Study design

- Generalizing
  - How do I want this to generalize?
  - What population to generalize to?
  - What is the scope of inference?
- Generalization is determined by the design not the analysis
- Study design is best done before data collection
  - simulation!

Study design

- Observational design
  - – focus: sampling
  - – estimation, prediction, weaker causal inference
- Experimental design
  - – manipulative
  - – causal inference

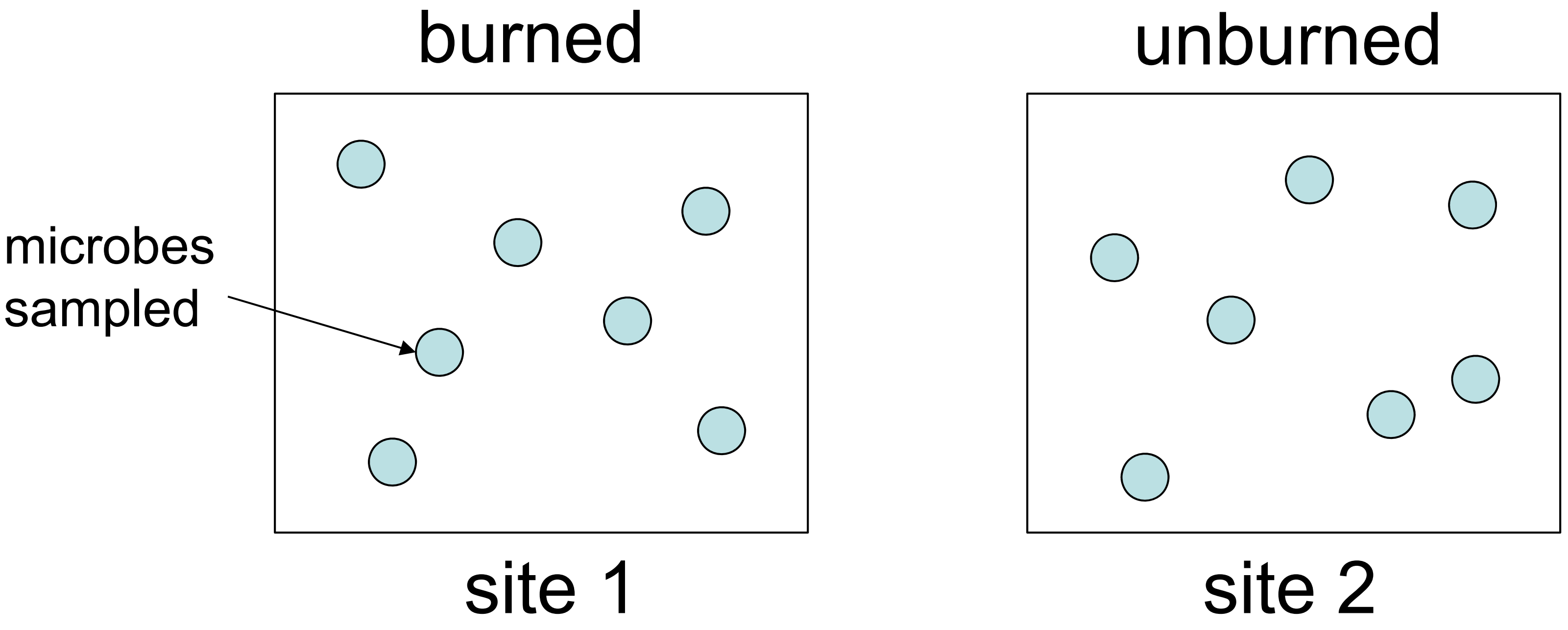
> To find out what happens when you change something,
> it is necessary to change it
>
> Box, Hunter, and Hunter (1978)

Design fundamentals

- Identify a population of inference: scope
- Identify sample or experimental unit
- Confounding — main issue
- Replication
- Randomization
- Control

main remedies

Confounding examples



burn and site are confounded

Confounding examples

Process all of treatment 1

before lunch

Process all of treatment 2

after lunch

What’s wrong?

