AUSTRIAN JOURNAL OF STATISTICS
Volume 41 (2012), Number 4, 243–265

Robust Unit-Level Small Area Estimation: A Fast Algorithm for Large Datasets

Tobias Schoch

School of Business,
University of Applied Sciences Northwestern Switzerland

Abstract: Small area estimation is a topic of increasing importance in official statistics. Although the classical EBLUP method is useful for estimating the small area means efficiently under the normality assumptions, it can be highly influenced by the presence of outliers. Therefore, Sinha and Rao (2009; The Canadian Journal of Statistics) proposed robust estimators/predictors for a large class of unit- and area-level models. We confine attention to the basic unit-level model and discuss a related, but slightly different, robustification. In particular, we develop a fast algorithm that avoids inversion and multiplication of large matrices, and thus permits the user to apply the method to large datasets. In addition, we derive much simpler expressions of the bounded-influence predicting equations to robustly predict the small-area means than Sinha and Rao (2009) did.

Zusammenfassung: Der zunehmende Einsatz von Methoden der Small Area Estimation in der Amtlichen Statistik ist Ausdruck eines paragdigmatischen Wandels, der die Bedeutung modell-unterstützer Schätzmethoden unterstreicht. Mitunter beruhen die Methoden auf den strikten parametrischen Verteilungsannahmen Gemischt-Linearer Modelle und sind daher nicht robust bei Ausreisserkontamination. Sinha und Rao (2009; The Canadian Journal of Statistics) haben eine vielbeachtete Robustifizierung der unit- und area-level Modelle vorgeschlagen, die jedoch hinsichtlich numerischer Stabilität und Anwendbarkeit für die, in der Amtlichen Statistik üblichen Stichprobengrössen, ungeeignet ist. In diesem Artikel wird eine, zu Sinha–Rao's Methode äquivalente, robuste Methode entwickelt und ein Algorithmus dafür beschrieben. Die Performance der Methode wird in einer kleinen Simulation nachgewiesen.

Keywords: Small Area Estimation, Robustness, M-Estimation.

1 Introduction

Small are estimation (SAE) has become of great importance in official statistics due to the growing demand for reliable small-area statistics (e.g., estimates on the level of Bundesländer/Kanton or communities). Sample surveys provide a cost-effective way of obtaining estimates for characteristics of interest at both population and subpopulation (or domain) level. An estimator of a domain characteristic is called direct if it is based only on data from sample units in the domain. In most practical applications, however, domain sample sizes are not large enough, or even zero for unplanned domains, to allow direct

estimation. In this context, *small area estimation* refers to a subpopulation for which reliable statistics of interest cannot be produced due to certain limitations of the available data or because the estimates have extremely large sampling errors.

When direct estimation is not possible (or very unreliable), one has to rely upon alternative methods that depend on the availability of population-level auxiliary information (e.g., census or administrative data). The methods are commonly referred to as indirect or model-based estimation methods. Among the model-based SAE methods, the class of (generalized) mixed linear models (MLM) has received considerable attention because these models explain between-area variation in the target variable using auxiliary information and include area-specific random effects to account for between-area variation beyond that explained by the auxiliary information (see e.g., Jiang and Lahiri, 2006). In this context, the small-area means (and totals) can be expressed as linear combinations of fixed and random effects, which are obtained by (empirical) best linear unbiased prediction (BLUP; EBLUP) estimators. In general, model-based estimation procedures enlarge the effective area-specific sample size and yield a smaller mean square (prediction) error of the statistic under consideration compared to direct estimation methods (Rao, 2003, chap. 5). However, such models depend on parametric assumptions as well as requiring specification of the random part of the model.

Although the EBLUP method is useful for estimating the small-area means efficiently under normality assumptions, it can be highly influenced by the presence of outliers in the data or departures from the assumed normal distribution of the random effects. In the presence of contamination, the bias of non-robust methods (e.g., maximum likelihood estimators) can be arbitrarily large and renders these estimators extremely inefficient. Sinha and Rao (2009) therefore introduced the robust EBLUP method (REBLUP), based on M-estimators for MLMs (Richardson and Welsh, 1995; Welsh and Richardson, 1997), into the field of small area estimation.

Although M-estimators for MLMs are theoretically convincing, no reliable algorithms have been available so far (Chambers and Tzavidis, 2006). Sinha and Rao (2009) proposed to obtain robust parameter estimates by a Newton–Raphson type (NR) numerical optimization method—but they did not indicate how to initialize the method. Moreover, the NR method is well-known – e.g., from robust regression analysis (Maronna, Martin, and Yohai, 2006, chap. 9.1; Huber, 1981, chapters 6.7 and 7.8) – to be rather unreliable. Notably, Richardson (1995) reports significant convergence problems of NR and similar numerical optimization methods (i.e., Anderson's method (T. W. Anderson, 1973; Miller, 1977) and the EM algorithm) in the context of M-estimators for MLMs. Moreover, Chaubey and Venkateswarlu (2002) report convergence failure in computing robust ML estimators in 5.8 %–25.8 % of all Monte Carlo trials in their simulation study.

Computational problems arise, though, already in the exercise of computing ML estimates for unbalanced data when no contamination is present at all. Searle, Casella, and McCulloch (1992) phrase it as follows: "[t]here are myriad difficulties involved in actually implementing these methods [i.e., ML and REML estimators in mixed linear models; ts] including, but not limited to, stability of numerical methods applied to the matrices involved, methods of avoiding the inversion of large matrices and the details of diagnosing convergence [...]." (Searle et al., 1992, p.312). The exercise of robustly fitting MLMs introduces further difficulties, such as (among others) the choice of robust starting val-

ues. The lack of suitable software has led to extensive efforts to study M-quantile-type methods in the field of SAE; see e.g., Chambers and Tzavidis (2006).

We focus on the basic unit-level SAE model (see Rao, 2003, chap. 7.2). The main contribution of this article is a numerically stable and fast algorithm (even for very large datasets) in order to compute (robust) M-estimates of the model parameters. In addition, we develop a method for robustly predicting the small-area means, which is much simpler and therefore considerably faster than the proposal of Sinha and Rao (2009).

This article is organized as follows. Section 2 introduces the model of interest, the ML estimator, and the EBLUP method. In Section 3, we discuss the robustification of the EBLUP and derive a numerically stable and fast algorithm. Subsequently, we derive robust predicting equations, based on the proposal of Copt and Victoria-Feser (2009), in order to robustly predict the small-area means. A bootstrap method for estimating the mean square prediction error (MSPE) of the estimated/predicted small-area means – paralleling the ideas of Sinha and Rao (2009) – is discussed in Section 4. In Section 5, we present the results of a simulation study. Finally, Section 6 draws together the main findings.

2 Preliminaries, Definitions and the Model

Let us first introduce some terminology and notation. Consider a finite survey population U whose units are labeled $1, \ldots, N$. The population U is assumed to be partitioned into g subsets $U_1, \ldots, U_g$, called small areas. Let N_i be the size of U_i and assume that $U = \bigcup_{i=1}^{g} U_i$, then we have $N = \sum_{i=1}^{g} N_i$. We are interested in estimating the domain-specific means of the variable of interest, y, from the data of the domains U_i, $i = 1, \ldots, g$. Let $\bar{Y}_i$ be the population mean of y based on N_i units in area i ($i = 1, \ldots, g$).

Suppose that $\bar{Y}_i$ is related to the p-vector $\mathbf{x}_{ij} = (x_{ij1}, \ldots, x_{ijp})^T$ of auxiliary data on the unit level. In what follows, we assume that this relationship is given at the population level by the so-called basic unit-level model (Rao, 2003, chap. 7.2), through a nested-error linear regression (Battese, Harter, and Fuller, 1988) of the form

$$y_{ij} = \mathbf{x}_{ij}^T \boldsymbol{\beta} + u_i + e_{ij}\,, \qquad j = 1, \ldots, N_i\,; \quad i = 1, \ldots, g\,, \tag{1}$$

where the area-specific random effects u_i are assumed to be independent $\mathcal{N}(0, \sigma_a^2)$, and independent of the unit-level errors e_{ij} which are assumed to be independent $\mathcal{N}(0, \sigma_e^2)$. The variance components σ_e^2 and σ_a^2 are stacked in $\boldsymbol{\theta} = (\sigma_e^2, \sigma_a^2)^T$.

A sample of $n_i \geq 1$ of units is assumed to be drawn (e.g., in a survey) from each area, $i = 1, \ldots, g$. The sampling is assumed to be ignorable such that the population model (1) also holds in the sample. The latter assumption is satisfied under simple random sampling from each area or more generally for sampling designs that use the auxiliary information $\mathbf{x}_{ij}$ in the selection of the samples; see Rao (2003, pp. 78–80) for more details. The following definition specifies the model based on sample data.

Definition 1. *Marginally, the basic unit-level model is defined as*

$$\mathbf{y}_i \sim \mathcal{N}(\mathbf{X}_i \boldsymbol{\beta}, \boldsymbol{\Omega}_i(\boldsymbol{\theta}))\,, \qquad i = 1, \ldots, g\,, \tag{2}$$

with $\boldsymbol{\theta} = (\sigma_e^2, \sigma_a^2)^T$ *and* $\boldsymbol{\Omega}_i(\boldsymbol{\theta}) = \sigma_e^2 \mathbf{I}_i + \sigma_a^2 \mathbf{1}_i \mathbf{1}_i^T$*, where* $\mathbf{I}_i$ *is the* $(n_i \times n_i)$ *identity matrix,* $\mathbf{1}_i$ *the* n_i*-vector of ones, and* $\mathbf{y}_i$ *a* n_i*-vector.*

Whenever no confusion can arise, we suppress the functional dependence of $\boldsymbol{\Omega}_i(\boldsymbol{\theta})$ on $\boldsymbol{\theta}$. Note that the area-specific response vectors $\mathbf{y}_i$ $(i = 1, \ldots, g)$ can be of different length (i.e., un-balanced data). For notational simplicity, it is sometimes useful to work with the stacked vectors/matrices: $\mathbf{y} = (\mathbf{y}_1^T, \ldots, \mathbf{y}_g^T)^T$, $\mathbf{X} = (\mathbf{X}_1^T, \ldots, \mathbf{X}_g^T)^T$, and $\boldsymbol{\Omega} = \text{diag}(\boldsymbol{\Omega}_1, \ldots, \boldsymbol{\Omega}_g)$. In addition, we impose the following assumptions (restrictions) on the definition of model (2).

Assumption 1. *The parameter space of model (2) is* $\boldsymbol{\Theta} = \boldsymbol{\Theta}_{\boldsymbol{\beta}} \times \boldsymbol{\Theta}_{\boldsymbol{\theta}}$ *with* $\boldsymbol{\Theta}_{\boldsymbol{\beta}} = \{\boldsymbol{\beta} \in \mathbb{R}^p\}$ *and* $\boldsymbol{\Theta}_{\boldsymbol{\theta}} = \{\boldsymbol{\theta} \in \mathbb{R}_+^2 | \sigma_e^2 > 0, \sigma_a^2 \geq 0\}$.

