Automated Power Analysis Simulations in R

(This poster describes the prototype for `simr` which is still under development.)

Peter Green, Catriona MacLeod

Landcare Research, Dunedin, New Zealand

greenp@landcareresearch.co.nz

Introduction

If we are going to invest in the collection of monitoring data, it is important that we know that our study design is suitable for making the inferences we need. Does the design have sufficient power to detect what we are trying to find?

When the design is complicated, e.g. in a mixed-effects model, simulation provides a general way of answering these questions (Gelman and Hill, 2007; Bolker, 2008). The `simr` package for R (R Core Team, 2013) provides tools that make it simple to set up and run these kinds of simulation experiments. It aims to provide a gentle learning curve for anyone already familiar with fitting linear mixed models in `lme4` (Bates et al., 2013).

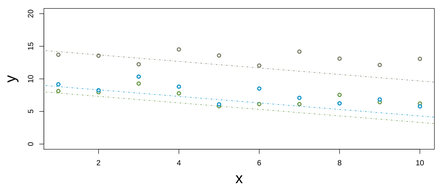
Why an R Package?

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 would like. Having a ready-made package available will make power analyses accessible to a wider range of scientists. Investigators who might otherwise be limited to fitting a model in R could supplement their study with a power analysis with minimal additional effort. For scientists who are already comfortable with R coding, a turnkey package will save them time and let them focus their efforts on a more advanced analysis.

This work is part of the New Zealand Sustainability Dashboard which is a project of the Agricultural Research Group on Sustainability.

Calculating a Power Curve

The `example` dataset has response variable `y` measured at ten levels of the explanatory variable `x` for three groups `g`.

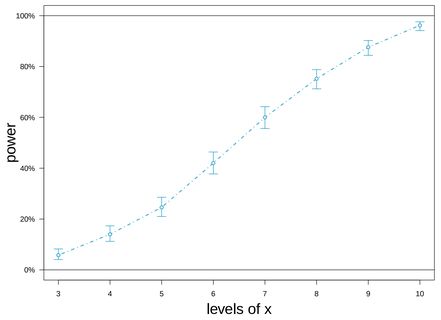


A power analysis by simulation would normally involve multiple steps, each requiring the user to write some amount of R code.

- Generate multiple simulated datasets.
- Refit the model to (subsets of) the new data.
- Apply statistical tests to the fitted models.
- Collate and report on the results.

With `simr`, these steps are automated and a power curve can be calculated and plotted with just a few lines of code:

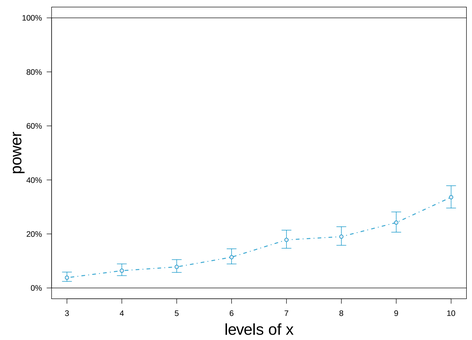
```
library(simr)
fit <- lmer(y ~ x + (1|g),
            data=example)
pa <- power(fit)
plot(pa)
```



Modifying Parameters

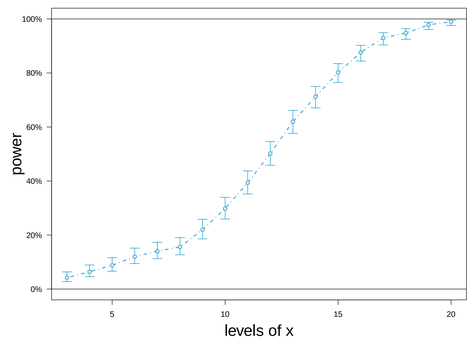
The default analysis calculates our power to detect the observed trend. If we need to detect a smaller trend we can modify the parameters of the fitted model.

```
fixef(fit)['x'] <- -0.10
plot(power(fit))
```



This analysis shows that we have insufficient power for any of the sample sizes considered. We can extend the maximum number of levels of `x` in the analysis:

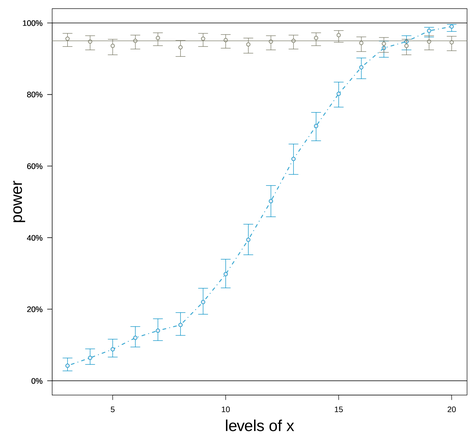
```
fit <- extend(fit, along='x', n=20)
plot(power(fit))
```



Significance

Simulation has applications beyond power analysis. For example, we might want to calculate the significance level of our test so that we can compare it to its nominal value.

```
plot(power(fit, null=TRUE))
```



References

Douglas Bates, Martin Maechler, Ben Bolker, and Steven Walker. *lme4: Linear mixed-effects models using Eigen and S4*, 2013. URL `http://CRAN.R-project.org/package=lme4`. R package version 1.0-5.

Benjamin M Bolker. *Ecological models and data in R*. Princeton University Press, 2008.

Andrew Gelman and Jennifer Hill. *Data analysis using regression and multilevel/hierarchical models*. Cambridge University Press, 2007.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2013. URL `http://www.R-project.org/`.