

A Cross-Entropy Approach to the Estimation of Generalised Linear Multilevel Models

Marco Bee,*

Giuseppe Espa, Diego Giuliani, Flavio Santi
Department of Economics and Management
University of Trento

October 14, 2016

Abstract

In this paper we use the cross-entropy method for noisy optimisation for fitting generalised linear multilevel models through maximum likelihood. We propose specifications of the instrumental distributions for positive and bounded parameters that improve the computational performance. We also introduce a new stopping criterion, which has the advantage of being problem-independent. In a second step we find, by means of extensive Monte Carlo experiments, the most suitable values of the input parameters of the algorithm. Finally, we compare the method to the benchmark estimation technique based on numerical integration. The cross-entropy approach turns out to be preferable from both the statistical and the computational point of view. In the last part of the paper, the method is used to model the probability of firm exits in the healthcare industry in Italy.

Keywords: multilevel logit, clustering effects, maximum likelihood estimation, adaptive Gaussian quadrature.

*The authors would like to thank four anonymous referees whose comments considerably improved an earlier version of the paper.

1 Introduction

Multilevel models, also known as mixed, hierarchical, or random coefficient models, have been developed to deal with hierarchically structured data. In general, a hierarchy can be defined as a set of units grouped at different levels (Goldstein, 2011, p. 1).

Consider, for example, a population of students from different schools. Students are grouped (or nested, or clustered) within schools and therefore, from a modelling point of view, it may be important to take into account the school they belong to. This empirical circumstance is characterised by a 2-level structure in which the population of students grouped within schools is a hierarchy where students are the level 1 units and schools are the level 2 units. If the location of schools is considered important as well, we may introduce a further level that takes into account the county each school belongs to: schools would then be grouped within counties, that would represent the level 3 units. This would be a 3-level structure.

It would be possible fitting a model based only on the first level, whilst neglecting the hierarchical structure of the data, but this may result in a loss of information. An early famous example in this regard is given by Aitkin et al. (1981) that analyse a previous study on the teaching styles used with elementary school children (Bennett, 1976). That study concluded that the children who were exposed to so-called “formal” styles of teaching exhibited more progress than those who were not. The analysis was carried out with standard multiple regression techniques based only on the individual children as the units of analysis. Their groupings within teachers and into classes were ignored, and the results showed statistically significant differences. Later on, Aitkin et al. (1981) performed a similar analysis accounting for the grouping of children into classes: the significant differences disappeared and the “formally” taught children turned out to be not different from the others. In other words, the statistical significance observed by Bennett (1976) was just the result of a misspecified model.

In the last three decades, multilevel models have become more and more important, with plenty of applications mostly in the social, biological and medical literatures; a comprehensive overview can be found, for example, in Goldstein (2011). The basic instance is the linear multilevel model, of which the Aitkin et al. (1981) analysis cited above is

one of the first applications. In linear multilevel models, estimation is not overly difficult: even though OLS estimator cannot be used because it generally leads to incorrect inference, maximum likelihood estimators (MLEs) can be obtained via iterative generalised least squares (see Goldstein, 2011, Chap. 2). This approach is relatively simple and robust, as the estimators are consistent even when the normality assumption is violated.

The extension to generalised linear multilevel models (GLMMs) is straightforward: as in the classical single-level case, the response variable to be modelled is discrete (often, but not necessarily, binary) rather than continuous. However, estimation procedures become more complicated. The likelihood function of GLMMs is derived from the probability density function of the model marginalized with respect to the unobserved random effects. Hence, the point values of the likelihood function result from an integration which, in general, cannot be handled analytically.

Although the estimation problem has been addressed in various ways (Goldstein, 2011), the most common solution employs numerical integration techniques for computing the likelihood and standard optimization routines for maximizing it. However, numerical integration methods become imprecise or even computationally prohibitive when the dimensionality of the problem gets larger, so that the performance of the process quickly deteriorates as the dimension increases.

In this paper we develop a cross-entropy approach for the MLE of GLMMs and investigate how the parameters of the algorithm affect the statistical properties of the estimators as well as the computational efficiency of the procedure. The goal consists in overcoming the main drawback of the approach based on numerical integration, that is the poor performance in presence of large-dimensional random effects. The cross-entropy technique is able to tackle high-dimensional problems and allows one to obtain both a good approximation and a computationally efficient procedure for maximizing the likelihood. Although the method can be successfully applied to fit any GLMM via MLE, here we focus on the logistic and the Poisson multilevel models, as they are two of the most relevant GLMMs in econometrics.

The cross-entropy method is a simulation-based technique for combinatorial and continuous optimization. It has first been proposed by Rubinstein (1997) in a rare-event

simulation setup; Rubinstein (1999) introduces a simple extension that permits to use it for optimization purposes. Rubinstein and Kroese (2004) give a comprehensive description of both theory and applications. The method can be considered a special case of importance sampling, where the instrumental distribution is chosen so as to minimise the cross-entropy with respect to the theoretically optimal distribution. The instrumental distribution is repeatedly updated such that it gets closer to the optimal sampling distribution, which usually depends on the parameters to be estimated.

The cross-entropy approach has a crucial advantage over Monte Carlo simulation: it simultaneously evaluates and optimises the objective function in case of both deterministic and noisy objective functions. The implication of this property is twofold. First, there is no need of a two-step (evaluation and optimisation) procedure. In principle, this should guarantee a computationally more efficient estimation process. Second, the maximisation of the likelihood does not rely on numerical algorithms for deterministic optimisation. This feature bypasses the problems originating from an approximate evaluation of the objective function. In fact, both numerical and Monte Carlo integration provide an estimation of the likelihood: if the approximation error is not small enough, the numerical optimisation procedures may not work properly.

With respect to the implementation of the cross-entropy method for noisy optimization, in this paper we introduce two major novelties. First, we propose to use lognormal instrumental distributions for positive parameters (variances) and logit-normal distributions for bounded parameters (correlations) in order to improve the computational performance of the algorithm. Second, we develop a new, problem independent, stopping criterion based on the Geweke's test (Geweke, 1992). The evidence of the simulation-based analysis suggests that the present implementation of the cross-entropy method outperforms the approach based on numerical integration from the computational as well as the statistical point of view.

The rest of the paper is organized as follows. Section 2 gives a formal definition of logistic and Poisson multilevel models and derives the corresponding likelihood functions. Section 3 describes and analyses in depth the cross-entropy approach to MLE of generalized linear multilevel models. In Section 4 the cross-entropy method is compared via Monte Carlo

simulation to the approach based on adaptive Gaussian quadrature (AGQ). In Section 5 an application to real data is provided. Finally, Section 6 concludes.

2 The model and the likelihood

In this paper we focus on the logistic and the Poisson multilevel models with random intercept and slopes. Formally, the 2-level logit model is defined as follows:

$$\begin{cases} Y_{ij} \sim \mathcal{B}(1, \pi_{ij}), \\ \pi_{ij} = \left[1 + e^{-\left(\sum_{s=1}^{k_1} \beta_s x_{ij}^{(s)} + \sum_{s=1}^{k_2} u_j^{(s)} x_{ij}^{(s)}\right)} \right]^{-1}, \end{cases} \quad (1)$$

where $x_{ij}^{(s)}$ is the i -th observation ($i = 1, \dots, n_j$) of the s -th fixed-level explanatory variable in the j -th level 2 group ($j = 1, \dots, G$), G is the number of level 2 groups, k_1 is the number of fixed-level explanatory variables and $k_2 \leq k_1$ is the number of random effects. Analogously, the 3-level model is given by:

$$\begin{cases} Y_{ijk} \sim \mathcal{B}(1, \pi_{ijk}), \\ \pi_{ijk} = \left[1 + e^{-\left(\sum_{s=1}^{k_1} \beta_s x_{ijk}^{(s)} + \sum_{s=1}^{k_2} u_{jk}^{(s)} x_{ijk}^{(s)} + \sum_{s=1}^{k_3} u_k^{(s)} x_{ijk}^{(s)}\right)} \right]^{-1}, \end{cases} \quad (2)$$

where now k_2 and k_3 are respectively the number of level 2 and level 3 random effects.

Similarly, the Poisson 2- and 3-level models are defined as:

$$\begin{cases} Y_{ij} \sim \text{Poi}(\lambda_{ij}), \\ \log \lambda_{ij} = \sum_{s=1}^{k_1} \beta_s x_{ij}^{(s)} + \sum_{s=1}^{k_2} u_j^{(s)} x_{ij}^{(s)}, \end{cases} \quad (3)$$

$$\begin{cases} Y_{ijk} \sim \text{Poi}(\lambda_{ijk}), \\ \log \lambda_{ijk} = \sum_{s=1}^{k_1} \beta_s x_{ijk}^{(s)} + \sum_{s=1}^{k_2} u_{jk}^{(s)} x_{ijk}^{(s)} + \sum_{s=1}^{k_3} u_k^{(s)} x_{ijk}^{(s)}. \end{cases} \quad (4)$$

An important feature of models (1)-(4) is that the random effects $u_j^{(s)}$ ($j = 1, \dots, G$, $s = 0, \dots, k_2$) are unobserved, so that they do not provide any information on the parameters of the model. This means that they should not appear in the likelihood: the marginalisation of the joint density $f_{(Y,U)}$ over the random effects $u_j^{(s)}$ permits to derive a likelihood function which is consistent with models (1)-(4) and depends only on the observed variables. Hence,

the likelihood function of (1) and (4) is:

$$\mathcal{L}(\beta, \Omega) = \int_{\mathbb{R}^{Gk_2}} f_{(Y,U)}(y, u; \beta, \Omega) du, \quad (5)$$

where $\Omega \in \mathbb{R}^{Gk_2 \times Gk_2}$ is the covariance matrix of the random effects $U \in \mathbb{R}^{G \times k_2}$, i.e. $\text{vec}(U) \sim \mathcal{N}_{Gk_2}(0, \Omega)$. The likelihood function (5) can be rewritten as follows:

$$\begin{aligned} \mathcal{L}(\beta, \Omega) &= \int_{\mathbb{R}^{Gk_2}} f_{(Y,U)}(y, u; \beta, \Omega) du = \int_{\mathbb{R}^{Gk_2}} f_{Y|U}(y|u; \beta, \Omega) f_U(u; \beta, \Omega) du = \\ &= \int_{\mathbb{R}^{Gk_2}} f_{Y|U}(y|u; \beta) f_U(u; \Omega) du = \\ &= \mathbb{E} (f_{Y|U}(y|U; \beta)) , \end{aligned} \quad (6)$$

where

$$f_{Y|U}(y|U; \beta) = \prod_{j=1}^G \prod_{i=1}^{n_j} \pi_{ij}^{y_{ij}} (1 - \pi_{ij})^{1-y_{ij}} \quad (7)$$

for (1) and

$$f_{Y|U}(y|U; \beta) = \prod_{j=1}^G \prod_{i=1}^{n_j} \frac{e^{-\lambda_{ij}}}{y_{ij}!} \lambda_{ij}^{y_{ij}} \quad (8)$$

for (4) and f_U is the probability density function of U .

Note that the expectation (6) is computed with respect to the matrix of random effects, so that it depends on Ω through the probability density function f_U used to compute the expectation. The maximisation of (6) cannot be handled analytically, and even the expected value cannot be derived in closed form, except for trivial models. It follows that MLE of model (1) requires a (typically not straightforward) estimation of the likelihood and its maximisation.

3 The Cross-Entropy Method

3.1 The Cross-Entropy Method for Noisy Optimisation

The original version of the cross-entropy method for estimating the tail probabilities consists on sampling iteratively from an instrumental distribution whose parameters are adjusted at each step, thus making the sampling distribution closer to the optimal importance

distribution. The name of the method derives from the fact that the “distance” is measured by the Kullback-Leibler divergence (also known as cross-entropy).

The noisy version of the cross-entropy method for optimisation is suitable for tackling the problem of maximising the likelihood (6). As illustrated in Rubinstein and Kroese (2004, pp. 204-205), the cross-entropy method for noisy optimisation aims at solving problems that can be stated as:

$$\max_{x \in \mathcal{D}} \mathbb{E}(S(x, U)), \quad (9)$$

where S is a real function, x is the vector with respect to which the function S is maximised, \mathcal{D} is the domain of S and U is the random vector that generates the noise on S .

We slightly generalise the definition of the noise component U given in Rubinstein and Kroese (2004), and assume that U is distributed according to the distribution function G which, in general, may depend on the point where S is evaluated. Hence, U can be defined as a spatial process $\{U_x\}$ over \mathcal{D} , where $U_{x_0} \sim G(x_0)$, for any $x_0 \in \mathcal{D}$. For notational simplicity, in the rest of the paper we will just write U instead of U_x . Finally, $F(\cdot; v)$ and $f(\cdot; v)$ are respectively the distribution function and density of the instrumental random variable that generates the points where S should be evaluated.

Following Rubinstein and Kroese (2004, p. 205), a pseudo-code of the cross-entropy method for noisy optimisation is as follows.¹

Algorithm 1 (Cross-Entropy Method for Noisy Optimisation).

1. Choose a starting value v_0 for the parameter vector v , N for the sample size, the rarity parameter ρ for the quantile estimation, and set $t := 1$;
2. Independently draw N vectors X_j ($j = 1, \dots, N$) from $F(v_{t-1})$ and N vectors U_j ($j = 1, \dots, N$) from $G(X_j)$;

¹Hereinafter, given a sample $\{X_1, \dots, X_n\}$, we indicate with $X_{(k)}$ the k -th order statistics, that is, the observation in the sample such that $X_{(1)} \leq X_{(2)} \leq \dots \leq X_{(k)} \leq \dots \leq X_{(n)}$.

Moreover, we use the floor function $\lfloor \cdot \rfloor$, the ceiling function $\lceil \cdot \rceil$, and the indicator function $\mathbb{1}_{\{\cdot\}}$, which are respectively defined as:

$$\lfloor x \rfloor \equiv \sup \{n \in \mathbb{N} : n \leq x\}, \quad \lceil x \rceil \equiv \inf \{n \in \mathbb{N} : n \geq x\}, \quad \mathbb{1}_{\{A\}} \equiv \begin{cases} 1 & \text{if } A \text{ is true} \\ 0 & \text{if } A \text{ is false} \end{cases}.$$

3. Compute the sample $\hat{S}_j \equiv S(X_j, U_j)$, for $j = 1, \dots, N$;

4. Compute the sample quantile $\hat{\gamma}_t \equiv \hat{S}_{(\lceil(1-\rho)N\rceil)}$;

5. Solve the following problem:

$$\max_v \frac{1}{N} \sum_{j=1}^N \mathbb{1}_{\{\hat{S}_j \geq \hat{\gamma}_t\}} \log f(X_j, v), \quad (10)$$

and determine the new vector v_t ;

6. If the stopping criterion is satisfied, stop; otherwise, set $t := t + 1$ and go back to step 2.

The interpretation of the steps of Algorithm 1 is rather intuitive:

Step 1 initialises the parameters of the procedure;

Steps 2 and 3 draw from $F(v_{t-1})$ the points X_j ($j = 1, \dots, N$), where the objective function S should be evaluated, and the noise U_j ($j = 1, \dots, N$) associated to each location X_j . Then the values $\mathbb{E}(S(X_j, U)|X_j)$ are estimated by $\hat{S}_j = S(X_j, U_j)$;

Steps 4 and 5 use the best performer points X_j , that is the $\lfloor \rho N \rfloor$ points associated with the $\lfloor \rho N \rfloor$ highest \hat{S}_j , for updating the parameters v of the distribution F . The solution of (10) is the vector v that minimises the Kullback-Leibler divergence (i.e. the cross-entropy) between the distribution function $F(v)$ and the optimal importance distribution. By so doing, at the next iteration, the algorithm will look for maximising points in the area of the domain where the highest values of \hat{S}_j were recorded;

Step 6 checks whether the algorithm has converged or not, according to some stopping criterion.

In the deterministic version of Algorithm 1 (see Rubinstein and Kroese, 2004, pp. 134-135) each marginal of F converges to the Dirac delta distribution as the number of iterations grows, whilst the point where the density mass is concentrated represents the solution of the optimisation problem. It can be proved that, under certain conditions, the cross-entropy algorithm almost surely converges to the maximum (Rubinstein and Kroese, 2004, 181-185). However, in the case of noisy optimisation, the convergence is not guaranteed. We discuss this point in the next section.

Before detailing Algorithm 1 for MLE of generalised linear 2-level models, note that the solution of (10) is the MLE of the parameters v based on a sample containing the points X_j such that $\hat{S}_j \geq \hat{\gamma}_t$. As stressed in Rubinstein and Kroese (2004, p. 38), when F belongs to the natural exponential family (Morris, 1982), it is always possible to solve (10) in closed form. Hence, since the family of F can be chosen by the researcher, it may be convenient to pick it out from the natural exponential family.

3.2 MLE through the Cross-Entropy Method

For the sake of simplicity, in the rest of the paper the condition $\Omega = \sigma^2 I$ is taken as the reference case. Possible extensions to the estimation of models with $\Omega \neq \sigma^2 I$ are discussed where necessary.

To apply the cross-entropy method for maximising the likelihood (6) when $\Omega = \sigma^2 I$, note that the function S in (9) corresponds to the conditional density function (7) for the logistic model and (8) for the Poisson model, whilst the expected value is computed with respect to U and maximised with respect to $\theta \equiv [\beta^\top \sigma^2]^\top$ over the parameter space $\Theta \subseteq \mathbb{R}^{k_1} \times \mathbb{R}^+$. Hence, if we define the function \mathcal{L}_c as $\mathcal{L}_c(\beta, U) \equiv f_{Y|U}(y|U; \beta)$, (9) becomes:

$$\max_{\theta \in \Theta} \mathbb{E}(\mathcal{L}_c(\beta, U)). \quad (11)$$

Note, however, that \mathcal{L}_c is not a likelihood function.

