# simr : an R Package for Power Analysis of Linear Mixed Models by Simulation

## Introduction

### Why we want to do power analysis.

Environmental monitoring is increasingly used to assess spatial and temporal trends in variables of interest (e.g. population decline, pollution increase) as well as the effectiveness of management policies (Yoccoz et al. Butchart et al. etc). A fundamental aim of environmental monitoring is to ensure that real change is detected and acted upon as promptly as possible. Detecting changes in ecological systems, however, is often technically and logistically challenging, especially when resources are limited. Careful planning and executing of sophisticated analyses of monitoring data are recommended for informing cost-effective and robust monitoring (Field et al. 2007). Such analyses can be used to identify monitoring efforts that have no realistic chance of detecting relevant changes and options for improving them (Legg & Nagy 2006; Field et al. 2007).

The extent and strength of interferences that can be drawn from monitoring programmes depends on their scale, design and intensity (Yoccoz et al. 2001). A robust monitoring design is one with sufficient statistical power to detect a specified change of interest if it actually occurs. This requires: (a) having a sufficient sample size in relation to the variability inherent in the system; (b) setting an ecologically appropriate level of power as a target; and (c) implementing a flexible design that allows for learning and improvement in the future (Field et al. 2007).

While there are a range of tools available for carrying out power analyses there is currently no single tool available that can automate the power analysis process based on arbitrary models fit with lme4. This paper describes a package simr, now available in R, which provides tools that make it simple to set up and run simulation experiments to help better inform, assess and improve monitoring designs. Key features of the package are as follows:

1. *Applicable to both simple and complex survey designs*: With simr you can perform a power analysis based on any linear mixed model fitted with the lmer command in the lme4 package. lme4 handles both crossed and nested designs, and so simr can handle both these classes of linear mixed model. When the design is complicated (e.g. spatial and/or temporal nested sampling protocols are implemented), linear mixed models allow us to model the random variation among units as well as the effect of explanatory variables [Bolker 2008].
2. *Flexibility to assess trade-offs in survey design*: Simulation is a general procedure for determining the properties of a statistical method or design [ref]. With simulation we do not need to find an analytical approach specialised to our particular analysis, and we do not need to rely on an approximation that may not be robust to departures from its assumptions. This can be especially important with mixed models where the sampling distributions of parameter estimates are difficult to work with [ref Bates?].
3. *Suitable for a range of analytical capabilities*: Simulation studies can be difficult or time consuming to set up. They would normally involve some degree of programming by the investigator. Depending on their comfort level, this might be beyond their current ability or might simply take longer than they might like. The ready-made package makes power analyses accessible to a wide range of scientists. Investigators who might otherwise be limited to fitting a model in R [ref] could supplement their study with a power analysis without much additional effort. For scientists comfortable with R coding, a turnkey package could save them time and let them focus their efforts on a more advanced analysis.

several[Describe the tutorials: The second …]. The tutorials are intended to provide a shallow learning curve. Increasingly complex analyses are made available, adding only a few commands or options at a time.

## Example Dataset

The example dataset has response variable y measured at ten levels of the explanatory variable x for three groups g. [change y, x, g to meaningful names?]

## 

## Post-hoc power analysis for a specified sampling design.

We can fit a simple random intercepts model as follows

library(simr) # note that this will also load lme4

fit <- lmer(y ~ x + (1|g), data=example)

show and discuss output

Test power of current design ……..

power(fit)

show output [explanation of the result; where it comes from, what it means]

## Assess trade-offs in sampling design and power

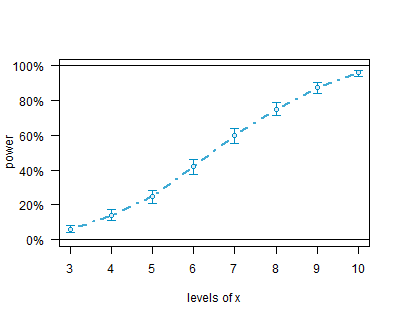
Calculating and plotting a power curve in simr is simple:

pc <- powercurve(fit)

plot(pc)

Note that this may take some time; we are analysing [R] simulated datasets, and for each of those simulated datasets we are fitting the model to 8 different subsets (we are varying the sample size from 3—10 levels of x).

The result is shown in figure [two].