Confounding examples

Process all of treatment 1	Process all of treatment 2
before lunch	after lunch
time 1 environment 1?	time 2 environment 2?

treatment and time are confounded

Confounding examples

Put all of treatment 1

left side of bench

Put all of treatment 2

right side of bench

What’s wrong?

Confounding examples

Put all of
treatment 1

left side of
bench

space 1
environment 1?

Put all of
treatment 2

right side of
bench

space 2
environment 2?

treatment and space are confounded

Replication

- How much replication?
  - depends on effect size and variance
  - rule of thumb:

    < 20 d.f. is treacherous

    > 100 d.f. is good (but unusual)
- Degrees of freedom (d.f.)

  = n – number of parameters
- Best to simulate designs

Pseudoreplication

- Replicates are grouped
- Grouping = confounding

Randomization

- Fixes confounding by shuffling potential confounders
- Random sampling: easiest inference to population (scope)
- Random assignment: allows causal inference about a treatment

Simple random sample

- Number each individual in the population
- Use a random number generator to draw individuals at random
- Unbiased sample
- Ensures unbiased estimate

Stratified random sample

- Divide the statistical population into sub-populations
- Random sample within sub-populations
- Examples
  - male/female
  - different habitat types
  - species 1 / species 2

Stratified random sample

Effects parameterization

or whatever

$$y_i \sim \mathrm{Normal}(\mu_i)$$

$$\mu_i = \beta_0 + \beta_1 x_{1,i}$$

species 2

species 1
(reference level)

