Universität Trier

Survey-weighted Generalized Linear Mixed Models

Jan Pablo Burgard
Patricia Dörr

Research Papers in Economics
No. 1/18

Survey-weighted Generalized Linear Mixed Models

Jan Pablo Burgard[1] and Patricia Dörr[2]

[1,2] *Economic and Social Statistics Department, Trier University, Germany*

January 2018

Abstract

Regression analysis aims at the revelation of interdependencies and causalities between variables observed in the population. That is, a structure between regressors and regressants that causes the realization of the finite population is assumed, the so-called data generating process or a superpopulation model. When data points occurr in an inherent clustering, mixed models are a natural modelling approach.
Given the finite population realization, a consistent estimation of the superpopulation parameters is possible. However, regression analysis seldomly takes place at the level of the finite population. Rather, a survey is conducted on the population and the analyst has to use the sample for regression modeling. Under a correct regression setup, derived estimators are consistent given the sample is non-informative. Though, these conditions are hard to verify, especially when the survey design is complex, employing clustering and unequal selection probabilities. The use of sampling weights may reduce a consequent estimation bias as they could contain additional information about the sampling process conditional on which the data generating process of the sampled units becomes closer to the one of the whole population.
Common estimation procedures that allow for survey weights in generalized linear mixed models require one unique survey-weight per sampling stage which are consequently nested and correspond to the random effects analyzed in the regression. However, the data inherent clustering (e.g. students in classes in schools) possibly does not correspond to the sampling stages (e.g. blocks of houses where the students' families live). Or the analyst has no access to the detailed sample design due to disclosure risk or the selection of units follows an unequal sampling probability scheme. Or the survey weights vary within clusters due to calibration. Therefore, we propose an estimation procedure that allows for unit-specific survey weights: The Monte-Carlo EM (MCEM) algorithm whose complete-data log-likelihood leads to a single-level modeling problem that allows a unit-specific weighting. In the E-step, the random effects are considered to be missing data. The expected (weighted) log-likelihood is approximated via Monte-Carlo integration and maximized with respect to the regression parameters. The method's performance is evaluated in a model-based simulation study with finite populations.

1 Introduction

Research institutes often conduct surveys using a complex survey design, either due to costs or due to optimality considering the precision of total estimators. This, on the other hand, complicates regression analysis using these surveys because the sampling might be informative. Further, there remains the question about the correctness of the modelling. In both cases, survey weights can contain the additional information conditional on which the sample's distribution corresponds to that of the finite population and thus allows conclusions about the superpopulation. Therefore, it may be useful to include survey weights in the analysis in order to get less biased regression parameters. An overview about the use of survey weights in regression analysis in general is given in [15].
A common regression setup is that of Generalized Linear Mixed Models (GLMMs). The maximum-likelihood (ML) estimators require to integrate on the random effects assumed to be in the regression model but not explicitely observed in the sample data. These integration turns survey-weighting problematic when the sampling stages are not supposed to correspond to the random effects structure. For example take a survey with clusters on house blocks. It is probable that children of the same block go to the same school. When an analysis about school performance is conducted, however, random effects on schools and classes are sensible which do not perfectly cover house blocks. Furthermore, weight calibration to known population totals by the survey designer or nonresponse adjustments could return unit-specific survey weights even if the regression setup reflected a clustered survey designs with equal selection probabilities within clusters.
For nested random effects, there is the chance to do a pseudo-likelihood estimation where each integral is weighted with respect to its inverse inclusion probability [18]. In the linear mixed model case (LMM), this approach is equivalent to iterative generalized least squares [16]. Besides the inconvenients listed above, the survey weights enter the likelihood as exponent and therefore their scaling is relevant for the estimation process [18]. In order to avoid this scaling problem, it is more convenient to consider the log-likelihood. Taking the logarithm, the survey weights are simple prefactors whose scale does not matter. In that sense, our estimation approach is similar to the estimating equations (that are often the score function) proposed in [20], although the cited authors consider exclusively the LMM scenario and restrict themselves to the two-stage cluster sampling scheme which simplifies formulas. However, we chose to keep the random effects structure (nested and/or crossed) as flexible as possible. Additionally, we want a single survey-weight per unit to be sufficient for the regression analysis rather to require the data user to have access to the inclusion probabilities of every stage of the survey design.
Given a realization of the random effects, population observations are considered to be independent, which turns the model under the sample to an estimation problem with missing (unobserved) data. Once we manage to predict the random effects, the mixed model reduces to a single-level regression where survey weights are easily implemented. Having the regression parameters, random effect predictions can be updated, leading to an iterative procedure well known as Expectation Maximization (EM) algorithm [5]. However, the expected (survey-weighted) log-likelihood can be difficult or even impossible to derive analytically when the dependent variable is binary or count data; for this reason we approx-

imate it through Monte-Carlo (MC) integration.
Section 2 introduces the estimation algorithm, discusses briefly the estimators consistency and highlights the problematic of the computational implementation. Section 3 handles briefly the variance estimation of the estimated regression parameters. Afterwards, a simulation study demonstrates the possible gains of the survey-weighted GLMM. The final section discusses possible further research and concludes.

2 Estimation Procedure

2.1 Population Loglikelihood

Let $\mathcal{U}$, $|\mathcal{U}| = N$ denote the universe from which sample $\mathcal{S}$, $|\mathcal{S}| = n$ is drawn under a complex survey design. This means that we employ a frequentist approach where each unit indexed i in the population, $i \in \mathcal{U}$, $i = 1, \ldots, N$, has a positive probability of being sampled. This inclusion probability is denoted by $\pi_i := P(\delta_{\mathcal{S}}(i) = 1)$, where $\delta_{\mathcal{S}}(i)$ is the i^{th} coordinate projection from an adequate probability space $(\Omega, \mathcal{A}, P)$ to $\left(\{0,1\}^N, \mathcal{P}\left(\{0,1\}^N\right)\right)$ and $\mathbf{I}_{\mathcal{S}} = (\delta_{\mathcal{S}}(1), \ldots, \delta_{\mathcal{S}}(N))^T$. The $N \times 1$ vector $\mathbf{w}$ is the vector of survey weights, which contain information on the sampling design, $\mathbf{w} = \mathbf{w}(P)$. Usually, survey weight w_i is the inverse inclusion probabilty. In the aftermath of sampling, the weights may be adapted to non-response and/or calibrated to known population values of auxiliary variables.
Assume that the finite population is a realization of the following superpopulation model, that is, we employ the term superpopulation in the sense of [13], [4] and [20]. In other words, the finite population is a realization of a Generalized Linear Mixed Model (GLMM):

$$\eta_i = \mathbf{x}_i^T \boldsymbol{\beta} + \mathbf{z}_i^T \boldsymbol{\gamma} \tag{2.1}$$
$$\mu_i = g(\eta_i) \tag{2.2}$$
$$Y_i \sim F(\mu_i, \boldsymbol{\varphi}) \tag{2.3}$$
$$G \sim N(\mathbf{0}, \Sigma) \tag{2.4}$$

where $\mathbf{x}_i$ and $\mathbf{z}_i$ are non-random explanatory variables. While $\mathbf{x}_i$ is linked to the linear prediction η_i through a fixed effects vector $\boldsymbol{\beta}$, $\mathbf{z}_i$ interacts with the multivariate gaussian random effects realization $G = \boldsymbol{\gamma}$. F is a distribution from the exponential family with corresponding density f, scale parameter $\boldsymbol{\varphi}$ and inverse link function g. Because $\boldsymbol{\varphi}$ is only a scaling parameter whose estimation is no problem and can be separately done [22], we omit it in the following formulas. As Σ is a positive definite symmetric matrix, let $\boldsymbol{\sigma}$ denote the vector of distinct matrix elements in it, $\Sigma = \Sigma(\boldsymbol{\sigma})$. Further, for brevity, write $\boldsymbol{\psi}^T := (\boldsymbol{\beta}^T, \boldsymbol{\sigma}^T)^T$. When assuming the canonical link, $\log f$ has the following shape:

$$\log f(y_i|\boldsymbol{\gamma}, \boldsymbol{\psi}) = y_i \eta_i - a(\eta_i) + b(y_i) \tag{2.5}$$

with $\frac{\partial a(\eta_i)}{\partial \eta_i} = \mu_i$. Note that the term $b(y_i)$ does not depend on the parameter vector $\boldsymbol{\psi}$ and can be ignored later on for optimization. To conclude the notation, take $\mathbf{x}_i$ and $\mathbf{z}_i$ as i^{th} row of matrices X and Z. Consequently, $X \in \mathbb{R}^{N \times p}$ and $Z \in \mathbb{R}^{N \times q}$.

