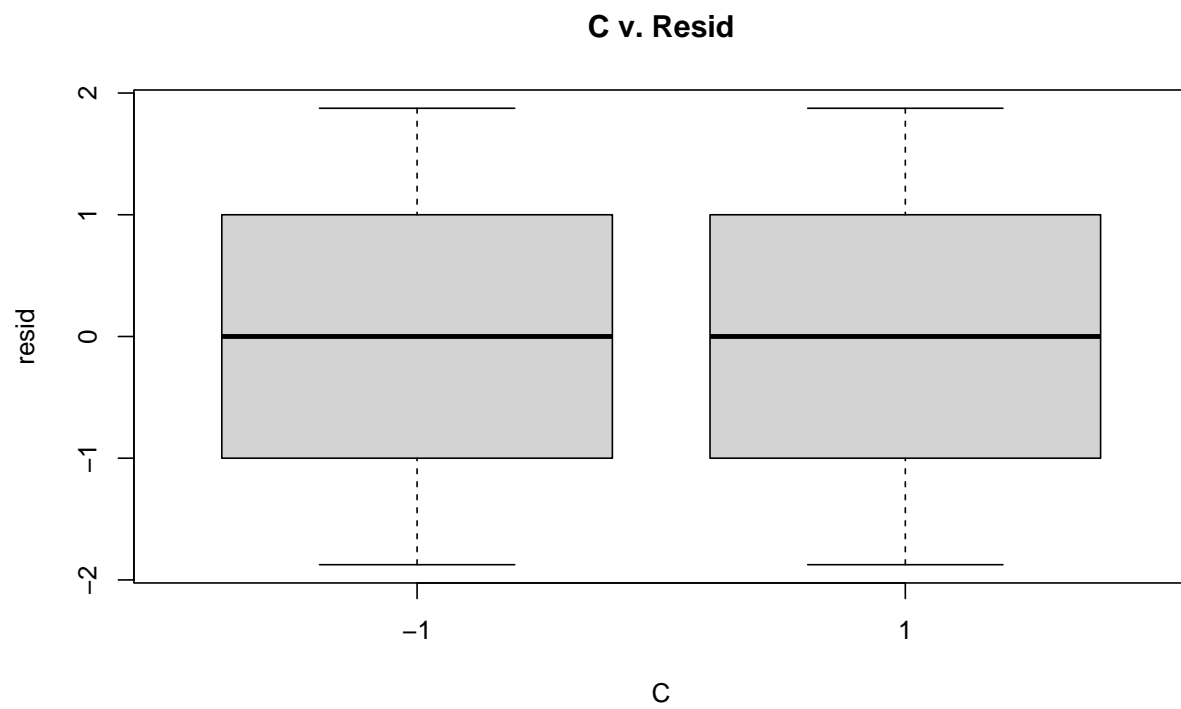
```
#Residuals v. predictors
resids <- resid(yield_aov_blk2)
plot(yield_blk$A, resids, xlab="A", ylab="resid", main = "A v. Resid")
```



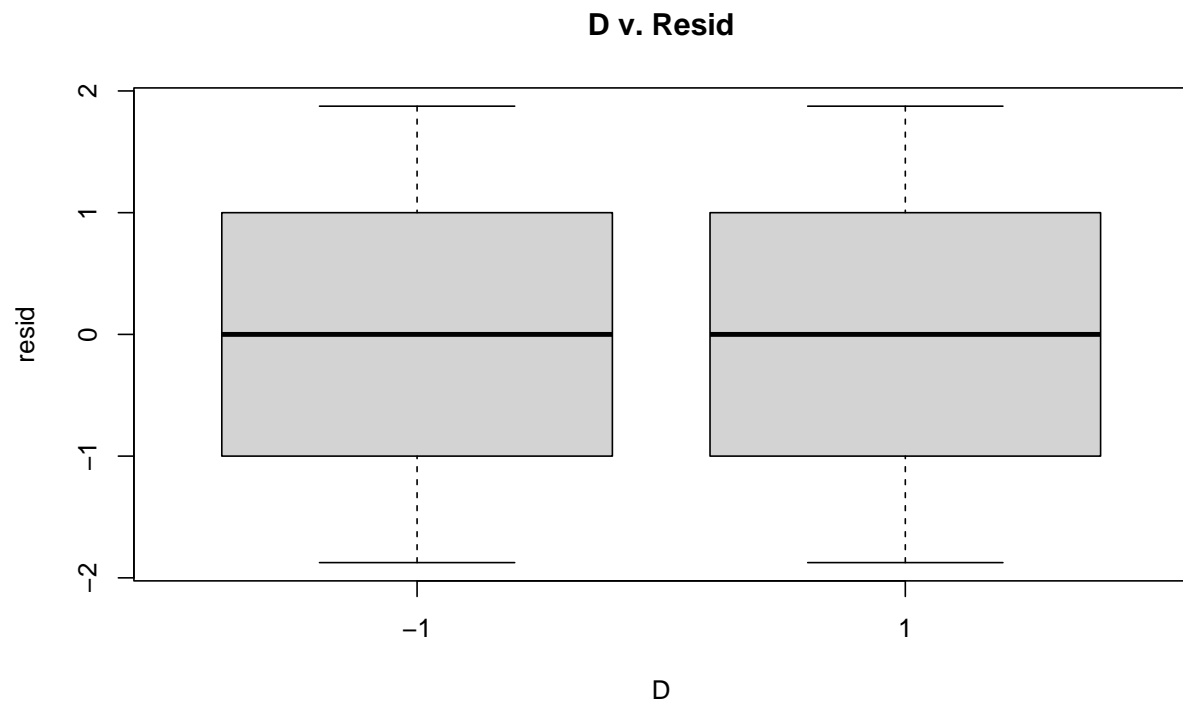
```
plot(yield_blk$B, resid, xlab="B", ylab="resid", main = "B v. Resid")
```



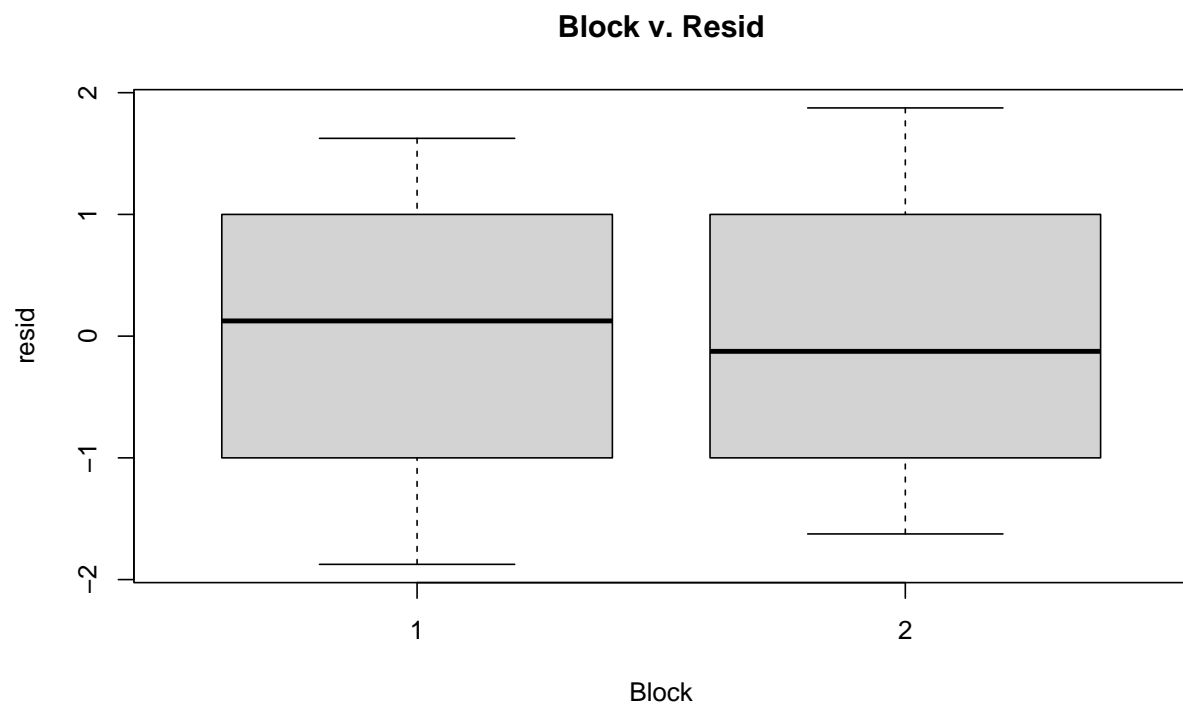
```
plot(yield_bk$C, resid, xlab="C", ylab="resid", main = "C v. Resid")
```



```
plot(yield_bk$D, resid, xlab="D", ylab="resid", main = "D v. Resid")