Note that we allow for the possibility that the random-effect variance, σ_a^2, can be zero.

Assumption 2. *The first column of the* $(n_i \times p)$ *matrices* $\mathbf{X}_i$ *consists of* n_i *ones,* $\forall i = 1, \ldots, g$.

As a consequence, the first element of $\boldsymbol{\beta}$ refers to the (regression) intercept.

Assumption 3. *The* $(n_i \times p)$ *matrices* $\mathbf{X}_i$ *have full rank* p*,* $\forall i = 1, \ldots, g$.

This assumption is imposed to simplify the discussion of the proposed algorithm. All results in this paper remain valid if Assumption 3 does not hold; however, the computational details are more involved.

The estimators considered in this article have to be computed by a numerical method in an iterative manner. On the sth iteration ($s = 1, 2, \ldots$) of an algorithm for producing an estimate of, say, $\boldsymbol{\beta}$, the current value for the estimate is converted into a new value. By way of example, denote by $\{\boldsymbol{\beta}\}^{(s)}$ the parameter estimates of $\boldsymbol{\beta}$ on the sth iteration, and, for any quantity f which is a function of $\boldsymbol{\beta}$, we use $\{f\}^{(s)}$ to represent the value of f at $\{\boldsymbol{\beta}\}^{(s)}$.

2.1 EBLUP Method

Suppose for the time being that $\boldsymbol{\beta}$ and u_i $(i = 1, \ldots, g)$ are known. For N_i large and the sampling fraction $f_i = n_i/N_i$ small (for all i), it follows that the area-specific means $\bar{Y}_i$ can be estimated/predicted by

$$\bar{Y}_i \approx \mu_i = \bar{\mathbf{x}}_{i\bullet}^T \boldsymbol{\beta} + u_i \,, \qquad i = 1, \ldots, g \,, \tag{3}$$

where $\bar{\mathbf{x}}_{i\bullet}$ is the p-vector of known population means for area i. Note that $\bar{\mathbf{x}}_{i\bullet}$ is the mean of $\mathbf{x}_{ij}$ based on all $j = 1, \ldots, N_i$ units of area i in population U. In the case of non-negligible sampling fractions (i.e., if $N_i \gg n_i$ does not hold), we cannot take the small area mean $\bar{Y}_i$ as $\bar{\mathbf{x}}_{i\bullet}^T \boldsymbol{\beta} + u_i$. However, we can write $\bar{Y}_i$ as

$$\bar{Y}_i = f_i \bar{y}_i + (1 - f_i) \bar{y}_i^{ns} \,, \tag{4}$$

where $\bar{y}_i$ is the sample mean and $\bar{y}_i^{ns}$ is the mean of the non-sampled values of y in area i. Under the population model, we replace the unobserved y_{ij}^{ns} by its estimator $(\mathbf{x}_{ij}^{ns})^T \boldsymbol{\beta} + u_i$,

where $\mathbf{x}_{ij}^{ns}$ is the p-vector of auxiliary data associated with y_{ij}^{ns}, and compute the means $\bar{y}_i^{ns}$ (Rao, 2003, p. 142). Sinha and Rao (2009) indicate that this approximation may not be adequate in the presence of representative outliers (Chambers, 1986), although adequate under the Gaussian mixed linear model.

Now, since the fixed effects, $\boldsymbol{\beta}$, and the variance components, $\boldsymbol{\theta}$, (and the realizations of the random effect u_i) are unknown in a real-word application, the predicting equation, (3), is of limited value. For known $\boldsymbol{\theta}$, however, the best linear unbiased estimator (BLUE), $\tilde{\boldsymbol{\beta}}$, is given by

$$\tilde{\boldsymbol{\beta}}(\boldsymbol{\theta}) = \left(\sum_{i=1}^{g} \mathbf{X}_i^T \boldsymbol{\Omega}_i^{-1}(\boldsymbol{\theta}) \mathbf{X}_i \right)^{-1} \left(\sum_{i=1}^{g} \mathbf{X}_i^T \boldsymbol{\Omega}_i^{-1}(\boldsymbol{\theta}) \mathbf{y}_i \right). \tag{5}$$

Appealing to well-known results of BLUE estimation, we obtain the best linear unbiased prediction (BLUP) estimator (Rao, 2003, pp. 96–98)

$$\tilde{\mu}_i(\boldsymbol{\theta}) = \bar{\mathbf{x}}_{i\bullet}^T \tilde{\boldsymbol{\beta}}(\boldsymbol{\theta}) + \sigma_a^2 \mathbf{1}_i^T \boldsymbol{\Omega}_i^{-1}(\boldsymbol{\theta}) \big(\mathbf{y}_i - \mathbf{X}_i \tilde{\boldsymbol{\beta}}(\boldsymbol{\theta}) \big), \qquad i = 1, \ldots, g. \tag{6}$$

In practice $\boldsymbol{\theta}$ has to be estimated as well. This can been accomplished by several methods, each of which has advantages and more or less severe disadvantages (Harville, 1977; Miller, 1977); see below for a discussion of the ML estimates. Once we have computed $\hat{\boldsymbol{\theta}}$, the empirical BLUP (EBLUP), $\hat{\mu}_i(\hat{\boldsymbol{\theta}})$, is obtained replacing $\boldsymbol{\theta}$ in (6) by $\hat{\boldsymbol{\theta}}$.

While EBLUP is fairly easy to obtain, estimation of a reasonable measure of uncertainty for the area-level predicted means is a challenging problem. In the context of finite population sampling, a variance estimate of a direct domain estimator of the mean (e.g., domain-specific Hajek estimator) is readily obtained, appealing to well-known results of randomization inference and the fact that the estimator is design unbiased (see e.g., Särndal, Swensson, and Wretman, 1992, chap. 10.3). These result do, however, not carry over to (E)BLUP estimation and we have to resort to mean square (prediction) error estimation. In their seminal paper, Prasad and Rao (1990) studied a second-order approximation to the mean square prediction error (MSPE) of the EBLUP. Datta and Lahiri (2000) extended the Prasad–Rao setting to a wider range of variance estimators, including the ML estimator. We focus on the promising (also with regard to robustness properties) parametric bootstrap discussed in Lahiri (2003); see Section 3.4 for further details.

2.2 Maximum Likelihood Estimator

To start with, it will prove useful to study the maximum likelihood (ML) estimator of the model parameters $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$. Subsequently, we shall derive a robustification of the Fisher-score functions. For model (2), the non-constant part of the log-likelihood, $l(\boldsymbol{\tau}, \mathbf{X}, \mathbf{y})$, is given by

$$-2l(\boldsymbol{\tau}, \mathbf{X}, \mathbf{y}) = \sum_{i=1}^{g} \log |\boldsymbol{\Omega}_i| + \sum_{i=1}^{g} (\mathbf{y}_i - \mathbf{X}_i \boldsymbol{\beta})^T \boldsymbol{\Omega}_i^{-1} (\mathbf{y}_i - \mathbf{X}_i \boldsymbol{\beta}), \tag{7}$$

where $\boldsymbol{\tau} = (\boldsymbol{\beta}^T, \boldsymbol{\theta}^T)^T$. Whenever no contamination is supposed to be present, estimates of the parameter vector $\boldsymbol{\tau}$ shall be obtained by means of ML. The ML estimator $\hat{\boldsymbol{\tau}}$ is

defined as $l(\hat{\boldsymbol{\tau}}, \mathbf{X}, \mathbf{y}) = \sup_{\boldsymbol{\tau}\in\boldsymbol{\Theta}} l(\boldsymbol{\tau}, \mathbf{X}, \mathbf{y})$, provided $\boldsymbol{\tau}$ is an interior point of $\boldsymbol{\Theta}$. It will prove useful to express the covariance matrix $\boldsymbol{\Omega}_i$ $(i = 1, \ldots, g)$ as follows

$$\boldsymbol{\Omega}_i = \sigma_e^2 \mathbf{I}_i + \sigma_a^2 \mathbf{J}_i = v(\mathbf{I}_i + d\mathbf{J}_i) = v\mathbf{V}_i\,, \tag{8}$$

where $v = \sigma_e^2$, say, and $d = \sigma_a^2/\sigma_e^2 \equiv a/v$ (the notation has been chosen for ease of simplicity—notably, it spares us from writing squared terms); cf. Hartley and Rao (1967) for the parametrization of the covariance matrix in terms of variance components ratios. The primary advantage of the Hartley–Rao parametrization is that we obtain a separate equation for v. Thus, on rewriting the log-likelihood, we obtain

$$-2l(\boldsymbol{\tau}, \mathbf{X}, \mathbf{y}) = \sum_{i=1}^{g} \log|\mathbf{V}_i| + \sum_{i=1}^{n} n_i \log v + \frac{1}{v}\sum_{i=1}^{g} (\mathbf{y}_i - \mathbf{X}_i\boldsymbol{\beta})^T \mathbf{V}_i^{-1}(\mathbf{y}_i - \mathbf{X}_i\boldsymbol{\beta})\,. \tag{9}$$

Provided the maximum is not attained on the boundary, the maximum likelihood estimates $\hat{\boldsymbol{\beta}}$, $\hat{v}$, and $\hat{d}$ are a solution to the system of Fisher-score equations, respectively,

$$-2(1/v)\sum_{i=1}^{g} \mathbf{X}_i^T \mathbf{V}_i^{-1}(\mathbf{y}_i - \mathbf{X}_i\boldsymbol{\beta}) = \mathbf{0}\,, \tag{10}$$

$$\sum_{i=1}^{g} \frac{n_i}{v} - (1/v^2)\sum_{i=1}^{g} (\mathbf{y}_i - \mathbf{X}_i\boldsymbol{\beta})^T \mathbf{V}_i^{-1}(\mathbf{y}_i - \mathbf{X}_i\boldsymbol{\beta}) = 0\,, \tag{11}$$

$$\sum_{i=1}^{g} \mathbf{1}_i^T \mathbf{V}_i^{-1}\mathbf{1}_i - (1/v)\mathbf{1}_i^T \mathbf{V}_i^{-1}(\mathbf{y}_i - \mathbf{X}_i\boldsymbol{\beta})(\mathbf{y}_i - \mathbf{X}_i\boldsymbol{\beta})^T \mathbf{V}_i^{-1}\mathbf{1}_i = 0\,. \tag{12}$$

It is easy to see that the MLE of v is given by

$$\hat{v} = \tfrac{1}{n}\sum_{i=1}^{g} (\mathbf{y}_i - \mathbf{X}_i\boldsymbol{\beta})^T \mathbf{V}_i^{-1}(\mathbf{y}_i - \mathbf{X}_i\boldsymbol{\beta})\,, \tag{13}$$

where $n = \sum_{i=1}^{g} n_i$. Lindstrom and Bates (1988) (among others) take advantage of (13) and propose the variance-profile log-likelihood substituting (13) back into (9). This leads to an equivalent maximization problem with v eliminated. That is, the variance-profile log-likelihood function, $l_p(\boldsymbol{\tau}^*, \mathbf{X}, \mathbf{y})$, with $\boldsymbol{\tau}^* = (\boldsymbol{\beta}^T, d)^T$ is an economical dimension-reduced parametrization. At first sight, the effect of parametrizing the variance components in terms of ratios and the implied simplification of the maximization problem seem to be rather limited, since only one parameter can be eliminated. From the perspective of computation, however, even such a small reduction simplifies the numerical optimization problem considerably. This parametrization brings along another good property in terms of numerical optimization: observe from (13) that estimates of v are non-negative.