This leads to the finite population likelihood

$$\begin{aligned}\mathcal{L}(\mathbf{y}, X, Z, \boldsymbol{\psi}, \boldsymbol{\gamma}) &= \prod_{i \in \mathcal{U}} f(y_i, \mathbf{x}_i, \mathbf{z}_i, \boldsymbol{\gamma}, \boldsymbol{\psi}) \\ &\propto \prod_{i \in \mathcal{U}} f(y_i | \mathbf{x}_i, \mathbf{z}_i, \boldsymbol{\gamma}, \boldsymbol{\psi}) \cdot \phi(\boldsymbol{\gamma} | \boldsymbol{\sigma}), \end{aligned} \tag{2.6}$$

where $\phi(\cdot|\boldsymbol{\sigma})$ is the density of the mulitvariate normal $N(\mathbf{0}, \Sigma)$. Due to better readability, we omit X and Z in the following, though the functions are conditioned on them. Taking the logarithm of (2.6) leads to

$$\mathcal{LL}(\mathbf{y}, \boldsymbol{\psi}, \boldsymbol{\gamma}) = \sum_{i \in \mathcal{U}} \log f(y_i | \boldsymbol{\gamma}, \boldsymbol{\psi}) + \log \phi(\boldsymbol{\gamma} | \boldsymbol{\sigma}). \tag{2.7}$$

We define the finite population parameter vector $\boldsymbol{\psi}^{pop}$ and the subvectors $\boldsymbol{\beta}^{pop}$ and $\boldsymbol{\sigma}^{pop}$ it is composed of, as

$$\boldsymbol{\psi}^{pop} = \arg\max_{\boldsymbol{\psi}} \mathcal{LL}, \tag{2.8}$$

That means, $\boldsymbol{\psi}^{pop}$ would be the maximum likelihood estimate of the superpopulation parameter if the total finite population $\mathcal{U}$ was observed. $\boldsymbol{\beta}^{pop}$ exists and is unique for the common Generalized Linear Models (GLM) such as Poisson-regression under the log-link, logistic regression and standard linear regression [22]. The population covariance matrix $\Sigma(\boldsymbol{\sigma}^{pop})$ is unique, too, thus $\boldsymbol{\psi}^{pop}$ is well defined. Later on, we will develop an EM [5] based algorithm to get an estimator of (2.8). Note that the EM-algorithm converges always to stationary points [3] and as (2.7) is globally concave, this means convergence to the global maximum of $\mathcal{LL}$. As the maximum is unique, this means that the parameter estimate will converge to the maximizer, that is $\boldsymbol{\psi}^{pop}$, too [23].
Even at the finite population level, the random effect realization $G = \boldsymbol{\gamma}$ is unobserved and thus direct evaluation of (2.7) is not possible. A common way to circumvent this problem is the application of the EM-algorithm [5], a method that has readily been applied to mixed models [8]. The expectation of (2.7) is calculated given a parameter estimate $\hat{\boldsymbol{\psi}}_s$ (E-step). Then, the expectation is maximized, yielding parameter estimate $\hat{\boldsymbol{\psi}}_{s+1}$ (M-step) that allows a new evaluation of the expectation. Concretely, the expectation of (2.7) is

$$\begin{aligned}E_m\left(\mathcal{LL}\right) = &\sum_{i \in \mathcal{U}} \int \left(\log f(y_i | \boldsymbol{\gamma}, \boldsymbol{\psi})\right) \cdot h(\boldsymbol{\gamma} | \mathbf{y}, \boldsymbol{\sigma}) \mathrm{d}\lambda(\boldsymbol{\gamma}) \\ &+ \int \left(\log \phi(\boldsymbol{\gamma} | \boldsymbol{\sigma})\right) \cdot h(\boldsymbol{\gamma} | \mathbf{y}, \boldsymbol{\sigma}) \mathrm{d}\lambda(\boldsymbol{\gamma}), \end{aligned} \tag{2.9}$$

where h is the conditional distribution density of $\boldsymbol{\gamma}$ given $\mathbf{y}$. We will refer to h as "posterior" density because it contains - in contrast to ϕ - information about the realizations of Y, meaning that it includes information *after* the finite population has been sampled from the superpopulation. In contrast, ϕ may be referred to as "prior". However, we stress that the framework is not Bayesian, we consider the superpopulation parameter $\boldsymbol{\psi}$ as fixed. E_m denotes expectation under the model, in order to distinguish from expectation under the sampling design, that is marked with subscript d.

As the population $\mathcal{U}$ is not completely observed in practice, we seek to combine estimation and maximization of (2.9) with sampling *design* randomization, characterized by probability measure P_d, that possibly hurts the self-weighting property: This means, we do not exclude the possibility that the unweighted sample does not obey the model defined in (2.1) to (2.4) [19, chap 5.3]. However, this property is necessary for consistent parameter estimation under an unweighted regression model. This means, possibly $f(\mathbf{y}, \boldsymbol{\gamma}, \mathbf{I}_\mathcal{S}) \neq P_d(\mathbf{I}_\mathcal{S}) \cdot f(\mathbf{y}, \boldsymbol{\gamma})$. $\mathbf{I}_\mathcal{S} = (\delta_\mathcal{S}(1), \ldots, \delta_\mathcal{S}(N))^T$, $\delta_\mathcal{S}$ being an indicator function for the index set $\mathcal{S}$. In other words, the sampling design is not ignorable and inference may be misleading [15].
For any element $\boldsymbol{\psi}$ in the parameter space, (2.7) and (2.9) may be seen as finite population totals which can be estimated through the Horvitz-Thompson estimator [7] when having available only a sample $\mathcal{S} \subset \mathcal{U}$:

$$\begin{aligned}\hat{E}_m\left(\mathcal{LL}\right) = &\sum_{i \in \mathcal{U}} w_i \cdot \delta_\mathcal{S}(i) \cdot \int \left(\log f(y_i|\boldsymbol{\gamma}_\mathcal{S}, \boldsymbol{\psi})\right) \cdot h(\boldsymbol{\gamma}_\mathcal{S}|\mathbf{y}, \boldsymbol{\sigma}) \mathrm{d}\lambda(\boldsymbol{\gamma}_\mathcal{S}) \\ &+ \int \left(\log \phi(\boldsymbol{\gamma}_\mathcal{S}|\boldsymbol{\sigma})\right) \cdot h(\boldsymbol{\gamma}_\mathcal{S}|\mathbf{y}, \boldsymbol{\sigma}) \mathrm{d}\lambda(\boldsymbol{\gamma}_\mathcal{S}), \end{aligned} \tag{2.10}$$

$\boldsymbol{\gamma}_\mathcal{S}$, being the subvector of random effects in the population that has made it into the sample:

$$\boldsymbol{\gamma} = \begin{pmatrix} \boldsymbol{\gamma}_\mathcal{S} \\ \boldsymbol{\gamma}_{\mathcal{U} \setminus \mathcal{S}} \end{pmatrix}.$$

This idea is similar to [4], though we calculate the HT of an expected likelihood and not of the likelihood itself. Note that h is not available, neither, when not the complete vector $\mathbf{y}$ is known, but only elements of the sample $\mathcal{S}$, which means, that h must be estimated, too. Therefore, we prefer the following definition of $\widehat{E_m(\mathcal{LL})}$ that we use henceforth:

$$\begin{aligned}\widehat{E_m\left(\mathcal{LL}\right)} = &\sum_{i \in \mathcal{U}} w_i \cdot \delta_\mathcal{S}(i) \cdot \int \left(\log f(y_i|\boldsymbol{\gamma}_\mathcal{S}, \boldsymbol{\psi})\right) \cdot \hat{h}(\boldsymbol{\gamma}_\mathcal{S}|\mathbf{y}, \boldsymbol{\sigma}) \mathrm{d}\lambda(\boldsymbol{\gamma}) \\ &+ \int \left(\log \phi(\boldsymbol{\gamma}_\mathcal{S}|\boldsymbol{\sigma})\right) \cdot \hat{h}(\boldsymbol{\gamma}_\mathcal{S}|\mathbf{y}, \boldsymbol{\sigma}) \mathrm{d}\lambda(\boldsymbol{\gamma}_\mathcal{S}) \end{aligned} \tag{2.11}$$

