08_Class_Activity

Bill Perry

In class activity 8: Study Design and Power Analysis

Introduction

This document demonstrates key concepts in experimental design using ecological examples, focusing on:

1. **Formulating research questions**
2. **Understanding different study designs**
3. **Recognizing proper replication vs. pseudoreplication**
4. **Designing appropriate controls**
5. **Conducting power analysis** (a priori and post hoc)
6. **Planning sampling strategies**

We'll work with simulated pine needle data to practice these concepts.

Let's start by exploring these concepts with hands-on examples!

Part 1: Load Required Packages

```
# Load required packages
library(tidyverse)  # For data manipulation and visualization
library(patchwork)  # For combining plots
library(pwr)        # For power analysis

# Set seed for reproducible results
set.seed(42)
```

> Package Overview
>
> - **tidyverse**: Collection of packages for data science (includes ggplot2, dplyr, etc.)
> - **patchwork**: Easily combine multiple ggplot2 plots
> - **pwr**: Functions for power analysis and sample size calculations

Part 2: Formulating Research Questions

Before we design any study, we need clear research questions. Let's practice with pine needle ecology.

! Activity 1: Research Question Practice

Think about pine trees on campus. Write down 2-3 specific research questions about:

- - Pine needle characteristics (length, density, color)
- - Environmental factors (wind, sunlight, soil)
- - Tree health or growth

Example questions:

- - Does wind exposure affect pine needle length?
- - Do pine needles on south-facing branches differ from north-facing branches?
- - Does tree size influence needle density?

Your questions:

1. 1. ________________________________
2. 2. ________________________________
3. 3. ________________________________

Part 3: Understanding Study Design Types

Let's simulate data for different types of studies to understand their strengths and limitations.

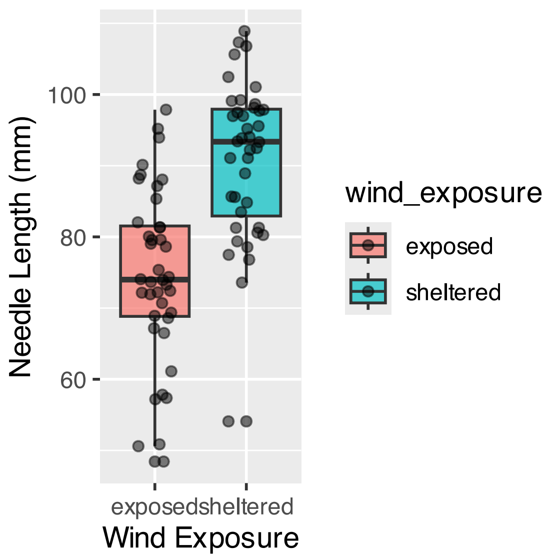
Natural Experiment: Wind Exposure Study

```
# Simulate pine needle data from naturally exposed and sheltered locations
# This represents a "natural experiment" - we didn't manipulate wind exposure

# Create data for exposed locations (shorter needles due to wind stress)
exposed_data <- data.frame(
  location = rep(paste0("Exposed_", 1:5), each = 8),
  wind_exposure = "exposed",
  needle_length_mm = rnorm(40, mean = 75, sd = 10),
  tree_id = rep(1:5, each = 8)
)

# Create data for sheltered locations (longer needles, less wind stress)
sheltered_data <- data.frame(
  location = rep(paste0("Sheltered_", 1:5), each = 8),
  wind_exposure = "sheltered",
  needle_length_mm = rnorm(40, mean = 90, sd = 12),
  tree_id = rep(6:10, each = 8)
)

# Combine the datasets
natural_exp_data <- rbind(exposed_data, sheltered_data)

# Look at the first few rows
head(natural_exp_data)
```

```
   location wind_exposure needle_length_mm tree_id
1 Exposed_1       exposed         88.70958       1
2 Exposed_1       exposed         69.35302       1
```

```
3 Exposed_1       exposed         78.63128       1
4 Exposed_1       exposed         81.32863       1
5 Exposed_1       exposed         79.04268       1
6 Exposed_1       exposed         73.93875       1
```

```
# Visualize the natural experiment data
natural_plot <- natural_exp_data %>%
  ggplot(aes(x = wind_exposure, y = needle_length_mm, fill = wind_exposure)) +
  geom_boxplot(alpha = 0.7) +
  geom_jitter(width = 0.2, alpha = 0.5) +
  labs(
       x = "Wind Exposure",
       y = "Needle Length (mm)")
natural_plot
```