```



```
plot(yield_blk$Block, resid, xlab="Block", ylab="resid", main = "Block v. Resid")
```



(6) [7 pts] (DAE 8.1) Define the one-half fraction using the principal part of the design that maximizes the

resolution. Verify that your sample variance for Yield across the resulting 8 runs is 64. By testing the formula $\text{Yield} \sim AD + AB + C$, I found that there were no significant factors that explained yield at $\alpha = 0.05$ level. However, the QQ residuals did not seem normal. The residuals v. fitted and predictors showed constant variance except for factor B.

```
yield_fc <- read.csv("./Datasets/Q6-7.csv")
yield_fc <- yield_fc[yield_fc$replicate=="I",] %>%
  mutate(ABCD=A*B*C*D) %>%
  mutate(Block= ifelse(ABCD == -1, 1,
                       ifelse(ABCD == 1, 2, NA))) %>%
  mutate_at(vars(-Yield), as.factor)

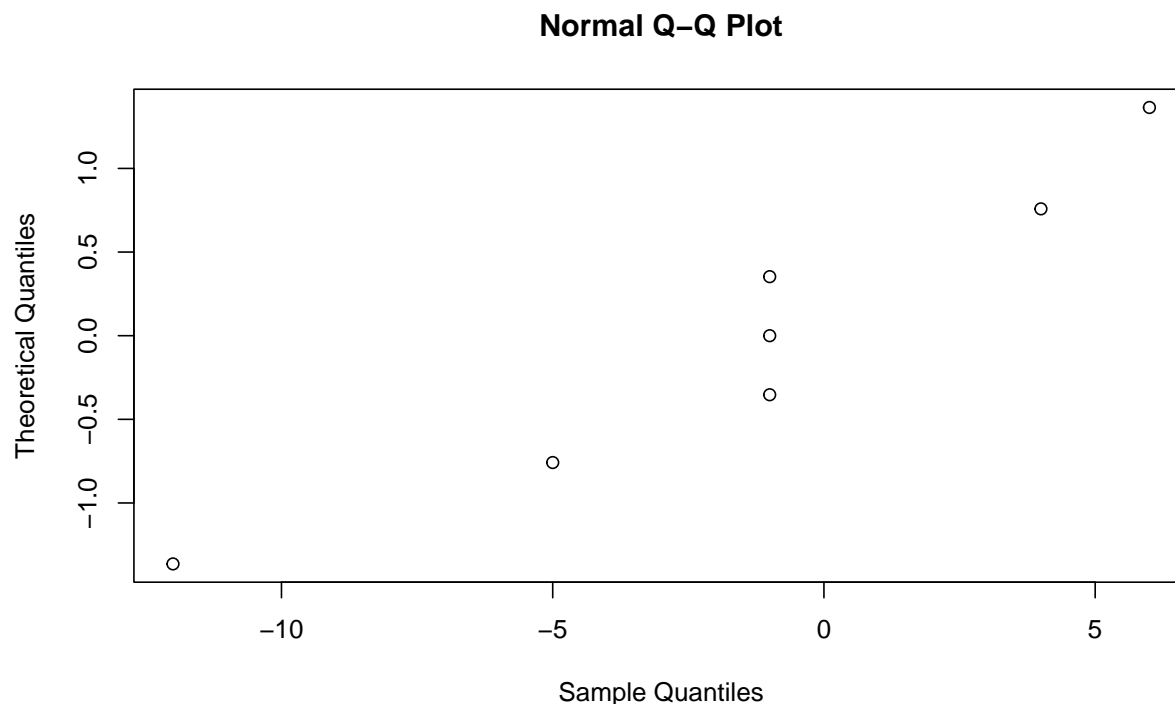
tapply(yield_fc$Yield, yield_fc$Block, sd)^2 # Take block 2; I=ABCD