Therefore, (2.11) is not completely equivalent to a classical HT-estimator. However, we can profit from the fact that h is proportional to $\exp \mathcal{LL}$ and $\mathcal{LL}$ could be estimated consistently as outlined in the previous paragraph if the random effect realizations were observed. The exact implementation of the estimation of $\hat{h}$ is described in the next section.
Another note of caution be on the term $\log \phi(\boldsymbol{\gamma}|\boldsymbol{\sigma})$. This term actually has - up to a normalizing constant - the structure

$$\log \phi(\boldsymbol{\gamma}|\boldsymbol{\sigma}) = -\frac{1}{2} \boldsymbol{\gamma}^T \Sigma^{-1} \boldsymbol{\gamma} - \log \det \Sigma.$$

Consequently, if the random effects do not include all random effects present in the population, i.e. if the domains are unplanned and $\boldsymbol{\gamma}_\mathcal{S} \neq \boldsymbol{\gamma}$, the second term of (2.11) needs further attention: Then, of course the covariance matrix $\Sigma_\mathcal{S}$ is

neither the population covariance matrix Σ. Then, (2.11) is not any longer a consistent estimator of the expected population likelihood as

$$-\frac{1}{2}\boldsymbol{\gamma}^T\Sigma^{-1}\boldsymbol{\gamma} - \log\det\Sigma \neq -\frac{1}{2}\boldsymbol{\gamma}_{\mathcal{S}}^T\Sigma_{\mathcal{S}}^{-1}\boldsymbol{\gamma}_{\mathcal{S}} - \log\det\Sigma_{\mathcal{S}}$$

and thus, the proportionality of h to $\exp\mathcal{LL}$ does not hold neither. On the other hand, $\log\phi(\boldsymbol{\gamma}_{\mathcal{S}}|\boldsymbol{\gamma})$ may be estimated at the sample level, that is, under the measure $\delta_{\mathcal{S}}(i)$ rather than $\delta_{\mathcal{U}}(i)$. Therefore, if the first term of (2.11) could be modified to return a consistent estimate of a simple random sample likelihood, the proportionality and consistency would be restablished. Referring to the proposition of Stolz-Cesàro, we propose to rescale the weights to

$$\frac{w_i}{\sum_{i\in\mathcal{S}} w_i}\cdot|\mathcal{S}|.$$

In the following, we assume the survey-weights to be adequatly rescaled and do not further distinguish between $\boldsymbol{\gamma}_{\mathcal{S}}$ and $\boldsymbol{\gamma}$.

2.2 Parameter Estimation

When (2.9) is considered to be a total, then $\boldsymbol{\psi}^{pop}$ is a functional of a total. Furthermore, for common GLMM regressions, $\mathcal{LL}$ is twice continuously differentiable and thus $\boldsymbol{\psi}^{pop}$ is implicitely defined as the solution to

$$\nabla\mathcal{LL}(\mathbf{y},\boldsymbol{\psi},\boldsymbol{\gamma}) = \mathbf{0}. \tag{2.12}$$

So $\arg\max$ is a continuous and differentiable function. Therefore, a natural choice to estimate $\boldsymbol{\gamma}^{pop}$ is the plug-in estimator

$$\hat{\boldsymbol{\psi}} = \arg\max_{\boldsymbol{\psi}} \widehat{E_m\left(\mathcal{LL}\right)}. \tag{2.13}$$

To get $\hat{\boldsymbol{\psi}}$, the EM-algorithm [5] is applied to the sample rather than to the (unavailable) finite population. For the moment, assume that we had access to h, and therefore (2.10) rather than (2.11) could be applied. Then, the algorithm would be in pseudo-code

Algorithm 2.1 EM-algorithm for GLMMs

Require: $S \in \mathbb{N}$, $\boldsymbol{\psi}_1$, $\varepsilon > 0$

for $s = 1,\ldots,S$ **do** ▷ Start EM-Loop

Calculate $\hat{E}_m(\mathcal{LL})\left(\mathbf{y},\boldsymbol{\psi}_s\right)$ ▷ E-Step

$\boldsymbol{\psi}_{s+1} \leftarrow \arg\max_{\boldsymbol{\psi}} \hat{E}_m(\mathcal{LL})\left(\mathbf{y},\boldsymbol{\psi}\right)$ ▷ M-Step

$d \leftarrow \frac{|\boldsymbol{\psi}_{s+1}-\boldsymbol{\psi}_s|}{|\boldsymbol{\psi}_s|}$

if $d < \varepsilon$ **then** ▷ Stopping rule

$S \leftarrow s$

break;

end if ▷ One EM-Step completed

end for

Ensure: $\hat{\boldsymbol{\psi}} \leftarrow \boldsymbol{\psi}_S$

Note that in (2.1), it is ignored that we do neither know h nor an estimator for $\hat{h}$ has been proposed. In addition, even for known h, evaluation of the integral in (2.10) might be cumbersome: The dimension of the integral equals the number of elements in $\boldsymbol{\gamma}$, the normalizing constant of h, C, is unknown:

$$h(\boldsymbol{\gamma}|\mathbf{y}, \boldsymbol{\psi}) = \left(\prod_{i \in \mathcal{U}} f(y_i|\boldsymbol{\beta}, \boldsymbol{\gamma}) \cdot \phi(\boldsymbol{\gamma}|\boldsymbol{\sigma}) \right) \cdot C^{-1} \tag{2.14}$$

$$C = \int \left(\prod_{i \in \mathcal{U}} f(y_i|\boldsymbol{\beta}, \boldsymbol{\gamma}) \cdot \phi(\boldsymbol{\gamma}|\boldsymbol{\sigma}) \right) \mathrm{d}\lambda(\boldsymbol{\gamma})$$

Besides, except for the LMM framework, h is a non-standard density from the exponential family [2]. Therefore, several authors propose to approximate the E-step in algorithm (2.1) by MC-integration ([12], [2], [24]), leading to a Monte Carlo EM, MCEM. Assume still that the "posterior" h was known. Then algorithm (2.1) would change to

Algorithm 2.2 MCEM-algorithm for GLMMs

Require: $S, B \in \mathbb{N}$, $\boldsymbol{\psi}_1$, $\varepsilon > 0$

for $s = 1, \ldots, S$ **do** ▷ Start EM-Loop

for $b = 1, \ldots, B$ **do** ▷ MC-Loop $\hat{=}$ E-Step

Sample $\boldsymbol{\gamma}_b \sim h(\boldsymbol{\gamma}|\mathbf{y}, \boldsymbol{\psi}_s, \mathbf{I}_{\mathcal{S}})$

end for ▷ End E-Step

$\boldsymbol{\psi}_{s+1} \leftarrow \arg\max_{\boldsymbol{\psi}} \sum_{b=1}^{B} \widehat{\mathcal{LL}}(\mathbf{y}, \boldsymbol{\psi}, \boldsymbol{\gamma}_b) = \hat{E}_m^{MC}(\widehat{\mathcal{LL}})$ ▷ M-Step

$d \leftarrow \frac{|\boldsymbol{\psi}_{s+1} - \boldsymbol{\psi}_s|}{|\boldsymbol{\psi}_s|}$

if $d < \varepsilon$ **then** ▷ Stopping rule

$S \leftarrow s$

break;

end if ▷ One EM-Step completed

end for

Ensure: $\hat{\boldsymbol{\psi}} \leftarrow \boldsymbol{\psi}_S$

However, as it was mentioned, h might be a non-standard density, high-dimensional and is thus not implemented in standard statistical software. Therefore, sampling from h (or $\hat{h}$ respectively) is rarely directly possible. An alternative is MC-sampling. This was proposed in [12], [2] and [24]. Among the proposed sampling algorithms are rejection sampling [2] and Metropolis Hastings (MH) [12], [2]. Due to the high-dimensionality of the sampling process, the rejection rate is relatively high for both algorithms so that the computational effort was too high in our first simulation studies. Especially, the use of the "prior" ϕ as proposal density as proposed in these works is problematic when the realizations $G = \boldsymbol{\gamma}$ are relatively far from the origin. In the working paper [24], Quasi-MC with a spherical radial transformation was proposed. This procedure requires the generation of at least one random orthogonal matrix of the same dimension as Σ per E-step, which becomes quickly critical, too. For

this reason, we concentrate on importance sampling like in [12] and [2]. Using proposal density l, a random draw $\boldsymbol{\gamma}_b$, $b = 1, \ldots, B$ has in general the following importance weight

$$w_b^{MC} = \left(\frac{h(\boldsymbol{\gamma}_b)}{l(\boldsymbol{\gamma}_b)} \right) \cdot \left(\sum_{b=1}^{B} \frac{h(\boldsymbol{\gamma}_b)}{l(\boldsymbol{\gamma}_b)} \right)^{-1}. \tag{2.15}$$

