SORT 44 (1) January-June 2020, 3-38 DOI: 10.2436/20.8080.02.93

Small area estimation of additive parameters under unit-level generalized linear mixed models

Tomáš Hobza[1], Yolanda Marhuenda[2] and Domingo Morales[2]

Abstract

Average incomes and poverty proportions are additive parameters obtained as averages of a given function of an income variable. As the variable income has an asymmetric distribution, it is not properly modelled via normal distributions. When dealing with this type of variable, a first option is to apply transformations that approximate normality. A second option is to use non-symmetric distributions from the exponential family. This paper proposes unit-level generalized linear mixed models for modelling asymmetric positive variables and for deriving three types of predictors of small area additive parameters, called empirical best, marginal and plug-in. The parameters of the introduced model are estimated by applying the maximum likelihood method to the Laplace approximation of the likelihood. The mean squared errors of the predictors are estimated by parametric bootstrap. The introduced methodology is applied and illustrated under unit-level gamma mixed models. Some simulation experiments are carried out to study the behaviour of the fitting algorithm, the small area predictors and the bootstrap estimator of the mean squared errors. By using data of the Spanish living condition survey of 2013, an application to the estimation of average incomes and poverty proportions in counties of the region of Valencia is given.

MSC: 62J12 Generalized linear models; 62P25 Applications to social sciences; 62D05 Sampling theory, sample surveys.

Keywords: Average income, poverty proportion, generalized linear mixed models, empirical best predictor, mean squared error, bootstrap.

1 Introduction

Many of the socioeconomic indicators published by statistical offices are additive parameters. These parameters are the sums of the transformed values that an objective variable takes in the population units and its definition depends on the selected variable and transformation. This paper deals with the small area estimation (SAE) of additive parameters, with particularizations to average incomes and poverty proportions. The problems of SAE appear when the sample sizes are small in the target population subsets,

[1] Department of Mathematics, Faculty of Nuclear Sciences and Physical Engineering, Czech Technical University in Prague, Trojanova 13, 120 00 Prague 2, Czech Republic. (corresponding author, email: tomas.hobza@fjfi.cvut.cz)

[2] Operations Research Center, Miguel Hernández University of Elche, Spain.
Received: September 2018
Accepted: November 2019

called small areas or domains, so that the direct estimators are not reliable. A domain direct estimator is obtained by using only the domain data. The low amount of data can be overcome by using statistical models that introduce additional information via auxiliary variables, data from other domains and variance-covariance structures. Model-based predictors of domain parameters are generally more efficient than direct estimators. See the monograph of Rao and Molina (2015) for an introduction to SAE, linear mixed models (LMM) and related issues.

Average incomes and some income-based poverty indicators are sums of transformed individual incomes. Some SAE methods based on unit-level models have been proposed in the literature for this type of parameters. Elbers, Lanjouw and Lanjouw (2003) introduced estimators based on the predictions of a fitted marginal nested error regression (NER) model. Molina and Rao (2010) proposed empirical best predictors (EBP) by employing the predictions of a NER model conditioned to the observed sample. This approach was extended to two-fold NER models by Marhuenda et al. (2017). Hobza and Morales (2013) derived predictors of means of household normalized net annual incomes under random regression coefficient models. Molina, Nandram and Rao (2015) proposed a hierarchical Bayes approach and Guadarrama, Molina and Rao (2014) compared several poverty mapping methods based on unit level models. Hobza and Morales (2016), Hobza, Morales and Santamaría (2018) derived EBPs based on unit-level logit mixed models, Tzavidis et al. (2008), Chambers, Salvati and Tzavidis (2012, 2016) introduced predictors based on M-quantile regression models. Karlberg (2014) proposed log-transformation mixed small area prediction models incorporating a logistic component for skewed data in the presence of zeroes. Dreassi, Petrucci and Rocco (2014), Fabrizi, Ferrante and Trivisano (2017) and Moura, Silva and Neves (2017) gave hierarchical Bayes procedures for skewed survey data. By using temporal and spatio-temporal area-level models, Esteban et al. (2012a, 2012b), Marhuenda, Molina and Morales (2013) and Morales, Pagliarella and Salvatore (2015) derived also model-based predictors of poverty indicators. Boubeta, Lombardía and Morales (2016, 2017) introduced empirical best predictors (EBP) of poverty proportions based on Poisson mixed models. Further references can be found in Pratesi (2016). A common feature of the above cited references is the use of predictors based on generalized linear mixed models (GLMM).

This paper extends the EBP methodology of Molina and Rao (2010) by introducing predictors of additive parameters based on unit-level GLMMs. The introduced methodology is applied to the prediction of small area average incomes and poverty indicators under unit-level gamma mixed models (GMM). The GLMMs have random effects taking into account the between-domains variability that is not explained by the auxiliary variables. The random effects are usually assumed to be normally distributed. The maximum likelihood (ML) estimation of GLMM parameters have some computational difficulties because the likelihood may involve high-dimensional integrals which cannot be evaluated analytically. For calculating the ML estimators of model parameters, this paper maximizes the Laplace approximation to the log-likelihood.

The paper introduces EBPs for estimating domain additive parameters. The proposed EBPs are based on unit-level GLMMs. Two more predictors, called plug-in and marginal, are also considered and empirically studied in a simulation experiment.

The mean squared error (MSE), also called prediction variance in the model-based approach to SAE, is a standard accuracy measure for predictors of domain parameters. Hall and Maiti (2006a,b) introduced bootstrap estimators of MSEs of predictors of functions of fixed and random effects under SAE models. As we are interested in estimating small area additive parameters, we consider the parametric bootstrap estimator of the MSE introduced by González-Manteiga et al. (2007), but adapted to GLMMs. This approach was extended by González-Manteiga et al. (2008a,b) to nested error regression models and to multivariate area-level models respectively.

In the particular case of GMMs, we carry out simulation experiments for investigating the behaviour of the fitting method, the predictors of average incomes and poverty proportions and the parametric bootstrap estimator of the MSE. We present an application to data from the Spanish living conditions survey (SLCS) of 2013 in the region of Valencia (east of Spain). The target is the estimation of 2013 average incomes and poverty proportions at county level.

The extension of the methodology of Molina and Rao (2010), where the EBPs are introduced under unit-level LMMs, to unit-level GLMMs have three main mathematical and computational difficulties: (1) under LMMs, the distribution of the unobserved part of the vector of target variables conditioned to the observed part can be calculated explicitly, but not in the case of GLMMs; (2) the likelihoods of GLMMs are high dimensional integrals, so they need more specialized fitting algorithms; (3) it can not be assumed that the shape parameters (or shape function) of GLMMs are all equal to a known common constant, so a procedure for estimating them is needed. This paper faces these three issues by studying the applicability of two unit-level GLMMs to the estimation of small area additive parameters.

The paper is organized as follows. Section 2 introduces two unit-level GLMMs. As the shape parameters of the second model are known constants multiplied by a common parameter, this model cannot be fitted by using standard software; for example by using the glmer function of lme4 library of the R programming language (R Core Team 2019). This is why Section 3 describes the employed ML-Laplace algorithm that we have programmed in R for fitting the model. Sections 4, 5, 5.1 and 5.2 present the empirical best, the marginal and the plug-in predictors of functions of model effects, additive parameters, means and poverty proportions respectively. The calculation of the EBPs uses a census file as auxiliary information. It is shown that this restriction can be avoided if the auxiliary variables are categorical. In that case, it is sufficient to have the population sizes of the domains crossed with the categories. Section 6 gives a parametric bootstrap method for estimating the MSE. Section 7 presents three simulation experiments. Simulation 1 analyses the behaviour of the fitting algorithm. Simulation 2 compares the performances of the three introduced predictors. Simulation 3 empirically studies the parametric bootstrap estimators of the MSEs. Section 8 applies the developed methodology to unit-level data from the 2013 SLCS and takes the aggregated auxiliary information from the Spanish Labour Force Survey (SLFS). The target is the estimation of 2013 average incomes and poverty proportions at county level. Section 9 gives some conclusions. The paper contains two appendices. Appendix A gives the components of the updating equation of the ML-Laplace algorithm for the GMM. Appendix B presents some complementary tables and figures for the application to real data.

2 The unit-level generalized linear mixed models

This section introduces two unit-level GLMMs. Let D denote the number of small areas (or domains) under consideration. Both models have a set of random area effects $\{v_d:\ d=1,\dots,D\}$ that are i.i.d. $N(0,1)$. In matrix notation, we have $\boldsymbol{v}=\underset{1\le d\le D}{\operatorname{col}}(v_d)\sim N_D(\boldsymbol{0},\boldsymbol{I}_D)$, i.e.

$$f_v(\boldsymbol{v})=(2\pi)^{-D/2}\exp\Big\{-\frac{1}{2}\boldsymbol{v}^{\intercal}\boldsymbol{v}\Big\}.$$

For $d=1,\dots,D$, $j=1,\dots,n_d$, the GLMMs assume that the conditional distribution of the target variable y_{dj}, conditioned to v_d, belongs to the exponential family, i.e. $y_{dj}|_{v_d}\sim \text{Exp}(\theta_{dj},\nu_{dj};a,b,c)$, with the probability density function (p.d.f.)

$$f(y_{dj}|v_d)=f(y_{dj}|\theta_{dj},\nu_{dj};a,b,c)=\exp\Big\{\frac{y_{dj}\theta_{dj}-b(\theta_{dj})}{a(\nu_{dj})}+c(y_{dj},\nu_{dj})\Big\}, \qquad (1)$$

where $a(\cdot)>0$, $b(\cdot)$ and $c(\cdot)$ are known real-valued functions specifying the selected distribution and $\nu_{dj}>0$. Further, we assume that $b(\cdot)$ is one-to-one and three times continuously differentiable with one-to-one first derivative. This is to say, we consider a nested data structure where subindexes d and j denote domain and unit (within domain) respectively and n_d is the sample size of domain d. Under (1), the expectation and variance of y_{dj}, given v_d, are

$$\mu_{dj}=E[y_{dj}|v_d]=\frac{\partial b(\theta_{dj})}{\partial\theta_{dj}}=\dot{b}(\theta_{dj}) \quad \text{var}[y_{dj}|v_d]=a(\nu_{dj})\frac{\partial^2 b(\theta_{dj})}{\partial\theta_{dj}^2}=a(\nu_{dj})\ddot{b}(\theta_{dj}).$$

Parameters μ_{dj} and ν_{dj} are called mean and shape parameters respectively. For a twice continuously differentiable and monotonous link function $g(\cdot)$ of the mean parameter, we assume that

$$\eta_{dj}=g(\mu_{dj})=\boldsymbol{x}_{dj}^{\intercal}\boldsymbol{\beta}+\phi v_d, \quad d=1,\dots,D,\ j=1,\dots,n_d,$$

where $\phi>0$ is a standard deviation parameter, $\boldsymbol{\beta}=\underset{1\le k\le p}{\operatorname{col}}(\beta_k)$ is a vector of regression parameters and $\boldsymbol{x}_{dj}=\underset{1\le k\le p}{\operatorname{col}}(x_{djk})$ is a vector of auxiliary variables which are assumed to be constant (fixed regression design). Further, we assume that the y_{dj}'s are independent conditioned to $\boldsymbol{v}$. The sample size is $n=\sum_{d=1}^{D} n_d$ and the domain target vector is $\boldsymbol{y}_d = \underset{1\le j\le n_d}{\text{col}}(y_{dj})$. The conditional p.d.f. of $\boldsymbol{y}=\underset{1\le d\le D}{\text{col}}(\boldsymbol{y}_d)$, given $\boldsymbol{v}$, and the marginal p.d.f. of $\boldsymbol{y}$ are

$$f(\boldsymbol{y}|\boldsymbol{v})=\prod_{d=1}^{D}\prod_{j=1}^{n_d} f(y_{dj}|v_d),\quad f(\boldsymbol{y})=\int_{R^D} f(\boldsymbol{y}|\boldsymbol{v})f_v(\boldsymbol{v})\,d\boldsymbol{v}.$$

Let us note that the assumption of normality of the random effects is typical for mixed models used in SAE. Sinha and Rao (2009) and Benavent and Morales (2016) carried out simulation experiments to investigate the robustness of EBLUPs of linear parameters against deviations from the hypothesis of normality under nested error regression and Fay-Herriot models respectively. They showed that EBLUPs works well when deviations are small, but their behaviour become poor when deviations are big. Similar conclusions hold also for EBP under the presented model. A specific comment concerning this issue is given in Remark 8.1 in Section 8.

An example of unit-level GLMM is the GMM, where

$$y_{dj}|v_d \sim \text{Gamma}(\nu_{dj},\,\mu_{dj}/\nu_{dj}),\quad d=1,\ldots,D,\ j=1,\ldots,n_d.$$

For $y_{dj}>0$, the conditioned p.d.f. is

$$f(y_{dj}|v_d)=\Big(\frac{\nu_{dj}}{\mu_{dj}}\Big)^{\nu_{dj}}\frac{y_{dj}^{\nu_{dj}-1}}{\Gamma(\nu_{dj})}\exp\Big\{-\frac{\nu_{dj}}{\mu_{dj}}y_{dj}\Big\} \tag{2}$$

$$=\exp\left\{\frac{y_{dj}\Big(-\frac{1}{\mu_{dj}}\Big)-\log\mu_{dj}}{\frac{1}{\nu_{dj}}}+\nu_{dj}\log\nu_{dj}-\log\Gamma(\nu_{dj})+(\nu_{dj}-1)\log y_{dj}\right\}.$$

Under (2), the expectation and variance of y_{dj}, given v_d, are

$$E[y_{dj}|v_d]=\frac{\nu_{dj}}{\nu_{dj}/\mu_{dj}}=\mu_{dj},\quad \text{var}[y_{dj}|v_d]=\frac{\nu_{dj}}{\nu_{dj}^2/\mu_{dj}^2}=\frac{\mu_{dj}^2}{\nu_{dj}}.$$

The natural parameter and the functions $a(\cdot)>0$, $b(\cdot)$ and $c(\cdot)$ of GMMs are

$$\theta_{dj}=-\tfrac{1}{\mu_{dj}},\qquad b(\theta_{dj})=\log\mu_{dj}=\log\Big(-\tfrac{1}{\theta_{dj}}\Big)=-\log(-\theta_{dj}),$$
$$a(\nu_{dj})=1/\nu_{dj},\quad c(y_{dj},\nu_{dj})=\nu_{dj}\log\nu_{dj}-\log\Gamma(\nu_{dj})+(\nu_{dj}-1)\log y_{dj}.$$

For the mean parameter in GMMs, we consider the link function

$$\eta_{dj}=g(\mu_{dj})=\frac{1}{\mu_{dj}}=\boldsymbol{x}_{dj}^{\intercal}\boldsymbol{\beta}+\phi v_d,\quad d=1,\ldots,D. \tag{3}$$

Depending on the assumptions on the shape parameters, we consider two GLMMs. Model 1 assumes that $\nu_{dj} = \nu > 0$, $d = 1,\dots,D$, $j = 1,\dots,n_d$ and ν is unknown. Model 2 assumes that $\nu_{dj} = a_{dj}\varphi$ with $a_{dj} > 0$ known and $\varphi > 0$ unknown, $d = 1,\dots,D$, $j = 1,\dots,n_d$. This is to say, Model 1 is Model 2 with $a_{dj} = 1$ and $\nu = \varphi > 0$ unknown. Under the gamma distribution (2) with the link function (3), these models are called gamma Model 1 and 2 respectively. For some distributions of the exponential family, Model 1 can be fitted with the glmer function of lme4 library of the R programming language. However, glmer cannot be applied to estimate the parameters of Model 2. Section 3 presents the ML-Laplace algorithm for fitting GLMMs, with a particularization to gamma Model 2.

Model 1 is quite popular in Gamma regression modelling. Under Model 1, the conditioned variance is $\text{var}[y_{dj}|v_d] = \nu^{-1}\mu_{dj}^2$. The direct proportionality to the mean is a rigid condition that sometimes does not allow a good fit of the model to the data. This fact was observed in the application to real data and motivated the use of Model 2. Under Model 2, a good selection of a_{dj} for the conditioned variance will produce a better fit of the GMM to the data. This is illustrated in Section 8.

Alternative link functions for gamma regression models are $g(\mu_{dj}) = \mu_{dj}$ and $g(\mu_{dj}) = \log\mu_{dj}$. The link function (3) allows giving linear predictors of the natural parameter θ_{dj} and moreover it is the canonical link function for the Gamma distribution which implies some good properties of the ML estimators. That is why we investigate GMMs with the inverse link function in the simulations and we use it in the application to real data.

