Module 2: Completely Randomized Designs

Designing a CRD

Completely Randomized Design (CRD)

- The simplest experimental design
- Treatments are assigned to experimental units **completely at random**

What Makes a Design a CRD?

- No blocking or grouping
- Random assignment only

All unexplained variability goes into the **experimental error**

When Is a CRD Appropriate?

A CRD works best when:

- Experimental units are relatively homogeneous
- The experiment is small
- Few extraneous variables are present

Advantages of a CRD

- Very flexible
- Unequal sample sizes allowed
- Analysis remains straightforward
- Missing data is not catastrophic

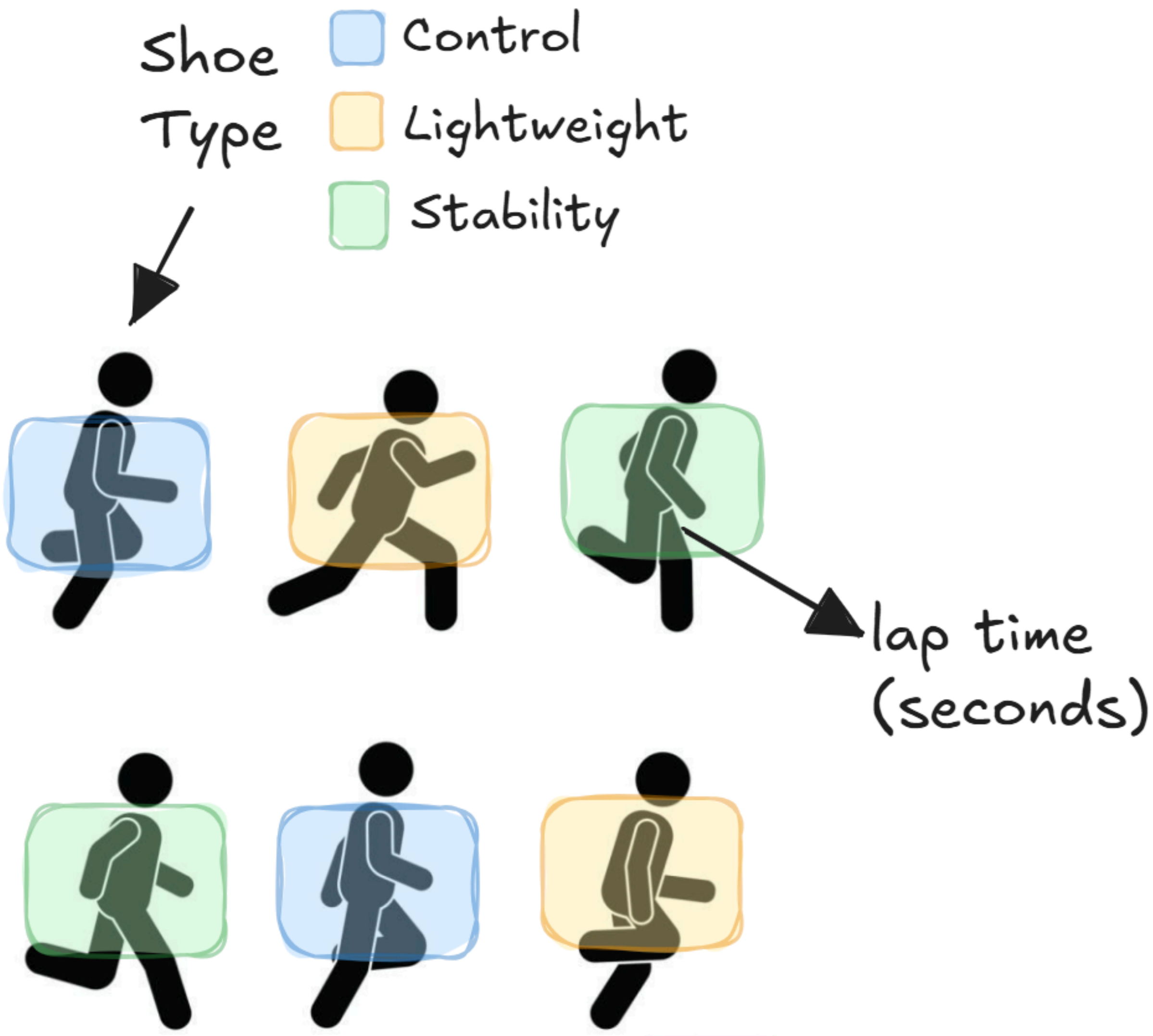
Disadvantages of a CRD

- Sensitive to heterogeneity
- Larger experimental error when experimental units differ

Example 2.1: Running Shoes

An experiment is conducted to investigate how different **running shoes** affect running performance (measured in seconds to complete a fixed route). We have three shoe types that we want to compare: *Control (regular trainers)*, *Lightweight shoes*, and *Stability shoes*. There are only 6 runners available for the experiment, so we will use two runners for each shoe type. The 6 runners gather at a track, and before beginning the trial they are each randomly assigned one shoe type. Each runner completes the same 400-meter route once, and their lap time is recorded.

Example 2.1: Running Shoes (Blueprint)



Example 2.1: Running Shoes

Response: Lap time (seconds)

Treatment structure: One-way treatment structure with 3 levels of shoe type (control, lightweight, and stability) making up the $t=3$ treatments.

Experimental structure: Shoe type is assigned to individuals (e.u.) in a CRD with $r = 2$ ($N = 6$). The lap time (seconds) is measured once for each individual (m.u.).

CRD Randomization (in R)

```r
library(edibble)

crd_des <- design(name = "Shoe Type") |>
   set_units(runner = 6) |>
   set_trts(shoe_type = c("Control",
                          "Lightweight",
                          "Stability")
          ) |>
   allot_trts(shoe_type ~ runner) |>
   assign_trts("random")

crd_table <- serve_table(crd_des)
crd_table
```

```
# An edibble: 6 x 2
   runner   shoe_type
   <U(6)>      <T(3)>
    <chr>       <chr>
1 runner1   Stability
2 runner2   Stability
3 runner3     Control
4 runner4     Control
5 runner5 Lightweight
6 runner6 Lightweight
```

CRD Randomization (in R)

```r
library(dplyr)

crd_table2 <- crd_table |>
  mutate(laptime_s = NA)

crd_table2
```

```
# An edibble: 6 x 3
   runner    shoe_type laptime_s
   <U(6)>       <T(3)>
    <chr>        <chr> <lgl>
1 runner1    Stability NA
2 runner2    Stability NA
3 runner3      Control NA
4 runner4      Control NA
5 runner5 Lightweight NA
6 runner6 Lightweight NA
```

- Alternatively, use `bind_cols()` to add the CRD assignment to an already existing data set.
- Use `write_csv()` or `write_xlsx()` to export your design for data entry.

CRD Randomization (in JMP)

Follow the Scribe tutorial using `DOE > Classical > Full Factorial Design` in JMP to create a randomization of treatments.