Two-Level Factors: The 2^k Factorial Design

- When several factors may affect a response, often each has just two levels; e.g.:
  - comparing two methods for one step in a process;
  - presence or absence of some ingredient;
  - low and high settings of a quantitative factor.

- k factors, each with 2 levels, give 2^k treatment combinations.

- The 2^k factorial design uses all 2^k treatments.

- It requires the fewest runs of any factorial design for k factors.

- Often used at an early stage: *factor screening* experiments.

Notation

- Factors are usually A, B, etc.
- The two levels of each are usually denoted "+" and "-".
- E.g. 2^2:

Factor A	Factor B	Treatment Combination	Label	Total of Responses
-	-	A low, B low	(1)	(1)
+	-	A high, B low	a	a
-	+	A low, B high	b	b
+	+	A high, B high	ab	ab

Effects

- Effect of A at low level of B is difference between average responses: $\frac{a}{n} - \frac{(1)}{n}$.

- Effect of A at high level of B is difference between average responses: $\frac{ab}{n} - \frac{b}{n}$.

- *Main effect* of A is the average of these:

$$A = \frac{1}{2n}[ab + a - b - (1)]$$

- Similarly main effect of B is

$$B = \frac{1}{2n}[ab + b - a - (1)]$$

- The *interaction effect* AB is *one half* of the difference between:

  - the effect of A at the *high* level of B; and

  - the effect of A at the *low* level of B:

$$AB = \frac{1}{2n}[ab + (1) - a - b]$$

- AB is also the difference between the effects of B at the two levels of A.

Example: 2^2 with 3 reps

A	B	Rep	Yield
-	-	I	28
+	-	I	36
-	+	I	18
+	+	I	31
-	-	II	25
+	-	II	32
-	+	II	19
+	+	II	30
-	-	III	27
+	-	III	32
-	+	III	23
+	+	III	29

- R analysis:

```
yield = read.table("yield.txt", header = TRUE);
summary(aov(Yield ~ A * B, yield));

            Df  Sum Sq Mean Sq F value    Pr(>F)
A            1 208.333 208.333 53.1915 8.444e-05 ***
B            1  75.000  75.000 19.1489  0.002362 **
A:B          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
```

- To get the effects:

```
coded = function(x) ifelse(x == x[1], -1, 1);
summary(lm(Yield ~ coded(A) * coded(B), yield));
```

- with output:

```
Call:
lm(formula = Yield ~ coded(A) * coded(B), data = yield)

Residuals:
   Min     1Q Median     3Q    Max
-2.000 -1.333 -0.500  1.083  3.000

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)        27.5000     0.5713  48.135 3.84e-11 ***
coded(A)            4.1667     0.5713   7.293 8.44e-05 ***
coded(B)           -2.5000     0.5713  -4.376  0.00236 **
coded(A):coded(B)   0.8333     0.5713   1.459  0.18278
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 1.979 on 8 degrees of freedom
Multiple R-Squared: 0.903,      Adjusted R-squared: 0.8666
F-statistic: 24.82 on 3 and 8 DF,  p-value: 0.0002093
```

- Note: As A changes from its low level to its high level, the coded variable changes by $1-(-1)=2$, so the effects and interaction(s) are *twice* the values in the "Estimate" column.

- Hand calculation (Yates's algorithm):

Sums	Intermediate Step	Effects
(1)	$(1)+a$	$(1)+a+b+ab=4n\widehat{\mu}$
a	$b+ab$	$-(1)+a-b-ab=2nA$
b	$-(1)+a$	$-(1)-a+b+ab=2nB$
ab	$-b+ab$	$(1)-a-b+ab=2nAB$

Eight additions and subtractions instead of twelve! (More advantage for large k.)