ℹ Natural Experiments: Pros and Cons

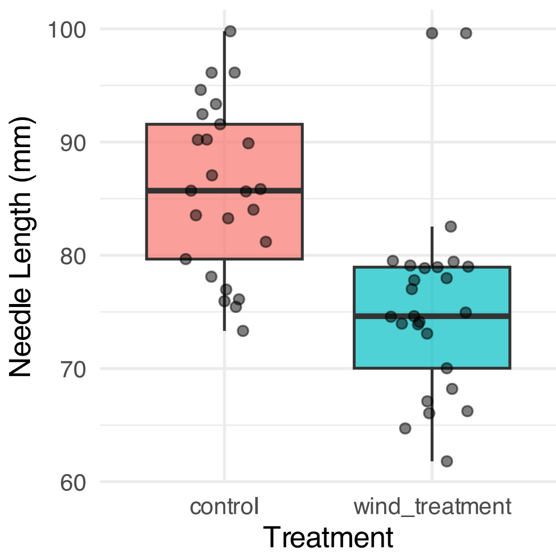
- **Advantages:** - Realistic conditions - Large scale possible - Cost-effective
- **Disadvantages:** - Cannot control confounding variables - Cannot determine causation direction - Many unmeasured factors might influence results

Question: What other factors besides wind might differ between "exposed" and "sheltered" locations?

Manipulative Experiment: Controlled Wind Study

```
# Simulate a controlled experiment where we manipulate wind exposure
# All trees start similar, then we apply treatments

# Create data for control group (normal conditions)
control_data <- data.frame(
  treatment = "control",
  needle_length_mm = rnorm(25, mean = 85, sd = 8),
  tree_id = 1:25
)
```

```r
# Create data for wind treatment (artificial wind exposure)
wind_treatment_data <- data.frame(
  treatment = "wind_treatment",
  needle_length_mm = rnorm(25, mean = 78, sd = 8),
  tree_id = 26:50
)

# Combine the datasets
manipulative_data <- rbind(control_data, wind_treatment_data)

# Visualize the manipulative experiment
manipulative_plot <- manipulative_data %>%
  ggplot(aes(x = treatment, y = needle_length_mm, fill = treatment)) +
  geom_boxplot(alpha = 0.7) +
  geom_jitter(width = 0.2, alpha = 0.5) +
  labs(
       x = "Treatment",
       y = "Needle Length (mm)") +
  theme_minimal() +
  theme(legend.position = "none")

manipulative_plot
```



💡 Manipulative Experiments: Key Features

Advantages: - Can establish causation - Control confounding variables - Random assignment eliminates bias

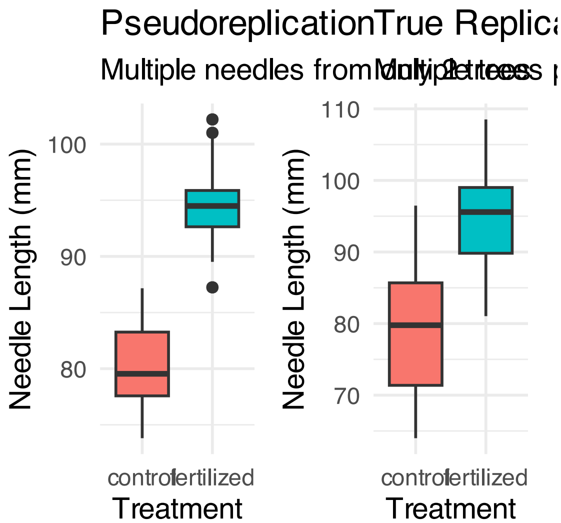
Disadvantages: - Often smaller scale - May not reflect natural conditions - Can be expensive and logistically challenging

Key Question: Which experiment gives stronger evidence for causation?