## Tutorial Three: Specify the sample and effect sizes.

The first two analyses made heavy use of the default settings in simr to keep things as simple as possible. For example, the simulated trend defaults to the trend estimated in the fitted model. Often we will have a specific value for an ecologically significant effect, and we are interested in the power to detect and effect of that size. We can access the fixed effects in an lmm with:

fixef(fit)

If we want a specific fixed effect, say the effect for x we use:

fixef(fit)[‘x’]

Suppose that our ecologically significant effect size for x is -0.1. simr uses the (hopefully) obvious idiom to change the size of fixed effects:

fixef(fit)[‘x’] <- -0.1

We can now calculate a power curve for our modified fitted model the same way we did for [tutorial two]:

pc3a <- powercurve(fit)

plot(pc3a)

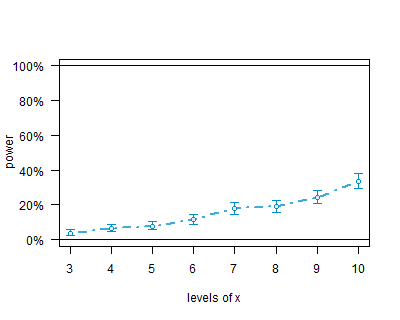
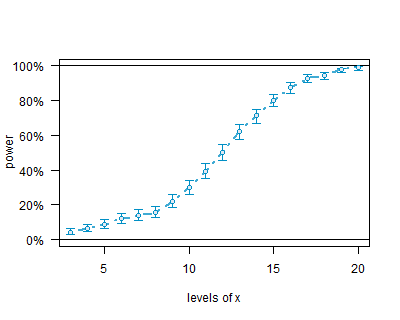
Again, this will take some time. However you can see the results now in figure [3a]. This analysis shows us that we have insufficient power at this effect size for any of the sample sizes considered. To get a better picture of the trade-off between power and sample size we need to increase the number of levels of x, which we can do using the extend command:

fit <- extend(fit, along=’x’, n=20)

pc3b <- powercurve(fit)

plot(pc3b)

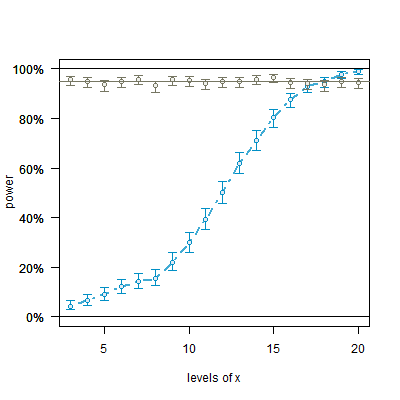
Figure [3b] shows the power curve for a larger range of levels of x (3—20).



## Tutorial Four: Include Type I error calculations.

pc4 <- powercurve(fit, null=TRUE)

plot(pc4)



## Other Features

simrsince there are analytic formulae for the linear model case, this feature sus to write simpler tutorials.

## Further Work.

As currently implemented, simr uses data from a pilot study to inform the power analysis, with the structure of the pilot data providing defaults for the simulation settings. Future versions will also include the ability to create data sets from scratch; this will not require a pilot study, but will require some domain expertise to select sensible parameters.

Version 1.0 of simr is designed for linear mixed models using lmer in lme4. The next version will add support for generalised linear models (glm in base) and generalised linear mixed models (glmm in lme4). At some point tools will be added to make it simple to create interfaces to arbitrary R packages.

## Notes: Which Dataset to Use?

Simulated dataset: conceptually simple, allows the package to be the focus of the paper.

Real dataset: allows practical scientists to get a handle on what’s actually happening. But we have to deal with introducing the data and with any peculiarities; this all takes away from the main focus of the paper.

Simulated “Real” Dataset: Combines advantages of both approaches. We generate a clean simulated dataset; this means we spend no time explaining data cleaning assumptions. But the variables are given meaningful names so that people can develop an intuitive grasp of what’s going on.

Table 1:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Survey design |  | Data processing skills |  |  | Package characteristics |  |  | Reference |
| Package | Nested design | Trade-offs in alternative survey designs? | Ease of use? | Trend? |  | Cope with nested designs? | Approximation? | Software |  |
|  |  |  | Availability of tutorials | Write own functions |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |