

Displays for Statistics 5303

Lecture 40

December 11, 2002

Christopher Bingham, Instructor

612-625-7023 (St. Paul)

612-625-1024 (Minneapolis)

Class Web Page

<http://www.stat.umn.edu/~kb/classes/5303>

© 2002 by Christopher Bingham

Fractional Factorial Designs

Fractional factorial designs are fairly closely related to **confounded** designs.

They are needed because for **complete factorial designs**

1. The number of treatments grows exponentially with the number of factors. $2^7 = 128, 2^8 = 256, 2^9 = 512, 2^{10} = 1024, \dots, 2^{20} = 1,048,576, \dots$
2. You get information about *all* the interactions (except those confounded with blocks) even when you think some of them are negligible.

Part of the motivation for confounding is willingness to sacrifice information on negligible high order interactions in order to have **smaller blocks**.

Fractional replication has a similar property. It sacrifices information on what are thought to be *negligible* interactions to allow using **fewer treatments**.

2

There is at least one big difference. In confounded complete factorial designs, only a few effects are confounded. You can estimate and test all the other effects from differences between blocks.

Moreover, when blocks are random, you can even estimate the confounded effects from contrasts between blocks, although there standard errors are much larger.

In a fractional replicate

- *Every* effect is confounded with at least one other effect (*not* block)
- There is *no* way to disentangle them other than by including more treatment combinations.

Difference in vocabulary

You actually *don't* say one effect is "confounded" with another. You say it is **aliased**.

Aliased effects are really the *same* effects under different names.

The basic construction for a fractional replicate is to choose as factor combinations the treatments in one block of a confounded design.

2^{3-1} with ABC confounded

I	II
(1)	a
ab	b
ac	c
bc	abc

In block I the treatment means are

$$\begin{aligned} \mu_{(1)} &= \mu - \alpha_2 - \beta_2 - \gamma_2 + \alpha\beta_{22} + \alpha\gamma_{22} + \beta\gamma_{22} - \alpha\beta\gamma_{222} \\ \mu_{ab} &= \mu + \alpha_2 + \beta_2 - \gamma_2 + \alpha\beta_{22} - \alpha\gamma_{22} - \beta\gamma_{22} - \alpha\beta\gamma_{222} \\ \mu_{ac} &= \mu + \alpha_2 - \beta_2 + \gamma_2 - \alpha\beta_{22} + \alpha\gamma_{22} - \beta\gamma_{22} - \alpha\beta\gamma_{222} \\ \mu_{bc} &= \mu - \alpha_2 + \beta_2 + \gamma_2 - \alpha\beta_{22} - \alpha\gamma_{22} + \beta\gamma_{22} - \alpha\beta\gamma_{222} \end{aligned}$$

The coefficients of $\mu, \alpha_2, \beta_2, \dots, \alpha\beta\gamma_{222}$ are from rows (1), ab, ac, and bc of the following table.

	I	A	B	AB	C	AC	BC	ABC
(1)	+1	-1	-1	+1	-1	+1	+1	-1
a	+1	+1	-1	-1	-1	-1	+1	+1
b	+1	-1	+1	-1	-1	+1	-1	+1
ab	+1	+1	+1	+1	-1	-1	-1	-1
c	+1	-1	-1	+1	+1	-1	-1	+1
ac	+1	+1	-1	-1	+1	+1	-1	-1
ad	+1	-1	+1	-1	+1	-1	+1	-1
abc	+1	+1	+1	+1	+1	+1	+1	+1

In these 4 rows,

- the coefficients in column I are the negative of those in column ABC
- the coefficients in column A are the negative of those in column BC
- the coefficients in column B are the negative of those in column AC
- the coefficients in column C are the negative of those in column AB

Similarly

$$\hat{\mu} = (y_{ab} + y_{ac} + y_{(1)} + y_{bc})/4 = \mu - \alpha\beta\gamma_{222} + (\epsilon_{ab} + \epsilon_{ac} + \epsilon_{(1)} + \epsilon_{bc})/4$$

So μ is aliased with $-\alpha\beta\gamma_{222}$.

In the notation of fractional factorials, μ is associated with "effect" I, so we also say I is aliased with -ABC.

-ABC is the *defining contrast*.

This is also the meaning of the *defining relation* **I = -ABC**.

You can get the entire alias structure from I = -ABC by using generalized multiplication in which I acts like 1.

