

A7. Versuchsplanung

Auswahl und Erzeugung eines passenden Versuchsdesigns vor der Datenerhebung

Dr. Paul Schmidt

Um alle in diesem Kapitel verwendeten Pakete zu installieren und zu laden, kann man folgenden Code ausführen:

```
for (pkg in c("agricolae", "desplot", "FieldHub", "glue", "tidyverse")) {
  if (!require(pkg, character.only = TRUE)) install.packages(pkg)
}

library(agricolae)
library(desplot)
library(FieldHub)
library(glue)
library(tidyverse)
```

Die übrigen Kapitel dieser Section führen durch die Analyse von Daten, die aus verschiedenen Versuchsdesigns stammen - einem CRD in Kapitel 1, einem RCBD in Kapitel 2, einem Lateinischen Quadrat in Kapitel 3, einem Alpha-Design in Kapitel 4, einem augmentierten Design in Kapitel 5 und einem auflösbaren Zeilen-Spalten-Design in Kapitel 6. Dieses Kapitel tritt einen Schritt zurück und stellt die Frage, die eigentlich vor jeder Datenerhebung beantwortet werden sollte: **Welches Design sollte überhaupt verwendet werden, und wie wird es erzeugt?**

Die Kurzversion

Gute Versuchsplanung ist günstiger als gute Statistik. Keine noch so ausgefeilte Modellierung kann Information zurückgewinnen, die nie erhoben wurde, und ein ungünstiger Feldplan kann selbst eine sorgfältige Analyse zu mehrdeutigen Ergebnissen führen lassen. Drei Prinzipien, die alle auf Ronald A. Fishers Klassiker *The Design of Experiments* (1935) zurückgehen, bilden den praktischen Kern:

- **Wiederholung** - jedes Prüfglied muss mehr als einmal auftreten, damit die Variation *zwischen* den Prüfgliedern von der Variation *innerhalb* eines Prüfglieds getrennt werden kann. Ohne Wiederholung gibt es keine Residualvarianz und damit keine Möglichkeit zu beurteilen, ob ein beobachteter Unterschied real ist.
- **Randomisierung** - die Zuordnung der Prüfglieder zu den Versuchseinheiten muss zufällig erfolgen. Die Randomisierung rechtfertigt überhaupt erst die statistische Inferenz und schützt vor unbekanntem Konfundieren wie etwa einem Gradienten in der Bodenfruchtbarkeit oder einer Drift in den Laborbedingungen über den Verlauf eines Tages.
- **Lokale Kontrolle (Blockbildung)** - wenn bekannt ist, dass sich Versuchseinheiten systematisch unterscheiden (z.B. entlang eines Hangs, zwischen Schalen, zwischen Bearbeitern), entfernt die Gruppierung ähnlicher Einheiten in Blöcke und die Randomisierung der Prüfglieder *innerhalb* jedes Blocks diese Störvariation aus dem Prüfgliedvergleich.

💡 Schnelle Entscheidungsregel

Wenn die Versuchseinheiten im Wesentlichen homogen sind und die Anzahl der Prüfglieder klein ist, verwendet man ein **CRD**. Lässt sich eine einzelne Quelle der Heterogenität identifizieren (Bodengradient, Position auf der Tischfläche, Messtag), verwendet man ein **RCBD**. Wirkt die Heterogenität in zwei orthogonalen Richtungen, verwendet man ein **Lateinisches Quadrat** (bei gleicher Anzahl von Zeilen, Spalten und Prüfgliedern) oder ein **auf lösbares Zeilen-Spalten-Design**. Gibt es zu viele Prüfglieder, um sie in einen vollständigen Block zu fassen, verwendet man ein **Alpha-Design** oder ein anderes Design mit unvollständigen Blöcken. Müssen viele neue Genotypen mit begrenzten Ressourcen gescreent werden, verwendet man ein **augmentiertes Design** oder ein **p-rep-Design**.