```

```
##          1          2
## 67.07143 64.00000
```

```
yield_aov_fc <- aov(Yield ~ (A+B+C+D)^2, data = yield_fc[yield_fc$Block==2,],
  contrasts = list(A="contr.helmert",
                  B="contr.helmert",
                  C="contr.helmert",
                  D="contr.helmert"))
```

normal plot to remove some interactions

```
coefs <- coef(yield_aov_fc)
qqnorm(coefs[-1] * 2, datax = T)
```



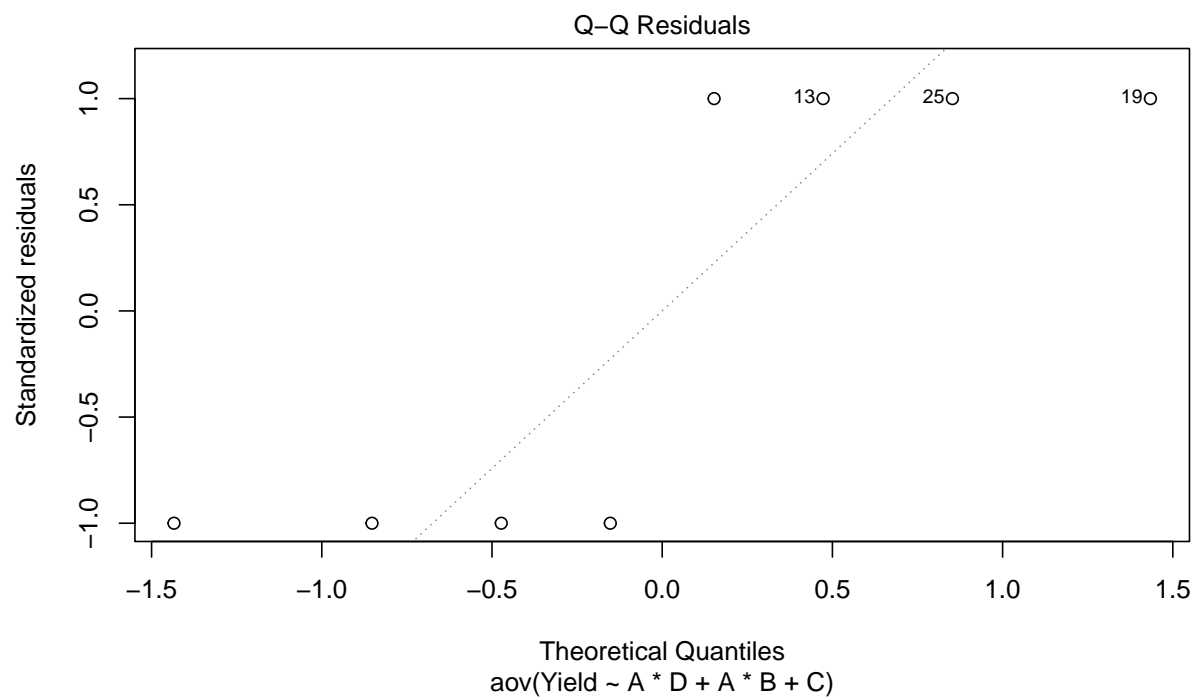
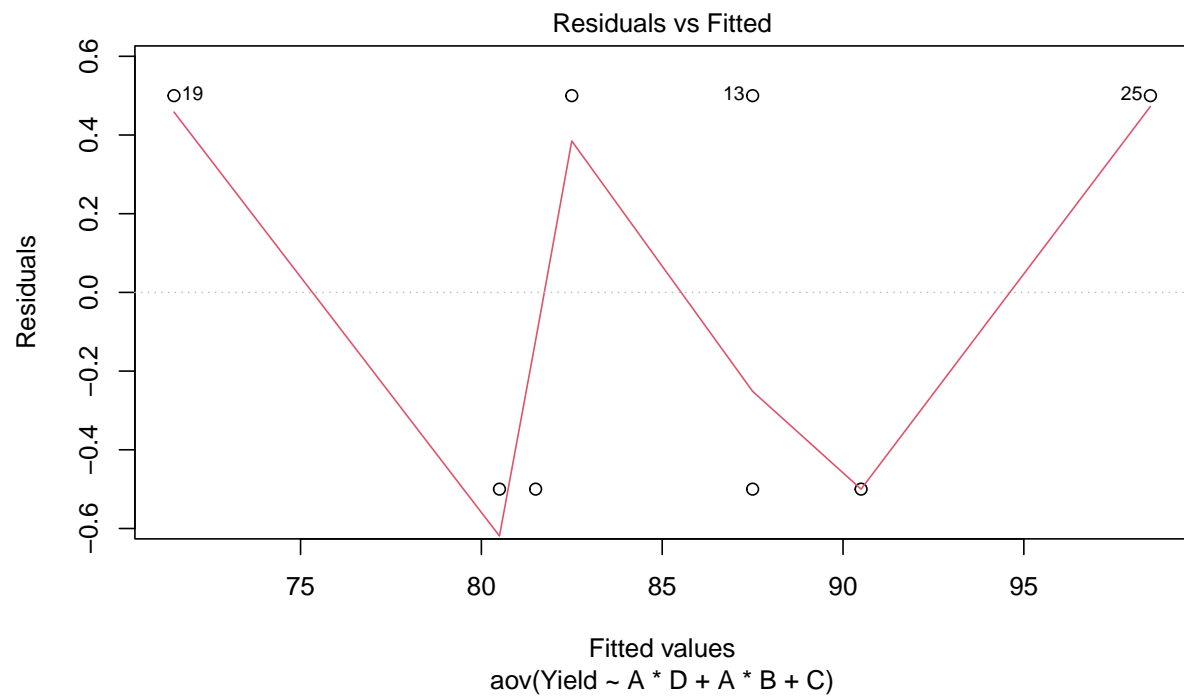