Several things about (2.15) are noteworthy: First, it is sufficient to know both, the proposal density l and the aimed distribution h only up to a normalizing constant. Division with the sum of importance weights, $\sum_{b=1}^{B} \frac{\exp \widehat{\mathcal{LL}}(\boldsymbol{\gamma}_b)}{l(\boldsymbol{\gamma}_b)}$ ensures that the (unknown) normalizing constants, those of l and C (cf. eq (2.14)) cancel out.
This leads to the next point: $h(\boldsymbol{\gamma}_b) \propto \exp \mathcal{LL}(\boldsymbol{\gamma}_b)$. However, given realization $\boldsymbol{\gamma}_b$, $\mathcal{LL}$ can be estimated consistently (even unbiasedly) through

$$\widehat{\mathcal{LL}} = \sum_{i \in \mathcal{U}} w_i \delta_{\mathcal{S}}(i) \log f(y_i | \boldsymbol{\gamma}_b, \boldsymbol{\psi}) + \log \phi(\boldsymbol{\gamma}_b | \boldsymbol{\sigma})$$

A possible bias of $\hat{h}(\boldsymbol{\gamma}_b) \propto \exp \widehat{\mathcal{LL}}(\boldsymbol{\gamma}_b)$ cancels out in the importance sample process when w_b^{MC} is normalized through $\sum_{b=1}^{B} \frac{\exp \widehat{\mathcal{LL}}(\boldsymbol{\gamma}_b)}{l(\boldsymbol{\gamma}_b)}$ in analogy to (2.15). Therefore, we have now an (unnormalized) estimator of h, $\hat{h}$. Consequently, by the strong law of large numbers ([14, chap 9])

$$\sum_{b=1}^{B} w_b^{MC} \widehat{\mathcal{LL}}(\boldsymbol{\gamma}_b, \hat{\boldsymbol{\psi}}_s, \mathbf{y}) \xrightarrow[B \to \infty]{a.s} \widehat{E_m(\mathcal{LL})}$$

and as $\exp \widehat{\mathcal{LL}}$ is a consistent estimator of the unnormalized h, we have with respect to the randomization probability measure P_d

$$\operatorname*{p\,lim}_{n \to \infty} \left(\lim_{B \to \infty} \sum_{b=1}^{B} w_b^{MC} \widehat{\mathcal{LL}}(\boldsymbol{\gamma}_b, \hat{\boldsymbol{\psi}}_s, \mathbf{y}) \right) = \operatorname*{p\,lim}_{n \to \infty} \widehat{E_m(\mathcal{LL})} = E_m(\mathcal{LL}). \tag{2.16}$$

The sample size n is implicitely contained in $\widehat{\mathcal{LL}}$, which is an estimated total of $\mathcal{LL}$, $|\mathcal{S}| = n$. Consequently, the algorithm returns a design-consistent estimator of the expected log-likelihood in the population. We have alreadey stated that the maximum point of $E_m(\mathcal{LL})$ is unique in our set-up and as the first derivative is obviously continuous (cf. eq. (2.9)), arg max is a continuous and measurable function, too. Then however, plugging a consistent estimator $\widehat{E_m(\mathcal{LL})}$ into arg max yields a design-consistent estimator, too. And $\boldsymbol{\psi}^{pop}$ is itself a consistent estimator of the superpopulation parameter $\boldsymbol{\psi}$ [21].
Finally, only the question about a good proposal distribution l is open. An easy-to-implement suggestion would be the multivariate normal distribution, especially with the variance matrix $\Sigma(\boldsymbol{\sigma})$, this lead to the importance weight

$$w_b^{MC} = \left(\frac{\exp \widehat{\mathcal{LL}}(\boldsymbol{\gamma}_b)}{\phi(\boldsymbol{\gamma}_b | \boldsymbol{\sigma})} \right) \cdot \left(\sum_{b=1}^{B} \frac{\exp \widehat{\mathcal{LL}}(\boldsymbol{\gamma}_b)}{\phi(\boldsymbol{\gamma}_b | \boldsymbol{\sigma})} \right)^{-1} = \frac{\exp \sum_{i \in \mathcal{S}} w_i f(y_i | \boldsymbol{\gamma}_b, \boldsymbol{\psi})}{\sum_{b=1}^{B} \exp \sum_{i \in \mathcal{S}} w_i f(y_i | \boldsymbol{\gamma}_b, \boldsymbol{\psi})}.$$

This is equivalent to the proposal suggested in [12] for the MH-algorithm. However, due to the high-dimensionality, the variability of the importance weights

may become very high so that this approach is only practicable when the number of random effects is very small. In SAE, this is rarely the case.
[2] suggests to use as proposal l a multivariate t-distribution where mode and the Hessian of $\widehat{\mathcal{LL}}$ match the first and second moments. However, our simulation results were rather good using a multivariate normal. The multivariate normal can be motivated through a second-order Taylor approximation. For EM-step s, let $\hat{\boldsymbol{\psi}}_s$ be the current estimate of $\boldsymbol{\psi}^{pop}$. Given the estimate, let $\hat{\boldsymbol{\gamma}}_s$ maximize $\widehat{\mathcal{LL}}$. Be $\hat{H}_s$ the Hessian of $\widehat{\mathcal{LL}}$ evaluated at $\hat{\boldsymbol{\gamma}}_s$. Then the gradient $\nabla_\gamma \widehat{\mathcal{LL}}(\hat{\boldsymbol{\gamma}}, \hat{\boldsymbol{\psi}}_s, \mathbf{y}) = \mathbf{0}$ and the Taylor approximation is:

$$\begin{aligned}
\widehat{\mathcal{LL}}(\boldsymbol{\gamma}, \hat{\boldsymbol{\psi}}_s, \mathbf{y}) &\approx \widehat{\mathcal{LL}}(\mathbf{y}, \hat{\boldsymbol{\psi}}_s, \hat{\boldsymbol{\gamma}}_s) + 0.5 \cdot (\boldsymbol{\gamma} - \hat{\boldsymbol{\gamma}}_s)^T \hat{H}_s^{-1} (\boldsymbol{\gamma} - \hat{\boldsymbol{\gamma}}_s) \\
\exp \widehat{\mathcal{LL}}(\boldsymbol{\gamma}, \hat{\boldsymbol{\psi}}_s, \mathbf{y}) &\approx \exp \widehat{\mathcal{LL}}(\hat{\boldsymbol{\gamma}}_s) \cdot \exp\left(-0.5 \cdot (\boldsymbol{\gamma} - \hat{\boldsymbol{\gamma}}_s)^T \left(-\hat{H}_s^{-1}\right) (\boldsymbol{\gamma} - \hat{\boldsymbol{\gamma}}_s)\right) \\
&\propto N\left(\hat{\boldsymbol{\gamma}}_s, -\hat{H}_s^{-1}\right) \qquad (2.17)
\end{aligned}$$

The pseudo-code for the final algorithm inclusive importance sampling in the E-step is given in algorithm (2.3)

Algorithm 2.3 MCEM-algorithm for GLMMs with Importance Sampling

Require: $S, B \in \mathbb{N}$, $\hat{\boldsymbol{\psi}}_1$, $\varepsilon > 0$

for $s = 1, \ldots, S$ **do** ▷ Start EM-Loop

for $b = 1, \ldots, B$ **do** ▷ MC-Loop $\hat{=}$ E-Step

Calculate $\hat{\boldsymbol{\gamma}}_s \leftarrow \underset{\boldsymbol{\gamma} \in \mathbb{R}^q}{\arg\max}\, \widehat{\mathcal{LL}}(\boldsymbol{\gamma}, \boldsymbol{\psi}_s, \mathbf{y})$