- A×I = A = -A²BC = -BC
- B×I = B = -AB²C = -AC
- C×I = C = -ABC² = -AB
- AB×I = AB = -A²B²C = -C
- Similarly AC = -B and BC = -A
- ABC×I = ABC = -A²B²C² = -I

The natural estimate of α_2 is

$$\hat{\alpha}_2 = (y_{ab} + y_{ac} - y_{(1)} - y_{bc})/4 = \alpha_2 - \beta\gamma_{22} + (\epsilon_{ab} + \epsilon_{ac} - \epsilon_{(1)} - \epsilon_{bc})/4$$

This has expectation

$$E(\hat{\alpha}_2) = \alpha_2 - \beta\gamma_{22}$$

This means A is aliased with -BC.

If $\beta\gamma_{22} = 0$, this is not a problem, since then $E(\hat{\alpha}_2) = \alpha_2$ and $\hat{\alpha}_2$ is *unbiased*.

Or if $\beta\gamma_{22}$ is small relative to α_2 , it may not be a real problem.

However, if $\beta\gamma_{22}$ is not small relative to α_2 , you have series bias and the interpretation of a large $\hat{\alpha}_2$ is suspect. $\hat{\alpha}_2$ may be large because $\beta\gamma_{22}$ is large, not because α_2 is large. Also, when both α_2 and $\beta\gamma_{22}$ are large $\alpha_2 - \beta\gamma_{22}$ may be small and not show up as significant.

Generally, in a 2^{k-p} design (k factors with 2 levels, in a $1/2^p$ replicate), the design is defined by p defining relations.

For example, one choice for a 2^{5-2} design (5 factors in quarter replicate of size 16), are the 2 defining relations

$$I = ABC \text{ and } I = ADE$$

In place of one of these you could use I = BCDE since $ABC \times ADE = A^2BCDE = BCDE$.

In block II the means are

$$\mu_a = \mu + \alpha_2 - \beta_2 - \gamma_2 - \alpha\beta_{22} - \alpha\gamma_{22} + \beta\gamma_{22} + \alpha\beta\gamma_{222}$$

$$\mu_b = \mu - \alpha_2 + \beta_2 - \gamma_2 - \alpha\beta_{22} + \alpha\gamma_{22} - \beta\gamma_{22} + \alpha\beta\gamma_{222}$$

$$\mu_c = \mu - \alpha_2 - \beta_2 + \gamma_2 + \alpha\beta_{22} - \alpha\gamma_{22} - \beta\gamma_{22} + \alpha\beta\gamma_{222}$$

$$\mu_{abc} = \mu + \alpha_2 + \beta_2 + \gamma_2 + \alpha\beta_{22} + \alpha\gamma_{22} + \beta\gamma_{22} + \alpha\beta\gamma_{222}$$

The coefficients are from rows a, b, c, and abc of the table of ± 1 's. In these rows the coefficients of μ and ABC are the same, as are the coefficients of A and BC, B and AC, and C and AB.

The natural estimate of α_2 is now

$$\hat{\alpha}_2 = (y_a + y_{abc} - y_b - y_c) / 4$$

$$= \alpha_2 + \beta\gamma_{22} + (\epsilon_a + \epsilon_{abc} - \epsilon_b - \epsilon_c) / 4$$

with expectation $E(\hat{\alpha}_2) = \alpha_2 + \beta\gamma_{22}$.

Now A is aliased with +BC and the defining relation is $I = ABC$

MacAnova has several commands for working with fractional replicate designs.

For two-series fractions, you specify generators (aliases) using a matrix.

The generator or *basis* matrix has

- a row for each generator
- a column for each factor
- elements which are ± 1 or 0. A non-zero value indicates the factor associated with the column is in the generator; the product of the nonzero elements in a generator determines the sign of the generator.

```
Cmd> gen <- matrix(vector(1,1,1,0,0, 1,0,0,1,1), 5)'# 2 by 5 matrix
Cmd> gen # ABC and ADE are generators
(1,1)      1      1      1      0      0
(2,1)      1      0      0      1      1
```

This specifies defining relations

$$I = ABC \text{ and } I = ADE$$

for a 2^{5-2} design ($1/2^2 = 1/4$ replicate with $2^{5-2} = 8$ treatments).

The alias structure is

$$I = ABC, A = BC, B = AC, C = BC.$$

This is essentially the same as before, except for differences in signs.

Because of the differences in signs, if you should decide to do a second 2^{3-1} experiment, if you use the other block you can disentangle all effects except ABC which would be confounded between the blocks.

`aliases2()` finds the aliases (of I) for a design generator.

```
Cmd> aliases2(gen) # gen contains generators ABC and ADE
(1) "I"
(2) "ABC"
(3) "ADE"
(4) "BCDE"
```

BCDE is the generalized product of ABC and ADE.