Equations (10) and (12), on the other hand, do not feature a closed-form expression (for unbalanced data) and have to be solved by means of some (iterative) numerical optimization methods.

A criticism of ML estimators of variance is that they are biased downward because they do not take into account the loss in degrees of freedom from the estimation of $\boldsymbol{\beta}$. The REML estimators correct for this deficiency; see e.g., Harville, 1977 for the details. Nonetheless, we focus on ML and robust M-estimators; see Richardson and Welsh (1995) for robust REML estimation.

3 Robust EBLUP

Although the classical EBLUP method is useful for estimating the small area means efficiently under normality assumptions, it can be highly influenced by the presence of outliers in the data. Furthermore, mixed linear models have, unlike location-scale or regression models, no nice invariance structure. Notably, this means that the parameters cannot be estimated consistently in the presence of contamination—there is an unavoidable asymptotic bias. In the presence of contamination, any method estimates the parameter at the core model plus an unknown bias. In the case of ML estimators, the bias can be arbitrarily large and renders these estimators extremely inefficient (Welsh and Richardson, 1997).

A large number of authors proposed methods for robust analysis in mixed level models, ranging from rather algorithmic approaches (Rocke, 1983, 1991; Stahel and Welsh, 1997) over robustification of Henderson's mixed-model equations (Fellner, 1986) to replacing the Fisher scores by robust Fréchet-differentiable statistical functionals (Bednarski and Zontek, 1996). Copt and Victoria-Feser (2006) have proposed an S-estimator and provide software for balanced data (cf. supporting website of Heritier, Cantoni, Copt, and Victoria-Feser, 2009). The M-estimator-type methods, based on either a robustified likelihood (RML 1; Richardson and Welsh, 1995; Stahel and Welsh, 1997) or bounded-influence estimating equations (RML 2; Richardson and Welsh, 1995; Welsh and Richardson, 1997), have received considerable attention in the literature. Notably, we focus on the RML 2 method (Richardson and Welsh, 1995) because it embodies a natural way of restricting the influence of outliers in the response variable and is very closely related to the ML approach. Further, the RML 2 method is equivalent to the proposal of Sinha and Rao (2009). For these estimators, the potential bias is bounded, the efficiency is reasonable if the model holds, and the estimators are much more efficient than e.g., ML estimators, if it does not (Welsh and Richardson, 1997).

The following assumptions are crucial to all robust estimators.

Assumption 4. *Outliers occur only in the response variable. No attempt is made to limit the effect of outliers in the model/design space of the model (i.e., influential/leverage observations).*

In order to limit the influence of outliers in both the response variable ($\mathbf{y}$) and the design matrix ($\mathbf{X}$), one has to resort to generalized regression M-estimators (GM) in the context of linear models (e.g., Mallows- or Schweppe-type estimators). Richardson (1997) extended the notion of GM-estimators to include MLMs. Although theoretically convincing, GM-estimators for the MLMs lack numerical stability (Richardson, 1995, chap. 6.5).

3.1 Robust M-Estimator EBLUP

In the presence of contamination, the ML estimates can be severely biased. It is therefore reasonable to replace the system of Fisher-score functions (10-12) by estimating equations (EE) whose influence functions are bounded—i.e., so-called bounded-influence estimating equations (BIEE).

3.1.1 BIEE for $\boldsymbol{\beta}$

For the fixed effects, $\boldsymbol{\beta}$, (10) shall be replaced by the BIEE

$$\sum_{i=1}^{g}(1/\sqrt{v})\mathbf{X}_i^T\mathbf{V}_i^{-1/2}\boldsymbol{\psi}_k\big[(1/\sqrt{v})\mathbf{V}_i^{-1/2}(\mathbf{y}_i-\mathbf{X}_i\hat{\boldsymbol{\beta}}^R)\big]=\mathbf{0}\,, \tag{14}$$

where $\boldsymbol{\psi}_k(\mathbf{z}_i)=[\psi_k(z_{i1}),\ldots,\psi_k(z_{in_i})]^T$ with ψ_k denoting a bounded, odd function indexed by some robustness tuning constant k. Without loss of generality we will assume that ψ_k denotes the Huber ψ-function in what follows (or any other bounded, monotone function). Equivariance considerations indicate that it is useful to studentized the estimator by an appropriate scale, paralleling the concept of regression M-estimators. Note from (2) that at the Gaussian core model, the (marginal) law $\mathbf{y}_i\sim\mathcal{N}(\mathbf{X}_i\boldsymbol{\beta},v\mathbf{V}_i)$ holds. Hence, the n_i-vector of residuals in (14) is scaled by $(1/\sqrt{v})\mathbf{V}_i^{-1/2}=:\mathbf{B}_i$, say, since $\mathbf{B}_i^T\mathbf{B}_i=v\mathbf{V}_i$ and $\mathbf{B}_i(\mathbf{y}_i-\mathbf{X}_i\boldsymbol{\beta})\sim\mathcal{N}(0,1)$. The specific definition of $\mathbf{V}_i^{-1/2}$ will be discussed later; for the time being, it is sufficient to assume that the square root of $\mathbf{V}_i^{-1}$ exists.

The Solution of (14) shall be obtained by an iteratively re-weighted least square (IRWLS) algorithm which is the workhorse for computing M-estimates of regression (Maronna et al., 2006, pp. 104–105). Notably, the IRWLS approach is numerically much more stable than the Newton-Raphson approach (Schoch, 2011a). Denote by $\{\boldsymbol{\beta}\}^{(s)}$ the estimate of $\boldsymbol{\beta}$ produced by the algorithm on the sth iteration ($s=1,2,\ldots$). An updated estimate is obtained from

$$\begin{aligned}\{\boldsymbol{\beta}\}^{(s+1)}=&\left(\sum_{i=1}^{g}\big(\{\mathbf{W}_i\}^{(s)}\{\mathbf{V}_i^{-1/2}\}^{(s)}\mathbf{X}_i\big)^T\{\mathbf{W}_i\}^{(s)}\{\mathbf{V}_i^{-1/2}\}^{(s)}\mathbf{X}_i\right)^{-1}\\&\times\left(\sum_{i=1}^{g}\big(\{\mathbf{W}_i\}^{(s)}\{\mathbf{V}_i^{-1/2}\}^{(s)}\mathbf{X}_i\big)^T\{\mathbf{W}_i\}^{(s)}\{\mathbf{V}_i^{-1/2}\}^{(s)}\mathbf{y}_i\right),\end{aligned} \tag{15}$$

where $\mathbf{W}_i=\mathrm{diag}(\mathbf{w}_i)$, with $\mathbf{w}_i=(w_{i1},\ldots,w_{in_i})^T$, $w_{ij}=[\psi_k(r_{ij})/r_{ij}]^{1/2}$, and $\mathbf{r}_i=(1/\sqrt{v})\mathbf{V}_i^{-1/2}(\mathbf{y}_i-\mathbf{X}_i\boldsymbol{\beta})$. Put $\tilde{\mathbf{X}}_i=(1/\sqrt{v})\mathbf{W}_i\mathbf{V}_i^{-1/2}\mathbf{X}_i$ and $\tilde{\mathbf{y}}_i=(1/\sqrt{v})\mathbf{W}_i\mathbf{V}_i^{-1/2}\mathbf{y}_i$, then we may write

$$\{\boldsymbol{\beta}\}^{(s+1)}=\left(\sum_{i=1}^{g}\{\tilde{\mathbf{X}}_i^T\}^{(s)}\{\tilde{\mathbf{X}}_i\}^{(s)}\right)^{-1}\left(\sum_{i=1}^{g}\{\tilde{\mathbf{X}}_i^T\}^{(s)}\{\tilde{\mathbf{y}}_i\}^{(s)}\right). \tag{16}$$

Now, since (16) is a standard least squares problem, we obtain (iteratively) updated estimates of $\boldsymbol{\beta}$ by standard regression technique. First we note that by Assumption 3, $\tilde{\mathbf{X}}_i$ has full rank given that $\mathbf{w}_i$ is not the null vector. Put $\tilde{\mathbf{X}}=(\tilde{\mathbf{X}}_1^T,\ldots,\tilde{\mathbf{X}}_g^T)^T$ and $\tilde{\mathbf{y}}=(\tilde{\mathbf{y}}_1^T,\ldots,\tilde{\mathbf{y}}_g^T)^T$, which are of size $(n\times p)$ and $(n\times 1)$, respectively, where $n=\sum_{i=1}^{g}n_i$. Hence, we shall decompose $\tilde{\mathbf{X}}$ by means of the "skinny" QR-factorization (see e.g., Gentle, 2007, pp. 188–189 and p. 226). Write $\tilde{\mathbf{X}}=\mathbf{QR}$, where $\mathbf{R}=(\mathbf{R}_1^T,\mathbf{0}^T)^T$, with $\mathbf{R}_1$ an $(p\times p)$ upper triangular matrix. $\mathbf{Q}$ is an $(n\times n)$ orthogonal matrix which can be partitioned likewise: $\mathbf{Q}=(\mathbf{Q}_1,\mathbf{Q}_2)$, where $\mathbf{Q}_1$ is an $(n\times p)$ matrix whose columns are orthonormal. This enables us to write $\tilde{\mathbf{X}}=\mathbf{Q}_1\mathbf{R}_1$. Consequently, the overdetermined

linear system $\tilde{\mathbf{X}}\boldsymbol{\beta} = \tilde{\mathbf{y}}$ can be expressed as $\mathbf{R}_1\boldsymbol{\beta} = \mathbf{Q}_1^T\tilde{\mathbf{y}}$. Since $\mathbf{R}_1$ is an $(p \times p)$ triangular matrix, the system is easy to solve: $\boldsymbol{\beta} = \mathbf{R}_1^{-1}\mathbf{Q}_1^T\tilde{\mathbf{y}}$. The IRWLS algorithm now consists of solving

$$\{\boldsymbol{\beta}\}^{(s+1)} = \{\mathbf{R}_1^{-1}\}^{(s)}\{\mathbf{Q}_1^T\}^{(s)}\{\tilde{\mathbf{y}}\}^{(s)} \tag{17}$$

in an iterative manner. The final value is regarded as the estimate $\hat{\boldsymbol{\beta}}^R$.

