A Tutorial on Tailored Simulation-Based Power Analysis for Experimental Designs with Generalized Linear Mixed Models

Timo K. Koch[†,1, 2], Florian Pargent[†,1], Anne-Kathrin Kleine[1], Eva Lermer[1,3], & Susanne Gaube[1,4]

[1] LMU Munich, Department of Psychology

[2] University of St. Gallen, Institute of Behavioral Science & Technology

[3] Technical University of Applied Sciences Augsburg, Department of Business Psychology

[4] University College London, Global Business School for Health

Author Note

This research was funded by a grant from the Volkswagen Foundation (Grant No. 98525). All materials (reproducible manuscript incl. R code, simulation results) are available in the project's repository on the Open Science Framework (OSF): https://osf.io/dhwf4/

The authors made the following contributions. Timo K. Koch: Conceptualization, Formal Analysis, Writing - Original Draft Preparation, Writing - Review & Editing; Florian Pargent: Formal Analysis, Methodology, Visualization, Writing - Original Draft Preparation, Writing - Review & Editing; Anne-Kathrin Kleine: Writing - Review & Editing; Eva Lermer: Resources; Susanne Gaube: Writing - Review & Editing, Supervision. † Timo K. Koch and Florian Pargent contributed equally to this work.

Correspondence concerning this article should be addressed to Timo K. Koch, Torstrasse 25, 9000 St. Gallen, Switzerland. E-mail: timo.koch@unisg.ch

Abstract

When planning experimental research, determining an appropriate sample size and using suitable statistical models are crucial for robust and informative results. However, the recent replication crisis in Human-Computer Interaction (HCI) and other empirical research fields underlines the need for more rigorous statistical methodology and well-powered designs. Generalized linear mixed models (GLMMs) offer a flexible statistical framework to analyze experimental data with complex (e.g., dependent and hierarchical) data structures. Yet, analytic methods and software cannot be applied to conduct a priori power analyses for GLMMs, necessitating data simulation approaches. Based on a practical case study, the current tutorial equips researchers with a step-by-step guide and corresponding code for conducting tailored a priori power analyses to determine appropriate sample sizes with GLMMs. Finally, we give an outlook on the increasing importance of simulation-based power analysis in experimental research.

Keywords: power analysis, data simulation, sample size, generalized linear mixed model

Word count: 7600

A Tutorial on Tailored Simulation-Based Power Analysis for Experimental Designs with Generalized Linear Mixed Models

Introduction

When planning experimental research, it is essential to determine an appropriate sample size to ensure that the results obtained are both robust and informative, and to use appropriate statistical models to analyze the data (Lakens, 2022b). However, the recent replication crisis in Human-Computer Interaction (HCI) and several other disciplines grounded on empirical research has illustrated many challenges surrounding the reproducibility and reliability of findings (Cockburn, Dragicevic, Besançon, & Gutwin, 2020; Robertson & Kaptein, 2016; Yarkoni, 2022). As a result, there is a growing need for more rigorous statistical methodology and the adoption of well-powered experimental designs. While software solutions exist for simple statistical models and experimental designs, many researchers lack the skills and tools to conduct "a priori" (i.e., before data collection) power analyses for more complex research designs using the flexible generalized linear mixed models (GLMM) framework in order to determine the required sample size in their experiments. In the present work, we provide a tutorial consisting of a concrete example for tailored a priori power analyses using data simulations based on GLMMs.

Statistical power

In empirical research relying on hypothesis testing, the most common strategy for determining an adequate sample size is based on statistical power (Lakens, 2022b). Statistical power is defined as the probability that a hypothesis test has a significant p-value when analyzing repeated samples from a population with a true effect of some pre-specified size. Less formally, power is described as the probability that a hypothesis test correctly rejects the null hypothesis when the alternative hypothesis is true. If the sample size (i.e., the number of participants and/or stimuli) used for data collection is insufficient to detect the effects or relationships being investigated with high probability, the study would be considered “underpowered”.

Conducting underpowered research has many negative consequences. First, relying on underpowered experiments may yield inconclusive (if researchers acknowledge the small evidential value of an underpowered study in the limitations section) or misleading (if low power is ignored by the researchers) results, hindering the accumulation of knowledge. Second, underpowered studies waste resources by consuming time, effort, and funding without delivering meaningful results.

A priori power analysis

A power analysis represents the act of calculating the statistical power for a given true effect and sample size. When running a power analysis before data collection, the required sample size can be determined so that researchers find an assumed true effect with the desired statistical power.

Thereby, a priori power analysis offers a valuable contribution to the research process by allowing researchers to estimate the appropriate sample sizes required to achieve sufficient statistical power for results with high evidential value. Moreover, conducting a careful a priori power analysis helps researchers decide which experimental design and statistical models are both feasible and appropriate for analyzing the data and answering their research questions. Also, when conducting a proper power analysis, researchers have to consider every aspect of the experimental design and will notice statistical or design challenges before starting with data collection. Adding a solid sample size calculation to the research process can act as a safeguard for ensuring high-quality research. Finally, many journals and funding agencies now require that a power analysis is included in study protocols and grant proposals, recognizing its significance in ensuring robust and meaningful findings.

For simple statistical models, like t-tests, ANOVA, and linear regression, with common study designs (e.g., mean comparison between two groups), user-friendly software for a priori power analysis is readily available (Champely et al., 2018; Faul, Erdfelder, Buchner, & Lang, 2009). However, these software packages are often not flexible enough to perform power analysis for complex designs.

Generalized linear mixed models (GLMMs)

As study designs become more complex, researchers require more sophisticated statistical models to capture the nuanced relationships and grouping structures introduced by their study designs (Yarkoni, 2022). GLMMs (also called multilevel models) are gaining increasing popularity in analyzing data in HCI and other empirical disciplines because they offer a flexible framework for handling data with outcome variables that are not normally distributed (e.g., categorical outcomes) while accounting for both fixed and random effects (Fahrmeir, Kneib, Lang, & Marx, 2021; Kaptein, 2016).

GLMMs are an extension of LMMs (Linear Mixed Models), which are, in turn, extensions of linear regression models that account for correlated data including hierarchical structures (Fahrmeir et al., 2021). In this context, correlated data means that the value in the outcome variable for one observation may be related (i.e., more similar or less similar) to the value for another observation in a systematic way that is not already accounted for by the usual (fixed) predictor variables (e.g., age of participants). This correlation can arise for various reasons: Responses to some stimuli from some participants might be more similar because the same person was measured twice (repeated measurements), both participants come from the same neighborhood (clustering) or both participants responded to the same stimulus (stimulus effects). Thus, modeling such correlations is especially important whenever the data has a clear structure, while the grouping variables can be hierarchically organized (e.g., students nested in schools, schools nested in districts) or not (e.g., students solve math exercises, but neither student sees all exercises). LMMs are used when the outcome variable is continuous and follows a normal distribution (when conditioned on all fixed and random effects). They allow for the modeling of fixed effects, which capture the relationships between our usual predictors and the outcome, as well as random effects, which account for the different types of correlation structure and grouping effects exemplified above. Random effects are typically assumed to follow a normal distribution with a mean of zero and a variance that quantifies the heterogeneity across groups.

As mentioned, GLMMs extend the LMM framework to accommodate non-normally distributed continuous and categorical outcome variables. GLMMs incorporate both fixed and random effects, similar to LMMs, but also involve a link function that connects the linear combination of predictor variables to the expected value of the outcome variable. The link function allows for modeling the relationship between predictors and the outcome in a non-linear way that is appropriate for the specific distribution family of the outcome variable. As an example, think of an experiment with different design factors (e.g., picture, headline) impacting the likelihood of users clicking on an online advertisement. Here, participants' behavior is measured repeatedly (e.g., over several sessions). The click patterns of participants in one session are likely to be correlated with their previous sessions. Finally, the outcome variable is binary (click/no click) for each interaction, which follows a binomial distribution.

Power analysis for GLMMs

Power analysis methods for multilevel models can be categorized into formula-based methods and simulation-based methods (Murayama, Usami, & Sakaki, 2022). Formula-based methods rely on often complicated formulas that can be used to directly calculate power while simulation-based methods rely on repeatedly simulating data with a known true effect size and estimating power empirically (i.e., how often the hypothesis test is significant for the simulated data). Currently available formula-based software packages for power analysis often do not include GLMMs or are limited to very simple designs (Murayama et al., 2022; Westfall, Kenny, & Judd, 2014), making it necessary to build data simulations tailored specifically to the study design. A number of tutorials have been published describing how to perform such simulation-based power analysis for multilevel models (Arend & Schäfer, 2019; Brysbaert & Stevens, 2018; DeBruine & Barr, 2021; Kumle, Võ, & Draschkow, 2021; Lafit et al., 2021; Zimmer, Henninger, & Debelak, 2022). However, most of these tutorials focus on linear mixed models (LMMs) and the most common designs (but see Kumle et al., 2021 for a tutorial that also covers more advanced settings). This narrow focus provides limited guidance for researchers faced with more complex study designs, especially when little prior knowledge about plausible effect sizes is available (see the discussion in Kumle et al., 2021). The necessary presumptions for simulation-based power analysis with GLMMs include assumptions about the distributional form of the outcome variable, the random effects, and the correlation structure. The distributional assumption specifies the distributional family for the outcome variable (when conditioned on all fixed and random effects). Assumptions about the random effects include the assumption of normality (i.e., that the random effects follow a normal distribution) and the covariance structure among the random effects (i.e., if and how they are correlated). Interpreting these presumptions entails understanding the underlying presumptions of the model and ensuring they align with the characteristics of the data being analyzed. Existing tutorials often rely on heuristics for specifying variance components (e.g., the standard deviation of random intercepts) or assume that results from meta-analyses or data from pilot studies are available to determine plausible values for all model parameters. However, in practice, knowledge about those parameters from prior studies is often limited, which makes specifying assumptions a practical challenge Kumle et al. (2021).

