Power and Sample Size for Proportions

Not covered in textbook

Emile Latour

March 2, 2026

Learning Objectives

By the end of today's lecture, you will be able to:

1. Recall and apply the four components of power analysis
2. Explain how effect size for proportions differs from Cohen's *d* for means
3. Calculate power and sample size for a **single proportion** using `pwr.p.test()`
4. Calculate power and sample size for **two independent proportions** using `pwr.2p.test()`
5. Explain when **correlated (paired) proportions** arise in biomedical research
6. Conduct McNemar's test in R and interpret results
7. Describe the key inputs needed to estimate power for paired proportion designs

Roadmap for Today

Part 1: Connecting Back to What We Know

- The four components, revisited
- What changes when outcomes are binary?
- Effect size for proportions: Cohen's *h*
- Using the `pwr` package for proportions

Part 2: Power for a Single Proportion

- One-sample proportion test recap
- The melanoma immunotherapy example
- Using `pwr.p.test()`
- Interpreting results

Part 3: Power for Two Independent Proportions

- Two-proportion test recap
- A new treatment comparison example
- Using `pwr.2p.test()`
- Sensitivity analysis: varying assumptions

Part 4: Correlated Proportions and McNemar's Test

- When observations are paired
- McNemar's test: the idea
- Running McNemar's test in R
- Power considerations for paired proportions

Part 1: Connecting Back to What We Know

Recall: The four components of power

From Lesson 12, every power calculation involves four quantities in equilibrium:

The Four Components

1. **Significance level** (α) — usually 0.05
2. **Power** ($1 - \beta$) — usually 80–90%
3. **Sample size** (n) — what we typically solve for
4. **Effect size** (Δ) — a property of reality

The key rule: Specify any 3 to solve for the 4th.

Study type	What we solve for
Prospective	Sample size (n)
Pilot/budget-limited	Effect size (Δ)
Retrospective	Power ($1 - \beta$)

What changed in Lessons 13–14?

In Lesson 12, we worked with **continuous outcomes** (means):

- One-sample t-test: detect difference from a known mean
- Paired t-test: detect before/after change
- Two-sample t-test: detect difference between groups

In Lessons 13–14, we shifted to categorical outcomes (proportions):

- One proportion: Is p different from some p_0?
- Two proportions: Is $p_1 - p_2 \neq 0$?

Today: Power for proportions

The **logic** of power is identical — we just need a different way to define effect size when our outcome is binary.

Effect size for proportions: Cohen's *h*

For **means**, we used Cohen's *d*: a standardized difference (one number)

$$d = \frac{\mu_1 - \mu_2}{\sigma}$$

For **proportions**, effect size is called Cohen's *h*:

$$h = 2\arcsin(\sqrt{p_1}) - 2\arcsin(\sqrt{p_2})$$

You don't need to memorize this!

The `pwr` package computes *h* for you using `ES.h(p1, p2)`.

What you **do** need to specify are the **two proportions** themselves — not a single standardized number.

Cohen's *h* in practice

```r
library(pwr)

# Effect size between p1 = 0.50 and p2 = 0.45
ES.h(p1 = 0.50, p2 = 0.45)
```

```
[1] 0.1001674
```

```r
# Effect size between p1 = 0.10 and p2 = 0.05
ES.h(p1 = 0.10, p2 = 0.05)
```

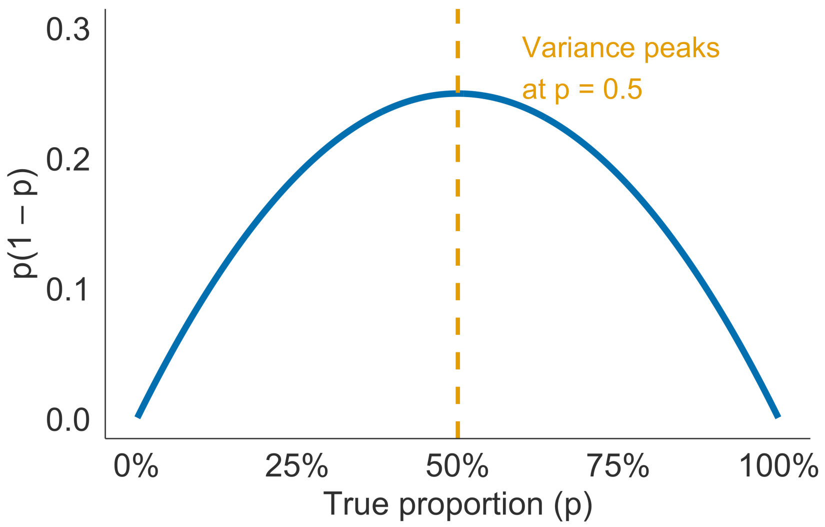
```
[1] 0.1924743
```

Key insight

The same **absolute difference** of 0.05 between two proportions is not always the same effect size! A difference between 0.50 and 0.45 has a smaller effect size than a difference between 0.10 and 0.05.

