BMEG 802 – Advanced Biomedical Experimental Design and Analysis

Effect Size and Power

Joshua G. A. Cashaback, PhD

Recap

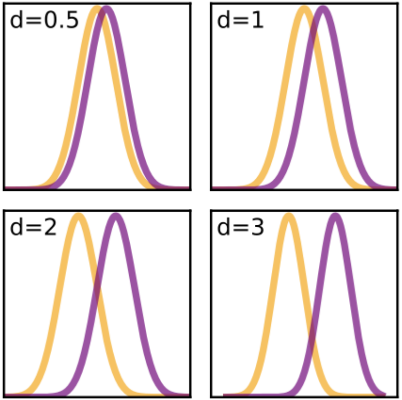
- Null Hypothesis testing
    - We can only falsify a theory!
        - p-value = likelihood of the observed data given the null hypothesis is true
    - single and two-sided tests
    - parametric vs. nonparametric
    - correcting for multiple comparisons

Learning Objectives

- Effect Size
    - Parametric
        - Cohen's D (1 sample, 2 sample, paired)
    - Nonparametric
        - Common Language Effect Size
- Power
    - Parametric
    - Numerical (sampling)

Effect Size

Effect size: Simple way to quantify the difference between two means / groups, by emphasizing the size of the difference rather than confounding with the sample size (like p-values).



Cohen’s D: 1 Sample

$$d = \frac{\bar{X} - \mu_0}{s}$$

d $=$ effect size

$\bar{X}$ $=$ sample mean

μ_0 $=$ theoretical mean against which the sample mean is compared

s $=$ sample standard deviation

Cohen’s D: 1 Sample

From last lecture: A cookie company claims that there are 15 chocolate chips per cookie, but you aren’t convinced. You take 10 cookies and count the number of chocolate chips in each cookie. Here is what the data looks like: [13,14,15,17,18,19,21,20,19,20]. Are the number of chocolate chips significantly different from 15 (reminder: yes, p = 0.015)? Report the effect size.

```r
data1 = c(13,14,15,17,18,19,21,20,19,20)
X = mean(data1)
s = sd(data1)
mu = 15
d = (X - mu) / s
abs(d) # report positive value
```

```
## [1] 0.9431191
```

Cohen’s D: 1 Sample Cont’d

Using a built in function

install.packages(“effsize”)

```r
library(effsize)
cohen.d(data1, NA, mu = 15)
```

```
##
## Cohen's d (single sample)
##
## d estimate: 0.9431191 (large)
## Reference mu: 15
## 95 percent confidence interval:
##      lower      upper
## -0.5650353  2.4512735
```

Cohen’s D: 2 Sample

$$d = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{(n_1-1)s_1^2+(n_2-1)s_2^2}{n_1+n_2-2}}}$$

d = effect size

$\bar{X}$ = sample mean

μ = theoretical mean against which the mean of our sample is compared

s = sample standard deviation

Cohen's D: 2 sample

From last lecture: A math test was given to 300 17 year old students in 1978 and again to another 350 17 year old students in 1992.

- Group 1: $X_1 = 300.4, s_1 = 34.9, n = 300$
- Group 2: $X_2 = 306.7, s_2 = 30.1, n = 350$

Report the effect size.

```
group1 <- rnorm(300, mean = 300.4, sd=34.9)
group2 <- rnorm(350, mean = 306.7, sd=30.1)
d2 = (mean(group1) - mean(group2)) / (sqrt(((length(group1) - 1) * sd(grou
abs(d2)
```

```
## [1] 0.2329287
```

Cohen’s D: 2 sample

Using a built in function

```
cohen.d(group1,group2, var.equal = False) # var.equal = False performs a W
```

```
##
## Cohen's d
##
## d estimate: -0.2329287 (small)
## 95 percent confidence interval:
##       lower       upper
## -0.38794629 -0.07791102
```