3 The Laplace approximation algorithm

This section describes an approximation of the loglikelihood of GLMMs and the corresponding algorithm for estimating the unknown parameters of Model 2. In what follows $\psi^{-1}(\cdot)$ denotes the inverse mapping of a one-to-one real valued function $\psi(\cdot)$. As $\mu_{dj} = g^{-1}(\eta_{dj})$ and $\theta_{dj} = (\dot{b})^{-1}(\mu_{dj})$, it holds that

$$\frac{\partial\mu_{dj}}{\partial\eta_{dj}} = \frac{1}{\frac{\partial\eta_{dj}}{\partial\mu_{dj}}} = \frac{1}{\dot{g}(\mu_{dj})}, \quad \frac{\partial\theta_{dj}}{\partial\mu_{dj}} = \frac{1}{\frac{\partial\mu_{dj}}{\partial\theta_{dj}}} = \frac{1}{\ddot{b}(\theta_{dj})}, \quad \frac{\partial\eta_{dj}}{\partial v_d} = \frac{\partial(\boldsymbol{x}_{dj}^{\intercal}\boldsymbol{\beta} + \phi v_d)}{\partial v_d} = \phi.$$

Therefore

$$\frac{\partial\mu_{dj}}{\partial v_d} = \frac{\partial\mu_{dj}}{\partial\eta_{dj}}\frac{\partial\eta_{dj}}{\partial v_d} = \frac{\phi}{\dot{g}(\mu_{dj})}, \quad \frac{\partial\dot{g}(\mu_{dj})}{\partial v_d} = \frac{\partial\dot{g}(\mu_{dj})}{\partial\mu_{dj}}\frac{\partial\mu_{dj}}{\partial v_d} = \ddot{g}(\mu_{dj})\frac{\phi}{\dot{g}(\mu_{dj})},$$

$$\frac{\partial\theta_{dj}}{\partial v_d} = \frac{\partial\theta_{dj}}{\partial\mu_{dj}}\frac{\partial\mu_{dj}}{\partial v_d} = \frac{\phi}{\ddot{b}(\theta_{dj})\dot{g}(\mu_{dj})}, \quad \frac{\partial b(\theta_{dj})}{\partial v_d} = \frac{\partial b(\theta_{dj})}{\partial\theta_{dj}}\frac{\partial\theta_{dj}}{\partial v_d} = \frac{\phi\dot{b}(\theta_{dj})}{\ddot{b}(\theta_{dj})\dot{g}(\mu_{dj})}.$$

The vectors $\boldsymbol{y}_1,\dots,\boldsymbol{y}_D$ are unconditionally independent with marginal p.d.f.

$$f(\boldsymbol{y}_d) = \int_{-\infty}^{\infty} \prod_{j=1}^{n_d} f(y_{dj}|v_d) f(v_d)\, dv_d$$

$$= \kappa_d \int_{-\infty}^{\infty} \exp\Big\{ -\frac{v_d^2}{2} + \sum_{j=1}^{n_d} \frac{y_{dj}\theta_{dj} - b(\theta_{dj})}{a(\nu_{dj})} \Big\}\, dv_d = \kappa_d \int_{-\infty}^{\infty} \exp\big\{h(v_d)\big\}\, dv_d,$$

where $\kappa_d = (2\pi)^{-1/2} \exp\Big\{ \sum_{j=1}^{n_d} c(y_{dj},\nu_{dj})\Big\}$,

$$h(v_d) = -\frac{v_d^2}{2} + \sum_{j=1}^{n_d} \frac{y_{dj}\theta_{dj} - b(\theta_{dj})}{a(\nu_{dj})}, \tag{4}$$

$$\begin{aligned}
\dot{h}(v_d) &= -v_d + \sum_{j=1}^{n_d} \frac{1}{a(\nu_{dj})} \Big\{ \frac{\phi y_{dj}}{\ddot{b}(\theta_{dj})\dot{g}(\mu_{dj})} - \frac{\phi \dot{b}(\theta_{dj})}{\ddot{b}(\theta_{dj})\dot{g}(\mu_{dj})} \Big\}, \\
&= -v_d + \phi \sum_{j=1}^{n_d} \frac{1}{a(\nu_{dj})\ddot{b}(\theta_{dj})\dot{g}(\mu_{dj})} \big(y_{dj} - \mu_{dj}\big),
\end{aligned}$$

and

$$\begin{aligned}
&\ddot{h}(v_d) = -1 + \phi^2 \sum_{j=1}^{n_d} \frac{1}{a(\nu_{dj})\ddot{b}^2(\theta_{dj})\dot{g}^2(\mu_{dj})} \\
&\cdot \left\{ -\frac{\ddot{b}(\theta_{dj})\dot{g}(\mu_{dj})}{\dot{g}(\mu_{dj})} - (y_{dj} - \mu_{dj}) \left[\frac{\dddot{b}(\theta_{dj})}{\ddot{b}(\theta_{dj})\dot{g}(\mu_{dj})} \dot{g}(\mu_{dj}) + \ddot{b}(\theta_{dj}) \frac{\ddot{g}(\mu_{dj})}{\dot{g}(\mu_{dj})} \right] \right\} \\
&= -1 - \phi^2 \sum_{j=1}^{n_d} \frac{1}{a(\nu_{dj})\ddot{b}^2(\theta_{dj})\dot{g}^2(\mu_{dj})} \left\{ \ddot{b}(\theta_{dj}) + (y_{dj} - \mu_{dj}) \left[\frac{\dddot{b}(\theta_{dj})}{\ddot{b}(\theta_{dj})} + \ddot{b}(\theta_{dj}) \frac{\ddot{g}(\mu_{dj})}{\dot{g}(\mu_{dj})} \right] \right\}.
\end{aligned}$$

The Laplace algorithm looks for v_{0d} maximizing the function $h(v_d)$, i.e. such that $\dot{h}(v_{0d}) = 0$ and $\ddot{h}(v_{0d}) < 0$. The Laplace approximation to $f(\boldsymbol{y}_d)$ is

$$\begin{aligned}
f(\boldsymbol{y}_d) \approx &\Bigg| 1 + \phi^2 \sum_{j=1}^{n_d} \frac{1}{a(\nu_{dj})\ddot{b}^2(\theta_{0dj})\dot{g}^2(\mu_{0dj})} \Big\{ \ddot{b}(\theta_{0dj}) + (y_{dj} - \mu_{0dj}) \Big[\frac{\dddot{b}(\theta_{0dj})}{\ddot{b}(\theta_{0dj})} \\
&+ \ddot{b}(\theta_{0dj}) \frac{\ddot{g}(\mu_{0dj})}{\dot{g}(\mu_{0dj})} \Big] \Big\} \Bigg|^{-1/2} \cdot \exp\Big\{ -\frac{v_{0d}^2}{2} + \sum_{j=1}^{n_d} \frac{y_{dj}\theta_{0dj} - b(\theta_{0dj})}{a(\nu_{dj})} \Big\} \exp\Big\{ \sum_{j=1}^{n_d} c(y_{dj},\nu_{dj}) \Big\},
\end{aligned}$$

where $\theta_{0dj}=(\dot{b})^{-1}(\mu_{0dj})$, $\mu_{0dj}=g^{-1}(\boldsymbol{x}_{dj}^{\intercal}\boldsymbol{\beta}+\phi v_{0d})$. The GLMM loglikelihood is $\ell=\sum_{d=1}^{D}\ell_d$ and

$$\ell_d=\log f(\boldsymbol{y}_d)\approx\ell_{0d}=\sum_{j=1}^{n_d}c(y_{dj},\nu_{dj})-\frac{1}{2}\log\xi_{0d}-\frac{v_{0d}^2}{2}+\sum_{j=1}^{n_d}\frac{y_{dj}\theta_{0dj}-b(\theta_{0dj})}{a(\nu_{dj})}, \tag{5}$$

where

$$\begin{aligned}\xi_{0d}=\Bigg|1+\phi^2\sum_{j=1}^{n_d}\frac{1}{a(\nu_{dj})}\Bigg\{&\frac{1}{\ddot{b}(\theta_{0dj})\dot{g}^2(\mu_{0dj})}\\ &+(y_{dj}-\mu_{0dj})\left[\frac{\dddot{b}(\theta_{0dj})}{\ddot{b}^3(\theta_{0dj})\dot{g}^2(\mu_{0dj})}+\frac{\ddot{g}(\mu_{0dj})}{\ddot{b}(\theta_{0dj})\dot{g}^3(\mu_{0dj})}\right]\Bigg\}\Bigg|.\end{aligned}$$

For the GMMs, we have $\theta_{dj}=-\frac{1}{\mu_{dj}}$, $g(\mu_{dj})=\frac{1}{\mu_{dj}}$, $\dot{g}(\mu_{dj})=-\frac{1}{\mu_{dj}^2}$, $\ddot{g}(\mu_{dj})=\frac{2}{\mu_{dj}^3}$, $b(\theta_{dj})=-\log(-\theta_{dj})$, $\dot{b}(\theta_{dj})=-\frac{1}{\theta_{dj}}=\mu_{dj}$, $\ddot{b}(\theta_{dj})=\frac{1}{\theta_{dj}^2}=\mu_{dj}^2$, $\dddot{b}(\theta_{dj})=-\frac{2}{\theta_{dj}^3}=2\mu_{dj}^3$,

$$h(v_d)=-\frac{v_d^2}{2}+\sum_{j=1}^{n_d}\left\{\nu_{dj}\log(\boldsymbol{x}_{dj}^{\intercal}\boldsymbol{\beta}+\phi v_d)-\nu_{dj}y_{dj}(\boldsymbol{x}_{dj}^{\intercal}\boldsymbol{\beta}+\phi v_d)\right\},$$

$$\begin{aligned}\dot{h}(v_d)&=-v_d+\sum_{j=1}^{n_d}\left\{\frac{\nu_{dj}\phi}{\boldsymbol{x}_{dj}^{\intercal}\boldsymbol{\beta}+\phi v_d}-\phi\nu_{dj}y_{dj}\right\}=-v_d+\phi\sum_{j=1}^{n_d}\nu_{dj}(\mu_{dj}-y_{dj}),\\ \ddot{h}(v_d)&=-\Big(1+\phi^2\sum_{j=1}^{n_d}\frac{\nu_{dj}}{(\boldsymbol{x}_{dj}^{\intercal}\boldsymbol{\beta}+\phi v_d)^2}\Big)=-\Big(1+\phi^2\sum_{j=1}^{n_d}\nu_{dj}\mu_{dj}^2\Big).\end{aligned}$$

In this particular case, it holds that $\ddot{h}(v_d)<0$ for all possible values of v_d. The components of the Laplace approximation to the GMM loglikelihood are

$$\begin{aligned}\ell_{0d}=&\sum_{j=1}^{n_d}\Big\{\nu_{dj}\log\nu_{dj}+(\nu_{dj}-1)\log y_{dj}-\log\Gamma(\nu_{dj})\Big\}-\frac{1}{2}\log\xi_{0d}-\frac{v_{0d}^2}{2}\\ &+\sum_{j=1}^{n_d}\left\{\nu_{dj}\log(\boldsymbol{x}_{dj}^{\intercal}\boldsymbol{\beta}+\phi v_{0d})-\nu_{dj}y_{dj}(\boldsymbol{x}_{dj}^{\intercal}\boldsymbol{\beta}+\phi v_{0d})\right\},\end{aligned} \tag{6}$$

where $\xi_{0d}=1+\phi^2\sum_{j=1}^{n_d}\nu_{dj}\mu_{0dj}^2$ and $\mu_{0dj}=(\boldsymbol{x}_{dj}^{\intercal}\boldsymbol{\beta}+\phi v_{0d})^{-1}$. Under gamma Model 2, i.e. under the assumption $\nu_{dj}=a_{dj}\varphi$, Appendix A gives the partial derivatives of ℓ_{0d} with respect to the components of $\boldsymbol{\theta}=(\boldsymbol{\beta}^{\intercal},\phi,\varphi)^{\intercal}$. It also gives the score vector $\boldsymbol{U}_0(\boldsymbol{\theta})$ and the Hessian matrix $\boldsymbol{H}_0(\boldsymbol{\theta})$ containing the first and the second partial derivatives of ℓ_{0d} respectively.

A first Newton-Raphson algorithm maximizes $\ell_0(\boldsymbol{\theta}) = \sum_{d=1}^{D} \ell_{0d}$, with fixed $v_d = v_{0d}$, $d = 1,\dots,D$. The updating equation is

$$\boldsymbol{\theta}^{(k+1)} = \boldsymbol{\theta}^{(k)} - \boldsymbol{H}_0^{-1}(\boldsymbol{\theta}^{(k)})\boldsymbol{U}_0(\boldsymbol{\theta}^{(k)}). \tag{7}$$

For $d = 1,\dots,D$, a second Newton-Raphson algorithm maximizes $h(v_d) = h(v_d,\boldsymbol{\theta})$, defined in (4), with $\boldsymbol{\theta} = (\boldsymbol{\beta}^{\intercal}, \phi, \varphi)^{\intercal} = \boldsymbol{\theta}_0$ fixed. The updating equation is

$$v_d^{(k+1)} = v_d^{(k)} - \frac{\dot{h}(v_d^{(k)},\boldsymbol{\theta}_0)}{\ddot{h}(v_d^{(k)},\boldsymbol{\theta}_0)}. \tag{8}$$

Algorithm. By combining the two Newton-Raphson algorithms, the ML-Laplace approximation algorithm for Model 2 is obtained. The steps are

1. Set the initial values $i = 0$, $\varepsilon_1 > 0$, $\varepsilon_2 > 0$, $\varepsilon_3 > 0$, $\varepsilon_4 > 0$, $\boldsymbol{\theta}^{(0)}$, $\boldsymbol{\theta}^{(-1)} = \boldsymbol{\theta}^{(0)} + \mathbf{1}$, $v_d^{(0)} = 0$, $v_d^{(-1)} = 1$, $d = 1,\dots,D$.
2. Until $|v_d^{(i)} - v_d^{(i-1)}| < \varepsilon_3$, $d = 1,\dots,D$, $\|\boldsymbol{\theta}^{(i)} - \boldsymbol{\theta}^{(i-1)}\|_2 < \varepsilon_4$, do
   (a) Apply algorithm (8) with seeds $v_d^{(i)}$, $d = 1,\dots,D$, convergence tolerance ε_1 and $\boldsymbol{\theta} = \boldsymbol{\theta}^{(i)}$ fixed. Output: $v_d^{(i+1)}$, $d = 1,\dots,D$.
   (b) Apply algorithm (7) with seed $\boldsymbol{\theta}^{(i)}$, convergence tolerance ε_2 and $v_{0d} = v_d^{(i+1)}$ fixed, $d = 1,\dots,D$. Output: $\boldsymbol{\theta}^{(i+1)}$.
   (c) $i \leftarrow i + 1$.
3. Output: $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}^{(i)}$, $\hat{v}_d = v_d^{(i)}$, $d = 1,\dots,D$.

To get some algorithm seed $\boldsymbol{\theta}^{(0)}$, we can e.g. fit Model 1 (this can be done for several distributions from the exponential family by using the glmer function of the R statistical package lme4) to obtain the estimates $\tilde{\boldsymbol{\beta}}$, $\tilde{\phi}$ and $\tilde{\nu}$ and use the seeds $\boldsymbol{\beta}^{(0)} = \tilde{\boldsymbol{\beta}}$, $\phi^{(0)} = \tilde{\phi}$ and $\varphi^{(0)} = \tilde{\nu}/\overline{a}_d$, where $\overline{a}_d = 1/n_d \sum_{j=1}^{n_d} a_{dj}$. Let us also note that the Laplace approximation algorithm gives at convergence not only estimators of the model parameters but also the mode predictors, $\hat{v}_d$, of the random effects.

4 Predictors of functions of model effects

This section considers a finite population U of N elements partitioned into D domains U_d of size N_d, $d = 1,\dots,D$. From the population, a sample s of size n is selected with subsamples s_d of sizes n_d from domains U_d. Let $\boldsymbol{y}_d = \underset{1\le j\le N_d}{\text{col}}(y_{dj})$ be the random vector containing the values of a target variable on the N_d units of domain d. Let $\boldsymbol{y}_{ds}$ be the sub-vector of $\boldsymbol{y}_d$ corresponding to the units in the sample s_d and $\boldsymbol{y}_{dr}$ the sub-vector of domain units in the non-sampled domain population $U_d - s_d$. By reordering the domain units, we can write $\boldsymbol{y}_d = (\boldsymbol{y}_{ds}^\top, \boldsymbol{y}_{dr}^\top)^\top$. We define $\boldsymbol{y}_s = \underset{1\leq d\leq D}{\text{col}}(\boldsymbol{y}_{ds})$ and $\boldsymbol{y}_r = \underset{1\leq d\leq D}{\text{col}}(\boldsymbol{y}_{dr})$ and we follow a fully model-based approach by assuming that $\boldsymbol{y} = (\boldsymbol{y}_s^\top, \boldsymbol{y}_r^\top)^\top$ follows Model 2. We are thus assuming a non-informative sampling setup.

