Full Factorial DOE

2023-07-16

Purpose: I will be designing and analyzing a data set, containing different factors and yield as a response variable, using the Full Factorial method.

For this DOE project I will be making use of the `AlgDesign` and `lattice` packages.

```
library(AlgDesign)
library(lattice)
```

I will be creating a full factorial DOE with three factors (F1, F2, F3) and with 3, 2, and 3 levels respectively.

Let's generate the factorial design:

```
gen.factorial(levels = c(3, 2, 3), nVars = 3, center = TRUE, varNames = c("F1", "F2", "F3"))
```

```
##    F1 F2 F3
## 1  -1 -1 -1
## 2   0 -1 -1
## 3   1 -1 -1
## 4  -1  1 -1
## 5   0  1 -1
## 6   1  1 -1
## 7  -1 -1  0
## 8   0 -1  0
## 9   1 -1  0
## 10 -1  1  0
## 11  0  1  0
## 12  1  1  0
## 13 -1 -1  1
## 14  0 -1  1
## 15  1 -1  1
## 16 -1  1  1
## 17  0  1  1
## 18  1  1  1
```

```
# We have set center parameter to TRUE so that all non-factors are centered, making the design symmetri
```

As seen above we will be running 18 experiments (known to us by multiplying out the factor levels 3 * 2 * 3).

```
# This converts the data frame into a full factorial design

dat = gen.factorial(levels = 2, nVars = 3, varNames = c("F1", "F2", "F3"))
dat
```

```
##   F1 F2 F3
## 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
```

```
# This is a factorial design example, two-level three factor design with Yield as the response variable
yield <- read.csv("FactorialDesign.csv")
yield
```

```
##   Run Yield Temp Conc Catalyst
## 1   1    60  160   20        0
## 2   2    72  180   20        0
## 3   3    54  160   40        0
## 4   4    68  180   40        0
## 5   5    52  160   20        1
## 6   6    83  180   20        1
## 7   7    45  160   40        1
## 8   8    80  180   40        1
```

We will clean up the data frame by removing the unneeded Run column

```
yield <- yield[, -1]
yield
```

```
##   Yield Temp Conc Catalyst
## 1    60  160   20        0
## 2    72  180   20        0
## 3    54  160   40        0
## 4    68  180   40        0
## 5    52  160   20        1
## 6    83  180   20        1
## 7    45  160   40        1
## 8    80  180   40        1
```

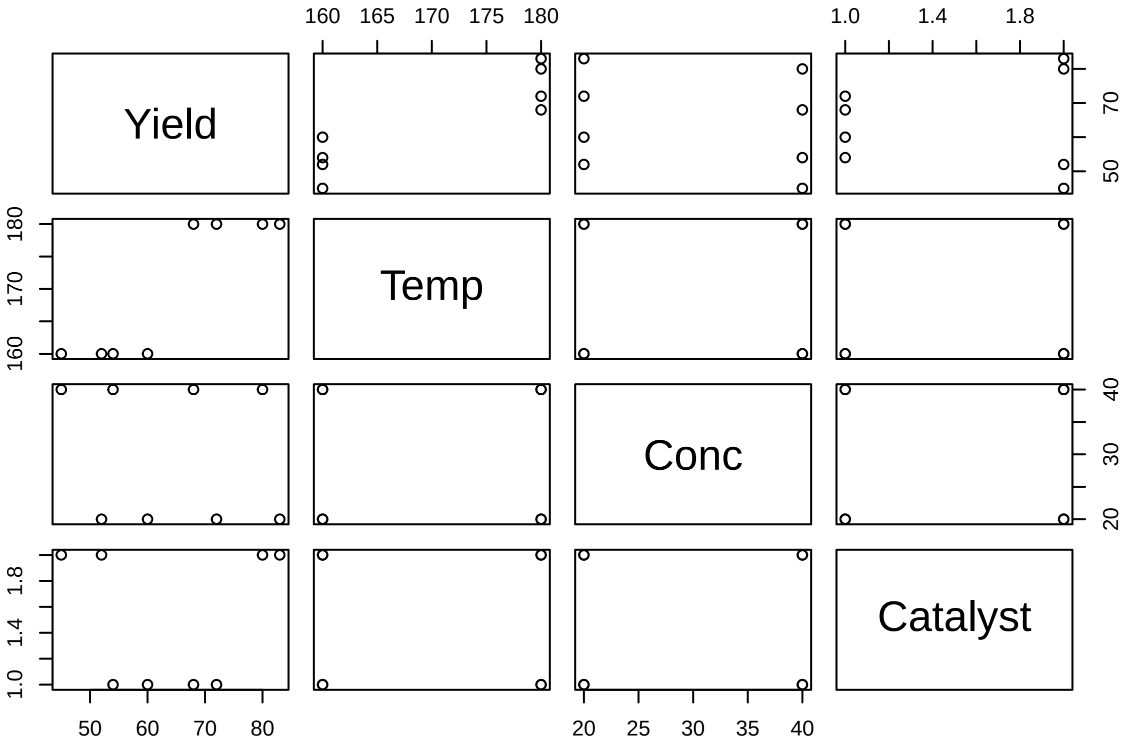
We can make further adjustments to our data such as setting the Catalyst data type as factor to store catalyst values as categorical.

```
yield$Catalyst = factor(yield$Catalyst)
yield
```

```
##   Yield Temp Conc Catalyst
## 1    60  160   20        0
## 2    72  180   20        0
## 3    54  160   40        0
## 4    68  180   40        0
## 5    52  160   20        1
## 6    83  180   20        1
## 7    45  160   40        1
## 8    80  180   40        1
```

