

# PUMP: Power Under Multiplicity Project

Kristen Hunter<sup>1</sup>, Luke Miratrix<sup>1,2</sup>,  
Kristin Porter<sup>3</sup>, Zarni Htet <sup>3</sup>

National Cancer Institute  
Statistical Adjustment for Multiplicity Workshop  
2022

Slides available: [github.com/kristenbhunter/presentations/tree/master/NCI2022](https://github.com/kristenbhunter/presentations/tree/master/NCI2022)

[1] Harvard Department of Statistics

[2] Harvard Graduate School of Education

[3] MDRC

# Multiple outcomes

- The use of multiple testing procedures (MTPs) changes statistical **power**
- **Problem:** In some fields, current practice for determining statistical power for RCTs does not take the use of MTPs into account
- **Solution:** Easy-to-use software for calculating power, sample size, and minimum detectable effect size (MDES)
- Easy exploration of power over changes in assumed parameter values



Image by Yama Zsuzsanna Márkus from Pixabay



Image by Katrin B. from Pixabay



Image by Katrin B. from Pixabay

# Introducing PUMP

- **PUMP:** R package on CRAN
- Power Under Multiplicity Project
- Calculates power for multiple hypotheses in multilevel **randomized controlled trials** (RCTs)
- **Multilevel:** hierarchical structure, such as patients nested within hospitals nested within hospital systems
- Assumes frequentist **mixed effects** models



Image by Jan Steiner from Pixabay

# Multilevel models

$$\underbrace{Y_{ijkm}}_{\text{Outcome}} = \underbrace{\psi_{1,m}}_{\text{Tx effect}} \underbrace{T_{jk}}_{\text{Tx asst}} + \underbrace{\mu_{0,km}}_{\text{Level 2 Grand mean}} + \sum_{r=1}^{g_{2,m}} \delta_{mr} \underbrace{X_{jkmr}}_{\text{Level 2 covariate}} + \sum_{p=1}^{g_{1,m}} \gamma_{mp} \underbrace{C_{ijkmp}}_{\text{Level 1 covariate}} + \underbrace{u_{0,jkm}}_{\text{Level 2 random effect}} + \underbrace{r_{ijkm}}_{\text{Level 1 random effect}}$$

# Example

I have 3 outcomes, with a 2-level blocked design. My power to detect the effect for any individual outcome:

- Without any adjustment (no MTP): 0.81.
- Using Bonferroni adjustment: 0.67.

Having multiple outcomes has reduced my power...or has it? Stay tuned!



Image by Brian Cragun from Pixabay

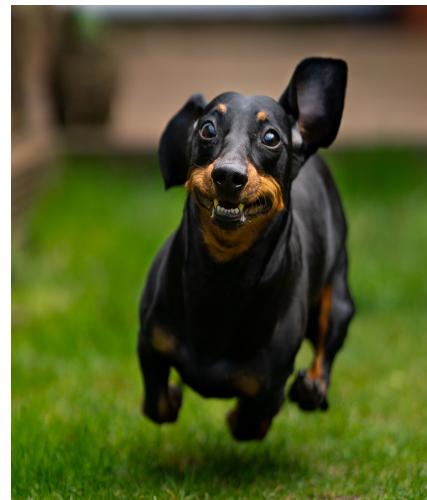


Photo by James Watson on Unsplash

# Factors affecting power in RCTs

With at least one outcome:

- design of the study and assumed model
- $\bar{n}, J, K$ : number of level 1/2/3 units
- $\bar{T}$ : proportion of units treated
- number of covariates
- $R^2$ : explanatory power of covariates
- $ICC$ : intraclass correlation (ratio of variance at level to overall variance)
- Treatment impact heterogeneity

Unique to **multiple** outcomes:

- definition of power
- $M$ : number of outcomes/tests
- $\rho$ : correlation between test statistics
- proportion of outcomes for which there are truly effects
- multiple testing procedure (MTP)

Note: terminology varies across fields!