Eine starke Referenz für die praktische Seite dieser Entscheidung ist M. D. Casler [1], das Wiederholung, Randomisierung und Blockbildung anhand landwirtschaftlicher Beispiele durchgeht und vor einem ersten Feldversuch lohnend in Gänze zu lesen ist.

Auswahl eines Designs

Die Wahl des Designs ist keine Geschmacksfrage, sondern eine Konsequenz aus der Versuchsfrage und dem Versuchsmaterial. Die folgende Checkliste geht die relevanten Aspekte in der Reihenfolge durch, in der sie typischerweise zu bindenden Restriktionen werden.

Anzahl und Art der Prüfglieder

Die Anzahl der Prüfgliedstufen ist meist die erste harte Restriktion. Mit 3 bis 8 Prüfgliedern bleiben die meisten klassischen Designs praktikabel. Mit 20, 50 oder mehr Einträgen - was in der Pflanzenzüchtung üblich ist, wo Hunderte von Kandidaten-Genotypen um Platz in einer Nursery konkurrieren - werden Designs mit vollständigen Blöcken undurchführbar, und eine Struktur mit unvollständigen Blöcken wird erforderlich.

Eine zweite Überlegung ist, ob die Prüfglieder von gleichem Interesse sind. Sollen alle Prüfglieder mit vergleichbarer Präzision geschätzt werden, ist ein Standarddesign (CRD, RCBD, Alpha) angemessen. Dient eine Gruppe von Prüfgliedern als Referenz (Checks, Standards) und besteht eine andere aus weniger getesteten Einträgen (neue Linien, neuartige Formulierungen), verteilt ein **augmentiertes Design** (Kapitel 5) die Wiederholungen dorthin, wo sie am wichtigsten sind.

Struktur des Versuchsmaterials

Versuchseinheiten sind selten wirklich homogen. In einem Feld variiert die Fruchtbarkeit tendenziell entlang eines Gradienten oder in Flecken. In einem Gewächshaus variieren Licht und Temperatur mit der Position auf der Tischfläche. In einem Labor führen Chargen von Reagenzien oder Messtage zu Variation, die nichts mit den Prüfgliedern zu tun hat. Diese Quellen vor der Randomisierung zu identifizieren, ist es, was aus einem möglicherweise veräuschten Experiment einen sauberen Vergleich macht.

- **Eine Richtung der Heterogenität** - man verwendet ein **RCBD**, wenn alle Prüfglieder in einen Block passen, oder ein **Alpha-Design**, wenn sie es nicht tun.

- **Zwei orthogonale Richtungen der Heterogenität** - man verwendet ein **Lateinisches Quadrat** für kleine, balancierte Settings oder ein **auflösbares Zeilen-Spalten-Design** (Kapitel 6) für größere Versuche.
- **Keine identifizierbare Quelle der Heterogenität** - ein **CRD** ist die einfachste und effizienteste Wahl. Blöcke "vorsichtshalber" hinzuzufügen kostet Freiheitsgrade ohne Nutzen.

Ressourcenrestriktionen

Wiederholung erfordert Versuchseinheiten, und Einheiten kosten Geld, Land und Zeit. Wenn die Ressourcen knapp sind, sind zwei Strategien üblich. **Augmentierte Designs** behalten eine kleine Anzahl wiederholter Checks bei und fügen daneben unwiederholte Testeinträge hinzu, was indirekte Vergleiche über die Check-Mittelwerte ermöglicht. **Partiell wiederholte (p-rep) Designs** wiederholen nur einen Bruchteil der Einträge (typischerweise 20 bis 40 Prozent), was eine Residualschätzung liefert und dabei deutlich weniger Parzellen verbraucht als eine vollständige Wiederholung. H.-P. Piepho *et al.* [2] diskutiert p-rep-Designs und ihre Abwägungen im Detail.

