The 2^k Factorial Design

Henry Scharf

MATH/STAT 571B

Module Goals:

Ch. 6 [DAE]: The 2^k Factorial Design

Students will be able to:

1. Explain interaction terms in a linear model from multiple perspectives.
2. Compute (using R) and interpret model parameters for linear models with interaction terms.
3. Explain the connections between multiple equivalent parameterizations/constraints for a linear model.
4. Carry out a conventional analysis for a factorial experiment.

The 2^2 Design

The 2^2 Design

- Factorial designs do not get any simpler than this.
- When a predictor/factor only has two levels, the distinction between categorical and quantitative variables is less meaningful.
- Only able to estimate interaction effect when $n > 1$.
- One degree of freedom for each factor, and one more for interaction.
- There are multiple representations/codings/parameterizations for what amounts to the same model.[1]

[1] Opportunity for checking understanding!

Example: Yield of Chemical Process

Factor A	Factor B	Treatment Combination	Replicate I	Replicate II	Replicate III	Total
−	−	*A* low, *B* low	28	25	27	80
+	−	*A* high, *B* low	36	32	32	100
−	+	*A* low, *B* high	18	19	23	60
+	+	*A* high, *B* high	31	30	29	90

$$y_{ijk} = \mu + \tau_i + \alpha_j + (\tau\alpha)_{ij} + \epsilon_{ijk},\ \epsilon_{ijk} \sim \mathrm{N}(0, \sigma^2)$$

- This model is perhaps of limited practical value in the context of a 2^2 design (e.g., does not appear in DAE 6.2) but it is still appropriate.

Regression Perspective

$$y_i = \beta_0 + \beta_A x_{i,A} + \beta_B x_{i,B} + \beta_{AB} x_{i,A} x_{i,B} + \epsilon_i,\ \epsilon_i \sim \mathrm{N}(0, \sigma^2)$$
$$x_{i,A} = 1_{\{A=+\}},\ x_{i,B} = 1_{\{B=+\}},\ i = 1, \ldots, abn$$

- 0-1 coding (DAE p. 242)
  - Agrees conceptually with $(1), a, b, ab$ notation for low-low, high-low, low-high, and high-high combinations.
  - Choice of coding does not affect ANOVA.
  - Default in R ("low" is arbitrarily defined based on alphabetical order of factors).
  - Can transform to other codings if needed.

Regression Perspective

```
chemical <-
  read.csv("chemical.csv",
     stringsAsFactors = T)
head(chemical)
##   reactant catalyst yield
## 1        -        -    28
## 2        +        -    36
## 3        -        +    18
## 4        +        +    31
## 5        -        -    25
## 6        +        -    32
```

```
model.matrix(~ reactant * catalyst, data = chemical)
##    (Intercept) reactant+ catalyst+ reactant+:catalyst+
## 1            1         0         0                   0
## 2            1         1         0                   0
## 3            1         0         1                   0
## 4            1         1         1                   1
## 5            1         0         0                   0
## 6            1         1         0                   0
## 7            1         0         1                   0
## 8            1         1         1                   1
## 9            1         0         0                   0
## 10           1         1         0                   0
## 11           1         0         1                   0
## 12           1         1         1                   1
```

```
chemical_aov <- aov(yield ~ reactant * catalyst, data = chemical)
summary(lm(chemical_aov))
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)           26.667      1.143  23.338 1.21e-08 ***
## reactant+              6.667      1.616   4.126  0.00332 **
## catalyst+             -6.667      1.616  -4.126  0.00332 **
## reactant+:catalyst+    3.333      2.285   1.459  0.18278
## ---
```

Average Effects

- Tempting to think of $\hat{\beta}_A$ as estimate of effect of Factor A (and associated t-test for $H_0 : \tau_1 = \tau_2 = 0$), **but that is not correct**.
  - $\hat{\beta}_A$ describes the effect of Factor A on the response *when Factor B is low*.
- Effect of each variable depends on the level of the other in presence of interaction.
- One attempt to summarize: average the effects for Factor A across both levels of Factor B.

$$
\begin{aligned}
A &= \frac{1}{2}\left[\underbrace{(\hat{\mu} + \hat{\beta}_A) - \hat{\mu}}_{\hat{y}_{+-} - \hat{y}_{--}} + \underbrace{(\hat{\mu} + \hat{\beta}_A + \hat{\beta}_B + \hat{\beta}_{AB}) - (\hat{\mu} + \hat{\beta}_B)}_{\hat{y}_{++} - \hat{y}_{-+}}\right] \\
&= \hat{\beta}_A + \frac{1}{2}\hat{\beta}_{AB}
\end{aligned}
$$