```
# Remove: AC interaction
sort(coefs[-1])
```

```
##      A1 A1:D1 A1:C1      B1      D1      C1 A1:B1
##    -6.0  -2.5  -0.5  -0.5  -0.5    2.0   3.0
```

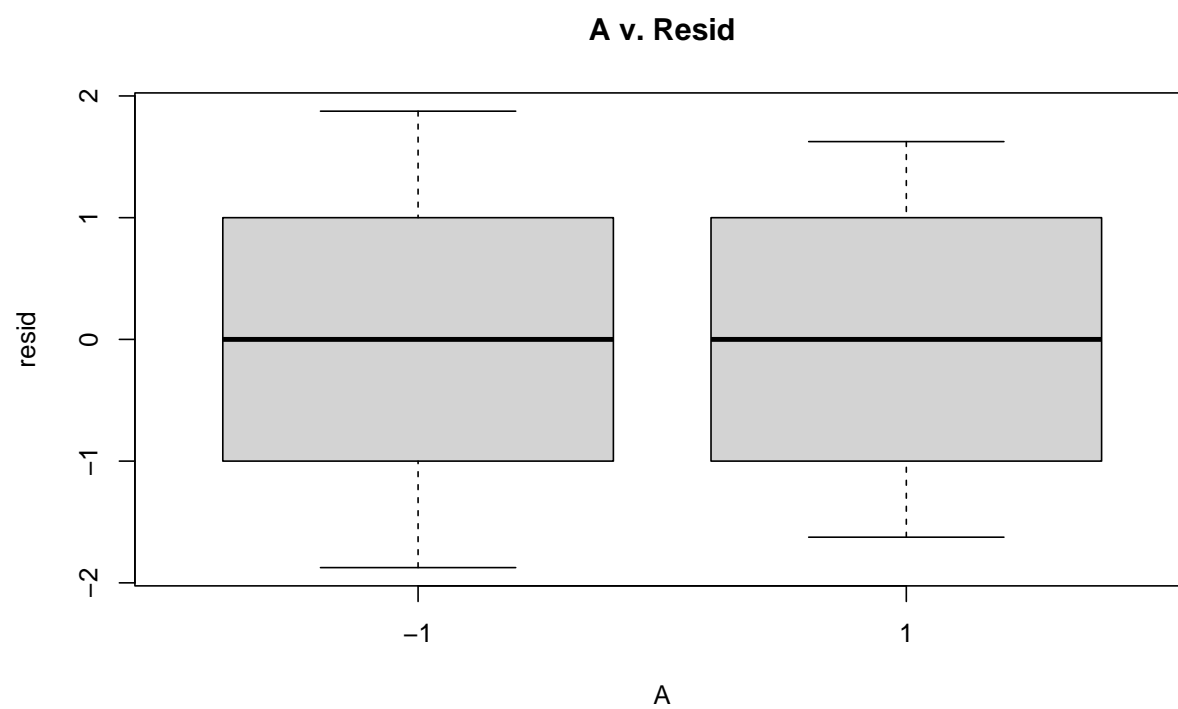
```
# Remove AC coefficient for re-running
yield_aov_fc <- aov(Yield ~ A*D + A*B + C, data = yield_fc[yield_fc$Block==2,],
  contrasts = list(A="contr.helmert",
    B="contr.helmert",
    C="contr.helmert",
    D="contr.helmert"))
summary(lm(yield_aov_fc))
```

```