Einzelort- versus Mehrumwelt-Versuche

Viele reale Experimente werden nicht einmalig durchgeführt, sondern über Orte und/oder Jahre wiederholt. Die Designwahl reicht dann über einen einzelnen Standort hinaus: Jeder Ort sollte ein für seine lokalen Bedingungen geeignetes Design verwenden, und die Menge der Orte sollte so gewählt werden, dass die Genotyp-Umwelt-Interaktion geschätzt werden kann. H. P. Piepho, A. Bückse, and K. Emrich [3] und H. P. Piepho, A. Bückse, and C. Richter [4] liefern den statistischen Rahmen für Serien amtlicher Versuche; die praktische Konsequenz für die Designerzeugung ist, dass dasselbe Skript pro Umwelt separat ausgeführt werden sollte, typischerweise mit einem anderen Zufalls-Seed, damit räumliche Muster nicht über die Standorte hinweg wiederholt werden.

Erzeugung eines Designs in R

Zwei R-Pakete decken nahezu alle in der Praxis vorkommenden Designs ab:

- `{FieldHub}` - ein moderner, gut dokumentierter Werkzeugkasten, der auch eine Shiny-App bereitstellt. Seine Funktionen geben eine Liste zurück, die sowohl das Feldbuch (`$fieldBook`) als auch ein fertiges ggplot-Layout enthält (`$p` innerhalb von `plot(out)`).
- `{agricolae}` - der langjährige Klassiker, weiterhin nützlich besonders für schnelle lehrbuchartige Designs. Seine Funktionen (z.B. `design.crd()`, `design.rcb()`, `design.alpha()`) geben die Randomisierung als `$book`-Data-Frame zurück.

Beide sind kennenswert. Die folgenden Beispiele verwenden `{FieldHub}` für Layout-Plots und `{agricolae}` für eine einfache textbasierte Randomisierung.

Ein CRD mit FieldHub

Ein CRD platziert die Prüfglieder ohne jede strukturelle Restriktion auf den Parzellen - die Randomisierung ist völlig frei. Das folgende Beispiel erzeugt ein CRD für vier Prüfglieder mit fünf Wiederholungen:

```
set.seed(42)
```

```

crd_out <- CRD(
  t      = paste0("T", 1:4),
  reps   = 5,
  plotNumber = 101,
  seed   = 42
)

crd_out$fieldBook %>% head()

```

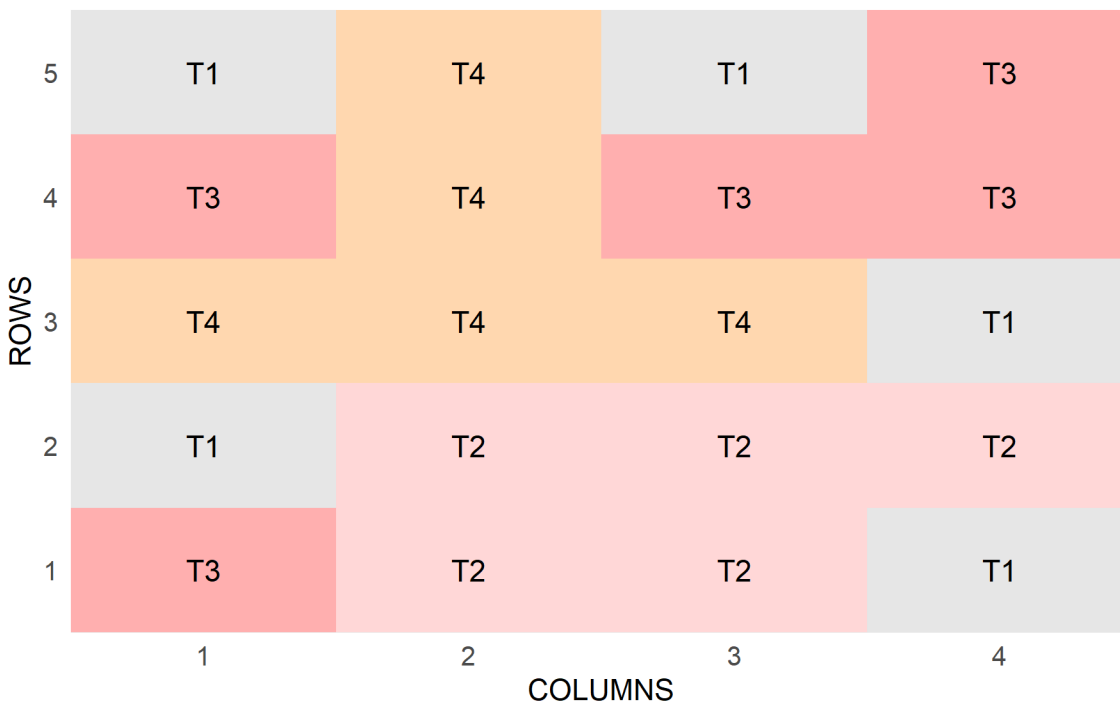
	ID	LOCATION	PLOT	REP	TREATMENT
1	1	1	101	3	T3
2	2	1	102	1	T2
3	3	1	103	3	T2
4	4	1	104	5	T1
5	5	1	105	2	T2
6	6	1	106	4	T2