The conditional distribution of $\boldsymbol{y}_s$, given $\boldsymbol{v}$, is

$$f(\boldsymbol{y}_s|\boldsymbol{v}) = \prod_{d=1}^{D} f(\boldsymbol{y}_{ds}|v_d),$$

where

$$f(\boldsymbol{y}_{ds}|v_d) = \exp\Big\{\sum_{j=1}^{n_d} \frac{y_{dj}\theta_{dj} - b(\theta_{dj})}{a(\nu_{dj})}\Big\} \exp\Big\{\sum_{j=1}^{n_d} c(y_{dj}, \nu_{dj})\Big\}$$

and the p.d.f. of $\boldsymbol{v}$ is

$$f(\boldsymbol{v}) = \prod_{d=1}^{D} f(v_d), \quad f(v_d) = (2\pi)^{-1/2} \exp\Big\{-\frac{1}{2}v_d^2\Big\}.$$

The conditional distribution of $\boldsymbol{y}_{dr}$, given $\boldsymbol{y}_{ds}$, is

$$f(\boldsymbol{y}_{dr}|\boldsymbol{y}_{ds}) = \frac{f(\boldsymbol{y}_{dr}, \boldsymbol{y}_{ds})}{f(\boldsymbol{y}_{ds})} = \frac{\int_R f(\boldsymbol{y}_{dr}, \boldsymbol{y}_{ds}|v_d) f(v_d)\, dv_d}{\int_R f(\boldsymbol{y}_{ds}|v_d) f(v_d)\, dv_d}$$
$$= \frac{\int_R f(\boldsymbol{y}_{dr}|v_d) f(\boldsymbol{y}_{ds}|v_d) f(v_d)\, dv_d}{\int_R f(\boldsymbol{y}_{ds}|v_d) f(v_d)\, dv_d}. \tag{9}$$

The aim of this section is to introduce the EBP and the plug-in predictor of μ_{dj} and $\bar{\mu}_d = \frac{1}{N_d}\sum_{j=1}^{N_d} \mu_{dj}$ under Model 2. The corresponding predictors under Model 1 can be obtained in a similar way.

If $\boldsymbol{\theta} = (\boldsymbol{\beta}^\top, \phi, \varphi)^\top$ is known, the best predictor of $\mu_{dj} = \mu_{dj}(\boldsymbol{\theta}, v_d) = g^{-1}(\boldsymbol{x}_{dj}^\top \boldsymbol{\beta} + \phi v_d)$ is $\hat{\mu}_{dj}(\boldsymbol{\theta}) = E_\theta[\mu_{dj}|\boldsymbol{y}_s]$. In this case, we have that $E_\theta[\mu_{dj}|\boldsymbol{y}_s] = E_\theta[\mu_{dj}|\boldsymbol{y}_{ds}]$ and

$$\hat{\mu}_{dj}(\boldsymbol{\theta}) = E_\theta[\mu_{dj}|\boldsymbol{y}_{ds}] = \frac{\int_R g^{-1}(\boldsymbol{x}_{dj}^\top \boldsymbol{\beta} + \phi v_d) f(\boldsymbol{y}_{ds}|v_d) f(v_d)\, dv_d}{\int_R f(\boldsymbol{y}_{ds}|v_d) f(v_d)\, dv_d} = \frac{A_{dj}(\boldsymbol{y}_{ds}, \boldsymbol{\theta})}{B_d(\boldsymbol{y}_{ds}, \boldsymbol{\theta})},$$

where $A_{dj} = A_{dj}(\boldsymbol{y}_{ds}, \boldsymbol{\theta})$ and $B_d = B_d(\boldsymbol{y}_{ds}, \boldsymbol{\theta})$ are

$$A_{dj} = \int_R g^{-1}(\boldsymbol{x}_{dj}^{\intercal}\boldsymbol{\beta} + \phi v_d) \exp\Big\{\sum_{i=1}^{n_d} \frac{y_{di}\theta_{di} - b(\theta_{di})}{a(\nu_{di})}\Big\} f(v_d)\, dv_d,$$
$$B_d = \int_R \exp\Big\{\sum_{i=1}^{n_d} \frac{y_{di}\theta_{di} - b(\theta_{di})}{a(\nu_{di})}\Big\} f(v_d)\, dv_d. \tag{10}$$

The EBP of μ_{dj} is $\hat{\mu}_{dj}(\hat{\boldsymbol{\theta}})$ and can be approximated by the following Monte Carlo procedure.

1. Estimate $\hat{\boldsymbol{\theta}} = (\hat{\boldsymbol{\beta}}^{\intercal}, \hat{\phi}, \hat{\varphi})^{\intercal}$ and put $\hat{\nu}_{dj} = a_{dj}\hat{\varphi}$.
2. For $\ell = 1, \ldots, L$, generate $v_d^{(\ell)}$ i.i.d. $N(0,1)$ and $v_d^{(L+\ell)} = -v_d^{(\ell)}$.
3. Calculate the approximation of the EBP $\hat{\mu}_{dj} = \hat{A}_{dj}/\hat{B}_d$, where

$$\hat{A}_{dj} = \frac{1}{2L}\sum_{\ell=1}^{2L} g^{-1}(\boldsymbol{x}_{dj}\hat{\boldsymbol{\beta}} + \hat{\phi} v_d^{(\ell)}) \exp\Big\{\sum_{i=1}^{n_d} \frac{y_{di}\hat{\theta}_{di}^{(\ell)} - b(\hat{\theta}_{di}^{(\ell)})}{a(\hat{\nu}_{di})}\Big\},$$
$$\hat{B}_d = \frac{1}{2L}\sum_{\ell=1}^{2L} \exp\Big\{\sum_{i=1}^{n_d} \frac{y_{di}\hat{\theta}_{di}^{(\ell)} - b(\hat{\theta}_{di}^{(\ell)})}{a(\hat{\nu}_{di})}\Big\} \tag{11}$$

and $\hat{\theta}_{di}^{(\ell)} = (\dot{b})^{-1}(\hat{\mu}_{di}^{(\ell)})$ for $\hat{\mu}_{di}^{(\ell)} = g^{-1}(\boldsymbol{x}_{di}^{\intercal}\hat{\boldsymbol{\beta}} + \hat{\phi} v_d^{(\ell)})$.

The derived best predictors have minimum MSE in the class of unbiased estimators. Unfortunately, this property does not hold for EBPs which are obtained by substituting the true parameters by their estimates and therefore they are not unbiased. The EBPs are asymptotically unbiased under the assumption that the estimates of the model parameters are consistent but the domain sample sizes are usually small in SAE problems. Thus, it make sense to empirically investigate the behaviour of the plug-in predictors which are less computationally demanding. The plug-in predictor of μ_{dj} is

$$\tilde{\mu}_{dj} = g^{-1}(\boldsymbol{x}_{dj}^{\intercal}\hat{\boldsymbol{\beta}} + \hat{\phi}\hat{v}_d), \tag{12}$$

where $\hat{\boldsymbol{\beta}}, \hat{\phi}$ and $\hat{v}_d$ are taken from the output of the ML-Laplace approximation algorithm. The EBP and the plug-in predictor of $\bar{\mu}_d = \frac{1}{N_d}\sum_{j=1}^{N_d}\mu_{dj}$ are

$$\hat{\bar{\mu}}_d^E = \hat{\bar{\mu}}_d^E(\hat{\boldsymbol{\theta}}) = \frac{1}{N_d}\sum_{j=1}^{N_d}\hat{\mu}_{dj}, \quad \hat{\bar{\mu}}_d^P = \frac{1}{N_d}\sum_{j=1}^{N_d}\tilde{\mu}_{dj}. \tag{13}$$

5 Predictors of additive parameters

This section introduces predictors of additive parameters of small areas under Model 2. Similar predictors can be obtained under Model 1. An additive domain parameter is

$$\delta_d = \frac{1}{N_d}\sum_{j=1}^{N_d} h(y_{dj}),$$

where h is a known measurable function. If $j \in s_d$, then $E_\theta[h(y_{dj})|\boldsymbol{y}_{ds}] = h(y_{dj})$. If $j \in U_d - s_d$, then $f(y_{dj}|\boldsymbol{y}_{ds})$ is obtained from (9). Therefore, the best predictor of δ_d is

$$\hat{\delta}_d^B = \frac{1}{N_d}\sum_{j=1}^{N_d} E_\theta\big[h(y_{dj})|\boldsymbol{y}_{ds}\big] = \frac{1}{N_d}\Big\{\sum_{j\in s_d} h(y_{dj}) + \sum_{j\in U_d-s_d} E_\theta\big[h(y_{dj})|\boldsymbol{y}_{ds}\big]\Big\},$$

where

$$E_\theta[h(y_{dj})|\boldsymbol{y}_{ds}] = \frac{\int_R\int_R h(y_{dj}) f(\boldsymbol{y}_{ds}|v_d) f(y_{dj}|v_d) f(v_d)\, dy_{dj}\, dv_d}{\int_R f(\boldsymbol{y}_{ds}|v_d) f(v_d)\, dv_d} = \frac{A_{hdj}(\boldsymbol{y}_{ds},\boldsymbol{\theta})}{B_d(\boldsymbol{y}_{ds},\boldsymbol{\theta})},$$

$$A_{hdj} = \int_R\int_R h(y_{dj}) \exp\Big\{\sum_{i=1}^{n_d} \frac{y_{di}\theta_{di} - b(\theta_{di})}{a(\nu_{di})}\Big\} f(y_{dj}|v_d) f(v_d)\, dy_{dj}\, dv_d$$

and B_d was defined in (10). The EBP of of δ_d is

$$\hat{\delta}_d^E = \frac{1}{N_d}\Big\{\sum_{j\in s_d} h(y_{dj}) + \sum_{j\in U_d-s_d} E_{\hat\theta}\big[h(y_{dj})|\boldsymbol{y}_{ds}\big]\Big\}, \quad E_{\hat\theta}\big[h(y_{dj})|\boldsymbol{y}_{ds}\big] = \frac{A_{hdj}(\boldsymbol{y}_{ds},\hat{\boldsymbol{\theta}})}{B_d(\boldsymbol{y}_{ds},\hat{\boldsymbol{\theta}})}.$$

If $j \in U_d - s_d$, then the numerator and denominator of $E_{\hat\theta}[h(y_{dj})|\boldsymbol{y}_{ds}]$ can be approximated by Monte Carlo simulation. The numerator is a bivariate integral that can be written in the form of two iterative univariate integrals. Therefore, we implement an iterative Monte Carlo algorithm which approximates the inner integral by simulating positive random numbers $y_{dj}^{(\ell_1,\ell_2)}$ from $f(y_{dj}|v_d^{(\ell_1)})$ and approximates the outer integral by simulating random numbers $v_d^{(\ell_1)}$ from $f(v_d)$. Under Model 2, the iterative Monte Carlo algorithm is

1. Estimate $\hat{\boldsymbol{\theta}} = (\hat{\boldsymbol{\beta}}^\intercal, \hat\phi, \hat\varphi)^\intercal$ and $\hat\nu_{dj} = a_{dj}\hat\varphi$.
2. For $\ell_1 = 1,\dots,L_1$, generate $v_d^{(\ell_1)}$ i.i.d. $N(0,1)$ and $v_d^{(L_1+\ell_1)} = -v_d^{(\ell_1)}$, calculate $\hat\mu_{dj}^{(\ell_1)} = g^{-1}(\boldsymbol{x}_{dj}^\intercal\hat{\boldsymbol{\beta}} + \hat\phi v_d^{(\ell_1)})$ and $\hat\theta_{dj}^{(\ell_1)} = (\dot{b})^{-1}(\hat\mu_{dj}^{(\ell_1)})$. For $\ell_1 = 1,\dots,2L_1$, $\ell_2 = 1,\dots,L_2$, generate $y_{dj}^{(\ell_1,\ell_2)} \sim \text{Exp}\big(\hat\theta_{dj}^{(\ell_1)}, \hat\nu_{dj}; a,b,c\big)$.

3. Approximate the EBP $\hat{E}_{\hat{\theta}}[h(y_{dj})|\boldsymbol{y}_{ds}] \approx \hat{A}_{hdj}/\hat{B}_d$, where

$$\hat{A}_{hdj} = \frac{1}{2L_1L_2}\sum_{\ell_1=1}^{2L_1}\exp\Big\{\sum_{i=1}^{n_d}\frac{y_{di}\hat{\theta}_{di}^{(\ell_1)} - b(\hat{\theta}_{di}^{(\ell_1)})}{a(\hat{\nu}_{di})}\Big\}\sum_{\ell_2=1}^{L_2} h(y_{dj}^{(\ell_1,\ell_2)}),$$

$$\hat{B}_d = \frac{1}{2L_1}\sum_{\ell_1=1}^{2L_1}\exp\Big\{\sum_{i=1}^{n_d}\frac{y_{di}\hat{\theta}_{di}^{(\ell_1)} - b(\hat{\theta}_{di}^{(\ell_1)})}{a(\hat{\nu}_{di})}\Big\}. \tag{14}$$

The plug-in predictor of δ_d is

$$\hat{\delta}_d^P = \frac{1}{N_d}\Big\{\sum_{j\in s_d} h(y_{dj}) + \sum_{j\in U_d-s_d} h(\tilde{\mu}_{dj})\Big\}, \quad \tilde{\mu}_{dj} = g^{-1}(\boldsymbol{x}_{dj}^{\intercal}\hat{\boldsymbol{\beta}} + \hat{\phi}\hat{v}_d).$$

Simulation 2 shows that the plug-in predictor does not work well in some situations. For this reason we propose another predictor of the additive domain parameter δ_d. Instead of using the conditional distribution deriving the EBPs, we consider the predicted marginal distribution of y_{dj} with parameters $\hat{\nu}_{dj}$ and $\tilde{\theta}_{dj} = (\dot{b})^{-1}(\tilde{\mu}_{dj})$, where $\tilde{\mu}_d$ is the plug-in predictor of μ_{dj}. This is to say, we consider the p.d.f. $f(y_{dj}|\tilde{\theta}_{dj},\hat{\nu}_{dj};a,b,c)$ from the exponential family. Based on the marginal distribution, we define the marginal predictor of δ_d,

$$\hat{\delta}_d^M = \frac{1}{N_d}\Big(\sum_{j\in s_d} h(y_{dj}) + \sum_{j\in U_d-s_d} \hat{h}_{dj}^M\Big),$$

where

$$\hat{h}_{dj}^M \triangleq E\left[h(y_{dj})|\,\tilde{\theta}_{dj},\hat{\nu}_{dj};a,b,c\right] = \int_R h(y)f(y|\tilde{\theta}_{dj},\hat{\nu}_{dj};a,b,c)\,dy.$$

For calculating the empirical best, the plug-in and the marginal predictors, we need two files: (1) a survey file with the unit-level sample data (main file), and (2) a census file containing the values of the employed explanatory variables in all the population units (auxiliary file). However, not all the values $\boldsymbol{x}_{dj}$, $d = 1,\dots,D$, $j = 1,\dots,N_d$, are available in many practical cases. If in addition some of the auxiliary variables are continuous, the three introduced predictors are not applicable. An important particular case, where these predictors can be calculated under the assumed fixed regression design, is when the number of values of the vector of auxiliary variables is finite and the a_{dj}'s take a common value a_{dk} in the domain d and the covariate class k. In this situation, called "categorical setup", we only need a smaller auxiliary file containing the aggregated (domain-level) values of the explanatory variables. More concretely, the categorical setup is

$$\boldsymbol{x}_{dj} \in \{\boldsymbol{z}_1,\dots,\boldsymbol{z}_K\},\ a_{dj} = a_{dk}\ \text{ if } \boldsymbol{x}_{dj} = \boldsymbol{z}_k, \quad j = 1,\dots,N_d,\ d = 1,\dots,D. \tag{15}$$

Under the categorical setup (15), the EBP of δ_d is

$$\hat{\delta}_d^E = \hat{\delta}_d^E(\hat{\boldsymbol{\theta}}) = \frac{1}{N_d}\left[\sum_{j\in s_d} h(y_{dj}) + \sum_{k=1}^{K}\sum_{i=1}^{N_{dk}-n_{dk}} E_{\hat{\theta}}\big[h(y_{dki})|\boldsymbol{y}_{ds}\big]\right]$$
$$= \frac{1}{N_d}\left[\sum_{j\in s_d} h(y_{dj}) + \sum_{k=1}^{K}(N_{dk}-n_{dk})E_{\hat{\theta}}\big[h(y_{dk})|\boldsymbol{y}_{ds}\big]\right], \qquad (16)$$

where the size N_{dk} of $U_{dk} = \{j \in U_d : \boldsymbol{x}_{dj} = \boldsymbol{z}_k\}$ is available from external data sources (aggregated auxiliary information), n_{dk} is the size of $s_{dk} = \{j \in s_d : \boldsymbol{x}_{dj} = \boldsymbol{z}_k\}$ and y_{dki} denotes the value that the target variable takes in the ith unit of the subset $U_{dk} - s_{dk}$. Under Model 2, the expectation $E_{\hat{\theta}}\big[h(y_{dk})|\boldsymbol{y}_{ds}\big]$ is approximated (similarly to (14)) by Monte Carlo integration as follows.