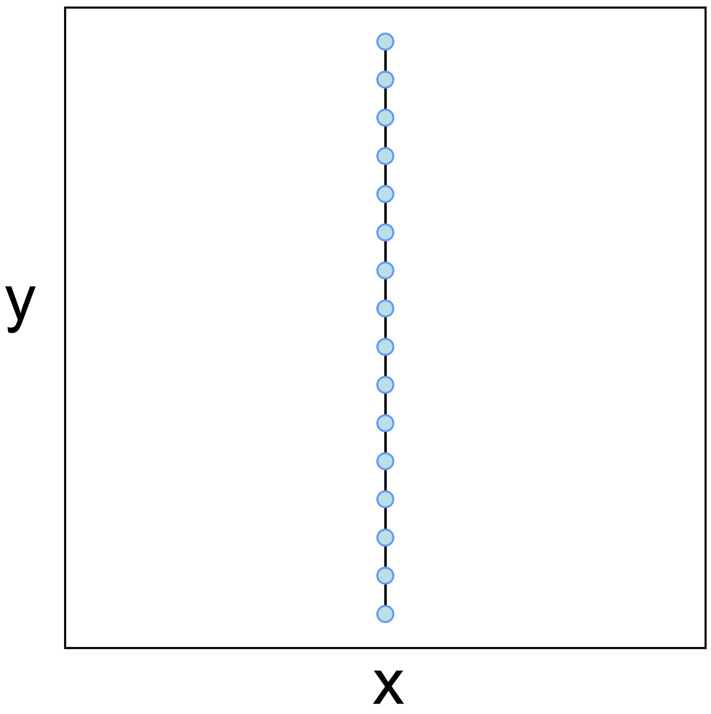
R code: `stan_lmer(y ~ species)`

Systematic sampling

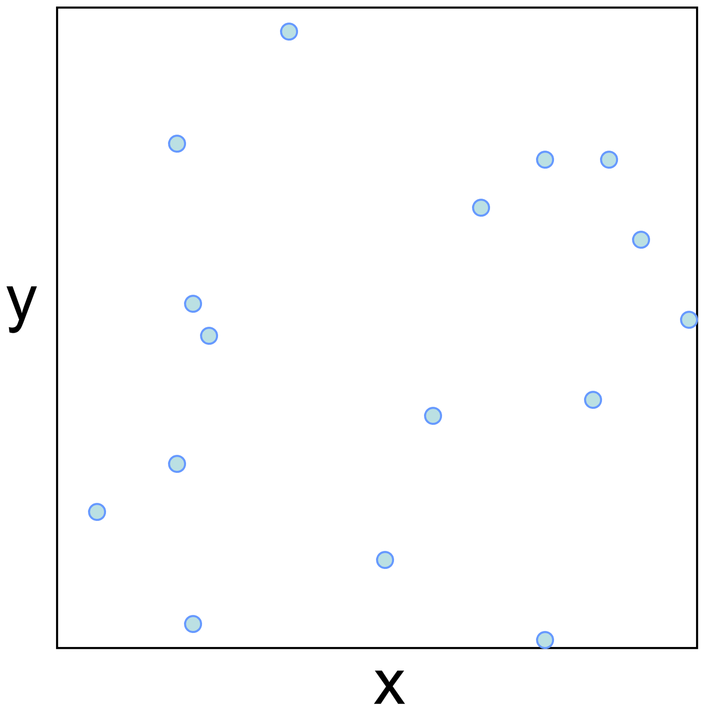
- Opposite of random
- Examples
  - transects with equal spacing of samples
  - spatial grid
  - every Thursday
- Bias
- Autocorrelation
- Scope