Der `$fieldBook`-Slot ist die eigentliche Randomisierungstabelle - eine Zeile pro Versuchseinheit, mit Parzellennummern, Prüfglied-Bezeichnungen und etwaigen Blockvariablen. Das ist die Datei, die an die Feldtechniker oder das Laborpersonal weitergegeben wird, die das Experiment durchführen.

Ein einfacher Layout-Plot kann direkt mit `{FieldHub}` erzeugt werden:

```
plot(crd_out)
```

Completely Randomized Design 5X4



Ein RCBD und ein Alpha-Design

Dasselbe Muster funktioniert für andere Designs. Für ein RCBD mit 6 Prüfgliedern in 4 vollständigen Blöcken:

```

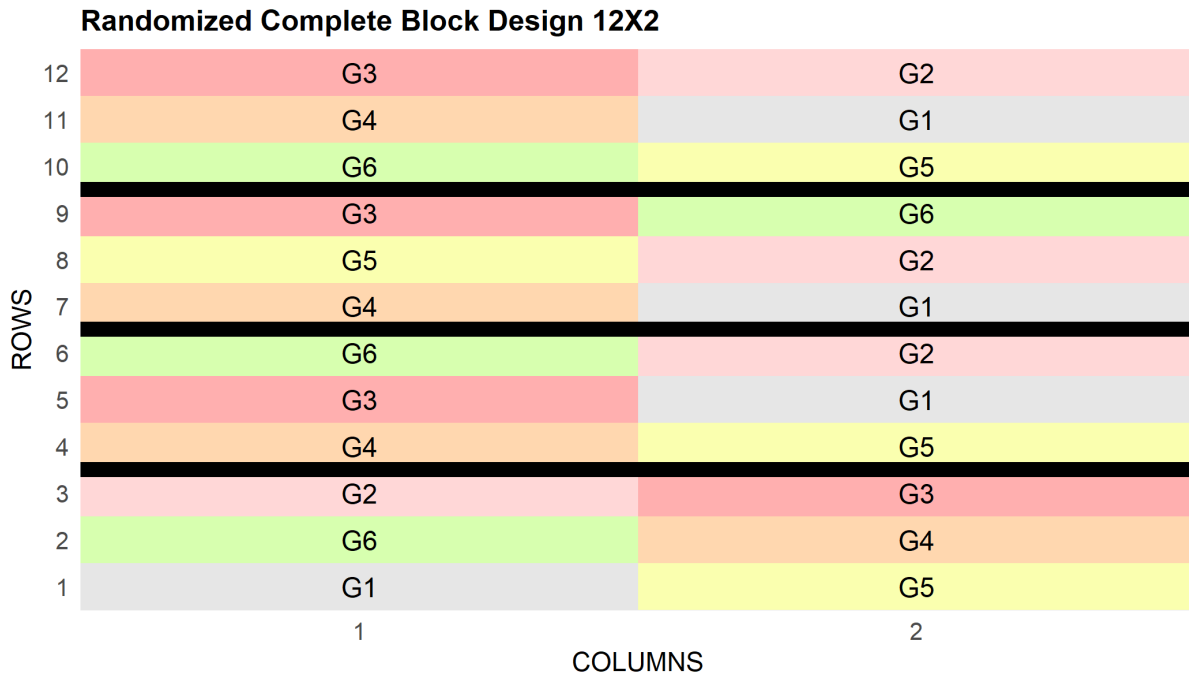
rcbd_out <- RCBD(
  t      = paste0("G", 1:6),
  reps   = 4,
  plotNumber = 101,