- Alternative perspective for same formula: difference in average expected response for high and low values of Factor A
  - $\frac{1}{2}(\hat{y}_{+-} + \hat{y}_{++}) - \frac{1}{2}(\hat{y}_{--} + \hat{y}_{-+})$

Average Effects

- Analogous result for Factor B
  - $B = \hat{\beta}_B + \frac{1}{2}\hat{\beta}_{AB}$
- Average interaction effect is average *difference in effect of Factor A at high level of B, compared to effect of Factor A at low level of B*

$$\begin{aligned} AB &= \frac{1}{2}\left[\underbrace{(\hat{\mu} + \hat{\beta}_A + \hat{\beta}_B + \hat{\beta}_{AB}) - (\hat{\mu} + \hat{\beta}_B)}_{\hat{y}_{++} - \hat{y}_{-+}} - \underbrace{(\hat{\mu} + \hat{\beta}_A - \hat{\mu})}_{\hat{y}_{+-} - \hat{y}_{--}}\right] \\ &= \frac{1}{2}\hat{\beta}_{AB} \end{aligned}$$

- Reversing the roles of A and B must give the same definition.

Reconciling with DAE

Coefficient	
μ	26.67
β_A	6.67
β_B	-6.67
β_{AB}	3.33

Average Effect	
A	8.33
B	-5
AB	1.67

(1) What is a situation where the average effect of Factor A would be useful to know?

```
coef(chemical_aov) %*% c(0, 1, 0, 0.5)
##           [,1]
## [1,] 8.333333
coef(chemical_aov) %*% c(0, 0, 1, 0.5)
##      [,1]
## [1,]   -5
coef(chemical_aov) %*% c(0, 0, 0, 0.5)
##           [,1]
## [1,] 1.666667
```

Average Effects: Another Perspective

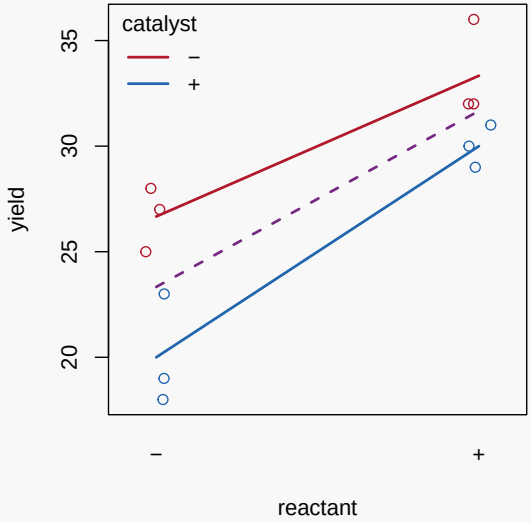
- A and B are what you would estimate for the effect of Factor A if you simply ignored the interaction.

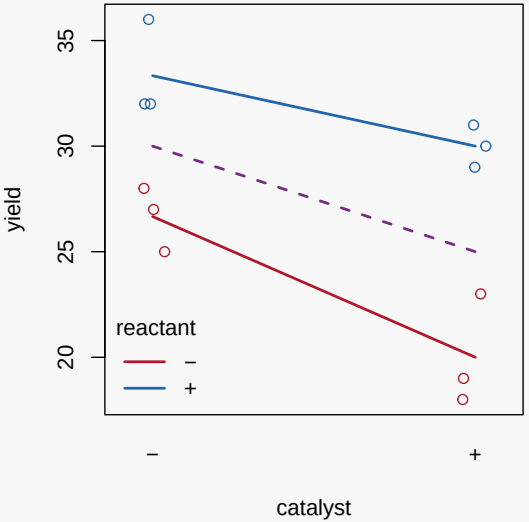
```
coef(aov(yield ~ reactant + catalyst, data = chemical))
## (Intercept)   reactant+   catalyst+
##   25.833333    8.333333   -5.000000
```