Based on the need for well-powered experimental research using GLMMs and the lack of tools to conduct corresponding power analyses, in this tutorial paper, we present a case study that serves as a practical demonstration of how to perform a simulation-based a priori power analysis with GLMMs. Thereby, we aim to equip researchers with the tools needed to simulate data and determine appropriate sample sizes for their own research.

The present case study

In this section, we outline the steps for performing data simulation and a priori power analysis for GLMMs using a case study based on a specific experimental study design from the area of human-AI (artificial intelligence) interaction research. All code in this manuscript and simulation results are available in the project's repository on the Open Science Framework (https://osf.io/dhwf4/).

Experimental study design

In the present case study, we simulate data for an experiment where the diagnostic performance of users of an AI-enabled diagnostic decision support system will be evaluated. The goal is to understand how AI advice influences medical decision-making. Participants, radiologists (task experts) and students/interns (non-task experts), review head computer tomography (CT) scans to assess the presence of a bleeding. To support their decision-making, an AI model provides initial diagnostic advice, which can be used as guidance by the participants. This AI advice can be either correct (80% of cases) or incorrect (20%). In the control condition, no AI advice is presented, meaning that the participants have to read the CT scan without any support. After reviewing the CT scan, participants deliver a medical diagnosis (bleeding or no bleeding), which may be either accurate or inaccurate. This experimental design introduces some missing values by design since the advice is neither correct nor incorrect when no advice is present, which must be taken into account when simulating and analyzing the data. With this experiment, we want to determine if (a) experts are better than non-experts in reading head CT scans and if (b) correct AI advice leads to better diagnostic accuracy than incorrect AI advice. In this example, recruiting task experts (i.e., radiologists) is more challenging due to their limited availability, while non-experts (i.e., students/interns) are more readily accessible. The goal of the present simulation-based power analysis is to determine how many task experts and non-experts must be recruited to achieve sufficient statistical power in the planned experiment.

The lme4 package in R

In our case study, we use the lme4 R package (Bates, Mächler, Bolker, & Walker, 2015), which is a state-of-the-art tool for fitting frequentist GLMMs.[1] The lme4 package includes a function called `simulate` that allows researchers to simulate the dependent variable based on the same model formula used for model fitting, enabling simulation-based power analyses and other related analyses.

However, the model parameterization used by the lme4 package is quite technical, making it difficult for applied researchers to determine whether their specified population model (i.e., the theoretical model that describes the underlying data generation process for a specific population of interest) implies plausible associations in their simulated data. Therefore, in this tutorial, we simulate data for GLMMs from first principles (i.e., creating synthetic data step by step instead of using black box functions) to assist applied researchers in better understanding all model assumptions and then use lme4 to analyze the simulated datasets.[2]

Our specific GLMM

In a GLMM, the expected value of the dependent variable Y conditioned on the vector of predictor variables $\mathbf{X}$ and random effects $\mathbf{U}$, transformed by a link function $g()$ is modeled as a linear combination η of the predictor variables $\mathbf{X}$, the random effects $\mathbf{U}$ and the model parameters β (Fahrmeir et al., 2021):

$$g(E(Y|\mathbf{X} = \mathbf{x}, \mathbf{U} = \mathbf{u})) = \eta$$

[1] For Bayesian GLMMs, the brms R package is currently the most prominent option (Bürkner, 2017).

[2] A less flexible alternative would be to use the simr package (Green & MacLeod, 2016), which can be used to both simulate data and perform power analysis for models supported by the lme4 package.

Equivalently, the conditional expected value is modeled as the linear combination η, transformed by the inverse link function $g^{-1}()$:

$$E(Y|\mathbf{X} = \mathbf{x}, \mathbf{U} = \mathbf{u})) = g^{-1}(\eta)$$

If the dependent variable (i.e., diagnostic decision) Y is a binary variable with values 0 (i.e., inaccurate), or 1 (i.e., accurate), the conditional expected value is equivalent to the probability:

$$P_{si} := P(Y = 1|\mathbf{X} = \mathbf{x}, \mathbf{U} = \mathbf{u})$$

In our case study, P_{si} is the conditional probability that a subject s gives the correct response to item (i.e., CT scan) i.

In such a setting, we model this probability as

$$P_{si} = inverse\_logit(\eta_{si})$$

with the inverse-logit link $g^{-1}(\eta_{si}) = inverse\_logit(\eta_{si}) = \frac{exp(\eta_{si})}{1+exp(\eta_{si})}$ or equivalently

$$logit(P_{si}) = \eta_{si}$$

with the logit link $g(P_{si}) = logit(P_{si}) = ln(\frac{P_{si}}{1-P_{si}})$.

In our case study, the probability of making an accurate diagnostic decision is assumed to depend on the predictors:

- $advice\_present_{si}$: whether subject s was presented with AI advice (1) or not (0) when asked to assess item i
- $advice\_correct_{si}$: whether this advice was correct (1) or not (0)
- $expert_s$: whether subject s was a task expert (1) or not (0)

and the random effects:

- u_{0s}: the deviation of subject s from the average ability to solve an item (i.e., CT scan) with average difficulty; assumed to be distributed as $u_{0s} \sim N(0, \sigma_S^2)$

- u_{0i}: the deviation of item (i.e., CT scan) i from the average difficulty to be solved by a person with average ability; assumed to be distributed as $u_{0i} \sim N(0, \sigma_I^2)$

In total, we assume the model

$$\begin{aligned} logit[P_{si}] = \ & (\beta_0 + u_{0s} + u_{0i}) + \\ & \beta_a \cdot advice\_present_{si} + \beta_c \cdot advice\_correct_{si} + \beta_e \cdot expert_s + \\ & \beta_{ea} \cdot expert_s \cdot advice\_present_{si} + \beta_{ec} \cdot expert_s \cdot advice\_correct_{si} \end{aligned}$$

or equivalently

$$\begin{aligned} P_{si} = inverse\_logit[& (\beta_0 + u_{0s} + u_{0i}) + \\ & \beta_a \cdot advice\_present_{si} + \beta_c \cdot advice\_correct_{si} + \beta_e \cdot expert_s + \\ & \beta_{ea} \cdot expert_s \cdot advice\_present_{si} + \beta_{ec} \cdot expert_s \cdot advice\_correct_{si}] \end{aligned}$$

with model parameters β_0, β_e, β_a, β_c, β_{ea}, β_{ec}, σ_S, and σ_I.

In the GLMM literature, this would be called a binomial GLMM with two random intercepts (for subjects and items), two level-1 predictors ($advice\_present$, $advice\_correct$), one level-2 predictor ($expert$) and two cross-level interactions ($expert \cdot advice\_present$, $expert \cdot advice\_correct$). To limit complexity, we do not consider random slopes, additional predictors or higher-level interactions.

Data simulation

The following R function simulates a full dataset structured according to the design of our case study. The faux package (DeBruine, 2023) contains useful functions when simulating factorial designs, including random effects.

```r
simulate <- function(n_subjects = 100, n_items = 50,
  b_0 = 0.847, b_e = 1.350, b_a = -1.253, b_c = 2.603,
  b_ea = 0.790, b_ec = -1.393,
  sd_u0s = 0.5, sd_u0i = 0.5, ...){
```

```r
  require(dplyr)
  require(faux)
  # simulate design
  dat <- add_random(subject = n_subjects, item = n_items) %>%
    add_between("subject", expert = c(1, 0), .prob = c(0.25, 0.75)) %>%
    mutate(advice_present = rbinom(n(), 1, prob = 2/3)) %>%
    mutate(advice_correct = if_else(advice_present == 1L,
                                    rbinom(n(), 1L, prob = 0.8), 0L)) %>%
    # add random effects
    add_ranef("subject", u0s = sd_u0s) %>%
    add_ranef("item", u0i = sd_u0i) %>%
    # compute dependent variable
    mutate(linpred = b_0 + u0i + u0s +
        b_e * expert + b_a * advice_present + b_c * advice_correct +
        b_ea * expert * advice_present + b_ec * expert * advice_correct) %>%
    mutate(y_prob = plogis(linpred)) %>%
    mutate(y_bin = rbinom(n = n(), size = 1, prob = y_prob))
  dat
}
```

In the first six lines of the function definition, we set some default parameter values (which we will explain in a later section) and load the packages we use to manipulate and simulate data. In our case study, each subject (`n_subjects` in total) is assumed to respond to each item (i.e., CT scan; `n_items` in total). Thus, the `add_random` command creates a fully-crossed `data.frame` with `n_subjects` × `n_items` rows. We add a between-subject effect with the `add_between` command, simulating that about 25% of subjects are experts. The next two lines simulate that in $\frac{2}{3}$ of trials, subjects will be presented with AI advice, and if advice is presented, the advice will be correct in about 80% of cases (the variable advice_correct is always 0 when no advice is presented). Next, we simulate one random effect for each subject (u0s) and for each item (u0i). As assumed by standard GLMMs, the add_ranef function draws the random effects from a normal distribution with a mean 0 and a standard deviation specified by the user. With all design variables done, we are ready to simulate our model equation outlined in the last section. The linear predictor variable linpred (η in the GLMM model equations) combines the predictor variables, random effects, and model parameters as assumed by our model. We then transform the linear predictor with the inverse-link function to compute y_prob, the probability that the subject correctly solved the item (in R, the inverse-logit link is computed with plogis and the logit link with qlogis). In the final step, we simulate the binary dependent variable y_bin (i.e., whether the subject makes an accurate diagnostic decision for the CT scan) by – for each trial – drawing from a Bernoulli distribution with success probability y_prob.