Part 4: Identifying Proper Replication

One of the most common mistakes in ecological studies is pseudoreplication. Let's practice identifying true replication vs. pseudoreplication.

```r
# Example 1: Pseudoreplication - multiple measurements from same trees
pseudo_data <- data.frame(
  treatment = rep(c("fertilized", "control"), each = 20),
  tree_id = rep(c("Tree_A", "Tree_B"), each = 20),  # Only 2 trees total!
  needle_length_mm = c(
    rnorm(20, mean = 95, sd = 5),  # Tree A (fertilized)
    rnorm(20, mean = 80, sd = 5)   # Tree B (control)
  ),
  measurement = rep(1:20, times = 2)
)

# Example 2: True replication - multiple trees per treatment
true_rep_data <- data.frame(
  treatment = rep(c("fertilized", "control"), each = 20),
  tree_id = paste0("Tree_", 1:40),  # 40 different trees
  needle_length_mm = c(
    rnorm(20, mean = 95, sd = 8),  # 20 fertilized trees
    rnorm(20, mean = 80, sd = 8)   # 20 control trees
  )
)

# Create comparison plots
pseudo_plot <- pseudo_data %>%
  ggplot(aes(x = treatment, y = needle_length_mm, fill = treatment)) +
  geom_boxplot() +
  labs(title = "Pseudoreplication",
       subtitle = "Multiple needles from only 2 trees",
       x = "Treatment", y = "Needle Length (mm)") +
  theme_minimal() +
  theme(legend.position = "none")

true_plot <- true_rep_data %>%
  ggplot(aes(x = treatment, y = needle_length_mm, fill = treatment)) +
  geom_boxplot() +
  labs(title = "True Replication",
       subtitle = "Multiple trees per treatment",
       x = "Treatment", y = "Needle Length (mm)") +
  theme_minimal() +
  theme(legend.position = "none")

# Combine plots
pseudo_plot + true_plot
```



⚠ Pseudoreplication Alert!

Pseudoreplication occurs when:

- - You treat subsamples as independent when they're not
- - Multiple measurements from the same experimental unit
- - Replication at wrong scale for your hypothesis

Common examples:

- - Multiple leaves from one plant
- - Multiple samples from one lake or from one fish
- - Multiple plots within one treatment area

Why it's bad:

- - Underestimates variability
- - Inflates sample size artificially
- - Increases Type I error (false positives)

Activity: Identify Replication Issues

! Activity 2: Replication Practice

For each scenario, identify if there's proper replication or pseudoreplication:

Scenario A: Testing fertilizer effects by using 1 large pot with fertilizer containing 10 pine seedlings, and 1 control pot with 10 seedlings.

- **Your answer:** ___________________

- **Fix:** ____________________________

Scenario B: Testing altitude effects by measuring needle length on 5 trees at 1000m elevation and 5 trees at 2000m elevation.

- **Your answer:** ___________________

- **Fix:** ____________________________

Scenario C: Testing soil pH by measuring 20 needles each from 10 trees in acidic soil and 10 trees in basic soil.

- **Your answer:** ___________________

- **Fix:** ____________________________

Part 5: Power Analysis - Planning Your Study

Power analysis helps us determine how many samples we need to detect an effect if it really exists.

A Priori Power Analysis (Before Data Collection)

```
# Scenario: We want to detect a difference in needle length between
# fertilized and control trees

# Based on pilot data, we expect:
control_mean <- 80      # mm
fertilized_mean <- 90   # mm
pooled_sd <- 12         # mm

# Calculate effect size (Cohen's d)
effect_size <- abs(fertilized_mean - control_mean) / pooled_sd
cat("Effect size (Cohen's d):", round(effect_size, 2), "\n")
```

```
Effect size (Cohen's d): 0.83
```

```
# Interpret effect size
if(effect_size < 0.2) {
  interpretation <- "small"
} else if(effect_size < 0.5) {
  interpretation <- "small-medium"
} else if(effect_size < 0.8) {
  interpretation <- "medium-large"
} else {
  interpretation <- "large"
}
cat("This is a", interpretation, "effect size\n")
```

```
This is a large effect size
```

```
# Calculate required sample size for 80% power
power_result <- pwr.t.test(
  d = effect_size,            # Effect size
  sig.level = 0.05,          # Alpha level (significance)
  power = 0.8,               # Desired power (80%)
  type = "two.sample"        # Two-sample t-test
)

print(power_result)
```

```

     Two-sample t test power calculation

              n = 23.60467
              d = 0.8333333
      sig.level = 0.05
          power = 0.8
    alternative = two.sided

NOTE: n is number in *each* group
```

```
cat("\nWe need", ceiling(power_result$n), "trees per group for 80% power\n")
```

```

We need 24 trees per group for 80% power
```