You use keyword `effect` to specify that you want aliases of another effect. This gets the aliases of AB.

```
Cmd> aliases2(gen, effect=vector(1,1,0,0,0))
(1) "AB"
(2) "C"
(3) "BDE"
(4) "ACDE"
```

The default value of `effect` is the vector of all 0's, that is "effect" I.

```
Cmd> aliases2(gen, effect=vector(0,0,0,0,0))
(1) "I"
(2) "ABC"
(3) "ADE"
(4) "BCDE"
```

`allaliases2()` prints the *complete* alias structure for the design.

```
Cmd> allaliases2(gen) # each effect is aliased with 3 others
(1) "I = ABC = ADE = BCDE"
(2) "A = BC = DE = ABCDE"
(3) "B = AC = ABDE = CDE"
(4) "AB = C = BDE = ACDE"
(5) "D = ABCD = AE = BCE"
(6) "AD = BCD = E = ABCE"
(7) "BD = ACD = ABE = CE"
(8) "ABD = CD = BE = ACE"
```

Every effect is aliased with 3 others.

Now try the generator for a 2^{8-4} fractional replicate design (1/16 replicate).

```
Cmd> gen <- matrix(vector(0,1,1,1,1,0,0,0, 1,0,1,1,0,1,0,0,\
1,1,1,0,0,0,1,0, 1,1,0,1,0,0,0,1),8)' # 4 by 8 matrix

Cmd> print(gen, format="2.0F")
gen:
(1,1) 0 1 1 1 1 0 0 0      I = BCDE
(2,1) 1 0 1 1 0 1 0 0      I = ACDF
(3,1) 1 1 1 0 0 0 1 0      I = ABCG
(4,1) 1 1 0 1 0 0 0 1      I = ABDH
```

```
Cmd> allaliases2(gen) # lots and lots of output
(1) "I = BCDE = ACDF = ABFE = ABCG = ADEG = BDFG = CEFG = ABDH = ACEH = BCFH = DEFH = CDGH = BEGH = AFGH = ABCDEFGH"
(2) "A = BCDE = CDF = BEF = BCG = DEG = ABDFG = ACEFG = BDH = CEH = ABCFH = ADEFH = ACDGH = ABEGH = FGH = BCDEFGH"
(3) "B = CDE = ABCDF = AEF = ACG = ABDEG = DFG = BCEFG = ADH = ABCEH = CFH = BDEFH = BCDGH = EGH = ABFGH = ACDEFGH"
(4) "AB = ACDE = BCDF = EF = CG = BDEG = ADFG = ABCEFG = DH = BCEH = ACFH = ABDEFH = ABCDGH = AEGH = BFGH = CDEFGH"
(5) "C = BDE = ADF = ABCEF = ABG = ACDEG = BCDFG = EFG = ABCDH = AEH = BFH = CDEFH = DGH = BCEGH = ACFGH = ABDEFH"
(6) "AC = ABDE = DF = BCEF = BG = CDEG = ABCDFG = AEFH = BCDH = EH = ABFH = ACDEFH = ADGH = ABCEGH = CFGH = BDEFH"
(7) "BC = DE = ABDF = ACEF = AG = ABCDEG = CDFG = BEFG = ACDH = ABEH = FH = BCDEFH = BDGH = CEH = ABCFGH = ADEFGH"
(8) "ABC = ADE = BDF = CEF = G = BCDEG = ACDFG = ABCEFG = CDH = BEH = AFH = ABCDEFH = ABDGH = ACEGH = BCFGH = DEFGH"
(9) "D = BCE = ACF = ABDEF = ABCDG = AEG = BFG = CDEFH = ABH = ACDEH = BCDFH = EFH = CGH = BDEGH = ADFGH = ABCEFGH"
(10) "AD = ABCE = CF = BDEF = BCDG = EG = ABFG = ACDEFH = BH = CDEH = ABCDFH = AEFH = ACEH = ABDEGH = DFGH = BCEFGH"
(11) "BD = CE = ABCF = ADEF = ACDG = ABEG = FG = BCDEFH = AH = ABCDEH = CDFH = BEFH = BCGH = DEGH = ABDFGH = ACEFGH"
(12) "ABD = ACE = BCF = DEF = CDG = BEG = AFG = ABCDEFH = H = BCDEH = ACDFH = ABFEH = ABCGH = ADEGH = BDFGH = CEFH"
(13) "CD = BE = AF = ABCDEF = ABDG = ACEG = BCFG = DEFG = ABCH = ADEH = BDFH = CEFH = GH = BCDEGH = ACDFGH = ABFEFGH"
(14) "ACD = ABE = F = BCDEF = BDG = CEG = ABCFG = ADEFG = BCH = DEH = ABDFH = ACEFH = AGH = ABCDEGH = CDFGH = BEFGH"
(15) "BCD = E = ABF = ACDEF = ADG = ABCEG = CFG = BDEFG = ACH = ABDEH = DFH = BCEFH = BGH = CDEGH = ABCDFGH = AEFH"
(16) "ABCD = AE = BF = CDEF = DG = BCEG = ACFG = ABDEFG = CH = BDEH = ADFH = ABCEFH = ABGH = ACDEGH = BCDFGH = EFGH"
```

```
Cmd> aliases2(gen) # the aliases of I
(1) "I"
(2) "BCDE"
(3) "ACDF"
(4) "ABFE"
(5) "ABCG"
(6) "ADEG"
(7) "BDFG"
(8) "CEFG"
(9) "ABDH"
(10) "ACEH"
(11) "BCFH"
(12) "DEFH"
(13) "CDGH"
(14) "BEGH"
(15) "AFGH"
(16) "ABCDEFGH"
```