An important issue in the implementation of Algorithm 1 for MLE is the choice of the family of distribution functions $\mathcal{F} = \{F(v), v \in \Theta_{\mathcal{F}}\}$ to which the sampling distribution F belongs. The cross-entropy method for rare-event simulation can be considered as an adaptive importance sampling technique in its own right, as it iteratively adapts the sampling distribution until the Kullback-Leibler divergence between the actual and the optimal sampling distribution (which is unknown) is minimised (see Rubinstein and Kroese, 2004, Section 2.3). In such a context, the effectiveness and the efficiency of the simulation process are heavily affected by the theoretical minimal value of the Kullback-Leibler divergence between the sampling distributions in the parametric family and the optimal distribution. In particular, when the optimal distribution belongs to the parametric family (or, more precisely, when the closure of the parametric family with respect to the pointwise limit

includes the optimal distribution), the theoretical minimum Kullback-Leibler divergence is zero.

Although the version of the cross-entropy algorithm for function optimisation reverts the rationale of the rare-event simulation algorithm, it preserves the iterative adaptive mechanism of the sampling distribution, which is still unknown. However, unlike the simulation case, the choice of the family of the sampling distribution only affects the computational efficiency of the algorithm, provided that the minimal closed support of the family coincides with the domain of the objective function for any point in the (family's) parametric space.

In terms of implementation, as noted by Rubinstein and Kroese (2004, p. 187), the easiest way to generate the vectors $\theta_1, \dots, \theta_N$ (X_1, \dots, X_N in Algorithm 1) is to draw their components independently from 2-parameter distributions. This implies that the family \mathcal{F} should be parametrized by a vector $v \in \Theta_{\mathcal{F}} \subseteq \mathbb{R}^{2m}$.

In the rest of this section we discuss how the marginal instrumental distributions can be chosen, and how their parameters should be updated.

3.2.1 The instrumental distribution on the real line

The Gaussian distribution is the most natural choice when a component θ_j of θ has a real unbounded domain. Its parameters can be easily updated according to (10) as the MLEs are available in closed form.

It is worth noting that the convergence of a Gaussian distribution $\mathcal{N}(\mu_t, \sigma_t^2)$ to the Dirac delta distribution $\delta(x - \theta_j)$ is achieved when $\mu_t \rightarrow \theta_j$ and $\sigma_t^2 \rightarrow 0$ as $t \rightarrow \infty$. It follows that the point estimate of θ_j according to the cross-entropy method should be $\hat{\theta}_j = \mu_t$.

3.2.2 The instrumental distribution on the positive real line

When a real parameter θ_j is constrained to be positive, we suggest to choose the log-normal distribution, as it shares the advantages of the Gaussian distribution. For an iid sample X_1, \dots, X_n drawn from $\log\mathcal{N}(\mu, \sigma^2)$, the closed-form MLEs are:

$$\hat{\mu}_n = \frac{1}{n} \sum_{j=1}^n \log X_j, \quad \hat{\sigma}_n^2 = \frac{1}{n} \sum_{j=1}^n (\log X_j - \hat{\mu}_n)^2,$$

and the convergence to the Dirac delta distribution $\delta(x - \theta_j)$ is reached for $\hat{\mu}_t \rightarrow \log \theta_j$ and $\hat{\sigma}_t^2 \rightarrow 0$ as $t \rightarrow \infty$. As for the Gaussian distribution, the cross-entropy point estimator corresponds to the estimated expected value, thus for the log-normal distribution we have $\hat{\theta}_j = e^{\hat{\mu}_t - \hat{\sigma}_t^2/2}$.

3.2.3 The instrumental distribution on a bounded interval

When $\Omega = \sigma^2 I$ is a too restrictive condition, it is necessary to adopt a more general (and flexible) parametrisation of the covariance matrix Ω . We may distinguish two possible cases.

First, the elements of U are independent, but heteroskedastic, that is

$$\Omega = \text{diag}(\sigma_1^2 I_{h_1}, \dots, \sigma_s^2 I_{h_s}) \in \mathbb{R}^{Gk_2 \times Gk_2}$$

with $1 \leq s \leq Gk_2$, $I_h \in \mathbb{R}^{h \times h}$, and $\sum_{j=1}^s h_j = Gk_2$ (when $s = 1$ we have $\Omega = \sigma^2 I_{Gk_2}$). In this case, the only difference with respect to the method illustrated in the previous subsection is that there are more parameters to be estimated and s of them (instead of only one) should be positive.

Second, Ω is not diagonal. In this case, both variances and covariances have to be estimated, and the estimates should satisfy the requirement of positive definiteness of Ω . In this setting it is more convenient to reparametrise Ω such that we estimate its diagonal elements (that is, the variances) and the correlation coefficients. In this manner we have two homogeneous sets of parameters: the positive ones (variances) and the ones taking values in $[-1, 1]$ (the correlations).

Most of the families of probability distributions with bounded support are defined on $[0, 1]$. Nevertheless, by means of the affine transformation $f(x) = 2x - 1$, their support changes to $[-1, 1]$. Hence, the instrumental distribution for the correlation parameters in θ can be one of the numerous distributions supported on $[0, 1]$. The beta distribution is often used in similar cases (see Rubinstein and Kroese, 2004). However, the MLEs of its parameters cannot be obtained in closed form, so that numerical procedures are necessary in order to solve the problem (10) and update the parameters.

A feasible alternative to the beta is represented by the logit-normal distribution (Frederic and Lad, 2008), whose support is $[0, 1]$. A random variable W is logit-normally dis-

tributed, that is $W \sim \text{logit}\mathcal{N}(\mu, \sigma^2)$, if:

$$\log \frac{W}{1-W} \sim \mathcal{N}(\mu, \sigma^2),$$

and it can be verified that its probability density function is:

$$f(w) = \frac{1}{\sqrt{2\pi} \sigma w(1-w)} e^{-\frac{1}{2\sigma^2} (\log \frac{w}{1-w} - \mu)^2}, \quad w \in [0, 1],$$

where $\mu \in \mathbb{R}$ and $\sigma \in \mathbb{R}^+$.

Although the expected value and the variance of the logit-normal distribution cannot be derived analytically, the MLEs are available in closed form. Given an iid sample W_1, \dots, W_n from $\text{logit}\mathcal{N}(\mu, \sigma^2)$, the MLEs are:

$$\hat{\mu}_n = \frac{1}{n} \sum_{j=1}^n \log \frac{W_j}{1-W_j}, \quad \hat{\sigma}_n^2 = \frac{1}{n} \sum_{j=1}^n \left(\log \frac{W_j}{1-W_j} - \hat{\mu}_n \right)^2.$$

The density of $X \equiv 2W - 1$ is:

$$f(x) = \frac{\sqrt{2/\pi}}{\sigma(1-x^2)} e^{-\frac{1}{2\sigma^2} (\log \frac{1+x}{1-x} - \mu)^2}, \quad x \in [-1, 1], \quad (12)$$

and the MLEs of μ and σ^2 for an iid sample X_1, \dots, X_n from (12) are:

$$\hat{\mu}_n = \frac{1}{n} \sum_{j=1}^n \log \frac{1+X_j}{1-X_j}, \quad \hat{\sigma}_n^2 = \frac{1}{n} \sum_{j=1}^n \left(\log \frac{1+X_j}{1-X_j} - \hat{\mu}_n \right)^2.$$

The random variable X with probability density function (12) converges to the Dirac delta distribution $\delta(x - \theta_j)$ when $\hat{\mu}_t \rightarrow \log \frac{1+\theta_j}{1-\theta_j}$ and $\hat{\sigma}_t^2 \rightarrow 0$ as $t \rightarrow \infty$. Unlike the normal and the log-normal distributions, we cannot express the cross-entropy point estimator in closed form, since the expected value of the logit-normal distribution cannot be derived analytically. Still, it can be estimated numerically, once the algorithm has converged.

The advantage in using the logit-normal instead of the beta distribution is computational, as the latter requires to update numerically the parameters at each iteration of Algorithm 1, whilst in the former the numerical estimation of the expected value is performed only once, when the algorithm has converged. Moreover, sampling the instrumental distribution for the correlation coefficients is straightforward, since the random variable X is distributed according to (12) when its stochastic representation is given by:

$$X \equiv 2(1 + e^{-\mu - \sigma Z})^{-1} - 1,$$

and $Z \sim \mathcal{N}(0, 1)$.

The estimation procedure just described for non-diagonal matrices Ω guarantees that the variances are positive and that each covariance satisfies the Cauchy-Schwarz inequality. Nevertheless, there is the possibility of obtaining estimated matrices $\hat{\Omega}$ that are not positive definite because of estimation errors. A possible solution consists in finding the closest positive-definite matrix with respect to some matrix norm, according to the method suggested by Higham (2002) and implemented in the function `nearPD` of the `Matrix` (Bates and Mächler, 2015) package of the statistical software R (R Core Team, 2015).

An alternative solution² to the problem of the negative definiteness of $\hat{\Omega}$ would be a reparametrisation of Ω by means of a lower triangular matrix $L \in \mathbb{R}^{Gk_2 \times Gk_2}$ such that $LL^T = \Omega$. Note, however, that in this case it is necessary to include into the optimisation process the restrictions which ensure that the matrix LL^T is consistent with the desired structure of Ω , which may be either diagonal, block-diagonal, homoskedastic, and so on. See Zhang (2013) and Bates et al. (2015).

3.2.4 Updating the parameters of the instrumental distributions

The last issue analysed in this section is the update rule of the vector v_t . Point 5 of Algorithm 1 does not prescribe that the solution of (10) has to be automatically taken as the new value v_t for the parameter vector of F , because in some circumstances it is advisable to smooth out the sequence $\{v_t\}$.

Rubinstein and Kroese (2004, p. 189) note that, if the convergence of the sampling distribution F is too quick, the algorithm may converge to a local maximum, especially if the likelihood function is particularly noisy or irregular. In order to avoid this problem, it may be useful to smooth out the sequence $\{v_t\}$, in particular the components that affect the dispersion of the sampling distribution. According to Rubinstein and Kroese (2004),

²We would like to thank an anonymous referee for this suggestion.

the smoothing rules for the location and scale parameters in v_t are:

$$v_{t,h} = \alpha \hat{v}_{t,h} + (1 - \alpha) v_{t-1,h}, \quad (13)$$

$$v_{t,h} = \beta_t \hat{v}_{t,h} + (1 - \beta_t) v_{t-1,h}, \quad (14)$$

$$\beta_t = \beta \left(1 - \frac{1}{t}\right)^q, \quad (15)$$

where $h \in \{1, \dots, 2m\}$ is the index of the component of v_t , \hat{v}_t is the vector of MLEs obtained by solving (10), and α , β and q are some constants such that $\alpha \in (0.7, 1]$, $\beta \in [0.8, 0.99]$, and $q \in \{5, 6, \dots, 10\}$.

The values of α , β and q affect the performances of the algorithm and the properties of the estimators. In general, as α gets closer to 1 the components of v_t are less smooth and the algorithm is free to explore the whole parameter space Θ ; as α gets smaller, the smoothness is higher and the convergence is expected to be faster. When β decreases and/or q increases the sequence of dispersion parameters in v_t get smoother, and this typically results in a slower reduction of the region of Θ explored by the algorithm. On the contrary, high values of β and small values of q increase the speed of convergence of the algorithm.

3.3 The Stopping Criterion

Unlike deterministic optimisation problems, when the objective function is noisy the convergence of the optimisation procedures cannot be assessed with respect to a fixed value of either the objective function or the solution. In particular, the stopping rule of the algorithm must verify whether the process of either the estimated solutions $\{\hat{x}_t^*\}$ or of the estimated maximum $\{\hat{S}(\hat{x}_t^*)\}$ has become stationary.

Since stationarity is more easily assessed for a univariate than a multivariate process, Rubinstein and Kroese (2004, 206) suggest to test the convergence of the sample quantile process $\{\hat{\gamma}_t\}$. From the statistical point of view, this stopping criterion is equivalent to testing the stationarity of the estimated maximum of the likelihood $\hat{\mathbb{E}}(\mathcal{L}_c(\hat{\beta}_t, U))$. However, working with the series $\{\hat{\gamma}_t\}$ is more convenient, as it is a by-product of Point 4 of Algorithm 1.

In this paper we formally test the convergence of the cross-entropy algorithm by means of the Geweke's test (Geweke, 1992), often used for testing the convergence of Markov

chains in Monte Carlo Markov Chain (MCMC) methods. According to this approach, the stationarity of a Markov chain is assessed by comparing the spectra of the first and the last portion of the chain: when they are not statistically different, the null hypothesis of stationarity is not rejected. The outcome of the test considers the effect of the size of each portion of the chain as well as their distance.

In the MCMC literature, where the chains usually consist of several thousands of observations, the Geweke’s test is applied to the whole chain and compares the first 10% and the last 50% of the observations. In the cross-entropy method, we typically rely on a much smaller sample, so that some modifications are needed.

We compute the test on a sliding window of $\{\hat{\gamma}_t\}$ defined as follows:

$$\{\hat{\gamma}_{1+\lfloor(1-v_b)(t-n_b)\rfloor}, \dots, \hat{\gamma}_{t-1}, \hat{\gamma}_t\}, \quad \text{for } t > n_b, \quad (16)$$

where n_b determines the minimum sample size (i.e. the test is performed only after n_b iterations of the algorithm), and $v_b \in (0, 1]$ measures how rapidly the sample size grows. This may be clarified by noting that the number of elements in (16) is approximately $v_b t - n_b(1 - v_b)$. In our setup, the test comparison is carried out on the first 30% and the last 40% of the window (16), and the algorithm stops when the p -value is smaller than a specified threshold.

The sliding window allows to discard the first observations, which are far from the solution, whilst its growing size guarantees that the test is asymptotically valid. The percentages are here modified in order to use more observations of the sample (70% instead of 60%) and consider the updating effect of the sliding window (the first portion is 30% rather than 10%).

The reason why we propose the Geweke’s test in place of the stopping criteria suggested in Rubinstein and Kroese (2004) is that the former is problem-independent, and takes into account the fact that $\hat{\gamma}_t$ may not (generally) converge to some fixed value when the maximum of the likelihood has been reached.

In fact, both in case of non-noisy and noisy optimisation, $\{\hat{\gamma}_t\}$ can be considered as a discrete-time stochastic process. Nonetheless, in the former case, if the cross-entropy algorithm is indefinitely iterated, $\{\hat{\gamma}_t\}$ converges in probability to some constant value. On the other hand, in the latter case, $\{\hat{\gamma}_t\}$ converges to some stationary distribution, as the

noise on the objective function hampers the convergence to a constant value.

This implies that any “good” stopping criterion should check whether $\{\hat{\gamma}_t\}$ has converged to the stationary distribution. The Geweke’s test complies with such requirement, and the stopping criterion based on the p -value of the test makes it problem-independent.

3.4 Setting the Parameters of the Algorithm

The implementation of Algorithm 1 requires to set several parameters and distributions, some of which may be problem-specific. Algorithm 2 details Algorithm 1 for MLE of generalised linear 2-level models. We assume that the covariance matrix Ω is parametrised by s variance parameters σ_j^2 ($j = 1, \dots, s$) and r correlation parameters ρ_j ($j = 1, \dots, r$), that is:

$$\Omega = \Omega(\sigma_1^2, \dots, \sigma_s^2, \rho_1, \dots, \rho_r).$$

In general, we may have that $s \leq Gk_2$ and $r \leq Gk_2(Gk_2 - 1)/2$, as some elements in Ω may be restricted to be equal, or some correlations may be restricted to be zero.

Algorithm 2 (Cross-Entropy Estimation of generalised linear 2-level models).

1. Choose starting values v_0 for the parameter vector v , set the sample size N , the rarity ρ for the quantile estimation and the threshold `pvalue` of the Geweke’s test;
2. Draw N iid vectors $\theta_1, \dots, \theta_N$ as follows:

$$\begin{aligned} \theta_{j,h} &\sim \mathcal{N}(v_{t-1,2h-1}, v_{t-1,2h}) && \text{for } h = 1, \dots, k_1, \\ \theta_{j,h} &\sim \log\mathcal{N}(v_{t-1,2h-1}, v_{t-1,2h}) && \text{for } h = k_1 + 1, \dots, k_1 + s, \\ \theta_{j,h} &\sim \text{logit}\mathcal{N}(v_{t-1,2h-1}, v_{t-1,2h}) && \text{for } h = k_1 + s + 1, \dots, k_1 + s + r, \end{aligned}$$

(where $\theta_{j,h}$ is the h -th element of θ_j and $j = 1, \dots, N$), then draw N iid matrices $U_j \in \mathbb{R}^{G \times k_2}$ such that:

$$\text{vec}(U_j) \sim \mathcal{N}_{Gk_2} \left(0, \Omega(\theta_{j,k_1+1}, \dots, \theta_{j,k_1+s}, \theta_{j,k_1+s+1}, \dots, \theta_{j,k_1+s+r}) \right), \quad j = 1, \dots, N.$$

3. Compute the sample $\hat{\mathcal{L}}_j \equiv \mathcal{L}_c(\theta_j, U_j)$, for $j = 1, \dots, N$;
4. Compute the sample quantile $\hat{\gamma}_t \equiv \hat{\mathcal{L}}_{(\lceil(1-\rho)N\rceil)}$;

5. Compute:

$$\hat{v}_{t,2h-1} = \frac{1}{\lfloor \rho N \rfloor} \sum_{j=1}^N \mathbb{1}_{\{\hat{\mathcal{L}}_j \geq \hat{\gamma}_t\}} \theta_{j,h},$$

$$\hat{v}_{t,2h} = \frac{1}{\lfloor \rho N \rfloor} \sum_{j=1}^N \mathbb{1}_{\{\hat{\mathcal{L}}_j \geq \hat{\gamma}_t\}} (\theta_{j,h} - \hat{v}_{t,2h-1})^2,$$

for $h = 1, \dots, k_1$, then:

$$\hat{v}_{t,2h-1} = \frac{1}{\lfloor \rho N \rfloor} \sum_{j=1}^N \mathbb{1}_{\{\hat{\mathcal{L}}_j \geq \hat{\gamma}_t\}} \log \theta_{j,h},$$

$$\hat{v}_{t,2h} = \frac{1}{\lfloor \rho N \rfloor} \sum_{j=1}^N \mathbb{1}_{\{\hat{\mathcal{L}}_j \geq \hat{\gamma}_t\}} (\log \theta_{j,h} - \hat{v}_{t,2h-1})^2,$$

for $h = k_1 + 1, \dots, k_1 + s$, and finally:

$$\hat{v}_{t,2h-1} = \frac{1}{\lfloor \rho N \rfloor} \sum_{j=1}^N \mathbb{1}_{\{\hat{\mathcal{L}}_j \geq \hat{\gamma}_t\}} \log \frac{1 + \theta_{j,h}}{1 - \theta_{j,h}},$$

$$\hat{v}_{t,2h} = \frac{1}{\lfloor \rho N \rfloor} \sum_{j=1}^N \mathbb{1}_{\{\hat{\mathcal{L}}_j \geq \hat{\gamma}_t\}} \left(\log \frac{1 + \theta_{j,h}}{1 - \theta_{j,h}} - \hat{v}_{t,2h-1} \right)^2,$$

for $h = k_1 + s + 1, \dots, k_1 + s + r$.