Cohen's D: Paired

$$d_{rm} = \frac{\bar{X}_D - \mu_0}{\frac{\sqrt{s_1^2 + s_2^2 - 2 \cdot r \cdot s_1 \cdot s_2}}{\sqrt{2(1-r)}}}$$

d = effect size

$\bar{X}$ = sample mean of the paired differences

μ = theoretical mean that the paired mean differences are compared against

s_i = sample standard deviation at a particular time (i)

r = correlation coefficient (we'll talk more about this next lecture)

Other versions:

https://pure.tue.nl/ws/portalfiles/portal/3835042/1236489301722996.pdf

Cohen's D: Paired Samples

From last class: A manufacturer claims it has developed an additive that increases gas mileage. But you are not sure whether the additive will increase or decrease performance. They recruit 10 drivers. Each driver drives a car on a well-conditioned track. They record the gas mileage without any additive, then with additive. Report the effect size.

```
data2a = c(22,25,17,24,16,29,20,23,19,20)
data2b = c(18,21,16,22,19,24,17,21,23,18)
data2diff = data2a - data2b
d2 = (mean(data2diff) - 0) /
  (sqrt(sd(data2a)^2 + sd(data2b)^2
        - 2 * cor(data2a,data2b) * sd(data2a) * sd(data2b))/
     sqrt(2*(1-cor(data2a,data2b))))
d2
```

```
## [1] 0.4475411
```

Cohen's D: Paired Samples

```r
cohen.d(data2a,data2b, paired = TRUE)
```

```
##
## Cohen's d
##
## d estimate: 0.4475411 (small)
## 95 percent confidence interval:
##      lower      upper
## -0.1277464  1.0228286
```

Common Language Effect Size

- nonparamtric way to calculate effect size
- intuitive
  - the probability that a score sampled at random from one distribution will be greater than a score sampled from some other distribution
  - e.g., the probability that a male will be taller than a female is 0.92. In other words, the male will be taller than the female in 92 out of 100 blind dates among young adults.
- brute force

Common Language Effect Size: 2 Sample, Procedure

- Compare every possible value in group A to every other possible value in group B.
  - Add 1 each time the difference goes in the expected direction.
  - Add 0.5 if there is a tie
  - Add 0.0 if the difference goes in the unexpected direction
  - Take the sum above, divide by the total number of comparisons, subtract 0.5, absolute, add 0.5, multiply by 100 (values should be between 50% and 100%)
- Lets try this out with a simple example first

Common Language Effect Size - 2 sample

```
a = c(1,3,5)
b = c(2,4,6)
c <- array(dim=c(length(a),length(b)))
for (i in 1:length(a)) {
  for (j in 1:length(b)){
    if (a[i] > b[j]) {
    c[i,j] = 1
    } else if (a[i] == b[j]) {
    c[i,j] = 0.5
    } else {
    c[i,j] = 0.0
}}}
CLES = (abs(sum(c) / (length(a) * length(b)) - 0.5) + 0.5) * 100
CLES
```

```
## [1] 66.66667
```

$$\hat{\theta} = 66.7\%$$

Common Language Effect Size - 2 sample

Now lets do this on the 2 Sample example we did earlier

```
a = group1
b = group2
c <- array(dim=c(length(a),length(b)))
for (i in 1:length(a)) {
  for (j in 1:length(b)){
    if (a[i] > b[j]) {
    c[i,j] = 1
    } else if (a[i] == b[j]) {
    c[i,j] = 0.5
    } else {
    c[i,j] = 0.0
}}}
CLES = (abs(sum(c) / (length(a) * length(b)) - 0.5) + 0.5) * 100
CLES
```

```
## [1] 57.7181
```