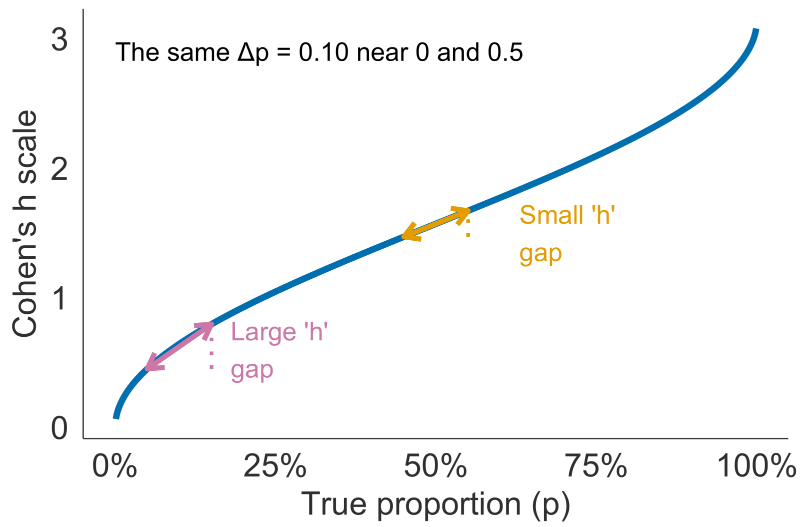
This is why we need the arcsine transformation — it accounts for the fact that variance depends on the proportion itself.

Why we need the arcsine transformation



Variance of $\hat{p}$ = p(1 - p)

Same n, very different variability



Arcsine transformation: 2·arcsin(√p)

Proportions near 0 and 1 are "stretched" to account for lower variability

The key insight

Left: The variance of a proportion estimate is not constant — a difference of 0.10 near p = 0.50 is much noisier than the same difference near p = 0.05 or p = 0.95.

Right: The arcsine transformation stretches the scale near 0 and 1, so that equal *transformed* gaps correspond to equal statistical difficulty — regardless of where on the [0,1] scale you are. This is what Cohen’s h captures.

The pwr functions for proportions

Two main functions in the pwr package:

One proportion:

```r
pwr.p.test(
  h = ES.h(p1, p0),      # effect size
  n = NULL,              # solve for n
  sig.level = 0.05,
  power = 0.80,
  alternative = "two.sided"
)
```

Use when: comparing a sample proportion to a known historical value

Two proportions (equal n):

```r
pwr.2p.test(
  h = ES.h(p1, p2),      # effect size
  n = NULL,              # n PER GROUP
  sig.level = 0.05,
  power = 0.80,
  alternative = "two.sided"
)
```

Use when: comparing two independent groups

Just like `pwr.t.test()` — leave the quantity you want to solve for as `NULL`!

Part 2: Power for a Single Proportion

Recall: The melanoma immunotherapy example

From Lesson 13, we worked with a melanoma immunotherapy study:

The question

Historical data show that approximately **30%** of melanoma patients respond to standard treatment.

A new immunotherapy is hypothesized to increase the response rate to **50%**.

Before running the trial, researchers want to know: **how many patients do we need?**

This is a one-sample proportion test:

- $H_0 : p = 0.30$
- $H_A : p \neq 0.30$

Power for a single proportion

Step 1: Define the proportions

```r
p_null <- 0.30      # historical/null proportion
p_alt  <- 0.50      # expected proportion under new treatment
```

Step 2: Calculate the effect size

```r
h <- ES.h(p1 = p_alt, p2 = p_null)
h
```

```
[1] 0.4115168
```

Step 3: Solve for sample size

```r
pwr.p.test(h = h, sig.level = 0.05, power = 0.80, alternative = "two.sided")
```

```

     proportion power calculation for binomial distribution (arcsine transformation) 

              h = 0.4115168
              n = 46.34804
      sig.level = 0.05
          power = 0.8
    alternative = two.sided
```

Interpreting the result

```
 proportion power calculation for binomial distribution (arcsine transformation)

          h = 0.4115168
          n = 46.34804
  sig.level = 0.05
      power = 0.8
alternative = two.sided
```

Interpretation

To detect an increase in response rate from 30% to 50% (Cohen's h = 0.412) with 80% power and α = 0.05, we would need **n = 47 patients**.