Model fitting

In this section, we show how to fit a GLMM with lme4, interpret the model, and test hypotheses derived from a research question. We simulate data according to our model, in which 100 subjects respond to 50 items (we use set.seed to make the simulation reproducible). However, for the sake of the exercise, we can imagine that this would be real data resulting from our future experiment and think about how we would analyze this data.

```r
library(tidyverse)
set.seed(1)
dat <- simulate(n_subjects = 100, n_items = 50)
```

The lme4 package uses a special syntax for model specification. Our specific GLMM is represented by the formula:

```r
library(lme4)
f <- y_bin ~ 1 + expert + advice_present + advice_correct +
  expert:advice_present + expert:advice_correct +
  (1|subject) + (1|item)
```

The first two lines look similar to any linear model in R (general intercept indicated by `1`; main effects indicated by variable names in the dataset; interactions indicated by `variable1:variable2`). The third line specifies a random intercept for each subject `(1|subject)` and for each item `(1|item)`. The complete set of rules for the syntax is outlined in Bates et al. (2015) and in the documentation of the lme4 package.

In lme4, a GLMM is fitted with the `glmer` function. By setting `family = "binomial"`, we request a binomial GLMM appropriate for our binary dependent variable `y_bin` (the binomial GLMM uses the canonical logit link by default), which is defined as an accurate (1) vs. inaccurate (0) diagnosis.

```r
fit <- glmer(f, data = dat, family = "binomial")
```

Model interpretation

We can inspect the estimates for all model parameters with the `summary` command:

```r
summary(fit)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula:
## y_bin ~ 1 + expert + advice_present + advice_correct + expert:advice_present +
##     expert:advice_correct + (1 | subject) + (1 | item)
##    Data: dat
##
##      AIC      BIC   logLik deviance df.resid
##   4149.4   4201.6  -2066.7   4133.4     4992
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -5.7669  0.2125  0.3046  0.4317  2.1056
##
## Random effects:
##  Groups  Name        Variance Std.Dev.
##  subject (Intercept) 0.3148   0.5611
##  item    (Intercept) 0.1624   0.4029
## Number of obs: 5000, groups:  subject, 100; item, 50
##
## Fixed effects:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)             1.0339     0.1103   9.374  < 2e-16 ***
## expert                  1.1849     0.2096   5.654 1.57e-08 ***
## advice_present         -1.3436     0.1206 -11.143  < 2e-16 ***
## advice_correct          2.6154     0.1273  20.540  < 2e-16 ***
## expert:advice_present   1.0589     0.2940   3.601 0.000317 ***
## expert:advice_correct  -1.8104     0.2915  -6.211 5.27e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
```

```
##             (Intr) expert advc_p advc_c exprt:dvc_p
## expert      -0.377
## advic_prsnt -0.349  0.176
## advic_crrct  0.023  0.001 -0.668
## exprt:dvc_p  0.143 -0.448 -0.412  0.276
## exprt:dvc_c -0.008  0.004  0.292 -0.435 -0.686
```

In the output, the `Estimate` column in the `Fixed effects` table contains the estimates for the β parameters, while the `Std.Dev.` column in the `Random effects` table contains the estimates for σ_S and σ_I.

Unfortunately, the model parameters in a binomial GLMM are hard to interpret because 1) the β parameters are connected to the modeled probability via the non-linear inverse-logit link, and 2) we also have to consider the random effects. The most simple interpretation works by imagining a subject with average ability ($u_{0s} = 0$) responding to an item (i.e., CT scan) with average difficulty ($u_{0i} = 0$). Then the model implied probability that such a person solves such an item accurately is given by:

$$
\begin{aligned}
&P(Y = 1 | \mathbf{X} = \mathbf{x}, \mathbf{U} = \mathbf{0}) = \\
&\qquad = inverse\_logit[\beta_0 + \beta_a \cdot advice\_present_{si} + \beta_c \cdot advice\_correct_{si} + \beta_e \cdot expert_s + \\
&\qquad\qquad \beta_{ea} \cdot expert_s \cdot advice\_present_{si} + \beta_{ec} \cdot expert_s \cdot advice\_correct_{si}]
\end{aligned}
$$

In fact, we would only need the full equation if the subject is an expert and correct advice is presented. In all other experimental conditions, some terms drop from the equation because they are multiplied by 0. The other extreme case would be the probability that a non-expert with average ability solves an item with average difficulty when no advice is presented:

$$
\begin{aligned}
&P(Y = 1 | expert = 0, advice\_present = 0, advice\_correct = 0, u_{0s} = 0, u_{0i} = 0) = \\
&\qquad = inverse\_logit[\beta_0]
\end{aligned}
$$

Due to this complicated relationship, we argue not to focus too much on interpreting single model parameters when working with GLMMs. Instead, it can be more intuitive to consider model predictions and the model-implied distribution of the dependent variable for each experimental condition across all subjects and items.

With the marginaleffects package (Arel-Bundock, 2023), we can easily compute predictions for all observations in the dataset based on the fitted GLMM (including all fixed **and** random effects), and plot the average probability with confidence intervals for each experimental condition in Figure 1:

```r
library(marginaleffects)
plot_predictions(fit, by = c("advice_present", "advice_correct", "expert"),
  type = "response") + ylim(c(0.3, 1))
```

Hypothesis testing

However, we need to think about the model parameters again when we want to test hypotheses that we have theoretically derived from some research question. Because the inverse-logit link is still a continuously increasing function, positive parameter values always correspond to increases in probability and vice versa.

The `Fixed effects` table in the lme4 summary output also includes p-values for hypothesis tests with null hypotheses of the style $H_0 : \beta = 0$. However, for many research questions of interest, we are not interested in these two-sided tests that refer to only a single parameter.

For our case study, imagine the following combined hypothesis: *We expect that for both experts and non-experts, correct advice leads to a higher probability of accurately diagnosing a CT scan compared to no advice presented, AND, we expect that for both experts and non-experts, incorrect advice leads to a lower probability of accurately diagnosing a CT scan*

