Simulating power in practice

Author: Nicholas G Reich

This material is part of the ***statsTeachR*** *project*

Today's Lecture

- What is statistical power?
- Why/how might we want to simulate it?
- An example

Refresher: statistical power

Definition of statistical power

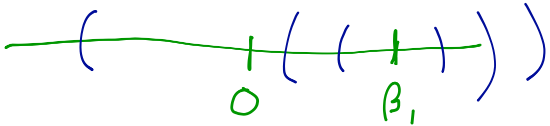
- The ability of a method/test to detect an effect, conditional on that effect actually existing.
- The probability that our test rejects the null hypothesis when the null hypothesis is not true.
- Or, “finding a signal that is really there”

Characteristics that impact power

What imapct do increases in these features have on power?

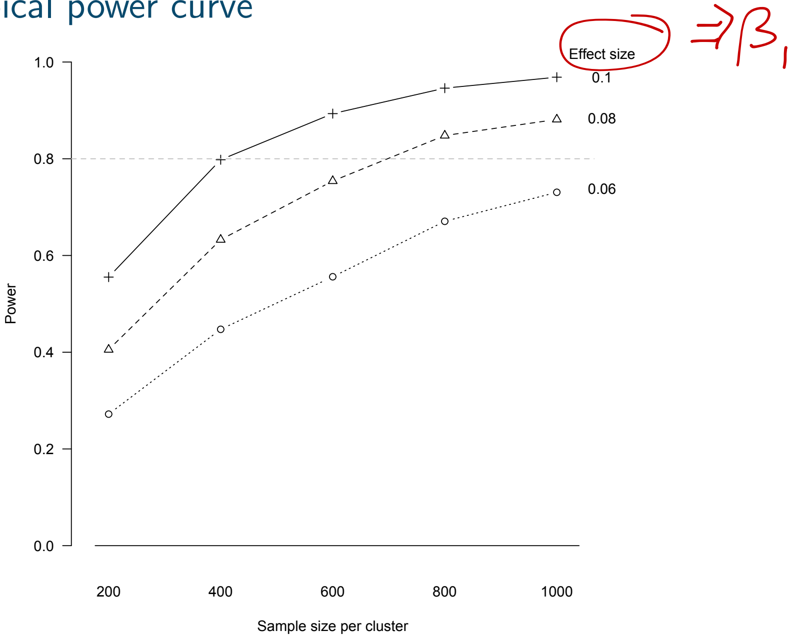
$$Var(\hat{\beta}_1) = \frac{\hat{\sigma}^2}{S_{xx}}$$

- sample size $\uparrow$
- effect size $\uparrow$
- variance of outcome $\downarrow$
- variance of predictors $\uparrow$
- number of predictors $(\downarrow)(\uparrow)$
- grouped/clustered observations $\downarrow$



$H_0: \beta_0 = 0$

Typical power curve



Reich NG, Myers JA, Obeng D, Milstone AM, Perl TM. Empirical power and sample size calculations for cluster-randomized and cluster-randomized crossover studies. PLoS ONE. 2012. 7(4): e35564.

“Post-hoc” power calculations are controversial

It is *always* preferable to calculate power prior to running your analysis. Better not to justify negative findings with a post-hoc power calculation!

For more, see The Abuse of Power, among others.

Formula-based power calculation

Many simple tests have formulas for power, these ...

- are easy to use
- may require you to estimate parameters from existing data (or make up justifiable numbers to plug in)
- are often appropriate for simple tests
- assume all standard assumptions are met
- are only available for simple/standard tests

Simulation-based power calculation

Calculating power via simulation is a tradeoff: computational complexity for customization and flexibility.

Power simulations...

- are available for any setting where you can simulate data (not limited to simple scenarios)
- can be used to preserve complex correlation sturctures in predictors (resample your X's)
- are not assumption- or parameter-free
- often require more complicated coding
- may be computationally intensive (i.e. need a long time to run)

Example: t-test power calculation “by hand”

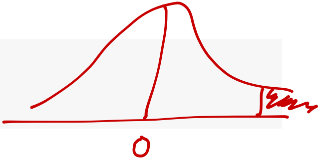
T-test: comparing mean between two groups

- $\mu_1 = 5, \mu_2 = 7$
- $\sigma_1^2 = \sigma_2^2 = 5$ (assume known)
- $n_1 = n_2 = 20$
- Type I error rate $= \alpha = 0.05$
- H_0: $\mu_1 - \mu_2 = 0$

Z_{crit} $1-\frac{\alpha}{2}$

$$Power = 1 - \beta = Pr\left(Z > 1.96 - \frac{|\mu_1 - \mu_2|}{\sqrt{2\sigma^2/n}}\right)$$

```
pnorm(1.96 - 2/sqrt(2*5/20), lower.tail = FALSE)

## [1] 0.8074197
```



Example: t-test power calculation “black box”

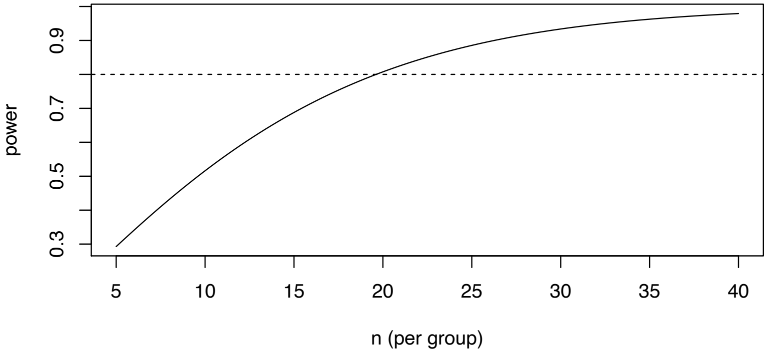
Compare to another method, which uses numerical optimization

```
pnorm(1.96 - 2/sqrt(2*5/20), lower.tail = FALSE)

## [1] 0.8074197

power.t.test(n = 20, delta = 2, sd = sqrt(5), sig.level=0.05)

##
##      Two-sample t test power calculation
##
##               n = 20
##           delta = 2
##              sd = 2.236068
##       sig.level = 0.05
##           power = 0.7870829
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

Example: t-test power calculation (graph)

Evaluate power across sample sizes