Example: spatial sample

Transect

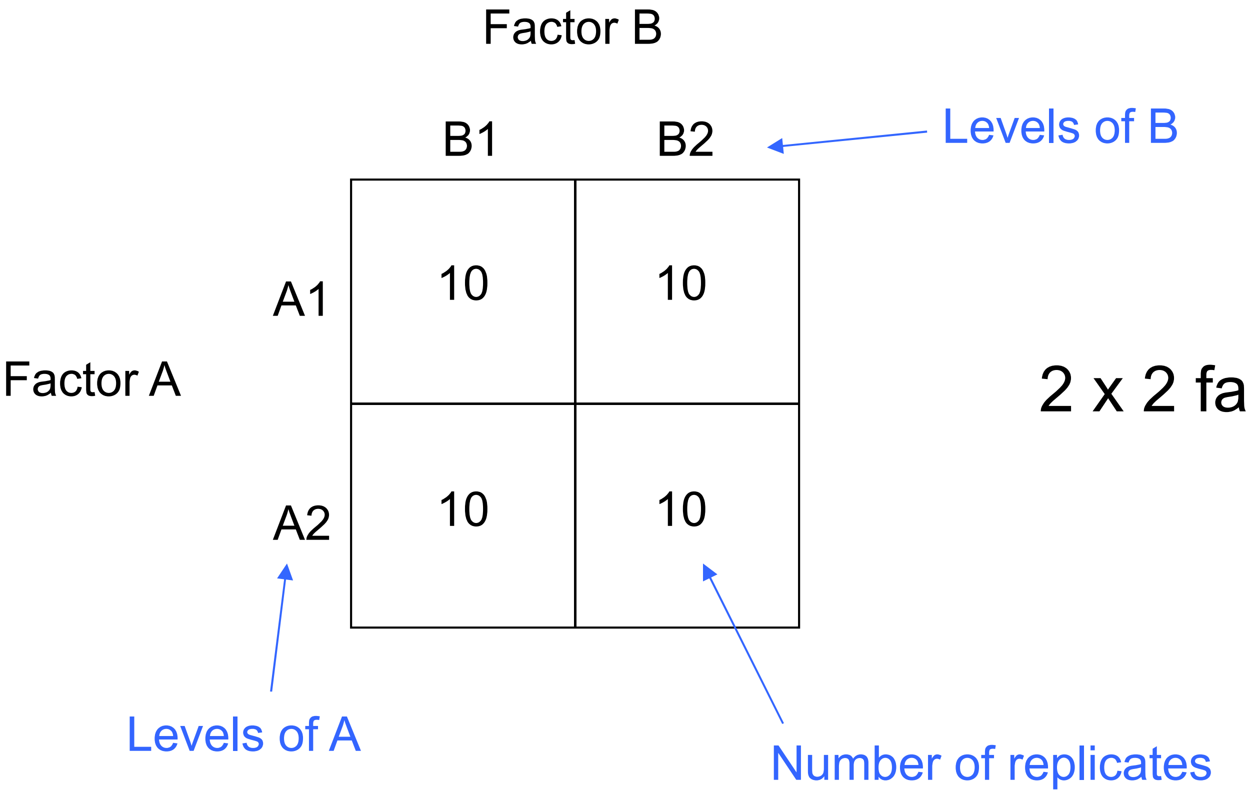


Simple random sample



Bias:	one x; gradient on y?	none
Autocorrelation:	strong, systematic	weak, diffuse
Scope:	this transect	population

Factorial design



		Factor B	
		B1	B2
Factor A	A1	10	10
	A2	10	10

Levels of B

Levels of A

Number of replicates

2 x 2 factorial

Advantage: allows us to estimate interactions

Factorial design

Effects parameterization
2 factors (A, B)

or whatever

$$y_i \sim \mathrm{Normal}(\mu_i)$$

Plot-level stochastic model

factor B2

$$\mu_i = \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i} + \beta_3 x_{1,i} x_{2,i}$$

factor A2

interaction
A2 B2

factor A1, B1
(reference or control level)

```
R code: stan_lmer(y ~ factor_A * factor_B)
```

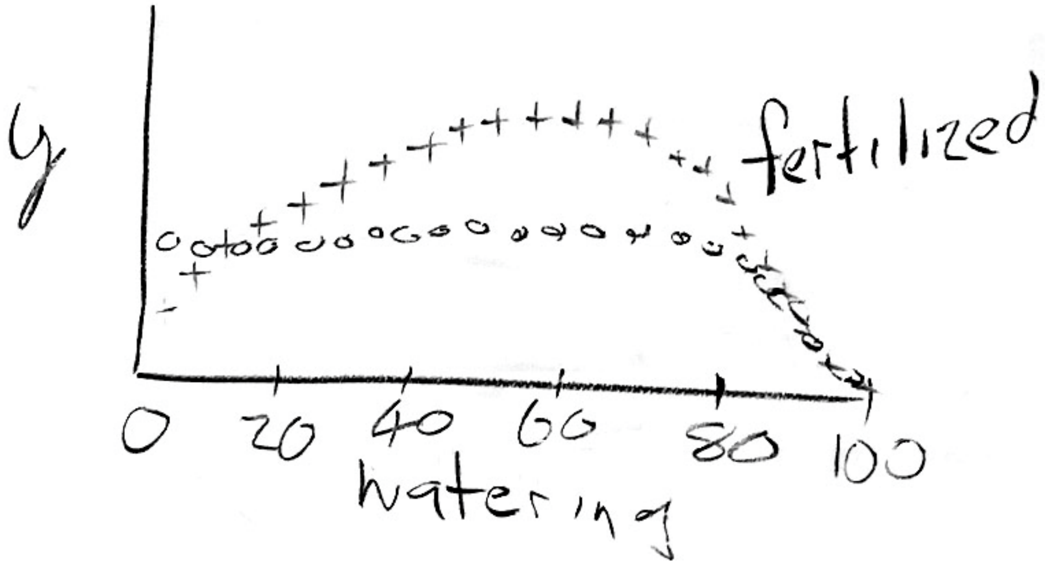
Factorial design

- Many possibilities
  - 2 x 2 x 2 = cube
  - 2 x 2 x 2 x 2
  - 3 x 2
  - 5 x 4
  - ...