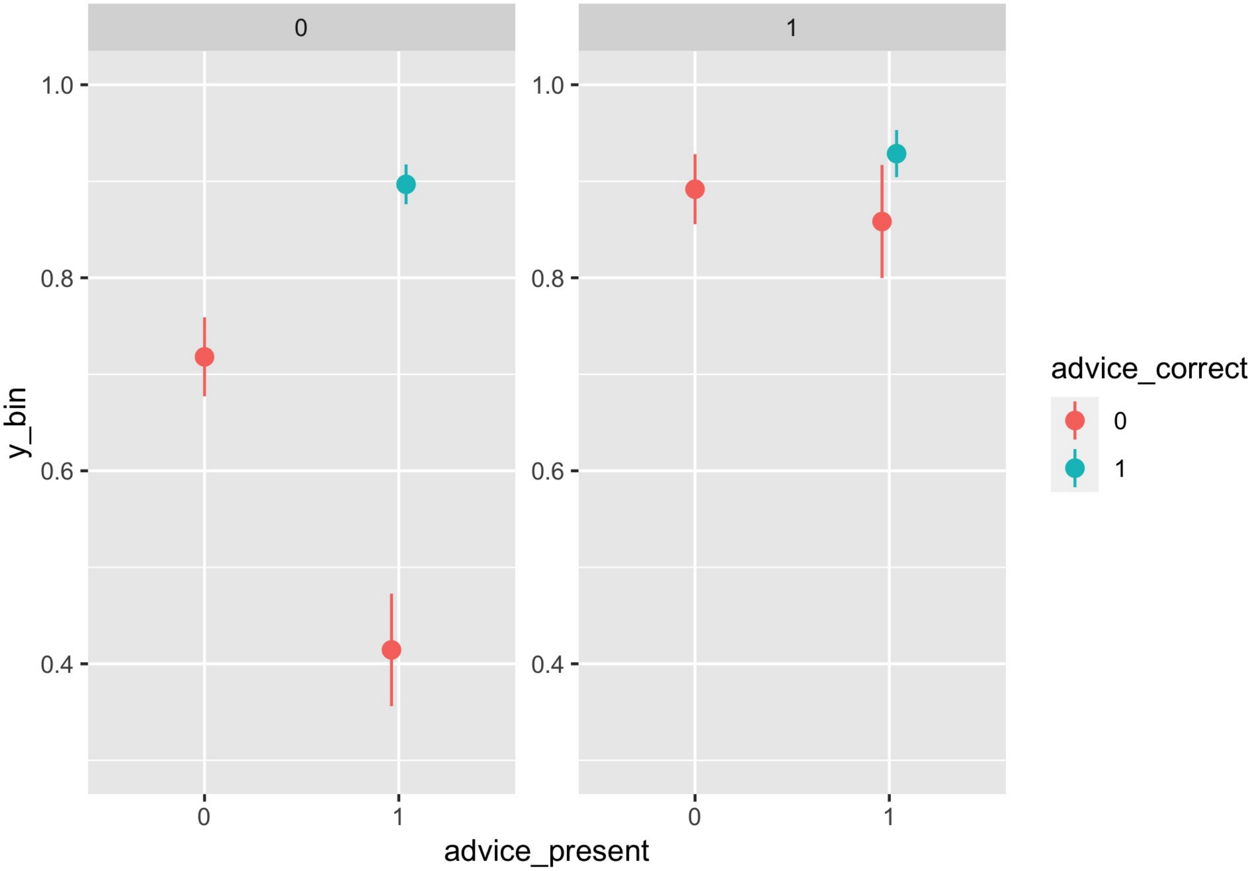

Figure 1. Marginal distributions including means and 95% confidence intervals for all experimental conditions computed with the marginaleffects package.

compared to no advice presented.

This combined hypothesis leads to the following four separate null hypotheses to be tested:

$$H_{01} : \beta_a + \beta_c + \beta_{ea} + \beta_{ec} \leq 0$$

$$H_{02} : \beta_a + \beta_c \leq 0$$

$$H_{03} : \beta_a + \beta_{ea} \geq 0$$

$$H_{04} : \beta_a \geq 0$$

We arrive at these inequalities based on the following logic, exemplified here only for H_{01}: The first null hypothesis states that *an expert responding to an item while presented* *with correct advice has a lower or equal probability of solving the item compared to the same expert facing the same item without any advice.* This implies the following inequality for each subject s and item i

$$inverse\_logit[(\beta_0 + u_{0s} + u_{0i}) + \beta_e + \beta_a + \beta_c + \beta_{ea} + \beta_{ec}] \leq inverse\_logit[(\beta_0 + u_{0s} + u_{0i}) + \beta_e]$$

which simplifies to $\beta_a + \beta_c + \beta_{ea} + \beta_{ec} \leq 0$.

We can specify and test hypotheses like these with the multcomp package (Hothorn, Bretz, & Westfall, 2008) as follows:

```r
library(multcomp)
null_hypotheses <- c(
  "advice_present + advice_correct + expert:advice_present +
   expert:advice_correct <= 0",
  "advice_present + advice_correct <= 0",
  "-1 * (advice_present + expert:advice_present) <= 0",
  "-1 * (advice_present) <= 0")
glht <- glht(fit, linfct = null_hypotheses)
summary(glht, test = univariate())$test$pvalues
```

```
## advice_present + advice_correct + expert:advice_present + expert:advice_correct
##                                                                     0.006407391
##                                                 advice_present + advice_correct
##                                                                     0.000000000
##                                   -1 * (advice_present + expert:advice_present)
##                                                                     0.143963670
##                                                           -1 * (advice_present)
##                                                                     0.000000000
```

Because all hypotheses tested simultaneously with the `glht` function must have the same direction, we flip the sign of inequalities three and four by multiplying them with -1. The multcomp package automatically adjusts p-values when multiple hypotheses are tested simultaneously (Hothorn et al., 2008). However, the combined null hypothesis in our exemplary research question should only be rejected if **all** individual null hypotheses are rejected [i.e., intersection-union setting; Dmitrienko and D'Agostino (2013)]. In such cases, the error probabilities do not accumulate, and we would waste power when correcting for multiple tests. Thus, we request unadjusted p-values by setting `test = univariate()` in the `summary` command. With a standard significance level of $\alpha = 0.05$, we would not reject all four null hypotheses (the p-value for hypothesis H_{03} is not significant) and therefore also not reject the combined null hypothesis for this simulated dataset. Note that this decision would be wrong because we have simulated the data such that the combined alternative hypothesis is actually true in the population.

Specification of plausible parameter values

When introducing our simulation function and simulating data for the above example, we have used theoretically plausible values as defaults for all model parameters (β_0, β_e, β_a, β_c, β_{ea}, β_{ec}, σ_S, and σ_I), but have not talked about where these numbers came from.

Ideally, one would rely on meta-analytic results or conclusive data from pilot studies. However, these are sometimes not readily available. All parameter values in our present case study have been determined based on results from related prior work. Additionally, we had repeated discussions with our affiliated domain experts in radiology to check our assumptions.

We now outline a few strategies on how to determine plausible parameter values. We have already seen in our discussion of model interpretation how we can derive the model implied probability for each experimental condition, that a subject with average ability solves an item with average difficulty. We can revert this perspective by choosing plausible

Table 1

Assumed probabilities that an average subject solves an average item in each experimental condition.

Experimental condition	$P(Y = 1 \vert \mathbf{X} = \mathbf{x}, \mathbf{U} = \mathbf{0})$	Implied equation
no advice, no expert	0.70	$logit(0.70) = \beta_0$
no advice, expert	0.90	$logit(0.90) = \beta_0 + \beta_e$
false advice, no expert	0.40	$logit(0.40) = \beta_0 + \beta_a$
false advice, expert	0.85	$logit(0.85) = \beta_0 + \beta_e + \beta_a + \beta_{ea}$
correct advice, no expert	0.90	$logit(0.90) = \beta_0 + \beta_a + \beta_c$
correct advice, expert	0.95	$logit(0.95) = \beta_0 + \beta_e + \beta_a + \beta_c + \beta_{ea} + \beta_{ec}$

Note. Implied equations are derived based on the model equations and setting all random intercept terms to 0.

probability values and deriving the parameter values implied by these probabilities (for an average subject and an average item).

Table 1 shows our set of assumptions concerning the probability that an average subject solves an average item for each experimental condition, as well as the corresponding equations implied by the model. The table can be used to compute the implied values for the β parameters, starting with the first equation and reinserting the computed β values in all following equations:

```r
b_0 <- qlogis(0.7)
b_e <- qlogis(0.9) - b_0
b_a <- qlogis(0.4) - b_0
b_ea <- qlogis(0.85) - b_0 - b_e - b_a
b_c <- qlogis(0.9) - b_0 - b_a
b_ec <- qlogis(0.95) - b_0 - b_e - b_a - b_c - b_ea
```

```r
c(b_0 = b_0, b_e = b_e, b_a = b_a, b_c = b_c, b_ea = b_ea, b_ec = b_ec)
```

```
##        b_0        b_e        b_a        b_c       b_ea       b_ec
##  0.8472979  1.3499267 -1.2527630  2.6026897  0.7901394 -1.3928518
```

It is always possible to double-check these computations by transforming the parameter values back to probabilities, e.g.

$$P(Y = 1|expert = 1, advice\_present = 1, advice\_correct = 1, u_{0s} = 0, u_{0i} = 0) =$$

$$= inverse\_logit[\beta_0 + \beta_e + \beta_a + \beta_c + \beta_{ea} + \beta_{ec}]$$

```r
plogis(b_0 + b_e + b_a + b_c + b_ea + b_ec)
```

```
## [1] 0.95
```

Although the derivations above are straightforward, it is important not to misinterpret their implications: In binomial GLMMs, the average probability to solve an item (averaged across persons of varying ability and items of varying difficulty) is **not** equal to the probability that a person with average ability solves an item with average difficulty. The first perspective implies a so-called marginal interpretation, while the second one implies a conditional interpretation. For example, we determined the β parameters in a way that corresponds to a desired conditional probability of 0.95, that an expert with average ability solves an item with average difficulty when presented with correct advice. However, even if the model were true, we would not observe this probability value if we estimated the marginal probability in a group of experts responding to items presented with correct advice from a big sample of subjects drawn from their natural distribution of ability and items drawn from their natural distribution of difficulty.

The inequality of conditional and marginal effects in GLMMs (Fahrmeir et al., 2021) makes their interpretation more difficult. One must be careful when specifying parameter values based on previous studies or pilot data that use the marginal interpretation (e.g., a pilot study providing an estimate of how often neurologists make an accurate diagnosis based on brain scans). However, this does not mean that we cannot use the marginal interpretation (average probability across persons and items) to inform plausible parameter values: When parameter values have been selected, we can compute the implied marginal distributions and compare this information to our domain knowledge. Then, we can iteratively adjust the parameter values until we are satisfied with the implied distributions.