Determine the new vector v_t according to (13) (for $h = 1, \dots, k_1, k_1 + s + 1, \dots, k_1 + s + r$) and (14) (for $h = k_1 + 1, \dots, k_1 + s$);

6. If $t > n_b$, perform the Geweke's test on the sample (16). If the p -value is higher than $pvalue$, stop; otherwise, set $t := t + 1$ and go back to step 2. If $t \leq n_b$, set $t := t + 1$ and go back to step 2.

The generalisation of Algorithm 2 to k -level models is straightforward, as the integrand \mathcal{L}_c in (11) would depend on $k - 1$ matrices of random effects (one for each level but the lowest), and problem (11) would be:

$$\max_{\theta \in \Theta} \mathbb{E} (\mathcal{L}_c (\beta, U^{(2)}, \dots, U^{(k)})) .$$

It follows that Point 2 of Algorithm 2 would require the simulation of $k - 1$ matrices of random effects for each of the N multilevel models, and it would make necessary to parametrise $k - 1$ covariance matrices $\Omega^{(i)}$ for each $\text{vec}(U^{(i)})$, $i = 1, \dots, k$.

The parameters of Algorithm 2 affect its performance and the precision of the estimators. In the remaining part of this section we illustrate the effect of each parameter and suggest how they should be set.

3.4.1 Number of points N and rarity parameter ρ

N is the number of points of Θ where the function \mathcal{L}_c is evaluated at each iteration. The larger is the value of N , the better is the exploration of Θ and the smaller is the probability that the algorithm converges to a local maximum. On the other hand, a large N implies a heavier computational burden.

The rarity parameter ρ determines the fraction of the N evaluations of the likelihood function $\mathbb{E}(\mathcal{L}_c(\beta, U))$ used for updating the parameters of the instrumental distribution. In particular, $\lfloor \rho N \rfloor$ is the number of best-performers points on Θ used at Point 5 of Algorithm 2.

Typical values of ρ are between 0.01 (for large sample sizes) and $\log(N)/N$ for $N < 100$ (Rubinstein and Kroese, 2004, p. 45). In general, as the rarity parameter gets closer to zero, the probability that the algorithm converges to a local maximum increases. However, the effect of ρ on the statistical properties of the estimators and the computational performance of the algorithm is strictly related to the value of N and the characteristics of the optimisation problem. When N is relatively large, a value of ρ close to zero can speed up the convergence of the algorithm, as it reduces more rapidly the region of Θ where the solution is sought. On the other hand, as ρ gets larger, the probability that the algorithm converges to a local maximum is smaller. This is especially important if the objective function is very noisy.

Accordingly, the relevance of the numerical value of ρ is particularly marked when N is not large (say, about less than 1000) because, as already noticed, a small value of N implies a poor exploration of the parameter space Θ . Although a higher value of ρ may reduce the probability of convergence to a local maximum, for ρ close to or greater than 0.1 this probability gets larger again. This happens because a small value of N along with a high value of ρ results in a slow convergence of the algorithm and a high variance of $\{\hat{\gamma}_t\}$. Intuitively, in this case, the stopping criterion may consider this high variance as the

stationary variance of the process $\{\hat{\gamma}_t\}$ and stop the algorithm too early.

2-level logistic model. To investigate the impact of N and ρ we have performed a Monte Carlo experiment. The simulated models are of type (1), with $n = \sum_{j=1}^G n_j = 250$ observations randomly clustered in $G = 3$ groups. The number of regressors is $k_1 = 4$ (including a constant), thus $X \in \mathbb{R}^{n \times 4}$, the vector of coefficients is $\beta = [1, 2, 3, 4]^T$, and there is a random effect on each component of β , that is, $k_2 = k_1 = 4$. Both X and the grouping of the observations are the same for all the models. The matrix U is distributed as $\text{vec}(U) \sim \mathcal{N}_{12}(0, \sigma^2 I)$, where $\sigma = 0.1$, whilst the elements of X are iid standard Gaussian. Although the variance of the random effect is the same for all coefficients, it affects them differently, as they have different values. The rarity parameter ρ takes values in $\{0.01, 0.02, \dots, 0.10\}$, whilst N has been chosen in $\{100, 180, 320, 560, 1000, 1800, \dots, 56000\}$.³ For each pair (N, ρ) , the number of replications is equal to 50. The cross-entropy algorithm uses $\alpha = 1$, $\beta = 0.975$, $q = 10$, `pvalue` = 0.1, $n_b = 20$ and $v_{n_b} = 0.5$.

All the simulations are implemented and performed in R 3.2.1 (R Core Team, 2015) on a dedicated server with an Intel® Core™ i3-2130 CPU (3.4 GHz, dual-core, 4 threads, 64 b), RAM 8 GB, operating system Debian 7.9 (Wheezy, old-stable). The hardware and software features of the machine guarantee that the actual computational capacity is stable over time (e.g. it is not affected by energy-saving devices implemented on all personal computers), and this makes it possible to obtain reliable measures of computing times.

Complete results are reported in Tables SM.1-SM.6 of the supplementary material, and basically confirm the remarks above about the relationship of N and ρ with the performance of the algorithm.

Figure 1 shows how the sample root mean squared error (RMSE) for $\hat{\beta}_1$ varies with N for various values of the rarity parameter ρ . It can be noticed that the RMSE of $\hat{\beta}_1$ decreases as the number of evaluations of the objective function increases, but the reduction is small for $N \geq 1000$, especially when $\rho \leq 0.7$.

Figure 2 displays the relation between ρ and the RMSE of $\hat{\beta}_1$ for some values of N . As expected, the RMSE tends to get larger when ρ is close either to zero or to 0.1: in the

³The values N_j of N are such that $\log N_j$ are roughly evenly spaced over \mathbb{R} . In fact, each value N_j is approximately 78% larger than its predecessor N_{j-1} .

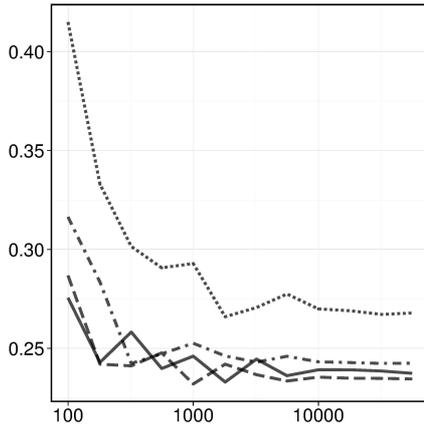


Figure 1: Sample RMSE of $\hat{\beta}_1$ as a function of N for $\rho = 0.01$ (solid line), $\rho = 0.04$ (dashed line), $\rho = 0.07$ (dot-dashed line), and $\rho = 0.10$ (dotted line). The abscissa axis is scaled logarithmically.

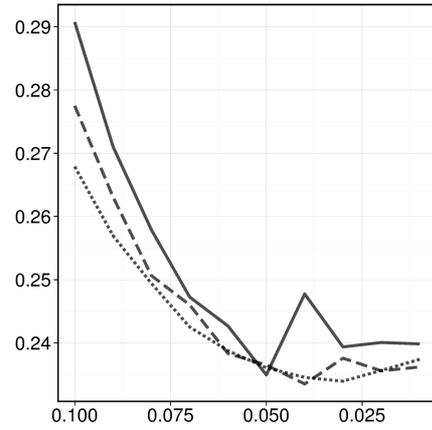


Figure 2: Sample RMSE of $\hat{\beta}_1$ as a function of the rarity parameter ρ for $N = 560$ (solid line), $N = 5600$ (dashed line), and $N = 56000$ (dotted line). The abscissa axis is reverse.

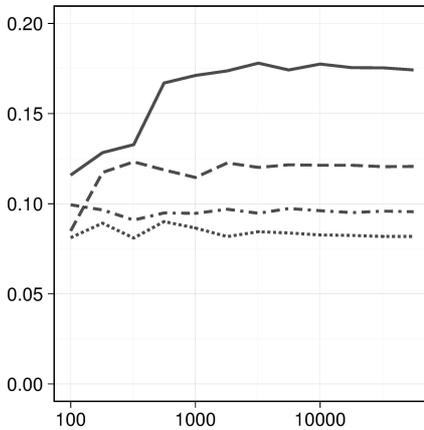


Figure 3: Sample RMSE of $\hat{\sigma}$ as a function of N for $\rho = 0.01$ (solid line), $\rho = 0.04$ (dashed line), $\rho = 0.07$ (dot-dashed line), and $\rho = 0.10$ (dotted line). The abscissa axis is scaled logarithmically.

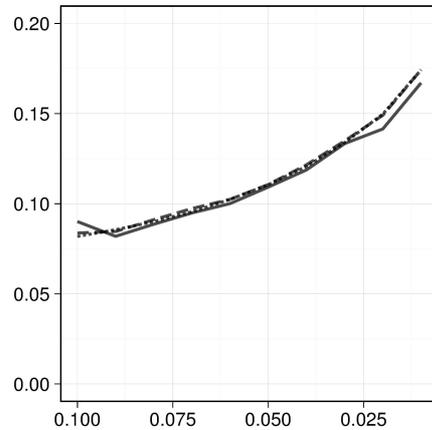


Figure 4: Sample RMSE of $\hat{\sigma}$ as a function of the rarity parameter ρ for $N = 560$ (solid line), $N = 5600$ (dashed line), and $N = 56000$ (dotted line). The abscissa axis is reverse.

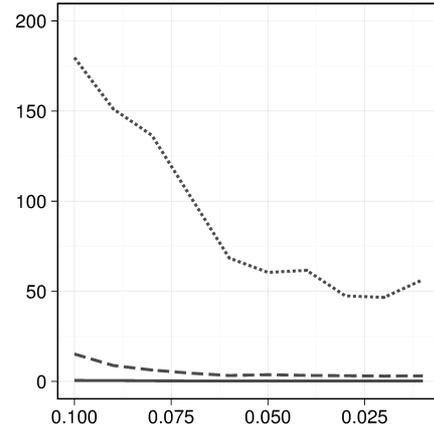
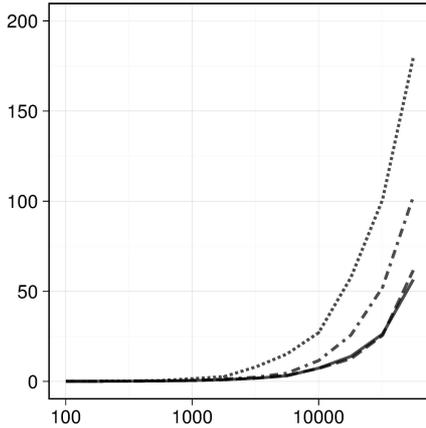


Figure 5: Computing times as a function of N for $\rho = 0.01$ (solid line), $\rho = 0.04$ (dashed line), $\rho = 0.07$ (dot-dashed line), and $\rho = 0.10$ (dotted line). The abscissa axis is scaled logarithmically. **Figure 6:** Computing times as a function of the rarity parameter ρ for $N = 560$ (solid line), $N = 5600$ (dashed line), and $N = 56000$ (dot-dashed line). The abscissa axis is reverse.

former case the algorithm converges (too rapidly) to a sub-optimal point, whereas in the latter the convergence is not complete, as the stopping criterion is triggered too early. This remark explains the shape of the curves in Figure 2, yet it is interesting to notice that the estimates worsen more markedly as ρ approaches 0.1 than when ρ is close to zero.

The shapes of the curves in Figures 1 and 2 describe the qualitative behaviour of all the estimators $\hat{\beta}_j$ ($j = 1, \dots, 4$), but the estimator $\hat{\sigma}^2$ behaves differently. In particular, as Figures 3 and 4 show, the RMSE of $\hat{\sigma}^2$ tends to decrease as ρ diverges. It follows that ρ should be set as large as possible so as to minimise the RMSE of $\hat{\sigma}^2$. Moreover, the RMSE of $\hat{\sigma}^2$ is basically independent from N as ρ increases, and it is rather stable (for any ρ) when $N \geq 1000$.

Figures 5 and 6 show the computing times of the estimation procedure. Unsurprisingly, they are heavily affected by N and grow as it diverges. However, they increase faster as ρ gets larger: the slope of the curve between N and the average computing times for $\rho = 0.1$ is more than three times larger than the slope for $\rho = 0.01$. On the other hand, as N gets smaller, the effect of ρ tends to get weaker. This pattern results from a positive asymmetry in the distribution of the computing times which gets stronger as N and ρ increase.

The results of the Monte Carlo simulations suggest to choose $N = 1000$, since larger

values of N lead to longer computing times and do not result in lower RMSE.

Unlike N , the optimal value of ρ cannot be easily identified. The RMSE of the estimators $\hat{\beta}_j$ ($j = 1, \dots, 4$) is usually minimised for ρ between 0.02 and 0.03. On the other hand, the RMSE of $\hat{\sigma}^2$ suggests to choose ρ as large as possible. Finally, Figures 5 and 6 show that smaller values of ρ are associated to shorter computing times.

In summary, there is a sort of trade-off between the accuracy and the precision of the estimates of β_j ($j = 1, \dots, k_1$), and the quality of the estimates of σ^2 . Nevertheless, it seems that a reasonable compromise between these two needs can be achieved by setting $\rho = 0.04$. Although this value is suboptimal in terms of RMSE both for $\hat{\beta}_j$ and $\hat{\sigma}^2$, it results in a negligible increase of the RMSE of $\hat{\beta}_j$, whilst the difference between the RMSE of $\hat{\sigma}^2$ in $\rho = 0.04$ and the minimum RMSE (in $\rho = 0.1$) is a minor portion of the total variation of the RMSE in $(0.01, 0.1)$.

2-level Poisson model. In order to verify how N and ρ affect the cross-entropy estimators of the Poisson model, we have performed another simulation. In particular, the experiment focuses on 2-level Poisson models of type (4), with $n = \sum_{j=1}^G n_j = 250$ observations randomly clustered in $G = 3$ groups. Similarly to the previous simulations, we set $k_1 = k_2 = 4$, but in this case $\beta = [-0.3, 0.2, 0.4, 0.8]^T$. Both X and the grouping of the observations are the same for all the models. U is distributed as $\text{vec}(U) \sim \mathcal{N}_{12}(0, \sigma^2 I)$, where $\sigma = 0.1$, whilst the elements of X are iid standard Gaussian. The rarity parameter ρ takes values in $\{0.01, 0.02, \dots, 0.10\}$, whilst $N \in \{560, 1000, 1800, 3200, 5600, 10000\}$. For each pair (N, ρ) , the number of replications is equal to 50. The cross-entropy algorithm uses $\alpha = 1$, $\beta = 0.975$, $q = 10$, $\text{pvalue} = 0.1$, $n_b = 20$ and $v_{n_b} = 0.5$. The complete results of the experiments are reported in Tables SM.7-SM.12, whereas Figures 7-12 summarise the most meaningful outcomes.

The effect of N and ρ on the cross-entropy estimators of the β 's is basically the same found for the logistic model. As Figures 7-8 show, the RMSE of $\hat{\beta}_1$ is roughly constant when N exceeds 1000, whereas it reaches its minimum when ρ is between 0.02 and 0.04. An analogous behaviour is observed for the RMSE of $\hat{\beta}_2$, $\hat{\beta}_3$, and $\hat{\beta}_4$.

A remarkable difference between the results of the logistic model and the Poisson model is observed for the estimation of σ and the computing times. As Figures 9-12 show, the

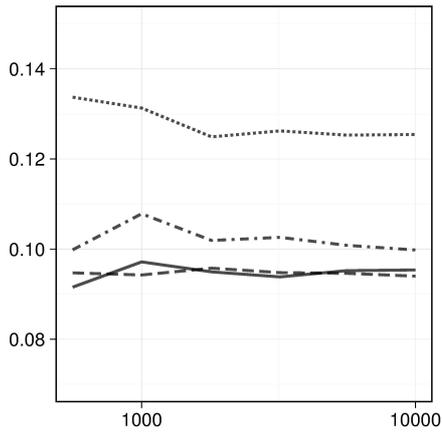


Figure 7: Sample RMSE of $\hat{\beta}_1$ as a function of N for $\rho = 0.01$ (solid line), $\rho = 0.04$ (dashed line), $\rho = 0.07$ (dot-dashed line), and $\rho = 0.10$ (dotted line). The abscissa axis is scaled logarithmically.

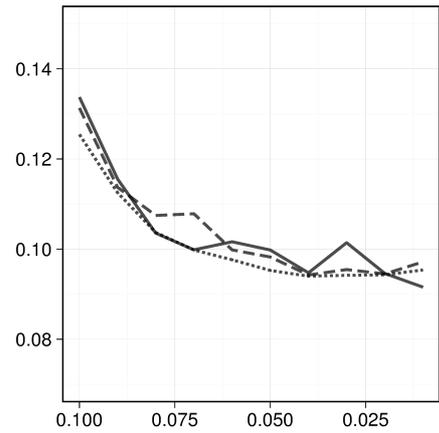


Figure 8: Sample RMSE of $\hat{\beta}_1$ as a function of the rarity parameter ρ for $N = 560$ (solid line), $N = 5600$ (dashed line), and $N = 56000$ (dotted line). The abscissa axis is reverse.