```
curve(pnorm(1.96 - 2/sqrt(2*5/x), lower.tail = FALSE),
      from=5, to=40, ylab="power", xlab="n (per group)")
abline(h=.8, lty=2)
```



Example: t-test power calculation "simulated"

```
nsim <- 1000
n <- 20
mu1 <- 5
mu2 <- 7
s2 <- 5
reject <- rep(0, nsim)
for(i in 1:nsim){
    x <- rnorm(20, mean=mu1, sd=sqrt(s2))
    y <- rnorm(20, mean=mu2, sd=sqrt(s2))
    tt <- t.test(x, y)
    reject[i] <- tt$p.value<.05
}
mean(reject)

## [1] 0.782
```

Power by simulation: two different flavors

Option 1: generate all data from scratch

- will generate "clean" data
- hard to insert authentic noise: outliers, missingness, correlated predictor structure

Option 2: resample predictors from a *training* dataset, simulate outcome

- preserves structure of real predictor data
- requires a large dataset similar to the one you will be analyzing
- you should not, in general, do this type of computation on the actual dataset that you are analyzing – best to have a "training" dataset, similar to but independent from the one you will be analyzing

Power by simulation: resampling algorithm

Resampled power algorithm for regression-style models

Inputs: *nsim*; *nobs*, X, a $n \times p$ design matrix; a simulation model $y \sim f(X|\theta)$, with associated paramters; H_0 to test; Type I error rate, α.

1. Define a zero vector r of length *nsim*.
2. **for** i in $1 : nsim$ **do**
3. Resample (with replacement) the rows of X to create X_i, a new $nobs \times p$ design matrix.
4. Simulate y.
5. Fit the model, calculate test statistic for to test evidence for evaluating H_0.
6. Save $r_i = 1$ if p-value $\leq \alpha$.
7. Calculate power as $1 - \beta = \frac{1}{nsim} \sum_{i=1}^{nsim} r_i$.

Adapted from Kleinman and Huang (2014), and Meyers et al. (2014)

Power by resampling: example

You plan to do a follow-up study to the one that generated our `lung` dataset. You want to replicate the results that show a significant impact of smoking on the severity of disease. You don't have a lot of money to conduct the study, so you want to enroll as few participants as possible.

mlr

```
##
## Call:
## lm(formula = disease ~ nutrition + airqual + crowding + smoking,
##     data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.1297 -2.1834 -0.5716  1.9412 13.3260
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 11.863333   2.578819   4.600 1.32e-05 ***
## nutrition   -0.032784   0.007954  -4.122 8.09e-05 ***
## airqual      0.257883   0.026799   9.623 1.17e-15 ***
## crowding     1.111126   0.102037  10.889  < 2e-16 ***
## smoking      4.960931   1.085292   4.571 1.48e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.644 on 94 degrees of freedom
## Multiple R-squared:  0.8664, Adjusted R-squared:  0.8607
```

Power by resampling: example

Inputs

- $nsim = 1000$
- X, taken from `lung` dataset.
- $\alpha = 0.05$
- Simulation model:

$$y_i = \beta_0 + \beta_1 \cdot Nut_i + \beta_2 \cdot airqual_i + \beta_3 \cdot crowd_i + \beta_4 \cdot smoke_i + \epsilon_i$$

$$\epsilon_i \sim Normal(0, \sigma^2)$$

- parameters, $\theta = (\beta_0, ..., \beta_4, \sigma^2)$ (taken from fitted model)
- H_0: $\beta_4 = 0$

nobs=30

Power by resampling: example code

```
nsim <- 1000
nobs <- 30
b0 <- coef(mlr)[1]
b1 <- coef(mlr)[2]
b2 <- coef(mlr)[3]
b3 <- coef(mlr)[4]
b4 <- coef(mlr)[5]
rej <- rep(0, nsim)
for(i in 1:nsim) {
    tmp_idx <- sample(1:nrow(data), replace=TRUE, size=nobs)
    new_data <- data[tmp_idx,]
    err <- rnorm(nobs, 0, s=summary(mlr)$sigma)
    new_data$dis <- with(new_data, (b0 + b1*nutrition + b2*airqual +
                           b3*crowding + b4*smoking + err))
    fm <- lm(dis ~ nutrition + airqual + crowding + smoking, data=new_data)
    rej[i] <- summary(fm)$coef["smoking", "Pr(>|t|)"] > 0.05
    }
(pwr <- sum(rej)/nsim)

## [1] 0.372
```

1-.372 = .628

Power by resampling: example code

.63

So we can say that if we were to repeat this study (in a similar population) with a sample size of 30, our study would have ~~0.37~~ to detect a true relationship between smoking and disease severity.

Power by simulation: wrap-up

- Power analyses can be a really useful tool to explore the likelihood of your data analysis producing valuable results.
- Simulating power can be valuable in settings where there is no simple formula for calculating power.
- It is also a good exercise to try to simulate your data – you learn a lot about the structure of your data in the process!