Common Language Effect Size - 1 Sample

We can do this with a 1 Sample test (same example used above)

```
a = data1 # data
b = array(15,dim=c(length(a)))  # mean value you are comparing to (i.e., mu_0 = 15)
c <- array(dim=c(length(a),length(b)))
for (i in 1:length(a)) {
  for (j in 1:length(b)){
    if (a[i] > b[j]) {
    c[i,j] = 1
    } else if (a[i] == b[j]) {
    c[i,j] = 0.5
    } else {
    c[i,j] = 0.0
}}}
CLES = (abs(sum(c) / (length(a) * length(b)) - 0.5) + 0.5) * 100
CLES
```

```
## [1] 75
```

Common Language Effect Size - Paired

As well as paired test (using the paired differences example above)

```r
a1 = data2a # e.g., pre intervention
a2 = data2b  # e.g., post intervention
a = a2 - a1
b = array(0,dim=c(length(a)))  # mean value you are comparing to (same as mu_0 = 0)
c <- array(dim=c(length(a),length(b)))
for (i in 1:length(a)) {
  for (j in 1:length(b)){
    if (a[i] > b[j]) {
    c[i,j] = 1
    } else if (a[i] == b[j]) {
    c[i,j] = 0.5
    } else {
    c[i,j] = 0.0
}}}
CLES = (abs(sum(c) / (length(a) * length(b)) - 0.5) + 0.5) * 100
CLES
```

```
## [1] 80
```

Effect Size Interpretation

report along p-value

- (p $=$ 0.01, d $=$ 0.5) or (p $=$ 0.01, $\hat{\theta} = 66.2\%$) Can state whether the effect is small ($d \approx 0.2$), medium ($d \approx 0.5$), or large ($d \approx 0.8$).
- For CLES ($\hat{\theta}$): small ($\hat{\theta} \approx 56\%$), medium ($\hat{\theta} \approx 64\%$), or large ($\hat{\theta} \approx 71\%$).
  - All of these definitions are somewhat arbitrary

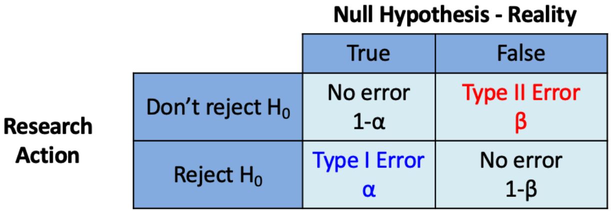
Other Effect Sizes

- Other effect sizes: e.g., Glass's delta, Hedges' g
- Regression (r-value)
- ANOVA
  - eta squared
  - omega squared

STATISTICAL POWER

Power

Power $(1-\beta)$: probability of rejecting the null hypothesis when it is false.

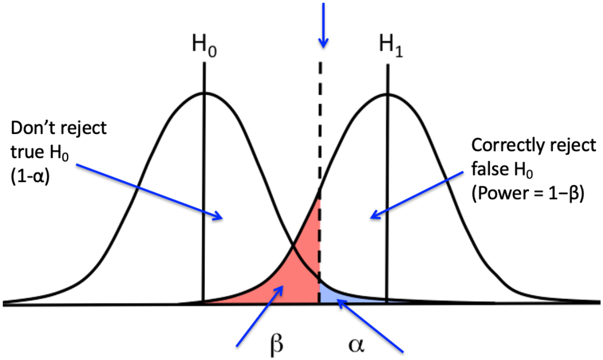


		Null Hypothesis - Reality	
		True	False
Research Action	Don't reject H_0	No error 1-α	Type II Error β
	Reject H_0	Type I Error α	No error 1-β

Power

		Null Hypothesis - Reality	
		True	False
Research Action	Don't reject H_0	No error 1-α	Type II Error β
	Reject H_0	Type I Error α	No error 1-β

Boundary of critical region of test statistic



Don't reject true H_0 (1-α)

Correctly reject false H_0 (Power = 1−β)