```

```

  seed = 42
)
plot(rcbd_out)

```



Und für ein Alpha-Design mit 15 Prüfgliedern in unvollständigen Blöcken der Größe 3, dreifach wiederholt:

```

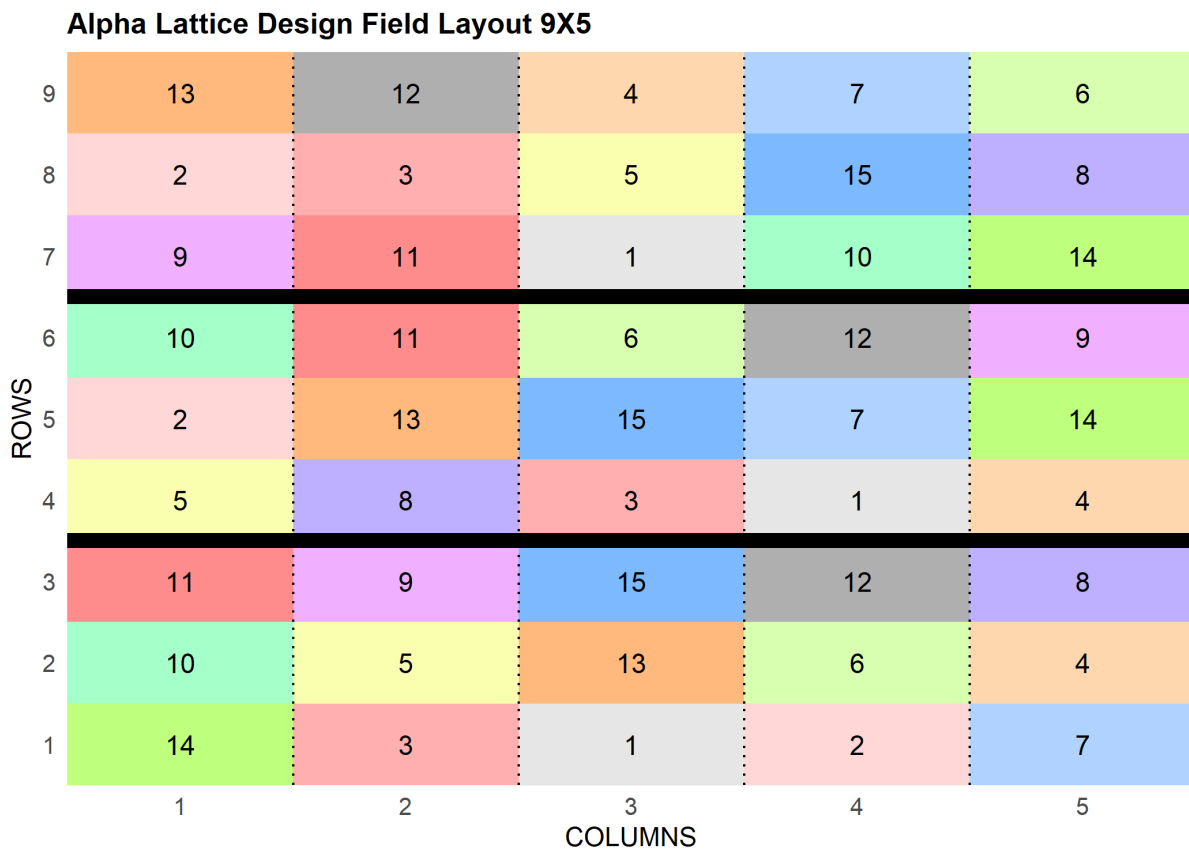
alpha_out <- alpha_lattice(
  t = 15,
  k = 3,
  r = 3,
  plotNumber = 101,
  seed = 42
)
plot(alpha_out)

```

```

Warning in geom_tileborder(aes(group = 1, grp = .data[[out2.string]]), color =
out2.gpar$col, : Ignoring empty aesthetic: `linewidth`.