Factorial versus response surface design

	Water				
Fertilize	20	40	60	80	100
+	5	5	5	5	5
-	5	5	5	5	5

50 experimental units
no interaction
# parameters = 7
df = 50 - 7 = 43
with interaction
# parameters = 11
df = 50 - 11 = 39

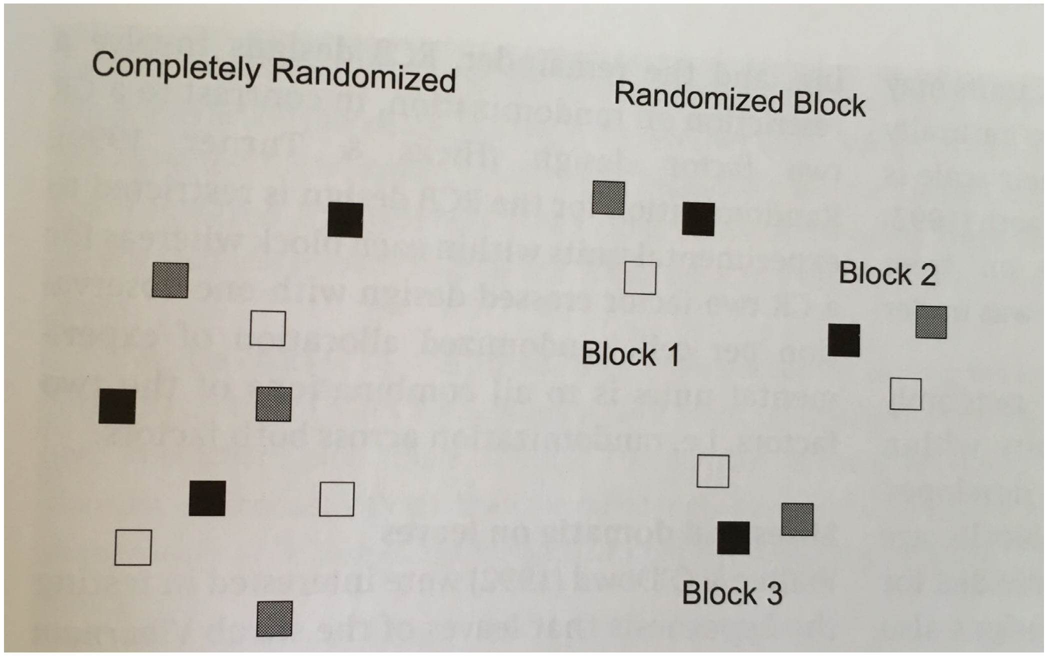


50 experimental units
3 parameters per curve
df = 50 - 7 = 43
5 parameters per curve
df = 50 - 11 = 39

Advantage: can get much better nonlinear resolution for same replication

Multilevel designs

- Randomized block



Example spatial design with three treatments (box colors)

Contrasted with completely randomized design

Pros: account for large scale variation
Cons: penalty for more complex model (grouping variable)
Whether it helps depends on this tradeoff