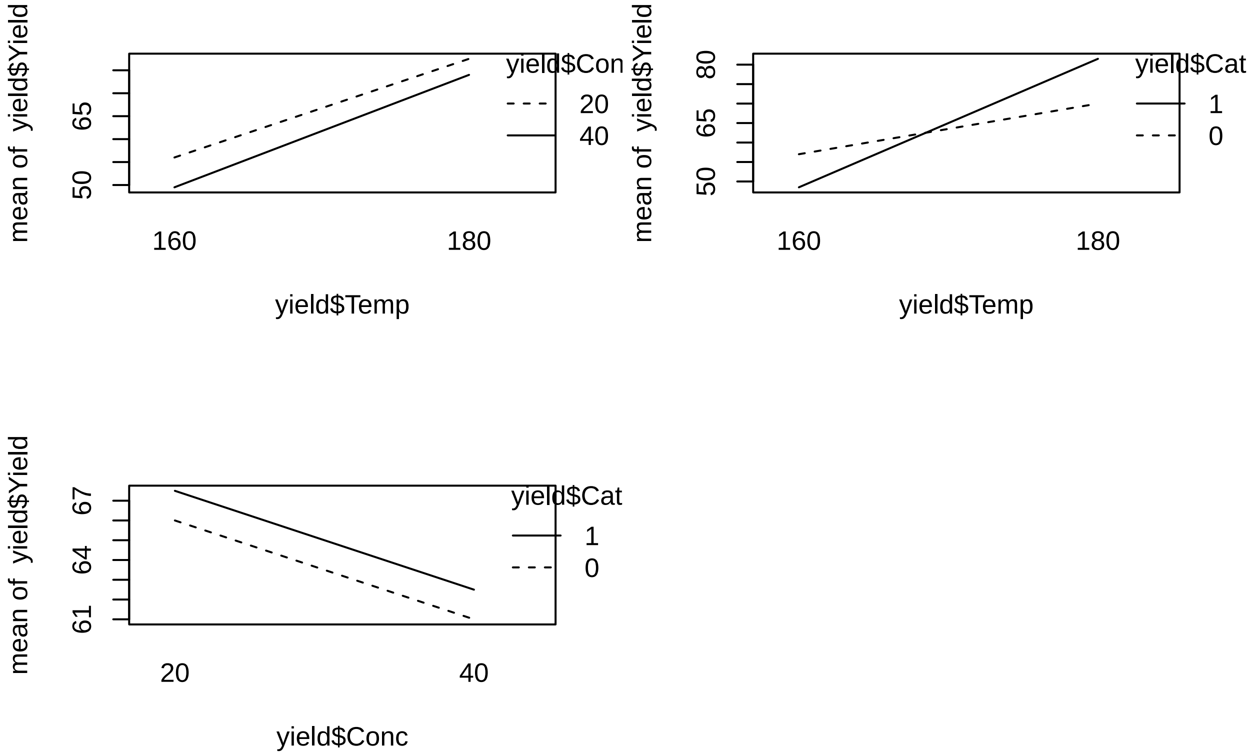
Visualization:

```r
plot(yield)
```



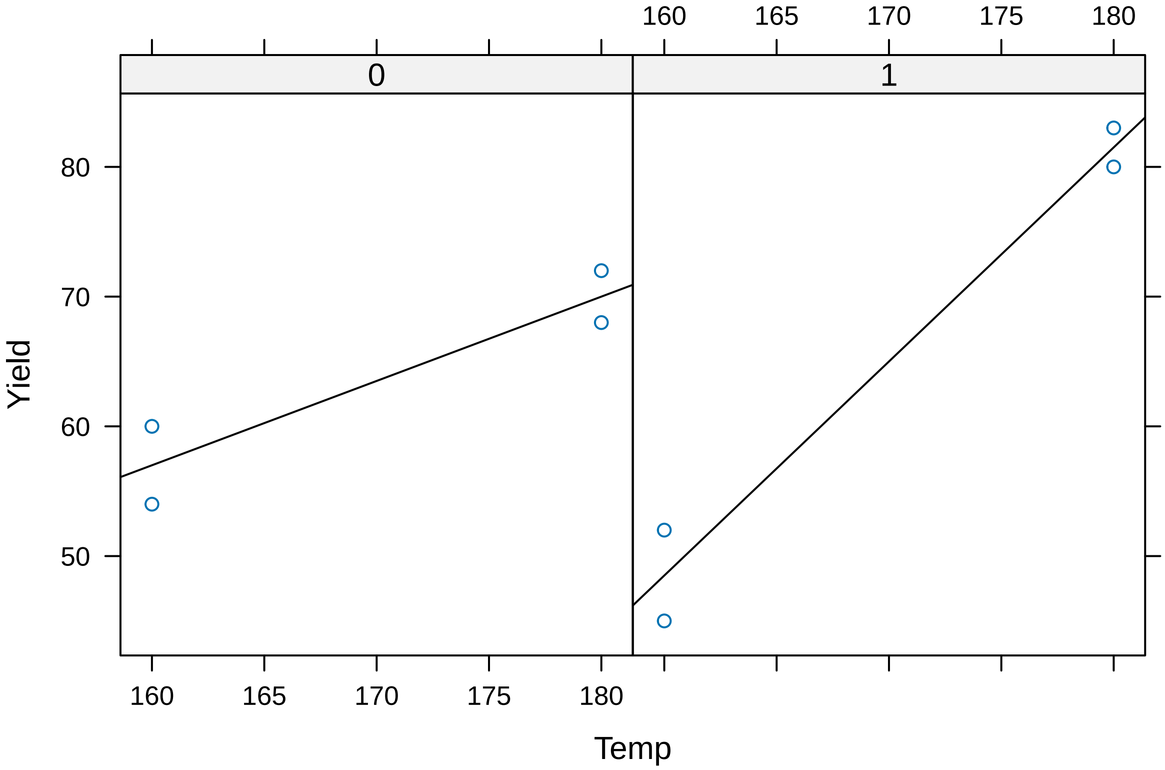
From the plot above, we can eyeball preliminary trends from the factors and response variable such as an increase in yield with a higher temperature.

```r
par(mfrow = c(2, 2))
interaction.plot(yield$Temp, yield$Conc, yield$Yield)
interaction.plot(yield$Temp, yield$Catalyst, yield$Yield)
interaction.plot(yield$Conc, yield$Catalyst, yield$Yield)
par(mfrow = c(1, 1))
```



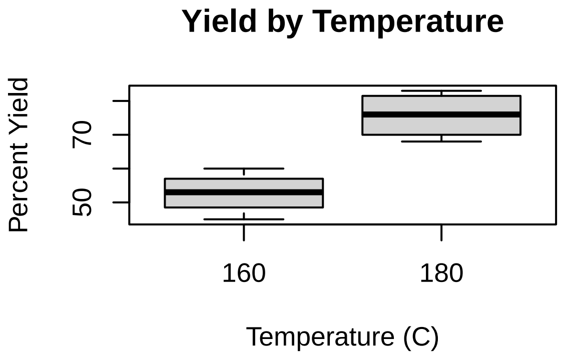
Inferences from the plot above: * Yield increases with higher temperature at both concentrations. * There is an interaction among temperature and catalyst. Temperature raises the yield but effect of yield is different depending on catalyst 0 or 1. * Higher concentration is producing lower yield with no interaction between concentration and catalyst.