β

Type II error

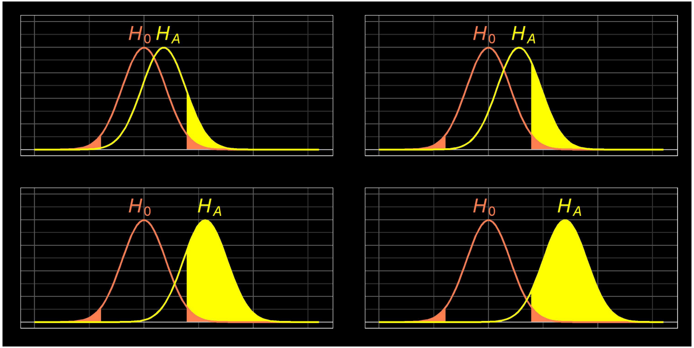
Don't reject a false H_0

α

Type I error

Reject a true H_0

Power



Power depends on:

1. sample size
2. effect size (e.g., d)
3. statistical significance criteria (α)

Power to estimate sample size

Calculate required sample size given a) effect size (e.g., d) b) significance level (α), c) desired power.

1. a priori
   - use literature to estimate effect size (set α, desired power)
2. pilot data
   - estimate effect size using current data (set α, desired power)

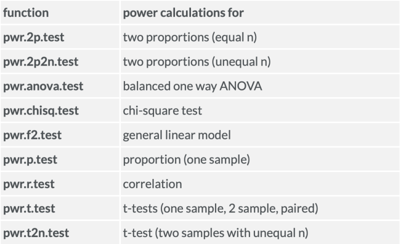
For both a priori and pilot data power analyses, you need to get an estimate of the effect size (e.g., d)

Analytical

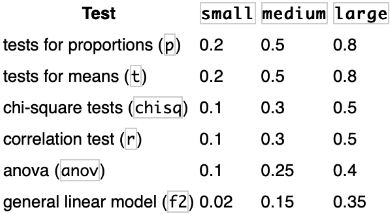
- relatively simple using normal distributions (z-tests, n $>$ 30)
  - see https://en.wikipedia.org/wiki/Power_of_a_test (bottom of page)
  - generally, we should be using t-distributions.
- VERY complex for t-tests analytically
  - Option 1: CDF involves the noncentral t-distribution, which is composed of normal distribution, regularized incomplete beta function.
  - Option 2: use tables (Cohen 1988 uses 66 pages!)
  - Option 3: Built in functions.

R - Power Analysis package

pwr package



function	power calculations for
pwr.2p.test	two proportions (equal n)
pwr.2p2n.test	two proportions (unequal n)
pwr.anova.test	balanced one way ANOVA
pwr.chisq.test	chi-square test
pwr.f2.test	general linear model
pwr.p.test	proportion (one sample)
pwr.r.test	correlation
pwr.t.test	t-tests (one sample, 2 sample, paired)
pwr.t2n.test	t-test (two samples with unequal n)



Test	small	medium	large
tests for proportions (p)	0.2	0.5	0.8
tests for means (t)	0.2	0.5	0.8
chi-square tests (chisq)	0.1	0.3	0.5
correlation test (r)	0.1	0.3	0.5
anova (anov)	0.1	0.25	0.4
general linear model (f2)	0.02	0.15	0.35

Power Analysis (a priori)

From last lecture: A math test was given to 17 year old students in 1978 and again to another 17 year old students in 1992. From that data we have an estimate of the following parameters.

- Group 1: $X_1 = 300.4, s_1 = 34.9, n = 300$
- Group 2: $X_2 = 306.7, s_2 = 30.1, n = 350$

We want to conduct a similar experiment and estimate how many people we should collect to achieve a desired power of 80%

Power Analysis (a priori) Cont'd

install.packages("pwr")

```
library(pwr)
d2 = (300.4 - 306.7) / (sqrt(((300 - 1) * 34.9^2 + (350 - 1) * 30.1^2) / (
d2abs = abs(d2)
# type = "two.sample", "one.sample", "paired"
# alternative = "two.sided", "less", "greater"
pwr.t.test(d = d2abs, power = 0.80, sig.level = 0.05,
           type = "two.sample", alternative = "two.sided")$n
```

