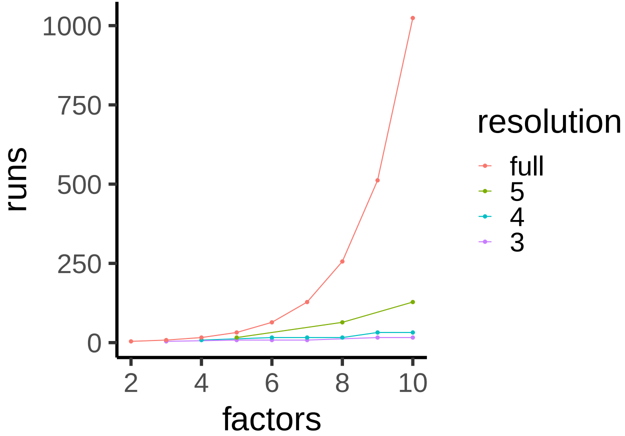
Alternative Fractional Factorial Designs

BIOE 498/598 PJ

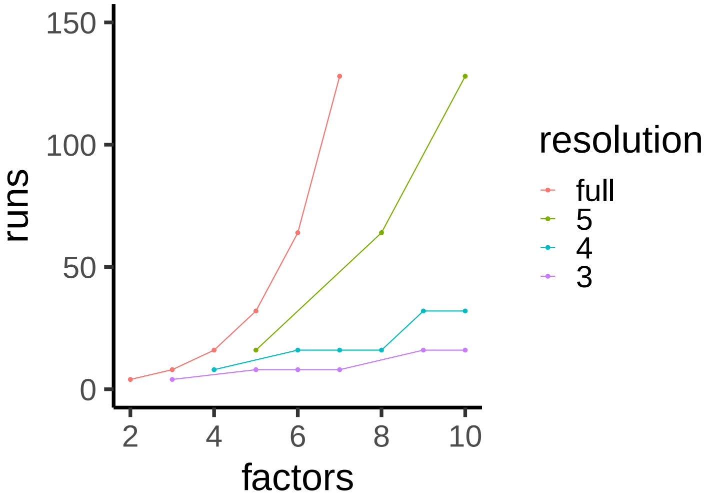
Spring 2022

How low can we go?

The efficiency of fractional factorial designs offsets the exponential increase in runs for factorial designs.



How low can we go? (zoomed in)



number of runs

number of factors	8	16	32	64	128	256	512	1024	2048	4096
						only the MA design				
3	full									
4	IV	full								
5	III	V	full							
6	III	IV	VI	full						
7	III	IV	IV	VII	full					
8		IV	IV	V	VIII	full				
9		III	IV	IV	VI	IX	full			
10		III	IV	IV	V	VI	X	full		
11		III	IV	IV	V	VI	VII	XI	full	
12		III	IV	IV	IV	VI	VI	VIII	XII	full
13		III	IV	IV	IV	V	VI	VII	VIII	XIII
14		III	IV	IV	IV	V	VI	VII	VIII	IX
15		III	IV	IV	IV	V	VI	VII	VIII	VIII
16			IV	IV	IV	V	VI	VI	VIII	VIII
17			III	IV	IV	V	VI	VI	VII	VIII
18			III	IV	IV	IV	VI	VI	VII	VIII
19			III	IV	IV	IV	V	VI	VII	VIII
20			III	IV	IV	IV	V	VI	VII	VIII
21			III	IV	IV	IV	V	VI	VII	VIII
22			III	IV	IV	IV	V	VI	VII	VIII
23			III	IV	IV	IV	V	VI	VII	VIII
24			III	IV	IV	IV	IV	VI	VI	VIII
Resolution III up to			31	63	127			factors.		
Resolution IV up to				32	64	80	160	factors.		
Resolution V up to number of factors:								33	47	65
Resolution VI up to number of factors:								24	34	48
First design is MA up to number of factors:			31	63	127	36	29	28	32	26

Gromping, 2014
J. Stat. Software

Foldover Designs

Imagine a 2^{6-3}_{III} design with

$$D = AB, \quad E = AC, \quad F = BC$$

$$I = ABD = ACE = BCF = DEF$$
$$= BCDE = ACDF = ABEF$$

After analysis, we find that both B and D are significant.

Since $D = AB$, the significance of D might be due to B and AB.

We can *augment* the design by doubling the runs *with D flipped*. This clears D and its interactions.