Earlier, we have already encountered one way to visualize the implied marginal distributions: We can fit our model to a simulated dataset and use the convenience functions from the marginaleffects package to compute averaged predictions that correspond to our quantities of interest. However, the model predictions will only be close to the true distribution if the simulated dataset is very large, but then the model fitting consumes a lot of time and memory. A more sophisticated strategy is to simulate a large dataset and directly compute the averages, contrasts and distributions we are interested in.

```r
library(tidyverse)
library(ggdist)
dat <- simulate(n_subjects = 2000, n_items = 2000, sd_u0s = 0.5, sd_u0i = 0.5)
dat %>%
  mutate(condition = fct_cross(
    factor(expert), factor(advice_present), factor(advice_correct))) %>%
  mutate(condition = fct_recode(condition,
    "no expert, no advice" = "0:0:0", "expert, no advice" = "1:0:0",
    "no expert, wrong advice" = "0:1:0", "expert, wrong advice" = "1:1:0",
    "no expert, correct advice" = "0:1:1", "expert, correct advice" = "1:1:1")) %>%
  ggplot(aes(x = y_prob, y = condition)) +
  stat_histinterval(point_interval = "mean_qi", slab_color = "gray45") +
  scale_x_continuous(breaks = seq(0, 1, 0.1), limits = c(0, 1))
```

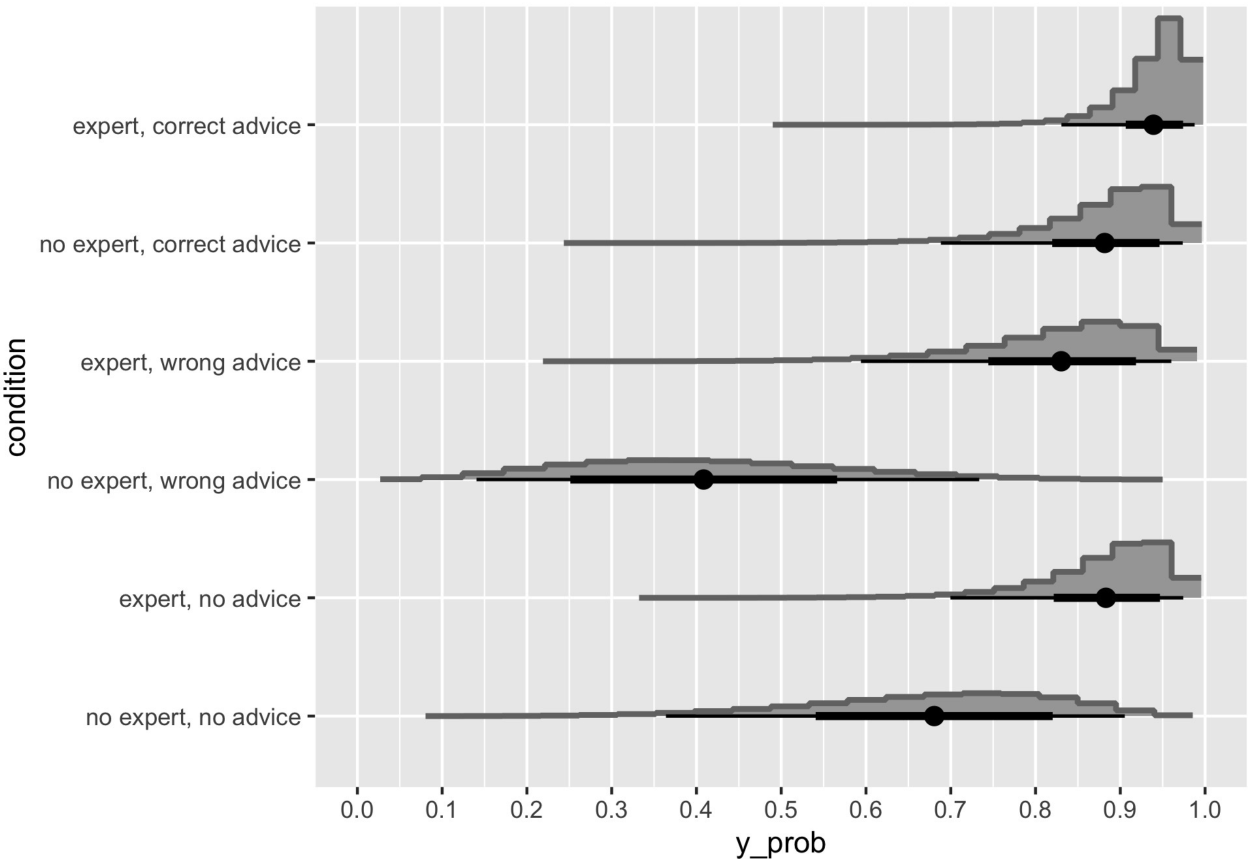

Figure 2. Marginal distributions including means, 66% and 95% confidence intervals for all experimental conditions.

Figure 2 shows the model implied marginal distributions, including the mean, 66% and 95% intervals. We can see that, indeed, the average probabilities (black dots) slightly differ from the probabilities of average subjects and items considered in the previous section. This difference increases with the variability of the random effects.

Up to this point, we have not talked about plausible values for the standard deviations of the subject and item random intercepts (σ_S and σ_I). Plots like the one above are a useful tool to decide whether the specified standard deviations are reasonable by comparing the ranges and overlap between conditions to domain knowledge.

In the next plot, we have set the item standard deviation to almost zero ($\sigma_I = 0.01$).

This gives us a better way to see the variability between persons.

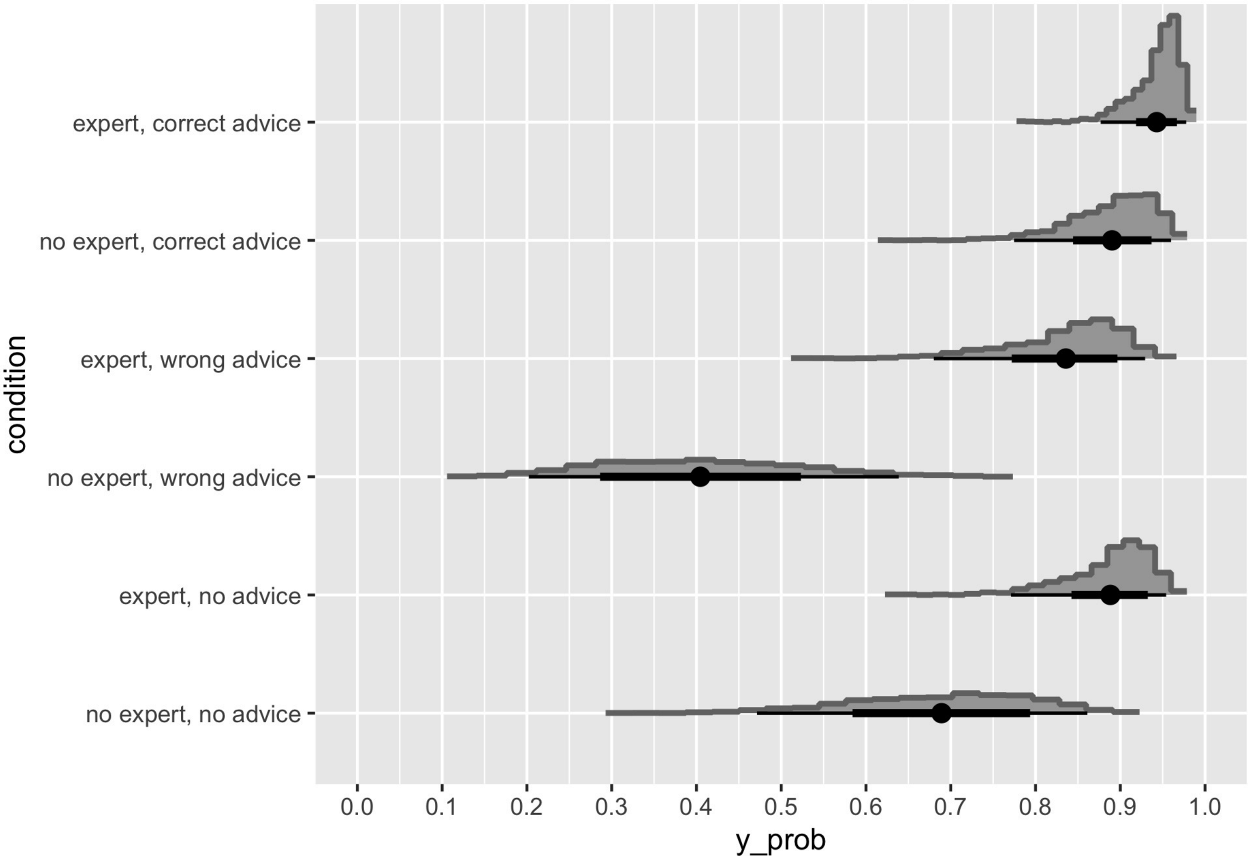

Figure 3. Marginal distributions including means, 66% and 95% confidence intervals for all experimental conditions while setting the standard deviation of item random intercepts to 0.01.