Understanding Effect Size (Cohen's d)

- **d = 0.2**: Small effect (subtle difference)
- **d = 0.5**: Medium effect (moderate difference)
- **d = 0.8**: Large effect (substantial difference)

Cohen's d formula: $d = (Mean_1 - Mean_2) / \text{Pooled Standard Deviation}$

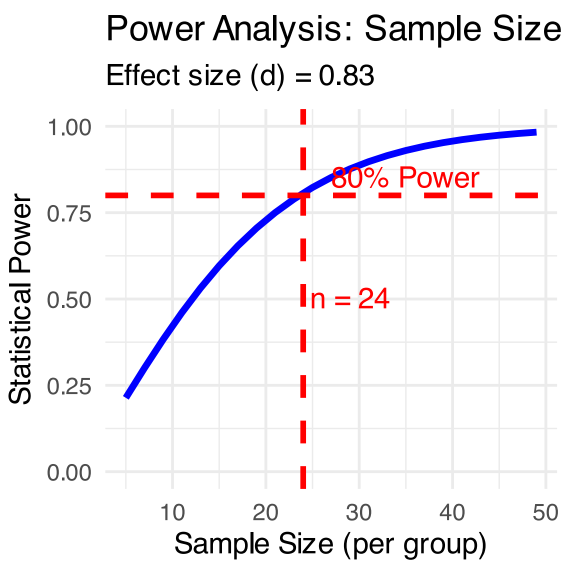
Visualizing Power Curves

```
# Create a power curve showing relationship between sample size and power
sample_sizes <- seq(5, 50, by = 2)

# Calculate power for each sample size
power_values <- sapply(sample_sizes, function(n) {
  power_test <- pwr.t.test(n = n, d = effect_size, sig.level = 0.05, type = "two.sample")
  return(power_test$power)
})

# Create data frame for plotting
power_df <- data.frame(
  sample_size = sample_sizes,
  power = power_values
```

```
)

# Create power curve plot
power_curve_plot <- ggplot(power_df, aes(x = sample_size, y = power)) +
  geom_line(color = "blue", size = 1.2) +
  geom_hline(yintercept = 0.8, linetype = "dashed", color = "red", size = 1) +
   geom_vline(xintercept = ceiling(power_result$n), linetype = "dashed", color = "red", size
= 1) +
  annotate("text", x = ceiling(power_result$n) + 5, y = 0.5,
           label = paste("n =", ceiling(power_result$n)), color = "red") +
  annotate("text", x = 35, y = 0.85, label = "80% Power", color = "red") +
  ylim(0, 1) +
  labs(title = "Power Analysis: Sample Size vs. Statistical Power",
       subtitle = paste("Effect size (d) =", round(effect_size, 2)),
       x = "Sample Size (per group)",
       y = "Statistical Power") +
  theme_minimal()
```

```
Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
i Please use `linewidth` instead.
```

```
power_curve_plot
```