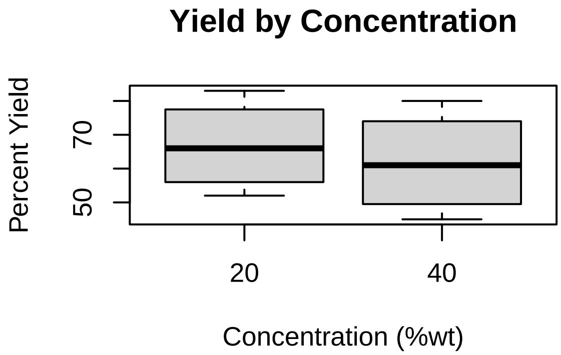
```
xyplot(Yield ~ Temp | Catalyst, data = yield, panel = function(x, y, ...)
       {
        panel.xyplot(x, y, ...)
        panel.lmline(x, y, ...)
       })
```

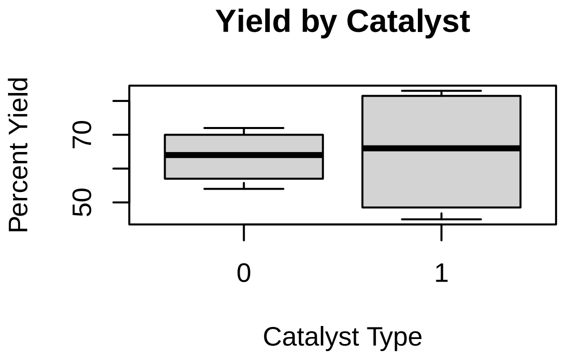


Plot above demonstrates a gradual rate increase of yield with catalyst 0 and a greater rate of yield with catalyst 1 with increase in temperature.

```r
par(mfrow = c(2, 2))

boxplot(Yield ~ Temp, data = yield, main = "Yield by Temperature",
        xlab = "Temperature (C)", ylab = "Percent Yield")

boxplot(Yield ~ Conc, data = yield, main = "Yield by Concentration",
        xlab = "Concentration (%wt)", ylab = "Percent Yield")

boxplot(Yield ~ Catalyst, data = yield, main = "Yield by Catalyst",
        xlab = "Catalyst Type", ylab = "Percent Yield")

par(mfrow = c(1, 1))
```







Analysis:

I will now create a linear model with the main effects observed:

```
model1 <- lm(Yield ~ ., data = yield)
summary(model1)
```

```
##
## Call:
## lm(formula = Yield ~ ., data = yield)
##
## Residuals:
##    1    2    3    4    5    6    7    8
##  5.5 -5.5  4.5 -4.5 -4.0  4.0 -6.0  6.0
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -124.5000    43.8392  -2.840   0.0469 *
## Temp           1.1500     0.2531   4.544   0.0105 *
## Conc          -0.2500     0.2531  -0.988   0.3792
## Catalyst1      1.5000     5.0621   0.296   0.7817
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.159 on 4 degrees of freedom
## Multiple R-squared:  0.8444, Adjusted R-squared:  0.7277
## F-statistic: 7.236 on 3 and 4 DF,  p-value: 0.04297
```

From above we can see that temperature is significant, but concentration and catalyst do not show statistical significance. R-squared value of 0.84 meaning 84% of the variance of the data is explained by the model, but perhaps concentration and catalyst are not explaining any of the variance.

Interactions can be significant, so in this next model it includes another level of depth by considering two-factor interactions:

```
# Inclusive of main effects and two-factor interactions

model2 <- lm(Yield ~ . ** 2, data = yield)
summary(model2)
```

```
##
## Call:
## lm(formula = Yield ~ .^2, data = yield)
##
## Residuals:
##     1     2     3     4     5     6     7     8
## -0.25  0.25  0.25 -0.25  0.25 -0.25 -0.25  0.25
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -1.250e+00  1.414e+01  -0.088   0.9439
## Temp             4.250e-01  8.292e-02   5.126   0.1227
## Conc            -1.525e+00  4.265e-01  -3.576   0.1736
## Catalyst1       -1.685e+02  8.646e+00 -19.489   0.0326 *
## Temp:Conc        7.500e-03  2.500e-03   3.000   0.2048
## Temp:Catalyst1   1.000e+00  5.000e-02  20.000   0.0318 *
## Conc:Catalyst1  -3.925e-17  5.000e-02   0.000   1.0000
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.7071 on 1 degrees of freedom
## Multiple R-squared:  0.9996, Adjusted R-squared:  0.9973
## F-statistic:   439 on 6 and 1 DF,  p-value: 0.03652
```

In model2, main effects have become less significant, however, now catalyst is significant. The interaction between temperature and catalyst are significant. Almost all the variance is explained by the model considering R-squared is now 99% by now including two-factor interactions.

