Sample Size Calculation with R

Generalized Linear Mixed Models

Dr. Mark Williamson, Statistician
Biostatistics, Epidemiology, and Research Design Core
DaCCoTA
Purpose

• This Module is a supplement to the Sample Size Calculation in R Module
• Gives the setup of Generalized Linear Mixed Models and Getting Sample Size Calculations
Background

- The Biostatistics, Epidemiology, and Research Design Core (BERDC) is a component of the DaCCoTA program.
- Dakota Cancer Collaborative on Translational Activity has as its goal to bring together researchers and clinicians with diverse experience from across the region to develop unique and innovative means of combating cancer in North and South Dakota.
- If you use this Module for research, please reference the DaCCoTA project.
Overview of Model Types

Level I:
    a) General linear models (lm): model with a normally distributed response variable (y) and predictor variables (x) with fixed effects

Level II:
    a) Generalized linear model (glm): model with non-normally distributed response variable (y) and predictor variables (x) with fixed effects
    b) General linear mixed model (lmer): model with a normally distributed response variable (y) and predictor variables (x) with fixed and/or random effects

Level III:
    a) Generalized linear mixed model (glmer): model with non-normally distributed response variable (y) and predictor variables (x) with fixed and/or random effects
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Range</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal (Gaussian)</td>
<td>Continuous</td>
<td>$-\infty &lt; x &lt; \infty$</td>
<td>$x$= dispersal from a central point, or diffusion through a Gaussian filter, with variance independent of mean</td>
</tr>
<tr>
<td>Log-normal</td>
<td>Continuous</td>
<td>$x &gt; 0$</td>
<td>$x$= probability distribution whose logarithm is normally distributed</td>
</tr>
<tr>
<td>Exponential</td>
<td>Continuous</td>
<td>$x &gt; 0$</td>
<td>$x$= time between events that occur at rate $\lambda = 1/\beta$</td>
</tr>
<tr>
<td>Gamma</td>
<td>Continuous</td>
<td>$x &gt; 0$</td>
<td>$x$= time it takes for $k$ event to occur with rate $\lambda = 1/\theta$, or the sum of $k$ exponential events</td>
</tr>
<tr>
<td>Beta</td>
<td>Continuous</td>
<td>$0 &lt; x &lt; 1$</td>
<td>$x$= distribution of probabilities based on $a$ successes and $b$ failures, when both $a$ and $b &gt; 1$</td>
</tr>
<tr>
<td>Binomial</td>
<td>Discrete</td>
<td>$x = 0, 1, 2...$</td>
<td>$x$= number of positive events out of $n$ trials each with a probability of success $p$</td>
</tr>
<tr>
<td>Geometric</td>
<td>Discrete</td>
<td>$x = 1, 1, 3...$</td>
<td>$x$= number of trials, with probability of success $p$, that are needed to obtain one success</td>
</tr>
<tr>
<td>Negative Binomial</td>
<td>Discrete</td>
<td>$x = 0, 1, 2...$</td>
<td>$x$= number of failures before $k$ successes occur in sequential independent trials, all with the same probability of success, $p$</td>
</tr>
<tr>
<td>Poisson</td>
<td>Discrete</td>
<td>$x = 0, 1, 2...$</td>
<td>$x$= count of items in a standardized unit of effort that occurs at rate $\lambda$</td>
</tr>
</tbody>
</table>
Notes on Effects

Terms:

Fixed effects:
- True effect size is same in all categories; summary effect is estimate of common effect size
- Categories with smaller sample size get less weight, ones with larger sample size get more weight
- Only source of uncertainty is within-study (sampling) error
- All categories {of interest} are present in the model
  - Example: Categories are all the state parks in a state

Random effects:
- True effect size varies across categories; effect sizes are random sample of effect sizes that could be observed; summary effect is estimate of the mean of these effects
- Categories are more balanced with weighting
- Uncertainty is within-study (sampling) error, plus between-studies variance
  - Leads to larger confidence intervals for summary effect
- Categories present in the model are subset of the total number of categories
  - Example: Categories are 10 state parks from states across the US

Description:
Combination of a Generalized Linear Model (GLM) and Mixed Model
• GLM: can be used with non-normal data
• Mixed Model: include both fixed and random effects
These models can be made very sophisticated and cover a very large range of models
• Need to understand how to create model and define variables
• We will create models with lme4
• We will get sample sizes with Simr
Description:

Provides functions to fit and analyze a variety of models, including generalized linear mixed models

https://cran.r-project.org/web/packages/lme4/lme4.pdf
https://cran.r-project.org/web/packages/lme4/vignettes/lmer.pdf
Package lme4: lmer

lmer(formula, data = NULL, REML = TRUE, control = lmerControl(), start = NULL, verbose = 0L, subset, weights, na.action, offset, contrasts = NULL, devFunOnly = FALSE, ...)