Run	A	B	C	D	E	F
1	$-$	$-$	$-$	$+$	$+$	$+$
2	$+$	$-$	$-$	$-$	$-$	$+$
3	$-$	$+$	$-$	$-$	$+$	$-$
4	$+$	$+$	$-$	$+$	$-$	$-$
5	$-$	$-$	$+$	$+$	$-$	$-$
6	$+$	$-$	$+$	$-$	$+$	$-$
7	$-$	$+$	$+$	$-$	$-$	$+$
8	$+$	$+$	$+$	$+$	$+$	$+$

Foldover Designs

Imagine a 2^{6-3}_{III} design with

$$D = AB, \quad E = AC, \quad F = BC$$

$$I = ABD = ACE = BCF = DEF$$
$$= BCDE = ACDF = ABEF$$

After analysis, we find that both B and D are significant.

Since $D = AB$, the significance of D might be due to B and AB.

We can *augment* the design by doubling the runs *with D flipped*. This clears D and its interactions.

Run	A	B	C	D	E	F
1	−	−	−	+	+	+
2	+	−	−	−	−	+
3	−	+	−	−	+	−
4	+	+	−	+	−	−
5	−	−	+	+	−	−
6	+	−	+	−	+	−
7	−	+	+	−	−	+
8	+	+	+	+	+	+
9	−	−	−	−	+	+
10	+	−	−	+	−	+
11	−	+	−	+	+	−
12	+	+	−	−	−	−
13	−	−	+	−	−	−
14	+	−	+	+	+	−
15	−	+	+	+	−	+
16	+	+	+	−	+	+

Mirror image designs

If we combine a Resolution III design with its mirror image (all factors flipped), we have a Resolution IV design with all main effects clear.

If we add a blocking factor we can perform the experimental batches sequentially.

As with foldover designs, mirror image designs are only necessary if more than one main effect is significant.

Blocked Designs

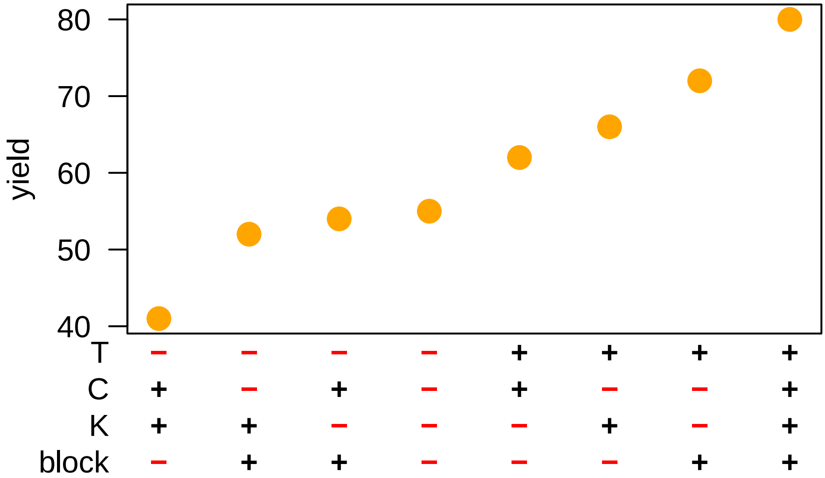
- Sometimes logistics force us to group runs into "blocks".
- Examples
  - Mice need to be housed in separate cages.
  - The experimenter cannot do all runs in a single day/batch.
  - Two experimenters need to split up the runs.
  - A single container of reagent doesn't cover the experiment.

Blocked Designs

- Sometimes logistics force us to group runs into "blocks".
- Examples
  - Mice need to be housed in separate cages.
  - The experimenter cannot do all runs in a single day/batch.
  - Two experimenters need to split up the runs.
  - A single container of reagent doesn't cover the experiment.
- A "blocking factor" is added to these experiment to capture inter-block differences.
- Blocks are added to designs and analyzed like any other factor.
- Most block $\times$ factor interactions are ignored.

Blocking our Pilot Plant Experiment

```r
pilot <- read.csv("PilotPlantBlocked.csv")
farplot(pilot, "yield", c("T","C","K","block"))
```



Building our normal model

```
model <- lm(yield ~ T*K*C, data=pilot); show_effects(model)
```

```
## (Intercept)    60.25
##           T     9.75
##           K     -.5
##           C    -1.
##         T:K     3.5
##         T:C     2.
##         K:C     1.75
##       T:K:C     4.25
```

Building our normal model

```
model <- lm(yield ~ T*K*C, data=pilot); show_effects(model)
```

```
## (Intercept)    60.25
##           T     9.75
##           K     -.5
##           C    -1.
##         T:K     3.5
##         T:C     2.
##         K:C     1.75
##       T:K:C     4.25
```

```
model <- lm(yield ~ T*K*C + block, data=pilot); show_effects(model)
```

```
## (Intercept)    60.25
##           T     9.75
##           K     -.5
##           C    -1.
##       block     4.25
##         T:K     3.5
##         T:C     2.
##         K:C     1.75
##       T:K:C    NA.
```