

10

Design of Experiments in Bioassay — Block Designs

10.1: Learning Objectives

1. Explain why classical “all-pairwise-contrasts-efficient” block designs are not automatically optimal for bioassay.
2. Construct randomized complete block (RCBD) and incomplete block designs for parallel-line and slope-ratio assays.
3. Analyze block-design bioassay data, adjusting dose contrasts for block effects.
4. Implement RCBD and incomplete-block ANOVA for bioassay data in R and Python.

10.2: Why Bioassay Needs Special Designs

Classical block-design theory (e.g. balanced incomplete block designs, BIBDs) is typically built to estimate **all** pairwise treatment differences with equal, minimum variance. In a bioassay, however, only three (or four) specific **contrasts** matter: preparation, combined regression, parallelism (parallel-line); or blank, intersection, and the two individual regressions (slope-ratio). A design that is globally “efficient” in the classical sense may not be optimal — or even convenient — for these particular contrasts. Bioassay design theory therefore asks: which block designs estimate the preparation/regression/parallelism (or blank/intersection) contrasts free of block effects, with minimum variance for those specific contrasts?

10.3: Randomised Complete Block Design (RCBD)

When the number of homogeneous experimental units per block equals the total number of dose combinations $v = m_1 + m_2$ (parallel-line) or $v = m_1 + m_2 + 1$ (slope-ratio, including the blank), a **randomized complete block design** — each block containing every dose combination exactly once, order randomized — is the natural, simplest choice. Because every dose appears once per block, all treatment contrasts (including the preparation/regression/parallelism contrasts) are automatically estimated free of block effects, and the analysis of Chapters 6–7 applies directly after subtracting a between-blocks sum of squares.

Table 10.1: ANOVA for an RCBD parallel-line assay (b blocks, $v = m_1 + m_2$ doses)

Source	d.f.
Blocks	$b - 1$
Doses (= Preparation + Combined Regression + Parallelism + Deviation)	$v - 1$
Error	$(b - 1)(v - 1)$
Total	$bv - 1$

10.4: Incomplete Block Designs

When block size $k < v$ (e.g. a litter of animals too small to receive all doses, or a microtiter plate row too short for all treatment combinations), an **incomplete block design** must be used. The classical requirement is that the design be **connected** for the specific contrasts of interest and, ideally, **balanced** (every pair of doses occurs together in the same number of blocks λ) so that all elementary contrasts — and hence, in particular, the preparation/regression/parallelism contrasts — are estimable free of block effects with equal precision. Analysis proceeds via the standard intra-block

BIBD adjustment: adjusted dose totals are obtained by subtracting a multiple of block totals, and adjusted sums of squares replace the raw dose sums of squares of the RCBD table.

Table 10.2: ANOVA for an incomplete block parallel-line assay

Source	d.f.
Blocks (unadjusted)	$b - 1$
Doses (adjusted)	$v - 1$
— Preparation	1
— Combined regression	1
— Parallelism	1
— Deviation from regression	$v - 4$
Intra-block error	by subtraction
Total	$N - 1$

This is precisely the structure of the 18-litter vitamin D assay analyzed in Chapter 6, Example 6.2, where each litter (block) of size 4 could not accommodate all doses.

10.5: Constructing a Simple Incomplete Block Design by Hand

A common, easily-constructed incomplete block scheme for a symmetric $2m$ -point parallel-line assay with block size $k < 2m$ is obtained by cyclic development: list the $2m$ treatments $0, 1, \dots, 2m - 1$; choose an “initial block” of size k ; generate further blocks by adding $1, 2, \dots \pmod{2m}$ to every element of the initial block, repeating until the desired number of blocks (and, ideally, balance) is achieved. Verification that every pair of doses co-occurs equally often (λ constant) confirms balance.

10.6: Worked Examples

Example 10.1: RCBD Parallel-Line ANOVA

A symmetric 6-point parallel-line assay ($m = 3$ per preparation) is run in $b = 5$ complete blocks (litters), one animal per dose per block.

R:

```
set.seed(10)
block <- factor(rep(1:5, each = 6))
dose <-
factor(rep(c("S1", "S2", "S3", "T1", "T2", "T3"),
times = 5))
logdose <-
rep(c(log(20), log(40), log(80), log(20), log(40),
log(80)), times = 5)
prep <-
factor(rep(c("S", "S", "S", "T", "T", "T"), times
= 5))
response <- 10 + 5 * logdose
+ as.numeric(prepare == "T") * 0.5 + rnorm(30, sd = 1.2)
+
rep(rnorm(5, sd = 1), each = 6) # block
effect
fit <- aov(response ~ block + prep * logdose)
summary(fit)
```

The block term absorbs the between-litter variation; the prep, logdose, and prep:logdose terms recover the preparation, combined-regression, and parallelism contrasts exactly as in the completely-randomized capon example of Chapter 6, but now with the between-block noise removed from the error term, generally sharpening all -tests.