Sample $\boldsymbol{\gamma}_b \sim N\left(\hat{\boldsymbol{\gamma}}_s, -\hat{H}_s^{-1}(\widehat{\mathcal{LL}})\right)$ ▷ Sampling from Proposal

end for

Calculate $w_b^{MC} = \frac{\exp\left(\widehat{\mathcal{LL}}(\boldsymbol{\gamma}_b, \hat{\boldsymbol{\psi}}_s, \mathbf{y})\right)}{0.5 \ \det\left(\hat{H}_s^{-1}\right) \cdot \exp\left(0.5 \ (\boldsymbol{\gamma}_b - \hat{\boldsymbol{\gamma}}_s)^T \hat{H}_s (\boldsymbol{\gamma}_b - \hat{\boldsymbol{\gamma}}_s)\right)}$

Normalize $w_b^{MC} \leftarrow \frac{w_b^{imp}}{\sum_{b=1}^B w_b^{imp}}$ ▷ End E-Step

Calculate and set ▷ M-Step

$\hat{\boldsymbol{\psi}}_{s+1} \leftarrow \underset{\boldsymbol{\psi}}{\arg\max} \sum_{b=1}^B w_b^{MC} \cdot \widehat{\mathcal{LL}}(\boldsymbol{\gamma}_b, \boldsymbol{\psi}, \mathbf{y})$

$d \leftarrow \frac{|\boldsymbol{\psi}_{s+1} - \boldsymbol{\psi}_s|}{|\boldsymbol{\psi}_s|}$

if $d < \varepsilon$ **then** ▷ Stopping rule

break;

end if ▷ One EM-Step completed

end for

Ensure: $\hat{\boldsymbol{\psi}} \leftarrow \hat{\boldsymbol{\psi}}_s$

Nevertheless, algorithm (2.3) is again computationally problematic because the Hessian $\hat{H}_s$, again of dimension $q \times q$, must be calculated and inverted in every E-step. Though, importance sampling has the advantage that none of the

sampled vectors is rejected and therefore, it is computationally advantageous in comparison to rejection sampling and MH.
We can think of two potential remedies of the computational effort. First, the mode $\hat{\boldsymbol{\gamma}}_s$ can be calculated via the BFGS-algorithm. The approximate inverse Hessian of the final iteration may be used instead of the exact Hessian. Then, matrix inversion is avoided. Furthermore, using the algorithm of [17], it is possible to get a squared matrix C_s such that $C_s C_s^T = -\hat{H}_s$, which simplifies the draw from a multivariate normal distribution with the desired covariance matrix.
Alternatively, the exact Hessian matrix of $\widehat{\mathcal{LL}}$ has the following structure

$$\begin{aligned} -\hat{H}_s &= Z^T W_s Z + \Sigma_s^{-1} \\ W_s &= \operatorname{diag}\left(w_1 \delta_{\mathcal{S}}(1) \frac{\partial^2 \log f(\hat{\eta}_{1,s})}{\partial \hat{\eta}_{1,s}^2}, \ldots, w_N \delta_{\mathcal{S}}(N) \frac{\partial^2 \log f(\hat{\eta}_{N,s})}{\partial \hat{\eta}_{N,s}^2} \right) \\ \hat{\eta}_{i,s} &= \mathbf{x}_i^T \hat{\boldsymbol{\beta}}_s + \mathbf{z}_i^T \hat{\boldsymbol{\gamma}}_s, \qquad i = 1, \ldots, N \end{aligned} \tag{2.18}$$

The Woodbury-identity yields

$$-\hat{H}_s^{-1} = -\Sigma_s - \Sigma_s Z^T (W^{-1} + Z \Sigma_s Z^T)^{-1} Z \Sigma_s. \tag{2.19}$$

Although the dimension of $W^{-1} + Z\Sigma_s Z^T$ is large (when omitting the observations that are not sampled, the dimension is still $n \times n$), it is often a sparse matrix and symmetric, therefore inversion is relatively cheap. In practice, we have implemented the first alternative as the sparsity pattern of $W^{-1} + Z\Sigma_s Z^T$ varies in each regression model.
To conclude, we remind that, as long as the support of $\log l(\cdot)$ corresponds to the support of $\widehat{\mathcal{LL}}$, the choice of l has no impact on the expectation of (2.11). Hence the choice impacts the practical implementation but not the theoretical considerations outlined before.

3 Variance Estimation of the Regression Parameter

The variance of $\hat{\boldsymbol{\beta}}$ could be approximated using the Fisher information matrix under the EM-algorithm. The relation between the expected and observed data information matrix is discussed in [9] and requires the expected Hessian and gradient of the complete data likelihood. However, these are simple to calculate because we employ MC-integration and thus have the simulated $\boldsymbol{\gamma}_b$, $b = 1, \ldots, B$ and importance weights w_b^{imp} available from the last E-step. Therefore, in analogy to [9], the Fisher information and its estimator may be written as

$$I_\beta = E_m\left(H_\beta(\boldsymbol{\psi}, \boldsymbol{\gamma})\right) - E_m\left(\nabla_\beta \mathcal{LL} \cdot \nabla_\beta^T \mathcal{LL}\right) + E_m\left(\nabla_\beta \mathcal{LL}\right) \cdot E_m\left(\nabla_\beta^T \mathcal{LL}\right) \tag{3.1}$$

And the components can be estimated through the MC-realizations in the last EM-step:

$$\hat{H}_\beta = \sum_{b=1}^{B} H_\beta(\hat{\boldsymbol{\psi}}, \boldsymbol{\gamma}_b) \cdot w_b^{imp} \tag{3.2}$$

$$\hat{E}_m \left(\nabla_\beta \mathcal{LL} \cdot \nabla_\beta^T \mathcal{LL} \right) = \sum_{b=1}^{B} \nabla_\beta \widehat{\mathcal{LL}}(\hat{\boldsymbol{\psi}}, \boldsymbol{\gamma}_b) \cdot \nabla_\beta^T \widehat{\mathcal{LL}}(\hat{\boldsymbol{\psi}}, \boldsymbol{\gamma}_b) \cdot w_b^{imp} \tag{3.3}$$

$$\hat{E}_m \left(\nabla_\beta \mathcal{LL} \right) = \sum_{b=1}^{B} \nabla_\beta \widehat{\mathcal{LL}}(\hat{\boldsymbol{\psi}}, \boldsymbol{\gamma}_b) \tag{3.4}$$

In the last EM-Step, the estimated expectation of the log-likelihood should be at its maximum thus turning the expected score estimate (3.4) to zero. However, as we may have reached the maximum number of iterations without convergence, we include that term nonetheless. Note that the estimated log-likelihood, the draw of $\boldsymbol{\gamma}_b$ and hence the likelihood-maximizer $\hat{\boldsymbol{\psi}}$ take into account survey weights and so does the estimated Fisher information.
For the components $\boldsymbol{\sigma}$ of the random effects covariance matrix $\Sigma(\boldsymbol{\sigma})$ we propose to use the fisher information, too. Here, the second part of the complete data log-likelihood $\mathcal{LL}$ is

$$-0.5 \det \Sigma - 0.5 \; \boldsymbol{\gamma}^T \Sigma^{-1} \boldsymbol{\gamma}$$

and the expected Fisher information of the k-th component σ_k is

$$-0.5 \; E \left(\operatorname{tr} \left(-\Sigma^{-1} \frac{\partial \Sigma}{\partial \sigma_k} \Sigma^{-1} \frac{\partial \Sigma}{\partial \sigma_k} \right) + \boldsymbol{\gamma}^T \Sigma^{-1} \frac{\partial \Sigma}{\partial \sigma_k} \Sigma^{-1} \frac{\partial \Sigma}{\partial \sigma_k} \Sigma^{-1} \boldsymbol{\gamma} \right). \tag{3.5}$$