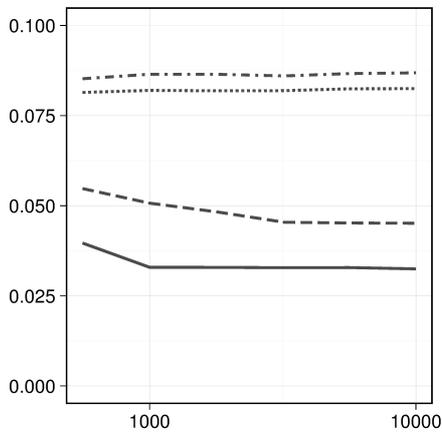


Figure 9: Sample RMSE of $\hat{\sigma}$ as a function of N for $\rho = 0.01$ (solid line), $\rho = 0.04$ (dashed line), $\rho = 0.07$ (dot-dashed line), and $\rho = 0.10$ (dotted line). The abscissa axis is scaled logarithmically.

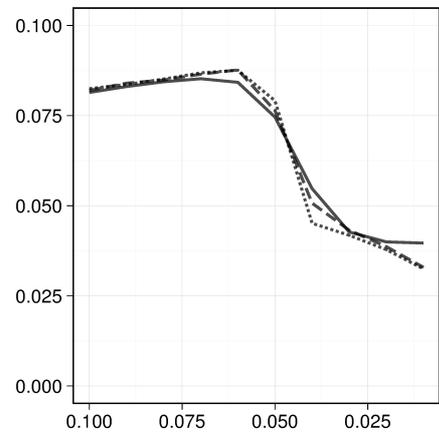


Figure 10: Sample RMSE of $\hat{\sigma}$ as a function of the rarity parameter ρ for $N = 560$ (solid line), $N = 5600$ (dashed line), and $N = 56000$ (dotted line). The abscissa axis is reverse.

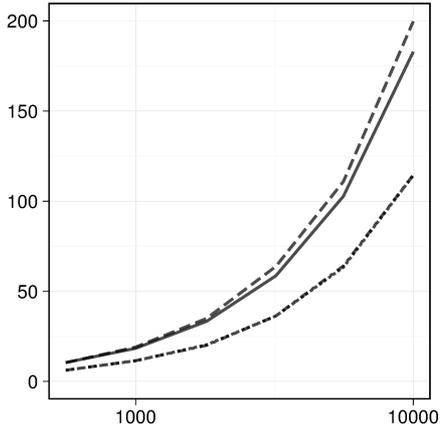


Figure 11: Computing times as a function of N for $\rho = 0.01$ (solid line), $\rho = 0.04$ (dashed line), $\rho = 0.07$ (dot-dashed line), and $\rho = 0.10$ (dotted line). The abscissa axis is scaled logarithmically.

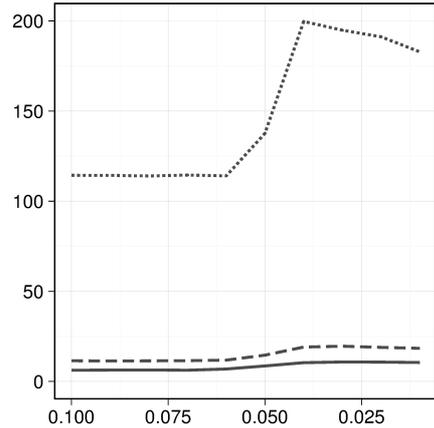


Figure 12: Computing times as a function of the rarity parameter ρ for $N = 560$ (solid line), $N = 5600$ (dashed line), and $N = 56000$ (dotted line). The abscissa axis is reverse.

RMSE drops when ρ exceeds ≈ 0.05 , and there is a corresponding increase in the computing times around the same threshold. This evidence suggests that the cross-entropy algorithm tends to stop too early when $\rho < 0.05$, and this results in shorter computing times and higher RMSE of $\hat{\sigma}$.

2-level logistic model with non-diagonal Ω . In this section we assess the effect of N and ρ on the RMSE and the computing times for non-independent and heteroskedastic random effects ($\Omega \neq \sigma^2 I_n$). To this aim, we have simulated 100 models for each pair (N, ρ) , where $N \in \{560, 1000, 1800, 3200, 5600, 10000\}$ and $\rho \in \{0.01, 0.02, \dots, 0.10\}$. The simulation scheme complies with the previous one, with the exception of the values of β and Ω , which are different. In particular, we set $\beta = \iota_4$ and Ω is non-diagonal and heteroskedastic. The structure of Ω is such that the random effects are still uncorrelated amongst groups:

$$\Omega = \Xi \otimes I_G \in \mathbb{R}^{Gk_2 \times Gk_2},$$

where $\Xi \in \mathbb{R}^{k_2 \times k_2}$ is the covariance matrix of the random effects amongst regressors.

The standard deviations of the random effects are $\sigma = [0.1, 0.2, 0.5, 1]^T$, whereas the

correlations $\psi_{s,t}$ ($s, t = 1, \dots, k_2$) are all equal to 0.2. It follows that

$$\Xi = \Lambda \mathcal{P} \Lambda,$$

where

$$\Lambda = \text{diag}(\sigma_1, \dots, \sigma_{k_2}), \quad \mathcal{P} = (1 - \psi)I_{k_2} + \psi \iota_{k_2} \iota_{k_2}^\top.$$

Although the random effects on the regressors are simulated according to an equicorrelated matrix, they are estimated independently (that is, they are not restricted to $\psi_{s,t} = \psi_{h,k}$, for any $(s, t) \neq (h, k)$). It follows that the parameter space Θ is 14-dimensional ($14 = 2k_2 + k_2(k_2 - 1)/2$, with $k_2 = 4$), and, in particular:

$$\theta = [\beta^\top, \sigma^\top, \psi_{2,1}, \psi_{3,1}, \psi_{3,2}, \dots, \psi_{k_2, k_2-1}]^\top.$$

In order to reduce the computing times of the experiment, we adopt the logit-normal on $(-0.25, 0.25)$ as instrumental distribution for the correlation coefficients in \mathcal{P} , instead of using the `nearPD` function or the reparametrisation based on the Cholesky decomposition illustrated in Section 3.2.3. The bounds of the interval $(-0.25, 0.25)$ guarantee that \mathcal{P} is positive definite, according to the Geršgorin theorem (Geršgorin, 1931).

The complete results of the simulations are reported in Tables SM.13-SM.29 of the supplementary material. All in all, the simulation results convey a threefold message.

First, the estimates of all correlation coefficients $\psi_{s,t}$ ($s, t = 1, \dots, k_2$, $s \neq t$) are equal to zero (more exactly, $|\psi_{s,t}| < 10^{-8}$) for any value of N and ρ , when the true value is 0.2. This is caused by a likelihood function which is almost flat with respect to these variables, and by the use, as a starting instrumental distribution, of a logit normal which is symmetric around zero.

Second, the RMSEs of $\hat{\beta}_j$ ($j = 1, \dots, k_1$) are basically independent of both N and ρ , and this behaviour is rather different from the previous simulations (see e.g. Figures 1-2, and 7-8). The reason is that the RMSE is mainly determined by the variance of the estimators, which is little affected by N and ρ , whereas the bias is very small in absolute value (see Tables SM.21-SM.24 in the supplementary material). Anyway, it is worth noting that the bias is still affected by N and ρ , and tends to become negative as ρ increases, whilst it is positive as ρ gets closer to 0.01. The change of sign in the interval $\rho \in [0.04, 0.02]$ implies

that, when looking at the squared bias, the U-shaped curve of Figures 2 and 8 appears again, although less evident. Accordingly, these outcomes are in line with the previous simulations, and the choices $N = 1000$ and $\rho \approx 0.04$ can be considered reliable.

Third, the bias of the coefficients $\hat{\beta}_j$ ($j = 1, \dots, k_1$) is little affected by the magnitude of the variance of the random effect (σ_j^2 , $j = 1, \dots, k_2$), whereas the RMSE tends to grow as this variance gets larger. On the other hand, the value of σ_j has a non-negligible impact on the bias of the estimators $\hat{\sigma}_j$ ($j = 1, \dots, k_2$), which becomes more and more negative as σ_j grows. The RMSE increases correspondingly.

3.4.2 Smoothness parameter α

According to (13), α regulates the smoothness of the location and correlation parameters in v_t . Some pilot simulations, whose outcomes are not shown here, suggest that the estimation of logistic multilevel models is rather insensitive to this parameter, which can therefore be set to 1 (no smoothing). In general, as α gets closer to 1 the components of v_t are less smooth and the algorithm is free to explore the whole parameter space Θ . On the other hand, as α lowers, the smoothness is higher and the convergence should be faster.

3.4.3 Smoothness parameters β and q

The parameters β and q regulate the smoothness of the sequence of the scale parameters in v_t . As β decreases and/or q increases, the sequence gets smoother, and this typically results in a slower reduction of the region of Θ explored by the algorithm. On the contrary, high values of β and small q increase the speed of convergence of the algorithm.

Both parameters affect the convergence of the algorithm and, therefore, the statistical properties of the estimators. This is the reason why we have run some ad hoc simulations aimed at analysing the effects of β and q on the estimates as well as the computing times. The experiment is structured in the same way as in Section 3.4.1 for the logistic model. The only difference is that here N and ρ are fixed ($N = 1000$ and $\rho = 0.04$), and the number of replications is 250 (instead of 50). According to Rubinstein and Kroese (2004), we have chosen $\beta \in \{0.90, 0.91, 0.92, \dots, 0.99\}$ and $q \in \{5, 6, \dots, 10\}$.

The RMSE of the parameters and the computing times are reported in Tables SM.30-

SM.35 of the supplementary material. The computing times grow as q increases and/or β decreases. On the other hand, the RMSEs of the parameters decrease as q increases and β gets smaller. It follows that, if the researcher is primarily interested in minimising the RMSE of the estimators, she should set $(q, \beta) = (10, 0.9)$. Nevertheless, it should be stressed that the parameter q has a larger effect than β both on the RMSE and the computing times. Thus, it is possible to reduce the computing times (by a percentage close to 10%) at a price of a slight increase of the RMSE (on average about 1%) by setting $q = 10$ and β close or equal to 0.99. Accordingly, in the simulations of Section 3.4.1 we use $(q, \beta) = (10, 0.975)$.

3.4.4 Parameters of the stopping criterion (`pvalue`, n_b , v_b)

The stopping criterion depends on a parameter that directly triggers the stopping rule (`pvalue`), and two parameters that select the sample (16) used to compute the Geweke’s test (n_b , v_b).

The higher the threshold `pvalue`, the stricter the stopping criterion and the longer the computing times. The simulations illustrated in this paper use `pvalue` = 0.1, in order to study the performance of the cross-entropy algorithm under stringent conditions. Nevertheless, we have verified that if we had stopped the algorithm when `pvalue` = 0.05, the statistical properties of the estimators would not have worsen significantly, and in some cases they would have been even better. On the other hand, the computing times would have been dramatically shorter, as the number of iterations would have been substantially reduced (up to 90% for the highest values of ρ and N).

As n_b increases, the outcome of the Geweke’s test becomes more reliable, but n_b determines the minimum number of iterations of the algorithm, and this may tend to increase the computing times. In the simulations we set $n_b = 20$, but the algorithm always requires at least 30 iterations to converge. It follows that n_b is not binding.

The parameter v_b determines the speed of growth of the sample (16). Values of v_b close to 1 tend to increase the number of iterations, nevertheless the Geweke’s test can be computed on larger samples, and this improves its reliability. On the other hand, if $v_b = 0$ the sample (16) has a fixed number of observations (n_b) but includes only “recent”

observations of the process $\{\hat{\gamma}_t\}$. In the Monte Carlo simulations we set $v_b = 0.5$.

4 A Comparison to Adaptive Gaussian Quadrature

In this section the cross-entropy approach is compared via simulation to the estimation technique based on the Adaptive Gaussian Quadrature (AGQ) algorithm developed by Pinheiro and Chao (2006), and implemented in the R package `lme4` (Bates et al., 2015).

The AGQ method permits to decompose a multivariate Gaussian integration problem in a sequence of unidimensional Gaussian quadratures, whose accuracy depends on the number of points used for approximating the unidimensional integrals.⁴ In particular, if m points are employed for each dimension, Gm^{k_2} evaluations of the log-likelihood function are necessary, as the method assumes independence of random effects amongst groups. According to Pinheiro and Chao (2006), $m = 1$ often provides a good approximation, whilst $m \leq 7$ is usually enough in order to obtain accurate estimates.

It is worth noting that the assumption of independence of the random effects amongst groups implies that the likelihood function (6) can be restated as

$$\mathcal{L}(\beta, \Omega) = \mathbb{E}(f_{Y|U}(y|U; \beta)) = \prod_{j=1}^G \int_{\mathbb{R}^{k_2}} f_{Y|U_j}(y|u_j; \beta) f_{U_j}(u_j; \Xi) du_j = \prod_{j=1}^G \mathbb{E}(f_{Y_j|U_j}(y_j|U_j; \beta)),$$

where $U_j \in \mathbb{R}^{k_2}$ is the random vector of random effects in group j , and f_{U_j} is its probability density function, so that the evaluation of the likelihood requires G k_2 -dimensional integrations.

The assumption of independence is often considered a reasonable assumption, and allows one to benefit from the reduced computational complexity resulting from the dimensional reduction of the integration space. However, this assumption is irrelevant for the cross-entropy algorithm, as it always integrates in a Gk_2 -dimensional space. This remark should be taken into account when comparing the performances of the cross-entropy method and of other estimation techniques.

The experiments are based on models of type (1) with $n = \sum_{j=1}^G n_j = 250$ observations randomly clustered into G groups. We fit 20 different models resulting from the

⁴Note that the evaluation of the likelihood function $\mathbb{E}(\mathcal{L}_c(\beta, U))$ requires to integrate \mathcal{L}_c with respect to $U \in \mathbb{R}^{G \times k_2}$. This means that it is necessary to perform a Gk_2 -dimensional integration.

RMSE of $\hat{\beta}_0$	σ				
	0.01	0.10	0.20	0.50	1.00
AG quadrature ($m = 1$)	269.64	137.11	45.04	34.29	187.36
AG quadrature ($m = 3$)	34.79	134.51	38.49	33.48	126.86
AG quadrature ($m = 5$)	35.84	113.69	39.38	33.79	258.89
Cross-entropy ($N = 1000$)	29.08	28.08	32.47	31.94	30.68
Cross-entropy ($N = 2000$)	29.42	28.18	32.36	31.82	30.76
Cross-entropy ($N = 5000$)	29.07	28.10	32.41	31.98	30.76
RMSE of $\hat{\beta}_1$					
AG quadrature ($m = 1$)	450.98	212.74	71.31	45.50	483.74
AG quadrature ($m = 3$)	50.70	225.85	53.12	44.40	294.71
AG quadrature ($m = 5$)	52.70	219.29	54.83	44.92	423.53
Cross-entropy ($N = 1000$)	41.91	39.34	43.67	41.86	51.45
Cross-entropy ($N = 2000$)	41.93	39.22	42.95	41.33	51.21
Cross-entropy ($N = 5000$)	41.88	39.58	43.29	41.63	51.61
RMSE of $\hat{\beta}_2$					
AG quadrature ($m = 1$)	523.82	363.11	87.71	68.97	658.52
AG quadrature ($m = 3$)	72.79	403.71	69.50	66.92	392.21
AG quadrature ($m = 5$)	75.80	392.26	71.83	67.84	645.28
Cross-entropy ($N = 1000$)	55.51	53.77	56.01	61.04	62.51
Cross-entropy ($N = 2000$)	55.61	53.81	55.72	60.59	61.95
Cross-entropy ($N = 5000$)	55.33	53.60	55.67	60.93	62.45
RMSE of $\hat{\beta}_3$					
AG quadrature ($m = 1$)	951.81	486.40	128.73	90.05	965.47
AG quadrature ($m = 3$)	103.17	524.54	98.23	87.73	516.14
AG quadrature ($m = 5$)	107.86	483.75	101.66	88.86	926.51
Cross-entropy ($N = 1000$)	76.99	73.74	78.09	81.96	81.37
Cross-entropy ($N = 2000$)	77.86	73.49	77.29	80.56	81.11
Cross-entropy ($N = 5000$)	77.61	73.81	78.02	81.16	81.62
RMSE of $\hat{\sigma}$					
AG quadrature ($m = 1$)	561.45	393.75	73.21	47.91	801.28
AG quadrature ($m = 3$)	58.16	305.73	51.27	45.98	250.10
AG quadrature ($m = 5$)	61.32	237.42	54.14	46.99	286.78
Cross-entropy ($N = 1000$)	40.51	31.54	22.06	10.81	59.19
Cross-entropy ($N = 2000$)	41.24	31.77	22.74	10.32	58.69
Cross-entropy ($N = 5000$)	41.24	31.83	22.63	10.13	58.45

Table 1: Sample Root Mean Squared Error (RMSE) of all the estimators of model (1) obtained by means of Monte Carlo simulations for various estimation techniques, various values of σ and $G = 50$. For each value of σ , the number of replications is equal to 200. The values of the RMSE are multiplied by 100.

combination of various values of σ and G . In particular, $\sigma \in \{0.01, 0.1, 0.2, 0.5, 1\}$ and $G \in \{8, 12, 25, 50\}$. The number of regressors is $k_1 = 4$ (including a constant), thus $X \in \mathbb{R}^{n \times 4}$, the vector of coefficients is $\beta = [1, 2, 3, 4]^T$, and there is a random effect on the constant term coefficient β_0 ($k_2 = 1$). Both X and the grouping of the observations are the same in all models with the same value of G . U is distributed as $U \sim \mathcal{N}_G(0, \sigma^2 I)$, whilst the elements of X are iid standard Gaussian. The number of replications is equal to 200. The AGQ estimation uses $m \in \{1, 3, 5\}$ nodes per dimension; the cross-entropy algorithm is based on $N \in \{1000, 2000, 5000\}$, $\rho = 0.04$, $\alpha = 1$, $\beta = 0.975$, $q = 10$, `pvalue` = 0.05, $n_b = 20$ and $v_b = 0.5$.

Table 1 shows the RMSE for $G = 50$, whilst the complete results are reported in Tables SM.36-SM.43 of the supplementary material. According to the outcomes of Section 3.4.1, the cross-entropy estimators do not improve, in terms of both RMSE and bias, as N exceeds 1000. The computing times increase proportionally to N .

The behaviour of the RMSE and the bias of the AGQ estimates is more involved. When $G \leq 25$, there is no perceivable improvement as m increases. On the other hand, for $G = 50$, both the RMSE and the bias tend to decrease markedly as m goes from 1 to 3, whilst the effect of an increase from 3 to 5 is negligible. This pattern holds for all models except those with $G = 50$ and $\sigma = 1$. The very large figures in the last column of Table 1 are due to a few replications where AGQ gives anomalously large estimates, as can be seen from the boxplots reported in the supplementary material. This is particularly evident when $m = 5$. Similarly, the large value of bias and RMSE of the AGQ estimates for $G = 50$ and $\sigma = 0.1$ is due to a very large estimate of one of the models. With respect to AGQ, the cross-entropy estimates have smaller RMSE and bias for any value of σ and G , and the difference tends to get larger as G increases, as confirmed by the results for $G \in \{8, 12\}$.

The outcomes displayed in Tables 1 and in the supplementary material reveal that both methods basically fail to estimate σ , as the estimate is on average quite stable with respect to σ . In particular, a large percentage (about 40%) of the estimates based on AGQ are equal to 0, i.e. to the non-negative variance constraint set in the optimisation routine.

The computing times of AGQ are about a hundredth of those of the cross-entropy

method with $N = 1000$; nonetheless the cross-entropy method is still appealing, as its computing times are short (less than thirty seconds).

We have repeated this simulation experiment also for a larger sample size ($n = 1000$). The results are similar to those obtained for $n = 250$, but the difference between the AGQ and the cross-entropy estimates in terms of RMSE gets smaller also for large G . Complete results are available in the supplementary material, see Tables SM.44-SM.51.

In summary, the simulations suggest that the cross-entropy estimates are as good as those obtained through AGQ when the number of groups is small (and thus the average number of units per group is large), but markedly better (in terms of both RMSE and bias) when the number of groups gets large. Moreover, the statistical properties of the AGQ estimates worsen as G grows, because of an increased instability. In particular, as G grows, the AGQ method sometimes produces anomalous estimates of the parameters, although this happens less frequently as the sample size gets larger.

5 Application

In this section we fit a 2-level logistic model to a sample of 1909 startup companies founded between 2004 and 2007 in the healthcare industry in Italy. The dependent variable is the status of the firms one year after the foundation: 0 if the company is still operating, 1 if it is out of business. Thus, we model the one-year firm death probability.

The highest level of local governments in Italy is represented by 20 regions (*Regioni*). We take them as the 2nd-level units. The regressors are the average number of employees of each firm (**avemp**), the real regional GDP growth rate (**gdp**) and the percentage of graduates in the region where the firms are located (**degree**). All the data are provided by the Italian National Institute of Statistics (ISTAT).

The variable **avemp** is a proxy for the size of the company. If the death probability of a firm is related to the firm size, this should result in a significant coefficient for **avemp**. **gdp** is related to the level of economic activity of the context where the firms operate. This variable is expected to have a negative and significant coefficient, since, as the economy of the region grows, the firms in that region should benefit from it, and this is expected to reduce their default risk. The variable **degree** has been selected as it is positively

correlated with the birth rate of startups, but not directly related to the probability of failure. It follows that it may be an indirect measure of the dynamism of the social and economic environment where the firms operate. Hence, it is expected to have a positive and significant sign.

We estimate both the logit and the 2-level logit models. The latter is specified with a random effect for each variable, and is estimated both through the AGQ (with $m = 1$) and the cross-entropy method for $N = 1000$ and $N = 10000$.

The other parameters of the cross-entropy algorithm are set according to the results of the simulations ($\rho = 0.04$, $\alpha = 1$, $\beta = 0.975$, $q = 10$, `pvalue` = 0.05, $n_b = 20$ and $v_{n_b} = 0.5$). The evaluation of the likelihood function of the 2-level logit model requires a 80-dimensional integration, as there are 20 regions and 4 random effects. The computing time for $N = 1000$ is 32.18 s, whilst the estimation procedure takes 286.85 s for $N = 10000$.

The AGQ method is only applied with $m = 1$, as the `lme4` package is not implemented for $m > 1$ when there is more than one random effect. The estimation has taken 26.60 s but it has not converged, thus the estimates of Table 2 may not be reliable.

Table 2 shows that none of the regressors (except the constant) is significant when the logit model is fitted. On the other hand, if the clustering effects are considered, `degree` and `gdp` become significant (at 0.01 for cross-entropy estimates with $N = 1000$, at 0.05 when $N = 10000$) and have the expected sign. It is worthwhile noting that the estimates of the significant coefficients are basically the same for the logit and the multilevel model. As for the random effects, the coefficient of `avemp` is the only significant one and can be interpreted as the result of the heterogeneity amongst Italian regions.

The estimates of the multilevel logit based on AGQ are rather close to the cross-entropy estimates, but the standard errors tends to be larger, and this results in non-significant regressors (except for the constant term coefficient) and non-significant random effects.

6 Conclusion

In this paper we have developed a cross-entropy algorithm for MLE of GLMM, focusing on the logistic and Poisson specification. With respect to the original cross-entropy algorithm, we have introduced two novelties that improve the computational efficiency and the

	Logit	2-level (AGQ, $m = 1$)		2-level (CE, $N = 1000$)		2-level (CE, $N = 10000$)	
	coeff.	coeff.	rand. eff.	coeff.	rand. eff.	coeff.	rand. eff.
Intercept	-2.03192 [0.37674] ***	-2.12427 [0.71536] **	0.33777 [0.58118]	-2.00489 [0.01433] ***	0.01569 [0.03018]	-2.02621 [0.01847] ***	0.02547 [0.03042]
avemp	0.01131 [0.03827]	-0.11545 [0.11520]	0.07973 [0.28236]	0.00449 [0.03242]	0.07426 [0.01750] ***	0.00528 [0.03181]	0.07455 [0.01639] ***
degree	0.03860 [0.03012]	0.05581 [0.05337]	0.00193 [0.04389]	0.03650 [0.01394] **	0.00153 [0.03578]	0.03812 [0.01813] *	0.00207 [0.02684]
gdp	-0.06226 [0.06442]	-0.07544 [0.09185]	0.01544 [0.12425]	-0.06400 [0.02235] **	0.00776 [0.04724]	-0.06290 [0.02490] *	0.01204 [0.02941]

Table 2: Results of the logit and the 2-level logit regressions. The 2-level logit regression is estimated by means of AGQ with $m = 1$ and the cross-entropy method with $N = 1000$ and $N = 10000$. The table reports the point estimates of the parameters of the models, their standard errors (in brackets), and the level of significance (“.” for p -values between 0.1 and 0.05, “*” for p -values between 0.05 and 0.01, “**” for p -values between 0.01 and 0.001, “***” for p -values smaller than 0.001, and nothing when the p -values are higher than 0.1).

applicability of the cross-entropy algorithm to MLE. First, we employ the logit-normal as sampling distribution for bounded variables, and this allows us to update the parameters in closed form. Second, we use a formal test of stationarity as a stopping criterion. This simplifies the application of the method to data generated by processes based on different values of the parameters and various probability distributions of the random effects. Moreover, since the algorithm is stopped according to the p -value of the Geweke’s test of stationarity, the stopping criterion is problem-independent.

The outcomes of the simulations provide evidence of the validity of this estimation approach as an effective alternative to numerical integration methods such as the adaptive Gaussian quadrature. The cross-entropy estimates are less biased and have lower RMSE than those based on AGQ, especially when the dimension of the integration space is large. This property makes the cross-entropy method well-suited for fitting complicated multilevel models where the hierarchy of data is particularly complex (in terms of number of levels

and/or groups) or the number of random effects is large. Although the computing times of the cross-entropy approach are longer than those of AGQ, they are still rather short, so that this is not a relevant issue for practical purposes.

Furthermore, since the cross-entropy method does not rely on the assumption of independence of random effects amongst groups, it may be a well performing estimation techniques for models where the independence assumption is relaxed, and large-dimensional integration is needed.

SUPPLEMENTARY MATERIAL

Complete results of Monte Carlo simulations. Complete results of Monte Carlo simulations and boxplots of the estimates for $G = 50$ (pdf file).

References

- Aitkin, M., D. Anderson, and J. Hinde (1981). Statistical modelling of data on teaching styles (with discussion). *Journal of the Royal Statistical Society. Series A 144*, 148–161.
- Bates, D. and M. Mächler (2015). *Matrix: Sparse and Dense Matrix Classes and Methods*. R package version 1.2-2.
- Bates, D., M. Mächler, B. Bolker, and S. Walker (2015). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software 67*(1), 1–48.
- Bennett, N. (1976). *Teaching Styles and Pupil Progress*. Open Books.
- Frederic, P. and F. Lad (2008). Two moments of the logitnormal distribution. *Communications in Statistics – Simulation and Computation 37*, 1263–1269.
- Geršgorin, S. (1931). Über die abgrenzung der eigenwerte einer matrix. *Izvestiya Akademii Nauk SSSR Otdelenie Matematicheskikh i Estestvennykh Nauk 7*, 749–754.
- Geweke, J. (1992). Evaluating the accuracy of sampling-based approaches to calculating posterior moments. In J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith (Eds.), *Bayesian Statistics 4: Proceedings of the Fourth Valencia International Meeting*.

- Goldstein, H. (2011). *Multilevel Statistical Models* (4 ed.). Wiley.
- Higham, N. J. (2002). Computing the nearest correlation matrix – a problem from finance. *IMA Journal of Numerical Analysis* 22, 329–343.
- Morris, C. N. (1982). Natural exponential families with quadratic variance functions. *The Annals of Statistics*, 10, 65–80.
- Pinheiro, J. and E. C. Chao (2006). Efficient Laplacian and adaptive Gaussian quadrature algorithms for multilevel generalized linear mixed models. *Journal of Computational and Graphical Statistics* 15, 58–81.
- R Core Team (2015). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing.
- Rubinstein, R. Y. (1997). Optimization of computer simulation models with rare events. *European Journal of Operational Research* 99, 89–112.
- Rubinstein, R. Y. (1999). The cross-entropy method for combinatorial and continuous optimization. *Methodology and Computing in Applied Probability* 2, 127–190.
- Rubinstein, R. Y. and D. P. Kroese (2004). *The Cross-Entropy Method*. Springer.
- Zhang, Y. (2013). Likelihood-based and Bayesian methods for Tweedie compound Poisson linear mixed models. *Statistics and Computing* 23(6), 743–757.

A Cross-Entropy Approach to the Estimation of Generalised Linear Multilevel Models

Marco Bee*, Giuseppe Espa[†],
Diego Giuliani[‡], Flavio Santi[§]

Supplementary Material

The present document includes the following tables and figures:

- **Tables SM.1-SM.6** – RMSE and computing times about simulations on the 2-level logistic model (1) with $\Omega = \sigma^2 I$. – Parameters N and ρ (Section 3.4.1 of the paper).
- **Tables SM.7-SM.12** – RMSE and computing times about simulations on the 2-level Poisson model (3) with $\Omega = \sigma^2 I$. – Parameters N and ρ (Section 3.4.1 of the paper).
- **Tables SM.13-SM.29** – RMSE, bias and computing times about simulations on the 2-level logistic model (1) with $\Omega = \Xi \otimes I_G$. Only RMSE for $\hat{\beta}_j$ ($j = 1, \dots, k_1$) and $\hat{\sigma}_j$ ($j = 1, \dots, k_2$) are reported. – Parameters N and ρ (Section 3.4.1 of the paper).
- **Tables SM.30-SM.35** – RMSE and computing times about simulations on the 2-level logistic model (1) with $\Omega = \sigma^2 I$. – Parameters β and q (Section 3.4.3 of the paper).
- **Tables SM.36-SM.51** – RMSE, bias and computing times about the simulations for the comparison between the cross-entropy and the AGQ estimates – For $G = 8, 12, 25, 50$ and $n = 250, 1000$ (Section 4 of the paper).
- **Figures SM.1-SM.15** – Boxplots of the point estimates about the simulations for the comparison between the cross-entropy and the AGQ estimates – For $G = 50$ and $n = 250$ (Section 4 of the paper).

*Department of Economics and Management, University of Trento, Via Inama 5, 38122 Trento. E-mail: marco.bee@unitn.it.

[†]Department of Economics and Management, University of Trento, Via Inama 5, 38122 Trento. E-mail: giuseppe.espa@unitn.it.

[‡]Department of Economics and Management, University of Trento, Via Inama 5, 38122 Trento. E-mail: diego.giuliani@unitn.it.

[§]Department of Economics and Management, University of Trento, Via Inama 5, 38122 Trento. E-mail: flavio.santi@unitn.it.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
100	27.56	32.48	25.54	28.68	31.28	26.67	31.64	39.55	42.27	41.50
180	24.30	23.56	25.16	24.20	25.72	28.15	28.35	28.54	30.51	33.30
320	25.83	23.52	23.57	24.12	23.50	23.57	24.24	27.60	28.62	30.15
560	23.98	24.01	23.94	24.77	23.49	24.27	24.73	25.79	27.19	29.07
1000	24.61	23.87	24.06	23.20	24.00	24.63	25.26	25.65	27.29	29.29
1800	23.30	23.41	23.62	24.20	24.03	24.01	24.62	25.62	26.43	26.60
3200	24.46	23.05	23.44	23.67	23.50	24.03	24.30	25.67	25.75	27.07
5600	23.62	23.56	23.76	23.35	23.65	23.83	24.60	25.07	26.31	27.75
10000	23.92	23.50	23.50	23.55	23.84	24.05	24.32	25.00	25.88	27.00
18000	23.92	23.36	23.47	23.49	23.64	23.82	24.28	24.97	25.64	26.90
32000	23.85	23.51	23.50	23.48	23.53	23.86	24.24	25.06	25.70	26.72
56000	23.74	23.57	23.39	23.46	23.61	23.88	24.25	24.94	25.69	26.79

Table SM.1: Sample Root Mean Squared Error (RMSE) for $\hat{\beta}_1$ of model (1) obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ), the number of replications is equal to 50. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
100	58.51	62.49	55.70	58.54	63.89	75.20	63.82	108.58	107.14	99.54
180	46.26	43.97	42.54	52.76	51.00	57.64	66.51	67.30	83.30	90.65
320	46.51	44.58	46.83	45.75	47.24	52.86	56.41	57.15	68.21	70.25
560	45.11	44.90	44.16	48.07	50.54	50.53	55.61	57.09	63.42	72.45
1000	44.69	43.84	44.21	45.10	47.43	50.86	52.70	57.27	66.70	72.42
1800	43.97	44.97	45.15	46.34	46.59	48.38	51.59	56.17	62.16	64.08
3200	45.92	44.36	44.87	44.92	46.55	48.24	51.13	54.77	60.25	63.08
5600	45.64	44.99	44.80	45.18	46.41	47.99	50.56	53.46	58.25	63.54
10000	44.71	45.20	44.89	45.25	46.31	48.17	50.58	53.57	58.35	62.03
18000	45.17	44.69	44.61	45.13	46.03	48.05	50.33	53.63	57.17	62.57
32000	45.47	44.78	44.64	44.95	45.94	47.85	49.83	53.04	57.81	61.88
56000	45.38	44.88	44.81	45.18	45.95	47.71	49.70	52.71	57.08	61.76

Table SM.2: Sample Root Mean Squared Error (RMSE) for $\hat{\beta}_2$ of model (1) obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ), the number of replications is equal to 50. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
100	80.57	76.55	76.87	86.71	95.30	98.70	86.66	141.60	140.92	132.80
180	63.44	63.32	58.26	67.61	66.87	73.45	89.89	87.30	108.71	126.67
320	63.81	58.49	61.62	62.62	64.72	68.50	77.05	78.00	87.95	93.92
560	58.27	59.00	60.44	61.77	66.83	66.31	72.94	75.02	84.31	96.93
1000	60.08	59.34	60.36	60.39	63.28	66.17	69.34	75.60	86.05	95.90
1800	59.55	59.36	60.35	61.69	62.59	64.71	68.63	73.41	82.87	86.54
3200	60.55	60.35	60.04	60.58	62.15	63.85	67.89	72.54	80.09	85.55
5600	60.77	60.48	60.46	60.44	61.60	64.00	66.54	70.29	77.20	83.95
10000	60.78	60.66	59.98	60.19	62.13	64.17	67.40	71.50	76.65	82.86
18000	61.09	60.22	60.11	60.63	61.52	64.18	66.82	71.44	75.72	83.31
32000	61.18	60.53	60.27	60.33	61.55	63.91	66.19	70.35	76.34	82.25
56000	61.46	60.38	60.24	60.50	61.48	63.72	66.25	70.09	75.39	82.07

Table SM.3: Sample Root Mean Squared Error (RMSE) for $\hat{\beta}_3$ of model (1) obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ), the number of replications is equal to 50. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
100	90.49	121.48	88.54	103.11	117.56	117.07	117.33	183.02	191.32	174.68
180	71.31	79.03	70.44	84.57	86.78	103.49	107.34	112.85	134.61	157.21
320	79.29	71.14	76.65	76.76	77.77	82.97	92.09	95.93	115.09	123.38
560	72.51	72.45	73.64	79.66	78.54	85.67	89.83	95.37	102.99	122.73
1000	75.86	72.12	73.10	73.31	79.26	84.73	88.04	95.94	108.78	119.13
1800	72.57	73.39	72.62	75.79	76.52	80.53	83.01	92.64	102.37	105.58
3200	75.40	72.41	73.29	74.15	76.38	78.30	83.63	90.87	97.92	105.70
5600	74.14	74.52	73.69	73.69	75.43	78.38	82.68	87.80	97.80	106.70
10 000	74.51	74.02	73.22	73.89	76.23	78.70	83.10	87.83	95.82	102.48
18 000	76.00	73.41	73.39	73.87	75.56	78.27	82.20	88.05	93.48	103.25
32 000	75.52	73.64	73.52	73.90	75.29	78.20	81.56	87.18	94.76	102.55
56 000	75.36	73.78	73.43	73.91	75.26	77.99	81.51	86.49	93.90	102.20