Example 10.2: Efficiency Gain From Blocking

Compare the residual mean square from Example 10.1's blocked analysis with what would be obtained ignoring the block structure (a completely randomized analysis of the same data).

R:

```
fit_crd <- aov(response ~ prep*logdose)  #
ignoring block
anova(fit_crd)
anova(fit)    # from Example 10.1, blocked
```

If litters differ appreciably in baseline responsiveness (a realistic assumption for most animal-based bioassays), the blocked analysis's error mean square will be smaller than the unblocked (CRD) analysis's, directly increasing power for the parallelism and preparation tests — the core practical justification for using RCBD whenever a plausible blocking factor (litter, day, plate, operator) exists.

Example 10.3: Cyclic Construction of an Incomplete Block Design

Construct an incomplete block design for $v = 6$ dose combinations (labelled $0, \dots, 5$) with block size $k = 4$, using cyclic development from the initial block $\{0, 1, 2, 4\}$ (a classical difference-set choice known to give a balanced design for $v = 6, k = 4$ up to replication).

Python:

```
initial_block = [0,1,2,4]
v = 6
blocks = []
for shift in range(v):
    blocks.append(sorted((x+shift) % v for x
in initial_block))
```

```
for b in blocks:
    print(b)
```

Running this generates 6 blocks of size 4; a quick tally of how often each pair (i, j) co-occurs across the 6 blocks confirms (or refutes) balance — an essential check before adopting a cyclic design in practice, since not every initial block yields a balanced design.

Example 10.4: Intra-Block Analysis of the Vitamin D Incomplete Block Assay

Reproduce the key adjusted sums of squares of the 18-litter, 6-dose vitamin D assay first presented in Chapter 6, Example 6.2, using R’s built-in incomplete-block handling.

R:

```
# vitamin_d is assumed to be a data frame
with columns: response, dose (6 levels),
block (18 levels)
library(MASS) # or use aov() directly;
unequal-size incomplete blocks handled via lm
with block+dose
fit <- aov(response ~ block + dose, data =
vitamin_d)
summary(fit)
# Adjusted dose sums of squares appear in the
'dose' row when block is fitted first (Type I
SS)
```

Comparing the dose row’s sum of squares here with the unadjusted between-dose sum of squares (ignoring block) quantifies exactly how much of the raw between-dose variation was actually attributable to which litter an observation happened to fall in — the practical meaning of “adjustment for block effects.”

Example 10.5: Choosing Between RCBD and CRD at the Design Stage

A pharmacologist has 24 mice, naturally arriving in 4 cages of 6 (a plausible blocking factor: cage). The assay requires 6 dose combinations. Recommend a design.

Solution: Since cage size (6) equals the number of dose combinations (6), a randomized complete block design — one mouse per dose per cage, doses randomly assigned within each cage — is both feasible and preferable to a completely randomized design, since cage-to-cage variation (litter effects, husbandry conditions) can be removed from the error term at essentially no cost in design complexity (Section 10.3).

Example 10.6 : When Balance Cannot Be Achieved

A slope-ratio assay requires $v = 7$ dose combinations (including blank), but the only available block size is $k = 3$ (small tissue-culture plates holding 3 wells each). Discuss the design implications.

Solution. With $k = 3 < v = 7$, no design can place every pair of doses together in every block; a resolvable, ideally balanced (or at least “regular graph design”/partially balanced) incomplete block design must be constructed, likely requiring more than v blocks to approach balance. Because $k = 3$ is small relative to $v = 7$, some pairs of doses may co-occur less often than others unless careful combinatorial design (e.g. a resolvable design based on a known combinatorial structure) is used; the resulting **partially balanced incomplete block design (PBIBD)** would give different (associate-class-dependent) variances for different contrasts, complicating — but not preventing — the blank/intersection/regression analysis of Chapter 7.

Example 10.7 — Simulating Design Efficiency in Python

Simulate data under both RCBD and CRD for the same total sample size and compare the resulting standard errors of the combined-regression contrast, to make Example 10.2's point computationally rather than just narratively.