Formula:  y ~ x1 + x2 + (x1 | x3)
  y=response, x=terms, (|)=distinguish random-effects terms, separating expression for design matrices from grouping factors

Example:  lmer(Reaction ~ Days + (Days | Subject)
  Models Reaction time as a function of Days, with the Subject as a random effect

<table>
<thead>
<tr>
<th>Formula</th>
<th>Alternative</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1</td>
<td>g)</td>
<td>1 + (1</td>
</tr>
<tr>
<td>0 + offset(o) + (1</td>
<td>g)</td>
<td>-1 + offset(o) + (1</td>
</tr>
<tr>
<td>(1</td>
<td>g1/g2)</td>
<td>(1</td>
</tr>
<tr>
<td>(1</td>
<td>g1) + (1</td>
<td>g2)</td>
</tr>
<tr>
<td>x + (x</td>
<td>g)</td>
<td>1 + x + (1 + x</td>
</tr>
<tr>
<td>x + (x</td>
<td>g)</td>
<td>1 + x + (1</td>
</tr>
</tbody>
</table>

Table 2: Examples of the right-hand-sides of mixed-effects model formulas. The names of grouping factors are denoted g, g1, and g2, and covariates and a priori known offsets as x and o.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \eta )</td>
<td>Length of the response vector, ( \mathbf{Y} )</td>
<td></td>
</tr>
<tr>
<td>( p )</td>
<td>Number of columns of fixed-effects model matrix, ( \mathbf{X} )</td>
<td></td>
</tr>
<tr>
<td>( \beta )</td>
<td>Number of columns of random-effects model matrix, ( \mathbf{Z} )</td>
<td></td>
</tr>
<tr>
<td>( q_i )</td>
<td>Number of columns of the raw model matrix, ( \mathbf{X}_i )</td>
<td></td>
</tr>
<tr>
<td>( t_i )</td>
<td>Number of levels of the grouping factor indices, ( i )</td>
<td></td>
</tr>
<tr>
<td>( q_k )</td>
<td>Number of columns of the term-wise model matrix, ( \mathbf{Z}_k )</td>
<td></td>
</tr>
<tr>
<td>( k )</td>
<td>Number of random-effects terms</td>
<td></td>
</tr>
<tr>
<td>( m_i = \binom{t_i + 1}{2} )</td>
<td>Number of covariance parameters for term ( i )</td>
<td></td>
</tr>
<tr>
<td>( m = \sum m_i )</td>
<td>Total number of covariance parameters</td>
<td></td>
</tr>
</tbody>
</table>

Table 3: Dimensions of linear mixed models. The subscript \( i = 1, \ldots, k \) denotes a specific random-effects term.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \mathbf{X}_i )</td>
<td>( n \times p_i )</td>
<td>Raw random-effects model matrix</td>
</tr>
<tr>
<td>( J_i )</td>
<td>( n \times t_i )</td>
<td>Indicator matrix of grouping factors indices</td>
</tr>
<tr>
<td>( \mathbf{X}_j )</td>
<td>( p_i \times 1 )</td>
<td>Column vector containing jth row of ( \mathbf{X}_i )</td>
</tr>
<tr>
<td>( \mathbf{A}_j )</td>
<td>( t_i \times 1 )</td>
<td>Column vector containing jth row of ( J_i )</td>
</tr>
<tr>
<td>( \mathbf{Z}_i )</td>
<td>( m )</td>
<td>Covariance parameters</td>
</tr>
<tr>
<td>( T_i )</td>
<td>( p_i \times p_i )</td>
<td>Lower triangular template matrix</td>
</tr>
<tr>
<td>( \mathbf{A}_i )</td>
<td>( q_k \times q_k )</td>
<td>Term-wise relative covariance factor</td>
</tr>
</tbody>
</table>

Table 4: Symbols used to describe the structure of the random-effects model matrix and the relative covariance factor. The subscript \( i = 1, \ldots, k \) denotes a specific random-effects term.
Package lme4: lmer cont.