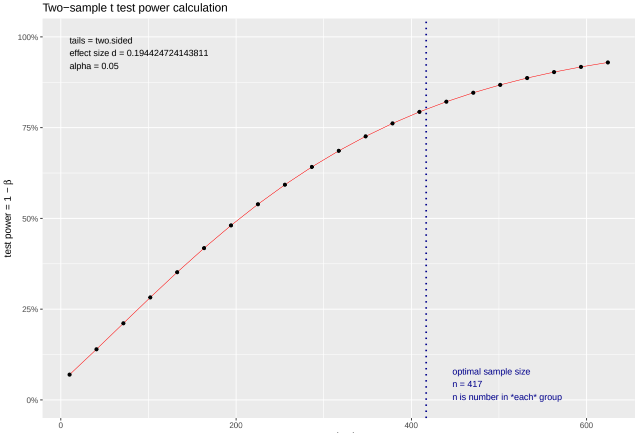
```
## [1] 416.2355
```

You will need to sample $N = 417$ participants (per group) to reach the desired power

- Round up to the next nearest integer

Power Analysis (a priori) Cont'd

```
p.out <- pwr.t.test(d = d2abs, power = 0.80, sig.level = 0.05,
                    type = "two.sample", alternative = "two.sided")
plot(p.out)
```



Power Analysis (Pilot Data)

From last lecture: A cookie company claims that there are 15 chocolate chips per cookie, but you aren't convinced. You take 10 cookies and count the number of chocolate chips in each cookie. Here is what the data looks like: [13,14,15,17,18,19,21,20,19,20]. Are the number of chocolate chips significantly different from 15 (reminder: yes, $p = 0.015$)?

- How many cookies should you sample to get 80% power?

Power Analysis (Pilot Data) Cont'd

```r
data1 = c(13,14,15,17,18,19,21,20,19,20)
d1 = cohen.d(data1, NA, mu = 15) $estimate
pwr.t.test(d = d1, power = 0.80, sig.level = 0.05,
           type = "one.sample", alternative = "two.sided")$n
```

```
## [1] 10.8989
```

We should have sufficient power with a sample size of 11 cookies.

Power Analysis (Pilot Data) Cont'd

Can calculate current Power by slightly adjusting the function

```
data1 = c(13,14,15,17,18,19,21,20,19,20)
d1 = cohen.d(data1, NA, mu = 15) $estimate
pwr.t.test(n = 10, d = d1, power = NULL, sig.level = 0.05,
           type = "one.sample", alternative = "two.sided")$power
```

```
## [1] 0.7563296
```

Currently we are at 75.6% power

Numerical Power Analysis

- It is often difficult to perform power analyses for nonparametric or complicated omnibus tests.
- A solution is numerical simulations
    - computational intensive, but can simulate any test
- Procedure
    - a priori estimates of group mean and standard deviation
        - can also use pilot data (we'll cover this in the bootstrapping lecture)
    - simulate the experiment at least 10,000 times (I often use 100,000 or 1,000,000 for more stability)
    - find proportion of p-values under α
        - this is the expected power!
    - repeat steps above and continually increase N until you reach desired power (e.g., 80%)

Numerical Power Analysis (a priori)

Using the same example as above: A math test was given to 17 year old students in 1978 and again to another 17 year old students in 1992. From that data we have an estimate of the following parameters. We want to conduct a similar experiment and estimate how many people we should collect.

- Group 1: $X_1 = 300.4, s_1 = 34.9, n = 300$
- Group 2: $X_2 = 306.7, s_2 = 30.1, n = 350$

What is the estimated power with n = 100 in each group?

What N in each group gives you 80% power?

Numerical Power Analysis (a priori) Cont'd

What is the estimated power with n = 100 in each group?