Randomized block

Effects parameterization
3 treatments

or whatever

$$y_i \sim \text{Normal}(\mu_i)$$

Plot-level stochastic model

treatment 3

$$\mu_i = \alpha_{j[i]} + \beta_1 x_{1,i} + \beta_2 x_{2,i}$$

treatment 2

$$\alpha_j \sim \text{Normal}(\mu_\alpha, \sigma_\alpha^2)$$

Block-level stochastic model

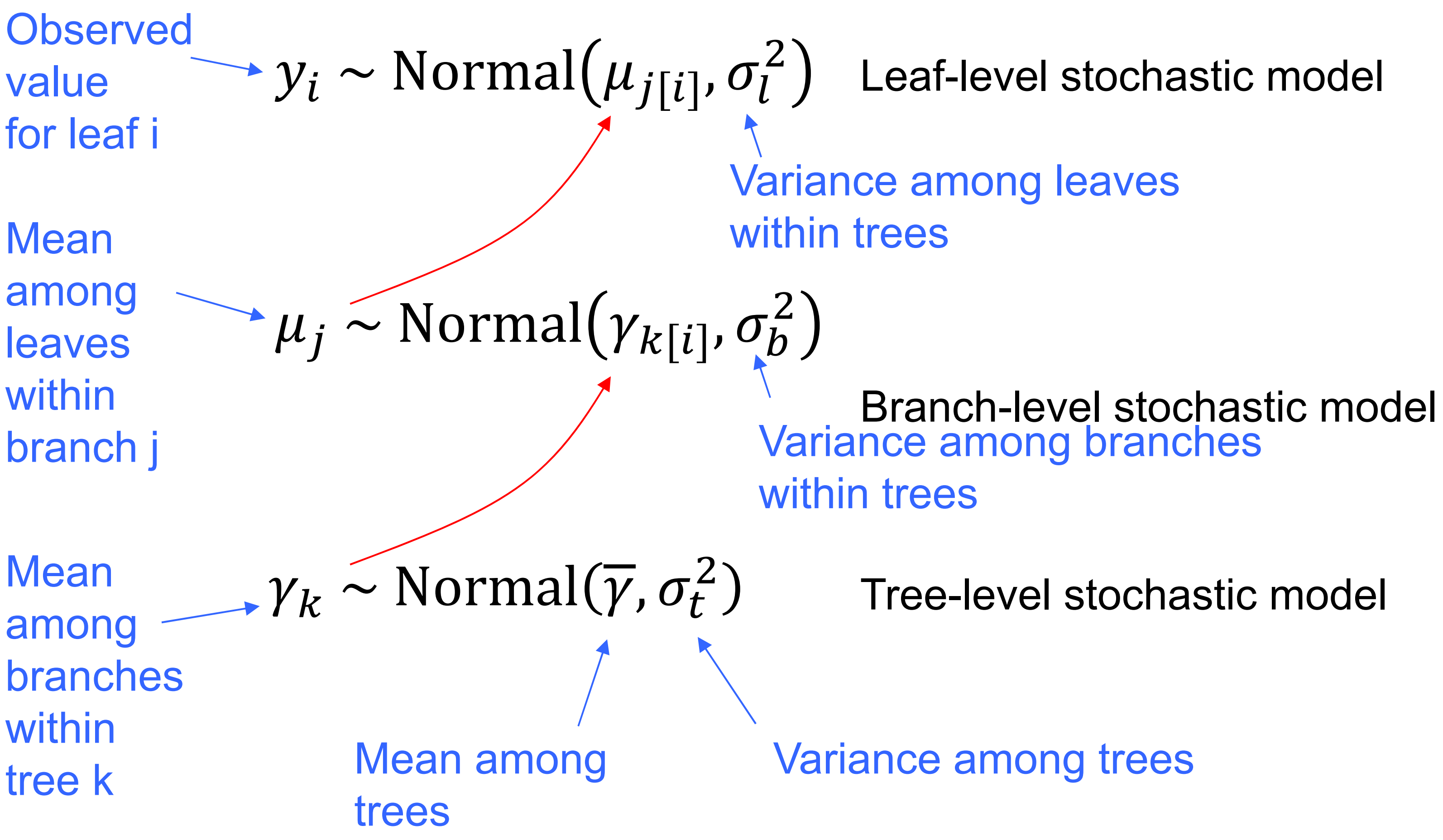
treatment 1
(reference or control)

R code: `stan_lmer(y ~ treatment + (1|block))`

Multilevel designs

- Nested random sample (example)
  - trees / branches / leaves
- Randomly sample trees within forest
- Randomly sample branches within trees
- Randomly sample leaves within branches
- Scope: leaves within a forest