##
## Call:
## lm(formula = yield_aov_fc)
##
## Residuals:
##      1      7     11     13     19     21     25     31
##    -0.5   0.5  -0.5   0.5   0.5  -0.5   0.5  -0.5
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    90.500      1.323   68.412  0.00931 **
## A1             -13.000      1.732  -7.506  0.08432 .
## D1              4.000      1.414   2.828  0.21635
## B1             -7.000      1.414  -4.950  0.12691
## C1              4.000      1.000   4.000  0.15596
## A1:D1          -10.000      2.000  -5.000  0.12567
## A1:B1           12.000      2.000   6.000  0.10514
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.414 on 1 degrees of freedom
## Multiple R-squared:  0.9955, Adjusted R-squared:  0.9688
## F-statistic: 37.17 on 6 and 1 DF, p-value: 0.1249
```

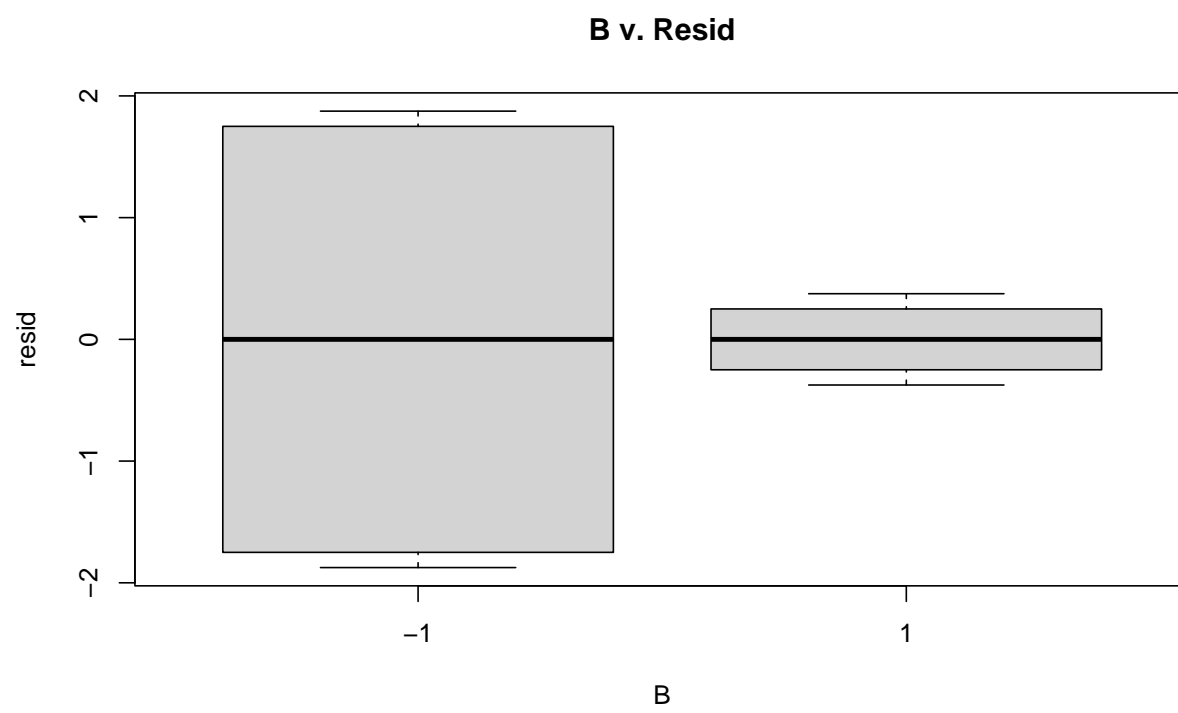
```
#QQ, residuals v. fitted
plot(yield_aov_fc, which = 1:2)
```



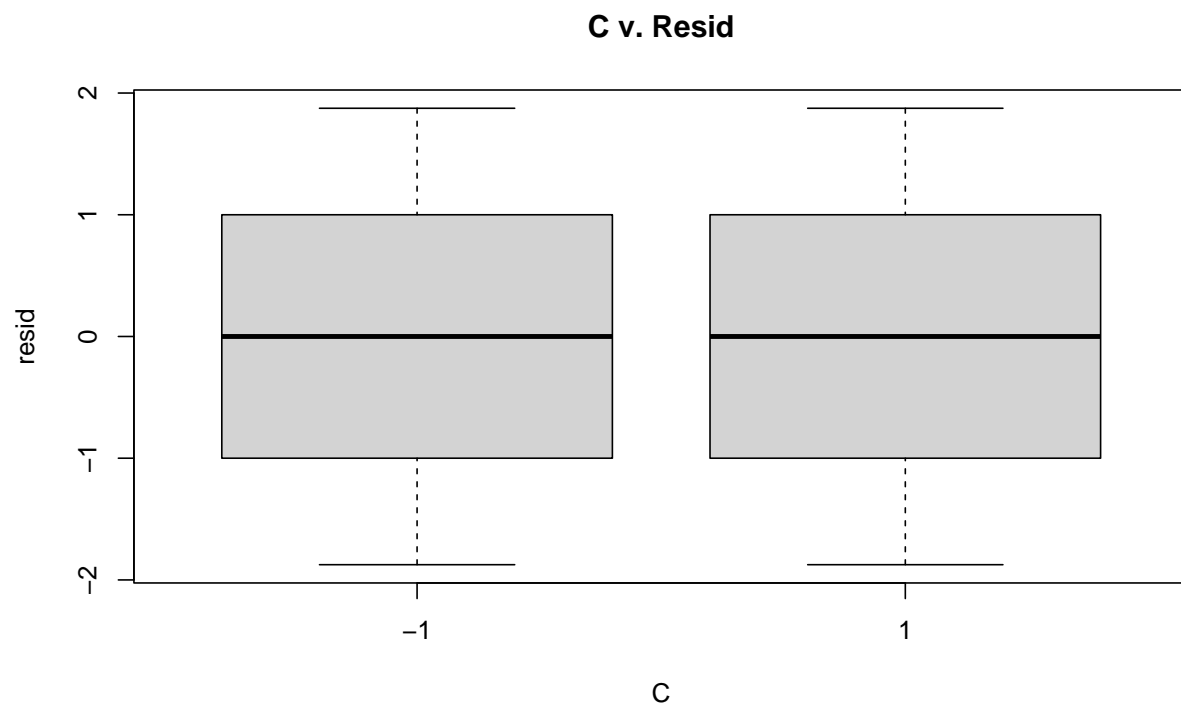
```
#Residuals v. predictors