**lmer: fits a linear mixed-effects model to data**
- Consists of four modules

<table>
<thead>
<tr>
<th>Module</th>
<th>R function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Formula module</td>
<td>lFormula</td>
<td>Accepts a mixed-model formula, data, and other user inputs, and returns a list of objects required to fit a linear mixed model.</td>
</tr>
<tr>
<td>Objective function module</td>
<td>mkLmerDevfun</td>
<td>Accepts the results of lFormula and returns a function to calculate the deviance (or restricted deviance) as a function of the covariance parameters, $\theta$.</td>
</tr>
<tr>
<td>Optimization module</td>
<td>optimizeLmer</td>
<td>Accepts a deviance function returned by mkLmerDevfun and returns the results of the optimization of that deviance function.</td>
</tr>
<tr>
<td>Output module</td>
<td>mkMerMod</td>
<td>Accepts an optimized deviance function and packages the results into a useful object.</td>
</tr>
</tbody>
</table>

Table 1: The high-level modular structure of lmer.
glmer: fits a generalized linear mixed-effects model to data

A generalized linear mixed model incorporates both fixed-effects parameters and random effects in a linear predictor, via maximum likelihood. The linear predictor is related to the conditional mean of the response through the inverse link function defined in the GLM family.

glmer(formula, data = NULL, family = gaussian, control = glmerControl(), start = NULL, verbose = 0L, nAGQ = 1L, subset, weights, na.action, offset, contrasts = NULL, mustart, etastart, devFunOnly = FALSE, ...)

Formula:  \( y \sim x_1 + x_2 + (x_1 | x_3) \)

\( y \) = response, \( x = \) terms, (|) = distinguish random-effects terms, separating expression for design matrices from grouping factors

Family: a GLM family -> binomial(link = "logit"), gaussian(link = "identity"), Gamma(link = "inverse"), inverse.gaussian(link = "1/mu^2"), poisson(link = "log"), quasi(link = "identity", variance = "constant"), quasibinomial(link = "log"), quasipoisson(link = "log")

Use glmer.nb to fit negative binomial GLMMs

Offset: this can be used to specify an a priori known component to be included in the linear predictor during fitting. This should be NULL or a numeric vector of length equal to the number of cases

https://www.rdocumentation.org/packages/lme4 versions/1.1-21/topics/glmer
glmer: Examples (R-code)

```r
## generalized linear mixed model
library(lme4)
library(lattice)

str(cbpp)
xyplot(incidence/size ~ period | herd, cbpp, type=c('g','p','l'),
      layout=c(3,5), index.cond = function(x,y)max(y))

## response as a matrix
(m1 <- glmer(cbind(incidence, size - incidence) ~ period + (1 | herd),
             family = binomial, data = cbpp))

## response as a vector of probabilities and usage of argument "weights"
(m1p <- glmer(incidence / size ~ period + (1 | herd), weights = size,
              family = binomial, data = cbpp))

'data.frame': 56 obs. of 4 variables:
$ herd : Factor w/ 15 levels "1","2","3","4",..: 1 1 1 1 2 2 2 3 3 3 ...
$ incidence: num 2 3 4 0 3 1 1 8 2 0 ...
$ size : num 14 12 9 5 22 18 21 22 16 16 ...
$ period : Factor w/ 4 levels "1","2","3","4": 1 2 3 4 1 2 3 1 2 3 ...

Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) ['glmerMod']
Family: binomial ( logit )
Formula: cbind(incidence, size - incidence) ~ period + (1 | herd)
Data: cbpp
  AIC   BIC logLik deviance df.resid
194.0531 204.1799 -92.0266 184.0531       51

Random effects:
  Groups   Name        Std.Dev
  herd     (Intercept) 0.6421

Number of obs: 56, groups:  herd, 15

Fixed Effects:
(Intercept)      period2      period3      period4
        -1.3983       -0.9919       -1.1282       -1.5797

Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) ['glmerMod']
Family: binomial ( logit )
Formula: incidence/size ~ period + (1 | herd)
Data: cbpp
  AIC   BIC logLik deviance df.resid
194.0531 204.1799 -92.0266 184.0531       51

Random effects:
  Groups   Name        Std.Dev
  herd     (Intercept) 0.6421

Number of obs: 56, groups:  herd, 15

Fixed Effects:
(Intercept)      period2      period3      period4
        -1.3983       -0.9919       -1.1282       -1.5797
```
### glmer: Formula examples