```



Die Funktion `alpha_lattice()` läuft nur, wenn sich die Anzahl der Prüfglieder als Blockgröße mal Blöcke pro Wiederholung faktorisieren lässt ($v = sk$); passen die Zahlen nicht, meldet sie, welche Bedingungen verletzt wurden. Schon diese Prüfung allein ist ein guter Grund, ein dediziertes Paket zu verwenden, statt die Randomisierung von Hand zu schreiben.

Das agricolae-Äquivalent

Für Nutzer, die mit `{agricolae}` bereits vertraut sind, lässt sich dasselbe CRD mit einem einzigen Aufruf erzeugen:

```
crd_agri <- design.crd(
  trt = paste0("T", 1:4),
  r = 5,
  seed = 42
)
crd_agri$book %>% head()
```

```
plots r paste0("T", 1:4)
1 101 1 T1
2 102 1 T2
3 103 1 T4
4 104 2 T1
5 105 2 T4
6 106 1 T3
```

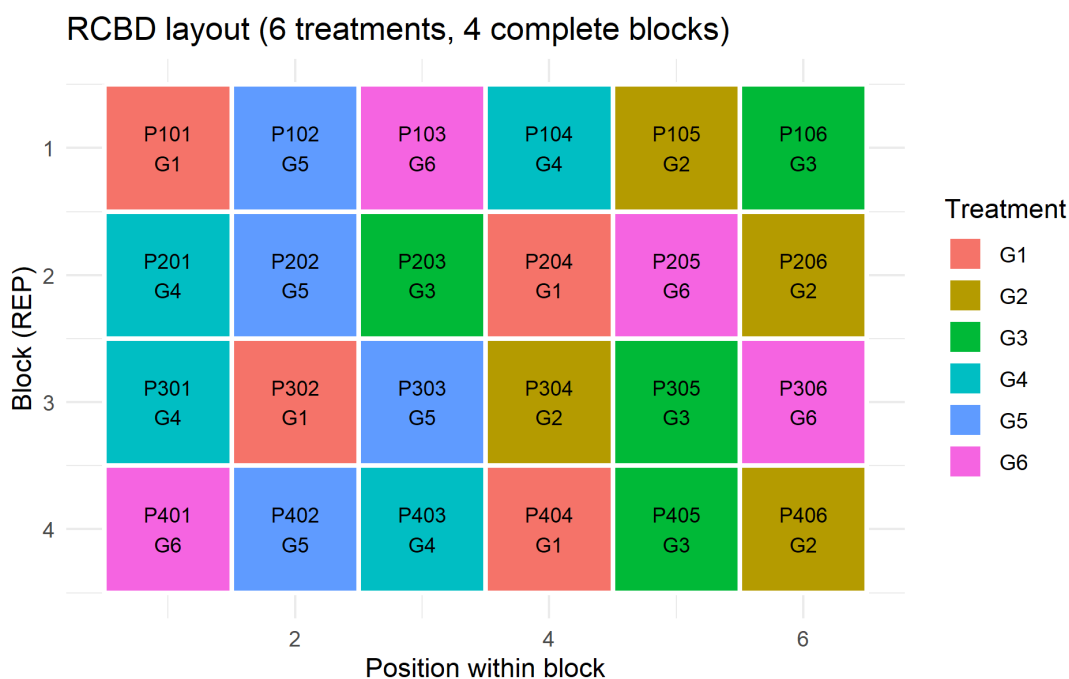
Die Ausgabe ist ein Data-Frame statt eines fertigen Plots, was praktisch ist, wenn die Randomisierung nur ein Schritt in einer größeren Pipeline ist (z.B. wenn sie mit einem Datenerfassungsbogen zusammengeführt oder in ein Feldmanagementsystem hochgeladen wird).