Post Hoc Power Analysis (After Data Collection)

```
# Imagine we collected data with n = 15 per group but found no significant difference
# Was our study adequately powered?

observed_n <- 15

# Calculate the power we actually had
actual_power <- pwr.t.test(
  n = observed_n,
  d = effect_size,
  sig.level = 0.05,
  type = "two.sample"
```

```r
)

print(actual_power)
```

```
     Two-sample t test power calculation 

              n = 15
              d = 0.8333333
      sig.level = 0.05
          power = 0.5962064
    alternative = two.sided

NOTE: n is number in *each* group
```

```r
cat("\nWith n =", observed_n, "per group, we only had",
    round(actual_power$power * 100, 1), "% power\n")
```

```

With n = 15 per group, we only had 59.6 % power
```

```r
if(actual_power$power < 0.8) {
   cat("This study was underpowered! A non-significant result might be due to insufficient
sample size.\n")
} else {
  cat("This study had adequate power. A non-significant result likely reflects no true effect.
\n")
}
```

```
This study was underpowered! A non-significant result might be due to insufficient sample size.
```

! Activity 3: Power Analysis Practice

Scenario: You want to study the effect of drought stress on pine needle length. Based on literature, you expect:

- - Control trees: mean = 85mm, SD = 10mm
- - Drought-stressed trees: mean = 75mm, SD = 10mm

Calculate the following:

```
# Your turn! Fill in the values and run the code

# Step 1: Calculate effect size
control_mean <-     9
drought_mean <-     99
pooled_sd <-        999

effect_size <- abs(control_mean - drought_mean) / pooled_sd
print(paste("Effect size:", round(effect_size, 2)))

# Step 2: Calculate required sample size for 80% power
power_result <- pwr.t.test(
  d = effect_size,
  sig.level = 0.05,
  power = 0.8,
  type = "two.sample"
)

print(power_result)
print(paste("Required sample size:", ceiling(power_result$n), "trees per group"))

# Step 3: What if you can only collect 12 trees per group?
limited_power <- pwr.t.test(
  n = 12,
  d = effect_size,
  sig.level = 0.05,
  type = "two.sample"
)

print(paste("Power with n=12:", round(limited_power$power * 100, 1), "%"))
```

Questions: 1. What is the effect size for this drought study? 2. How many trees do you need per group for 80% power? 3. If you can only sample 12 trees per group, what power will you have?

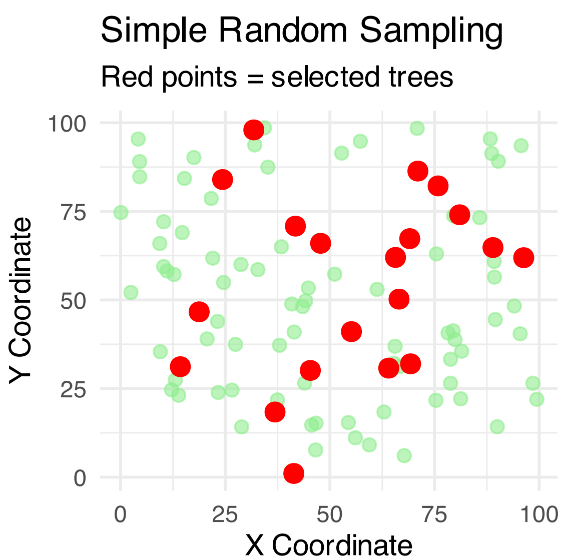
Part 6: Sampling Design Strategies

Different research questions require different sampling approaches. Let's explore the main types.

Simple Random Sampling

```
# Simulate a campus with pine trees scattered randomly
set.seed(123)
campus_trees <- data.frame(
  tree_id = 1:100,
  x_coordinate = runif(100, 0, 100),  # Random x positions
  y_coordinate = runif(100, 0, 100),  # Random y positions
  needle_length = rnorm(100, mean = 80, sd = 12)
```