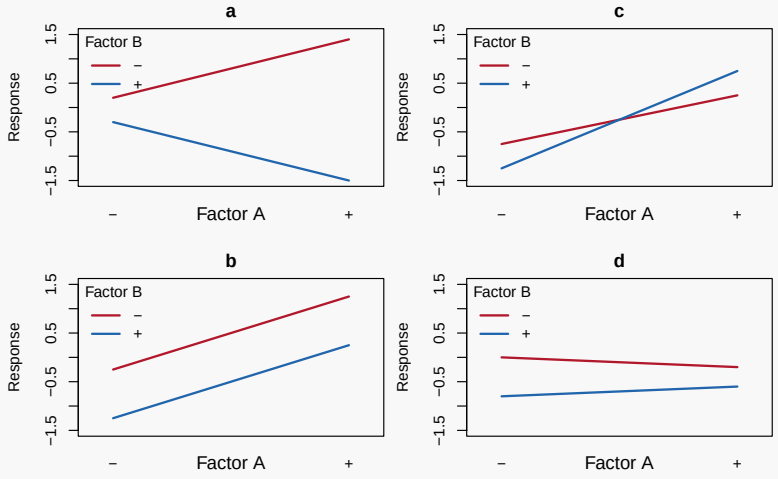
- Because the design is orthogonal, these are also the estimates you would get when simply ignoring the other factor altogether.

```
coef(aov(yield ~ reactant, data = chemical))
## (Intercept)   reactant+
##   23.333333    8.333333
coef(aov(yield ~ catalyst, data = chemical))
## (Intercept)   catalyst+
##          30          -5
```

Graphical Representations of Effects



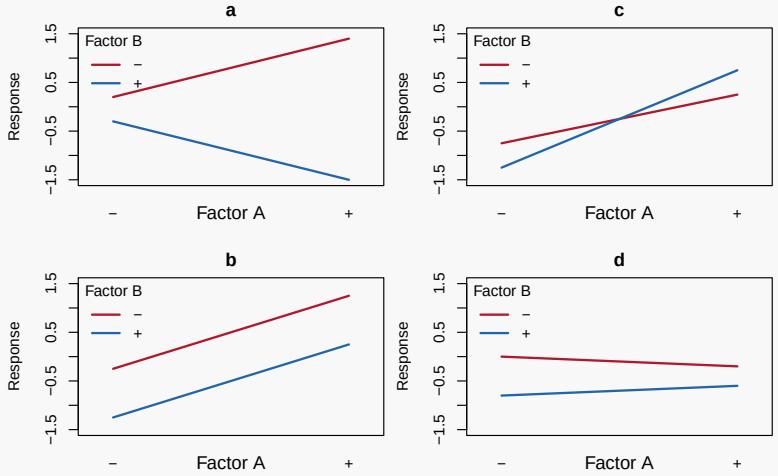




(2) Which figure(s) show(s) $\beta_A > 0$?

(3) Which figure(s) show(s) $A = \beta_A + \frac{1}{2}\beta_{AB} = 0$?

(4) Which figure(s) show(s) $A = \beta_A + \frac{1}{2}\beta_{AB} > 0$?



(5) Which figure(s) show(s) $AB = \frac{1}{2}\beta_{AB} = 0$?

(6) Draw a diagram representing $\beta_A = 1, \beta_B = 0, \beta_{AB} = -2$.

Interactions' Curse + Promise

- Average effects do not avoid the complexity of interpreting interactions, they rephrase it.
- There is no way to describe two arbitrary slopes with a single number.
- Interactions are inherently more difficult to grasp and communicate, but...
- ...on the bright side, interaction effects have the potential to reveal real knowledge obscured by additivity!

ANOVA

$$SS_T = SS_A + SS_B + SS_{AB} + SS_E$$

- Factorial design leads to orthogonality/decomposition of sums of squares.
- While the interaction term is in the model, interpreting p-values associated with first-order effects is tentative at best.

```
anova(chemical_aov)
##                    Df  Sum Sq Mean Sq F value    Pr(>F)
## reactant            1 208.333 208.333 53.1915 8.444e-05 ***
## catalyst            1  75.000  75.000 19.1489  0.002362 **
## reactant:catalyst   1   8.333   8.333  2.1277  0.182776
## Residuals           8  31.333   3.917
```

```
anova(aov(yield ~ reactant + catalyst, data = chemical))
##           Df  Sum Sq Mean Sq F value    Pr(>F)
## reactant   1 208.333 208.333  47.269 7.265e-05 ***
## catalyst   1  75.000  75.000  17.017  0.002578 **
## Residuals  9  39.667   4.407
## ---
```