Once you have decided on the alias structure, you can use `ffdesign2()` to find which combinations of factor levels are actually used.

```
Cmd> ffdesign2(gen) # the principal fraction
(1) "(1)"
(2) "afgh"
(3) "begh"
(4) "abef"
(5) "cefg"
(6) "aceh"
(7) "bcfh"
(8) "abcg"
(9) "defh"
(10) "adeg"
(11) "bdfg"
(12) "abdh"
(13) "cdgh"
(14) "acdf"
(15) "bcde"
(16) "abcdefgh"
```

There are $2^4 = 16$ treatment combinations, or $1/2^4 = 1/16$ of the $2^8 = 256$ possible choices.

Each choice corresponds to a choice of possible signs in the defining relations.

This is just one possible fraction.
 Change the generators so that the last one is -ABDH instead of ABDH.

```

Cmd> gen2 <- gen; gen2[4,8]<- -1; print(gen2,format:"2.0f")
gen2:
(1,1) 0 1 1 1 1 0 0 0      I = BCDE
(2,1) 1 0 1 1 0 1 0 0      I = ACDF
(3,1) 1 1 1 0 0 0 1 0      I = ABCG
(4,1) 1 1 0 1 0 0 0 -1     I = -ABDH

Cmd> aliases2(gen2)
(1) "I"
(2) "BCDE"
(3) "ACDF"
(4) "ABEF"
(5) "ABCG"
(6) "ADEG"
(7) "BDFG"
(8) "CEFG"
(9) "-ABDH"
(10) "-ACEH"
(11) "-BCFH"
(12) "-DEFH"
(13) "-CDGH"
(14) "-BEGH"
(15) "-AFGH"
(16) "-ABCDEFGH"

Cmd> allaliases2(gen2)
(1) "I = BCDE = ACDF = ABEF = ABCG = ADEG = BDFG = CEFG = -ABDH
= -ACEH = -BCFH = -DEFH = -CDGH = -BEGH = -AFGH = -ABCDEFGH"
(2) "A = ABCDE = CDF = BEF = BCG = DEG = ABDFG = ACEFG = -BDH =
-CEH = -ABCFH = -ADEFH = -ACDGH = -ABEGH = -FGH = -BCDEFH" . . .
. . . . .
(15) "BCD = E = ABF = ACDEF = ADG = ABCEG = CFG = BDEFG = -ACH =
-ABDEH = -DFH = -BCEFH = -BGH = -CDEGH = -ABCDFGH = -AEFGH"
(16) "ABCD = AE = BF = CDEF = DG = BCEG = ACFG = ABDEFG = -CH =
-BDEH = -ADFH = -ABCFH = -ABGH = -ACDEGH = -BCDFGH = -EFGH"
    
```

The same sets of letters appear as before but there are some negative signs.
 The treatments in the design change completely.

```

Cmd> ffdesign2(gen2)
(1) "h"
(2) "afg"
(3) "beg"
(4) "abefh"
(5) "cefgh"
(6) "ace"
(7) "bcf"
(8) "abcgh"
(9) "def"
(10) "adegh"
(11) "bdfgh"
(12) "abd"
(13) "cdg"
(14) "acdfh"
(15) "bcdeh"
(16) "abcdefg"
    
```

Wherever there was an h before, it is now gone, and wherever h was missing before, it is now present.