While the first term of (3.5) is constant, calculus of the second term is again no problem thanks to the MC-integration in the previous step.
Though the random effect predictions are not parameters, we can nonetheless wish to give some Mean Squared Prediction Error of $\hat{\boldsymbol{\gamma}}$. $\hat{\boldsymbol{\gamma}}$ maximizes $\widehat{\mathcal{LL}}$ given $\hat{\boldsymbol{\psi}}$. Consequently, a Taylor approximation of $\nabla_\gamma \widehat{\mathcal{LL}}$ around $\boldsymbol{\gamma}^\star$ yields

$$\begin{aligned} 0 = \nabla_\gamma \widehat{\mathcal{LL}}(\hat{\boldsymbol{\gamma}}) &\approx \nabla_\gamma \widehat{\mathcal{LL}}(\boldsymbol{\gamma}^\star) + \hat{H} \left(\hat{\boldsymbol{\gamma}} - \boldsymbol{\gamma}^\star \right) \\ \Leftrightarrow E_d \left((\hat{\boldsymbol{\gamma}} - \boldsymbol{\gamma}^\star)^2 \right) &\approx \hat{H}^{-1} \mathrm{Var}_d \left(\nabla_\gamma \widehat{\mathcal{LL}} \right) \hat{H}^{-1} \end{aligned} \tag{3.6}$$

The (approximate) Hessian from the last EM-step is available, cf. (2.3) and $\mathrm{Var}\left(\nabla_\gamma \widehat{\mathcal{LL}} \right)$ is simply the variance of a total. When $\mathrm{Var}_d \left(\nabla\gamma \widehat{\mathcal{LL}} \right)$ is identified with the Hessian, this gives again the Cramér-Rao bound as an approximation of the squared prediction error.

4 Simulation Study

4.1 Simulation Set-up

In order to account for both - the superpopulation setup and the survey design - we conduct a model-based simulation study under finite populations. That

is, in each of the $K = 1000$ simulation runs, a population of size $N = 3000$ is generated according to the following superpopulation model

$$\eta_i^k = 4 - 2 \cdot x_{1,i} - x_{2,i} + \gamma_{d_1}^k + (1,\ x_{1,i}) \cdot \boldsymbol{\gamma}_{d_2}^k \tag{4.1}$$

$$\gamma_{d_1}^k \sim N(0, 2^2) \qquad , d_1 = 1, \ldots, 20 \tag{4.2}$$

$$\boldsymbol{\gamma}_{d_2}^k \sim N\left(\mathbf{0}, \begin{pmatrix} 0.7 & 0.5 \\ 0.5 & 1.3 \end{pmatrix}\right) \qquad , d_2 = 1, \ldots, 30 \tag{4.3}$$

$$Y_i^k \sim F \tag{4.4}$$

The parameters are chosen that the linear predictors return sensible expectations μ_i^k under various exponential family distributions $F \in \mathcal{F}$, namely the normal ($g = \text{id}$) and the Bernoulli ($g = \text{logit}^{-1}$) distribution.
Next, in each finite population, we sample once under the following sampling schemes. First, the population is stratified according to domains $d_1 = 1, \ldots, 20$. All strata are of equal size and allocation is equal, too: $N_{d_1} = 150$ and $n_{d_1} = 25$. That means, while these domains are planned, domains $d_2 = 1, \ldots, 0$ are unplanned and cross the sampling strata. That is, alternative survey-weighted estimation procedures as discussed in the introduction, would have troubles to take into account random effects $\boldsymbol{\gamma}_{d_2}$ and as the domains d_2 are unplanned, their inclusion requires us to rescale the survey weights as previously mentioned. Domain affiliation of unit i is constant over all simulation runs. Second, we sample in each strata with unequal probability. The first design is uninformative, the inclusion probability for unit i in strata d_1 is proportional to the total of X_2 in the respective domain:

$$\pi_{d_1,i} = \frac{x_{2,d_1,i}}{\sum_{i=1}^{N_{d_1}} x_{2,d_1,i}}.$$

As X_2 is observed, and we estimate the correct model, survey-weighting should not have an effect under this sampling design. Therefore, this design is some sort of control for the estimation quality when weighting is not necessary. Using the weights nontheless should lead to some loss in efficiency but should not impact the estimator's expected outcome. A second sampling scenario is informative in order to underpin the importance of survey weights when model assumptions do not hold. For this scenario, we set the inclusion probability

$$\varepsilon_i^k = y_i^k - \mu_i^k \tag{4.5}$$

$$\tilde{\pi}_{d_1,i} = \begin{cases} 0.1, \text{ if } \varepsilon_{d_1,i}^k \leq \tau_{d_1,0.25} \\ 0.2, \text{ if } \tau_{d_1,0.25} < \varepsilon_{d_1,i}^k \leq \tau_{d_1,0.5} \\ 0.4, \text{ if } \varepsilon_{d_1,i}^k > \tau_{d_1,0.75} \end{cases} \tag{4.6}$$

$$\pi_{d_1,i} = \frac{\tilde{\pi}_{d_1,i}}{\sum_{i=1}^{N_{d_1}} \tilde{\pi}_{d_1,i}} \cdot \frac{25}{150}, \tag{4.7}$$

where $\tau_{d_1,p}$ is the p-quantile of the errors ε_i in domain d_1 in the k-th finite population. Consequently, the inclusion probability is a function of an unobserved model error, is purely at random and is not modeled in the standard regression models. As the regression concept seeks to minize the random error and observations with a highly positive error are oversampled, we expect the regression coefficients to be biased upwards.

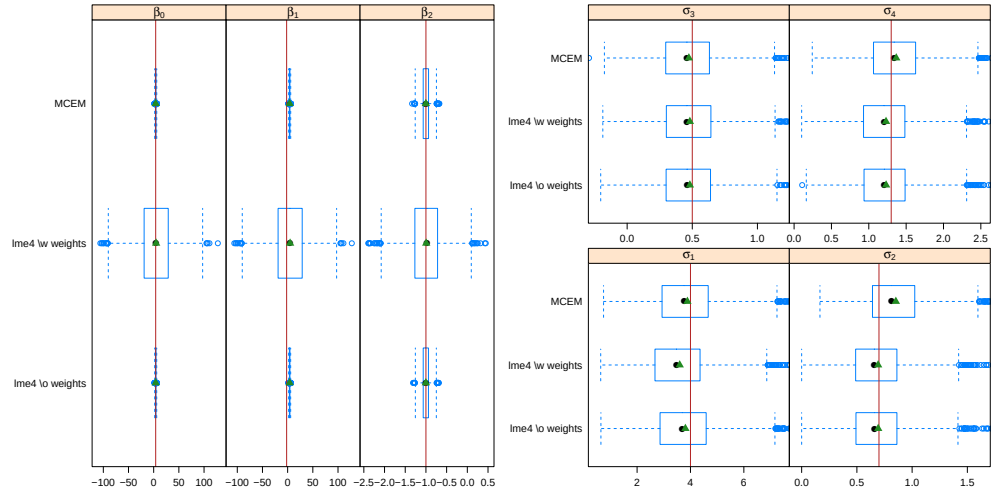
Our benchmark is the regression parameter estimation by the R-package `lme4` [6], which allows a very flexible random effects structure. `lme4` estimates GLMMs by Penalized Least Squares (PLS) which can be rewritten to be equivalent to the Maximum Likelihood estimate and the REML criterion under a normal repsonse. As we apply the EM algorithm, a fair comparison requires to use the ML estimation procedure of `lme4` rather than REML. When the random variable Y belongs to another exponential family distribution than the Gaussian, the criterion to be minimized is a Laplace approximation of the deviance. `lme4` does not allow to include survey weights, but the use of prior weights, which, however, enter *inversely* the estimation process. Therefore, as some sort of control for the survey design, we run the `lme4` estimation also with the survey weights solely and as interaction with the explanatories as regressors [15].