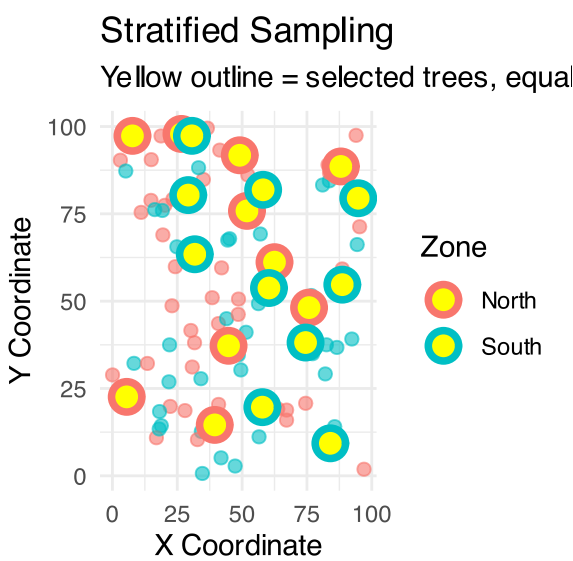
```r
)

# Simple random sampling: select 20 trees randomly
random_sample_ids <- sample(1:100, size = 20, replace = FALSE)
random_sample <- campus_trees[campus_trees$tree_id %in% random_sample_ids, ]

# Visualize sampling design
campus_plot <- ggplot(campus_trees, aes(x = x_coordinate, y = y_coordinate)) +
  geom_point(color = "lightgreen", size = 2, alpha = 0.6) +
  geom_point(data = random_sample, color = "red", size = 3) +
  labs(title = "Simple Random Sampling",
       subtitle = "Red points = selected trees",
       x = "X Coordinate", y = "Y Coordinate") +
  theme_minimal()

campus_plot
```



Stratified Sampling

```r
# Simulate campus with different zones (north vs south)
set.seed(124)
stratified_trees <- data.frame(
  tree_id = 1:100,
  x_coordinate = runif(100, 0, 100),
  y_coordinate = runif(100, 0, 100),
  zone = ifelse(runif(100) > 0.5, "North", "South"),
  needle_length = rnorm(100, mean = 80, sd = 12)
)

# Add zone effect to needle length
stratified_trees$needle_length[stratified_trees$zone == "South"] <-
  stratified_trees$needle_length[stratified_trees$zone == "South"] + 8

# Stratified sampling: sample equally from each zone
north_trees <- stratified_trees[stratified_trees$zone == "North", ]
south_trees <- stratified_trees[stratified_trees$zone == "South", ]

# Sample 10 from each zone
```

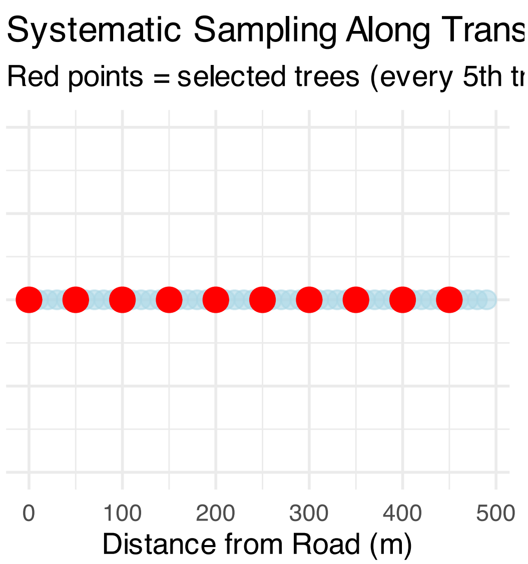
```r
north_sample <- north_trees[sample(nrow(north_trees), 10), ]
south_sample <- south_trees[sample(nrow(south_trees), 10), ]
stratified_sample <- rbind(north_sample, south_sample)

# Visualize stratified sampling
stratified_plot <- ggplot(stratified_trees, aes(x = x_coordinate, y = y_coordinate, color =
zone)) +
  geom_point(size = 2, alpha = 0.6) +
  geom_point(data = stratified_sample, size = 4, shape = 21, fill = "yellow", stroke = 2) +
  labs(title = "Stratified Sampling",
       subtitle = "Yellow outline = selected trees, equal sampling from each zone",
       x = "X Coordinate", y = "Y Coordinate", color = "Zone") +
  theme_minimal()

stratified_plot
```