3.1.2 BIEE for v

A bounded-influence EE for v that replaces the non-robust Fisher score (11) is obtained – in the spirit of Huber's proposal 2 (Huber, 1964) – as the solution $\hat{v}^R$ to the bounded-influence estimating equation

$$\left[\sum_{i=1}^{g} n_i\right]^{-1} \frac{1}{\delta_k} \sum_{i=1}^{g} \boldsymbol{\psi}_k\left(\frac{\mathbf{V}_i^{-1/2}\mathbf{r}_i}{\sqrt{\hat{v}^R}}\right)^T \boldsymbol{\psi}_k\left(\frac{\mathbf{V}_i^{-1/2}\mathbf{r}_i}{\sqrt{\hat{v}^R}}\right) = 1\,, \tag{18}$$

where $\boldsymbol{\psi}_k(\mathbf{z}_i) = (\psi_k(z_{i1}), \ldots, \psi_k(z_{in_i}))^T$ and $\delta_k = \mathrm{E}[\psi_k(u)^2]$ is a consistency correction term that ensures consistency of the estimate at the Gaussian core model with $u \sim \mathcal{N}(0,1)$ (where expectation is w.r.t. the model). From the perspective of computation, it is worth to consider another representation of the BIEE. The solution of (18) can be expressed as a weighted estimator, paralleling the concept of computing M-estimates of scale (cf. Maronna et al., 2006, pp. 40–41). Define a weight function

$$w_k(z) = \begin{cases} \psi_k(z)^2/z^2 & \text{if } z \neq 0\,, \\ \psi_k'(z) & \text{if } z = 0\,, \end{cases}$$

where ψ_k' denotes the first derivative and put $\mathbf{W}_i = \mathrm{diag}(w_k(u_{i1}), \ldots, w_k(u_{in_i}))$, with $\mathbf{u}_i = (1/\sqrt{v})\mathbf{V}_i^{-1/2}\mathbf{r}_i$. An updated estimate, $\{v\}^{s+1}$, is given by

$$\{v\}^{s+1} = \frac{1}{n\delta_k} \sum_{i=1}^{g} \{\mathbf{W}_i\}^s\{\mathbf{r}_i^T\}^s\{\mathbf{V}_i^{-1}\}^s\{\mathbf{r}_i\}^s\,, \tag{19}$$

where $n = \sum_{i=1}^{g} n_i$. The fact that all elements of the diagonal matrix $\mathbf{W}_i$ are non-negative, implies that the quadratic form in (19) and therefore the estimates of v are non-negative as well. This is in sharp contrast compared to the estimates obtained by (the inherently unconstrained) NR approach of Sinha and Rao (2009).

3.1.3 BIEE for d

For the estimator of d, we also have to replace the non-robust Fisher-score function (12) by a bounded-influence estimating equation. In contrast to v, the BIEE of d has no closed-form solution. If we put $\mathbf{u}_i(d) = (1/\sqrt{v})\mathbf{V}_i(d)^{-1/2}[\mathbf{y}_i - \mathbf{X}_i\boldsymbol{\beta}]$, then a robust estimate of d, say $\hat{d}^R$, is obtained as the solution to

$$\sum_{i=1}^{g} \mathbf{1}_i^T\mathbf{V}_i(\hat{d}^R)^{-1}\mathbf{1}_i - \mathbf{1}_i^T\mathbf{V}_i(\hat{d}^R)^{-1/2}\boldsymbol{\psi}_k\big[\mathbf{u}_i(\hat{d}^R)\big]\boldsymbol{\psi}_k\big[\mathbf{u}_i(\hat{d}^R)\big]^T\mathbf{V}_i(\hat{d}^R)^{-1/2}\mathbf{1}_i = 0\,. \tag{20}$$

Note that – for ease of simplicity – we highlighted only the functional dependence on d, i.e., $\mathbf{V}_i(d)^{-1/2}$, instead of reporting all parameters, $\mathbf{V}_i(\boldsymbol{\beta}, v, d)^{-1/2}$.

Among all available methods for finding the root of (20) (i.e., a root of a real-valued, continuous function in d), bisection is arguably the most reliable approach, but quite slow. The method of regula falsi has been found to converge at a faster rate than linear. However, it can go quite wrong in the case the function is not approximately linear over the interval (Small and Wang, 2003, pp. 43–45).[1] The Newton-Raphson method is well-known to converge with a quadratic convergence rate within some neighborhood of the root, but has the severe drawback of being very unreliable. In particular, the neighborhood of the root can be very small and (still more important) is not known beforehand. Divergence of the NR algorithm is a severe drawback and happens more frequently than many of the references admit (see also Jiang, Luan, and Wang, 2007). Chaubey and Venkateswarlu (2002), for instance, report convergence failure in 25.8 % of their Monte Carlo trials. Moreover, NR does not explicitly take into account that the problem at hand is constrained (i.e., d must be ≥ 0). Obviously, one may deploy a watchdog function which prevents d (through modifying search direction and/or step length) from becoming negative. However, this intervention may impede superlinear convergence.

We propose to solve (20) by means of Brent's root-finding algorithm (Brent, 1973, chap. 4). The search for a root is constrained to the interval $(0, a]$ (where $a > 0$ has to be chosen), and thus ensures non-negativeness of d and $\sigma_a^2 = \sigma_e^2 d$. Brent's method combines the sureness of bisection with the speed of a higher-order method. It keeps track of whether a supposedly superlinear method is actually converging the way it is supposed to, and, if it is not, it intersperses bisection steps so as to guarantee at least linear convergence. This kind of super-strategy requires attention to bookkeeping detail, and also careful consideration of how roundoff errors can affect the guiding strategy (Press, Teukolsky, Vetterling, and Flannery, 1986, pp. 352–354). Hence, we use (a modification of) Brent's "zeroin" Fortran 77 code.

3.2 Estimation Bounds

Given some initial values, $\boldsymbol{\beta}_0$, v_0, and d_0, we may consider updating these estimates by solving equations (17), (19), and (20) in some sequential order right away. From a theoretical point of view, there is no objective against doing so. From the perspective of computation, however, it will prove useful to introduce two (pre-) estimation bounds for the variance components. As d is concerned, we have to consider two limiting situations: $d = 0$ and $d \to \infty$. Accordingly, we obtain v_{zero} and v_∞, respectively. These two cases depict a lower and an upper bound of estimates of v. It is easy to prove that the following relations hold

$$v_\infty \leq v_{ML} \leq v_{zero}\,, \tag{21}$$

[1] Speed of convergence: Suppose the sequence $\{\vartheta\}^{(s)}$ converges to ϑ_0 ($s = 1, 2, \ldots$). In numerical analysis, the speed at which a convergent sequence approaches the limit is determined by the values c and p in $\|\vartheta^{(s+1)} - \vartheta_0\| \leq c\|\vartheta^{(s)} - \vartheta_0\|^p$. For $0 < c \leq 1$ and $p = 1$, we shall say that the algorithm converges linearly. Likewise, we call the convergence superlinear, if $p > 1$ (given that a $c > 0$ exists). Note that convergence of order p means that the number of correct decimal places is roughly p times the number of iterations (see e.g., Small and Wang, 2003, chap. 3.1).

where v_{ML} denotes the ML estimator (see e.g., Demidenko, 2004, pp. 78–79). From the perspective of numerical optimization, these bounds are extremely useful since they determine the range of plausible values, which may guide the choice of initial values and indicate potential run-away values. Subsequently we shall study robust estimators of v_{zero} and v_∞.

3.2.1 Case I: Robust Estimate of v_{zero}

In the first case, we have $d = 0$ which implies that $\mathbf{\Omega}_i(v, d) = v(\mathbf{I}_i + d\mathbf{J}_i)$ reduces to $v_{zero}\mathbf{I}_i$ $(i = 1, \ldots, g)$. As a consequence, the estimator of $\boldsymbol{\beta}$ collapses to the Ordinary Least Squares (OLS) estimator, $\hat{\boldsymbol{\beta}}_0$, and the corresponding estimator $\hat{v}_{zero}$ of v_{zero} is an estimator of the residual variance. A robust estimate of v_{zero}, say, $\hat{v}^R_{zero}$, comes along with e.g. least trimmed squares (LTS) regression (Rousseeuw, 1984) or an M- or S-estimator of regression (see e.g., Maronna et al., 2006, chap. 5). This robust regression exercise not only yields an estimate of v_{zero} but also provides us with a starting value for $\boldsymbol{\beta}$ in order to initialized the iterative algorithm (see below). From the perspective of computation, the fast LTS method of Rousseeuw and Van Driessen (2006) offers a good trade-off between robustness and computation time for sample sizes up to about 20,000 (this limit depends heavily on the number of auxiliary variables). For larger data, a regression S-estimator is considerably faster.

3.2.2 Case II: Robust estimate of v_∞

In the second case, we consider letting $\lim_{d\to\infty} \mathbf{V}_i^{-1}(v, d)$, and obtain $\lim_{d\to\infty}[\mathbf{I}_{n_i} - d/(1 + dn_i)\mathbf{J}_{n_i}] = [\mathbf{I}_{n_i} - (1/n_i)\mathbf{J}_{n_i}]$, $i = 1, \ldots, g$. Note that letting the random-effect variance, d, approach infinity, corresponds to treating the u_i, $i = 1, \ldots, g$, in (1) as if they were fixed effects. Indeed, we shall consider the fixed-effects model as an alternative to the mixed linear model. Now, let the model matrix be partitioned as $[\mathbf{1}_n, \mathbf{X}]$, as well as the corresponding parameter vector, $\boldsymbol{\beta} = (\alpha, \boldsymbol{\gamma}^T)^T$ (cf. Assumption 2). The fixed-effects model writes

$$\mathbf{y}_i = \alpha\mathbf{1}_{n_i} + \mathbf{X}_i\boldsymbol{\gamma} + \mathbf{1}_{n_i}u_i + \boldsymbol{\varepsilon}_i\,, \qquad i = 1, \ldots, g\,, \tag{22}$$

where the $\{u_i; i = 1, \ldots, g\}$ are, in contrast to (2), unknown, but fixed parameters (not realizations of the area-level random effects). Model (22) is traditionally called 1-way classification model for the analysis of covariance (Searle, 1987, chap. 11.2).