1. Estimate $\hat{\boldsymbol{\theta}} = (\hat{\boldsymbol{\beta}}^{\intercal}, \hat{\phi}, \hat{\varphi})^{\intercal}$ and put $\hat{\nu}_{dk} = a_{dk}\hat{\varphi}$.
2. For $\ell_1 = 1,\dots,L_1$ generate $v_d^{(\ell_1)}$ i.i.d. $N(0,1)$ and $v_d^{(L_1+\ell_1)} = -v_d^{(\ell_1)}$, calculate $\hat{\mu}_{dk}^{(\ell_1)} = g^{-1}(\boldsymbol{z}_k^{\intercal}\hat{\boldsymbol{\beta}} + \hat{\phi}v_d^{(\ell_1)})$ and $\hat{\theta}_{dk}^{(\ell_1)} = (\dot{b})^{-1}(\hat{\mu}_{dk}^{(\ell_1)})$. For $\ell_1 = 1,\dots,2L_1$, $\ell_2 = 1,\dots,L_2$, generate $y_{dk}^{(\ell_1,\ell_2)} \sim \operatorname{Exp}\big(\hat{\theta}_{dk}^{(\ell_1)}, \hat{\nu}_{dk}; a,b,c\big)$.
3. Calculate $E_{\hat{\theta}}\big[h(y_{dk})|\boldsymbol{y}_{ds}\big] = \hat{A}_{hdk}/\hat{B}_d$, where

$$\hat{A}_{hdk} = \frac{1}{2L_1L_2}\sum_{\ell_1=1}^{2L_1}\exp\Big\{\sum_{i=1}^{n_d}\frac{y_{di}\hat{\theta}_{di}^{(\ell_1)} - b(\hat{\theta}_{di}^{(\ell_1)})}{a(\hat{\nu}_{di})}\Big\}\sum_{\ell_2=1}^{L_2} h(y_{dk}^{(\ell_1,\ell_2)}),$$
$$\hat{B}_d = \frac{1}{2L_1}\sum_{\ell=1}^{2L_1}\exp\Big\{\sum_{i=1}^{n_d}\frac{y_{di}\hat{\theta}_{di}^{(\ell_1)} - b(\hat{\theta}_{di}^{(\ell_1)})}{a(\hat{\nu}_{di})}\Big\}. \qquad (17)$$

If (15) holds, the plug-in predictor of δ_d is

$$\hat{\delta}_d^P = \frac{1}{N_d}\Big\{\sum_{j\in s_d} h(y_{dj}) + \sum_{k=1}^{K}(N_{dk}-n_{dk})h(\tilde{\mu}_{dk})\Big\}, \quad \tilde{\mu}_{dk} = g^{-1}(\boldsymbol{z}_k^{\intercal}\hat{\boldsymbol{\beta}} + \hat{\phi}\hat{v}_d),$$

and the marginal predictor of δ_d is

$$\hat{\delta}_d^M = \frac{1}{N_d}\Big\{\sum_{j\in s_d} h(y_{dj}) + \sum_{k=1}^{K}(N_{dk}-n_{dk})\hat{h}_{dk}^M\Big\}, \quad \hat{h}_{dk}^M = \int_R h(y)f(y|\tilde{\theta}_{dk},\hat{\nu}_{dk};a,b,c)dy.$$

5.1 Predictors of small area means

This section introduces predictors of the small area mean

$$\overline{Y}_d = \frac{1}{N_d} \sum_{j=1}^{N_d} y_{dj},$$

which is an additive parameter with $h(y) = y$. The best predictor of y_{dj} is $\hat{y}_{dj}(\boldsymbol{\theta}) = E_\theta[y_{dj}|\boldsymbol{y}_s]$. If $j \in s_d$, then $E_\theta[y_{dj}|\boldsymbol{y}_s] = y_{dj}$. If $j \in U_d - s_d$, then $E_\theta[y_{dj}|\boldsymbol{y}_s] = E_\theta[y_{dj}|\boldsymbol{y}_{ds}]$ and

$$\hat{y}_{dj}(\boldsymbol{\theta}) = E_\theta[y_{dj}|\boldsymbol{y}_{ds}] = \frac{\int_R \int_R y_{dj} f(y_{dj}|v_d) f(\boldsymbol{y}_{ds}|v_d) f(v_d)\, dy_{dj}\, dv_d}{\int_R f(\boldsymbol{y}_{ds}|v_d) f(v_d)\, dv_d}$$

$$= \frac{\int_R \mu_{dj} f(\boldsymbol{y}_{ds}|v_d) f(v_d)\, dv_d}{\int_R f(\boldsymbol{y}_{ds}|v_d) f(v_d)\, dv_d} = \frac{A_{dj}(\boldsymbol{y}_{ds},\boldsymbol{\theta})}{B_d(\boldsymbol{y}_{ds},\boldsymbol{\theta})} = E_\theta[\mu_{dj}|\boldsymbol{y}_{ds}] = \hat{\mu}_{dj}(\boldsymbol{\theta}),$$

where $A_{dj} = A_{dj}(\boldsymbol{y}_{ds},\boldsymbol{\theta})$ and $B_d = B_d(\boldsymbol{y}_{ds},\boldsymbol{\theta})$ are defined in (10). The EBP of y_{dj} is $\hat{y}_{dj} = \hat{y}_{dj}(\hat{\boldsymbol{\theta}}) = \hat{\mu}_{dj}(\hat{\boldsymbol{\theta}})$. Thus, the EBP of y_{dj} is $\hat{y}_{dj} = y_{dj}$ if $j \in s_d$ and $\hat{y}_{dj} = \hat{\mu}_{dj}$ if $j \in U_d - s_d$, where $\hat{\mu}_{dj}$ is the EBP of μ_{dj} given in (11).

The EBP and the plug-in and marginal predictors of $\overline{Y}_d$ are

$$\hat{\overline{Y}}_d^E = \frac{1}{N_d}\Big[\sum_{j\in s_d} y_{dj} + \sum_{j\in U_d - s_d} \hat{\mu}_{dj}\Big], \hat{\overline{Y}}_d^P = \frac{1}{N_d}\Big[\sum_{j\in s_d} y_{dj} + \sum_{j\in U_d - s_d} \tilde{\mu}_{dj}\Big],$$

$$\hat{\overline{Y}}_d^M = \frac{1}{N_d}\Big[\sum_{j\in s_d} y_{dj} + \sum_{j\in U_d - s_d} \hat{\mu}_{dj}^M\Big],$$

where $\tilde{\mu}_{dj}$ is the plug-in predictor of μ_{dj} given in (12) and

$$\hat{\mu}_{dj}^M \triangleq E\left[y_{dj}|\,\tilde{\theta}_{dj},\hat{\nu}_{dj};a,b,c\right] = \int_R y f(y|\tilde{\theta}_{dj},\hat{\nu}_{dj};a,b,c)\, dy = g^{-1}(\boldsymbol{x}_{dj}^\intercal\hat{\boldsymbol{\beta}} + \hat{\phi}\hat{v}_d) = \tilde{\mu}_{dj},$$

so that $\hat{\overline{Y}}_d^M = \hat{\overline{Y}}_d^P$. Under the categorical setup (15), the EBP and the plug-in predictors of $\overline{Y}_d$ are

$$\hat{\overline{Y}}_d^E = \frac{1}{N_d}\Big[\sum_{j\in s_d} y_{dj} + \sum_{k=1}^K (N_{dk} - n_{dk})\hat{\mu}_{dk}\Big], \quad \hat{\overline{Y}}_d^P = \frac{1}{N_d}\Big[\sum_{j\in s_d} y_{dj} + \sum_{k=1}^K (N_{dk} - n_{dk})\tilde{\mu}_{dk}\Big],$$

where $\tilde{\mu}_{dk} = g^{-1}(\boldsymbol{z}_k^\intercal\hat{\boldsymbol{\beta}} + \hat{\phi}\hat{v}_d)$, $\hat{\mu}_{dk} = \hat{A}_{dk}^z/\hat{B}_d$, $\hat{B}_d$ is defined in (10) and $\hat{A}_{dk}^z$ is the Monte-Carlo approximation of

$$A_{dk}^{z} = \int_R g^{-1}(\boldsymbol{z}_k^{\intercal}\boldsymbol{\beta} + \phi v_d) \exp\Big\{ \sum_{j=1}^{n_d} \frac{y_{dj}\theta_{dj} - b(\theta_{dj})}{a(\nu_{dj})} \Big\} f(v_d)\, dv_d.$$

This is to say, $\hat{A}_{dk}^{z}$ can be calculated as

$$\hat{A}_{dk}^{z} = \frac{1}{2L} \sum_{\ell=1}^{2L} g^{-1}(\boldsymbol{z}_k^{\intercal}\hat{\boldsymbol{\beta}} + \hat{\phi} v_d^{(\ell)}) \exp\Big\{ \sum_{j=1}^{n_d} \frac{y_{dj}\hat{\theta}_{dj}^{(\ell)} - b(\hat{\theta}_{dj}^{(\ell)})}{a(\nu_{dj})} \Big\},$$

where $v_d^{(\ell)}$ are i.i.d. $N(0,1)$ and $v_d^{(L+\ell)} = -v_d^{(\ell)}$, $\ell = 1,\ldots,L$, $\hat{\theta}_{dj}^{(\ell)} = (\dot{b})^{-1}(\hat{\mu}_{dj}^{(\ell)})$ and $\hat{\mu}_{dj}^{(\ell)} = g^{-1}(\boldsymbol{x}_{dj}^{\intercal}\hat{\boldsymbol{\beta}} + \hat{\phi} v_d^{(\ell)})$.

5.2 Predictors of poverty proportions

This section deals with the estimation of domain poverty proportions, which are the proportion of people in the domain whose welfare is below the poverty line. Let y_{dj} be a welfare variable (i.e. income or expenditure) for individual j from domain d and let z be the poverty line. Then, the poverty proportion is the additive parameter

$$p_d = \frac{1}{N_d} \sum_{j=1}^{N_d} h_0(y_{dj}),$$

where $h_0(y_{dj}) = I\big(y_{dj} < z\big)$. The EBP of p_d is

$$\hat{p}_d^{E} = \frac{1}{N_d} \Big(\sum_{j \in s_d} h_0(y_{dj}) + \sum_{j \in U_d - s_d} E_{\hat{\theta}}\left[h_0(y_{dj}) | \boldsymbol{y}_{ds}\right] \Big),$$

where $E_{\hat{\theta}}\left[h_0(y_{dj}) | \boldsymbol{y}_{ds}\right]$ is calculated by applying (14) with $h = h_0$.

The plug-in and the marginal predictors of p_d are

$$\hat{p}_d^{P} = \frac{1}{N_d} \Big(\sum_{j \in s_d} h_0(y_{dj}) + \sum_{j \in U_d - s_d} h_0(\tilde{\mu}_{dj}) \Big), \;\; \hat{p}_d^{M} = \frac{1}{N_d} \Big(\sum_{j \in s_d} h_0(y_{dj}) + \sum_{j \in U_d - s_d} \hat{p}_{dj}^{M} \Big),$$

where $\tilde{\mu}_{dj} = g^{-1}(\boldsymbol{x}_{dj}^{\intercal}\hat{\boldsymbol{\beta}} + \hat{\phi}\hat{v}_d)$ and

$$\begin{aligned} \hat{p}_{dj}^{M} &= E\left[I(y_{dj} < z) | \tilde{\theta}_{dj}, \hat{\nu}_{dj}; a,b,c\right] = P\Big(\text{Exp}(\tilde{\theta}_{dj}, \hat{\nu}_{dj}; a,b,c) < z\Big) \\ &\triangleq F(z | \tilde{\theta}_{dj}, \hat{\nu}_{dj}; a,b,c). \end{aligned}$$

In the previous formula, $F(\cdot)$ denotes the cumulative distribution function of the corresponding distribution from the exponential family.

Under the categorical setup (15), the EBP of p_d is

$$\hat{p}_d^E = \frac{1}{N_d}\left[\sum_{j\in s_d} h_0(y_{dj}) + \sum_{k=1}^{K}(N_{dk}-n_{dk})E_{\hat{\theta}}\left[h_0(y_{dk})|\boldsymbol{y}_{ds}\right]\right],$$

where $E_{\hat{\theta}}\left[h_0(y_{dk})|\boldsymbol{y}_{ds}\right]$ is calculated by applying (17) with $h = h_0$. The marginal and the plug-in predictors of p_d are

$$\hat{p}_d^M = \frac{1}{N_d}\Big[\sum_{j\in s_d} h_0(y_{dj}) + \sum_{k=1}^{K}(N_{dk}-n_{dk})\hat{p}_{dk}^M\Big],$$

$$\hat{p}_d^P = \frac{1}{N_d}\Big[\sum_{j\in s_d} h_0(y_{dj}) + \sum_{k=1}^{K}(N_{dk}-n_{dk})h_0(\tilde{\mu}_{dk})\Big],$$

where $\hat{p}_{dk}^M = F(z|\tilde{\theta}_{dk},\hat{\nu}_{dk};a,b,c)$, $\tilde{\theta}_{dk} = (\dot{b})^{-1}(\tilde{\mu}_{dk})$, $\hat{\nu}_{dk} = a_{dk}\hat{\varphi}$ and $\tilde{\mu}_{dk} = g^{-1}(\boldsymbol{z}_k^{\intercal}\hat{\boldsymbol{\beta}} + \hat{\phi}\hat{v}_d)$.

6 Bootstrap estimation of the MSE

This section presents a parametric bootstrap estimator of the MSE of $\hat{\bar{\mu}}_d^E$ and $\hat{\delta}_d^E$ applicable to the categorical setup (15). Under Model 2, the algorithm steps are

1. Fit the model to the sample and calculate $\hat{\boldsymbol{\theta}} = (\hat{\boldsymbol{\beta}}^{\intercal},\hat{\phi},\hat{\varphi})^{\intercal}$, put $\hat{\nu}_{dk} = a_{dk}\hat{\varphi}$.
2. Repeat B times ($b = 1,\dots,B$):

   (a) The population. For $d = 1,\dots,D$, $k = 1,\dots,K$ generate $v_d^{*(b)}$ i.i.d. $N(0,1)$ and calculate $\mu_{dk}^{*(b)} = g^{-1}\big(\boldsymbol{z}_k^{\intercal}\hat{\boldsymbol{\beta}} + \hat{\phi}v_d^{*(b)}\big)$ and $\theta_{dk}^{*(b)} = (\dot{b})^{-1}(\mu_{dk}^{*(b)})$. For $j = 1,\dots,N_d$ generate

   $$y_{dj}^{*(b)} \sim \text{Exp}\Big(\theta_{dk}^{*(b)},\hat{\nu}_{dk};a,b,c\Big), \quad \text{where } k \text{ is such that } \boldsymbol{x}_{dj} = \boldsymbol{z}_k.$$

   Calculate the true bootstrap quantities

   $$\bar{\mu}_d^{*(b)} = \bar{\mu}_d(\hat{\boldsymbol{\theta}},v_d^{*(b)}) = \frac{1}{N_d}\sum_{k=1}^{K}N_{dk}\mu_{dk}^{*(b)}, \quad \delta_d^{*(b)} = \frac{1}{N_d}\sum_{j=1}^{N_d}h(y_{dj}^{*(b)}).$$

   (b) The sample. The bootstrap sample has the same units as the real data sample. It is not extracted at random. The model is on the population, therefore the source of randomness comes from the generation of the population. For each bootstrap sample, calculate $\hat{\boldsymbol{\theta}}^{*(b)}$ and the EBPs

$$\hat{\mu}_d^{E*(b)} = \hat{\mu}_d^E(\hat{\boldsymbol{\theta}}^{*(b)}), \quad \hat{\delta}_d^{E*(b)} = \hat{\delta}_d^E(\hat{\boldsymbol{\theta}}^{*(b)}).$$

3. Output: $mse^*(\hat{\mu}_d^E) = \frac{1}{B}\sum_{b=1}^B \big(\hat{\mu}_d^{E*(b)} - \bar{\mu}_d^{*(b)}\big)^2$, $mse^*(\hat{\delta}_d^E) = \frac{1}{B}\sum_{b=1}^B \big(\hat{\delta}_d^{E*(b)} - \delta_d^{*(b)}\big)^2$.