Table SM.4: Sample Root Mean Squared Error (RMSE) for $\hat{\beta}_4$ of model (1) obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N , ρ), the number of replications is equal to 50. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
100	11.59	14.73	11.74	8.50	11.34	8.80	9.94	10.06	10.17	8.12
180	12.83	14.92	11.76	11.73	9.96	9.44	9.65	8.15	8.90	8.92
320	13.28	13.13	12.8	12.32	10.16	11.22	9.09	8.75	8.56	8.10
560	16.70	14.15	13.33	11.88	10.92	10.01	9.49	8.86	8.19	9.01
1000	17.12	14.51	13.34	11.45	10.77	10.77	9.46	9.25	9.03	8.65
1800	17.37	15.16	13.22	12.26	10.74	10.27	9.69	8.92	8.65	8.18
3200	17.80	14.92	13.32	12.01	11.15	10.34	9.47	9.26	8.78	8.45
5600	17.42	14.91	13.48	12.16	11.06	10.24	9.73	9.12	8.46	8.38
10 000	17.75	14.91	13.38	12.13	11.14	10.18	9.61	8.95	8.53	8.27
18 000	17.55	14.91	13.37	12.13	10.96	10.27	9.51	9.02	8.56	8.24
32 000	17.54	15.11	13.34	12.06	11.08	10.25	9.59	9.05	8.59	8.18
56 000	17.42	14.97	13.39	12.07	11.05	10.24	9.55	9.01	8.56	8.19

Table SM.5: Sample Root Mean Squared Error (RMSE) for $\hat{\sigma}$ of model (1) obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N , ρ), the number of replications is equal to 50. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
100	0.05	0.04	0.04	0.05	0.05	0.06	0.06	0.07	0.07	0.09
180	0.08	0.08	0.07	0.09	0.08	0.09	0.09	0.10	0.13	0.17
320	0.15	0.13	0.12	0.15	0.15	0.15	0.17	0.18	0.20	0.28
560	0.28	0.25	0.23	0.27	0.28	0.25	0.28	0.33	0.49	0.54
1000	0.49	0.51	0.39	0.49	0.45	0.51	0.51	0.80	0.91	1.52
1800	0.93	0.89	0.80	0.97	1.02	0.92	1.31	1.70	2.29	2.71
3200	1.78	1.56	1.57	1.77	1.95	1.84	2.37	3.85	4.57	8.11
5600	3.10	2.98	3.17	3.36	3.73	3.29	4.57	6.33	8.84	15.20
10 000	7.37	5.80	6.45	7.22	7.41	7.18	11.56	15.01	23.15	27.02
18 000	13.90	11.57	12.06	12.60	13.78	14.58	25.71	32.73	45.27	57.71
32 000	26.27	23.69	24.93	25.54	27.72	28.12	52.18	68.06	84.42	100.77
56 000	56.43	46.61	47.47	61.65	60.47	68.46	102.44	136.53	151.21	179.59

Table SM.6: Computing times of model (1) obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N , ρ), the number of replications is equal to 50. The table reports the average computing times for one model, expressed in minutes.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	9.16	9.46	10.14	9.47	9.98	10.16	9.98	10.35	11.55	13.37
1000	9.72	9.45	9.55	9.43	9.82	9.98	10.78	10.75	11.37	13.13
1800	9.49	9.32	9.49	9.58	9.60	9.90	10.19	10.47	11.07	12.49
3200	9.38	9.38	9.45	9.48	9.61	9.83	10.26	10.39	10.99	12.62
5600	9.52	9.40	9.43	9.46	9.56	9.79	10.08	10.56	11.12	12.53
10000	9.54	9.43	9.42	9.40	9.53	9.76	9.98	10.37	11.25	12.54

Table SM.7: Sample Root Mean Squared Error (RMSE) for $\hat{\beta}_1$ of model (3) obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ), the number of replications is equal to 50. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	7.52	7.48	7.04	7.18	7.16	6.81	7.18	7.14	7.04	8.16
1000	7.31	7.03	6.99	7.13	7.07	7.22	7.24	7.31	7.51	7.47
1800	6.90	7.07	7.02	7.05	7.04	7.29	7.04	7.26	7.24	7.44
3200	7.15	7.16	7.04	7.12	7.10	7.31	7.22	7.21	7.14	7.11
5600	6.94	6.95	6.90	7.03	7.09	7.08	7.19	7.12	7.17	7.27
10000	7.07	7.06	6.95	7.03	7.02	7.11	7.16	7.05	7.20	7.30

Table SM.8: Sample Root Mean Squared Error (RMSE) for $\hat{\beta}_2$ of model (3) obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ), the number of replications is equal to 50. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	9.23	9.51	8.86	9.13	9.29	9.43	9.86	9.63	9.76	9.84
1000	9.04	9.16	9.06	9.44	9.23	9.33	9.50	9.40	9.69	9.34
1800	9.02	8.83	9.20	9.04	9.23	9.59	9.44	9.46	9.67	9.71
3200	9.13	9.06	8.93	9.27	9.23	9.33	9.22	9.32	9.34	9.72
5600	9.19	8.96	9.08	9.15	9.14	9.33	9.43	9.48	9.37	9.62
10000	8.89	8.90	9.06	9.11	9.13	9.31	9.34	9.47	9.57	9.82

Table SM.9: Sample Root Mean Squared Error (RMSE) for $\hat{\beta}_3$ of model (3) obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ), the number of replications is equal to 50. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	9.22	8.82	8.75	8.94	9.15	9.05	9.18	9.06	10.07	9.97
1000	8.77	9.05	8.71	8.78	9.10	9.08	9.43	9.40	9.50	10.32
1800	8.74	8.66	8.81	8.74	8.97	9.11	9.08	9.28	9.53	9.72
3200	8.81	8.71	8.58	8.68	8.89	8.95	9.02	9.24	9.34	9.82
5600	8.61	8.69	8.79	8.81	8.96	9.07	8.99	9.33	9.53	9.71
10000	8.80	8.79	8.69	8.75	8.91	9.06	9.01	9.15	9.52	9.84

Table SM.10: Sample Root Mean Squared Error (RMSE) for $\hat{\beta}_4$ of model (3) obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ), the number of replications is equal to 50. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	3.96	4.00	4.27	5.48	7.46	8.42	8.52	8.44	8.31	8.14
1000	3.29	3.87	4.32	5.07	7.64	8.77	8.65	8.49	8.40	8.20
1800	3.29	3.87	4.23	4.83	7.97	8.79	8.64	8.52	8.35	8.19
3200	3.28	3.86	4.21	4.54	7.69	8.75	8.60	8.52	8.38	8.19
5600	3.28	3.84	4.19	4.52	7.73	8.81	8.66	8.55	8.39	8.24
10 000	3.25	3.78	4.18	4.51	7.91	8.76	8.69	8.51	8.36	8.25

Table SM.11: Sample Root Mean Squared Error (RMSE) for $\hat{\sigma}$ of model (3) obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ), the number of replications is equal to 50. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	10.50	10.73	10.78	10.40	8.59	6.89	6.23	6.35	6.30	6.22
1000	18.37	18.89	19.54	19.06	14.58	11.81	11.52	11.42	11.34	11.47
1800	33.33	34.11	35.20	34.93	25.00	20.49	20.02	20.50	20.43	20.32
3200	58.61	60.56	62.68	63.81	45.61	36.25	36.36	36.31	36.46	36.32
5600	102.70	105.50	108.88	110.84	80.55	64.35	64.04	63.56	63.58	63.35
10 000	182.85	191.19	194.84	199.70	137.64	114.04	114.48	113.98	114.32	114.35

Table SM.12: Computing times of model (3) obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ), the number of replications is equal to 50. The table reports the average computing times for one model, expressed in minutes.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	25.57	23.63	23.75	23.31	23.66	23.46	24.37	24.05	23.21	24.00
1000	25.06	25.18	24.42	24.13	23.47	23.91	23.94	23.96	24.09	23.63
1800	24.69	24.75	23.85	23.27	23.43	23.69	23.76	23.18	23.74	23.56
3200	25.44	24.62	24.12	23.88	23.77	24.12	23.41	23.50	23.40	23.39
5600	24.79	24.25	24.12	24.15	23.72	23.60	23.69	23.55	23.67	23.67
10000	25.01	24.25	24.06	24.10	23.99	23.73	23.54	23.51	23.37	23.52

Table SM.13: Sample Root Mean Squared Error (RMSE) for $\hat{\beta}_1$ of model (1) with $\Omega = \Xi \otimes I_G$ obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ), the number of replications is equal to 100. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	26.69	26.09	26.22	26.11	25.27	24.81	25.05	25.53	25.16	25.18
1000	25.33	25.46	25.38	25.26	25.38	24.87	24.88	24.60	25.60	25.36
1800	26.32	25.65	25.20	24.99	25.38	25.25	25.04	25.33	25.03	25.25
3200	25.47	25.56	24.89	25.27	25.13	24.77	24.68	24.73	24.89	25.05
5600	25.49	25.34	24.94	25.17	24.88	25.06	25.10	25.21	25.19	24.88
10000	25.61	25.47	25.16	25.16	24.88	25.04	24.95	25.00	24.99	25.05

Table SM.14: Sample Root Mean Squared Error (RMSE) for $\hat{\beta}_2$ of model (1) with $\Omega = \Xi \otimes I_G$ obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ), the number of replications is equal to 100. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	37.38	36.45	35.97	35.29	34.61	34.42	34.78	34.16	33.45	33.71
1000	36.48	37.21	35.46	35.66	33.76	34.35	33.95	33.48	33.85	32.79
1800	37.08	37.33	36.21	35.38	34.45	34.39	34.21	34.23	33.62	33.45
3200	37.81	36.50	36.41	35.62	35.46	35.36	34.31	34.18	33.50	33.49
5600	37.46	36.26	36.28	35.57	34.43	34.92	34.75	34.05	33.76	33.41
10000	38.14	36.55	36.00	35.90	35.04	35.14	34.42	34.26	33.66	34.01

Table SM.15: Sample Root Mean Squared Error (RMSE) for $\hat{\beta}_3$ of model (1) with $\Omega = \Xi \otimes I_G$ obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ), the number of replications is equal to 100. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	67.60	64.71	63.33	64.23	64.20	62.53	63.99	63.78	63.11	63.83
1000	66.72	65.18	64.51	65.52	63.81	64.43	63.53	62.78	63.15	63.06
1800	67.07	66.12	65.37	64.60	63.87	64.15	63.65	63.58	63.64	63.06
3200	67.04	65.41	63.80	65.00	63.86	63.68	63.08	63.47	63.35	63.17
5600	66.88	65.28	65.09	64.54	63.90	63.72	63.73	63.53	63.72	63.21
10000	66.23	65.42	64.84	64.58	64.30	63.82	63.80	63.28	63.32	63.04

Table SM.16: Sample Root Mean Squared Error (RMSE) for $\hat{\beta}_4$ of model (1) with $\Omega = \Xi \otimes I_G$ obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ), the number of replications is equal to 100. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	12.14	11.12	12.47	12.77	11.86	11.31	12.42	9.85	10.82	11.12
1000	12.25	11.67	12.07	12.22	10.22	11.37	11.14	11.58	10.78	11.30
1800	12.65	12.74	11.98	11.78	11.80	12.29	11.23	10.78	10.76	10.63
3200	13.06	12.59	11.53	12.17	11.57	11.52	11.85	10.96	11.10	10.94
5600	11.75	12.08	12.20	12.01	11.50	11.79	11.60	11.15	11.23	10.87
10 000	12.72	12.17	12.34	12.00	11.84	11.50	11.48	11.25	11.44	10.78

Table SM.17: Sample Root Mean Squared Error (RMSE) for $\hat{\sigma}_1$ of model (1) with $\Omega = \Xi \otimes I_G$ obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ), the number of replications is equal to 100. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	17.57	17.78	18.38	18.37	17.81	18.14	18.07	17.84	17.84	17.76
1000	18.36	18.63	19.12	18.51	18.70	18.26	18.46	18.54	18.02	18.16
1800	18.86	18.83	18.92	18.77	18.75	18.28	18.62	18.18	18.44	18.26
3200	18.55	18.61	18.88	18.90	18.93	18.52	18.64	18.50	18.27	18.19
5600	19.07	18.92	19.10	18.96	18.81	18.66	18.63	18.48	18.40	18.46
10 000	19.02	19.07	19.08	18.95	18.83	18.69	18.53	18.45	18.36	18.32

Table SM.18: Sample Root Mean Squared Error (RMSE) for $\hat{\sigma}_2$ of model (1) with $\Omega = \Xi \otimes I_G$ obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ), the number of replications is equal to 100. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	42.28	4184.69	42.57	42.55	43.30	42.75	42.63	42.73	42.79	42.95
1000	41.70	42.51	42.77	43.53	43.17	43.08	42.97	43.12	43.71	42.91
1800	40.47	42.89	43.12	44.11	43.93	43.28	43.52	43.38	43.55	43.65
3200	39.81	42.56	42.86	43.64	43.71	43.89	43.45	43.82	43.76	43.69
5600	39.85	42.00	42.71	43.43	44.16	44.21	43.83	43.82	43.46	43.97
10 000	39.71	41.91	42.97	43.68	43.96	44.03	43.81	43.68	43.68	43.54

Table SM.19: Sample Root Mean Squared Error (RMSE) for $\hat{\sigma}_3$ of model (1) with $\Omega = \Xi \otimes I_G$ obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ), the number of replications is equal to 100. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	63.30	62.63	63.13	64.48	64.01	66.00	66.58	65.03	66.65	67.22
1000	58.32	60.95	62.56	62.69	62.52	64.39	65.15	66.12	64.63	66.95
1800	59.62	60.23	62.24	62.70	62.59	64.32	63.99	65.61	64.96	65.59
3200	57.43	60.57	60.90	62.53	63.14	63.64	65.06	64.51	65.80	66.41
5600	57.49	59.91	61.47	62.97	62.97	64.55	64.56	65.15	65.49	65.66
10 000	57.28	59.51	62.37	63.04	63.30	63.85	64.56	64.85	66.06	66.11

Table SM.20: Sample Root Mean Squared Error (RMSE) for $\hat{\sigma}_4$ of model (1) with $\Omega = \Xi \otimes I_G$ obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ), the number of replications is equal to 100. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	0.03	0.01	0.01	0.01	-0.01	0.00	-0.01	-0.01	-0.01	-0.02
1000	0.04	0.02	0.02	0.01	0.00	0.00	-0.01	-0.01	-0.01	-0.01
1800	0.04	0.03	0.01	0.01	0.00	0.00	0.00	-0.01	-0.01	-0.01
3200	0.03	0.02	0.02	0.01	0.01	0.00	0.00	-0.01	-0.01	-0.01
5600	0.03	0.03	0.02	0.01	0.01	0.00	0.00	0.00	-0.01	-0.01
10000	0.04	0.02	0.02	0.01	0.01	0.00	0.00	-0.01	-0.01	-0.01

Table SM.21: Sample bias for $\hat{\beta}_1$ of model (1) with $\Omega = \Xi \otimes I_G$ obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ) , the number of replications is equal to 100. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	0.02	0.00	0.00	0.00	-0.01	-0.02	-0.02	-0.03	-0.03	-0.03
1000	0.02	0.01	0.00	-0.01	-0.02	-0.01	-0.02	-0.02	-0.03	-0.03
1800	0.02	0.01	0.00	-0.01	-0.01	-0.02	-0.02	-0.03	-0.03	-0.03
3200	0.02	0.01	0.00	0.00	-0.01	-0.02	-0.02	-0.02	-0.03	-0.03
5600	0.02	0.01	0.00	0.00	-0.01	-0.01	-0.02	-0.02	-0.03	-0.03
10000	0.02	0.01	0.00	0.00	-0.01	-0.01	-0.02	-0.02	-0.02	-0.03

Table SM.22: Sample bias for $\hat{\beta}_2$ of model (1) with $\Omega = \Xi \otimes I_G$ obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ) , the number of replications is equal to 100. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	0.09	0.08	0.08	0.08	0.07	0.06	0.06	0.05	0.05	0.04
1000	0.10	0.09	0.08	0.07	0.06	0.07	0.06	0.05	0.05	0.04
1800	0.10	0.09	0.08	0.07	0.07	0.06	0.06	0.05	0.05	0.05
3200	0.10	0.09	0.08	0.08	0.07	0.07	0.06	0.05	0.05	0.05
5600	0.11	0.09	0.08	0.08	0.07	0.07	0.06	0.05	0.05	0.05
10000	0.11	0.10	0.08	0.08	0.07	0.07	0.06	0.06	0.05	0.05

Table SM.23: Sample bias for $\hat{\beta}_3$ of model (1) with $\Omega = \Xi \otimes I_G$ obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ) , the number of replications is equal to 100. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	0.01	-0.01	-0.02	-0.01	-0.02	-0.03	-0.02	-0.03	-0.03	-0.04
1000	0.01	0.02	0.00	-0.01	-0.01	-0.02	-0.03	-0.03	-0.04	-0.04
1800	0.03	0.01	0.00	0.00	-0.01	-0.02	-0.02	-0.03	-0.03	-0.04
3200	0.03	0.01	0.01	0.00	-0.01	-0.02	-0.02	-0.03	-0.03	-0.04
5600	0.02	0.01	0.01	0.00	-0.01	-0.01	-0.02	-0.02	-0.03	-0.03
10000	0.04	0.02	0.01	0.00	0.00	-0.01	-0.02	-0.02	-0.03	-0.03

Table SM.24: Sample bias for $\hat{\beta}_4$ of model (1) with $\Omega = \Xi \otimes I_G$ obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ) , the number of replications is equal to 100. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	-0.04	-0.06	-0.05	-0.05	-0.05	-0.06	-0.05	-0.07	-0.06	-0.06
1000	-0.04	-0.05	-0.05	-0.06	-0.07	-0.06	-0.06	-0.06	-0.06	-0.06
1800	-0.04	-0.05	-0.06	-0.06	-0.06	-0.06	-0.06	-0.07	-0.07	-0.07
3200	-0.04	-0.05	-0.06	-0.06	-0.06	-0.06	-0.06	-0.07	-0.06	-0.06
5600	-0.05	-0.06	-0.06	-0.06	-0.07	-0.06	-0.06	-0.07	-0.06	-0.06
10 000	-0.05	-0.06	-0.06	-0.06	-0.06	-0.07	-0.06	-0.07	-0.06	-0.07