As an example, Figure 3 reveals a number of implicit assumptions about the comparison between experts and non-experts: With wrong advice, virtually all experts have a higher probability of making a correct diagnosis compared to non-experts when considering only items with average difficulty. In contrast, there is considerable overlap in probability between experts and non-experts with no advice and even higher overlap with correct advice. Patterns like these should be considered carefully and discussed with the domain experts. Parameter values (β parameters, and σ_S) should be adjusted if the implications do not seem reasonable.

We could also have a closer look at variability between items by setting the subject standard deviation to almost zero ($\sigma_S = 0.01$).

The final plot demonstrates that these plots are also useful for spotting standard deviations that are specified too high. For Figure 4, we have set $\sigma_S = 3$ and $\sigma_I = 3$. This implies that in each experimental condition, the probabilities that a subject solves an item are usually close to either 0 or 1, which is not a plausible assumption. However, these high standard deviations do not account for the inherent variability and complexity of human performance. For example, we would expect that a participant with low ability compared to other task experts to solve a difficult item with a probability substantially larger than zero even when presented with wrong advice.

Results

With all these considerations addressed, we are finally ready to perform a power analysis. Wrapping the simulate function already constructed earlier, the helper function sim_and_analyse performs all previous steps (simulate a dataset, fit a GLMM, compute p-values) in a single command.

```
sim_and_analyse <- function(
  formula_chr = "y_bin ~ 1 + expert + advice_present + advice_correct +
    expert:advice_present + expert:advice_correct + (1|subject) + (1|item)",
  null_hypotheses = c("advice_present + advice_correct +
    expert:advice_present + expert:advice_correct <= 0",
    "advice_present + advice_correct <= 0",
    "-1 * (advice_present + expert:advice_present) <= 0",
    "-1 * (advice_present) <= 0"), ...){
  require(lme4)
  require(multcomp)
```

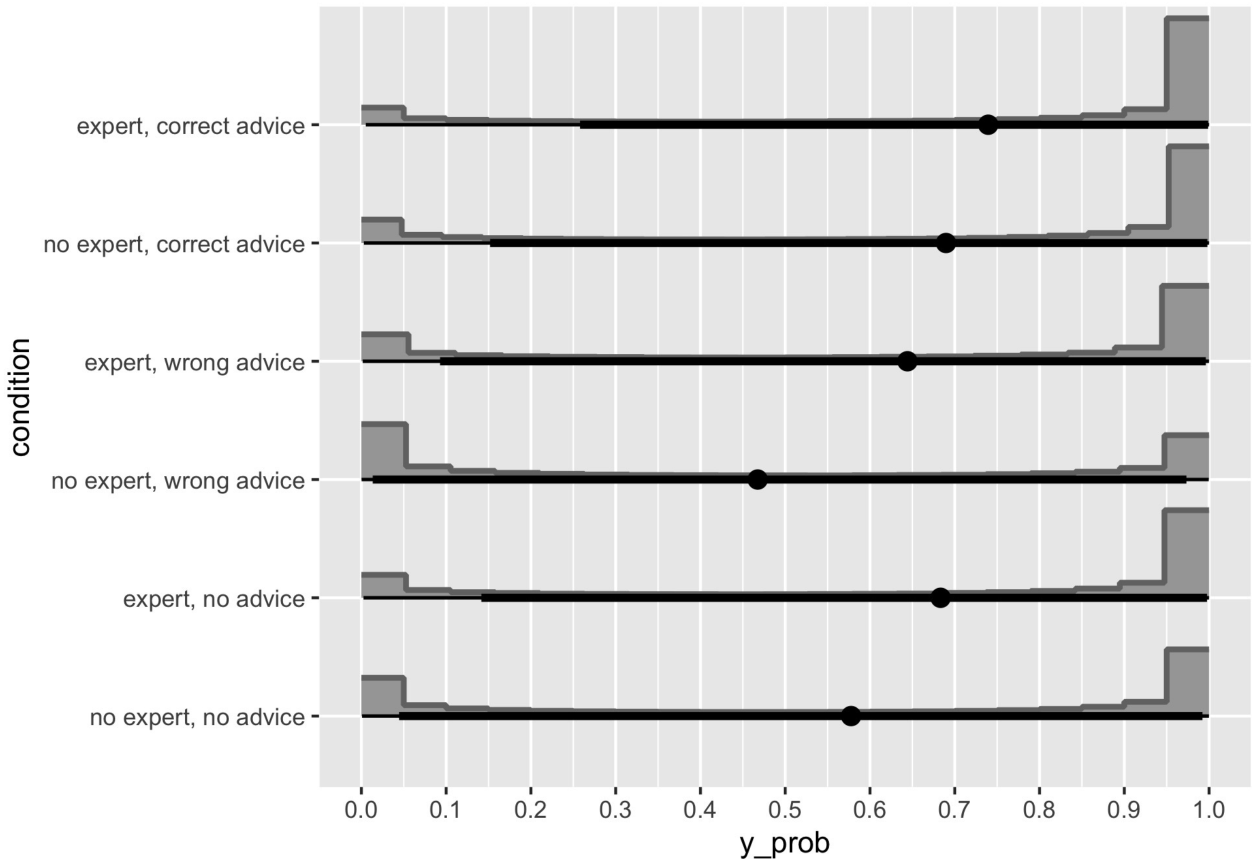

Figure 4. Marginal distributions including means, 66% and 95% confidence intervals for all experimental conditions while setting the standard deviation of subject and item random intercepts to 3.

```
  # simulate data
  dat <- simulate(...)
  # fit model
  model <- glmer(as.formula(formula_chr), data = dat, family = "binomial")
  # compute p-values
  glht <- glht(model, linfct = null_hypotheses)
  pvalues <- summary(glht, test = univariate())$test$pvalues
  setNames(pvalues, paste0("p_H0", 1:length(null_hypotheses)))
}
```

Power analysis can quickly become computationally intensive when we repeatedly simulate data and fit models for different parameter combinations or sample sizes. Thus, we use the future (Bengtsson, 2021) and furrr (Vaughan & Dancho, 2022) packages to perform computations in parallel. First, we enable parallelization and specify how many parallel cores ("workers") of our computer to use (users can find out the maximum number of cores on their computer with the command `parallel::detectCores()`), and set a seed to make the simulation reproducible.

```r
library(future)
plan("multisession", workers = 6)
set.seed(2)
```

The next code chunk specifies a simulation grid with different settings for both the number of subjects (`n_subjects`) and the number of items (`n_items`), each combination being repeated `rep` times. We chose 300 repetitions for the data simulation at hand as it strikes a balance between achieving a robust statistical estimate and remaining computationally feasible. With the current settings, this simulation takes about one hour on a MacBook Pro from 2020 with M1 chip and 16 GB working memory. If you want to quickly experiment with the code yourself, a setting with `workers = 4` and `rep = 5` should finish in less than 5 minutes, even on smaller machines.

```r
library(furrr)
sim_design <- crossing(
  rep = 1:300,
  n_subjects = c(100, 150, 200, 250),
  n_items = c(10, 30, 50, 70)
) %>%
  mutate(pvalues = future_pmap(., sim_and_analyse,
    .options = furrr_options(seed = TRUE))) %>%
  unnest_wider(pvalues)
```

The result of the computation is a data frame that contains the p-values of all tested hypotheses for each simulated dataset. In some iterations (predominantly in conditions with small sample sizes), model estimation did not converge with the lme4 package. When the model fails to converge, it means that the statistical model being fitted to the data failed to reach a stable or valid solution during the estimation process. We do not remove these results because non-convergence can also happen when analyzing the real data we plan to collect, thus, we want to factor in this possibility to keep our simulation more realistic.

For our exemplary combined hypothesis, power is defined as the (long-run) percentage of simulations in which all four p-values of our component hypotheses are significant at the $\alpha = 0.05$ level. Based on our simulation outcomes, we compute a power estimate for each combination of `n_subjects` × `n_items` (including 95% confidence intervals) and visualize the results with the following code.[3]

```
library(binom)
alpha <- 0.05
power <- sim_design %>%
  group_by(n_subjects, n_items) %>%
  summarise(power = mean(p_H01 < alpha & p_H02 < alpha &
                           p_H03 < alpha & p_H04 < alpha),
    n_sig = sum(p_H01 < alpha & p_H02 < alpha &
                  p_H03 < alpha & p_H04 < alpha),
    n = n(),
    ci.lwr = binom.confint(n_sig, n, method = "wilson")$lower,
```

[3] This code was inspired by the “Mixed Design Simulation” vignette of the faux package at https://debruine.github.io/faux/articles/sim_mixed.html.

```
    ci.upr = binom.confint(n_sig, n, method = "wilson")$upper,
    .groups = "drop")
power %>%
  mutate(across(c(n_subjects, n_items), factor)) %>%
  ggplot(aes(n_subjects, n_items, fill = power)) +
  geom_tile() +
  geom_text(aes(label = sprintf("%.2f \n [%.2f; %.2f]",
                               power, ci.lwr, ci.upr)),
    color = "white", size = 4) +
  scale_fill_viridis_c(limits = c(0, 1)) +
  xlab("number of subjects") + ylab("number of items")
```

As should be the case, power estimates in Figure 5 increase with both the number of subjects and the number of items. The confidence intervals indicate how precisely power was estimated by our simulation. Higher precision (which would be reflected in narrower confidence intervals) could be obtained by increasing the number of repetitions (`rep`) in the simulation. In practice, data simulations are often run multiple times with adjusted combinations of sample sizes. When running for the first time, it might be revealed that power is way too low (or much higher than required) for some combinations of `n_subjects` and `n_items`. When narrowing down the best combination that achieves sufficient power while at the same time striking a good balance of how many subjects and items are practically feasible, later rounds of data simulation will typically include a smaller grid of sample sizes combined with a higher number of repetitions. This will assure high precision for the final power estimates, which are then used for the sample size justification of the future study.