Python:

```
import numpy as np, statsmodels.formula.api
as smf
import pandas as pd

rng = np.random.default_rng(20)
n_blocks = 6
logdose = np.tile([np.log(20), np.log(40),
np.log(80)]*2, n_blocks)
prep = np.tile(['S']*3+ ['T']*3, n_blocks)
block = np.repeat(np.arange(n_blocks), 6)
block_effect = np.repeat(rng.normal(0, 2.0,
n_blocks), 6) # sizable block variation
y = 10+5*logdose + (prep=='T')*0.5+
block_effect + rng.normal(scale=1.2,
size=6*n_blocks)

df = pd.DataFrame({'y':y, 'logdose':logdose,
'prep':prep, 'block':block.astype(str)})
fit_block = smf.ols('y ~ C(block) +
prep*logdose', data=df).fit()
fit_noblock = smf.ols('y ~ prep*logdose',
data=df).fit()
print("Blocked residual SE:",
np.sqrt(fit_block.mse_resid))
print("Unblocked residual SE:",
np.sqrt(fit_noblock.mse_resid))
```

With a deliberately large simulated block effect, the blocked model's residual standard error should be visibly smaller, giving students a hands-on demonstration of the design principle of Section 10.3.

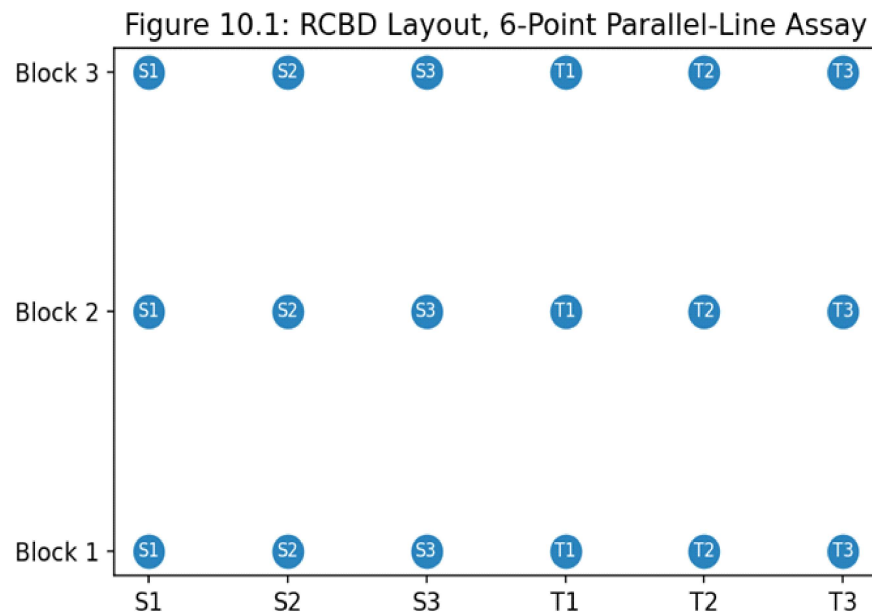


Figure 10.1: *RCBD layout, 6-point parallel-line assay*

10.7: Chapter Summary

- Bioassay designs must protect the specific preparation/regression/parallelism (or blank/intersection) contrasts from block effects, not merely all pairwise contrasts.
- RCBD is the natural choice whenever block size equals the number of dose combinations.
- Incomplete block designs (ideally balanced) are required when block size is smaller; cyclic construction from a well-chosen initial block is a practical method for generating them.

- Blocking reduces residual error variance whenever a real, non-negligible blocking factor (litter, cage, plate, day) exists, directly sharpening every downstream hypothesis test and Fieller interval in Chapters 3–9.

10.8: Exercises

Exercise 10.1: A parallel-line assay needs $v = 8$ dose combinations ($m_1 = m_2 = 4$), and blocks (operating days) can accommodate exactly 8 runs each. Sketch the RCBD layout for 3 blocks and write the corresponding ANOVA degrees-of-freedom table (blocks, doses split into preparation/regression/parallelism/deviation, error, total).

Exercise 10.2: Using the cyclic-construction method of Example 10.3, construct an incomplete block design for $v = 5$ treatments with block size $k = 3$, starting from initial block $\{0, 1, 3\}$ (a known balanced difference set for $v = 5$). Verify balance by tallying pairwise co-occurrences.

Exercise 10.3: Explain, in your own words, why a design that is “efficient” for estimating all $\binom{v}{2}$ pairwise treatment differences (the classical BIBD optimality criterion) is not necessarily the design that minimizes the variance of the parallelism contrast specifically. (Hint: think about how the parallelism contrast weights the doses very differently from a simple pairwise difference.)

Exercise 10.4: Using R or Python, simulate an incomplete block parallel-line assay (choose your own v , k , block count, and effect sizes) and compare the adjusted vs. unadjusted (for blocks) dose sum of squares, reporting the percentage reduction in residual variance attributable to blocking.

(Full worked solutions appear in Appendix A.)