<table>
<thead>
<tr>
<th>Formula</th>
<th>Alternative</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1</td>
<td>g)</td>
<td>1 + (1</td>
</tr>
<tr>
<td>0 + offset(o) + (1</td>
<td>g)</td>
<td>-1 + offset(o) + (1</td>
</tr>
<tr>
<td>(1</td>
<td>g1/g2)</td>
<td>(1</td>
</tr>
<tr>
<td>(1</td>
<td>g1) + (1</td>
<td>g2)</td>
</tr>
<tr>
<td>x + (x</td>
<td>g)</td>
<td>1 + x + (1 + x</td>
</tr>
<tr>
<td>x + (x</td>
<td></td>
<td>g)</td>
</tr>
</tbody>
</table>

Table 2: Examples of the right-hand-sides of mixed-effects model formulas. The names of grouping factors are denoted g, g1, and g2, and covariates and *a priori* known offsets as x and o.

[https://cran.r-project.org/web/packages/lme4/vignettes/lmer.pdf](https://cran.r-project.org/web/packages/lme4/vignettes/lmer.pdf)
**Package Simr: Description**

**Description:**
Allows user to calculate power for generalized linear mixed models from lme4 using Monte Carlo simulations

[https://cran.r-project.org/web/packages/simr/simr.pdf](https://cran.r-project.org/web/packages/simr/simr.pdf)

Version 1.0 of **SIMR** is designed for any LMM or GLMM fitted using lmer or glmer in the LME4 package, and for any linear or generalized linear model using lm or glm, and is focused on calculating power for hypothesis tests. In future versions we plan to:

- Increase the number of models supported by adding interfaces to additional R packages.
- Extend the package to include precision analysis for confidence intervals.
- Improve the speed of the package by allowing simulations to run in parallel
Package Simr: doTest

doTest: applies a hypothesis test to a fitted model

powerSim(object, test)

object=usually a fitted model

test=a test function

• fixed(xname, method=c("z", "t", "f", "chisq", "anova", "lr", "sa", "kr", "pb")
• compare(model, method=c("lr", "pb")
• fcompare(model, method=c("lr", "kr", "pb")
• rcompare(model, method=c("lr", "pb")
• random()

Example:

doTest(gm1, fixed("variable", "z")
Package Sim: powerSim

powerSim: estimates power by simulation

Performs a power analysis for a mixed model

```
powerSim(fit, test = fixed(getDefaultXname(fit)), sim = fit, fitOpts = list(), testOpts = list(), simOpts = list(), seed, ...)
```

- **fit** = fitted model object
- **test** = specifies test to perform
- **sim** = object to simulate from

**Example:**
```
fm1 <- lmer(y ~ x + (1|g), data=simdata)
powerSim(fm1, nsim=10)
```
Package Simr: powerCurve

powerCurve: estimate power at a range of sample sizes

This function runs powerSim over a range of sample sizes

\[
\text{powerCurve}(\text{fit, test = fixed(getDefaultXname(fit)), sim = fit, along = getDefaultXname(fit), within, breaks, seed, fitOpts = list(), testOpts = list(), simOpts = list(), ...})
\]

fit=fitted model object
test=specifies test to perform
sim=object to simulate from

Example:

```r
fm <- lmer(y ~ x + (1|g), data=simdata)
p1 <- powerCurve(fm)
p2 <- powerCurve(fm, breaks=c(4,6,8,10))
print(pc2)
plot(pc2)
```
Steps to GLMM power analysis

1) Get and describe data
2) Create model with lme4
   • confirm output and interpretation
3) Run model with SIMR
   • tweak/extend model as needed
4) Run power curve
## Examples

Each example will have three parts

I. Introduction
II. R-code
III. Results

### Examples

1. Basic Poisson
2. Binomial
3. Simulation 1
4. Simulation 2
5. Environmental

In R, make sure to install the lme4 and simr packages first and call them with the library function:

```r
>library(lme4)
>library(simr)
```
Example 1: Basic Poisson Introduction

- Adapted from *SIMR: an R package for power analysis of generalized linear mixed models by simulation*
- Research Question: Do prescription rates at a certain hospital differ by day, controlling for doctor?
- Modeling prescription counts for ten days each across three doctors
  - `glmer(Pr_rate ~ Day + (1|doctor), family="poisson", data=simdata)`
- Random intercept model where each group (doctor) has own intercept, but the doctors share a common trend
- Doctor is a random effect, while Day is a fixed effect