Eine selbst gezeichnete Feldkarte mit ggplot2

Unabhängig davon, welches Paket die Randomisierung erzeugt, verwandelt ein kurzes `ggplot2`-Rezept das Feldbuch in einen anpassbaren Layout-Plot. Das folgende Beispiel verwendet das oben erzeugte RCBD:

```
field_df <- rcbd_out$fieldBook %>%
  group_by(REP) %>%
  mutate(
    col = row_number(),
    plot_label = glue("P{PLOT}\n{TREATMENT}")
  ) %>%
  ungroup()

ggplot(field_df, aes(x = col, y = REP, fill = TREATMENT)) +
  geom_tile(colour = "white", linewidth = 1) +
  geom_text(aes(label = plot_label), size = 3) +
  scale_y_reverse(breaks = unique(field_df$REP)) +
  coord_equal() +
  labs(
    title = "RCBD layout (6 treatments, 4 complete blocks)",
    x = "Position within block",
    y = "Block (REP)",
    fill = "Treatment"
  ) +
  theme_minimal()
```



Dieses Rezept skaliert auf jedes der von `{FieldHub}` erzeugten Designs, weil sie alle ein `$fieldBook` mit denselben Spalten zurückgeben. Den Plot anzupassen - Blockränder hinzuzufügen, Checks hervorzuheben oder mehrere Umwelten als Facets darzustellen - ist dann eine Frage von Standard-`ggplot2`.

Praktische Tipps

Einige Punkte tauchen in realen Versuchen immer wieder auf und verdienen eine kurze Erwähnung.

Randeffekte

Parzellen am Rand eines Feldes, einer Schale oder einer Platte verhalten sich anders als Parzellen im Inneren: Sie erhalten mehr Licht, sind mehr Wind ausgesetzt oder sind für Schädlinge leichter zugänglich. Diesen Randeffekt zu ignorieren, kann die Prüfgliedmittelwerte von allem verzerren, was am Rand platziert wird. Zwei gängige Gegenmaßnahmen:

- Eine nicht beerntete **Schutzreihe oder Schutzspalte** rund um das gesamte Experiment hinzufügen (und mitunter zwischen den Blöcken). Das Schutzmaterial ist üblicherweise ein einzelner Check oder eine Füllsorte und wird nicht analysiert.
- Zeilen- und Spalteneffekte explizit modellieren (siehe Kapitel 6), was einen Teil des räumlichen Gradienten absorbiert, unabhängig davon, wo auf dem Feld ein Prüfglied platziert wurde.

Wiederholte Checks

Selbst in Designs, in denen die meisten Einträge nicht wiederholt werden (augmentiert, p-rep), wird üblicherweise eine kleine Menge gut bekannter **Check-Sorten** aufgenommen und über das gesamte Experiment wiederholt. Diese Checks dienen zwei Zwecken: Sie verankern die Analyse, indem sie in jedem Block einen Referenzpunkt liefern, und sie erlauben dem Experimentator, ungewöhnliche Bedingungen zu überwachen (wenn sich ein normalerweise stabiler Check in einem Block seltsam verhält, ist dieser Block verdächtig).

Seed-Werte und Reproduzierbarkeit

Alle oben gezeigten Funktionen nehmen ein `seed`-Argument entgegen. Den Seed zusammen mit dem Feldbuch festzuhalten, ist für die Reproduzierbarkeit zwingend - ohne ihn kann ein erneuter Lauf nicht exakt dieselbe Randomisierung erzeugen. In Versuchsserien über mehrere Umwelten sollte jede Umwelt einen *anderen* Seed verwenden, damit räumliche Muster nicht versehentlich von einem Standort zum nächsten kopiert werden.