If we put $\mathbf{K} = [\mathbf{1}_n|\mathbf{Z}]$, where $\mathbf{Z} = \text{diag}(\mathbf{1}_1, \ldots, \mathbf{1}_g)$, then model (22) can be written as a general linear model, $\mathbf{y} = \mathbf{X}\boldsymbol{\gamma} + \mathbf{K}\mathbf{d} + \boldsymbol{\varepsilon}$, where $\mathbf{d} = (\alpha, \mathbf{u}^T)^T$ with $\mathbf{u} = (u_1, \ldots, u_g)^T$; and $\boldsymbol{\varepsilon} = \text{diag}(\boldsymbol{\varepsilon}_1, \ldots, \boldsymbol{\varepsilon}_g)$. On rewriting the first equation of the normal equations,

$$\begin{bmatrix} \mathbf{K}^T\mathbf{K} & \mathbf{K}^T\mathbf{X} \\ \mathbf{X}^T\mathbf{K} & \mathbf{X}^T\mathbf{X} \end{bmatrix} \begin{bmatrix} \mathbf{d} \\ \boldsymbol{\gamma} \end{bmatrix} = \begin{bmatrix} \mathbf{K}^T\mathbf{y} \\ \mathbf{X}^T\mathbf{y} \end{bmatrix}\,, \tag{23}$$

of the general linear model, we obtain (denoting the generalized inverse by the superscript "-")

$$\mathbf{d} = (\alpha, \mathbf{u}^T)^T = (\mathbf{K}^T\mathbf{K})^-\mathbf{K}^T(\mathbf{y} - \mathbf{X}\boldsymbol{\gamma})\,, \tag{24}$$

which can be substituted into (23) to yield

$$\mathbf{X}^T\mathbf{P}^*\mathbf{X}\boldsymbol{\gamma} = \mathbf{X}^T\mathbf{P}^*\mathbf{y}\,, \qquad \text{with} \quad \mathbf{P}^* = \mathbf{I} - \mathbf{K}(\mathbf{K}^T\mathbf{K})^-\mathbf{K}^T\,, \tag{25}$$

where $\mathbf{P}^*$ is both symmetric and idempotent (Searle, 1971, pp. 341–42). Note that, although $(\mathbf{K}^T\mathbf{K})^-$ is not unique (since $\mathbf{K}^T\mathbf{K}$ has not full rank), it enters only in the form $\mathbf{K}(\mathbf{K}^T\mathbf{K})^-\mathbf{K}^T$, which is invariant to whatever generalized inverse, $\mathbf{G}$, is used. Thus the non-full rank property does not itself lead to manifold solutions of $\boldsymbol{\gamma}$. However, for reasons of numerical stability, we will avoid computing a brute-force generalized inverse. Instead we derive a very simple variant of $\mathbf{G}$ as follows. First, note that (24) is not invariant to the particular choice of $\mathbf{G} := (\mathbf{K}^T\mathbf{K})^-$. However, since any linear combination of $\mathbf{d}$, say, $\boldsymbol{\lambda}^T\mathbf{d}$, is estimable when $\boldsymbol{\lambda}^T = \mathbf{t}^T\mathbf{K}$ for some $\mathbf{t}$, we deliberately put one element of $\mathbf{d}$ equal to zero, and cross out the corresponding element in the normal equations (Searle, 1971, pp. 232–33). The obvious element to equate to zero in (24) is α. Thus, our generalized inverse shall be given by

$$\mathbf{G} = \begin{bmatrix} 0 & \mathbf{0}^T \\ \mathbf{0} & (\mathbf{Z}^T\mathbf{Z})^{-1} \end{bmatrix}, \qquad \text{where} \quad (\mathbf{Z}^T\mathbf{Z})^{-1} = \operatorname{diag}(1/n_1, \ldots, 1/n_g)\,. \tag{26}$$

Substituting $\mathbf{G}$ into (24) yields $\mathbf{d} = (\alpha, \mathbf{u}^T)^T$ with $\alpha = 0$ (by assumption) and $\mathbf{u} = (\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})$. In addition, we replace $\mathbf{P}^*$ in (25) by

$$\mathbf{P} = \mathbf{I} - \mathbf{Z}(\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T\,, \tag{27}$$

because it is computationally much simpler than $\mathbf{P}^*$. Note that pre-multiplying a matrix by $\mathbf{P}$ corresponds to centering the particular matrix by its column-wise arithmetic means. In the present context, column-wise centering corresponds to centering by the area-specific means. By symmetry and idempotency of $\mathbf{P}$ we shall use

$$\hat{\boldsymbol{\gamma}} = \big[(\mathbf{PX})^T\mathbf{PX}\big]^{-1}(\mathbf{PX})^T\mathbf{Py} \tag{28}$$

instead of (25), where $\boldsymbol{\gamma}$ is based on the centered data, $\mathbf{PX}$ and $\mathbf{Py}$.

It is evident from (28) that the influence of outliers in y on the estimates is unbounded. Thus we obtain robust estimates of $\boldsymbol{\gamma}$, say $\hat{\boldsymbol{\gamma}}^R$, by means of M-estimation of regression, which is defined by the estimating equations

$$(\mathbf{PX})^T\boldsymbol{\psi}_k\big[(\mathbf{Py} - \mathbf{PX}\hat{\boldsymbol{\gamma}}^R)/\hat{S}\big] = \mathbf{0}\,, \tag{29}$$

where $\hat{S} = 1.481 \cdot \text{median}\big(|r_{ij}|; r_{ij} \neq 0\big)$ is the (normalized) median absolute deviation (MAD) of the non-null residuals of (29) about zero. The condition of taking only non-null residuals prevents from underestimating the scale, which becomes an issue if the number of auxiliary variables is relatively large (cf. Maronna et al., 2006, p. 100). By $\boldsymbol{\psi}_k(\cdot)$ we denote the Huber ψ-function indexed by the tuning constant k. Another approach could be to estimate regression and scale simultaneously.

It is fruitful to note that $\mathbf{Py}$ can be expressed as $\mathbf{y} - \mathbf{Z}\boldsymbol{\mu}$, where $\boldsymbol{\mu}$ is the g-vector of area-specific means, $(\bar{y}_1, \ldots, \bar{y}_i, \ldots, \bar{y}_g)^T$ with $\bar{y}_i = (1/n_i)\sum_{j=1}^{n_i} y_{ij}$, $i = 1, \ldots, g$. This representation indicates that the breakdown point of (29) may be much lower than the one of a regular M-estimator of regression. This is a consequence of the centering procedure: centering the within-area units in a particular area i by the mean may turn $(n_i - 1)$ ordinary observations into outliers (typically with a reversed sign) if the area contains one single heavy outlier. From the perspective of breakdown point, a simple remedy is

to center the data by the area-specific median instead of the mean. This corresponds to replacing $\mathbf{Py}$ in (29) by $\bar{\mathbf{y}}^{med} = \mathbf{y} - \mathbf{Z}\boldsymbol{\eta}$, where $\boldsymbol{\eta}$ is the g-vector of area-specific medians, $(\text{med}_{i=1}(y_{1j}), \ldots, \text{med}_{i=g}(y_{gj}))^T$. This approach resembles the "median polish" strategy, which has been proposed for the 2-way analysis of variance (Tukey, 1977).

Now, in order to obtain the robust variance pre-estimation bound, v_∞, we have to solve (29) with $\mathbf{Py}$ replaced by $\bar{\mathbf{y}}^{med}$ and obtain $\hat{v}_\infty^R$ from

$$\hat{v}_\infty^R = \hat{S}^2 \,. \tag{30}$$

An alternative approach has been proposed by Birch and Myers (1982). They obtain M-estimates for $\boldsymbol{\gamma}$ and $\{u_i, i = 1, \ldots, g\}$, solving the system of EE, respectively,

$$\sum_{i=1}^{g}\sum_{j=1}^{n_i} \psi_k(r_{ij}/S)\mathbf{x}_{ij} = \mathbf{0} \,, \tag{31}$$

$$\sum_{j=1}^{n_i} \psi_k(r_{ij}/S) = 0 \,, \qquad i = 1, \ldots, g \,, \tag{32}$$

where $r_{ij} = y_{ij} - u_i - \mathbf{x}_{ij}^T\boldsymbol{\gamma}$ and S is the normalized MAD of the residuals. In essence, this strategy consists of computing a relatively large number of M-estimates which is rather time consuming and therefore not the optimal strategy in order to compute pre-estimation bounds.

3.3 Algorithmic Details

Up to this point, we studied estimating equations and pre-estimation bounds. In this subsection we focus on implementation and algorithmic issues.

3.3.1 Computational Issues

As concerned with computing speed, memory allocation, and floating-point arithmetic considerations, we arranged all vector and matrix operations to make them rich in level-1 procedures—i.e., procedures which operate on vectors of size n and involve $\mathcal{O}(n)$ floating-point operations (cf. Golub and Loan, 1996).[2] With respect to elementary operations, we rely on the procedures in BLAS (Blackford et al., 2002) and LAPACK (E. Anderson et al., 2000). Further, we avoid computing any brute-force matrix inverse and use the expressions

$$|\mathbf{V}_i| = |\mathbf{I}_i + d\mathbf{1}_i\mathbf{1}_i^T| = 1 + dn_i \qquad \text{and} \qquad \mathbf{V}_i^{-1} = \mathbf{I}_i - \frac{d}{1 + dn_i}\mathbf{1}_i\mathbf{1}_i^T \tag{33}$$

for determinant and inverse of the matrix $\mathbf{V}_i$, $i = 1, \ldots, g$, (see e.g., Searle et al., 1992, p. 79). In addition, we obtain a closed-form expression of $\mathbf{V}_i^{-1/2}$ as follows. Denote by $\mathbf{L}_i\mathbf{D}_i\mathbf{L}_i^T$ the eigenvalue decomposition of the $(n_i \times n_i)$ matrix $\mathbf{V}_i$, where $\mathbf{L}_i$ is the $(n_i \times n_i)$ matrix whose columns correspond to the eigenvectors of $\mathbf{V}_i$; $\mathbf{D}_i = \text{diag}(\lambda_1, \ldots, \lambda_{n_i})$ is

[2] Level-2 matrix procedures involve arrays of size mn and are of order $\mathcal{O}(mn)$.

the $(n_i \times n_i)$ matrix of the eigenvalues λ_j $(j = 1, \ldots, n_i)$. It is not difficult to see that $\mathbf{V}_i$ has only two distinct eigenvalues: the first eigenvalue is $\lambda_1 = 1 + dn_i$ (with multiplicity one) and the remaining $(n_i - 1)$ eigenvalues are one. Now, we define a real-valued function f of the matrix $\mathbf{V}_i$ that corresponds to a function of a scalar as $f(\mathbf{V}_i) = \mathbf{L}_i \text{diag}(f(\lambda_1), \ldots, f(\lambda_{n_i}))\mathbf{L}_i^T$. Some straightforward computations with $f(u) = u^{-1/2}$ give

$$\mathbf{V}_i^{-1/2} = \frac{1}{n}\left(\frac{1}{\sqrt{1+dn_i}} - 1\right)\mathbf{1}_i\mathbf{1}_i^T + \mathbf{I}_i\,. \tag{34}$$

The formulas (33) and (34) simplify computation considerably.