The above algorithm can be easily modified to the case of fitting a GLMM with at least one continuous auxiliary variable. For this sake, a census file is needed with the values of $\boldsymbol{x}_{dj}$ for all the units of the population. In addition, the census file must have the same unit identifier variable as the sample file. This modification is equivalent to adapting the parametric bootstrap method of González-Manteiga et al. (2007) to the current unit-level GLMMs.

7 Simulation experiments

This section presents three simulation experiments for gamma Model 2. Simulation 1 analyses the behaviour of the ML-Laplace approximation algorithm for estimating parameters. Simulation 2 compares the performances of the EBPs, the plug-in predictors and the marginal predictors. Finally, Simulation 3 empirically studies the bootstrap estimators of the MSEs.

In all the experiments data are simulated in the following way. For $d = 1,\dots,D$ and $j = 1,\dots,n_d$, define regressors representing four possible classes of labour status. This is to say, $(x_{dj1}, x_{dj2}) = (0,0)$ for unemployed, $(x_{dj1}, x_{dj2}) = (0,1)$ for employed, $(x_{dj1}, x_{dj2}) = (1,0)$ for inactive and $(x_{dj1}, x_{dj2}) = (1,1)$ for ≤ 15. Generate $(x_{dj1}, x_{dj2}) \in \{(0,0),(0,1),(1,0),(1,1)\}$ with probabilities $p_{00} = 0.1 + \frac{(d-1)}{D-1}0.2$, $p_{01} = 0.5 - \frac{(d-1)}{D-1}0.2$, $p_{10} = 0.2$, and $p_{11} = 0.2$, respectively. For each covariate class and domain d, the constants a_{dj} (which are assumed to be known in Model 2) are generated independently from a normal distribution with mean 1.5 and standard deviation 0.2 in order to be close to the values appearing in the application to the real data. The auxiliary variables, x_{dj1}, x_{dj2}, and the shape constants, a_{dj}, are generated before starting the simulation loop, so they are constant in the three simulation experiments. The model parameters are taken as $\beta_0 = 0.8$, $\beta_1 = -0.15$, $\beta_2 = 0.2$, $\phi = 0.1$ and $\varphi = 2.5$.

Within each iteration, the three simulation algorithms generate the random effect $v_d \sim N(0,1)$, $d = 1,\dots,D$, and the income variable $y_{dj} \sim \text{Gamma}(\nu_{dj}, \frac{\nu_{dj}}{\mu_{dj}})$, where $\nu_{dj} = a_{dj}\varphi$ and

$$\mu_{dj} = (\beta_0 + x_{dj1}\beta_1 + x_{dj2}\beta_2 + \phi v_d)^{-1}, \;\; d = 1,\dots,D,\; j = 1,\dots,n_d.$$

7.1 Simulation 1

The target of Simulation 1 is to check the behaviour of the fitting algorithm. We take $D = 30, 60, 120, 180$ and $n_d = 10, 25, 50$. The steps of simulation 1 are

1. Repeat $I = 1000$ times ($i = 1, \ldots, I$)
   1.1. Generate a sample $\{y_{dj}^{(i)} : d = 1, \ldots, D,\ j = 1 \ldots, n_d\}$ from Model 2.
   1.2. Calculate $\hat{\beta}_0^{(i)}$, $\hat{\beta}_1^{(i)}$, $\hat{\beta}_2^{(i)}$, $\hat{\phi}^{(i)}$, $\hat{\varphi}^{(i)}$.
2. Output: For $\theta \in \{\beta_0, \beta_1, \beta_2, \phi, \varphi\}$, calculate the relative bias and relative root-MSE, i.e.

$$RBIAS = \frac{1}{|\theta|}\frac{1}{I}\sum_{i=1}^{I}(\hat{\theta}^{(i)} - \theta),\ RRMSE = \frac{1}{|\theta|}\Big(\frac{1}{I}\sum_{i=1}^{I}(\hat{\theta}^{(i)} - \theta)^2\Big)^{1/2}.$$

Tables 1, 2 and 3 presents the results of the simulation experiment for $n_d = 10$, $n_d = 25$ and $n_d = 50$ respectively. The relative bias is basically negligible. The relative root-MSE decreases as D or n_d increases. Simulation 1 empirically illustrates the consistency of the implemented ML-Laplace approximation algorithm.

Table 1: *RBIAS (left) and RRMSE (right) in % for* $n_d = 10$.

	RBIAS				RRMSE			
	$D = 30$	$D = 60$	$D = 120$	$D = 180$	$D = 30$	$D = 60$	$D = 120$	$D = 180$
$\hat{\beta}_0$	0.9932	0.6967	0.7074	0.6254	6.1918	4.5492	3.3014	2.5208
$\hat{\beta}_1$	0.3361	0.5187	−0.0944	0.0275	32.9073	24.1085	16.6784	13.0500
$\hat{\beta}_2$	−0.2629	0.0504	0.0892	0.3631	25.7537	18.1448	12.4199	9.8741
$\hat{\phi}$	−11.1515	−3.9887	−0.5083	0.1928	41.0300	27.3513	18.3789	15.0615
$\hat{\varphi}$	1.5736	0.9871	0.5349	0.3370	8.5560	6.0136	4.1093	3.4281

Table 2: *RBIAS (left) and RRMSE (right) in % for* $n_d = 25$.

	RBIAS				RRMSE			
	$D = 30$	$D = 60$	$D = 120$	$D = 180$	$D = 30$	$D = 60$	$D = 120$	$D = 180$
$\hat{\beta}_0$	0.8245	0.9708	0.8585	0.9259	4.2377	3.2867	2.2593	1.9043
$\hat{\beta}_1$	−0.4093	0.3290	0.9417	0.2324	20.0357	14.1744	10.1065	8.3685
$\hat{\beta}_2$	0.4219	0.2353	−0.1183	0.0690	14.8132	11.1234	7.7572	6.3451
$\hat{\phi}$	−4.1686	−2.7112	−1.3648	−0.7450	22.0699	15.8384	10.9345	8.6654
$\hat{\varphi}$	0.6263	0.2041	0.2512	0.2021	5.1735	3.5982	2.5533	2.0827

Table 3: *RBIAS (left) and RRMSE (right) in % for $n_d = 50$.*

	RBIAS				RRMSE			
	$D=30$	$D=60$	$D=120$	$D=180$	$D=30$	$D=60$	$D=120$	$D=180$
$\hat{\beta}_0$	0.9193	1.2258	1.1286	1.2008	3.4704	2.6633	2.0673	1.8352
$\hat{\beta}_1$	0.1209	0.2595	0.0680	−0.0676	14.3711	10.0332	7.3633	5.8110
$\hat{\beta}_2$	0.5331	−0.0164	−0.0452	0.0878	11.1165	7.7034	5.3727	4.4457
$\hat{\phi}$	−5.3541	−2.2712	−1.3643	−1.1853	18.5763	12.1674	8.7474	6.9339
$\hat{\varphi}$	0.2439	0.1738	0.1221	0.0634	3.4888	2.5094	1.8352	1.4415

7.2 Simulation 2

The target of Simulation 2 is to investigate the behaviour of the EBP, the marginal and plug-in predictors of the mean $\overline{Y}_d$ and the poverty proportion p_d. Before starting the simulation loop, a first set of target variables $\{y_{dj}^{(0)}: d=1,\dots,D,\, j=1,\dots,N_d\}$ is generated and the poverty threshold z is taken as the first sample quartile of these variables.

The model is fitted by the ML-Laplace approximation algorithm. The EBP is approximated with $L_1 = L_2 = 100$ Monte Carlo iterations and the domain sizes are $N_d = 1000$, $d = 1,\dots,D$. The steps of Simulation 2 are

1. Repeat $I = 10^4$ times ($i = 1,\dots,I$)

   1.1. Generate the i-th population $\{y_{dj}^{(i)} : d = 1,\dots,D,\, j = 1\dots,N_d\}$ in the same way as the sample in Simulation 1.

   1.2. Calculate $\overline{Y}_d^{(i)} = \frac{1}{N_d}\sum_{j=1}^{N_d} y_{dj}^{(i)}$, $p_d^{(i)} = \frac{1}{N_d}\sum_{j=1}^{N_d} I(y_{dj}^{(i)} < z)$, $d = 1,\dots,D$.

   1.3. For $d = 1,\dots,D$, select a sample $s_d^{(i)}$ of size n_d. The indexes of the samples $s_d^{(i)}$ remain constant across the iterations. Calculate $\hat{\beta}_0^{(i)}$, $\hat{\beta}_1^{(i)}$, $\hat{\beta}_2^{(i)}$, $\hat{\phi}^{(i)}$, $\hat{\varphi}^{(i)}$.

   1.4 Calculate the predictors $\widehat{\overline{Y}}_d^{E,i}$, $\widehat{\overline{Y}}_d^{P,i}$, $\widehat{\overline{Y}}_d^{M,i}$, $\hat{p}_d^{E,i}$, $\hat{p}_d^{P,i}$, $\hat{p}_d^{M,i}$ and the direct estimators $\widehat{\overline{Y}}_d^{dir,i} = \frac{1}{n_d}\sum_{j=1}^{n_d} y_{dj}^{(i)}$, $\hat{p}_d^{dir,i} = \frac{1}{n_d}\sum_{j=1}^{n_d} I(y_{dj}^{(i)} < z)$, $d = 1,\dots,D$.

2. For $\xi_d^{(i)} \in \{\overline{Y}_d^{(i)}, p_d^{(i)}\}$ and $\hat{\xi}_d^{(i)} \in \{\widehat{\overline{Y}}_d^{dir,i}, \widehat{\overline{Y}}_d^{E,i}, \widehat{\overline{Y}}_d^{P,i}, \widehat{\overline{Y}}_d^{M,i}, \hat{p}_d^{dir,i}, \hat{p}_d^{E,i}, \hat{p}_d^{P,i}, \hat{p}_d^{M,i}\}$, $d = 1,\dots,D$, calculate the performance measures

$$\xi_d = \frac{1}{I}\sum_{i=1}^{I}\xi_d^{(i)},\; RE_d = \Big(\frac{1}{I}\sum_{i=1}^{I}\big(\hat{\xi}_d^{(i)} - \xi_d^{(i)}\big)^2\Big)^{1/2},\; RB_d = \frac{1}{|\xi_d|}\frac{1}{I}\sum_{i=1}^{I}(\hat{\xi}_d^{(i)} - \xi_d^{(i)}),$$

$$RRE_d = \frac{RE_d}{|\xi_d|},\; RB = \frac{1}{D}\sum_{d=1}^{D}|RB_d|,\; RRE = \frac{1}{D}\sum_{d=1}^{D}RRE_d.$$

Table 4 presents the results of Simulation 2 for sample sizes $n_d = 10, 25, 50, 75, 100$, which are of similar magnitude to most of the county sample sizes presented in Table 10. For estimating the domain mean, the EBP $\hat{\overline{Y}}_d^E$ has had lower *RB* but higher *RRE* than the marginal/plug-in predictor $\hat{\overline{Y}}_d^P = \hat{\overline{Y}}_d^M$. Further, the model-based predictors are more efficient than the direct estimator. For estimating the domain proportion, the plug-in estimator is not recommended and the marginal and EB predictors are both more efficient than the direct estimator. The marginal predictor has obtained slightly better results than the EBP in relative bias and root-MSE. This fact can be explained by the variability derived from approximating the EBP with $L_1 = L_2 = 100$ Monte Carlo iterations.

Table 4: *RB (left) and RRE (right) in % for* $D = 30$.

n_d	10	25	50	75	100	10	25	50	75	100
$\hat{\overline{Y}}_d^{Dir}$	4.27	2.18	1.63	1.43	0.98	17.69	10.99	7.71	6.22	5.25
$\hat{\overline{Y}}_d^E$	0.33	0.31	0.38	0.23	0.25	11.11	8.50	6.57	5.54	4.82
$\hat{\overline{Y}}_d^P$	0.79	0.53	0.46	0.24	0.25	11.09	8.45	6.54	5.46	4.75
$\hat{\overline{Y}}_d^M$	0.79	0.53	0.46	0.24	0.25	11.09	8.45	6.54	5.46	4.75
$\hat{p}_d^{Dir}$	7.53	4.21	3.03	2.49	1.78	55.10	34.32	24.03	19.33	16.37
$\hat{p}_d^E$	0.57	0.56	0.68	0.43	0.39	21.18	16.62	13.17	11.29	10.06
$\hat{p}_d^P$	98.99	97.50	95.01	92.55	90.04	101.78	100.22	97.65	95.15	92.55
$\hat{p}_d^M$	0.76	0.49	0.31	0.27	0.21	21.17	16.56	13.08	11.18	9.97

Remark 7.1 *Since it is complicated to calculate analytically the error of the Monte Carlo approximation because we approximate the integrals of numerator and denominator, we tried to study the accuracy numerically. Namely, for one choice of* $n_d = 25$ *and one selected iteration of Simulation 2 we have approximated the EBP 1000 times for each domain and different values of L. Then we calculated the standard deviation of these approximations in each domain. This quantity express the variability of the Monte Carlo approximations. The means of the obtained standard deviations over areas are presented in Table 5 for the two cases of predicting the area mean* $\overline{Y}_d$ *and the area proportion* p_d.

Table 5: *Standard deviations of 1000 MC approximations of EBP for* $n_d = 25$ *and different values of L.*

Predictor	$L = 50$	$L = 100$	$L = 200$	$L = 300$	$L = 500$
$\hat{\overline{Y}}_d^E$	0.01370	0.00960	0.00665	0.00552	0.00421
$\hat{p}_d^E$	0.00593	0.00380	0.00250	0.00202	0.00152

In the simulation experiments we have used $L = 100$ *from time reasons but from the Table 5 it follows that in practical applications a higher value of L, e.g.* $L = 300$, *could be recommended. Higher values of L than 300 increase substantially the computing time.*

As a consequence of the results of Simulation 2, we will consider only the marginal predictor in Simulation 3 and we recommend using the marginal predictor in applications to real data.

For explaining the poor behaviour of plug-in predictors $\hat{p}_d^P$ of domain proportions p_d, we recall that

$$\hat{p}_d^P = \frac{1}{N_d}\Big[\sum_{j\in s_d} I(y_{dj}<z) + \sum_{k=1}^{K}(N_{dk}-n_{dk})I(\tilde{\mu}_{dk}<z)\Big],$$

where $K=4$ and the poverty threshold z is taken as the lower sample quartile of the generated target variables. Since $\tilde{\mu}_{dk}$ is the predictor of the expectation of the target variable distribution, then $\tilde{\mu}_{dk}$ tends to be greater than z in most of the iterations of the simulation experiment. This is to say, the probability that one of the terms $I(\tilde{\mu}_{dk}<z)$ is equal to 1 is very small. In other words, the observed value of a Bernoulli variable is a bad estimator of the probability of success. This means that the summands in the second sum will be equal to zero with high probability and therefore the auxiliary information is almost not taken into account. Moreover, only the sample s_d is used in the first sum as in the case of direct estimators, but this sum is divided by the population size N_d instead of the sample size n_d.

7.3 Simulation 3

Simulation 3 investigates the behaviour of the bootstrap MSE estimator of the marginal predictors. We take $D=30$, $n_d=50$, $N_d=1000$, $I=500$ and $B=25,50,100,200,300,400$. We take $E_d=(RE_d)^2$ from the output of Simulation 2. The steps of Simulation 3 are

1. Repeat $I=500$ times ($i=1,\ldots,I$)

   1.1. Generate the population in the same way as sample in Simulation 2.

   1.2. For $d=1,\ldots,D$, select a sample s_d of size n_d with fixed indexes and calculate $\hat{\beta}_0^{(i)}$, $\hat{\beta}_1^{(i)}$, $\hat{\beta}_2^{(i)}$, $\hat{\phi}^{(i)}$, $\hat{\varphi}^{(i)}$.