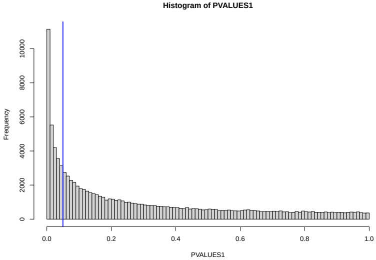
```r
PVALUES1 = array(NA,100000) # initialize NaN array
n = 100
for (i in 1:100000) {
group1 <- rnorm(n, mean = 300.4, sd=34.9)
group2 <- rnorm(n, mean = 306.7, sd=30.1)
pval = t.test(group1, group2, alternative = "two.sided")$p.value
PVALUES1[i] = pval
}
```

Numerical Power Analysis Cont'd (a priori)

What is the estimated power with n = 100 in each group?

```
hist(PVALUES1, breaks = 80)
abline(v=0.05, col="blue", lty=1, lwd=2)
#The estimated power with n = 100 is:
sum(PVALUES1 < 0.05) / 100000 * 100
```

```
## [1] 27.553
```



Numerical Power Analysis Cont’d (a priori)

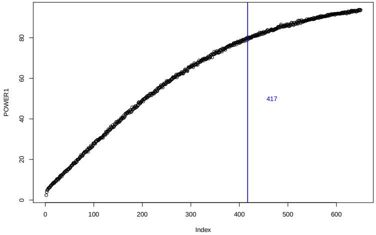
What N gives you 80% power?

```
min_n = 2
max_n = 650
POWER1 = array(NA,max_n - min_n)
for (n in min_n:max_n){
  PVALUES1 = array(NA,10000)
  for (i in 1:10000) {
  group1 <- rnorm(n, mean = 300.4, sd=34.9)
  group2 <- rnorm(n, mean = 306.7, sd=30.1)
  pval = t.test(group1, group2, alternative = "two.sided")$p.value
  PVALUES1[i] = pval
  }
  POWER1[n] = sum(PVALUES1 < 0.05) / 10000 * 100
  #print(n)
}
```

Numerical Power Analysis Cont'd (a priori) - Power Curve

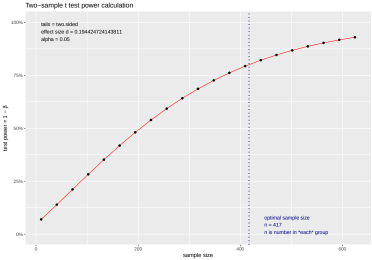
What N gives you 80% power?

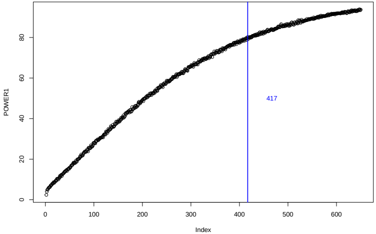
```r
minsub = min(which(POWER1 > 80)) # finds 80% crossing
sub = toString(minsub)
plot(POWER1)
abline(v=minsub, col="blue", lty=1, lwd=2)
text(minsub+50, 50, sub, col = "blue")
```



Compare Analytical vs. Numerical Power Curves

```
p.out <- pwr.t.test(d = d2abs, power = 0.80, sig.level = 0.05,
                type = "two.sample", alternative = "two.sided")
plot(p.out)
plot(POWER1)
abline(v=minsub, col="blue", lty=1, lwd=2)
text(minsub+50, 50, sub, col = "blue")
```





Numerical Power Analysis (pilot data)

We'll do some more numerical power analyses with pilot data during the bootstrapping lecture, as well as using nonparametric tests (e.g., Mann Whitney-U)

Notes on Power

- Addresses replication crisis in science
    - do not check p-values until collected sample size complete – checking before is known as p-hacking (by chance you might falsely find significance)
- perform power analysis on smallest effect you hope to find
- posthoc power analysis after data collection is completely redundant with p-values
    - do not perform if requested by reviewer: see Hoenig & Heisey (2001)
- 80% is typically the desired power

Next Week

- linear regression (next week)
  - Pearson's r and Spearman