Systematic Sampling

```r
# Systematic sampling along a transect
set.seed(125)
transect_trees <- data.frame(
  tree_id = 1:50,
  distance_m = seq(0, 490, by = 10),  # Trees every 10m along transect
  needle_length = rnorm(50, mean = 80, sd = 10)
)

# Add distance effect (trees farther from road have longer needles)
transect_trees$needle_length <- transect_trees$needle_length +
  (transect_trees$distance_m * 0.02)

# Systematic sampling: every 5th tree
systematic_sample <- transect_trees[seq(1, 50, by = 5), ]

# Visualize systematic sampling
systematic_plot <- ggplot(transect_trees, aes(x = distance_m, y = 1)) +
  geom_point(size = 3, alpha = 0.6, color = "lightblue") +
  geom_point(data = systematic_sample, size = 4, color = "red") +
  labs(title = "Systematic Sampling Along Transect",
```

```r
      subtitle = "Red points = selected trees (every 5th tree)",
      x = "Distance from Road (m)", y = "") +
  theme_minimal() +
  theme(axis.text.y = element_blank(), axis.ticks.y = element_blank()) +
  ylim(0.5, 1.5)

systematic_plot
```



Sampling Strategy Comparison

Simple Random Sampling:

- - Best for: General population estimates
- - Pros: Unbiased, simple analysis
- - Cons: May miss important subgroups

Stratified Sampling:

- - Best for: When you know there are distinct subgroups
- - Pros: Ensures representation of all strata
- - Cons: Requires prior knowledge of strata

Systematic Sampling:

- - Best for: Studying gradients or patterns
- - Pros: Good spatial coverage, easy to implement
- - Cons: Risk of bias if there's hidden periodicity

Part 7: Putting It All Together - Design Your Own Study

! Activity 4: Complete Study Design

Research Question: Does fertilizer application affect pine needle length?

Design your study by answering these questions:

1. **Study Type:** Will this be a natural experiment or manipulative experiment? Why?
   - Your answer: ________________________________
2. **Sample Size:** Using the following parameters, calculate required sample size:
   - Expected control mean: 80mm
   - Expected fertilized mean: 88mm
   - Expected SD for both groups: 10mm
   - Desired power: 80%

```
# Calculate effect size and sample size needed
control_mean <- 80
fertilized_mean <- 88
pooled_sd <- 10

effect_size <- abs(fertilized_mean - control_mean) / pooled_sd

power_result <- pwr.t.test(
  d = effect_size,
  sig.level = 0.05,
  power = 0.8,
  type = "two.sample"
)

print(power_result)
```

3. **Controls:** What controls will you include? Consider both positive and negative controls.
   - Your answer: ________________________________
4. **Randomization:** How will you randomize tree assignment to treatments?
   - Your answer: ________________________________
5. **Replication:** How will you ensure proper replication? What would be pseudoreplication?
   - Proper replication: __________________________
   - Pseudoreplication to avoid: ___________________
6. **Independence:** What factors might violate independence? How will you address them?
   - Your answer: ________________________________
7. **Potential Confounds:** What other variables might affect needle length that you need to control for?
   - Your answer: ________________________________