The 2^3 Design

The 2^3 Design

Run	*A*	*B*	*C*	Labels	*A*	*B*	*C*
1	−	−	−	(1)	0	0	0
2	+	−	−	*a*	1	0	0
3	−	+	−	*b*	0	1	0
4	+	+	−	*ab*	1	1	0
5	−	−	+	*c*	0	0	1
6	+	−	+	*ac*	1	0	1
7	−	+	+	*bc*	0	1	1
8	+	+	+	*abc*	1	1	1

The 2^3 Design

- Many aspects of 2^k factorial design extend naturally to $k > 2$.
- Interpreting interactions for $k \geq 3$ tests the bounds of human cognitive capacity.
- One example: An outcome requires multiple "ingredients" to occur.
  - Butter, sugar, flour interact to make cookies. No one or two alone make cookies.

```
levels <- factor(c("-", "+"))
df <- expand.grid(A = levels,
                  B = levels,
                  C = levels)
df
##   A B C
## 1 - - -
## 2 + - -
## 3 - + -
## 4 + + -
## 5 - - +
## 6 + - +
## 7 - + +
## 8 + + +
```

```
model.matrix(~ A * B * C, data = df)
##   (Intercept) A+ B+ C+ A+:B+ A+:C+ B+:C+ A+:B+:C+
## 1           1  0  0  0     0     0     0       0
## 2           1  1  0  0     0     0     0       0
## 3           1  0  1  0     0     0     0       0
## 4           1  1  1  0     1     0     0       0
## 5           1  0  0  1     0     0     0       0
## 6           1  1  0  1     0     1     0       0
## 7           1  0  1  1     0     0     1       0
## 8           1  1  1  1     1     1     1       1
```

(7) How many interaction terms are there in a 2^4 full factorial experiment?

The General 2^k Design

Analysis Procedure

1. Estimate factor effects
2. Form initial model
   a. If the design is replicated, fit the full model
   b. If there is no replication, form the model using a normal probability plot of the effects (*coming up later today!*)
3. Perform statistical testing
4. Refine model
5. Analyze residuals
6. Interpret results[2]

[2] Be wary of "inference after model selection" problem: Buckland, S. T., Burnham, K. P., & Augustin, N. H. (1997). **Model Selection: An Integral Part of Inference**. Chatfield, C. (1995). **Model Uncertainty, Data Mining and Statistical Inference**. Draper, D. (1995). **Assessment and Propagation of Model Uncertainty**.

Orthogonal Coding

- 0-1 coding:
  - β_A represents the effect of changing Factor A from "low" to "high" while all other factors are at their "low" value.
  - default R behavior
- An alternative coding pursued in the book for factors with two levels is -1, 1 coding:
  - resulting predictors are orthogonal to each other
  - β_A has an alternative interpretation
  - goes under names: **geometric**, **orthogonal**, and **effects** coding

Orthogonal Coding

- An alternative coding pursued in the book for factors with two levels is -1, 1 coding:
  - **resulting predictors are orthogonal to each other**
- Check this for 2^3 design: $\mathbf{x}_A'\mathbf{x}_B = \sum_{i=1}^{abn} x_{iA}x_{iB} = 0$

Run	*A*	*B*	*C*	Labels	*A*	*B*	*C*
1	−	−	−	(1)	0	0	0
2	+	−	−	*a*	1	0	0
3	−	+	−	*b*	0	1	0
4	+	+	−	*ab*	1	1	0
5	−	−	+	*c*	0	0	1
6	+	−	+	*ac*	1	0	1
7	−	+	+	*bc*	0	1	1
8	+	+	+	*abc*	1	1	1

- All predictors are orthogonal to each other, including the intercept
- For any single predictor, $\mathbf{x}_A'\mathbf{x}_A = 2^k n$.
- Thus, $\mathbf{X}'\mathbf{X} = 2^k n\mathbf{I}$