   1.3. For each $\hat{\xi}_d^{(i)}\in\{\hat{\overline{Y}}_d^{M,i},\hat{p}_d^{M,i}\}$, calculate $mse_d^{*(i)}=mse_d^*(\hat{\xi}_d^{(i)})$.

2. Calculate the relative performance measures

$$Rb_d=\frac{1}{|E_d|}\frac{1}{I}\sum_{i=1}^{I}(mse_d^{*(i)}-E_d),\; Re_d=\frac{1}{|E_d|}\Big(\frac{1}{I}\sum_{i=1}^{I}(mse_d^{*(i)}-E_d)^2\Big)^{1/2},$$

$$Rb = \frac{1}{D}\sum_{d=1}^{D}|Rb_d|, \quad Re = \frac{1}{D}\sum_{d=1}^{D}Re_d.$$

Table 6 summarizes the results of Simulation 3. Figure 1 presents the boxplots of the relative biases Rb_d (left) and the relative root-MSEs Re_d (right) of MSE estimators for average incomes. Figure 2 presents the same boxplots for poverty proportions.

Table 6: *Relative biases and root-MSEs (in %) of MSE estimators for average incomes (top) and poverty proportions (bottom).*

		$B=25$	$B=50$	$B=100$	$B=200$	$B=300$	$B=400$
$\overline{Y}_d$	*Rb*	5.91	5.59	6.46	6.91	6.87	6.47
	Re	32.74	24.65	19.51	16.76	15.54	14.84
p_d	*Rb*	1.77	1.51	1.62	2.06	1.57	1.77
	Re	31.10	23.20	18.09	14.83	14.23	13.29

From the figures we observe that the parametric bootstrap method slightly underestimate the MSEs of the marginal predictors and that root mean squared error of the estimates is decreasing with increasing B. On the basis of the results we recommend to use at least $B = 200$ bootstrap iterations for estimating the MSEs of the marginal predictors.

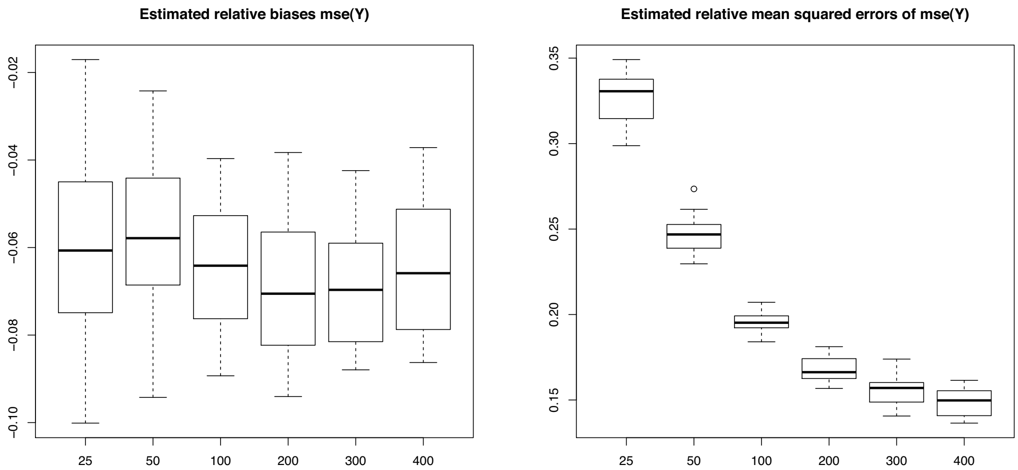


Figure 1: *Boxplots of relative biases Rb_d (left) and root-MSEs Re_d (right) of MSE estimators of average incomes.*

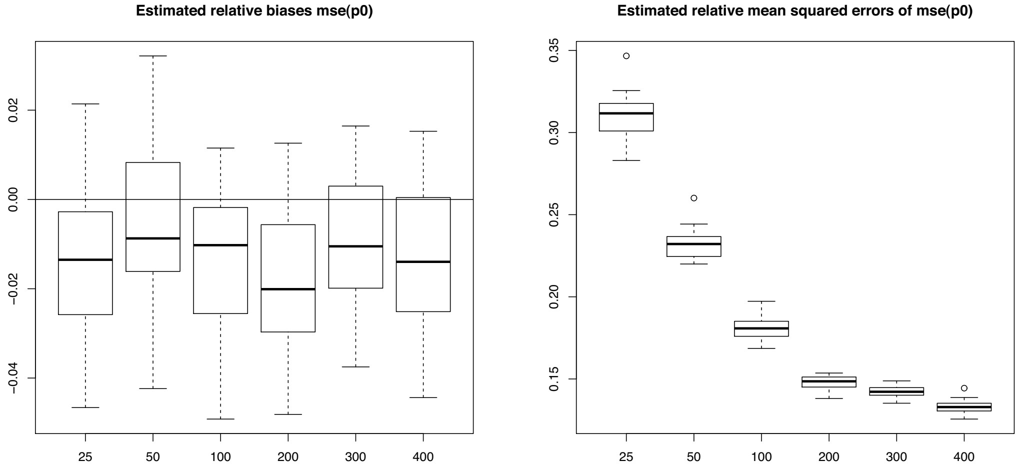


Figure 2: *Boxplots of relative biases Rb_d (left) and root-MSEs Re_d (right) of MSE estimators of poverty proportions.*

8 Application to SLCS data

This section presents an application to unit-level data from the 2013 SLCS of the region of Valencia. The SLCS is carried out in all the Spanish territory and the planned domains are the regions. The counties, within the region of Valencia, have rather small sample sizes and they are considered as small areas by the Spanish Statistical Office. We estimate average incomes and poverty proportions. The SLCS contains data from $D = 26$ Valencian counties and these counties are the domains of interest. The target variable y_{dj} is the average annual net income (in 10^4 euros) of individual j from domain d. The selected auxiliary variables are the labour status categories (employed, unemployed, inactive and below 15 years old). In addition to the SLCS data, we take auxiliary aggregated data from the SLFS, which contains survey data about the labour market. As the regional sample sizes of the SLFS are much greater than the corresponding ones of the SLCS, the sizes of counties crossed by the labour status categories are estimated from the SLFS and considered as known quantities.

We start the data analysis by doing a preliminary step. We fit gamma Model 1 to the data $(y_{dj}, x_{dj1}, x_{dj2})$, $d = 1,\dots,D$, $j = 1,\dots,n_d$, where x_{dj1} and x_{dj2} are the dichotomic variables indicating if an individual is employed and unemployed (yes = 1, no = 0) respectively. The $K = 3$ covariate classes are $z_1 = (1,0)$, $z_2 = (0,1)$ and $z_3 = (0,0)$ for employed, unemployed and rest (≤ 15 or inactive) respectively. For $d = 1,\dots,D$, $j = 1,\dots,N_d$, we consider the population model (Model 1)

$$y_{dj}|_{v_d} \sim \text{Gamma}\big(\nu, \nu/\mu_{dj}\big), \quad g(\mu_{dj}) = \frac{1}{\mu_{dj}} = \beta_0 + \beta_1 x_{dj1} + \beta_2 x_{dj2} + \phi v_d.$$

By using the glmer function of the R statistical package lme4, we fit Model 1 by applying the ML-Laplace approximation algorithm. Table 7 (left) gives the estimates of the regression parameters and their estimated standard deviations and p-values. Table 7 (right) presents the estimates of parameters ϕ and ν and the corresponding quantile-based 95% confidence intervals calculated from 1000 parametric bootstrap samples.

***Table 7**: Regression (left) and var/shape (right) parameter estimates under Model 1.*

	estimate	standard error	p-value
$\tilde{\beta}_0$	0.771	0.0310	< 2E-16
$\tilde{\beta}_1$	-0.142	0.0163	< 2E-16
$\tilde{\beta}_2$	0.142	0.0287	7.42E-07

	estimate	95% conf. interval
$\tilde{\phi}$	0.093	(0.0496, 0.0991)
$\tilde{\nu}$	2.858	(2.270, 2.952)

Let us note that initially we have calculated the EBP's and marginal predictors of average incomes and poverty proportions under Model 1 but the results were unsatisfactory. The predicted values had very small variability between domains and for large sample sizes n_d they did not correspond to direct estimators. So the assumption $\nu_{dj} = \nu$ for all d and j is too rigid in this case and a more general model is needed.

We take the plug-in predictors $\tilde{\mu}_{dj} = (\tilde{\beta}_0 + \tilde{\beta}_1 x_{dj1} + \tilde{\beta}_2 x_{dj2} + \tilde{\phi}\tilde{v}_d)^{-1}$ calculated under Model 1 as a preliminary step and we use them as inputs of the algorithmic procedure for fitting the more complex unit-level gamma Model 2. For $d = 1,\dots,D$, $j = 1,\dots,N_d$, we consider the population model (Model 2)

$$y_{dj}|_{v_d} \sim \text{Gamma}\big(\nu_{dj}, \nu_{dj}/\mu_{dj}\big),\ \nu_{dj} = a_{dj}\varphi,\ g(\mu_{dj}) = \frac{1}{\mu_{dj}} = \beta_0 + \beta_1 x_{dj1} + \beta_2 x_{dj2} + \phi v_d.$$

For fitting Model 2 to the data, we first need the constants a_{dj}. Since they are not known in our case, we estimate them by the following algorithmic procedure.

1. For a grid of values of t in the interval (0.25, 3) and step equal to 0.01, fit the Model 2 to the data, assuming that $a_{dj} = \tilde{\mu}_{dj}^t$ is known. If $t = 2$ and a_{dj} is equal to μ_{dj}^t, then $var[y_{dj}|v_d] = 1/\varphi$, which corresponds to the homoscedastic case. Calculate the estimator $\hat{\varphi}(t)$ for each considered t.

2. For each considered t, calculate the plug-in predictors $\hat{\mu}_{dj}^{(t)}$, the raw residuals $\hat{e}_{dj}^{(t)} = y_{dj} - \hat{\mu}_{dj}^{(t)}$ and the sum of the squared residuals $r^2(t)$. Select t_* minimizing $r^2(t)$.

3. Do the inferences with Model 2 and $a_{dj} = \tilde{\mu}_{dj}^{t_*}$ known, i.e. $\nu_{dj} = \hat{\varphi}(t_*)\tilde{\mu}_{dj}^{t_*}$.

For the considered data set, the selected optimal choice of t is $t_* = 0.60$. Figure 3 presents a plot of the function $r^2(t)$ and a boxplot of the optimal shape constants $a_{dj} = \tilde{\mu}_{dj}^{t_*}$.

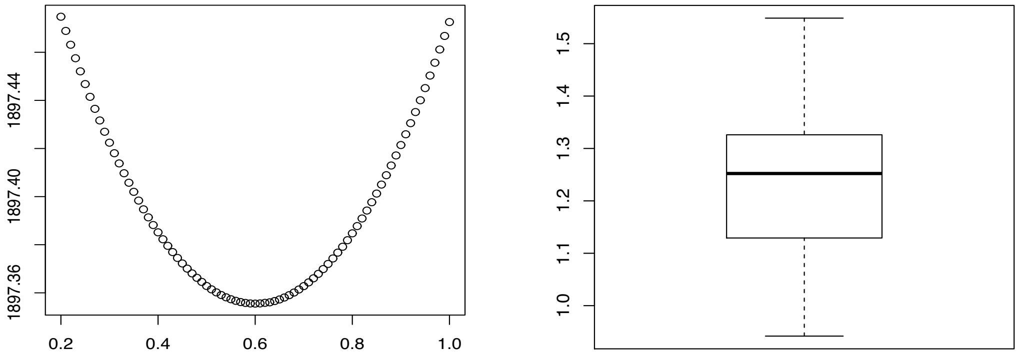


Figure 3: *Function $r^2(t)$ (left) and boxplot of a_{dj} (right).*

Table 8 gives the parameter estimates of Model 2. The signs of the regression parameter and the form of the link function indicate that employment (unemployment) has a positive (negative) effect on income.

Table 8: *Parameter estimates under Model 2.*

	estimate	standard error	p-value
$\hat{\beta}_0$	0.775	0.0132	< 2E-16
$\hat{\beta}_1$	-0.141	0.0157	< 2E-16
$\hat{\beta}_2$	0.140	0.0300	3.09E-06
$\hat{\phi}$	0.1113	0.0112	< 2E-16
$\hat{\varphi}$	2.4646	0.0675	< 2E-16

For the sake of comparisons, we also fit the unit-level log-linear normal mixed model (Model 3)

$$z_{dj} = b_0 + b_1 x_{dj1} + b_2 x_{dj2} + u_d + e_{dj}, \quad d = 1,\dots,D,\ j = 1,\dots,n_d,$$

where $z_{dj} = \log(y_{dj}+1)$, $u_d \sim N(0,\sigma_u^2)$, $e_{dj} \sim N(0,\sigma_e^2)$ and the random effects $u_d \sim N(0,\sigma_u^2)$ and the random errors $e_{dj} \sim N(0,\sigma_e^2)$ are mutually independent. By using the lmer function of the R statistical package lme4, we fit Model 3 by applying the REML method. The estimates of the model standard deviations are $\sigma_u = 0.0886$ and $\sigma_e = 0.3036$. Table 9 presents the estimates of the regression parameters of Model 3.

Table 9: *Parameter estimates under Model 3.*

	estimate	standard error	p-value
$\hat{b}_0$	0.803	0.0201	< 2E-16
$\hat{b}_1$	0.137	0.0135	< 2E-16
$\hat{b}_2$	-0.112	0.0180	5.41E-10

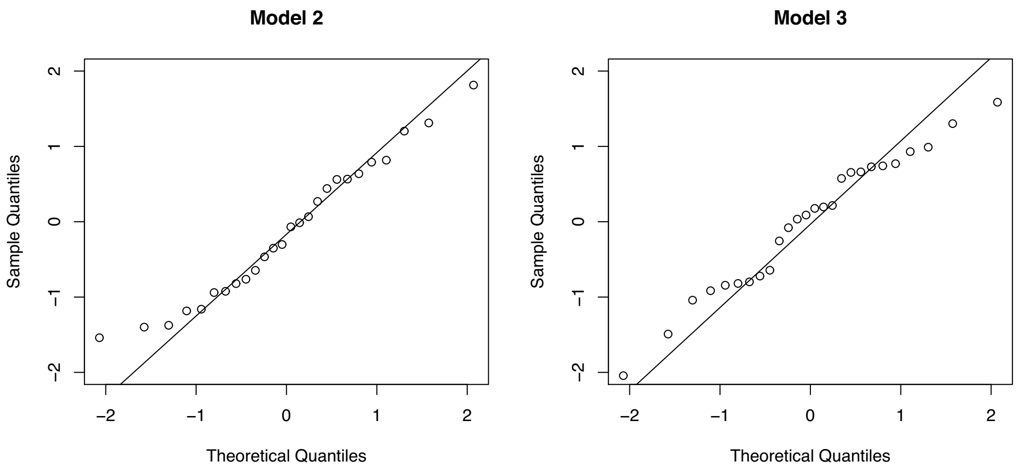


Figure 4: *Q-Q plot of random effects for models 2 (left) and 3 (right).*

In order to check the assumption that the random effects have the standard normal distribution, we take the mode predictors $\hat{v}_d$ and $\hat{u}_d$ of the random effects under Model 2 and Model 3 respectively and we plot the normal Q-Q plots in Figure 4. We do not observe significant deviations from normality. Moreover, the Kolmogorov-Smirnov test does not reject the hypothesis $H_0 : F_{\hat{v}_{1,d}} = F_{N(0,1)}$ with p-values equal to 0.763 (Model 2) and 0.925 (Model 3).

Remark 8.1 *In Figure 4 (left) there are two domains that are far from the straight line indicating normality in the bottom-left corner. To illustrate robustness of the method we have investigated what happens if we drop out all the observations of these two domains. We have fitted Model 2 without the mentioned observations and the results are presented in Table 11 of Appendix B. Since the parameter estimates are very similar to those given in Table 8, we can say that the methodology is robust with respect to small deviations from the hypothesis of normality of the random effects.*

Figure 5 presents graphs of raw residuals for Model 2 (left) and Model 3 (right). There are not significant visual differences between both models.

The sum of squares of raw residuals for models 2 and 3 are

$$r_2^2 = \sum_{d=1}^{D}\sum_{j=1}^{n_d}(y_{dj} - \hat{\mu}_{dj})^2 = 1897.35, \quad r_3^2 = \sum_{d=1}^{D}\sum_{j=1}^{n_d}(y_{dj} - (\exp(\hat{z}_{dj}) - 1))^2 = 1938.30.$$

As we observe that Model 2 has a slightly better fit to data, we do the estimation of the small area parameters (average income and poverty proportion) under Model 2. This application illustrates that it may have sense to consider more general GLMM instead of using normal mixed model for some transformation of data.