Nested random sample



Observed value for leaf i

$$y_i \sim \text{Normal}\left(\mu_{j[i]}, \sigma_l^2\right)$$

Leaf-level stochastic model

Variance among leaves within trees

Mean among leaves within branch j

$$\mu_j \sim \text{Normal}\left(\gamma_{k[i]}, \sigma_b^2\right)$$

Branch-level stochastic model

Variance among branches within trees

Mean among branches within tree k

$$\gamma_k \sim \text{Normal}\left(\overline{\gamma}, \sigma_t^2\right)$$

Tree-level stochastic model

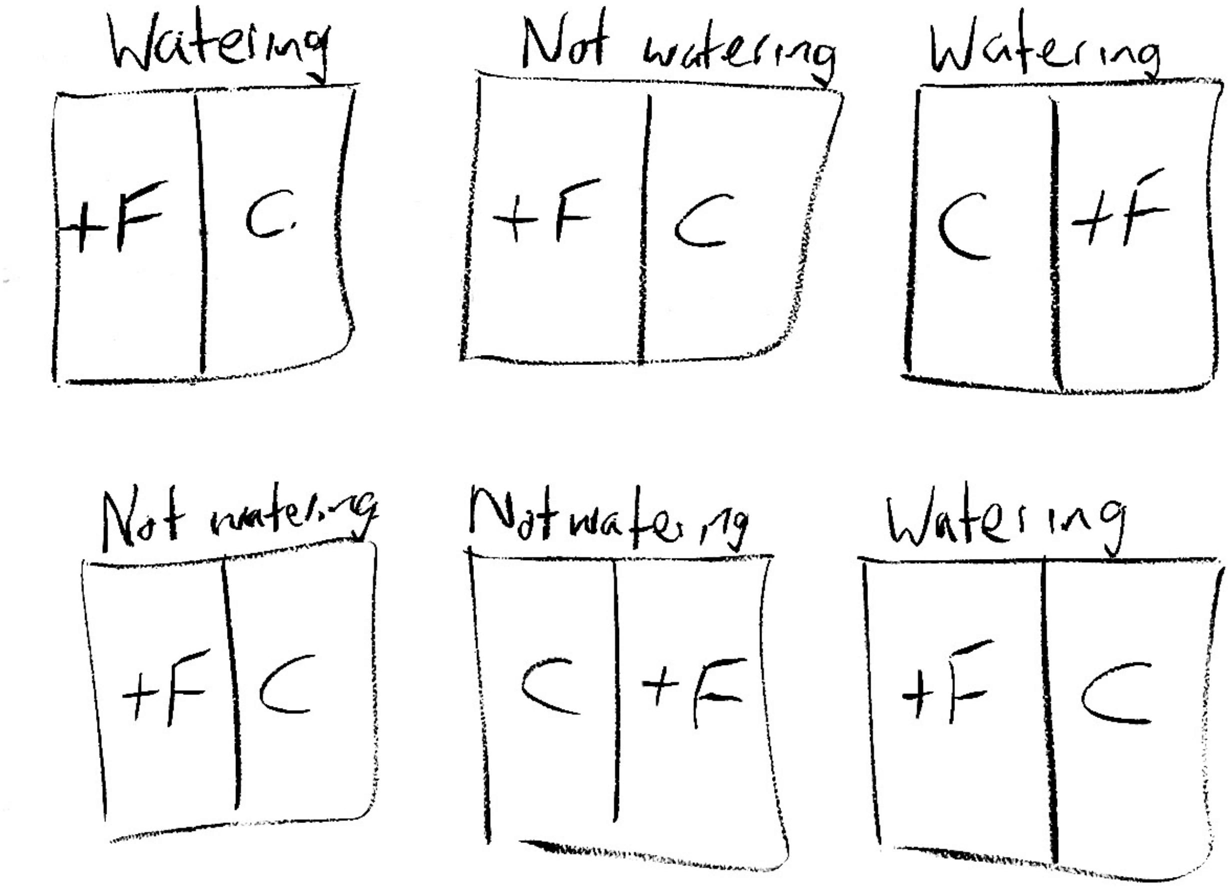
Mean among trees

Variance among trees

```
R code: stan_lmer(y ~ (1|tree) + (1|branch))
        stan_lmer(y ~ (1|tree/branch))
```

Multilevel designs

- Split plot experiment



Plots are split into sub-plots.

Watering treatment is at large scale (plot), fertilizer treatment is at small scale (sub-plot).