1) Get data
2) Create model
3) Run model
4) Power curve
Example 1: Basic Poisson R-Code

#1) GET DATA
> simdata2 <- read.csv("SIMR_RW_example.csv")  #download csv first
  > https://med.und.edu/daccota/_files/docs/simr_RW_example.csv or at end of slides
> head(simdata2)

#2) CREATE MODEL
> model1 <- glmer(Pr_rate ~ Day + (1|Doctor), family="poisson", data=simdata2)
> summary(model1)

change size of fixed effect
> fixef(model1)["Day"]  #view effect
> fixef(model1)["Day"] <- -0.06 #change effect to about half

#3) SIMULATE POWER
simulate power (given sample size of 10 per group)
> powerSim(model1)

extend the model to increase sample size (20 per group) and simulate power
> model2 <- extend(model1, along="Day", n=20)
> powerSim(model2)

#4) POWER CURVE
id minimum sample size required
> pc2 <- powerCurve(model2)
> print(pc2)
> plot(pc2)

varying the number and size of groups
adding more groups
> model3 <- extend(model1, along="Doctor", n=15)
> pc3 <- powerCurve(model3, along="Doctor")
> plot(pc3)

increase size within group
> model4 <- extend(model1, within="Day+Doctor", n=5)
> pc4 <- powerCurve(model4, within="Day+Doctor", breaks=1:5)
> print(pc4)
> plot(pc4)
Example 1: Basic Poisson Results*

- Power for first model = 42.70%  
  \(-\text{not enough}\)
- Power for second model = 98.30%  
  \(-\text{overkill}\)
- Minimum sample size found to be about 14 Days \((\text{for 3 Doctors})\) for a power of 80%  
  \(\text{(figure 1)}\)
- Minimum group size found to be about 8 Doctors \((\text{each with 10 Days})\) for a power of 80%  
  \(\text{(figure 2)}\)
- Need 3 repeats \((\text{3 Doctors X 10 Days})\) to get a power of 80%  
  \(\text{(figure 3)}\)

*will be mildly different each run based on simulation seed
Example 2: Binomial Introduction

- Taken from *SIMR: an R package for power analysis of generalized linear mixed models by simulation: Appendix 1*


- Research Question: Is the incidence of cbpp (contagious bovine pleurophenomia) different across period or size?

- Modeling Incidence across periods and size while controlling for herds

- `glmer(cbind(incidence, size - incidence) ~ period + size + (1 | herd), data=cbpp, family=binomial)`

- Size is a continuous predictor (effect), period is a fixed effect, and herd is a random effect

1) Get data
2) Create model
3) Run model
4) Power curve
Example 2: Binomial R Code

#1) GET DATA
> head(cbpp) #dataset found in lme4

#2) CREATE MODEL
*Period only*
> gm1 <- glmer(cbind(incidence, size - incidence) ~ period + (1 | herd),
  data=cbpp, family=binomial)
> doTest(gm1, fixed("period", "lr"))

*Add size to model*
> gm2 <- glmer(cbind(incidence, size - incidence) ~ period + size + (1 | herd),
  data=cbpp, family=binomial)
> doTest(gm2, fixed("size", "z"))

#3) SIMULATE POWER
*Period model*
> powerSim(gm1, fixed("period", "lr"),
  nsim=50)

*Size model*
> powerSim(gm2, fixed("size", "z"),
  nsim=50)

*change size of fixed effect (bigger)*
> fixef(gm2)["size"] <- 0.05
> powerSim(gm2, fixed("size", "z"),
  nsim=50)

#4) POWER CURVE
*Double the number of herds*
> gm3 <- extend(gm2, along="herd", n=30)
> powerSim(gm3, fixed("size", "z"),
  nsim=50)

*Power curve not applicable here*
Example 2: Binomial Results*

- Power for period model=92.00%  <- more than enough power
- Power for size model with original effect size=10.00%  <- need more power
- Power for size model with larger effect size=58%^  <- still need more power, try extending herds
- Power for size model with larger effect size and double the number of herds=80.00%  <- good enough

* will be mildly different each run based on simulation seed
^was significantly higher than the appendix example, despite all the other numbers similar, so not sure why
Example 3: Simulation w/o Data Introduction

- Taken from https://humburg.github.io/Power-Analysis/simr_power_analysis.html

- You can simulate without real data

- Research Question: Is there a difference between a treatment and control for schoolchildren across classes