Alternatively, one may define $\mathbf{V}_i^{-1/2} := \mathbf{A}_i$, where the upper triangular matrix $\mathbf{A}_i$, with positive diagonal elements, is obtained from the Cholesky decomposition, $\mathbf{A}_i^T\mathbf{A}_i = \mathbf{V}_i^{-1}$. Sinha and Rao (2009), on the other hand, use $\mathbf{U}_i^{-1/2}[\mathbf{y}_i - \mathbf{X}_i\boldsymbol{\beta}]$, instead of $\mathbf{V}_i^{-1/2}[\mathbf{y}_i - \mathbf{X}_i\boldsymbol{\beta}]$, where $\mathbf{U}_i$ is a diagonal matrix with elements u_{jj}, $j = 1, \ldots, n_i$, equal to the diagonal elements of the covariance matrix of $\mathbf{y}_i$ (for all $i = 1, \ldots, g$).

3.3.2 Initialization

The choice of starting values is crucial in terms of speed and numerical stability of the algorithm. Extensive simulation showed that the method is best initialized by the choice

$$\boldsymbol{\beta}_0^T \leftarrow \left(0, \hat{\boldsymbol{\gamma}}^R\right), \qquad v_0 \leftarrow \hat{v}_\infty^R\,, \qquad d_0 \leftarrow 200\,, \tag{35}$$

where $\hat{\boldsymbol{\gamma}}^R$ and $\hat{v}_\infty^R$ are obtained from the pre-estimation–bound exercise in (29) and (30), respectively. For d_0 to be a reasonable starting value, it is sufficient to choose a relatively large number (200 works for most applications).

3.3.3 Algorithm

Given the starting values $\boldsymbol{\beta}_0$, v_0, and d_0, [say, $\boldsymbol{\tau}_0^T = (\boldsymbol{\beta}_0^T, v_0, d_0)$], we consider solving the estimating equations iteratively. The core part of the algorithm is a series of nested loops. Denote by $i = 0, 1, 2, \ldots,$ the running index of the main loop. Define the termination-rule constants δ, $\delta_{\boldsymbol{\beta}}$, and δ_v such that $\epsilon^{1/2} \leq \delta < 0.001$, $\epsilon^{1/2} \leq \delta_{\boldsymbol{\beta}} < 0.001$, and $\epsilon^{1/2} \leq \delta_v < 0.001$, where $\epsilon = 2.2 \times 10^{-16}$ is the machine epsilon (in the 64-bit double precision floating-point model; on most modern computers).

For $i = 0, 1, 2, \ldots, I$ Do

1. On the ith iteration, given $\boldsymbol{\beta}_i$, v_i, and d_i, put $\boldsymbol{\beta}_i^0 \leftarrow \boldsymbol{\beta}_i$ and compute an update: $\boldsymbol{\beta}_i^{j+1} \leftarrow f(\boldsymbol{\beta}_i^j; \cdot)$ while looping over $j = 0, 1, 2, \ldots,$ where f denotes (17). If $\|\boldsymbol{\beta}_i^{j+1} - \boldsymbol{\beta}_i^j\|_p \leq \delta_{\boldsymbol{\beta}}$, then stop and put $\boldsymbol{\beta}_{i+1} \leftarrow \boldsymbol{\beta}_i^{j+1}$.

2. On the ith iteration, given $\boldsymbol{\beta}_{i+1}$, v_i, and d_i, put $v_i^0 \leftarrow v_i$ and compute an update: $v_i^{j+1} \leftarrow f(v_i^j; \cdot)$ while looping over $j = 0, 1, 2, \ldots$. Here, f denotes (19). If $|v_i^{j+1} - v_i^j| \leq \delta_v$, then stop and put $v_{i+1} \leftarrow v_i^{j+1}$.

3. At the outset of the ith iteration, we have $\boldsymbol{\beta}_{i+1}$, v_{i+1}, and d_i. Solve $d_{i+1} \leftarrow f(d_i; \cdot)$ by means of Brent's algorithm, where f denotes (20).

If $\|\boldsymbol{\tau}_{i+1} - \boldsymbol{\tau}_i\|_{p+2} \leq \delta$, then stop.

The sketch of the algorithm comprises only the most important elements. The numerical tests whether an updated value behaves well (e.g., lies within the pre-estimation bounds), have been omitted in the above display. The final estimates are given by: $\hat{\boldsymbol{\beta}}^R \leftarrow \boldsymbol{\beta}_{i+1}$, $[\hat{\sigma}_e^2]^R = \hat{v}^R \leftarrow v_{i+1}$, and $\hat{d}^R \leftarrow d_{i+1}$. By means of identity (8), we obtain $[\hat{\sigma}_a^2]^R = [\hat{\sigma}_e^2]^R \hat{d}^R$.

3.4 Robust Prediction

Up to this point we have dealt with (robustly) estimating the model parameters $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$. Now, we consider robustly predicting the random effects (and subsequently the small-area means). We assume that $N_i \gg n_i$ ($\forall i = 1, \ldots, g$) holds. On rewriting (6) using (8), the area-level predicting equations are given by

$$\hat{\mu}_i = \bar{\mathbf{x}}_{i\bullet}^T \hat{\boldsymbol{\beta}} + \hat{u}_i \,, \qquad i = 1, \ldots, g \,, \tag{36}$$

with

$$\hat{u}_i = \hat{a} \mathbf{1}_i^T \boldsymbol{\Omega}_i(\hat{v}, \hat{a})^{-1} \big[\mathbf{y}_i - \mathbf{X}_i \hat{\boldsymbol{\beta}}\big] \,, \tag{37}$$

where $\boldsymbol{\Omega}_i(\hat{v}, \hat{a})^{-1} \equiv (1/\hat{v}) \mathbf{V}_i(\hat{d})^{-1}$. If $N_i \gg n_i$ does not hold we may proceed as in Section 2.1. From the mathematical display, it is apparent that replacing $\hat{\boldsymbol{\beta}}$, $\hat{v}$, and $\hat{a}$ by robust estimates is not sufficient in order to robustly predict μ_i. As Sinha and Rao (2009) indicate $\hat{u}_i$ has to be replaced by a robustly predicted random effect, $\hat{u}_i^R$, as well. They therefore propose to solve Fellner's robust mixed-model equation (Fellner, 1986)

$$\mathbf{1}_i^T \frac{1}{\sqrt{v}} \boldsymbol{\psi}_k \left(\frac{1}{\sqrt{v}} (\mathbf{y}_i - \mathbf{X}_i \boldsymbol{\beta} - \mathbf{1}_i \hat{u}_i^R) \right) - \frac{1}{\sqrt{a}} \boldsymbol{\psi}_k \left(\frac{1}{\sqrt{a}} \hat{u}_i^R \right) = 0 \,, \tag{38}$$

for u_i. In order to solve (38), Sinha and Rao (2009) use a Newton-Raphson algorithm using another first-order Taylor series expansion of (38) w.r.t. u_i. Consequently, computation is very involved. In particular, we may encounter all the numerical difficulties associated with the NR method (as discussed above) here as well.

However, one can obtain robust predictions far more easily (cf. Copt and Victoria-Feser, 2009). If we put $\boldsymbol{\psi}_c(\mathbf{u}_i) = \big(\psi_c(u_{i1}), \ldots, \psi_c(u_{in_i})\big)^T$, where $\psi_c(\cdot)$ is the Huber ψ-function indexed by the robustness tuning constant c, then we may write

$$\hat{u}_i^R = \kappa \frac{\hat{a}^R}{\sqrt{\hat{v}^R}} \mathbf{1}_i^T \mathbf{V}_i^{-1/2}(\hat{d}^R) \boldsymbol{\psi}_c \left[\frac{1}{\sqrt{\hat{v}^R}} \mathbf{V}_i^{-1/2}(\hat{d}^R) [\mathbf{y}_i - \mathbf{X}_i \hat{\boldsymbol{\beta}}] \right] \,, \tag{39}$$

where $\kappa = [-2c\phi(c) + 2\Phi(c) - 1 + 2c^2(1 - \Phi(c))]^{-1/2}$, with ϕ and Φ the pdf and cdf of the standard normal distribution, respectively. Note that κ is kind of a consistency correction term which has been chosen in order $\hat{u}_i^R$ to behave similarly to $\tilde{u}_i$ at the core model. In essence, we follow Heritier et al. (2009) and impose the (implicit) moment conditions that $\mathrm{E}[\hat{u}_i^R] = 0$ and $\mathrm{var}[\hat{u}_i^R] = \mathrm{var}[\tilde{u}_i]$ (Heritier et al., 2009, pp. 113–114). Thus, the robust predictor of the area mean, $\hat{\mu}_i^R$ ($i = 1, \ldots, g$), – referred to as robust EBLUP (REBLUP) of μ_i – is given by $\hat{\mu}_i^R = \bar{\mathbf{x}}_{i\bullet}^T \boldsymbol{\beta}^R + \hat{u}_i^R$.

Some issues of the robust prediction method are noteworthy to comment on. First, note that pre-multiplying the area-specific vector of residuals, $\mathbf{r}_i = \mathbf{y}_i - \mathbf{X}_i\hat{\boldsymbol{\beta}}$, in (39) by $\mathbf{V}_i^{-1/2}$ from (34) will transmit the effect of even one single outlying residual, $r_{ij} = y_{ij} - \mathbf{x}_{ij}^T\hat{\boldsymbol{\beta}}$, say, to the vector of all other within-area residuals. That is, on pre-multiplying $\mathbf{r}_i$, the first term of (34) yields the mean of $\mathbf{r}_i$ (times a constant), which is non-robust per se. Thus, from the perspective of robustness (and with regard to breakdown point considerations), the term $\mathbf{V}_i^{-1/2}\mathbf{r}_i$ with $\mathbf{r}_i = \mathbf{y}_i - \mathbf{X}_i\hat{\boldsymbol{\beta}}$ in (39) should be replaced by

$$\left(\frac{1}{\sqrt{1+dn_i}} - 1\right)\mathbf{1}_i\bar{r}_i^{med} + \mathbf{r}_i\,, \tag{40}$$

$\bar{r}_i^{med}$ denoting the median of $\mathbf{r}_i$. Alternatively, and by Assumption 4 it is sufficient to use $\bar{y}_i^{med} - (1/n_i)\mathbf{1}_i^T\mathbf{X}_i\hat{\boldsymbol{\beta}}$, where $\bar{y}_i^{med}$ is the median of $\mathbf{y}_i$, instead of $\bar{r}_i^{med}$ in (40).

Second, $\psi_c(\cdot)$ in (39) can be replaced by any other non-redescending, odd, bounded function. The restriction on non-redescending functions is crucial since redescending ψ-functions lead, for sufficiently large residuals in (39), to realizations of u_i equal to zero (i.e., mimicking a synthetic estimator) which is not meaningful under model (2). Third, the choice of c can, in principle, be different from the choice of k in the BIEEs.