Pro: watering simpler
Con: replication of large scale factor is reduced (3)
Con: penalty for model complexity (need a grouping variable)

Split plot

Effects parameterization
Treatments at 2 scales

$$y_i \sim \mathrm{Normal}(\mu_i)$$

Sub-plot-level stochastic model

interaction

$$\ln(\mu_i) = \alpha_{j[i]} + \beta_1 x_{1,i} + \beta_3 x_{1,i} x_{2,j[i]}$$

fertilizer

$$\alpha_j \sim \mathrm{Normal}(\mu_\alpha, \sigma_\alpha^2)$$

Plot-level stochastic model

$$\mu_\alpha = \beta_0 + \beta_2\, x_{2,j}$$

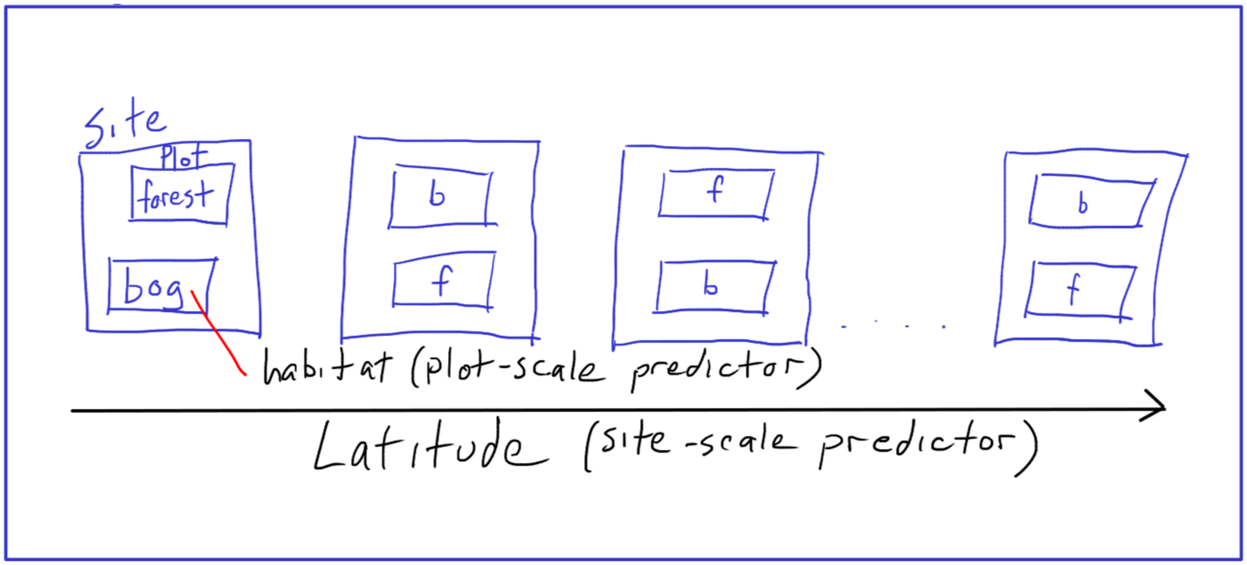
control
(no fertilizer or water)

watering

```
R code: stan_lmer(y ~ watering * fertilizer + (1|plot))
```

Multilevel designs

- Split plot (ants sampling)


Site
Plot
forest
bog
b
f
f
b
b
f
habitat (plot-scale predictor)
Latitude (site-scale predictor)

Sites (aka plots) are split into plots (aka sub-plots).

Latitude is at large scale (site), habitat is at small scale (plot).

Pro: travel simpler, control large scale var
Con: replication of large scale factor is reduced (22)
Con: penalty for model complexity (need a grouping variable)

Split plot - ants

Effects parameterization
Predictors at 2 scales

$$y_i \sim \text{Poisson}(\mu_i)$$

Plot-level stochastic model

interaction

$$\ln(\mu_i) = \alpha_{j[i]} + \beta_1 x_{1,i} + \beta_3 x_{1,i} x_{2,j[i]} + e_i$$

forest (habitat)

overdispersion

$$\alpha_j \sim \text{Normal}(\mu_\alpha, \sigma_\alpha^2)$$

Site-level stochastic model

$$\mu_\alpha = \beta_0 + \beta_2\, x_{2,j}$$

bog
(intercept)

latitude

$$e_i \sim \text{Normal}(0, \sigma_e^2)$$

```
R code: stan_lmer(y ~ habitat * latitude + (1|site/unit))
```