4.2 Simulation Results

Like [12], we find that the MCEM-algorithm moves relatively quickly close to the true parameters but then keeps oscillating, sometimes even removing from the ML neighbourhood. While the EM-algorithm is designed to increase the log-likelihood in each step, the MC random component leads to increased volatility such that this is not always the case. We reduced this volatility using antithetic random draws. Nevertheless, we keep track of the ML values and when the algorithm cuts off (either because a pre-defined number of oscillations or the pre-defined alllowed maximum number of detoriations of the ML or convergence is reached), the parameter vector that returned the best simulated ML is kept. With more fine-tuning of the estimation hyperparameters (number of MC-draws in the algorithm (2.3), convergence criterion of the BFGS algorithm, ...), we are confident that the jitter in our simulations results would would be even less.
The simulation results are summarized in the following boxplots, where the true vectors are $(\beta_0, \beta_1, \beta_3) = (4, -2, -1)$ and $(\sigma_1, \sigma_2, \sigma_3, \sigma_4) = (4, 0.7, 0.5, 1.3)$. Note that we handled the regression estimation in a very automated way in order to do 3000 simulation runs. When the survey weights were included as regressors, the standard `lme4` settings sometimes returned an error for the binary data (due to the necessity to rescale the variables $w_i \cdot x_{1,i}$ and $w_i \cdot x_{2,i}$) such that for these boxplots, only 1741 simulations are considered.
Figure (4.1) summarizes the results for the LMM case under the non-informative design. As the sample obeys exactly the data generating process (DGP), i.e. the superpopulation model, all parameters should in average equal the superpopulation parameters. The use of survey weights should only inflate the efficiency of the estimators. For all three estimation procedures, using either algorithm (2.3) or `lme4` with and without survey weights as regressors, this is true except for the estimates for σ_2^2 and σ_4^2 of the MCEM algorithm.
One possible explanation for this behaviour is the following: The first term of (2.11) is estimated using a Hàjek estimator. Though their focus is different from ours, [11] show in their simulation study that the Hàjek estimator tends in some cases to overestimate for random clusters. This leads to a greater weight on the GLM component in the importance sampling process. And the GLM component seeks to have a good prediction of the random effects whereas the second component, $\log \phi(\cdot|\boldsymbol{\sigma})$ counterweights that tendency. Therefore, it is plausible that the unplanned domain variance components are slightly overestimated.

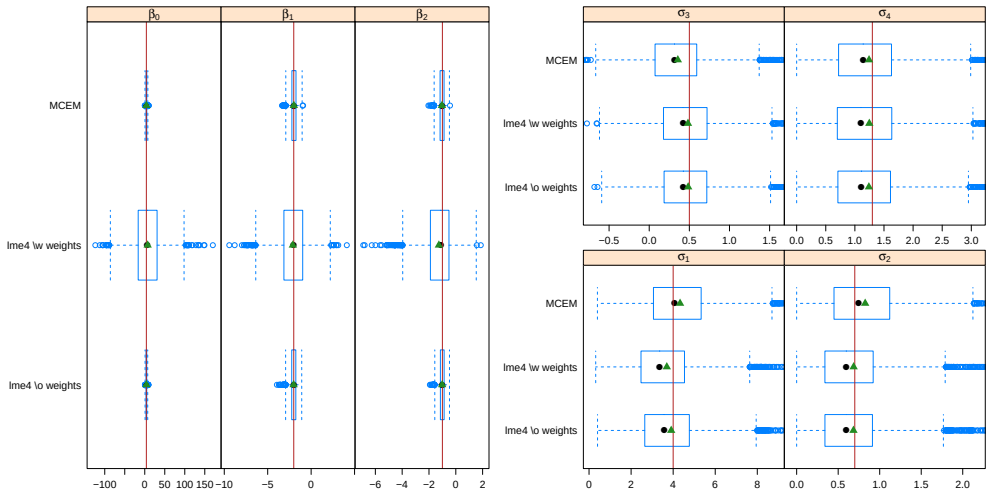
Further, we note that the width of the boxplots of our proposed algorithm are close to that of `lme4 o\weights` for both the fixed effects and the variance components.

Figure 4.1: LMM under Non-informative Design



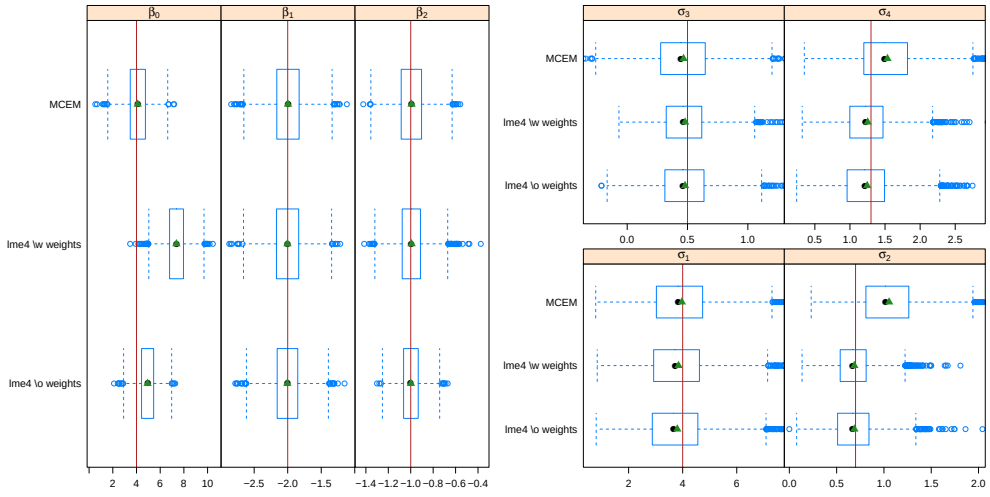
Next, we inspect the non-informative design under a binary response. The results are plotted in (4.2). Results are similar to the LMM setting. However, our method shows a little less stability concerning the covariance matrix of the unplanned domain. This seems plausible because the root of the score function has no closed form solution, which is known from the GLM literature, and may be remedied by increasing the size of the MC-sample. Like in the LMM setting, we find the variance components of the unplanned domains upward-biased. This makes us confident that our hypothesis holds.

Figure 4.2: GLMM under Non-informative Design



Concerning the fixed effects, note that the inclusion of survey-weights as covariates in the regression model leads to lower efficiency than our proposed MCEM-based approach.
Finally, we turn to the informative design that was described in equations (4.5) to (4.7). A summarizing plot is given in figure (4.3).

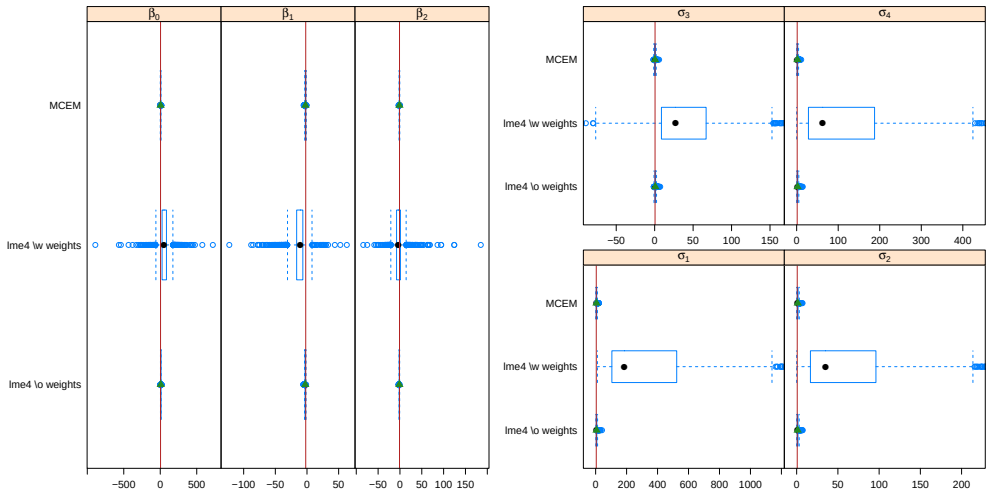
Figure 4.3: LMM under Informative Design



It can be seen that the LMM without survey weights is biased upwards for β_0 as it was expected from the informativeness set-up. However, surprisingly, the