Now to compare both models using ANOVA:

```
anova(model1, model2)
```

```
## Analysis of Variance Table
##
## Model 1: Yield ~ Temp + Conc + Catalyst
## Model 2: Yield ~ (Temp + Conc + Catalyst)^2
##   Res.Df   RSS Df Sum of Sq      F  Pr(>F)
## 1      4 205.0
## 2      1   0.5  3     204.5 136.33 0.06286 .
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

I will now create another model that includes three-factor interactions, and therefore, the full factorial model:

```
model3 <- lm(Yield ~ . ** 3, data = yield)
summary(model3)
```

```
##
## Call:
## lm(formula = Yield ~ .^3, data = yield)
##
## Residuals:
## ALL 8 residuals are 0: no residual degrees of freedom!
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          -14.000        NaN     NaN      NaN
## Temp                   0.500        NaN     NaN      NaN
## Conc                  -1.100        NaN     NaN      NaN
## Catalyst1           -143.000        NaN     NaN      NaN
## Temp:Conc              0.005        NaN     NaN      NaN
## Temp:Catalyst1         0.850        NaN     NaN      NaN
## Conc:Catalyst1        -0.850        NaN     NaN      NaN
## Temp:Conc:Catalyst1    0.005        NaN     NaN      NaN
##
## Residual standard error: NaN on 0 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:    NaN
## F-statistic:   NaN on 7 and 0 DF,  p-value: NA
```

Since we have no replicates, we have no way of assessing the significance of any of the coefficients of this model with all interactions.

A more customized model including some but not all of the interactions would be the following (inclusive of conc, temp, catalyst, as well as the interaction between temp and catalyst):

```
model4 <- lm(Yield ~ Conc + Temp * Catalyst , data = yield)
summary(model4)
```

```
##
## Call:
## lm(formula = Yield ~ Conc + Temp * Catalyst, data = yield)
##
## Residuals:
##    1    2    3    4    5    6    7    8
##  0.5 -0.5 -0.5  0.5  1.0 -1.0 -1.0  1.0
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -39.50000   11.07738  -3.566  0.03766 *
## Conc             -0.25000    0.04564  -5.477  0.01197 *
## Temp              0.65000    0.06455  10.070  0.00209 **
## Catalyst1      -168.50000   15.54563 -10.839  0.00168 **
## Temp:Catalyst1    1.00000    0.09129  10.954  0.00163 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
##
## Residual standard error: 1.291 on 3 degrees of freedom
## Multiple R-squared:  0.9962, Adjusted R-squared:  0.9911
## F-statistic: 196.9 on 4 and 3 DF,  p-value: 0.0005831
```

Every parameter is significant and model explains 99.6% of the variance.

Now to compare model 4 (custom model) and model 2 (all interactions):

```
anova(model4, model2)
```

```
## Analysis of Variance Table
##
## Model 1: Yield ~ Conc + Temp * Catalyst
## Model 2: Yield ~ (Temp + Conc + Catalyst)^2
##   Res.Df RSS Df Sum of Sq   F Pr(>F)
## 1      3 5.0
## 2      1 0.5  2       4.5 4.5 0.3162
```

There is not a statistically significant difference ($p > 0.05$), therefore, I will go with model4 which is a much more simple model of only one interaction term.

Another way to model the response as a function of the factors is with an ANOVA table:

```
yield.aov <- yield
yield.aov$Conc <- factor(yield.aov$Conc)
yield.aov$Temp <- factor(yield.aov$Temp)
```

```
aov1.out <- aov(Yield ~ ., data = yield.aov)
summary(aov1.out)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## Temp         1 1058.0  1058.0  20.644 0.0105 *
## Conc         1   50.0    50.0   0.976 0.3792
## Catalyst     1    4.5     4.5   0.088 0.7817
## Residuals    4  205.0    51.2
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
aov2.out <- aov(Yield ~ . ** 2, data = yield.aov)
summary(aov2.out)
```