Note: always **round up** when solving for sample size — you can't have a fraction of a person!

What if we can only enroll 40 patients?

Sometimes enrollment is limited by budget or feasibility. We can flip the question: what is our power with a fixed n?

```r
p_null <- 0.30      # historical/null proportion
p_alt  <- 0.50      # expected proportion under new treatment

pwr.p.test(h = ES.h(p1 = p_alt, p2 = p_null),
           n = 40,
           sig.level = 0.05,
           alternative = "two.sided")
```

```
     proportion power calculation for binomial distribution (arcsine transformation) 

              h = 0.4115168
              n = 40
      sig.level = 0.05
          power = 0.7397922
    alternative = two.sided
```

Discussion

With only 40 patients, our power drops substantially. Is that acceptable? **This depends on the context**. For an early-phase pilot study, lower power may be acceptable. For a confirmatory trial, probably not.

Sensitivity analysis: varying the alternative proportion

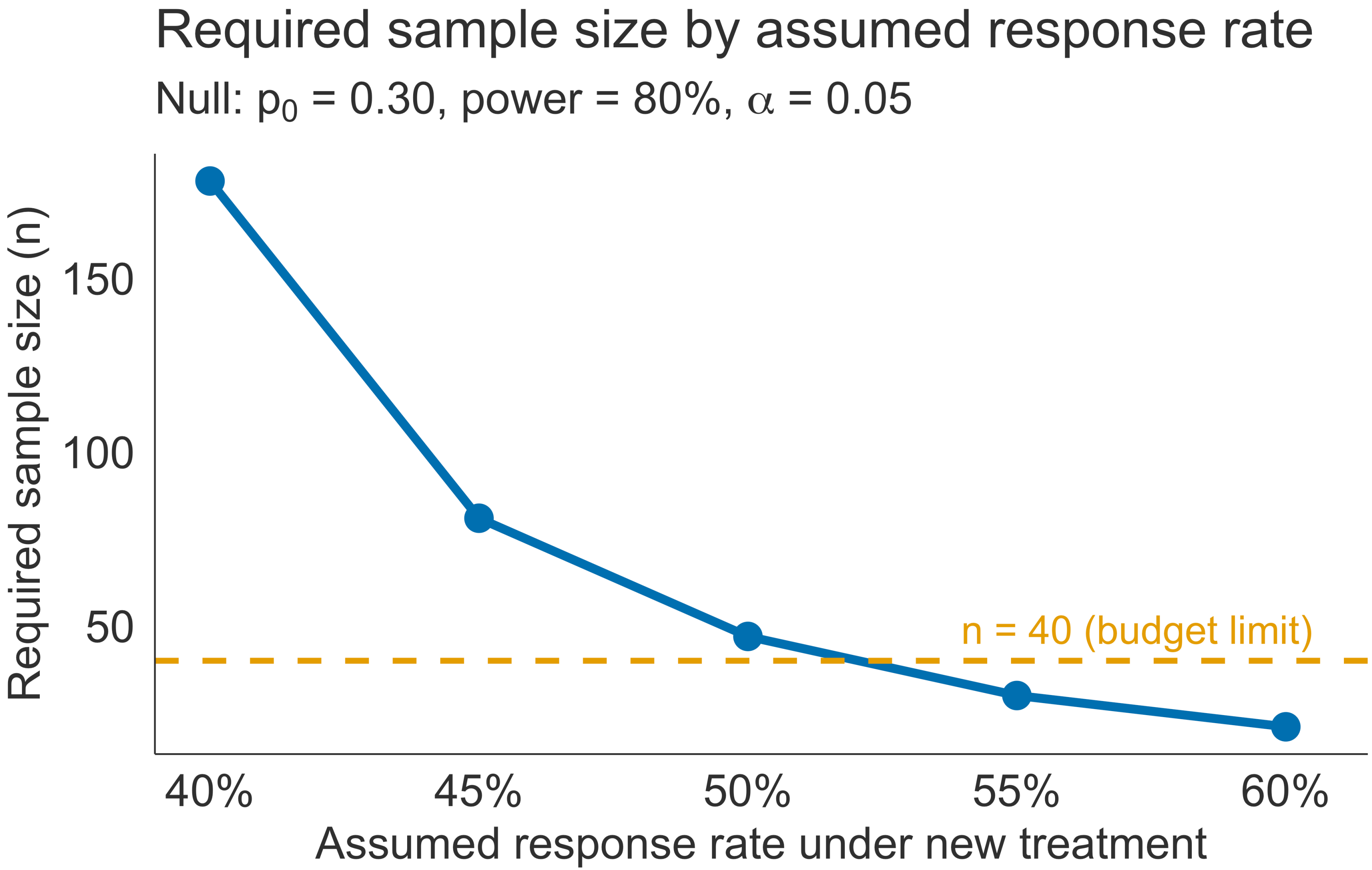
What if we're not sure the new treatment achieves 50%? We can calculate sample size for a range of alternatives:

```r
scenarios <- tibble(
  p_alt = c(0.40, 0.45, 0.50, 0.55, 0.60),
  p_null = 0.30,
  h = ES.h(p_alt, p_null)) %>%
  rowwise() %>%
  mutate(
    n_needed = ceiling(pwr.p.test(h = h,
                                  sig.level = 0.05,
                                  power = 0.80,
                                  alternative = "two.sided")$n)
    ) %>%
  ungroup()

scenarios
```

```
# A tibble: 5 × 4
  p_alt p_null     h n_needed
  <dbl>  <dbl> <dbl>    <dbl>
1  0.4     0.3 0.210      178
2  0.45    0.3 0.311       81
3  0.5     0.3 0.412       47
4  0.55    0.3 0.512       30
5  0.6     0.3 0.613       21
```

