

# Experimental Design to Control Type II Error

By Levi Phippen

## Some Basics

- **Type II Error:** Accepting the null hypothesis when the null is false
  - $P(\text{Type II Error}) = \beta$
- **Power = 1 -  $\beta$** 
  - Power is the complement of a type II error.
  - If one doesn't accept  $H_0$  when it is false, then you must have rejected it.
  - So power is the probability of rejecting  $H_0$  when it is false, or in other words, making the right conclusion.

## Reasons for Power Analysis

- The greater the power of a hypothesis test, the more confidence we can have that something is truly different between two groups.
- Power analysis prior to an experiment will help us design a study that maximizes our ability to detect these differences.

## Power Refresher

- Let's remind ourselves of how to find the power of a hypothesis test comparing two population means
- Need to know a few things:
  - $n$  and  $m$ , the sample sizes from the two populations
  - Significance level  $\alpha$
  - The population standard deviation  $\sigma$
  - The true difference between  $\mu_X$  and  $\mu_Y$
- So lets work a problem

## Controlling Type II Error

- The probability of a type II error depends on four things:
  - Sample size
  - The standard deviation
  - The true difference between means
  - The level of significance
- Of these four items, really we can only control two:
  - Sample size
  - The level of significance

## Controlling Type II Error

- Significance Level
  - As  $\alpha$  increases  $\beta$  decreases
    - Increasing  $\alpha$  too much can be unwise
- Sample Size
  - Increasing the sample size of an experiment allows the researcher to hold  $\alpha$  at the desired level while still reducing  $\beta$

## Packages in R

- `pwr`
  - `pwr` is a very simple package for controlling the type II error in various situations
  - `power.t.test`
    - Used to test the power of one and two sample t-test with the same sample size for both groups
    - Does not work in a multiple hypothesis testing situation
  - Examples

## pwr Example

```
power.t.test(n=10, delta = 1, sd = 1,  
             alternative = 'one.sided', type =  
             'one.sample')
```

One-sample t test power calculation

```
          n = 10  
        delta = 1  
          sd = 1  
sig.level = 0.05  
    power = 0.897517  
alternative = one.sided
```

## pwr

- pwr package contains many other functions, for example:
  - `pwr.t2n.test`, t - test (two samples with unequal n)
  - `pwr.2p.test`, two proportions (equal n)
  - `pwr.anova.test`, balanced one way ANOVA

## sizepower

- sizepower is an R package used to calculate sample size and power in a microarray study.
  - It's scope, however, is somewhat limited
    - Only provides sample size calculation for two types of experiments
      - Completely randomized treatment-control designs
      - matched-pairs designs
    - Does not control the FDR, but rather the FWER

## sizepower

- Power calculation
  - Provides power calculation for four types of experimental designs
    - completely randomized treatment-control designs
    - matched-pairs designs
    - multiple-treatment designs having an isolated treatment effect
    - randomized block designs

## sizepower

- The `sampleSize.randomized` function:
  - Arguments
    - `G0` - True number of genes that are not differentially expressed
    - `ER0` - Expected number of false positives
    - `absMu1` - Difference in means stated in the alternative hypothesis
    - `sigmad` - Std of difference in expression between treatment and control samples

## sizepower Examples

```
library(sizepower)
sampleSize.randomized(ER0 = 1, G0 =
  2000, power = 0.9, absMu1 = 1,
  sigmad = 0.566)

$n
[1] 8

$d
[1] 1.766784
```

## sizepower Examples

- $n = 8$  means that eight is the smallest the sample size can be in each group and still achieve the desired power.
- $d = 1.766784$  is the statistical distance between treatment and control means under  $H_a$ , ie.  $d = |\mu_1|/\sigma_d$

## sizepower Examples

```
power.randomized(ER0 = 1, G0 =
  2000, absMu1 = 1, sigmad =
  0.5666, n = 8)

$power
[1] 0.9346278
```

## sizepower Examples

- In this example we have specified the sample size we wanted and used the `power.randomized` function to determine the power given that sample size.

## sizepower G0

- G0 - The true number of differentially expressed genes
  - How do we know this number?
  - We don't, must be estimated somehow
    - Usually, you will depend on a content area specialist to estimate this for you
    - If one is not available, rely on results from similar previous studies
    - The `limma` package has a formula to estimate this given a vector of p-values

## sizepower G0

- Estimating G0 with the `limma` package
  - The `limma` package has a function, `propTrueNull`, that given a vector of p-values, estimates the true proportion of non-differentially expressed genes.

## sizepower G0

- For an example of how this function works, we can use the `top.Cell` table we created in class from the ALL dataset.
  - As a reminder, when generating this table we used the `limma` / `eBayes` approach to test for differential expression between T-cells and B-cells.
  - `top.Cell$P.Value` is a vector of p-values.

## sizepower G0

```
propTrueNull(top.Cell$P.Value)
[1] 0.5049036
```

- Another way to estimate this proportion from a previous study, is to divide the number of non-significant p-values by the total number of tests.

```
t = top.Cell$P.Value < 0.05
G0 = 4305 - sum(t)
G0/4305 #0.5805
```

## sizepower Examples

- As a last example, let's check the power of a test we did on the ALL dataset to see if there was differential expression between the T-cells and B-cells. The null hypothesis was that there was no difference.
- To do that, we'll estimate  $\sigma_d = 1.8789$ , and  $G_0 = 3024.764$

## sizepower Examples

```
power.randomized(ER0 = 1, G0 =  
  3024.764, absMu1 = 1, sigmad =  
  1.8789, n = 64)
```

```
$power
```

```
[1] 0.7478542
```