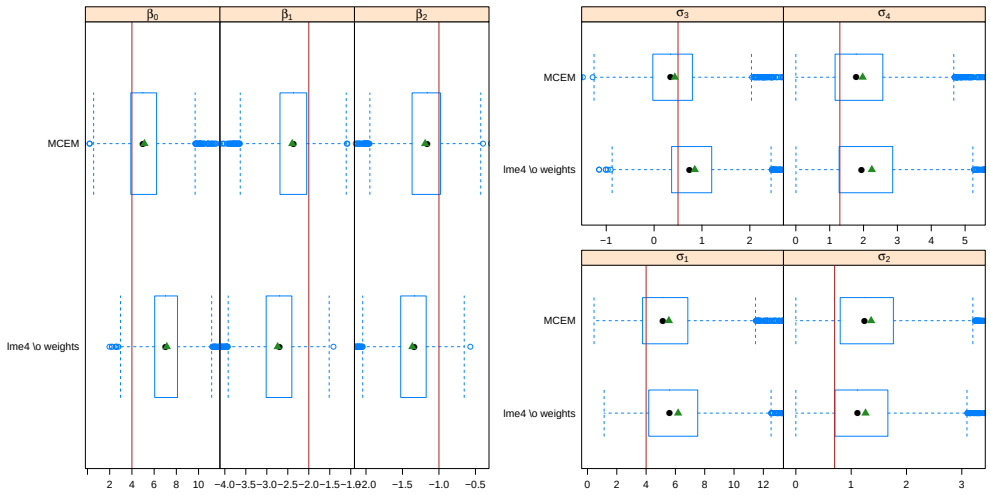
estimator of β_1 and β_2 seem to be unbiased. As x_2 is not associated with an unobserved random vector, whereas there are for both the intercept and x_1 random components, this may be due to less confusion of the unobserved components with the oversampled errors ε_i. Furthermore, it is notable that the inclusion of the survey weights as regressors does not work in the intended way, at least not when the aim is inference on the regression coefficients of the vector $(1, x_1, x_2)^T$. That could be to the (imperfect) collinearity between $x_{1,i}$ and $x_{1,i} \cdot w_i$, which attributes some of the influence of x_1 to the product. Notwithstanding, we find that the MC-average of fixed effects estimates of the proposed $MCEM$-algorithm is closer to the true superpopulation values than those estimates using `lme4`. On the other hand, we find the variance components of the unplanned domain slightly biased. There seem to be two reasons: First, the problem of the Hàjek estimator explained above. Second, the proposed procedure might attribute correctly the sampling design to an unobserved component but fails to distinguish between the error component ε and the random effects of d_2. This is still not perfect, but considering applications of multilevel modelling, such as small area estimation, a correctly estimated intercept is more important because solely the fixed effects determine the point estimates of unsampled domains.
Under the informative design and a *binary* response, the ignorance of the survey design leads to more severe problems as is illustrated in figure (4.4).

Figure 4.4: GLMM under Informative Design



Leaving the method `lme4 w\weights` that does obviously not work even in the fixed effects estimation we get figure (4.5)

Figure 4.5: GLMM under Informative Design - Zoom



Though the MC-average still deviates a little from the superpopulation parameters $\boldsymbol{\beta}$, we find that the boxes are by far more centered around the "true" values than in the GLMM regression without survey weights (`lme4 o\weights`). In addition, we can now even recognize some advantage of the use of survey weights concerning the variance components estimation. That the biases of the unplanned domain variance components go into the same direction like in the LMM setting makes us confident that the given explanations hold.

5 Discussion

In this paper, we proposed to use a well known approach of GLMM estimation, the Monte-Carlo EM-algorithm ([12], [2] and [24]) for the inclusion of survey weights. To the best of our knowledge, survey weighting in GLMMs has until now come from another direction ([16], [18], and [20]) that makes it difficult to include a single, unit-specific survey weight and crossed random effect patterns. Besides the theoretical approach, we have discussed the computational problems that come from a flexible covariance matrices and consequent high dimensionality of random variables in $\boldsymbol{\gamma}$ that must be simulated simultaneously. A final simulation study revealed that besides the numerical problems that come with a high-dimensional Monte-Carlo integration in the E-step, the fixed effects estimates are not harmed by survey weighting when the design is non-informative and do profit when the design is informative. When the domains, for which a random effect pattern is intended, we get the same evidence for the variance component estimation. However, further investigation is necessary in how to stabilize the variance component estimation for unplanned domains, especially, when the endogenous variable is binary.
We can think of several applications of the proposed estimation procedure. For example, small area estimation often uses a multilevel framework in order to increase the efficiency of mean estimates in small domains [1] . However, those estimates are only model-unbiased under the correct model and non-informative

sampling. One could use survey weights in order to protect the point estimates from the possible invalidity of the model assumptions. Another, yet to be studied appication, would be to use rather nonresponse propensities than survey weights in order to correct for a missing at random nonresponse pattern. However, though we managed to deal with a large size of random effects, there are also open questions concerning the handling of the stochastic volatiliy mentioned above.

References

[1] Battese, G. E., Harter, R. M., & Fuller, W. A. (1988). An error-components model for prediction of county crop areas using survey and satellite data. Journal of the American Statistical Association, 83(401), 28-36.

[2] Booth, J. G., & Hobert, J. P. (1999). Maximizing generalized linear mixed model likelihoods with an automated Monte Carlo EM algorithm. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 61(1), 265-285.

[3] Boyles, R. A. (1983). On the convergence of the EM algorithm. Journal of the Royal Statistical Society. Series B (Methodological), 47-50.

[4] Chambless, L. E., & Boyle, K. E. (1985). Maximum likelihood methods for complex sample data: logistic regression and discrete proportional hazards models. Communications in Statistics-Theory and Methods, 14(6), 1377-1392.

[5] Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. Journal of the royal statistical society. Series B (methodological), 1-38.

[6] Douglas, B., Mächler, M., Bolker, B. & Walker, S. (2015). Fitting Linear Mixed-Effects Models Using lme4. Journal of Statistical Software, 67(1), 1-48. DOI 10.18637/jss.v067.i01.

[7] Horvitz, D. G., & Thompson, D. J. (1952). A generalization of sampling without replacement from a finite universe. Journal of the American statistical Association, 47(260), 663-685.

[8] Laird, N. M., & Ware, J. H. (1982). Random-effects models for longitudinal data. Biometrics, 963-974.

[9] Louis, T. A. (1982). Finding the observed information matrix when using the EM algorithm. Journal of the Royal Statistical Society. Series B (Methodological), 226-233.

[10] Lehtonen, R., & Veijanen, A. (1998). Logistic generalized regression estimators. Survey Methodology, 24, 51-56.

[11] Makela, S., Si, Y. and Gelman, A. (2017). Bayesian Inference under Cluster Sampling with Probability Proportional to Size. arXiv preprint arXiv:1710.00959.

[12] McCulloch, C. E. (1997). Maximum likelihood algorithms for generalized linear mixed models. Journal of the American statistical Association, 92(437), 162-170.

[13] Nathan, G., & Holt, D. (1980). The effect of survey design on regression analysis. Journal of the Royal Statistical Society. Series B (Methodological), 377-386.

[14] Owen, A. B. (2013). Monte Carlo theory , methods and examples. URL `http://statweb.stanford.edu/~owen/mc/`

[15] Pfeffermann, D. (1993). The role of sampling weights when modeling survey data. International Statistical Review/Revue Internationale de Statistique, 317-337.

[16] Pfeffermann, D., Skinner, C. J., Holmes, D. J., Goldstein, H., & Rasbash, J. (1998). Weighting for unequal selection probabilities in multilevel models. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 60(1), 34-40.

[17] Powell, M. J. D. (1987). Updating conjugate directions by the BFGS formula. Mathematical Programming, 38(1), 29.

[18] Rabe–Hesketh, S., & Skrondal, A. (2006). Multilevel modelling of complex survey data. Journal of the Royal Statistical Society: Series A (Statistics in Society), 169(4), 805-827.

[19] Rao, J. N. K. (2003). Small Area Estimation. Wiley, New York.

[20] Rao, J. N. K., Verret, F., & Hidiroglou, M. A. (2014). A weighted composite likelihood approach to inference for two-level models from survey data.

[21] Wald, A. (1949). Note on the consistency of the maximum likelihood estimate. The Annals of Mathematical Statistics, 20(4), 595-601.

[22] Wedderburn, R. W. M. (1976). On the existence and uniqueness of the maximum likelihood estimates for certain generalized linear models. Biometrika, 63(1), 27-32.

[23] Wu, C. J. (1983). On the convergence properties of the EM algorithm. The Annals of statistics, 95-103.

[24] Zipunnikov, V. V., & Booth, J. G. (2006). Monte Carlo EM for generalized linear mixed models using randomized spherical radial integration.