Orthogonal Coding: Main Effects

- An alternative coding pursued in the book for factors with two levels is -1, 1 coding:
  - β_A **has an alternative interpretation**
- Consider a 2^2 experiment: $y_i = \beta_0 + \beta_A x_{iA} + \beta_B x_{iB} + \beta_{AB} x_{iA} x_{iB} + \epsilon_i$
- The average (main) effect, A, of changing Factor A from "low" to "high" averaged over both levels of Factor B is:

$$A = \frac{1}{2}[(\hat{y}_{+-} - \hat{y}_{--}) + (\hat{y}_{++} - \hat{y}_{-+})]$$

$$= \frac{1}{2}\Big\{[(\beta_0+\beta_A \cdot 1 + \beta_B \cdot -1 + \beta_{AB} \cdot 1 \cdot -1) - (\beta_0 + \beta_A \cdot -1 + \beta_B \cdot -1 + \beta_{AB} \cdot -1 \cdot -1)] +$$

$$[(\beta_0+\beta_A \cdot 1 + \beta_B \cdot 1 + \beta_{AB} \cdot 1 \cdot 1) - (\beta_0 + \beta_A \cdot -1 + \beta_B \cdot 1 + \beta_{AB} \cdot -1 \cdot 1)]\Big\}$$

$$= \frac{1}{2}[(2\beta_A - 2\beta_{AB}) + (2\beta_A + 2\beta_{AB})]$$

$$= 2\beta_A$$

- Analogously, $B = 2\beta_B$

Contrast Coding: Main Effects

- The average (main) effect, AB, is:

$$\begin{aligned} AB &= \frac{1}{2}\left[(\hat{y}_{++} - \hat{y}_{-+}) - (\hat{y}_{+-} - \hat{y}_{--})\right] \\ &= \frac{1}{2}\left[(2\beta_A + 2\beta_{AB}) - (2\beta_A - 2\beta_{AB})\right] \\ &= 2\beta_{AB} \end{aligned}$$

- Thus, the -1, 1 coding yields regression coefficients that are twice the average factor effects, hence the term **effects** coding.
- These results extend naturally to 2^k designs (e.g., $ABC = 2\beta_{ABC}$).

(8) Which coding strategy is most intuitive to you?

Contrast Coding: In R

- See `?contrasts` and `?contr.helmert` for more info about coding in R

```
model.matrix(~ A * B * C, data = df,
             contrasts.arg = list(A = "contr.helmert",
                                  B = "contr.helmert",
                                  C = "contr.helmert"))
##   (Intercept) A1 B1 C1 A1:B1 A1:C1 B1:C1 A1:B1:C1
## 1           1 -1 -1 -1     1     1     1       -1
## 2           1  1 -1 -1    -1    -1     1        1
## 3           1 -1  1 -1    -1     1    -1        1
## 4           1  1  1 -1     1    -1    -1       -1
## 5           1 -1 -1  1     1    -1    -1        1
## 6           1  1 -1  1    -1     1    -1       -1
## 7           1 -1  1  1    -1    -1     1       -1
## 8           1  1  1  1     1     1     1        1
```

Contrast Coding: In R

- See `?contrasts` and `?contr.helmert` for more info about coding in R

```
chemical_aov_orth <- aov(yield ~ reactant * catalyst, data = chemical,
                         contrasts = list(reactant = "contr.helmert",
                                          catalyst = "contr.helmert"))
coef(chemical_aov_orth) * 2
##         (Intercept)           reactant1           catalyst1 reactant1:catalyst1
##           55.000000            8.333333           -5.000000            1.666667
```

```
anova(chemical_aov_orth)
## Analysis of Variance Table
##
## Response: yield
##                    Df  Sum Sq Mean Sq F value    Pr(>F)
## reactant            1 208.333 208.333 53.1915 8.444e-05 ***
## catalyst            1  75.000  75.000 19.1489  0.002362 **
## reactant:catalyst   1   8.333   8.333  2.1277  0.182776
## Residuals           8  31.333   3.917
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Average Effect	
A	8.33
B	-5
AB	1.67

A Single Replicate

A Single Replicate

- When $n = 1$, the full model perfectly fits the response: $\hat{y}_i = y_i$.
- Cannot estimate all effects, including all possible interactions, as well as the residual error.
- $MSE = 0$, all F-statistics are infinite, but denominator $df = 0$ so null distribution not defined.
- R will warn you about this.