4 Mean Squared Error Estimation

Estimation of mean squared prediction error (MSPE) is a very challenging problem. Given the complex nature of the REBLUP estimators and the lack of knowledge on the underlying distribution of the u_i and e_{ij}, Sinha and Rao (2009) noted that is not possible to adopt the existing methods used for MSPE estimation. We follow the proposal of Sinha and Rao (2009) and adopt a parametric bootstrap (see Lahiri, 2003 and Hall and Maiti, 2006 for more details on bootstrap estimates in SAE) based on the robust quantities $\hat{\boldsymbol{\beta}}^R$ and $\hat{\boldsymbol{\theta}}^R$ to estimate

$$\text{MSPE}(\hat{\mu}_i^R) = \text{E}\{\hat{\mu}_i^R - \mu_i\}^2\,. \tag{41}$$

The method works as follows.

1. For the given $\hat{\boldsymbol{\beta}}^R$ and $\hat{\boldsymbol{\theta}}^R = (\hat{v}^R, \hat{a}^R)^T$, generate area-specific random effects u_i^* and random errors e_{ij}^* from $\mathcal{N}(0, \hat{a}^R)$ and $\mathcal{N}(0, \hat{v}^R)$, respectively. Then we create a bootstrap sample from the model

$$y_{ij}^* = \mathbf{X}_i\hat{\boldsymbol{\beta}}^R + u_i^* + e_{ij}^*\,, \qquad j = 1, \ldots, n_i\,; \quad i = 1, \ldots, g\,, \tag{42}$$

2. Generate $b = 1, \ldots, B$ bootstrap samples $\{\mathbf{y}^{*[1]}, \mathbf{y}^{*[2]}, \ldots, \mathbf{y}^{*[B]}\}$ from the bootstrap population model (42). For each bootstrap sample $\mathbf{y}^{*[b]}$ ($b = 1, \ldots, B$), compute the robust bootstrap estimates $\hat{\boldsymbol{\beta}}^{R[b]}$, $\hat{\boldsymbol{\theta}}^{R[b]}$, and $\hat{u}^{R[b]}$ and robustly predict/estimate $\hat{\mu}_i^{R[b]} = \bar{\mathbf{x}}_{i\bullet}^T\hat{\boldsymbol{\beta}}^{R[b]} + \hat{u}_i^{R[b]}$.

3. Compute a bootstrap estimate of MSPE$\{\hat{u}_i^R\}$ as

$$\text{MSPE}_B\{\hat{u}_i^R)\} = \frac{1}{B}\sum_{b=1}^{B}\left(\hat{\mu}_i^R(\hat{\boldsymbol{\beta}}^{[R[b]}, \hat{\boldsymbol{\theta}}^{R[b]}, \hat{u}_i^{R[b]}) - \mu_i(\hat{\boldsymbol{\beta}}^R, \hat{\boldsymbol{\theta}}^R, \hat{u}_i^{*[b]})\right)^2. \tag{43}$$

This method works remarkably good with respect to computing time and in terms of providing a reasonably precise estimate of the uncertainty associated with predicting the area-level means. However, the method tends to slightly underestimate the true MSPE. The underestimation results mainly because the uncertainty of estimating $\boldsymbol{\beta}$ has not been taken into account.

5 Simulation

We implemented a small model-based simulation study to investigate the performance of the proposed method.The proposed algorithm is implemented in the R-package `rsae` (Schoch, 2011c); see R Development Core Team (2011) for more details on the R language and environment for statistical computing.

The data were generated from model

$$y_{ij} = (1, x_{ij})^T (1, 1) + u_i + e_{ij}\,, \qquad j = 1, \ldots, n\,; \quad i = 1, \ldots, g\,,$$

where $x_{ij} \sim \mathcal{N}(0, 1)$. Each Monte Carlo sample comprises $g = 20$ areas and $n = 5$ within-area units (overall, $N = 80$ observations; balanced data). In line with Stahel and Welsh (1997), we allow for contamination (by means of a normal mixture, $(1 - \varepsilon) \cdot \mathcal{N}(0, 1) + \varepsilon \cdot \mathcal{N}(0, \gamma)$, where γ can be chosen) in either or both of the random effect distributions, producing four combinations:

(0,0) no contamination; $e_{ij} \sim \mathcal{N}(0, 1)$ and $u_i \sim \mathcal{N}(0, 1)$,

(e,0) $e_{ij} \sim (1 - \varepsilon) \cdot \mathcal{N}(0, 1) + \varepsilon \cdot \mathcal{N}(0, 41)$ and $u_i \sim \mathcal{N}(0, 1)$,

(0,u) $e_{ij} \sim \mathcal{N}(0, 1)$ and $u_i \sim (1 - \varepsilon) \cdot \mathcal{N}(0, 1) + \varepsilon \cdot \mathcal{N}(0, 41)$,

(e,u) $e_{ij} \sim (1 - \varepsilon) \cdot \mathcal{N}(0, 1) + \varepsilon \cdot \mathcal{N}(0, 41)$ and $u_i \sim (1 - \varepsilon) \cdot \mathcal{N}(0, 1) + \varepsilon \cdot \mathcal{N}(0, 41)$.

The simulation study in this paper serves primarily as a proof of concept. We therefore provide only a small number of contamination scenarios. The relative amount of contamination is $\varepsilon = 0.05$ for all simulations. Each scenario is evaluated by 1000 Monte Carlo (MC) simulation trials. The simulated MC distribution of an estimator of ϑ, say $\hat{\vartheta}$, is summarized by the average bias, $\mathrm{B}(\hat{\vartheta})$, and mean square error, $\mathrm{MSE}(\hat{\vartheta})$, and the robust analogues based on medians, respectively, $\mathrm{rB}(\hat{\vartheta}) = \mathrm{med}[\hat{\vartheta} - \vartheta^*]$ and $\mathrm{rMSE}(\hat{\vartheta}) = \mathrm{med}[|\hat{\vartheta} - \vartheta^*|]$, where ϑ^* denotes the true value. For ease of simplicity (and by equivariance considerations), it is sufficient to set the true parameters equal to one, i.e., $\boldsymbol{\beta}^* = (1, 1)^T$, $[\sigma_e^2]^* = 1$, and $[\sigma_a^2]^* = 1$.

We report only the simulation results for the variance components σ_e^2 and σ_a^2 (Table 1 and 2). For the scenario of uncontamined data, $(0, 0)$, we also report the results of the maximum likelihood method of the "lme" function (i.e., gold-standard function) in the R-package "nlme" (Pinheiro, Bates, DebRoy, Sarkar, and R Core Team, 2009; denoted by "lme(ml)" in Tables 1 and 2). Note that the M-estimator (denoted by "huberm" in the tables) with robustness tuning constant $k = 2000$ mimics the ML estimator.

The findings of the small simulation exercise can be summarized as follows.

- The "huberm" *method converged in all 1000 Monte Carlo trials* for each simulation configuration (i.e., contamination scenario and choice of k). This is in sharp contrast to the results reported by Richardson (1995) and Chaubey and Venkateswarlu (2002) (among others). The proposed algorithm may, on the other hand, fail to converge when the amount of contamination, ε, is larger than the breakdown point (which can be rather low in the case of unbalanced data; see Schoch (2011b)).

- The results of the "huberm" method mimicking the ML estimator are equal (up to the 6th or 7th decimal place) with those of the gold-standard method "lme".

- In the presence of contamination, the M-estimator has a smaller bias than the corresponding ML estimator. Even more important, the MSE is considerably smaller. In contrast, the loss of efficiency of the M-estimator in the absence of contamination is almost negligible. These findings remain valid if one considers the robust criteria (rB and rMSE in Table 2). In the presence of contamination, rMSE tends to be smaller than MSE which indicates that the MC distribution is skewed.

- Contamination of the model error, e_{ij}, affects the robust estimates of σ_a^2 very little, since the contamination affects the diagonal elements of the variance of y_{ij} but not the off-diagonal elements. Contamination of the area-specific random effects, u_i, affects both diagonal and off-diagonal elements of the variance (see also Welsh and Richardson, 1997, p. 348). When both components are contaminated, the effects on the estimates are the combination of the effects of contaminating the components one at a time (see also Stahel and Welsh, 1997, p. 315).

- In the simulation exercise, we had focused on two choices of the tuning constant k. It goes without saying that one may obtain better estimates (i.e., better in terms of a reasonable risk/loss function) trying different choices. Our experience supports the finding of Stahel and Welsh (1997) that fine tuning pays more in estimating these models than it does with simpler models (p. 315). Nonetheless, the gains in efficiency are large.

- Computing the robust estimates based on data consisting of $g = 500$ areas, each of which has $n = 20$ units (i.e., $500 \times 20 = 10000$ observations), takes on average 1.3 seconds on an ordinary desktop computer (computing on a single core of an x86_64 processor, 2.83 GHz, 4 GB RAM; R v.2.13.1; openSUSE Linux 11.4).

6 Conclusion

In this paper, we developed a robust method for the basic unit-level model which is based on M-estimators in mixed-linear models (Welsh and Richardson, 1997) and therefore conceptionally equivalent, but slightly different, to the proposal of Sinha and Rao (2009). In contrast to Newton–Raphson- or Fisher-scoring-type algorithms, the proposed algorithm is numerically stable and fast (also for very large datasets). Notably, the estimates of the variance components are non-negative in contrast to those from e.g., the NR method. In sharp contrast to other algorithms (cf. Chaubey and Venkateswarlu, 2002; Jiang et al.,

Table 1: Bias and MSE estimates of the variance components for the four contaminations scenarios

(e, a)	method	k	B($\hat{\sigma}_a^2$)	MSE($\hat{\sigma}_a^2$)	B($\hat{\sigma}_e^2$)	MSE($\hat{\sigma}_e^2$)
$(0, 0)$	lme(ml)	–	-0.0801	0.1437	-0.0018	0.0348
	huberm	2000	-0.0801	0.1437	-0.0018	0.0348
	huberm	1.4	-0.0463	0.1655	-0.0235	0.0453
	huberm	1.2	-0.0701	0.1793	-0.0235	0.0462
$(0.05, 0)$	huberm	2000	-0.1027	0.2834	1.9474	2.0677
	huberm	1.4	-0.0764	0.2301	0.2587	0.0700
	huberm	1.2	-0.0450	0.2579	0.2164	0.0834
$(0, 0.05)$	huberm	2000	1.7364	7.1491	-0.0097	0.0352
	huberm	1.4	0.6322	0.6821	-0.0321	0.0428
	huberm	1.2	0.4996	0.5373	-0.0204	0.0545
$(0.05, 0.05)$	huberm	2000	1.8484	9.7617	2.0203	2.1784
	huberm	1.4	0.7560	1.4080	0.2628	0.0734
	huberm	1.2	0.6732	1.0512	0.2521	0.0838

Notes: each criterion is computed based on 1000 Monte Carlo replications; (e, a) denotes the contamination scheme, where $a, e \in (0, 0.5]$; k is the robustness tuning constant of the Huber-type M-estimator; R packages: nlme (v. 3.1-96) and rsae (v. 0.1-3).