# Define the experimental setup

How to **design** the experiment

- Levels: 1, 2, 3
- Randomization level: 1, 2, 3

How to **model** the experiment

- Assumes mixed effects regression models
- Intercepts: fixed or random
- Treatment effects: constant, fixed, or random

Supported designs and models:

- d1.1\_m1c
- d2.1\_m2fc
- d2.1\_m2ff
- d2.1\_m2fr
- d2.1\_m2rr
- d2.2\_m2rc
- d3.1\_m3rr2rr
- d3.2\_m3fc2rc
- d3.2\_m3ff2rc
- d3.2\_m3rr2rc
- d3.3\_m3rc2rc

# Definitions of power

How do we define power if we have *multiple* hypotheses/outcomes?

- **Individual** power: probability of rejecting a particular null hypothesis
- **1-Minimal** power: probability of rejecting at least one null hypothesis
- **D-Minimal** power: probability of rejecting at least d null hypotheses
- **Complete** power: probability of rejecting all the null hypotheses

All valid options--the choice depends on how we want to define success!



Image by woodsilver from Pixabay



Image by SnottyBoggins from Pixabay

# Multiple testing procedures

- **Bonferroni**
  - simple
  - most conservative
- **Holm**
  - step down version of Bonferroni
  - less conservative for larger  $p$ -values than Bonferroni
- **Benjamini-Hochberg**
  - step up procedure
  - controls the false discovery rate (less conservative)
- **Westfall-Young** (single step and step down versions)
  - permutation-based approach
  - takes into account correlation structure of outcomes
  - computationally intensive
  - not overly conservative

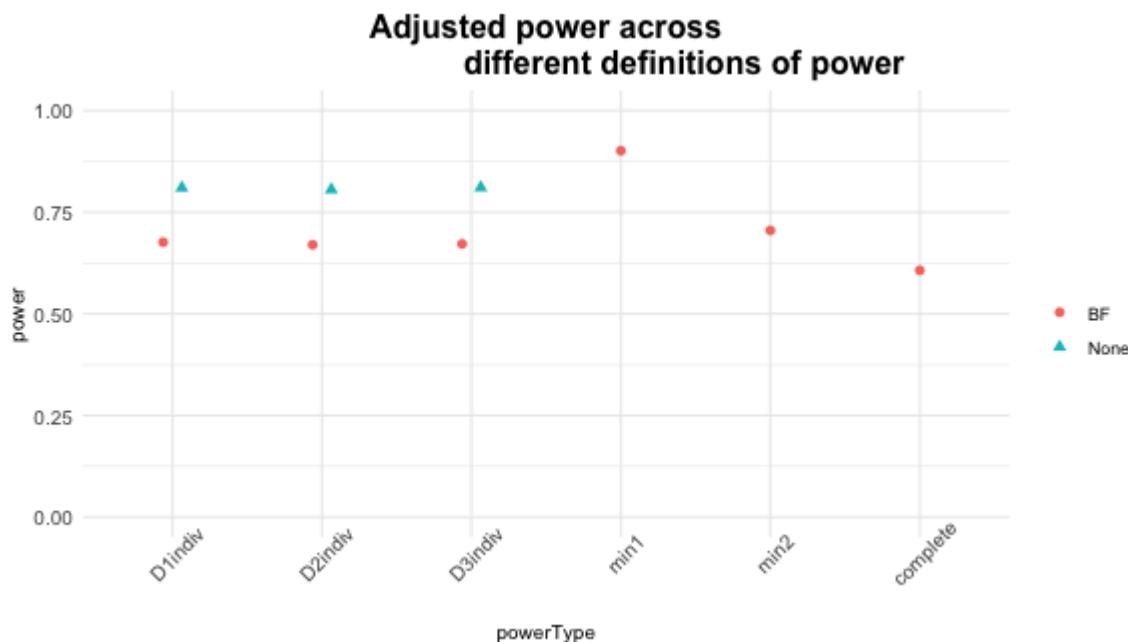
# Diving in!

```
library( PUMP )

pow <- pump_power(
  d_m = "d2.1_m2fc",      # choice of design and model
  MTP = "BF",              # multiple testing procedure
  MDES = rep( 0.10, 3 ),   # assumed effect size
  M = 3,                   # number of outcomes
  J = 10,                  # number of schools/blocks
  nbar = 275,              # average number of students per school
  Tbar = 0.50,              # proportion of students treated per school
  alpha = 0.05,             # significance level
  numCovar.1 = 5,           # number of covariates at level 1
  R2.1 = 0.1,               # assumed R^2 of level 1 covariates
  ICC.2 = 0.05,             # intraclass correlation
  rho = 0.4                # test statistic correlation
)
```