Visualizing the sensitivity analysis



The further the true proportion is from the null, the smaller the sample we need — because the effect is easier to detect.

Part 3: Power for Two Independent Proportions

A new treatment comparison

Research scenario

A clinical trial is planned to compare two immunotherapy regimens for melanoma:

- **Standard immunotherapy** (control): historical response rate of **40%**
- **Novel combination therapy** (treatment): expected response rate of **60%**

Two groups of equal size will be randomized. **How many patients per group do we need?**

This is a **two independent proportions** problem:

- $H_0 : p_1 = p_2$
- $H_A : p_1 \neq p_2$

Power for two proportions: sample size

Step 1: Define the two proportions

```
p_control   <- 0.40    # response rate, standard treatment
p_treatment <- 0.60    # expected response rate, novel treatment
```

Step 2: Compute effect size and solve for n

```
pwr.2p.test(
  h = ES.h(p1 = p_treatment, p2 = p_control),
  sig.level = 0.05,
  power = 0.80,
  alternative = "two.sided"
)
```

```

     Difference of proportion power calculation for binomial distribution (arcsine transformation) 

              h = 0.4027158
              n = 96.79194
      sig.level = 0.05
          power = 0.8
    alternative = two.sided

NOTE: same sample sizes
```

Interpreting the result

```
     Difference of proportion power calculation for binomial distribution (arcsine transformation) 

              h = 0.4027158
              n = 96.79194
      sig.level = 0.05
          power = 0.8
    alternative = two.sided

NOTE: same sample sizes
```

Interpretation

To detect a difference in response rates from 40% to 60% (Cohen's $h = 0.403$) with 80% power and $\alpha = 0.05$:

- **n = 97 per group**
- **Total N = 194** (both groups combined)

Add 10–20% buffer for dropout: plan for approximately **112 per group** in practice.

What difference can we detect with fixed resources?