resids <- resid(yield_aov_blk2)
plot(yield_blk$A, resids, xlab="A", ylab="resid", main = "A v. Resid")
```



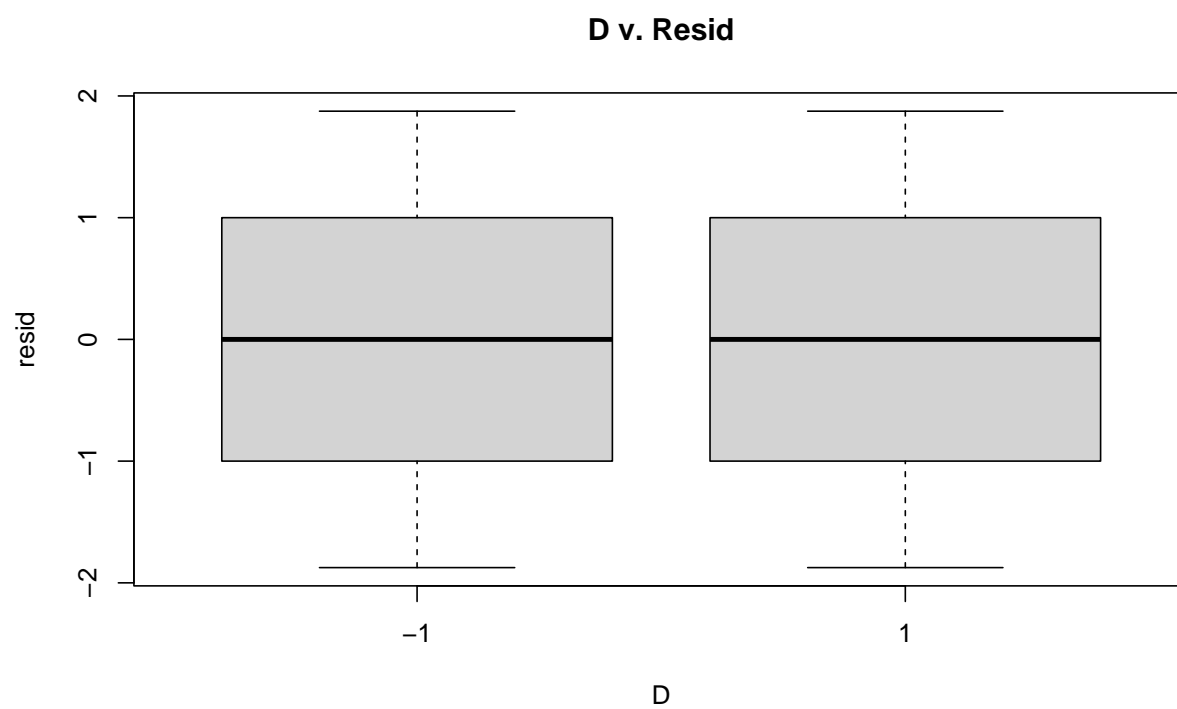
```
plot(yield_bk$B, resid, xlab="B", ylab="resid", main = "B v. Resid")
```




```
plot(yield_bk$C, resid, xlab="C", ylab="resid", main = "C v. Resid")
```



```
plot(yield_bk$D, resid, xlab="D", ylab="resid", main = "D v. Resid")
```



```
plot(yield_bk$Block, resid, xlab="Block", ylab="resid", main = "Block v. Resid")
```