Table 2: Robust bias and robust MSE estimates of the variance components for the four contamination scenarios

(e, a)	method	k	B($\hat{\sigma}_a^2$)	MSE($\hat{\sigma}_a^2$)	B($\hat{\sigma}_e^2$)	MSE($\hat{\sigma}_e^2$)
$(0, 0)$	lme(ml)	–	-0.1064	0.2707	-0.0134	0.1279
	huberm	2000	-0.1064	0.2707	-0.0134	0.1279
	huberm	1.4	-0.0920	0.2824	-0.0329	0.1490
	huberm	1.2	-0.1460	0.3064	-0.0318	0.1509
$(0.05, 0)$	huberm	2000	-0.1675	0.3790	1.6308	1.6308
	huberm	1.4	-0.1579	0.3401	0.2525	0.2630
	huberm	1.2	-0.1330	0.3427	0.1902	0.2230
$(0, 0.05)$	huberm	2000	0.8582	0.8582	-0.0177	0.1277
	huberm	1.4	0.4690	0.5333	-0.0526	0.1448
	huberm	1.2	0.3645	0.4583	-0.0365	0.1632
$(0.05, 0.05)$	huberm	2000	0.7833	0.8540	1.6719	1.6719
	huberm	1.4	0.5175	0.5856	0.2540	0.2642
	huberm	1.2	0.4392	0.5256	0.2300	0.2451

Notes: each criterion is computed based on 1000 Monte Carlo replications; (e, a) denotes the contamination scheme, where $a, e \in (0, 0.5]$; k is the robustness tuning constant of the Huber-type M-estimator; R packages: nlme (v. 3.1-96) and rsae (v. 0.1-3).

2007) the Monte Carlo simulation study showed that the method converges always (given that the amount of contamination is lower than the breakdown point). Further, we derived a much simpler (thus considerably faster) method for robustly predicting the small-area means than Sinha and Rao (2009) did. All the methods of this paper are implemented in R (R Development Core Team, 2011), R-package: `rsae`, see Schoch (2011c).

Acknowledgements

I would like to thank Beat Hulliger for very stimulating discussions and an anonymous referee for valuable comments. In addition, I am grateful to Matthias Templ and Ralf Münnich for the support during the AMELI project. Last but not least, I am very grateful to Nikos Tzavidis and Timo Schmid for critically evaluating the software.

This work was partly funded by the European Union (represented by the European Commission) within the 7th Framework Programme for research (theme 8; SSH-2007-6.2-01), project: AMELI (Advanced Methodology for European Laeken Indicators), grant agreement no.: 217322).

Some of this research is part of the authors Ph.D. Thesis at the University of Trier.

References

Anderson, E., Bai, Z., Bischof, C., Blackford, L. S., Demmel, J., Dongarra, J., et al. (2000). *LAPACK users' guide* (3rd ed.). Philadelphia: Society for Industrial and Applied Mathematics (SIAM).

Anderson, T. W. (1973). Asymptotically efficient estimation of covariance matrices with linear covariance structure. *The Annals of Statistics*, *1*, 135-141.

Battese, G. E., Harter, R. M., and Fuller, W. A. (1988). An error component model for prediction of county crop areas using. *Journal of the American Statistical Association*, *83*, 28-36.

Bednarski, T., and Zontek, S. (1996). Robust estimation of parameters in a mixed unbalanced model. *The Annals of Statistics*, *24*, 1493-1510.

Birch, J. B., and Myers, R. H. (1982). Robust analysis of covariance. *Biometrics*, *38*, 699-713.

Blackford, L. S., Petitet, A., Pozo, R., Remington, K., Whaley, R. C., Demmel, J., et al. (2002). An updated set of basic linear algebra subprograms (BLAS). *ACM Transactions on Mathematical Software*, *28*, 135-151.

Brent, R. P. (1973). *Algorithms for Minimization without Derivatives*. Englewood Cliffs. NJ: Prentice-Hall.

Chambers, R. (1986). Outlier robust finite population estimation. *Journal of the American Statistical Association*, *81*(396), 1063-1069.

Chambers, R., and Tzavidis, N. (2006). M-quantile models for small area estimation. *Biometrika*, *92*(2), 255-268.

Chaubey, Y. P., and Venkateswarlu, K. (2002). Robust estimators for the one-way variance component model. In N. Balakrishnan (Ed.), *Advances on Methodological and Applied Aspects of Probability and Statistics* (p. 241-260). New York: Taylor & Francis.

Copt, S., and Victoria-Feser, M.-P. (2006). High breakdown inference in the mixed linear model. *Journal of the American Statistical Association*, *101*, 292-300.

Copt, S., and Victoria-Feser, M.-P. (2009). *Robust predictions in mixed linear models* (Technical Report). Geneva: University of Geneva.

Datta, G. S., and Lahiri, P. (2000). A unified measure of uncertainty of estimated best

linear unbiased predictors in small area estimation problems. *Statistica Sinica*, *10*, 613-627.

Demidenko, E. (2004). *Mixed Models: Theory and Applications*. Hoboken: John Wiley & Sons.

Fellner, W. (1986). Robust estimation of variance components. *Technometrics*, *28*, 51-60.

Gentle, J. E. (2007). *Matrix Algebra: Theory. Computations. and Applications in Statistics*. New York: Springer.

Golub, G., and Loan, C. van. (1996). *Matrix Computations* (3rd ed.). London: The Johns Hopkins University Press.

Hall, P., and Maiti, T. (2006). On parametric bootstrap methods for small area prediction. *Journal of the Royal Statistical Society. Series B*, *68*, 221-238.

Hartley, H. O., and Rao, J. N. K. (1967). Maximum likelihood estimation for the mixed analysis of variance. *Biometrika*, *54*, 93-108.

Harville, D. A. (1977). Approaches to variance component estimation and to related problems. *Journal of the American Statistical Association*, *72*, 320-338.

Heritier, S., Cantoni, E., Copt, S., and Victoria-Feser, M.-P. (2009). *Robust Methods in Biostatistics*. New York: John Wiley & Sons.

Huber, P. J. (1964). Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, *35*(1), 73-101.

Huber, P. J. (1981). *Robust Statistics*. New York: John Wiley & Sons.

Jiang, J., and Lahiri, P. (2006). Mixed model prediction and small area estimation. *TEST*, *15*(1), 1-96.

Jiang, J., Luan, Y., and Wang, Y.-G. (2007). Iterative estimating equations: linear convergence and asymptotic properties. *The Annals of Statistics*, *35*, 2233-2260.

Lahiri, P. (2003). On the impact of bootstrap in survey sampling and small area estimation. *Statistical Science*, *18*, 199-210.

Lindstrom, M. J., and Bates, D. M. (1988). Newton-Raphson and EM algorithms for linear mixed-effects models for repeated-measures data. *Journal of the American Statistical Association*, *83*, 1014-1022.

Maronna, R. A., Martin, D., and Yohai, V. J. (2006). *Robust Statistics: Theory and Methods*. Chichester: John Wiley.

Miller, J. J. (1977). Asymptotic properties of maximum likelihood estimates in the mixed model of the analysis of variance. *The Annals of Statistics*, *5*, 746-765.

Pinheiro, J., Bates, D., DebRoy, S., Sarkar, D., and R Core Team the. (2009). nlme: Linear and Nonlinear Mixed Effects Models [Computer software manual]. (R package version 3.1-96)

Prasad, N., and Rao, J. (1990). The estimation of the mean squared error of small-area estimators. *Journal of the American Statistical Association*, *85*(409), 163-171.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (1986). *Numerical Recipes in Fortran 77: The Art of Scientific Computing*. Cambridge. UK: Cambridge University Press.

R Development Core Team. (2011). R: A Language and Environment for Statistical Computing [Computer software manual]. Vienna. Austria. Available from `http://www.R-project.org/` (ISBN 3-900051-07-0)

Rao, J. (2003). *Small Area Estimation*. Hoboken: Wiley.

Richardson, A. M. (1995). Some problems in estimation in mixed linear models. *PhD thesis. Department of Statistics. Australian National University. Canberra*.

Richardson, A. M. (1997). Bounded influence estimation in the mixed linear model. *Journal of the American Statistical Association*, *92*(437), 151-161.

Richardson, A. M., and Welsh, A. H. (1995). Robust restricted maximum likelihood in mixed linear models. *Biometrics*, *51*(4), 1429-1439.

Rocke, D. M. (1983). Robust statistical analysis of interlaboratory studies. *Biometrika*, *70*, 421-431.

Rocke, D. M. (1991). Robustness and balance in the mixed model. *Biometrics*, *47*, 303-309.

Rousseeuw, P. J. (1984). Least median of squares regression. *Journal of the American Statistical Association*, *79*, 871-880.

Rousseeuw, P. J., and Van Driessen, K. (2006). Computing LTS regression for large data sets. *Data Mining and Knowledge Discovery*, *12*, 29-45.

Särndal, C.-E., Swensson, B., and Wretman, J. (1992). *Model Assisted Survey Sampling*. New York: Springer.

Schoch, T. (2011a). The robust basic unit-level small area model: A simple and fast Fisher-scoring algorithm for large datasets. In *Proceedings of the Conference on New Technologies and Techniques in Statistics (NTTS).* Brussels: EUROSTAT.

Schoch, T. (2011b). Robust high breakdown point estimation in unit-level SAE models. In *Proceedings of the Conference on Small Area Estimation (SAE)* (p. 72-78). Trier. August 11-13: SAE.

Schoch, T. (2011c). rsae: Robust small area estimation [Computer software manual]. Available from `http://CRAN.R-project.org/package="rsae"` (R package version 0.1-4)

Searle, S. (1971). *Linear Models*. New York: John Wiley & Sons.

Searle, S. (1987). *Linear Models for Unbalanced Data*. New York: John Wiley & Sons.

Searle, S., Casella, G., and McCulloch, C. E. (1992). *Variance Components*. Hoboken: John Wiley & Sons.

Sinha, S. K., and Rao, J. (2009). Robust small area estimation. *The Canadian Journal of Statistics*, *37*(3), 381-399.

Small, C. G., and Wang, J. (2003). *Numerical Methods for Nonlinear Estimating Equations*. Oxford: Oxford University Press.

Stahel, W. A., and Welsh, A. H. (1997). Approaches to robust estimation in the simplest variance components model. *Journal of Statistical Planning and Inference*, *57*, 295-319.

Tukey, J. W. (1977). *Exploratory Data Analysis*. Reading. MA: Addison-Wesley.

Welsh, A. H., and Richardson, A. M. (1997). Approaches to the robust estimation of mixed models. In G. Maddala and C. Rao (Eds.), *Robust Inference* (Vol. 13, p. 343-384). Amsterdam: Elsevier Science.

Author's address:

Tobias Schoch

University of Applied Sciences Northwestern Switzerland
School of Business
Riggenbachstrasse 16
CH-4600 Olten
Switzerland

E-Mail: tobias.schoch@fhnw.ch