Suppose budget limits enrollment to **50 patients per group**. What’s our power, and what minimum difference can we detect?

```r
# What is our power with n = 50 per group?
pwr.2p.test(
  h = ES.h(p1 = p_treatment, p2 = p_control),
  n = 50,
  sig.level = 0.05,
  alternative = "two.sided"
)
```

```
     Difference of proportion power calculation for binomial distribution (arcsine transformation)

              h = 0.4027158
              n = 50
      sig.level = 0.05
          power = 0.5214145
    alternative = two.sided

NOTE: same sample sizes
```

Sensitivity analysis: varying the treatment proportion (1/2)

```r
scenarios2 <- tibble(
  p_tx   = c(0.50, 0.55, 0.60, 0.65, 0.70),
  p_ctrl = 0.40,
  h = ES.h(p_tx, p_ctrl)
) %>%
  rowwise() %>%
  mutate(
    n_per_group = ceiling(pwr.2p.test(h = h,
                                      sig.level = 0.05,
                                      power = 0.80,
                                      alternative = "two.sided")$n),
    n_total = n_per_group * 2
  ) %>%
  ungroup()
```

Sensitivity analysis: varying the treatment proportion (2/2)

```r
scenarios2
```

```
# A tibble: 5 × 5
   p_tx p_ctrl     h n_per_group n_total
  <dbl>  <dbl> <dbl>       <dbl>   <dbl>
1  0.5     0.4 0.201         388     776
2  0.55    0.4 0.302         173     346
3  0.6     0.4 0.403          97     194
4  0.65    0.4 0.506          62     124
5  0.7     0.4 0.613          42      84
```

Tip

Ask: For each scenario, is the required sample size feasible given your study constraints? Is the assumed treatment effect realistic? Is it clinically meaningful?

Comparing power calculations: means vs. proportions

	Means	**Proportions**
Effect size	Cohen's *d*: one standardized number	Cohen's *h*: computed from two proportions
Key inputs	μ_1, μ_2, σ	p_1, p_2
R function (one group)	`pwr.t.test(type = "one.sample")`	`pwr.p.test()`
R function (two groups)	`pwr.t.test(type = "two.sample")`	`pwr.2p.test()`
R function (paired)	`pwr.t.test(type = "paired")`	*Today: Part 4*
Solve for n?	Leave `n = NULL`	Leave `n = NULL`
Solve for power?	Leave `power = NULL`	Leave `power = NULL`

The workflow is the same — just different inputs and functions!

Part 4: Correlated Proportions and McNemar's Test

When are proportions correlated?

Recall from earlier in the course: observations can be **paired or matched**

We've seen this with means:

- Paired t-test: cholesterol before/after treatment in the *same patient*
- Within-subject design: measurements are correlated

The same situation arises with proportions:

- Does a screening test result change before and after a training intervention?
- In a matched case-control study: does exposure status differ between cases and their matched controls?
- Pre/post binary outcomes measured in the same individuals

Key idea

When binary outcomes are paired or matched, the observations are **correlated** — we cannot treat them as independent. Using the two-proportion test would be wrong!

A matched study example (1/2)

Study design

Researchers want to evaluate a new patient education program for melanoma early detection. They recruit **50 patients** and test each patient’s ability to correctly identify suspicious lesions **before** and **after** the program.

- **Outcome:** Correctly identified suspicious lesion (Yes/No)
- **Design:** Each patient is their own control (paired)

A matched study example (2/2)

The data structure: Each patient has two binary outcomes (before, after)

```r
# Simulated data
set.seed(620)

education_data <- tibble(
  patient_id = 1:50,
  before = rbinom(50, 1, 0.40),   # 40% correct before
  after  = rbinom(50, 1, 0.70)    # 70% correct after (simulated improvement)
) %>%
  mutate(before = if_else(before == 1, "Correct", "Incorrect"),
    after  = if_else(after == 1, "Correct", "Incorrect"))

head(education_data)
```

```
# A tibble: 6 × 3
  patient_id before    after
       <int> <chr>     <chr>
1          1 Incorrect Correct
2          2 Incorrect Correct
3          3 Incorrect Incorrect
4          4 Incorrect Incorrect
5          5 Incorrect Correct
6          6 Correct   Correct
```

The 2×2 table for paired proportions

```r
# Cross-tabulate before vs. after
edu_table <- education_data %>%
  janitor::tabyl(before, after) %>%
  janitor::adorn_title(placement = "combined")

edu_table
```

```
before/after Correct Incorrect
     Correct      17         4
   Incorrect      16        13
```

What these cells mean:

	After: Correct	After: Incorrect
Before: Correct	Concordant (+/+)	Discordant (+/−)
Before: Incorrect	Discordant (−/+)	Concordant (−/−)

Key insight

The **concordant pairs** (where before = after) give us no information about change.

Only the **discordant pairs** tell us something changed — and McNemar’s test focuses entirely on those.