- Model-> \( y \sim \text{treatment} + \text{time} + \text{treatment x time} + (1|\text{class/id}) + \epsilon \)

- Testing the effect of treatment, time, the interaction of treatment and time on our \( y \) variable, with class and id as random effects and id nested inside class

1) Get data
2) Create model
3) Run model
4) Power curve
Example 3: Simulation w/o Data R Code

#1) GET DATA

create covariates
> subj <- factor(1:10)
> class_id <- letters[1:5]
> time <- 0:2
> group <- c("control", "intervention")
> subj_full <- rep(subj, 15)
> class_full <- rep(rep(class_id, each=10), 3)
> time_full <- rep(time, each=50)
> group_full <- rep(rep(group, each=5), 15)
> covars <- data.frame(id=subj_full, class=class_full, treat=group_full, time=factor(time_full))
> covars

Intercept and slopes for intervention, time1, time2, intervention:time1, intervention:time2
> fixed <- c(5, 0, 0.1, 0.2, 1, 0.9)

Random intercepts for participants clustered by class
> rand <- list(0.5, 0.1)

residual variance
> res <- 2

#2) CREATE MODEL

> model <- makeLmer(y ~ treat*time + (1|id), fixef=fixed, VarCorr=rand, sigma=res, data=covars)
> model

#3) SIMULATE POWER

> sim_treat <- powerSim(model, nsim=100, test = fcompare(y~time))
> sim_treat
> sim_time <- powerSim(model, nsim=100, test = fcompare(y~treat))
> sim_time

#4) POWER CURVE

changing effect size
> model_large <- model
> fixef(model_large)["treatintervention:time1"] <- 2
> sim_treat_large <- powerSim(model_large, nsim=100, test = fcompare(y~time))
> sim_treat_large

changing number of classes
> model_ext_class <- extend(model, along="class", n=20)
> model_ext_class
> sim_treat_class <- powerSim(model_ext_class, nsim=100, test = fcompare(y~time))
> sim_treat_class
> p_curve_treat <- powerCurve(model_ext_class, test=fcompare(y~time), along="class")
> plot(p_curve_treat)

changing number within classes
> model_ext_subj <- extend(model, within="class+treat+time", n=20)
> model_ext_subj
> sim_treat_subj <- powerSim(model_ext_subj, nsim=100, test = fcompare(y~time))
> sim_treat_subj
> p_curve_treat <- powerCurve(model_ext_subj, test=fcompare(y~time), within="class+treat+time", breaks=c(5,10,15,20))
> plot(p_curve_treat)

changing both
> model_final <- extend(model, along="class", n=8)
> model_final
> model_final <- extend(model_final, within="class+treat+time", n=10)
> sim_final <- powerSim(model_final, nsim=100, test = fcompare(y~time))
> sim_final
Example 3: Simulation w/o Data Results*

- Power for base treatment model = 33.00%  <-not enough power
- Power for base time model = 37.00%  <-not enough power
- Power for treatment model (> effect size) = 95.00%  <-enough power
- Power for treatment model (more classes) = 94.00%  <-enough power
  - Would need 13 classes to have 80% power (figure 1)
- Power for treatment model (more within class) = 89.00%  <-enough power
  - Would need 15 students within each class to have 80% power (figure 2)
- Power for treatment model (more + more w/ i classess) = 88.00%  <-enough power

*will be mildly different each run based on simulation seed
Example 4: Simulation w/o Data 2 Introduction

- Taken from [http://www.alexanderdemos.org/Mixed9.html](http://www.alexanderdemos.org/Mixed9.html)
- You can simulate data that is similar to real data
- Research Question: Is there a difference in the dependent variable (DV) across Conditions, controlling for Subject and Item?
- Model-> lmer(DV ~ Condition1 + (Condition1|Subject)+(Condition1|Item))
- Condition is a fixed effect, while Subject and Item are random effects
**Example 4: Simulation w/o Data 2 R Code**

### #1 GET DATA

- `Item <- as.factor(rep(1:10))`
- `Subject <- as.factor(rep(1:30))`
- `Condition1 <- rep(-.5:.5)`
- `X <- expand.grid(Subject=Subject, Item=Item, Condition1=Condition1)` 
  #creates a ‘frame’
- `b <- c(10.1912, 4.7773)`
- `SubVC <- matrix(c(23.471, 0, 0, 3.98), 2)`
- `ItemVC <- matrix(c(2.197, 0, 0, 5.009), 2)`
- `s <- 7.13^.5`