Erst randomisieren, dann das Layout prüfen

Eine erzeugte Randomisierung ist nicht automatisch eine sinnvolle Randomisierung. Bevor das Experiment beginnt, sollte das Layout visuell inspiziert werden: Liegen zwei bestimmte Prüfglieder immer nebeneinander? Steckt ein einzelnes Prüfglied in der Ecke fest? Hat der Zufallszahlengenerator ein Muster erzeugt, das zufällig mit einem bekannten Gradienten zusammenfällt? Wenn etwas falsch aussieht, ändert man einfach den Seed und erzeugt ein neues Layout - das ist günstig und legitim, solange die Entscheidung *vor* der Datenerhebung getroffen wird.

Weiterführende Literatur

Für eine vertiefte Behandlung der hier abgedeckten Themen:

- **Allgemeine Prinzipien und Philosophie** - Fishers *The Design of Experiments* (1935) ist die ursprüngliche Quelle; eine sehr gut lesbare moderne Alternative ist Cochran & Cox, *Experimental Designs* (2. Aufl., Wiley, 1957), das nach wie vor das Referenzhandbuch für praktische Feldpläne ist.
- **Agromischer Fokus** - M. D. Casler [1] gibt einen prägnanten, praxisnahen Überblick über die drei grundlegenden Prinzipien anhand landwirtschaftlicher Beispiele.

- **Mehrumwelt-Versuche** - H. P. Piepho, A. BÜchse, and K. Emrich [3] und H. P. Piepho, A. BÜchse, and C. Richter [4] beschreiben den statistischen Rahmen für Versuchsserien, einschließlich der Frage, wie sich Information über Standorte und Jahre hinweg kombinieren lässt.
- **Moderne p-rep- und partiell wiederholte Designs** - H.-P. Piepho *et al.* [2] behandelt ressourceneffiziente Alternativen zur vollständigen Wiederholung, die in der Pflanzenzüchtung zum Standard geworden sind.
- **Software** - die Paket-Vignetten von `{FieldHub}` und `{agricolae}` sind der praktischste Ausgangspunkt, um eines der oben genannten Designs zu erzeugen.

i Kernaussagen

1. **Design vor Daten.** Das Design muss vor dem Beginn des Experiments gewählt und erzeugt werden; keine statistische Methode kann ein schlechtes Design im Nachhinein reparieren.
2. **Wiederholung, Randomisierung, lokale Kontrolle** - Fishers drei Prinzipien bleiben das Fundament jedes soliden Designs.
3. **Lass die Restriktionen das Design wählen.** Anzahl der Prüfglieder, Heterogenität des Materials und verfügbare Ressourcen engen die Wahl meist auf eine oder zwei Optionen ein.
4. **Verwende ein Paket.** `{FieldHub}` und `{agricolae}` decken nahezu alle in der Praxis vorkommenden Designs ab und eliminieren ganze Klassen von handcodierten Fehlern.
5. **Halte den Seed fest.** Die Randomisierung ist Teil des Experiments und muss reproduzierbar sein.
6. **Inspiziere das Layout visuell,** bevor das Experiment beginnt. Wenn es falsch aussieht, erzeuge es neu.

Bibliography

- [1] M. D. Casler, "Fundamentals of Experimental Design: Guidelines for Designing Successful Experiments," *Agronomy Journal*, vol. 107, no. 2, pp. 692–705, 2015, doi: 10.2134/agronj2013.0114.
- [2] H.-P. Piepho *et al.*, "One, two, three: portable sample size in agricultural research," *The Journal of Agricultural Science*, vol. 160, no. 6, pp. 459–482, 2022, doi: 10.1017/S0021859622000466.
- [3] H. P. Piepho, A. BÜchse, and K. Emrich, "A Hitchhiker's Guide to Mixed Models for Randomized Experiments," *Journal of Agronomy and Crop Science*, vol. 189, no. 5, pp. 310–322, 2003, doi: 10.1046/j.1439-037X.2003.00049.x.
- [4] H. P. Piepho, A. BÜchse, and C. Richter, "A Mixed Modelling Approach for Randomized Experiments with Repeated Measures," *Journal of Agronomy and Crop Science*, vol. 190, no. 4, pp. 230–247, 2004, doi: 10.1111/j.1439-037X.2004.00097.x.