Much has been written on the optimal amount of power to target in empirical research. The most prominent heuristic is to target a power of 0.8 (when combined with a type I error

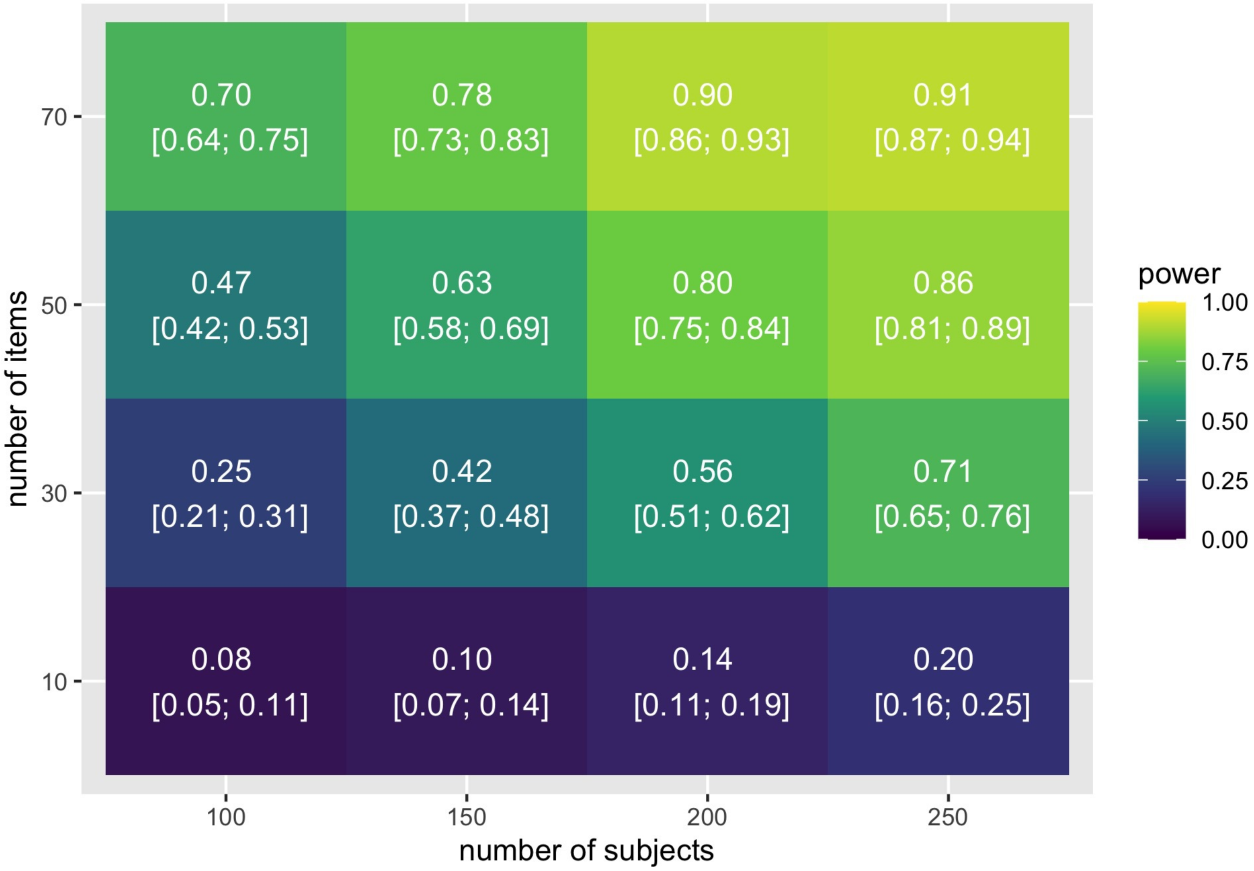

Figure 5. Simulation-based power estimates including 95% confidence interval of the case study for different numbers of subjects and items, based on a significance level of 0.05.

rate of $\alpha = 0.05$), but depending on the research goals of the study, there are often good reasons to move away from this standard depending on the research goals and resource constraints (Lakens, 2022b; Lakens et al., 2018). When target power has been specified, the number of subjects and the number of items in our study design can be traded against each other based on practical considerations. For the sake of the example, let the targeted power be indeed about 0.8, using an α of 0.05 to detect an effect of the expected size implied by our data simulation. This could be achieved by collecting data from 200 subjects (about 25% of which will be experts), each completing the same 50 items (with advice present in about 67% of cases, which is correct in about 80% of cases with present advice). If collecting data from 200 subjects is not feasible, an alternative would be to recruit 150 subjects but increase the length of the experiment to over 70 items. However, 70 items might take too long to complete for the radiologists participating in the study, who have a busy schedule. The simulation suggests that it might also be possible to plan a shorter experiment with only 30 items if it is feasible to recruit an even higher number of subjects (> 250, to be determined by additional rounds of power analysis). Design parameters that also affect power, and which could be investigated in the simulation to find a more optimal trade-off, are the ratio of experts, the frequency of whether advice is presented and whether it is correct.

Discussion

Experimental research requires careful planning and consideration of statistical power to ensure robust and meaningful results. While heuristics and user-friendly software can be useful for simple designs and models, they often fall short when more complex and customized simulations with GLMMs are required. The present tutorial presents a specific case study with corresponding code of how to conduct a simulation-based power analysis for experimental designs with GLMMs.

Expected effect size vs. smallest effect size of interest: sensitivity power analysis

In our case study, we have performed simulation-based power analysis from a single set of parameter values that reflect our assumptions of an expected effect size. Instead of extracting this expected effect size from meta-analyses or pilot data, which has been the main focus of previous tutorials, we have demonstrated some strategies to determine plausible parameter values in GLMMs based on domain knowledge. Domain knowledge can be considered a vague theoretical model about the data-generating process that is less formal and can only be accessed by a back-and-forth exchange in which domain experts assess the plausibility of simulated data. When sample sizes are chosen based on the results of our simulation-based power analysis, a future study will be informative to reject the null hypothesis if an effect of our *expected size* is present. However, if the true effect is indeed smaller, the power will be lower, and the study might not be sufficiently informative. A common, more conservative strategy for sample size justification is to perform power analysis for the smallest effect size of interest (SESOI). An effect smaller than the SESOI would be considered too small to be interesting or practically meaningful, even if the effect is not actually zero (King, 2011). For strategies on the even more difficult task of specifying a plausible SESOI, as well as a thorough discussion of various topics concerning power analysis, see (Lakens, 2022a). When domain knowledge or formal theories about the research topic of interest are too vague to specify a meaningful SESOI, it is still recommended to demonstrate power for different effect sizes in what is called *sensitivity power analysis*. By simulating power for different effect sizes (in addition to the different number of subjects and items), one can make sure that power would still be sufficient to detect smaller effect sizes than our expected effect or at least get an impression of how strongly power depends on the size of the true effect. In simple study designs, it is possible to perform sensitivity power analysis based on a single standardized effect size (e.g., analyze power in a two-sample t-test for a standardized mean difference varying between 0.1 and 0.8). However, for our case study that investigates combined hypotheses in a GLMM modeling framework, the effect size is implicitly represented by the complex distribution of probabilities within and between experimental conditions. In this setting, sensitivity power analysis would require manually specifying additional sets of plausible parameter values that reflect scenarios with smaller or larger differences between groups with respect to our specific research question. Power could then be simulated for several of these scenarios (across different numbers of subjects and items, as considered earlier).

Outlook

Beyond the specifics of our concrete case study, we want to outline six developments regarding the future role of simulation-based power analysis in experimental research:

1. The growing need for simulation-based power analyses in experimental research: In order to conduct well-powered research using varying complex experimental designs

with GLMMs formula-based heuristics and user-friendly software tools for a priori power analysis are often not suitable. Therefore, simulation-based power analysis is becoming increasingly needed since it provides experimental researchers with a tailored approach to estimating required sample sizes before data collection.

2. Managing data simulations more easily with discrete predictor variables: Simulation-based power analysis becomes more manageable when all predictor variables are discrete (like in the presented case study) and fixed by the study design. This allows researchers to focus on simulating outcome variables while avoiding the need for complex simulations of predictor values, which would introduce additional assumptions. By simplifying the simulation process, researchers can obtain reliable power estimates without compromising realistic assumptions about the data-generating process implied by the study design.