```
chem2 <- read.csv("chemical2.csv")
chem2_aov <- aov(filtration ~ A * B * C * D, data = chem2)
anova(chem2_aov)
## Warning in anova.lm(chem2_aov): ANOVA F-tests on an essentially perfect fit are
## unreliable
## Analysis of Variance Table
##
## Response: filtration
##           Df  Sum Sq Mean Sq F value Pr(>F)
## A          1 1870.56 1870.56     NaN    NaN
## B          1   39.06   39.06     NaN    NaN
## C          1  390.06  390.06     NaN    NaN
## D          1  855.56  855.56     NaN    NaN
```

A Single Replicate: Brief Detour

- Method for identifying non-zero effects by Cuthbert Daniel[3]
- Cuthbert (1904–1997) worked as a statistician primarily in industry.
- He was **interviewed by Edward Tufte in 1988**
  - Before becoming interested in statistics, he studied Physics and Chemical Engineering, taught High School ("I had been wandering in various wildernesses..."), and then around age 30 his girlfriend at Harvard showed him a book by Fisher.
  - Took a job at a penicillin plant near Princeton, walked into the Statistics Department and asked a couple professors for lessons.
  - Never took a formal statistics course.
  - Tried to tell his bosses at the penicillin plant how to use statistics to predict yield and got fired.
  - Became involved in the Manhattan project.
  - Stated affiliation for the 1959 paper is simply "New York City".

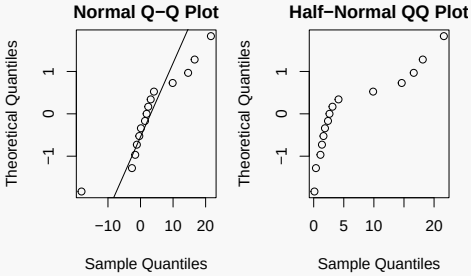
[3]Daniel, C. (1959). **Use of Half-Normal Plots in Interpreting Factorial Two Level Experiments**. *Technometrics*

A Single Replicate

- Zero/negligible effects should be normally distributed with mean 0
- Non-zero effects will deviate from normality

```
chem2_aov_orth <-
  aov(filtration ~ A * B * C * D,
      data = chem2,
      contrasts = list(A = "contr.helmert",
                       B = "contr.helmert",
                       C = "contr.helmert",
                       D = "contr.helmert"))
coefs <- coef(chem2_aov_orth)
sort(coefs[-1])
##       A1:C1    B1:C1:D1    A1:C1:D1       C1:D1       B1:D1       A1:B1
##    -9.0625     -1.3125     -0.8125     -0.5625     -0.1875      0.0625
## A1:B1:C1:D1    A1:B1:C1       B1:C1          B1    A1:B1:D1          C1
##      0.6875      0.9375      1.1875      1.5625      2.0625      4.9375
##          D1       A1:D1          A1
##      7.3125      8.3125     10.8125
```

```
qqnorm(coefs[-1] * 2, datax = T)
qqline(coefs[-1] * 2, datax = T)
qqnorm(abs(coefs[-1] * 2), datax = T)
```



(9) Why the [-1]?

A Single Replicate

- Factor B appears unrelated to response.
- If we are willing to remove it, we can carry out an ANOVA with remaining factors.

```
chem2_aov_ACD <- aov(filtration ~ A * C * D, data = chem2)
anova(chem2_aov_ACD)
## Analysis of Variance Table
##
## Response: filtration
##           Df  Sum Sq Mean Sq F value    Pr(>F)
## A          1 1870.56 1870.56 83.3677 1.667e-05 ***
## C          1  390.06  390.06 17.3844 0.0031244 **
## D          1  855.56  855.56 38.1309 0.0002666 ***
## A:C        1 1314.06 1314.06 58.5655 6.001e-05 ***
## A:D        1 1105.56 1105.56 49.2730 0.0001105 ***
## C:D        1    5.06    5.06  0.2256 0.6474830
## A:C:D      1   10.56   10.56  0.4708 0.5120321
## Residuals  8  179.50   22.44
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

A Single Replicate

- Another alternative to Daniels (1959) is to focus our attention on lower-order effects.

```
chem2_aov_lower <- aov(filtration ~ A * B + A * C  + A * D + B * C + B * D + C * D,
                       data = chem2)
anova(chem2_aov_lower)
## Analysis of Variance Table
##
## Response: filtration
##           Df  Sum Sq Mean Sq F value    Pr(>F)
## A          1 1870.56 1870.56 73.1760 0.0003596 ***
## B          1   39.06   39.06  1.5281 0.2712969
## C          1  390.06  390.06 15.2592 0.0113371 *
## D          1  855.56  855.56 33.4694 0.0021718 **
## A:B        1    0.06    0.06  0.0024 0.9624777
## A:C        1 1314.06 1314.06 51.4059 0.0008208 ***
## A:D        1 1105.56 1105.56 43.2494 0.0012200 **
## B:C        1   22.56   22.56  0.8826 0.3906126
## B:D        1    0.56    0.56  0.0220 0.8878710
## C:D        1    5.06    5.06  0.1980 0.6749089
## Residuals  5  127.81   25.56
```