Table SM.25: Sample bias for $\hat{\sigma}_1$ of model (1) with $\Omega = \Xi \otimes I_G$ obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ), the number of replications is equal to 100. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	-0.12	-0.14	-0.13	-0.16	-0.15	-0.15	-0.15	-0.15	-0.15	-0.16
1000	-0.13	-0.14	-0.15	-0.15	-0.15	-0.16	-0.15	-0.15	-0.16	-0.16
1800	-0.13	-0.14	-0.15	-0.15	-0.16	-0.16	-0.16	-0.16	-0.16	-0.16
3200	-0.14	-0.15	-0.15	-0.15	-0.15	-0.16	-0.16	-0.16	-0.16	-0.16
5600	-0.14	-0.15	-0.15	-0.16	-0.16	-0.16	-0.16	-0.16	-0.16	-0.16
10 000	-0.14	-0.15	-0.15	-0.15	-0.15	-0.16	-0.16	-0.16	-0.16	-0.16

Table SM.26: Sample bias for $\hat{\sigma}_2$ of model (1) with $\Omega = \Xi \otimes I_G$ obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ), the number of replications is equal to 100. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	-0.30	3.89	-0.33	-0.33	-0.34	-0.34	-0.35	-0.35	-0.36	-0.37
1000	-0.27	-0.31	-0.32	-0.35	-0.35	-0.35	-0.36	-0.36	-0.38	-0.36
1800	-0.27	-0.32	-0.32	-0.35	-0.35	-0.35	-0.36	-0.35	-0.37	-0.37
3200	-0.26	-0.30	-0.32	-0.34	-0.35	-0.36	-0.36	-0.36	-0.37	-0.37
5600	-0.26	-0.30	-0.32	-0.34	-0.35	-0.35	-0.36	-0.36	-0.37	-0.38
10 000	-0.26	-0.30	-0.33	-0.34	-0.35	-0.35	-0.36	-0.36	-0.37	-0.37

Table SM.27: Sample bias for $\hat{\sigma}_3$ of model (1) with $\Omega = \Xi \otimes I_G$ obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ), the number of replications is equal to 100. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	-0.37	-0.43	-0.45	-0.50	-0.49	-0.51	-0.53	-0.53	-0.55	-0.57
1000	-0.38	-0.43	-0.44	-0.47	-0.47	-0.50	-0.52	-0.54	-0.54	-0.56
1800	-0.37	-0.41	-0.45	-0.47	-0.48	-0.50	-0.51	-0.53	-0.54	-0.55
3200	-0.38	-0.42	-0.44	-0.46	-0.48	-0.50	-0.52	-0.52	-0.54	-0.56
5600	-0.37	-0.41	-0.44	-0.47	-0.48	-0.50	-0.52	-0.53	-0.54	-0.55
10 000	-0.38	-0.41	-0.45	-0.47	-0.48	-0.49	-0.51	-0.52	-0.54	-0.55

Table SM.28: Sample bias for $\hat{\sigma}_4$ of model (1) with $\Omega = \Xi \otimes I_G$ obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ), the number of replications is equal to 100. All values of the RMSE are multiplied by 100.

N	ρ									
	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
560	0.68	0.68	0.69	0.78	0.74	0.78	0.84	0.83	0.85	0.89
1000	1.30	1.36	1.55	1.57	1.58	1.70	1.61	1.74	1.75	1.75
1800	2.44	2.70	2.91	3.37	3.32	3.43	3.63	3.59	3.49	3.82
3200	4.43	5.11	5.91	6.52	6.99	7.03	7.31	7.40	7.37	7.25
5600	9.11	10.81	12.84	13.59	13.47	13.27	15.00	14.33	14.45	14.34
10000	18.52	22.93	25.16	27.15	27.59	29.11	31.65	32.84	30.58	29.81

Table SM.29: Computing times of model (1) with $\Omega = \Xi \otimes I_G$ obtained by means of Monte Carlo simulations for various values of the number of points (N) and the rarity parameter (ρ). For each pair (N, ρ) , the number of replications is equal to 100. The table reports the average computing times for one model, expressed in minutes.

q	β									
	0.90	0.91	0.92	0.93	0.94	0.95	0.96	0.97	0.98	0.99
5	33.30	33.56	33.90	34.11	33.85	34.14	34.30	34.46	34.76	35.06
6	32.59	32.65	32.86	33.28	33.09	33.48	33.46	33.49	33.55	33.91
7	32.05	32.34	32.47	32.66	32.96	32.57	33.08	32.75	32.62	32.95
8	32.29	32.13	31.92	31.85	32.10	32.55	32.18	32.08	32.57	32.67
9	31.64	31.84	31.91	31.76	31.91	32.01	32.04	32.35	32.05	32.10
10	31.81	31.64	31.55	31.86	31.92	31.88	31.90	32.06	32.05	31.78

Table SM.30: Sample Root Mean Squared Error (RMSE) for $\hat{\beta}_1$ of model (1) obtained by means of Monte Carlo simulations for various values of the smoothing parameters q and β of Equation (15). For each pair (q, β) , the number of replications is equal to 250. All values of the RMSE are multiplied by 100.

q	β									
	0.90	0.91	0.92	0.93	0.94	0.95	0.96	0.97	0.98	0.99
5	52.67	53.31	53.62	54.03	54.25	54.74	55.46	55.65	56.07	56.79
6	50.36	50.62	51.08	51.80	51.91	52.59	52.15	52.51	53.33	53.17
7	48.81	49.70	50.29	50.54	50.74	50.51	50.74	50.74	50.90	50.89
8	48.75	48.49	48.63	48.78	48.87	49.27	49.11	49.13	50.20	50.45
9	47.23	47.76	48.16	48.16	48.25	48.34	48.09	48.87	49.19	48.89
10	47.16	47.16	47.27	47.66	47.58	47.61	48.04	48.16	47.95	48.08

Table SM.31: Sample Root Mean Squared Error (RMSE) for $\hat{\beta}_2$ of model (1) obtained by means of Monte Carlo simulations for various values of the smoothing parameters q and β of Equation (15). For each pair (q, β) , the number of replications is equal to 250. All values of the RMSE are multiplied by 100.

q	β									
	0.90	0.91	0.92	0.93	0.94	0.95	0.96	0.97	0.98	0.99
5	69.46	70.90	71.62	71.88	72.28	73.19	73.42	73.86	74.67	75.43
6	67.10	67.30	67.57	68.09	68.78	69.84	69.43	69.86	71.37	70.40
7	65.55	66.10	66.62	67.06	67.36	67.02	67.85	67.48	67.83	67.70
8	64.57	64.58	64.89	65.00	65.57	65.55	65.27	65.13	66.30	66.97
9	62.92	64.09	64.13	64.10	64.18	64.07	63.95	64.83	65.26	64.52
10	62.81	63.09	62.71	63.00	63.57	63.16	64.15	63.79	63.81	64.31

Table SM.32: Sample Root Mean Squared Error (RMSE) for $\hat{\beta}_3$ of model (1) obtained by means of Monte Carlo simulations for various values of the smoothing parameters q and β of Equation (15). For each pair (q, β) , the number of replications is equal to 250. All values of the RMSE are multiplied by 100.

q	β									
	0.90	0.91	0.92	0.93	0.94	0.95	0.96	0.97	0.98	0.99
5	92.02	94.05	94.18	95.28	95.89	96.67	97.24	98.14	100.30	101.00
6	88.36	89.00	89.67	90.62	90.78	92.51	92.34	92.65	94.23	93.50
7	86.30	87.30	87.87	88.97	89.41	88.90	90.24	89.26	89.22	89.79
8	85.69	84.93	85.07	85.58	86.28	86.75	86.15	85.96	87.41	88.56
9	82.89	83.93	84.37	84.35	84.39	84.45	84.95	85.40	86.23	85.45
10	83.04	82.79	83.00	83.24	83.67	82.90	84.50	84.04	84.14	84.67

Table SM.33: Sample Root Mean Squared Error (RMSE) for $\hat{\beta}_4$ of model (1) obtained by means of Monte Carlo simulations for various values of the smoothing parameters q and β of Equation (15). For each pair (q, β) , the number of replications is equal to 250. All values of the RMSE are multiplied by 100.

q	β									
	0.90	0.91	0.92	0.93	0.94	0.95	0.96	0.97	0.98	0.99
5	12.38	12.38	12.32	12.50	12.42	12.51	12.52	12.53	12.57	12.66
6	12.25	12.41	12.31	12.36	12.29	12.37	12.43	12.35	12.32	12.31
7	12.16	12.27	12.37	12.18	12.36	12.27	12.24	12.32	12.38	12.16
8	12.21	12.31	12.27	12.11	12.11	12.24	12.22	12.11	12.44	12.19
9	12.19	12.30	12.21	12.14	12.17	12.15	12.15	12.19	12.22	12.19
10	12.20	12.16	12.13	12.08	12.10	12.17	12.28	12.32	12.20	12.13

Table SM.34: Sample Root Mean Squared Error (RMSE) for $\hat{\sigma}$ of model (1) obtained by means of Monte Carlo simulations for various values of the smoothing parameters q and β of Equation (15). For each pair (q, β) , the number of replications is equal to 250. All values of the RMSE are multiplied by 100.

q	β									
	0.90	0.91	0.92	0.93	0.94	0.95	0.96	0.97	0.98	0.99
5	19.57	18.79	18.48	18.25	17.99	17.82	17.60	17.09	17.36	16.91
6	18.74	18.21	17.96	18.15	17.24	17.56	17.66	17.45	16.88	16.86
7	19.18	18.87	18.90	17.97	17.96	17.78	17.84	17.39	17.53	17.60
8	20.26	19.84	20.05	19.58	19.90	19.79	19.08	18.16	18.69	19.11
9	22.16	23.48	22.78	22.74	22.01	21.63	21.43	20.89	21.32	20.88
10	25.23	24.72	23.79	24.35	23.12	22.94	22.68	22.85	22.70	22.76

Table SM.35: Computing times of model (1) obtained by means of Monte Carlo simulations for various values of the smoothing parameters q and β of Equation (15). For each pair (q, β) , the number of replications is equal to 250. The table reports the average computing times for one model, expressed in seconds.

RMSE of $\hat{\beta}_0$	σ				
	0.01	0.10	0.20	0.50	1.00
AG quadrature ($m = 1$)	30.85	32.19	27.82	28.59	29.31
AG quadrature ($m = 3$)	30.85	32.18	27.82	28.58	29.31
AG quadrature ($m = 5$)	30.85	32.19	27.82	28.58	29.31
Cross-entropy ($N = 1000$)	30.28	32.17	27.47	28.74	29.44
Cross-entropy ($N = 2000$)	30.60	32.31	27.55	28.71	29.60
Cross-entropy ($N = 5000$)	30.38	32.23	27.47	28.73	29.47
RMSE of $\hat{\beta}_1$					
AG quadrature ($m = 1$)	35.81	45.97	40.33	38.72	41.78
AG quadrature ($m = 3$)	35.80	45.96	40.32	38.71	41.78
AG quadrature ($m = 5$)	35.81	45.97	40.33	38.72	41.78
Cross-entropy ($N = 1000$)	35.90	45.65	39.93	39.17	42.13
Cross-entropy ($N = 2000$)	35.22	45.25	40.03	38.81	42.16
Cross-entropy ($N = 5000$)	35.42	45.52	40.16	38.94	41.94
RMSE of $\hat{\beta}_2$					
AG quadrature ($m = 1$)	53.39	57.88	52.47	57.48	57.77
AG quadrature ($m = 3$)	53.38	57.86	52.46	57.46	57.77
AG quadrature ($m = 5$)	53.39	57.87	52.47	57.47	57.77
Cross-entropy ($N = 1000$)	52.61	58.28	52.48	58.24	57.90
Cross-entropy ($N = 2000$)	52.22	57.93	52.71	56.81	58.09
Cross-entropy ($N = 5000$)	52.02	58.06	52.62	57.52	57.92
RMSE of $\hat{\beta}_3$					
AG quadrature ($m = 1$)	69.37	84.50	72.04	72.93	77.63
AG quadrature ($m = 3$)	69.37	84.49	72.03	72.92	77.62
AG quadrature ($m = 5$)	69.38	84.51	72.04	72.93	77.63
Cross-entropy ($N = 1000$)	67.95	85.13	71.55	73.92	78.84
Cross-entropy ($N = 2000$)	67.36	84.24	71.58	73.14	78.55
Cross-entropy ($N = 5000$)	67.42	84.99	71.73	73.23	78.57
RMSE of $\hat{\sigma}$					
AG quadrature ($m = 1$)	28.08	26.17	25.44	43.18	90.11
AG quadrature ($m = 3$)	28.10	26.20	25.45	43.18	90.10
AG quadrature ($m = 5$)	28.13	26.23	25.47	43.18	90.09
Cross-entropy ($N = 1000$)	40.02	30.85	22.20	15.83	63.34
Cross-entropy ($N = 2000$)	39.92	31.72	21.83	15.84	63.15
Cross-entropy ($N = 5000$)	40.01	31.36	22.00	15.76	63.13
Computing time					
AG quadrature ($m = 1$)	0.30	0.21	0.19	0.19	0.19
AG quadrature ($m = 3$)	0.26	0.24	0.22	0.22	0.22
AG quadrature ($m = 5$)	0.28	0.26	0.24	0.24	0.24
Cross-entropy ($N = 1000$)	23.65	21.79	20.47	19.83	20.34
Cross-entropy ($N = 2000$)	45.32	41.21	40.93	39.19	39.76
Cross-entropy ($N = 5000$)	110.79	98.41	99.85	98.27	97.90

Table SM.36: Sample Root Mean Squared Error (RMSE) for all the parameters of model (1) and computing times obtained by means of Monte Carlo simulation for various estimation techniques and various values of σ and for $G = 8$ and $n = 250$. For each value of σ , the number of replications is equal to 200. All the values of the RMSE are multiplied by 100; computing times are in seconds.

Bias of $\hat{\beta}_0$	σ				
	0.01	0.10	0.20	0.50	1.00
AG quadrature ($m = 1$)	6.19	7.60	3.07	9.12	7.34
AG quadrature ($m = 3$)	6.19	7.59	3.06	9.12	7.34
AG quadrature ($m = 5$)	6.19	7.60	3.06	9.12	7.34
Cross-entropy ($N = 1000$)	6.15	7.57	3.21	9.48	7.24
Cross-entropy ($N = 2000$)	6.00	7.67	3.01	9.53	7.54
Cross-entropy ($N = 5000$)	6.14	7.70	3.14	9.54	7.55
Bias of $\hat{\beta}_1$					
AG quadrature ($m = 1$)	11.05	16.27	10.10	12.69	10.19
AG quadrature ($m = 3$)	11.04	16.27	10.09	12.69	10.19
AG quadrature ($m = 5$)	11.05	16.28	10.09	12.69	10.19
Cross-entropy ($N = 1000$)	11.45	15.96	10.03	13.13	10.60
Cross-entropy ($N = 2000$)	10.88	16.08	10.15	12.72	10.66
Cross-entropy ($N = 5000$)	10.82	16.18	10.26	12.91	10.72
Bias of $\hat{\beta}_2$					
AG quadrature ($m = 1$)	18.73	21.32	10.75	19.64	15.19
AG quadrature ($m = 3$)	18.73	21.31	10.74	19.64	15.19
AG quadrature ($m = 5$)	18.74	21.32	10.75	19.64	15.19
Cross-entropy ($N = 1000$)	18.99	21.24	11.43	20.33	15.33
Cross-entropy ($N = 2000$)	18.67	21.07	10.85	19.68	15.63
Cross-entropy ($N = 5000$)	18.53	21.30	11.27	20.05	15.83
Bias of $\hat{\beta}_3$					
AG quadrature ($m = 1$)	23.76	32.95	19.61	28.74	26.11
AG quadrature ($m = 3$)	23.76	32.96	19.61	28.73	26.11
AG quadrature ($m = 5$)	23.77	32.97	19.61	28.74	26.12
Cross-entropy ($N = 1000$)	23.94	32.51	19.39	29.35	26.38
Cross-entropy ($N = 2000$)	23.79	32.83	19.35	29.21	26.63
Cross-entropy ($N = 5000$)	23.65	33.04	19.75	29.39	26.71
Bias of $\hat{\sigma}$					
AG quadrature ($m = 1$)	14.23	5.16	-5.54	-36.41	-87.19
AG quadrature ($m = 3$)	14.24	5.18	-5.53	-36.40	-87.18
AG quadrature ($m = 5$)	14.26	5.19	-5.52	-36.39	-87.16
Cross-entropy ($N = 1000$)	38.39	28.80	19.30	-11.40	-62.39
Cross-entropy ($N = 2000$)	38.41	29.51	19.09	-11.40	-62.19
Cross-entropy ($N = 5000$)	38.50	29.26	19.23	-11.10	-62.19

Table SM.37: Bias for all the parameters of model (1) obtained by means of Monte Carlo simulation for various estimation techniques and various values of σ and for $G = 8$ and $n = 250$. For each value of σ , the number of replications is equal to 200. All the values of the bias are multiplied by 100.