McNemar's test: the big idea

McNemar's test asks: **among the discordant pairs, are they evenly split?**

Let:

- b = pairs where outcome changed from **correct → incorrect**
- c = pairs where outcome changed from **incorrect → correct**

Hypotheses

H_0: The probability of changing in each direction is equal ($p_{12} = p_{21}$, or equivalently $b = c$)

H_A: The probability of changing differs by direction ($p_{12} \neq p_{21}$)

Test statistic:

$$\chi^2 = \frac{(b-c)^2}{b+c}$$

This follows a chi-squared distribution with 1 degree of freedom.

McNemar’s test in R

```r
# Create a table for mcnemar.test()
edu_tab <- table(education_data$before, education_data$after)

edu_tab
```

```
           Correct Incorrect
  Correct       17         4
  Incorrect     16        13
```

```r
# Run McNemar's test
mcnemar.test(edu_tab)
```

```

	McNemar's Chi-squared test with continuity correction

data:  edu_tab
McNemar's chi-squared = 6.05, df = 1, p-value = 0.01391
```

Interpreting the result

```

	McNemar's Chi-squared test with continuity correction

data:  edu_tab
McNemar's chi-squared = 6.05, df = 1, p-value = 0.01391
```

Interpretation

21 / 50 (42%) patients correctly identified suspicious lesions before the education program compared to 33 / 50 (66%) after.

McNemar's chi-squared test ($\chi^2 = 6.05$, df = 1, $p = 0.014$) provides evidence that the patient education program changed the proportion of patients who correctly identified suspicious lesions.

What makes this different from a chi-squared test?

A regular chi-squared test would treat the before/after observations as independent. McNemar's test correctly accounts for the paired structure by focusing only on discordant pairs.

McNemar's test in R with janitor or rstatix

Using janitor:

```r
education_data %>%
  tabyl(before, after) %>%
  column_to_rownames("before") %>% # Extra step using tibble::column_to_rownames
  as.matrix() %>%                  # Extra step to convert to matrix
  mcnemar.test() %>%
  broom::tidy()
```

```
# A tibble: 1 × 4
  statistic p.value parameter method
      <dbl>   <dbl>     <dbl> <chr>
1      6.05  0.0139         1 McNemar's Chi-squared test with continuity correc…
```

Using rstatix:

```r
# Create a table for mcnemar.test()
edu_tab <- table(education_data$before, education_data$after)

rstatix::mcnemar_test(edu_tab)
```

```
# A tibble: 1 × 6
      n statistic    df      p p.signif method
* <int>     <dbl> <dbl>  <dbl> <chr>    <chr>
1    50      6.05     1 0.0139 *        McNemar test
```

McNemar's test vs. two-proportion test

Two-proportion test (Lesson 13)

- Two **independent** groups
- Example: treatment A vs. treatment B in different patients
- Test: `prop.test()`
- Asks: are the proportions the same across groups?

McNemar's test (today)

- **Paired/matched** observations on the same individuals
- Example: before vs. after in the same patients, or matched case-control pairs
- Test: `mcnemar.test()`
- Asks: are the discordant pairs symmetric?

The parallel to t-tests

This mirrors the distinction between the **independent two-sample t-test** (different people) and the **paired t-test** (same person measured twice). Using the wrong test leads to incorrect inference!

Paired proportions: the 2×2 table

Each pair produces two binary outcomes (e.g., before vs. after treatment):

	Post: +	**Post: –**
Pre: +	p_{11} (concordant)	p_{12} (discordant)
Pre: –	p_{21} (discordant)	p_{22} (concordant)

- **Concordant pairs** (p_{11}, p_{22}): outcome is the same at both time points
- **Discordant pairs** (p_{12}, p_{21}): outcome changes
  - p_{12}: changed from **+** to **–** (e.g., “got worse”)
  - p_{21}: changed from **–** to **+** (e.g., “improved”)

McNemar’s test asks: among the discordant pairs, is the split between p_{12} and p_{21} different from 50/50?

Power for paired proportions: the key inputs

Power analysis for McNemar's test is more complex than for independent proportions. The key insight:

McNemar's test only uses the discordant pairs. So power depends on:

1. **Total sample size** (n) — total number of matched pairs
2. **Proportion of discordant pairs** $(p_d = p_{12} + p_{21})$ — pairs where the outcome changes
3. **The direction of discordance** — which way is the change expected to go?