# Power results

MTP	D1indiv	D2indiv	D3indiv	indiv.mean	min1	min2	complete
None	0.81	0.81	0.81	0.81	0.81	NA	NA
BF	0.68	0.67	0.67	0.67	0.67	0.9	0.71



# How it works

- For simple designs and one outcome, we often have a formula for power
- It would be difficult (in some cases impossible) to derive explicit formulas for every design, model, number of outcomes, MTP, and definition of power

Instead, we use **simulation!** A full simulation approach would be:

1. *Simulate data* according to the alternative hypotheses
2. *Calculate test statistics* under the alternative hypotheses
3. Use these test statistics to calculate  $p$ -values
4. Calculate power using the distribution of  $p$ -values

# How it works

- We can simplify this approach by skipping step 1
- Given:
  - design and model
  - correlation between test statistics for different hypotheses
- We know the joint alternative distribution of test statistics!
- Results in **simpler** and **faster** power calculations

Simulation approach to calculating power:

1. *Sample test statistics* under the alternative hypotheses.
2. Use these test statistics to calculate  $p$ -values.
3. Calculate power using the distribution of  $p$ -values.

Note: because we use simulations to calculate power, estimates are approximate, but the user can increase the number of test statistic draws to increase precision.

# Sample size and MDES

We can also calculate:

- `pump_mdes()`: minimum detectable effect size (MDES) for a particular target power
- `pump_sample()`: sample size for a given target power and MDES

Types of sample size calculations:

- K: number of level 3 units (hospital systems)
- J: number of level 2 units (hospitals)
- nbar: number of level 1 units (patients)

# Sample size example

```
ss <- pump_sample(  
  target.power = 0.8,           # target power  
  power.definition = "min1",   # power definition  
  typesample = "J",            # type of sample size procedure  
  tol = 0.01,                 # tolerance  
  d_m = "d2.1_m2fc", MTP = "BF",  
  MDES = 0.1, M = 3, nbar = 350, Tbar = 0.50, alpha = 0.05,  
  numCovar.1 = 5, R2.1 = 0.1, ICC.2 = 0.05, rho = 0.4  
)
```