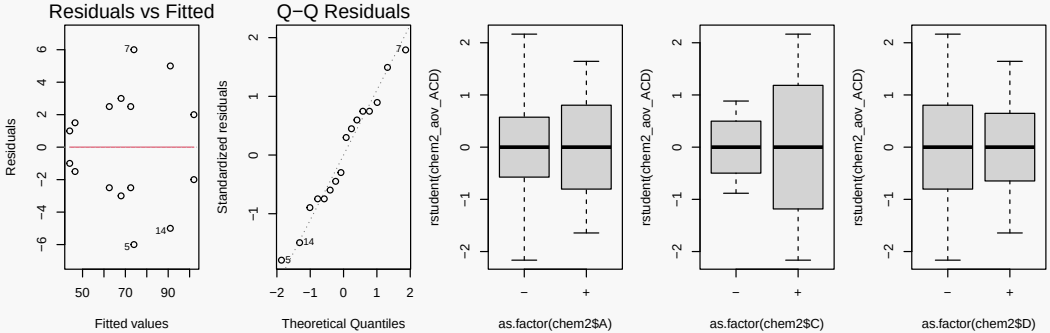
(10) What would an R formula look like if we wanted to consider all 3-way and lower interactions?

A Single Replicate

- Inference is fairly robust to outliers, but with/without comparisons are also a good idea

(11) What potential issue(s) do you see in these diagnostic plots? How severe are they?

```
layout(matrix(1:5, 1, 5)); par(mar = c(4, 4, 1.5, 1))
plot(chem2_aov_ACD, which = 1:2); plot(rstudent(chem2_aov_ACD) ~ as.factor(chem2$A))
plot(rstudent(chem2_aov_ACD) ~ as.factor(chem2$C))
plot(rstudent(chem2_aov_ACD) ~ as.factor(chem2$D))
```



2^k Designs are Optimal

What do we mean by optimal?

- Optimality is with respect to some particular score, which is a *one number* summary of some characteristic(s) of inference.
  - Margin of error (ME) for particular effect
  - Average ME across all effects
  - Maximum ME across all effects
  - ME for predicted value at particular combination of predictors
  - Average ME for predicted value across all combinations of predictors
- Optimality is also (sometimes implicitly) with respect particular constraints such as cost and/or time.
  - Limited number of total runs
  - Some combinations of factors lead to more expensive experiments than others

What do we mean by optimal?

- Consider an experiment to determine the effects of a certain fertilizer and amount of water on the yield of tomatoes.
  1. What is the magnitude of the effect of the fertilizer?
  2. How many pounds of tomatoes can we expect to get for the optimal combination of fertilizer and water?

(12) Will two designs optimized to provide the most precise answers to these respective questions be the same? Why/why not?

A-optimality

- In the absence of clear experimental goals, we seek designs that are as good as possible for a wide range of questions.
- ***A*-optimal**: minimize the average sampling variance across all effects, $\hat{\boldsymbol{\beta}}$.

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon},\ \boldsymbol{\epsilon} \sim \mathrm{N}(0, \sigma^2\mathbf{I})$$

- As designers, we get to control $\mathbf{X}$.
- A-optimality corresponds to minimizing the average of the values along the diagonal of $\mathrm{Var}(\hat{\boldsymbol{\beta}})$

$$\begin{aligned}\hat{\boldsymbol{\beta}} &= (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y} \\ \mathrm{Var}(\hat{\boldsymbol{\beta}}) &= (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\sigma^2\mathbf{I}\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1} \\ &= \sigma^2(\mathbf{X}'\mathbf{X})^{-1}\end{aligned}$$

- Optimal design will not depend on σ^2

A-optimality

- Consider an experiment with three factors, A, B, and C assumed to have an *additive* effect on the response. Suppose the experiment will be run at one of two possible levels of each factor.
- Define $\hat{\beta}$ through the constraint that the effect of one reference level for each factor is 0 (the default in R).