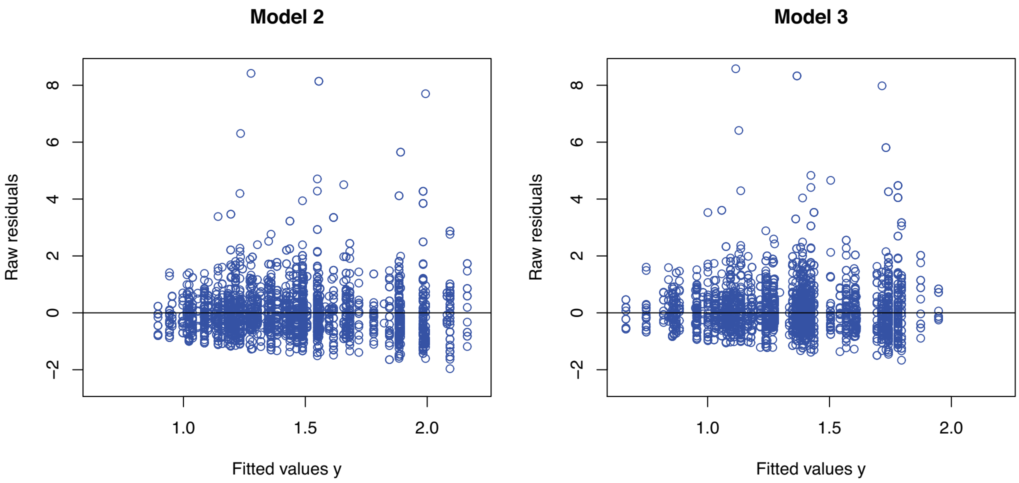


Figure 5: *Dispersion graphs of raw residuals for Model 2 (left) and Model 3 (right).*

Remark 8.2 *In order to study the importance of the random area effects in the model, we have fitted the following Gamma model without area effects (referred to as Model 0)*

$$y_{dj}|v_d \sim Gamma(\nu, \nu/\mu_{dj}), \quad d = 1,\dots,D,\ j = 1,\dots,n_d,$$

$$\eta_{dj} = g(\mu_{dj}) = \frac{1}{\mu_{dj}} = \boldsymbol{x}_{dj}^{\intercal}\boldsymbol{\beta}, \tag{18}$$

and calculated the corresponding marginal predictors based on this model. The parameter estimates can be obtained by the R function glm and are presented in Table 12 in Appendix B. Unfortunately this model has a bad fit to data (the sum of squares of raw residuals $r_0^2 = 2010.56$ *in comparison with* $r_2^2 = 1897.35$ *obtained for model 2) and it does not explain the between domain variability which is not described by the auxiliary variables. Since the sizes of population classes are quite homogeneous between domains, it results in a quite over-smoothed behaviour of the predictors as can be seen from Table 13 of Appendix B. This table presents the marginal predictors under Model 2 and Model 0 and the corresponding bootstrap estimates of the MSE. The results for proportion predictions are presented also in Figure 8. From this figure one can see the smoothing effect of Model 0 and also that the estimated MSEs are for this model higher than the estimated MSEs of the direct estimators for sample sizes higher than 60.*

In order to get the marginal predictor of proportions, we need the a_{dj} values for the whole population or at least the values a_{dk} for the covariate classes z_k, $k = 1,2,3$. Figures 6 and 7 were obtained by the choice

$$a_{dk} = \tilde{\mu}_{dk}^{t_*},$$

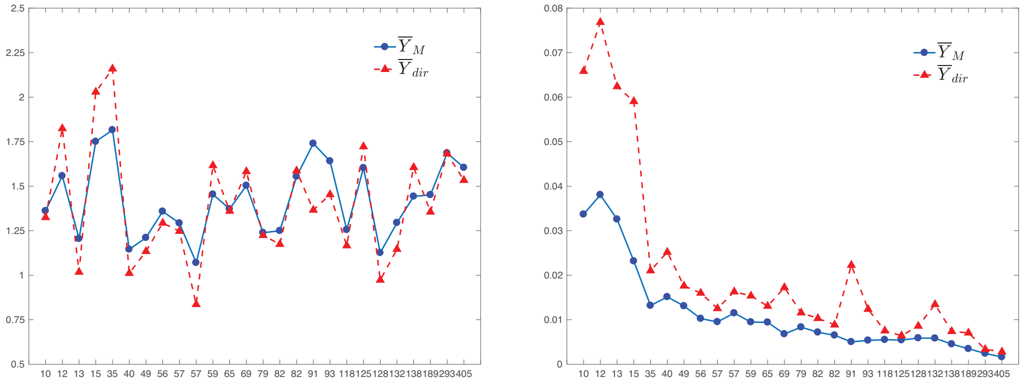


Figure 6: *Predictions of average incomes in* 10^4 *euros (left) and estimated MSEs (right).*

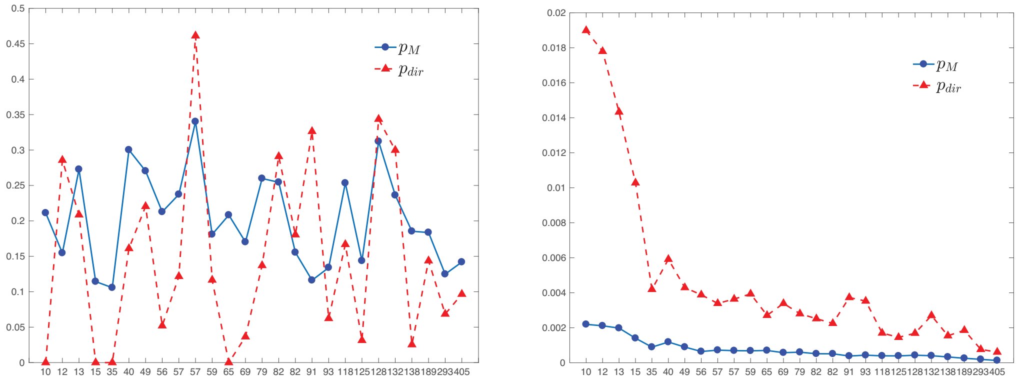


Figure 7: *Predictions of poverty proportions (left) and estimated MSEs (right).*

where $\tilde{\mu}_{dk}$ is the predictor of μ_{dk} derived under Model 1 ($\nu_{dj} = \nu$) and $t_* = 0.60$ is the optimal choice of t. This means that for every unit j of domain d and covariate class $z_k \in \{z_1, z_2, z_3\}$, we take $a_{dj} = a_{dk}$ if $\boldsymbol{x}_{dj} = z_k$, $d = 1, \ldots, 26$, $j = 1, \ldots, n_d$, $k = 1, 2, 3$.

Table 10 presents county codes (c), sample sizes (n_d), population sizes (N_d), marginal predictions of average incomes in 10^4 euros ($\overline{Y}_M$), marginal predictions of poverty proportions (p_M), direct estimates of average incomes in 10^4 euros ($\overline{Y}_{dir}$) and direct estimates of poverty proportions (p_{dir}). It also gives the corresponding MSE estimates (mse) based on 500 bootstrap samples generated from the fitted Model 2. As auxiliary population data, we took the SLFS data file of the region of Valencia in 2013. The poverty line for the region of Valencia was 6999.6 euros in 2013.

Figure 6 plots the marginal predictions ($\overline{Y}_M$) and the direct estimates ($\overline{Y}_{dir}$) of average incomes in 10^4 euros (left) and their corresponding model-based MSE bootstrap estimates (right). Figure 7 plots the marginal predictions (p_M) and the direct estimates (p_{dir}) of poverty proportions (left) and their corresponding model-based MSE bootstrap estimates (right). In all cases, the counties are sorted by sample size. We observe that the model-based marginal predictions have a more smooth behaviour across counties than the direct estimates. Further, the marginal predictions and the direct estimates tend to be close when sample size increases. As expected, the marginal predictors have had lower estimated MSEs than the direct estimators.

Table 10: *Predictions and estimated MSEs for average incomes and poverty proportions.*

c	n_d	N_d	$\overline{Y}_M$	mse	p_M	mse	$\overline{Y}_{dir}$	mse	p_{dir}	mse
27	82	124270	1.2496	0.00718	0.2545	0.00051	1.1738	0.01031	0.2909	0.00252
28	57	70944	1.2924	0.00951	0.2374	0.00072	1.2471	0.01253	0.1217	0.00339
29	69	225440	1.5033	0.00680	0.1702	0.00058	1.5825	0.01725	0.0364	0.00339
30	132	227463	1.2945	0.00583	0.2362	0.00040	1.1461	0.01344	0.2998	0.00270
31	56	166774	1.3584	0.01025	0.2127	0.00064	1.2932	0.01600	0.0520	0.00388
32	293	459626	1.6860	0.00243	0.1248	0.00019	1.6821	0.00335	0.0685	0.00076
33	128	268924	1.1260	0.00589	0.3121	0.00043	0.9728	0.00858	0.3436	0.00169
34	59	292243	1.4540	0.00947	0.1809	0.00068	1.6162	0.01536	0.1166	0.00393
3	57	87560	1.0701	0.01149	0.3401	0.00069	0.8361	0.01629	0.4612	0.00364
5	91	246942	1.7397	0.00503	0.1162	0.00039	1.3659	0.02225	0.3263	0.00373
6	82	179798	1.5543	0.00651	0.1555	0.00051	1.5869	0.00885	0.1805	0.00225
7	10	26007	1.3613	0.03368	0.2112	0.00219	1.3245	0.06582	0.0000	0.01899
11	118	189865	1.2552	0.00551	0.2534	0.00040	1.1662	0.00752	0.1668	0.00170
12	15	89136	1.7504	0.02317	0.1144	0.00140	2.0290	0.05904	0.0000	0.01028
13	138	187515	1.4426	0.00454	0.1853	0.00033	1.6057	0.00738	0.0253	0.00154
14	189	370540	1.4513	0.00347	0.1836	0.00026	1.3552	0.00705	0.1438	0.00185
15	405	771129	1.6043	0.00163	0.1419	0.00013	1.5332	0.00279	0.0966	0.00061
16	93	131337	1.6408	0.00536	0.1340	0.00043	1.4531	0.01237	0.0623	0.00353
17	12	33122	1.5576	0.03806	0.1547	0.00211	1.8238	0.07683	0.2857	0.01779
18	35	54545	1.8157	0.01318	0.1057	0.00090	2.1590	0.02103	0.0000	0.00419
20	125	256553	1.6029	0.00543	0.1437	0.00039	1.7224	0.00642	0.0314	0.00145
21	49	52958	1.2107	0.01308	0.2704	0.00090	1.1340	0.01760	0.2205	0.00430
22	13	33126	1.2050	0.03257	0.2727	0.00198	1.0174	0.06239	0.2086	0.01433
23	40	70642	1.1452	0.01513	0.3002	0.00118	1.0109	0.02518	0.1611	0.00591
24	65	80434	1.3719	0.00942	0.2082	0.00070	1.3600	0.01306	0.0000	0.00271
25	79	180619	1.2386	0.00833	0.2598	0.00060	1.2238	0.01157	0.1371	0.00280

9 Concluding remarks

This paper introduces predictors of additive parameters under unit-level GLMMs. The introduced models are applicable also to continuous positive target random variables that have asymmetric distributions, like income or expenditure. In some practical cases, a GLMM can be a good alternative to the log-normal nested error model considered by Molina and Rao (2010). In the application to real data, we give a three-step procedure to determine the shape constants a_{dj} of gamma Model 2. This model has a high flexibility for fitting real data because the a_{dj}'s depend on d and j and therefore they vary within and between domains.

Among the three considered predictors, the simulations show that the empirical best and the marginal predictors have a similarly good behaviour. As the computation of the marginal predictor is less time demanding, we recommend it. Overall when reporting MSEs estimated by parametric bootstrap.

The new small area estimation methodology is applied to the SLCS data from Valencia, a region in east of Spain, in the period January-December 2013. The selected gamma Model 2 has had a slightly better fit to the data than the corresponding log-normal nested error regression model. Therefore the average incomes and poverty proportions per county are finally estimated by using the marginal predictors with its MSEs calculated by parametric bootstrap under gamma Model 2.

The simulations and the application to real data have been carried out with the programming language R. The codes are available upon request to the authors.

Acknowledgements

This work was supported by the Spanish grant ePGC2018-096840-B-I00 from Ministerio de Economía y Competitividad, the Czech grant SGS18/188/OHK4/3T/14 provided by Ministry of Education, Youth and Sports (MŠMT ČR) and from European Regional Development Fund-Project "Center of Advanced Applied Sciences"
(No. CZ.02.1.01/0.0/0.0/16_019/0000778).

References

Benavent, R. and Morales, D. (2016). Multivariate Fay-Herriot models for small area estimation. *Computational Statistics and Data Analysis*, 94, 372-390.

Boubeta, M., Lombardía, M.J. and Morales, D. (2016). Empirical best prediction under area-level Poisson mixed models. *TEST*, 25, 548-569.

Boubeta, M., Lombardía, M.J. and Morales, D. (2017). Poisson mixed models for studying the poverty in small areas. *Computational Statistics and Data Analysis*, 107, 32-47.

Chambers, R., Salvati, N. and Tzavidis, N. (2012). *M-Quantile Regression for Binary Data with Application to Small Area Estimation.* Centre for Statistical and Survey Methodology, University of Wollongong. (Working Paper 12-12, 2012, 24). Available at: http://ro.uow.edu.au/cssmwp/101.

Chambers, R., Salvati, N. and Tzavidis, N. (2016). Semiparametric small area estimation for binary outcomes with application to unemployment estimation for local authorities in the UK. *Journal of the Royal Statistical Society, Series A*, 179, 2, 453-479.

Dreassi, E., Petrucci, A. and Rocco, E. (2014). Small area estimation for semicontinuous skewed spatial data: An application to the grape wine production in Tuscany. *Biometrical Journal*, 56, 1, 141-156

Elbers, C., Lanjouw, J.O. and Lanjouw, P. (2003). Micro-level estimation of poverty and inequality. *Econometrica*, 71, 355-364.

Esteban, M.D., Morales, D., Pérez, A. and Santamaría, L. (2012a). Two Area-Level Time Models for Estimating Small Area Poverty Indicators. *Journal of the Indian Society of Agricultural Statistics*, 66, 1, 75-89.

Esteban, M.D., Morales, D., Pérez, A. and Santamaría, L. (2012b). Small area estimation of poverty proportions under area-level time models. *Computational Statistics and Data Analysis*, 56, 2840-2855.

Fabrizi, E., Ferrante, M.R. and Trivisano, C. (2017). Bayesian small area estimation for skewed business survey variables. *Jornal of the Royal Statistical Society, Series C*. doi.org/10.1111/rssc.12254

González-Manteiga, W., Lombardía, M.J., Molina, I., Morales, D. and Santamaría, L. (2007). Estimation of the mean squared error of predictors of small area linear parameters under a logistic mixed model. *Computational Statistics and Data Analysis*, 51, 2720-2733.

González-Manteiga, W., Lombardía, M.J., Molina, I., Morales, D. and Santamaría, L. (2008a). Bootstrap mean squared error of small-area EBLUP. *Journal of Statistical Computation and Simulation*, 78, 443-462.

González-Manteiga, W., Lombardía, M.J., Molina, I., Morales, D. and Santamaría, L. (2008b). Analytic and bootstrap approximations of prediction errors under a multivariate Fay-Herriot model. *Computational Statistics and Data Analysis*, 52, 5242-5252.

Guadarrama, M., Molina, I. and Rao, J.N.K. (2014). A comparison of small area estimation methods for poverty mapping. *Statistics in Transition new series and Survey Methodology*, 17, 41-66.

Hall, P. and Maiti, T. (2006a). Nonparametric estimation of mean-squared prediction error in nested-error regression models. *Annals of Statistics*, 34, 4, 1733-1750.

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

Hobza, T. and Morales, D. (2013). Small area estimation under random regression coefficient models. *Journal of Statistical Computation and Simulation*, 83, 11, 2160-2177.

Hobza, T. and Morales, D. (2016). Empirical Best Prediction Under Unit-Level Logit Mixed Models. *Journal of official statistics*, 32, 3, 661-692.

Hobza, T., Morales, D. and Santamaría, L. (2018). Small area estimation of poverty proportions under unit-level temporal binomial-logit mixed models. *TEST*, 27, N. 2., 270-294.

Karlberg, F. (2014). Small area estimation for skewed data in the presence of zeroes. *Statistics in Transition*, 16, 4, 541-562.