```
##               Df Sum Sq Mean Sq F value Pr(>F)
## Temp           1 1058.0  1058.0    2116 0.0138 *
## Conc           1   50.0    50.0     100 0.0635 .
## Catalyst       1    4.5     4.5       9 0.2048
## Temp:Conc      1    4.5     4.5       9 0.2048
## Temp:Catalyst  1  200.0   200.0     400 0.0318 *
## Conc:Catalyst  1    0.0     0.0       0 1.0000
## Residuals      1    0.5     0.5
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```r
model.tables(aov2.out, type = "means", se = TRUE)
```

```
## Tables of means
## Grand mean
##
## 64.25
##
##  Temp
## Temp
##   160   180
## 52.75 75.75
##
##  Conc
## Conc
##    20    40
## 66.75 61.75
##
##  Catalyst
## Catalyst
##    0    1
## 63.5 65.0
##
##  Temp:Conc
##      Conc
## Temp  20   40
##   160 56.0 49.5
##   180 77.5 74.0
##
##  Temp:Catalyst
##      Catalyst
## Temp  0    1
##   160 57.0 48.5
##   180 70.0 81.5
##
##  Conc:Catalyst
##     Catalyst
## Conc 0    1
##   20 66.0 67.5
##   40 61.0 62.5
##
## Standard errors for differences of means
##           Temp   Conc Catalyst Temp:Conc Temp:Catalyst Conc:Catalyst
##         0.5000 0.5000   0.5000    0.7071        0.7071        0.7071
## replic.      4      4        4         2             2             2
```

```r
aov3.out <- aov(Yield ~ . ** 3, data = yield.aov)
summary(aov3.out)
```

```
##                     Df Sum Sq Mean Sq
## Temp                 1 1058.0  1058.0
## Conc                 1   50.0    50.0
## Catalyst             1    4.5     4.5
## Temp:Conc            1    4.5     4.5
## Temp:Catalyst        1  200.0   200.0
## Conc:Catalyst        1    0.0     0.0
## Temp:Conc:Catalyst   1    0.5     0.5
```

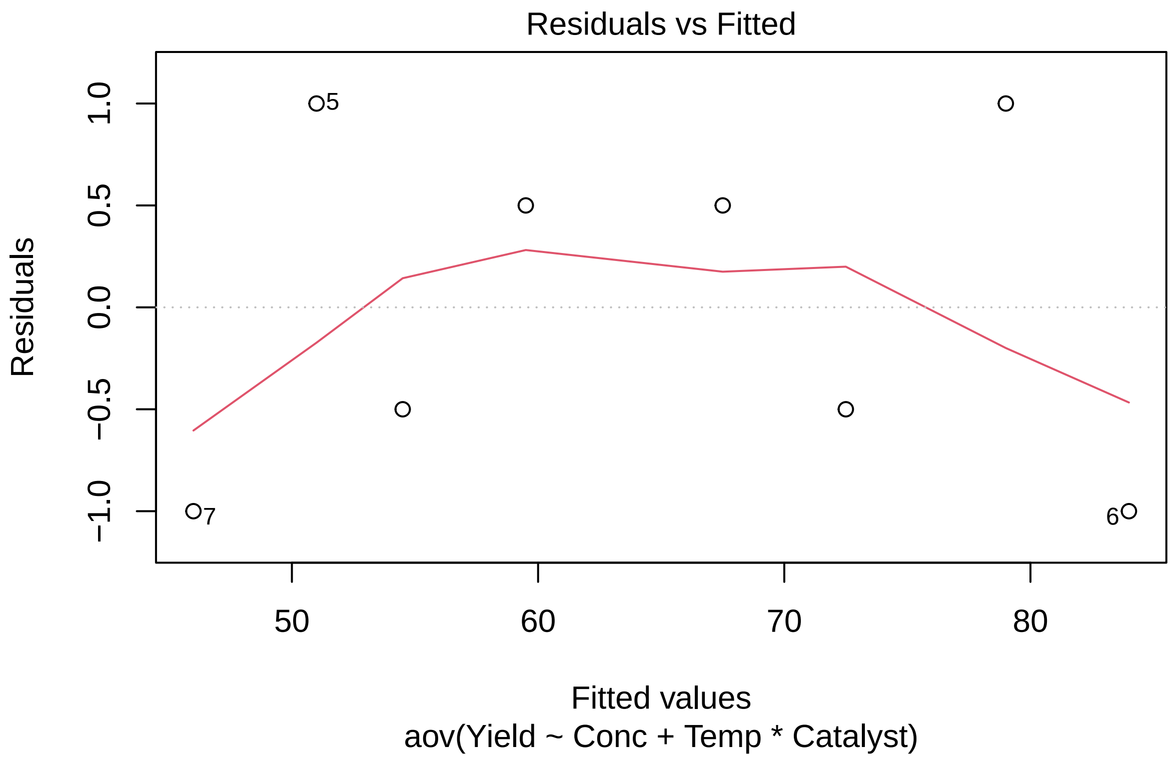
Based on the results, model4 (Yield ~ Conc + Temp * Catalyst) is the better fitting model:

```
aov4.out <- aov(Yield ~ Conc + Temp * Catalyst, data = yield.aov)
summary(aov4.out)
```

```
##                Df Sum Sq Mean Sq F value   Pr(>F)
## Conc            1   50.0    50.0    30.0 0.011967 *
## Temp            1 1058.0  1058.0   634.8 0.000137 ***
## Catalyst        1    4.5     4.5     2.7 0.198892
## Temp:Catalyst   1  200.0   200.0   120.0 0.001629 **
## Residuals       3    5.0     1.7
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

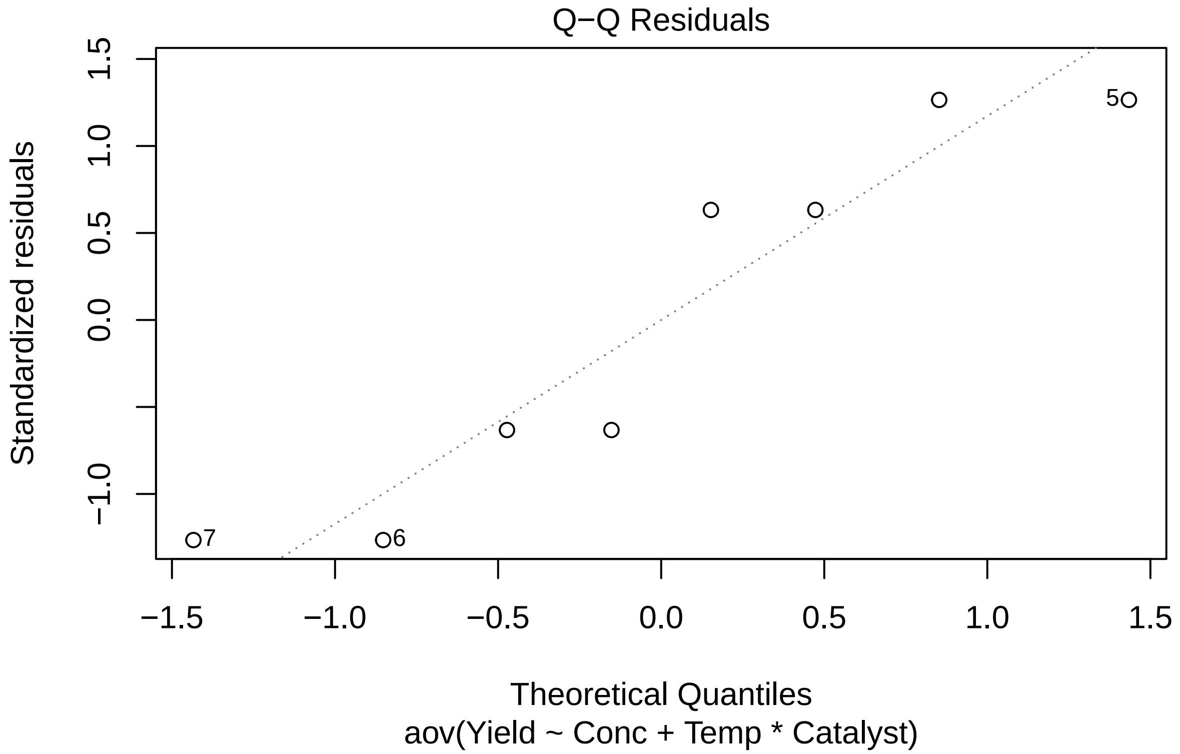
I will now plot the results of the ANOVA by the following:

```
# Residuals vs Fitted Values
plot(aov4.out, 1)
```



The residuals vs fitted plot demonstrates the differences between observed and predicted/fitted response using the chosen model.

```
# QQ Plot
plot(aov4.out, 2)
```



The Quantile-Quantile residuals plot assesses the normality assumption of the residuals in the model. We know the residuals follow a normal distribution given the fact that the data points closely align a diagonal line.