### #2 CREATE MODEL

Next, make a `lmer` object (feed in fixed effects, random effects, residual, and the frame of the data)

- `SimModel <- makeLmer(DV ~ Condition1 + (Condition1|Subject)+(Condition1|Item), fixef=b, VarCorr=list(SubVC,ItemVC), sigma=s, data=X)`
- `summary(SimModel)`

### #3 SIMULATE POWER

Test the number of subjects needed

- `Item <- as.factor(rep(1:20))`
- `Subject <- as.factor(rep(1:80))`
- `Condition1 <- rep(-.5:.5)`
- `X <- expand.grid(Subject=Subject, Item=Item, Condition1=Condition1)`
- `b2 <- c(0, .112)`
- `SubVC2 <- matrix(c(.368, 0, 0, .004), 2)`
- `ItemVC2 <- matrix(c(.068, 0, 0, .001), 2)`
- `s2 <- (3.59)^.5`  
  # residual sd
- `SimCurve <- makeLmer(DV ~ Condition1 + (Condition1|Subject)+(Condition1|Item), fixef=b2, VarCorr=list(SubVC2,ItemVC2), sigma=s2, data=X)`
- `summary(SimCurve)`
- `SCurve1 <- powerCurve(SimCurve, fixed("Condition1", "lr"), along="Subject", breaks=c(20,40,60,80), nsim=100, alpha=.045, progress=FALSE)`
- `plot(SCurve1)`

Test across number of Items

- `SCurve2 <- powerCurve(SimCurve, fixed("Condition1", "lr"), along="Item", breaks=c(5,10,15), nsim=100, alpha=.045, progress=FALSE)`
- `plot(SCurve2)`

### #4 POWER CURVE

Test the number of subjects needed

- `Item <- as.factor(rep(1:20))`
- `Subject <- as.factor(rep(1:80))`
- `Condition1 <- rep(-.5:.5)`
- `X <- expand.grid(Subject=Subject, Item=Item, Condition1=Condition1)`
- `b3 <- c(0, .05, -.05, .1)`  
  # fixed intercept and slope
- `SubVC3 <- diag(c(.35,.005,.005,.005))`
- `ItemVC3 <- diag(c(.1,.005,.005,.005))`  
  # random intercept and slope variance-covariance matrix
- `s3 <- (1-(sum(SubVC3)+sum(ItemVC3)))^.5`  
  # residual sd
- `SimInter <- makeLmer(DV ~ C1*C2 + (C1*C2|Subject)+(C1*C2|Item), fixef=b3, VarCorr=list(SubVC3,ItemVC3), sigma=s3, data=X)`
- `summary(SimInter)`
- `SimPower.Inter <- powerSim(SimInter, fixed("C1*C2", "lr"), nsim=100, alpha=.045, progress=FALSE)`
- `plot(SimPower.Inter)`
Example 4: Simulation w/o Data 2 Results*

- Power for base model = 100.0% <- based on observed power, so not surprising
- The number of subjects needed was over 55 (figure 1)
- The number of items needed was 14 (figure 2)
- Power for the Interaction term = 19.00% <- weak interaction

*will be mildly different each run based on simulation seed
Example 5: Environmental Data Introduction

- Taken from [http://environmentalcomputing.net/power-analysis/](http://environmentalcomputing.net/power-analysis/)
- Research Question: Is there an effect of estuary modification on the abundance of a marine species?
- Model: \( \text{glmer}(\text{abundance} \sim \text{temperature} + \text{modification} + (1|\text{site})) \)
- Temperature is a continuous fixed effect, modification is a fixed effect, and site is a random effect (nested in modification)
Example 5: Environmental Data R Code

#1) GET DATA
> Pilot <- read.csv(file = "Pilot.csv", header = TRUE) #get csv

Change modification into factor
> Pilot$modification <- factor(Pilot$modification)
> par(mfrow=c(1,2))
> boxplot(abundance~modification,data=Pilot,main="modification")
> boxplot(abundance~site,data=Pilot,main="site")
> par(mfrow=c(1,1))

#2) CREATE MODEL
> Pilot.mod <- glmer(abundance ~ temperature + modification + (1|site), family="poisson", data=Pilot)
> summary(Pilot.mod)
> fixef(Pilot.mod)["modification2"] <- 0.1 #lowers from 1.14
> fixef(Pilot.mod)["modification3"] <- 0.2 #lowers from 1.24