Marhuenda, Y., Molina, I. and Morales, D. (2013). Small area estimation with spatio-temporal Fay-Herriot models. *Computational Statistics and Data Analysis*, 58, 308-325.

Marhuenda, Y., Molina, I., Morales, D. and Rao, J.N.K. (2017). Poverty mapping in small areas under a two-fold nested error regression model. *Journal of the Royal Statistical Society, series A*, 180, 4, 1111-1136.

Molina, I., Nandram, B. and Rao, J.N.K. (2015). Small area estimation of general parameters with application to poverty indicators: a hierarchical Bayes approach. *The Annals of Applied Statistics*, 8, 852-885.

Molina, I. and Rao, J.N.K. (2010). Small area estimation of poverty indicators. *The Canadian Journal of Statistics*, 38, 369-385.

Morales, D., Pagliarella, M.C. and Salvatore, R. (2015). Small area estimation of poverty indicators under partitioned area-level time models. *SORT-Statistics and Operations Research Transactions*, 39, 1, 19-34.

Moura, F.A.S., Silva, D. B. N. and Neves, A.F. (2017). Small area models for skewed Brazilian business survey data. *Jornal of the Royal Statistical Society, Series A*, 180, 4, 1039-1055.

Pratesi, M. (2016). *Analysis of Poverty Data by Small Area Estimation*. John Wiley.

R Core Team (2019). R: A language and environment for statistical computing. *R Foundation for Statistical Computing*, Vienna, Austria. https://www.R-project.org/

Rao J. N. K. and Molina, I. (2015). *Small Area Estimation*, 2nd Edition, New York: Wiley.

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

Tzavidis, N., Salvati, N., Pratesi, M. and Chambers, R. (2008). M-quantile models with application to poverty mapping. *Statistical Methods and Applications*, 17, 3, 393-411.

A Appendix

This appendix gives the partial derivatives of the function ℓ_{0d}, defined in (6), for gamma Model 2. The first derivatives of μ_{0dj} and ξ_{0d} are

$$\frac{\partial \mu_{0dj}}{\partial \beta_r} = -x_{djr}\mu_{0dj}^2, \quad \frac{\partial \mu_{0dj}}{\partial \phi} = -v_{0d}\mu_{0dj}^2, \quad \frac{\partial \mu_{0dj}}{\partial \varphi} = 0,$$

$$\eta_{0dr} = \frac{\partial \xi_{0d}}{\partial \beta_r} = -2\phi^2 \sum_{j=1}^{n_d} a_{dj}\varphi x_{djr}\mu_{0dj}^3, \quad \eta_{0d\varphi} = \frac{\partial \xi_{0d}}{\partial \varphi} = \phi^2 \sum_{j=1}^{n_d} a_{dj}\mu_{0dj}^2,$$

$$\eta_{0d} = \frac{\partial \xi_{0d}}{\partial \phi} = \sum_{j=1}^{n_d} \big\{2\phi a_{dj}\varphi\mu_{0dj}^2 - 2\phi^2 a_{dj}\varphi v_{0d}\mu_{0dj}^3\big\}.$$

The first derivatives of ℓ_{0d} with respect to β_r, ϕ and φ are

$$\frac{\partial \ell_{0d}}{\partial \beta_r} = -\frac{1}{2}\frac{\eta_{0dr}}{\xi_{0d}} + \sum_{j=1}^{n_d} \big\{a_{dj}\varphi x_{djr}\mu_{0dj} - a_{dj}\varphi x_{djr} y_{dj}\big\},$$

$$\frac{\partial \ell_{0d}}{\partial \phi} = -\frac{1}{2}\frac{\eta_{0d}}{\xi_{0d}} + \sum_{j=1}^{n_d} \big\{a_{dj}\varphi v_{0d}\mu_{0dj} - a_{dj}\varphi v_{0d} y_{dj}\big\},$$

$$\frac{\partial \ell_{0d}}{\partial \varphi} = \sum_{j=1}^{n_d} \big\{a_{dj}\log a_{dj} + a_{dj}\log\varphi + a_{dj} + a_{dj}\log y_{dj} - a_{dj}\psi(a_{dj}\varphi)\big\} - \frac{1}{2}\frac{\eta_{0d\varphi}}{\xi_{0d}}$$
$$+ \sum_{j=1}^{n_d} \big\{a_{dj}\log(\boldsymbol{x}_{dj}^{\intercal}\boldsymbol{\beta} + \phi v_{0d}) - a_{dj} y_{dj}(\boldsymbol{x}_{dj}^{\intercal}\boldsymbol{\beta} + \phi v_{0d})\big\},$$

where $\psi(z) = \frac{d\log\Gamma(z)}{dz}$ is the digamma function. It holds that

$$\frac{\partial \eta_{0dr}}{\partial \beta_s} = \sum_{j=1}^{n_d} \big\{6\phi^2 a_{dj}\varphi x_{djr}x_{djs}\mu_{0dj}^4\big\}, \quad \frac{\partial \eta_{0dr}}{\partial \varphi} = -2\phi^2 \sum_{j=1}^{n_d} a_{dj}x_{djr}\mu_{0dj}^3,$$

$$\frac{\partial \eta_{0dr}}{\partial \phi} = \sum_{j=1}^{n_d} \big\{-4\phi a_{dj}\varphi x_{djr}\mu_{0dj}^3 + 6\phi^2 a_{dj}\varphi x_{djr} v_{0d}\mu_{0dj}^4\big\},$$

$$\frac{\partial \eta_{0d}}{\partial \beta_r} = \frac{\partial \eta_{0dr}}{\partial \phi}, \quad \frac{\partial \eta_{0d}}{\partial \varphi} = 2\sum_{j=1}^{n_d} \big\{\phi a_{dj}\mu_{0dj}^2 - \phi^2 a_{dj} v_{0d}\mu_{0dj}^3\big\},$$

$$\frac{\partial \eta_{0d}}{\partial \phi} = \sum_{j=1}^{n_d} \big\{2a_{dj}\varphi\mu_{0dj}^2 - 8\phi a_{dj}\varphi v_{0d}\mu_{0dj}^3 + 6\phi^2 a_{dj}\varphi v_{0d}^2\mu_{0dj}^4\big\},$$

$$\frac{\partial \eta_{0d\varphi}}{\partial \beta_r} = \frac{\partial \eta_{0dr}}{\partial \varphi}, \quad \frac{\partial \eta_{0d\varphi}}{\partial \phi} = \frac{\partial \eta_{0d}}{\partial \varphi}, \quad \frac{\partial \eta_{0d\varphi}}{\partial \varphi} = 0.$$

The second partial derivatives of ℓ_{0d} are

$$\frac{\partial^2 \ell_{0d}}{\partial \beta_s \partial \beta_r} = -\frac{1}{2}\frac{\frac{\partial \eta_{0dr}}{\partial \beta_s}\xi_{0d} - \eta_{0dr}\eta_{0ds}}{\xi_{0d}^2} - \sum_{j=1}^{n_d} a_{dj}\varphi x_{djr}x_{djs}\mu_{0dj}^2,$$

$$\frac{\partial^2 \ell_{0d}}{\partial \phi \partial \beta_r} = -\frac{1}{2}\frac{\frac{\partial \eta_{0dr}}{\partial \phi}\xi_{0d} - \eta_{0dr}\eta_{0d}}{\xi_{0d}^2} - \sum_{j=1}^{n_d} a_{dj}\varphi v_{0d}x_{djr}\mu_{0dj}^2,$$

$$\frac{\partial^2 \ell_{0d}}{\partial \varphi \partial \beta_r} = -\frac{1}{2}\frac{\frac{\partial \eta_{0dr}}{\partial \varphi}\xi_{0d} - \eta_{0dr}\eta_{0d\varphi}}{\xi_{0d}^2} + \sum_{j=1}^{n_d}\left\{a_{dj}x_{djr}\mu_{0dj} - a_{dj}x_{djr}y_{dj}\right\},$$

$$\frac{\partial^2 \ell_{0d}}{\partial \phi^2} = -\frac{1}{2}\frac{\frac{\partial \eta_{0d}}{\partial \phi}\xi_{0d} - \eta_{0d}^2}{\xi_{0d}^2} - \sum_{j=1}^{n_d} a_{dj}\varphi v_{0d}^2\mu_{0dj}^2,$$

$$\frac{\partial^2 \ell_{0d}}{\partial \varphi \partial \phi} = -\frac{1}{2}\frac{\frac{\partial \eta_{0d}}{\partial \varphi}\xi_{0d} - \eta_{0d}\eta_{0d\varphi}}{\xi_{0d}^2} + \sum_{j=1}^{n_d}\left\{a_{dj}v_{0d}\mu_{0dj} - a_{dj}v_{0d}y_{dj}\right\},$$

$$\frac{\partial^2 \ell_{0d}}{\partial \varphi^2} = \sum_{j=1}^{n_d}\left\{a_{dj}\varphi^{-1} - a_{dj}^2\dot{\psi}(a_{dj}\varphi)\right\} - \frac{1}{2}\frac{\frac{\partial \eta_{0d\varphi}}{\partial \varphi}\xi_{0d} - \eta_{0d\varphi}^2}{\xi_{0d}^2},$$

where $\dot{\psi}(z) = \frac{d\psi(z)}{dz}$ is the trigamma function.

For $r,s = 1,\dots,p$ the components of the score vector and the Hessian matrix are

$$U_{0r} = \sum_{d=1}^{D}\frac{\partial \ell_{0d}}{\partial \beta_r}, \quad U_{0p+1} = \sum_{d=1}^{D}\frac{\partial \ell_{0d}}{\partial \phi}, \quad U_{0p+2} = \sum_{d=1}^{D}\frac{\partial \ell_{0d}}{\partial \varphi},$$

$$H_{0rs} = H_{0sr} = \sum_{d=1}^{D}\frac{\partial^2 \ell_{0d}}{\partial \beta_s \partial \beta_r}, \quad H_{rp+1} = H_{p+1r} = \sum_{d=1}^{D}\frac{\partial^2 \ell_{0d}}{\partial \phi \partial \beta_r},$$

$$H_{rp+2} = H_{p+2r} = \sum_{d=1}^{D}\frac{\partial^2 \ell_{0d}}{\partial \varphi \partial \beta_r}, H_{0p+1p+1} = \sum_{d=1}^{D}\frac{\partial^2 \ell_{0d}}{\partial \phi^2},$$

$$H_{0p+1p+2} = H_{0p+2p+1} = \sum_{d=1}^{D}\frac{\partial^2 \ell_{0d}}{\partial \phi \partial \varphi}, \quad H_{0p+2p+2} = \sum_{d=1}^{D}\frac{\partial^2 \ell_{0d}}{\partial \varphi^2}.$$

In matrix form, we have $\boldsymbol{U}_0 = \boldsymbol{U}_0(\boldsymbol{\theta}) = \underset{1\leq r\leq p+2}{\text{col}}(U_{0rs})$ and $\boldsymbol{H}_0 = \boldsymbol{H}_0(\boldsymbol{\theta}) = (H_{0rs})_{r,s=1,\dots,p+2}$, where $\boldsymbol{\theta} = (\boldsymbol{\beta}^{\intercal}, \phi, \varphi)^{\intercal}$.

B Appendix

This appendix presents some additional results in the form of tables and figures.

Table 11: *Parameter estimates under Model 2 without two domains.*

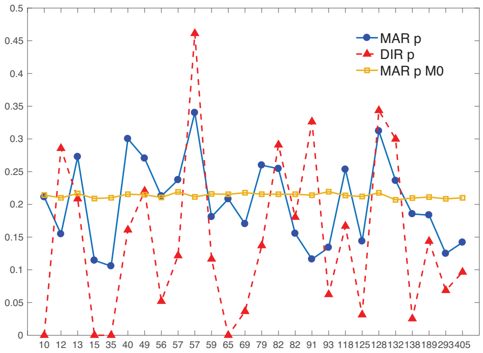
	estimate	standard error	p-value
$\hat{\beta}_0$	0.790	0.0137	$<$ 2E-16
$\hat{\beta}_1$	-0.142	0.0165	$<$ 2E-16
$\hat{\beta}_2$	0.134	0.0297	6.68E-06
$\hat{\phi}$	0.1081	0.0111	$<$ 2E-16
$\hat{\varphi}$	2.8269	0.0792	$<$ 2E-16

Table 12: *Parameter estimates under Model 0 without random effects.*

	estimate	standard error	p-value
$\hat{\beta}_0$	0.731	0.0130	$<$ 2E-16
$\hat{\beta}_1$	-0.158	0.0179	$<$ 2E-16
$\hat{\beta}_2$	0.151	0.0315	1.77E-06
$\hat{\nu}$	2.532		

Table 13: *Predictions and estimated MSEs for average incomes and poverty proportions under Model 2 (left) and Model 0 (right).*

c	n_d	N_d	$\overline{Y}_M$	mse	p_M	mse	$\overline{Y}_M^0$	mse	p_M^0	mse
27	82	124270	1.2496	0.00718	0.2545	0.00051	1.4581	0.06089	0.2155	0.00362
28	57	70944	1.2924	0.00951	0.2374	0.00072	1.4429	0.04642	0.2191	0.00328
29	69	225440	1.5033	0.00680	0.1702	0.00058	1.4554	0.05166	0.2178	0.00342
30	132	227463	1.2945	0.00583	0.2362	0.00040	1.4916	0.05851	0.2069	0.00304
31	56	166774	1.3584	0.01025	0.2127	0.00064	1.4776	0.06149	0.2107	0.00334
32	293	459626	1.6860	0.00243	0.1248	0.00019	1.4871	0.04135	0.2085	0.00286
33	128	268924	1.1260	0.00589	0.3121	0.00043	1.4505	0.05769	0.2176	0.00398
34	59	292243	1.4540	0.00947	0.1809	0.00068	1.4485	0.04983	0.2160	0.00341
3	57	87560	1.0701	0.01149	0.3401	0.00069	1.4739	0.05191	0.2114	0.00373
5	91	246942	1.7397	0.00503	0.1162	0.00039	1.4657	0.04944	0.2140	0.00345
6	82	179798	1.5543	0.00651	0.1555	0.00051	1.4591	0.04928	0.2155	0.00343
7	10	26007	1.3613	0.03368	0.2112	0.00219	1.4591	0.06097	0.2143	0.00382
11	118	189865	1.2552	0.00551	0.2534	0.00040	1.4683	0.05872	0.2137	0.00333
12	15	89136	1.7504	0.02317	0.1144	0.00140	1.4808	0.06417	0.2089	0.00405
13	138	187515	1.4426	0.00454	0.1853	0.00033	1.4804	0.04735	0.2097	0.00298
14	189	370540	1.4513	0.00347	0.1836	0.00026	1.4777	0.04929	0.2112	0.00297
15	405	771129	1.6043	0.00163	0.1419	0.00013	1.4792	0.03782	0.2100	0.00270
16	93	131337	1.6408	0.00536	0.1340	0.00043	1.4422	0.04166	0.2194	0.00322
17	12	33122	1.5576	0.03806	0.1547	0.00211	1.4776	0.05186	0.2100	0.00351
18	35	54545	1.8157	0.01318	0.1057	0.00090	1.4850	0.04623	0.2101	0.00329
20	125	256553	1.6029	0.00543	0.1437	0.00039	1.4736	0.04957	0.2121	0.00350
21	49	52958	1.2107	0.01308	0.2704	0.00090	1.4572	0.05194	0.2147	0.00338
22	13	33126	1.2050	0.03257	0.2727	0.00198	1.4573	0.05235	0.2167	0.00337
23	40	70642	1.1452	0.01513	0.3002	0.00118	1.4558	0.04925	0.2155	0.00342
24	65	80434	1.3719	0.00942	0.2082	0.00070	1.4588	0.05811	0.2155	0.00364
25	79	180619	1.2386	0.00833	0.2598	0.00060	1.4628	0.05700	0.2156	0.00342



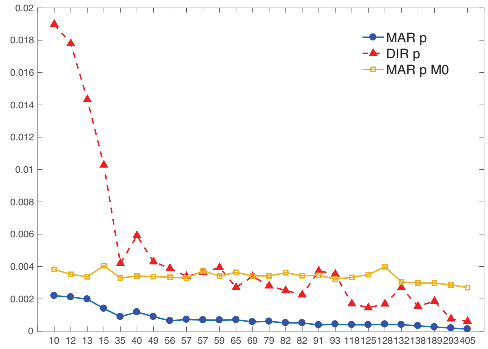


Figure 8: *Direct estimators and marginal predictors (under Model 2 and Model 0) of poverty proportions (left) and corresponding estimated MSEs (right).*