MTP	Sample.type	Sample.size	min1.power
BF	J	7	0.81

# Assessing sensitivity

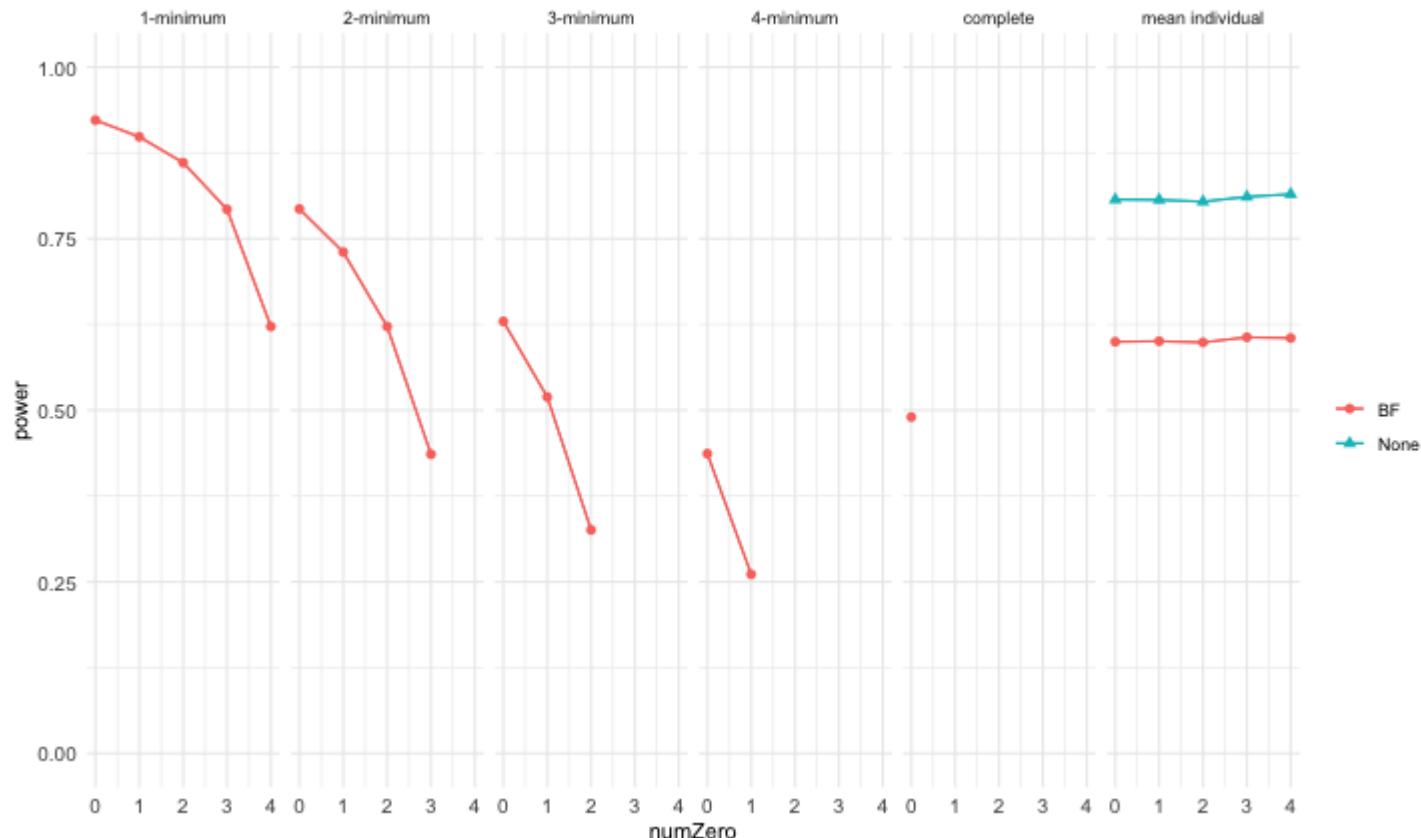
We can use the grid function to assess sensitivity to different model and design parameters.

Example: how many outcomes are assumed to have zero effect?

```
gridZero <- update_grid(  
  pow,  
  # vary parameter  
  numZero = 0:4,  
  # update number of outcomes  
  M = 5  
)
```

# Assessing sensitivity

```
plot( gridZero, nrow = 1 )
```



# Features for ease of use

Update function allows you to just update certain parameter values:

```
p_d <- update( pow,  
                 M = 5,  
                 R2.1 = c( 0.1, 0.3, 0.1, 0.2, 0.2 ))
```

Update grid:

```
gridICC <- update_grid( pow,  
                         ICC.2 = seq( 0, 0.3, 0.05 ))
```

# Bonus features

- Functions to simulate data from multilevel RCTs
- Function to estimate the approximate correlation between *test statistics* based on the correlation between *outcomes* (using a simulation approach)

```
covariate.corr.matrix <- gen_corr_matrix(M = 3, rho.scalar = 1)
cor.tstat <- check_cor(
  pow,
  rho.C = covariate.corr.matrix,
  n.sims = 500
)
est.cor <- mean(cor.tstat[lower.tri(cor.tstat)])
print( est.cor )

## [1] 0.374517
```

# Summary: PUMP R package

- Estimates power for multiple outcomes for multilevel RCTs
- Takes into account multiple testing procedures
- Calculates minimum detectable effect size (MDES) and sample size
- Allows user to assess sensitivity of power to different parameter choices

## Acknowledgments

- MDRC
- Institute of Education Sciences (Grant R305D170030)
- Harvard CARES Lab

# Available now!

- CRAN: CRAN.R-project.org/package=PUMP
- Shiny app: mdrc.shinyapps.io/pump
- Github: github.com/MDRCNY/PUMP
- arXiv: arxiv.org/abs/2112.15273
- Slides: github.com/kristenbhunter/presentations/tree/master/NCI2022
- Contact: kristenbhunter@gmail.com



Image by NYTimes