  (13) Does this correspond to -1, 1 or 0-1 coding?
- We seek a design that is A-optimal and are considering a 2^3 design.
- We can afford 16 runs, or enough for $n = 2$ replicates in a factorial design.

```
levels <- factor(c("-", "+"))
df <- expand.grid(A = levels,
                  B = levels,
                  C = levels)
df <- rbind(df, df)
```

A-optimality

```
head(df)
##   A B C
## 1 - - -
## 2 + - -
## 3 - + -
## 4 + + -
## 5 - - +
## 6 + - +
df_mod <- df
df_mod[4, 1] <- df_mod[4, 2] <- "-"
head(df_mod)
##   A B C
## 1 - - -
## 2 + - -
## 3 - + -
## 4 - - -
## 5 - - +
## 6 + - +
```

```
formula <- ~ A + B + C
X <- model.matrix(formula, data = df)
X_mod <- model.matrix(formula, data = df_mod)
XtXinv <- solve(t(X) %*% X)
XtXinv_mod <- solve(t(X_mod) %*% X_mod)
cbind(diag(XtXinv), diag(XtXinv_mod))
##             [,1]      [,2]
## (Intercept) 0.25 0.2000000
## A+          0.25 0.2583333
## B+          0.25 0.2583333
## C+          0.25 0.2583333
mean(diag(XtXinv))
## [1] 0.25
mean(diag(XtXinv_mod))
## [1] 0.24375
```

(14) Is this design *A-optimal* for $\hat{\beta}$? Could your answer change for a different set of effects constraints/coding?

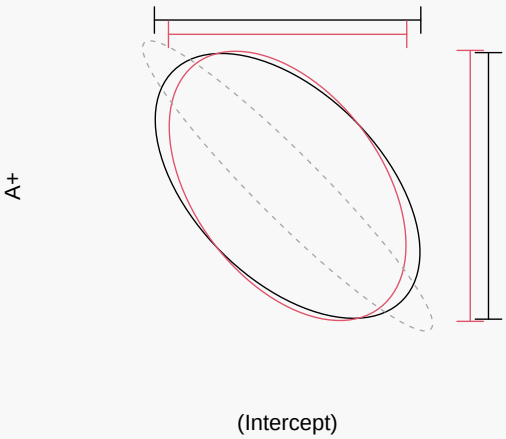
A-optimality

$\mathbf{y} = \mathbf{X}^{\text{eff}}\boldsymbol{\beta}^{\mathbf{eff}} + \boldsymbol{\epsilon}$ where $\mathbf{X}^{\text{eff}}$ represents the design matrix using the "effects" coding.

```
X_eff <- model.matrix(formula, data = df, contrasts.arg = list(A = contr.helmert,
                                                               B = contr.helmert,
                                                               C = contr.helmert))
X_eff_mod <- model.matrix(formula, data = df_mod, contrasts.arg = list(A = contr.helmert,
                                                                       B = contr.helmert,
                                                                       C = contr.helmert))

XtXinv_eff <- solve(t(X_eff) %*% X_eff)
XtXinv_eff_mod <- solve(t(X_eff_mod) %*% X_eff_mod)
cbind(diag(XtXinv_eff), diag(XtXinv_eff_mod))
##                [,1]       [,2]
## (Intercept) 0.0625 0.06458333
## A1          0.0625 0.06458333
## B1          0.0625 0.06458333
## C1          0.0625 0.06458333
mean(diag(XtXinv_eff))
## [1] 0.0625
mean(diag(XtXinv_eff_mod))
## [1] 0.06458333
```

A-optimality



(15) Which confidence ellipse for the intercept and effect of the high level of A represents the greatest amount of precision? Why?

D-optimality

- ***D*-optimal** designs minimize the volume of the joint confidence region for $\hat{\beta}$.
- The volume of the joint confidence region is proportional to $|(\mathbf{X}'\mathbf{X})^{-1}|^{1/2} = |\mathbf{X}'\mathbf{X}|^{-1/2}$.

```
det(XtXinv)
## [1] 0.0009765625
det(XtXinv_mod)
## [1] 0.001041667
```

- "The 2^k design is a D-optimal design for fitting the first-order model or the first-order model with interaction." –DAE p.282
- Other **optimality criteria**