#3) SIMULATE POWER
Pilot study model
> powerSim(Pilot.mod, fixed("modification", "lr"), nsim=50)
> xtabs(~modification+site,data=Pilot)

#4) POWER CURVE
Full model (extend by increasing number of sites)
> full1 <- extend(Pilot.mod, along="site", n=24)
> xtabs(~site,data=attributes(full1)$newData)
> powerSim(full1, fixed("modification", "lr"), nsim=50)

Full model 2 (extend by increasing observations w/i sites)
> full2 <- extend(Pilot.mod,within="site",n=12)
> xtabs(~site,data=attributes(full2)$newData)
> powerSim(full2, fixed("modification", "lr"), nsim=50)

Full model 3 (extend both number of sites and obs. w/i sites)
> full3 <- extend(Pilot.mod,within="site",n=6)
> full3 <- extend(full3,along="site",n=12)
> xtabs(~site,data=attributes(full3)$newData)
> powerSim(full3, fixed("modification", "lr"), nsim=50)
Example 5: Environmental Data Results*

- Power for pilot model = 52.00% < low, but pilot studies are designed to be underpowered
- Power for full model 1 (increase in sites) = 78.00% < good, close to needed power
- Power for full model 2 (increase within) = 68.00% < not as good
- Power for full model 3 (increase in and within sites) = 74.00% < better, but not as good as first
- In summary, these results showed that it was better to increase the number of sites used rather than the number of observations within sites for the full study

*will be mildly different each run based on simulation seed
Acknowledgements

• The DaCCoTA is supported by the National Institute of General Medical Sciences of the National Institutes of Health under Award Number U54GM128729.

• For the labs that use the Biostatistics, Epidemiology, and Research Design Core in any way, including this Module, please acknowledge us for publications. "Research reported in this publication was supported by DaCCoTA (the National Institute of General Medical Sciences of the National Institutes of Health under Award Number U54GM128729)."
References

- https://cran.r-project.org/web/packages/lme4/lme4.pdf
- https://cran.r-project.org/web/packages/lme4/vignettes/lmer.pdf
- https://www.rdocumentation.org/packages/lme4/versions/1.1-21/topics/glmer
- https://cran.r-project.org/web/packages/simr/simr.pdf
- https://humburg.github.io/Power-Analysis/simr_power_analysis.html
- http://environmentalcomputing.net/power-analysis/
<table>
<thead>
<tr>
<th>Cancer, Day, Doctor, Pr_rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.139081,1,a,3</td>
</tr>
<tr>
<td>8.947861,2,a,3</td>
</tr>
<tr>
<td>9.283638,3,a,3</td>
</tr>
<tr>
<td>7.779489,4,a,2</td>
</tr>
<tr>
<td>5.803512,5,a,3</td>
</tr>
<tr>
<td>6.13172,6,a,2</td>
</tr>
<tr>
<td>6.122911,7,a,3</td>
</tr>
<tr>
<td>7.542655,8,a,0</td>
</tr>
<tr>
<td>6.428379,9,a,2</td>
</tr>
<tr>
<td>6.219345,10,a,1</td>
</tr>
<tr>
<td>9.17015,1,b,2</td>
</tr>
<tr>
<td>8.253626,2,b,3</td>
</tr>
<tr>
<td>10.346381,3,b,2</td>
</tr>
<tr>
<td>8.807318,4,b,1</td>
</tr>
<tr>
<td>6.09285,5,b,1</td>
</tr>
<tr>
<td>8.510823,6,b,3</td>
</tr>
<tr>
<td>7.086676,7,b,2</td>
</tr>
<tr>
<td>6.241644,8,b,3</td>
</tr>
<tr>
<td>6.841693,9,b,0</td>
</tr>
<tr>
<td>5.783463,10,b,2</td>
</tr>
<tr>
<td>13.697234,1,c,9</td>
</tr>
<tr>
<td>13.551086,2,c,7</td>
</tr>
<tr>
<td>12.239432,3,c,6</td>
</tr>
<tr>
<td>14.513912,4,c,2</td>
</tr>
<tr>
<td>13.579498,5,c,2</td>
</tr>
<tr>
<td>12.037988,6,c,4</td>
</tr>
<tr>
<td>14.17994,7,c,3</td>
</tr>
<tr>
<td>13.102589,8,c,4</td>
</tr>
<tr>
<td>12.131053,9,c,2</td>
</tr>
<tr>
<td>13.071251,10,c,2</td>
</tr>
</tbody>
</table>