3. Teaching data simulation skills: The ability to conduct simulation-based power analysis is a valuable skill that should be taught to experimental researchers. By incorporating such training into research methods courses and workshops, researchers can gain a deeper understanding of statistical power and improve the quality of their experimental designs. Equipping researchers with the knowledge and tools to perform simulation-based power analyses enables them to make informed decisions and enhance the rigor of their studies. The need to reason about how to simulate plausible data that is in line with the research hypothesis, while not violating domain expertise on how plausible data should look, might also contribute to planning more insightful studies that can answer more precise research questions (Yarkoni, 2022).

4. Addressing the mismatch in effort perception: There is often a significant disconnect between the amount of effort required to perform simulation-based a priori power analysis and the perceived effort estimated by researchers and collaborators in experimental research. Many researchers request simulation-based power analyses from

statisticians or methodological experts without fully comprehending the complexity and time-consuming nature of these tailored simulations. It is crucial to raise awareness about the effort involved to ensure realistic expectations and effective collaboration between researchers and methodological experts.

5. Recognizing the value of simulation-based power analysis: Simulation-based power analyses are not mere technicalities; they are valuable research contributions that deserve recognition in experimental research. They offer insights into the robustness and sensitivity of experimental designs, helping researchers make informed decisions about sample sizes, effect sizes, and statistical power. Their importance can be reflected by allocating them a separate publication or incorporating them as a significant component of stage 1 preregistered reports (Chambers & Tzavella, 2022).

6. Integration with Open Science and preregistration practices: Simulation-based powers analysis aligns well with the principles of Open Science and preregistration in experimental research. When researchers have access to simulated data based on their pre-specified model, analyzing the collected dataset becomes straightforward and unambiguous. By preregistering their simulation-based power analysis, researchers enhance the transparency and accountability of their experimental procedures, contributing to the credibility and reproducibility of research.

Conclusion

In the wake of the replication crisis and myriad of underpowered experimental work, generalized linear mixed models (GLMMs) offer a flexible statistical framework to analyze experimental data with complex (e.g., dependent and hierarchical) data structures. Yet, analytic methods and software cannot be applied to conduct a priori power analyses for GLMMs necessitating data simulation-based approaches. Through this applied tutorial, we aim to provide researchers with the necessary skills and tools to perform simulation-based power analysis with GLMMs themselves. By incorporating GLMMs and a priori power analysis into their work, researchers can enhance the replicability and credibility of their experiments (Yarkoni, 2022).

References

Arel-Bundock, V. (2023). *Marginaleffects: Predictions, comparisons, slopes, marginal means, and hypothesis tests*. Retrieved from https://CRAN.R-project.org/package=marginaleffects

Arend, M. G., & Schäfer, T. (2019). Statistical power in two-level models: A tutorial based on Monte Carlo simulation. *Psychological Methods*, *24*(1), 1–19. https://doi.org/10.1037/met0000195

Bates, D., Mächler, M., Bolker, B., & Walker, S. (2015). Fitting Linear Mixed-Effects Models Using **Lme4**. *Journal of Statistical Software*, *67*(1). https://doi.org/10.18637/jss.v067.i01

Bengtsson, H. (2021). A unifying framework for parallel and distributed processing in r using futures. *The R Journal*, *13*(2), 208–227. https://doi.org/10.32614/RJ-2021-048

Brysbaert, M., & Stevens, M. (2018). Power Analysis and Effect Size in Mixed Effects Models: A Tutorial. *Journal of Cognition*, *1*(1), 9. https://doi.org/10.5334/joc.10

Bürkner, P.-C. (2017). Brms: An R Package for Bayesian Multilevel Models Using Stan. *Journal of Statistical Software*, *80*, 1–28. https://doi.org/10.18637/jss.v080.i01

Chambers, C. D., & Tzavella, L. (2022). The past, present and future of Registered Reports. *Nature Human Behaviour*, *6*(1), 29–42. https://doi.org/10.1038/s41562-021-01193-7

Champely, S., Ekstrom, C., Dalgaard, P., Gill, J., Weibelzahl, S., Anandkumar, A., … De Rosario, M. H. (2018). Package “pwr.” *R Package Version*, *1*(2).

Cockburn, A., Dragicevic, P., Besançon, L., & Gutwin, C. (2020). Threats of a replication crisis in empirical computer science. *Communications of the ACM*, *63*(8), 70–79. https://doi.org/10.1145/3360311

DeBruine, L. (2023). *Faux: Simulation for factorial designs*. Zenodo. https://doi.org/10.5281/zenodo.2669586

DeBruine, L., & Barr, D. J. (2021). Understanding Mixed-Effects Models Through Data Simulation. *Advances in Methods and Practices in Psychological Science*, *4*(1), 2515245920965119. https://doi.org/10.1177/2515245920965119

Dmitrienko, A., & D'Agostino, R. (2013). Traditional multiplicity adjustment methods in clinical trials. *Statistics in Medicine*, *32*(29), 5172–5218. https://doi.org/10.1002/sim.5990

Fahrmeir, L., Kneib, T., Lang, S., & Marx, B. D. (2021). *Regression: Models, Methods and Applications*. Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-662-63882-8

Faul, F., Erdfelder, E., Buchner, A., & Lang, A.-G. (2009). Statistical power analyses using G*Power 3.1: Tests for correlation and regression analyses. *Behavior Research Methods*, *41*(4), 1149–1160. https://doi.org/10.3758/BRM.41.4.1149

Green, P., & MacLeod, C. J. (2016). SIMR: An R package for power analysis of generalized linear mixed models by simulation. *Methods in Ecology and Evolution*, *7*(4), 493–498. https://doi.org/10.1111/2041-210X.12504

Hothorn, T., Bretz, F., & Westfall, P. (2008). Simultaneous Inference in General Parametric Models. *Biometrical Journal*, *50*(3), 346–363. https://doi.org/10.1002/bimj.200810425

Kaptein, M. (2016). Using Generalized Linear (Mixed) Models in HCI. In J. Robertson & M. Kaptein (Eds.), *Modern Statistical Methods for HCI* (pp. 251–274). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-26633-6_11

King, M. T. (2011). A point of minimal important difference (MID): A critique of terminology and methods. *Expert Review of Pharmacoeconomics & Outcomes Research*, *11*(2), 171–184. https://doi.org/10.1586/erp.11.9

Kumle, L., Võ, M. L.-H., & Draschkow, D. (2021). Estimating power in (generalized) linear mixed models: An open introduction and tutorial in R. *Behavior Research Methods*, *53*(6), 2528–2543. https://doi.org/10.3758/s13428-021-01546-0

Lafit, G., Adolf, J. K., Dejonckheere, E., Myin-Germeys, I., Viechtbauer, W., & Ceulemans, E. (2021). Selection of the Number of Participants in Intensive Longitudinal Studies: A User-Friendly Shiny App and Tutorial for Performing Power Analysis in Multilevel

Regression Models That Account for Temporal Dependencies. *Advances in Methods and Practices in Psychological Science*, *4*(1), 251524592097873. https://doi.org/10.1177/2515245920978738

Lakens, D. (2022a). *Improving Your Statistical Inferences*. Zenodo. https://doi.org/10.5281/ZENODO.6409077

Lakens, D. (2022b). Sample Size Justification. *Collabra: Psychology*, *8*(1), 33267. https://doi.org/10.1525/collabra.33267

Lakens, D., Adolfi, F. G., Albers, C. J., Anvari, F., Apps, M. A. J., Argamon, S. E., … Zwaan, R. A. (2018). Justify your alpha. *Nature Human Behaviour*, *2*(3), 168–171. https://doi.org/10.1038/s41562-018-0311-x

Maxwell, S. E., Kelley, K., & Rausch, J. R. (2008). Sample Size Planning for Statistical Power and Accuracy in Parameter Estimation. *Annual Review of Psychology*, *59*(1), 537–563. https://doi.org/10.1146/annurev.psych.59.103006.093735

Murayama, K., Usami, S., & Sakaki, M. (2022). Summary-statistics-based power analysis: A new and practical method to determine sample size for mixed-effects modeling. *Psychological Methods*. https://doi.org/10.1037/met0000330

Robertson, J., & Kaptein, M. (2016). Improving Statistical Practice in HCI. In J. Robertson & M. Kaptein (Eds.), *Modern Statistical Methods for HCI* (pp. 331–348). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-26633-6_14

Vaughan, D., & Dancho, M. (2022). *Furrr: Apply mapping functions in parallel using futures*. Retrieved from https://CRAN.R-project.org/package=furrr

Westfall, J., Kenny, D. A., & Judd, C. M. (2014). Statistical power and optimal design in experiments in which samples of participants respond to samples of stimuli. *Journal of Experimental Psychology: General*, *143*, 2020–2045. https://doi.org/10.1037/xge0000014

Yarkoni, T. (2022). The generalizability crisis. *Behavioral and Brain Sciences*, *45*, e1. https://doi.org/10.1017/S0140525X20001685

Zimmer, F., Henninger, M., & Debelak, R. (2022). *Sample Size Planning for Complex Study*

Designs: A Tutorial for the mlpwr Package. PsyArXiv. https://doi.org/10.31234/osf.io/r9w6t