Practical implication

A study with a high proportion of concordant pairs (most people don't change) is effectively working with a smaller sample — and thus has lower power — than the total n would suggest.

When planning a study using McNemar's test, you need estimates of the **discordant pair proportions** from prior literature or pilot data.

Estimating power for McNemar's: a simplified approach

One practical approach: McNemar's test on n total pairs is equivalent to a one-sample proportion test on the **discordant pairs** only.

If we expect:

- Proportion of pairs with outcome change: $p_d = p_{12} + p_{21}$ (total discordant pairs)
- Among discordant pairs, proportion "improving" (p_{21}): $\phi = p_{21}/p_d$
- Under H_0: $\phi = 0.50$ (changes equally likely in both directions)

Interpreting power for McNemar’s

```
     proportion power calculation for binomial distribution (arcsine transformation) 

              h = 0.4115168
              n = 46.34804
      sig.level = 0.05
          power = 0.8
    alternative = two.sided
```

This tells us we need **47 discordant pairs**. To get the total sample size:

```r
n_discordant_needed <- ceiling(mcnemar_power$n)
p_discordant <- 0.25

n_total_needed <- ceiling(n_discordant_needed / p_discordant)
n_total_needed
```

```
[1] 188
```

> **Note**
>
> If only 25% of pairs are expected to be discordant, we need to enroll **188 total pairs** to get enough discordant pairs for adequate power.

Wrap-up and Key Takeaways

Summary: Power for proportions

What's the same as means:

- Four components still in equilibrium (α, power, n, effect)
- Leave one `= NULL` to solve for it
- 80% power is the standard target
- Always round n up
- Report your power analysis!

Key R functions:

- `ES.h(p1, p2)` — compute effect size
- `pwr.p.test()` — one proportion
- `pwr.2p.test()` — two independent proportions
- `mcnemar.test()` — test for paired proportions

What's different for proportions:

- Effect size requires **two** proportions, not a single *d*
- The same absolute difference has different effect sizes at different baseline proportions
- For paired proportions: McNemar's test focuses on discordant pairs
- Power for McNemar's requires knowing the proportion of discordant pairs

Decision guide:

Design	Analysis	Power function
One proportion vs. p_0	`prop.test()`	`pwr.p.test()`
Two independent groups	`prop.test()`	`pwr.2p.test()`
Paired/matched binary	`mcnemar.test()`	`pwr.p.test()` on discordant n

Connecting the course together

We've now covered power and sample size for the full set of tests we've studied:

Test	Outcome	Power function
One-sample t-test	Continuous	`pwr.t.test(type = "one.sample")`
Paired t-test	Continuous	`pwr.t.test(type = "paired")`
Two-sample t-test	Continuous	`pwr.t.test(type = "two.sample")`
One proportion	Binary	`pwr.p.test()`
Two independent proportions	Binary	`pwr.2p.test()`
Paired proportions (McNemar's)	Binary	`pwr.p.test()` on discordant *n*

Tip

The framework is always the same: four components, three known, one to solve for. The function changes, but the logic doesn't.

Looking ahead

Next class (Lesson 16): ANOVA — Comparing means across 3 or more groups

- When the two-sample t-test isn’t enough
- The F-test
- Post-hoc comparisons

Remaining lectures:

- Lesson 17: Nonparametric tests
- Lesson 18: Correlation and Simple Linear Regression
- Lesson 19: Finals review
- Lesson 20: TBD

HW 7 due Sunday 03/08 — will cover material from

- Lesson 14: Chi-squared tests, Fishers exact test
- Lesson 15 (today): Power for proportions and correlated proportions

Additional resources

For deeper reading on power for proportions:

- PASS documentation: Two Proportions
- Sample size calculators from UCSF — web-based, user friendly
- G*Power — free desktop software with proportion-specific calculators

For power analysis with chi-squared tests (beyond this course):

- Abdul Rahman et al. (2025). "Practical guide to calculate sample size for chi-square test in biomedical research." *BMC Medical Research Methodology*. https://pmc.ncbi.nlm.nih.gov/articles/PMC12107878/ — introduces Cohen's *w*, includes a free web-based calculator
- `pwr.chisq.test()` in the `pwr` package and `PowerChisqTest()` in the `DescTools` package are available in R if you need to do this programmatically

For McNemar's test and power:

- PASS documentation: McNemar's Test
- The `exact2x2` package in R has additional tools for exact McNemar's calculations