Part 8: Analyzing Your Designed Study

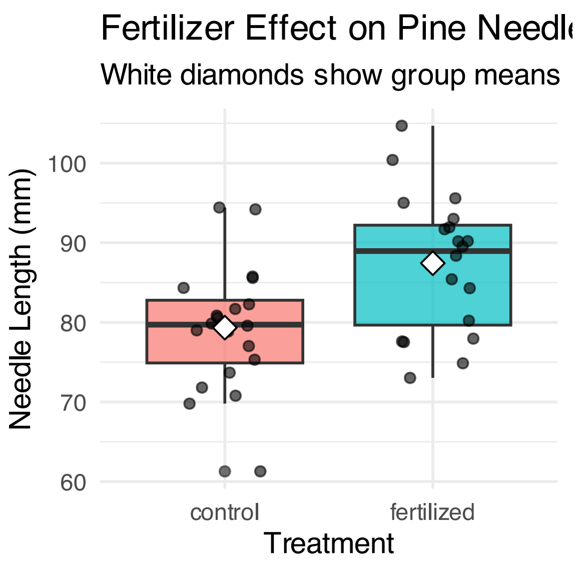
Let's simulate data from the study you designed and analyze it:

```
# Simulate data based on your study design
set.seed(200)

# Use the sample size you calculated (or use 20 if you didn't calculate)
n_per_group <- 20  # Replace with your calculated sample size

# Create the experimental data
```

```r
study_data <- data.frame(
  tree_id = 1:(2 * n_per_group),
  treatment = rep(c("control", "fertilized"), each = n_per_group),
  needle_length_mm = c(
    rnorm(n_per_group, mean = 80, sd = 10),  # Control group
    rnorm(n_per_group, mean = 88, sd = 10)   # Fertilized group
  )
)

# Calculate summary statistics
summary_stats <- study_data %>%
  group_by(treatment) %>%
  summarise(
    n = n(),
    mean_length = mean(needle_length_mm),
    sd_length = sd(needle_length_mm),
    se_length = sd_length / sqrt(n)
  )

print(summary_stats)
```

```
# A tibble: 2 × 5
  treatment      n mean_length sd_length se_length
  <chr>      <int>       <dbl>     <dbl>     <dbl>
1 control       20        79.3      7.85      1.76
2 fertilized    20        87.4      8.57      1.92
```

```r
# Create visualization
study_plot <- study_data %>%
  ggplot(aes(x = treatment, y = needle_length_mm, fill = treatment)) +
  geom_boxplot(alpha = 0.7) +
  geom_jitter(width = 0.2, alpha = 0.6) +
  stat_summary(fun = mean, geom = "point", shape = 23, size = 3, fill = "white") +
  labs(title = "Fertilizer Effect on Pine Needle Length",
       subtitle = "White diamonds show group means",
       x = "Treatment",
       y = "Needle Length (mm)") +
  theme_minimal() +
  theme(legend.position = "none")

study_plot
```




```
# Conduct statistical test
t_test_result <- t.test(needle_length_mm ~ treatment, data = study_data)
print(t_test_result)
```

```

    Welch Two Sample t-test

data:  needle_length_mm by treatment
t = -3.1164, df = 37.715, p-value = 0.003493
alternative hypothesis: true difference in means between group control and group fertilized
is not equal to 0
95 percent confidence interval:
 -13.364541  -2.837295
sample estimates:
   mean in group control mean in group fertilized
                79.33243                 87.43335
```

```
# Interpret results
if(t_test_result$p.value < 0.05) {
  cat("\nResult: Significant difference found!\n")
  cat("Fertilizer significantly affects needle length (p =",
      round(t_test_result$p.value, 4), ")\n")
} else {
  cat("\nResult: No significant difference found.\n")
  cat("No evidence that fertilizer affects needle length (p =",
      round(t_test_result$p.value, 4), ")\n")
}
```

```

Result: Significant difference found!
Fertilizer significantly affects needle length (p = 0.0035 )
```

```
# Calculate actual effect size observed
observed_effect_size <- abs(diff(t_test_result$estimate)) /
  sqrt(((n_per_group-1) * var(study_data$needle_length_mm[study_data$treatment == "control"])
```

```
+
                (n_per_group-1)  *  var(study_data$needle_length_mm[study_data$treatment  ==
"fertilized"])) /
         (2*n_per_group - 2))

cat("Observed effect size (Cohen's d):", round(observed_effect_size, 2), "\n")
```

```
Observed effect size (Cohen's d): 0.99
```

Summary and Key Takeaways

What We Learned Today

1. **Study Design Matters:** Statistics cannot fix a poorly designed study
2. **Replication:** Must be at the appropriate scale for your research question
3. **Controls:** Essential for ruling out alternative explanations
4. **Power Analysis:** Plan your sample size before collecting data
5. **Sampling Strategy:** Choose the approach that best fits your research question
6. **Integration:** Good analysis flows naturally from good design

Remember:

- - Design before you collect data
- - Consider practical and logistical constraints
- - Be transparent about limitations
- - Correlation ≠ causation (especially in natural experiments)

Common Pitfalls to Avoid

1. **Pseudoreplication:** Taking multiple measurements from the same experimental unit
2. **Inadequate Power:** Collecting too few samples to detect meaningful effects
3. **Poor Controls:** Not controlling for important confounding variables
4. **Non-random Sampling:** Introducing bias through convenience sampling
5. **HARKing:** Hypothesizing After Results are Known

The Golden Rule: Plan your analysis when you plan your experiment!