RMSE of $\hat{\beta}_0$	σ				
	0.01	0.10	0.20	0.50	1.00
AG quadrature ($m = 1$)	29.16	28.73	30.61	30.34	30.52
AG quadrature ($m = 3$)	29.13	28.71	30.59	30.33	30.50
AG quadrature ($m = 5$)	29.14	28.72	30.60	30.33	30.51
Cross-entropy ($N = 1000$)	28.73	27.90	30.03	30.11	30.38
Cross-entropy ($N = 2000$)	28.59	28.36	29.92	29.80	30.76
Cross-entropy ($N = 5000$)	28.69	28.20	29.76	29.86	30.54
RMSE of $\hat{\beta}_1$					
AG quadrature ($m = 1$)	48.02	43.05	48.24	45.08	49.16
AG quadrature ($m = 3$)	47.99	43.04	48.23	45.05	49.14
AG quadrature ($m = 5$)	48.02	43.08	48.24	45.07	49.15
Cross-entropy ($N = 1000$)	46.05	39.59	48.68	44.08	48.71
Cross-entropy ($N = 2000$)	45.63	39.67	48.61	44.49	49.29
Cross-entropy ($N = 5000$)	45.90	39.97	48.42	44.39	48.94
RMSE of $\hat{\beta}_2$					
AG quadrature ($m = 1$)	61.14	63.12	56.78	68.26	64.05
AG quadrature ($m = 3$)	61.09	63.17	56.76	68.19	64.00
AG quadrature ($m = 5$)	61.13	63.23	56.77	68.21	64.02
Cross-entropy ($N = 1000$)	58.60	56.62	56.67	67.54	63.38
Cross-entropy ($N = 2000$)	57.98	57.30	56.85	67.70	63.88
Cross-entropy ($N = 5000$)	58.44	57.57	56.56	67.43	63.87
RMSE of $\hat{\beta}_3$					
AG quadrature ($m = 1$)	79.63	83.91	80.30	80.85	89.07
AG quadrature ($m = 3$)	79.60	83.98	80.28	80.79	89.02
AG quadrature ($m = 5$)	79.64	84.06	80.31	80.82	89.06
Cross-entropy ($N = 1000$)	77.56	74.08	79.29	80.55	88.37
Cross-entropy ($N = 2000$)	77.03	75.18	79.35	81.03	88.66
Cross-entropy ($N = 5000$)	77.63	75.04	79.05	80.37	88.39
RMSE of $\hat{\sigma}$					
AG quadrature ($m = 1$)	34.08	33.94	28.05	40.08	85.46
AG quadrature ($m = 3$)	34.13	34.04	28.07	40.09	85.45
AG quadrature ($m = 5$)	34.21	34.14	28.13	40.10	85.43
Cross-entropy ($N = 1000$)	40.45	31.83	22.08	13.30	60.38
Cross-entropy ($N = 2000$)	40.83	32.74	22.45	13.34	60.07
Cross-entropy ($N = 5000$)	41.02	32.50	22.37	13.13	60.03
Computing time					
AG quadrature ($m = 1$)	0.20	0.20	0.19	0.20	0.21
AG quadrature ($m = 3$)	0.23	0.23	0.23	0.23	0.24
AG quadrature ($m = 5$)	0.25	0.25	0.25	0.26	0.26
Cross-entropy ($N = 1000$)	20.28	19.68	20.37	20.25	21.02
Cross-entropy ($N = 2000$)	39.39	39.18	40.20	40.67	42.74
Cross-entropy ($N = 5000$)	99.93	98.18	99.43	101.17	105.74

Table SM.38: Sample Root Mean Squared Error (RMSE) for all the parameters of model (1) and computing times obtained by means of Monte Carlo simulation for various estimation techniques and various values of σ and for $G = 12$ and $n = 250$. For each value of σ , the number of replications is equal to 200. All the values of the RMSE are multiplied by 100; computing times are in seconds.

Bias of $\hat{\beta}_0$	σ				
	0.01	0.10	0.20	0.50	1.00
AG quadrature ($m = 1$)	6.03	9.90	9.48	4.00	11.49
AG quadrature ($m = 3$)	6.02	9.89	9.47	3.99	11.48
AG quadrature ($m = 5$)	6.03	9.90	9.48	4.00	11.48
Cross-entropy ($N = 1000$)	5.74	9.11	8.91	3.62	11.34
Cross-entropy ($N = 2000$)	5.64	9.54	8.82	3.49	11.50
Cross-entropy ($N = 5000$)	5.74	9.13	8.92	3.72	11.40
Bias of $\hat{\beta}_1$					
AG quadrature ($m = 1$)	14.15	16.00	18.24	13.98	20.23
AG quadrature ($m = 3$)	14.14	15.99	18.23	13.97	20.22
AG quadrature ($m = 5$)	14.16	16.01	18.25	13.98	20.23
Cross-entropy ($N = 1000$)	12.75	13.90	17.36	12.81	19.51
Cross-entropy ($N = 2000$)	12.86	14.25	17.34	12.86	19.89
Cross-entropy ($N = 5000$)	13.06	14.15	17.27	13.22	19.67
Bias of $\hat{\beta}_2$					
AG quadrature ($m = 1$)	21.36	21.37	23.86	22.08	26.88
AG quadrature ($m = 3$)	21.35	21.37	23.85	22.06	26.85
AG quadrature ($m = 5$)	21.37	21.41	23.87	22.08	26.87
Cross-entropy ($N = 1000$)	19.42	18.13	22.91	20.45	25.63
Cross-entropy ($N = 2000$)	19.18	18.50	22.74	20.83	26.25
Cross-entropy ($N = 5000$)	19.89	18.42	22.71	20.91	26.27
Bias of $\hat{\beta}_3$					
AG quadrature ($m = 1$)	30.73	30.08	31.64	25.80	42.36
AG quadrature ($m = 3$)	30.73	30.09	31.64	25.79	42.35
AG quadrature ($m = 5$)	30.76	30.14	31.67	25.82	42.37
Cross-entropy ($N = 1000$)	28.61	25.41	29.87	23.19	41.26
Cross-entropy ($N = 2000$)	28.18	26.14	30.06	24.14	41.75
Cross-entropy ($N = 5000$)	29.01	25.67	29.74	24.34	41.45
Bias of $\hat{\sigma}$					
AG quadrature ($m = 1$)	18.54	11.62	-1.47	-29.10	-81.52
AG quadrature ($m = 3$)	18.57	11.66	-1.45	-29.08	-81.51
AG quadrature ($m = 5$)	18.61	11.70	-1.41	-29.04	-81.48
Cross-entropy ($N = 1000$)	39.38	30.41	19.96	-9.36	-59.66
Cross-entropy ($N = 2000$)	39.78	31.18	20.32	-9.27	-59.37
Cross-entropy ($N = 5000$)	40.04	31.06	20.36	-9.25	-59.35

Table SM.39: Bias for all the parameters of model (1) obtained by means of Monte Carlo simulation for various estimation techniques and various values of σ and for $G = 12$ and $n = 250$. For each value of σ , the number of replications is equal to 200. All the values of the bias are multiplied by 100.

RMSE of $\hat{\beta}_0$	σ				
	0.01	0.10	0.20	0.50	1.00
AG quadrature ($m = 1$)	30.26	37.22	37.24	30.79	33.94
AG quadrature ($m = 3$)	30.18	36.77	36.98	30.70	33.72
AG quadrature ($m = 5$)	30.22	37.04	37.11	30.75	33.85
Cross-entropy ($N = 1000$)	30.05	33.30	34.57	29.43	31.54
Cross-entropy ($N = 2000$)	30.26	33.20	34.51	29.51	31.52
Cross-entropy ($N = 5000$)	30.36	33.14	34.54	29.62	31.44
RMSE of $\hat{\beta}_1$					
AG quadrature ($m = 1$)	48.92	53.11	56.87	47.95	44.24
AG quadrature ($m = 3$)	48.75	52.43	56.49	47.82	43.95
AG quadrature ($m = 5$)	48.88	53.10	56.74	47.92	44.17
Cross-entropy ($N = 1000$)	47.57	43.75	52.12	46.77	40.38
Cross-entropy ($N = 2000$)	47.18	43.14	51.52	46.67	40.36
Cross-entropy ($N = 5000$)	47.60	43.12	52.08	46.77	40.43
RMSE of $\hat{\beta}_2$					
AG quadrature ($m = 1$)	58.91	80.12	74.15	70.76	59.14
AG quadrature ($m = 3$)	58.55	78.91	73.53	70.52	58.64
AG quadrature ($m = 5$)	58.80	79.89	73.98	70.71	59.01
Cross-entropy ($N = 1000$)	55.30	66.05	65.86	66.99	54.73
Cross-entropy ($N = 2000$)	54.24	64.89	65.21	67.29	54.51
Cross-entropy ($N = 5000$)	54.93	64.63	65.64	67.70	54.21
RMSE of $\hat{\beta}_3$					
AG quadrature ($m = 1$)	82.07	108.53	104.37	95.68	80.64
AG quadrature ($m = 3$)	81.75	106.90	103.56	95.51	80.14
AG quadrature ($m = 5$)	82.02	108.45	104.18	95.81	80.64
Cross-entropy ($N = 1000$)	79.08	88.06	92.33	89.50	74.04
Cross-entropy ($N = 2000$)	78.55	86.77	91.90	89.64	74.32
Cross-entropy ($N = 5000$)	78.94	86.17	92.03	89.79	73.75
RMSE of $\hat{\sigma}$					
AG quadrature ($m = 1$)	38.16	43.79	39.55	43.05	86.16
AG quadrature ($m = 3$)	37.89	42.99	38.81	42.98	86.06
AG quadrature ($m = 5$)	38.28	43.79	39.42	43.14	86.09
Cross-entropy ($N = 1000$)	39.61	32.14	22.40	10.93	59.90
Cross-entropy ($N = 2000$)	40.30	32.01	22.60	10.79	59.73
Cross-entropy ($N = 5000$)	40.23	32.50	22.76	10.57	59.65
Computing time					
AG quadrature ($m = 1$)	0.34	0.24	0.22	0.22	0.21
AG quadrature ($m = 3$)	0.30	0.28	0.26	0.26	0.24
AG quadrature ($m = 5$)	0.33	0.31	0.28	0.28	0.27
Cross-entropy ($N = 1000$)	26.38	23.24	22.49	22.23	21.55
Cross-entropy ($N = 2000$)	49.39	43.91	43.95	42.75	43.42
Cross-entropy ($N = 5000$)	116.40	108.10	108.12	107.31	107.54

Table SM.40: Sample Root Mean Squared Error (RMSE) for all the parameters of model (1) obtained by means of Monte Carlo simulations for various estimation techniques and various values of σ and for $G = 25$ and $n = 250$. For each value of σ , the number of replications is equal to 200. All values of the RMSE are multiplied by 100.

Bias of $\hat{\beta}_0$	σ				
	0.01	0.10	0.20	0.50	1.00
AG quadrature ($m = 1$)	9.39	10.54	11.71	11.28	8.15
AG quadrature ($m = 3$)	9.32	10.35	11.56	11.22	8.05
AG quadrature ($m = 5$)	9.36	10.47	11.65	11.27	8.11
Cross-entropy ($N = 1000$)	8.55	7.79	9.23	9.83	7.00
Cross-entropy ($N = 2000$)	8.53	7.97	9.51	9.90	7.00
Cross-entropy ($N = 5000$)	8.66	7.92	9.48	10.00	7.03
Bias of $\hat{\beta}_1$					
AG quadrature ($m = 1$)	17.25	19.22	25.19	18.12	15.83
AG quadrature ($m = 3$)	17.17	19.00	25.00	18.06	15.71
AG quadrature ($m = 5$)	17.25	19.22	25.15	18.14	15.82
Cross-entropy ($N = 1000$)	15.30	14.15	20.81	16.02	13.41
Cross-entropy ($N = 2000$)	15.32	14.02	20.72	15.74	13.08
Cross-entropy ($N = 5000$)	15.35	14.21	21.09	16.05	13.17
Bias of $\hat{\beta}_2$					
AG quadrature ($m = 1$)	20.34	30.40	32.84	26.41	20.31
AG quadrature ($m = 3$)	20.21	30.01	32.58	26.30	20.12
AG quadrature ($m = 5$)	20.33	30.35	32.80	26.44	20.28
Cross-entropy ($N = 1000$)	17.23	23.03	26.29	22.48	16.45
Cross-entropy ($N = 2000$)	17.01	22.72	26.52	22.69	16.90
Cross-entropy ($N = 5000$)	17.11	22.89	27.08	22.80	16.61
Bias of $\hat{\beta}_3$					
AG quadrature ($m = 1$)	34.85	41.02	50.75	40.21	31.64
AG quadrature ($m = 3$)	34.72	40.54	50.41	40.12	31.45
AG quadrature ($m = 5$)	34.88	41.02	50.72	40.30	31.66
Cross-entropy ($N = 1000$)	30.74	30.75	41.74	34.69	26.73
Cross-entropy ($N = 2000$)	30.79	30.29	41.69	35.15	27.18
Cross-entropy ($N = 5000$)	30.82	30.49	42.10	34.91	26.77
Bias of $\hat{\sigma}$					
AG quadrature ($m = 1$)	19.54	16.23	5.88	-27.35	-78.87
AG quadrature ($m = 3$)	19.49	15.94	5.67	-27.40	-79.02
AG quadrature ($m = 5$)	19.65	16.26	5.92	-27.23	-78.83
Cross-entropy ($N = 1000$)	39.14	31.12	21.30	-8.83	-59.51
Cross-entropy ($N = 2000$)	39.81	31.14	21.50	-8.65	-59.33
Cross-entropy ($N = 5000$)	39.78	31.59	21.68	-8.39	-59.30

Table SM.41: Bias for all the parameters of model (1) obtained by means of Monte Carlo simulation for various estimation techniques and various values of σ and for $G = 25$ and $n = 250$. For each value of σ , the number of replications is equal to 200. All the values of the bias are multiplied by 100.

	σ				
	0.01	0.10	0.20	0.50	1.00
RMSE of $\hat{\beta}_0$					
AG quadrature ($m = 1$)	269.64	137.11	45.04	34.29	187.36
AG quadrature ($m = 3$)	34.79	134.51	38.49	33.48	126.86
AG quadrature ($m = 5$)	35.84	113.69	39.38	33.79	258.89
Cross-entropy ($N = 1000$)	29.08	28.08	32.47	31.94	30.68
Cross-entropy ($N = 2000$)	29.42	28.18	32.36	31.82	30.76
Cross-entropy ($N = 5000$)	29.07	28.10	32.41	31.98	30.76
RMSE of $\hat{\beta}_1$					
AG quadrature ($m = 1$)	450.98	212.74	71.31	45.50	483.74
AG quadrature ($m = 3$)	50.70	225.85	53.12	44.40	294.71
AG quadrature ($m = 5$)	52.70	219.29	54.83	44.92	423.53
Cross-entropy ($N = 1000$)	41.91	39.34	43.67	41.86	51.45
Cross-entropy ($N = 2000$)	41.93	39.22	42.95	41.33	51.21
Cross-entropy ($N = 5000$)	41.88	39.58	43.29	41.63	51.61
RMSE of $\hat{\beta}_2$					
AG quadrature ($m = 1$)	523.82	363.11	87.71	68.97	658.52
AG quadrature ($m = 3$)	72.79	403.71	69.50	66.92	392.21
AG quadrature ($m = 5$)	75.80	392.26	71.83	67.84	645.28
Cross-entropy ($N = 1000$)	55.51	53.77	56.01	61.04	62.51
Cross-entropy ($N = 2000$)	55.61	53.81	55.72	60.59	61.95
Cross-entropy ($N = 5000$)	55.33	53.60	55.67	60.93	62.45
RMSE of $\hat{\beta}_3$					
AG quadrature ($m = 1$)	951.81	486.40	128.73	90.05	965.47
AG quadrature ($m = 3$)	103.17	524.54	98.23	87.73	516.14
AG quadrature ($m = 5$)	107.86	483.75	101.66	88.86	926.51
Cross-entropy ($N = 1000$)	76.99	73.74	78.09	81.96	81.37
Cross-entropy ($N = 2000$)	77.86	73.49	77.29	80.56	81.11
Cross-entropy ($N = 5000$)	77.61	73.81	78.02	81.16	81.62
RMSE of $\hat{\sigma}$					
AG quadrature ($m = 1$)	561.45	393.75	73.21	47.91	801.28
AG quadrature ($m = 3$)	58.16	305.73	51.27	45.98	250.10
AG quadrature ($m = 5$)	61.32	237.42	54.14	46.99	286.78
Cross-entropy ($N = 1000$)	40.51	31.54	22.06	10.81	59.19
Cross-entropy ($N = 2000$)	41.24	31.77	22.74	10.32	58.69
Cross-entropy ($N = 5000$)	41.24	31.83	22.63	10.13	58.45
Computing time					
AG quadrature ($m = 1$)	0.27	0.25	0.25	0.24	0.26
AG quadrature ($m = 3$)	0.29	0.29	0.29	0.28	0.30
AG quadrature ($m = 5$)	0.32	0.32	0.31	0.30	0.32
Cross-entropy ($N = 1000$)	24.39	25.06	24.59	24.83	24.80
Cross-entropy ($N = 2000$)	48.07	48.42	49.11	48.78	48.45
Cross-entropy ($N = 5000$)	119.64	120.12	120.57	120.86	120.87

Table SM.42: Sample Root Mean Squared Error (RMSE) for all the parameters of model (1) obtained by means of Monte Carlo simulations for various estimation techniques and various values of σ and for $G = 50$ and $n = 250$. For each value of σ , the number of replications is equal to 200. All values of the RMSE are multiplied by 100.

Bias of $\hat{\beta}_0$	σ				
	0.01	0.10	0.20	0.50	1.00
AG quadrature ($m = 1$)	30.67	20.75	12.78	9.97	30.72
AG quadrature ($m = 3$)	11.26	19.11	10.56	9.51	25.54
AG quadrature ($m = 5$)	11.78	15.16	11.02	9.73	33.17
Cross-entropy ($N = 1000$)	7.10	3.02	6.25	7.31	9.96
Cross-entropy ($N = 2000$)	7.41	2.91	6.15	7.10	9.79
Cross-entropy ($N = 5000$)	7.25	3.02	6.40	7.06	9.94
Bias of $\hat{\beta}_1$					
AG quadrature ($m = 1$)	54.32	39.76	29.89	21.16	69.91
AG quadrature ($m = 3$)	22.05	39.40	25.84	20.44	56.04
AG quadrature ($m = 5$)	23.04	34.15	26.68	20.86	60.04
Cross-entropy ($N = 1000$)	13.83	10.41	17.50	15.44	20.49
Cross-entropy ($N = 2000$)	14.00	10.73	17.71	15.32	20.01
Cross-entropy ($N = 5000$)	13.92	10.90	17.60	15.57	20.22
Bias of $\hat{\beta}_2$					
AG quadrature ($m = 1$)	69.82	64.62	40.90	27.08	87.97
AG quadrature ($m = 3$)	31.76	65.68	35.98	25.97	68.50
AG quadrature ($m = 5$)	33.19	56.20	37.18	26.60	79.86
Cross-entropy ($N = 1000$)	19.31	16.70	24.55	18.31	21.26
Cross-entropy ($N = 2000$)	19.43	16.81	24.04	18.21	20.67
Cross-entropy ($N = 5000$)	19.36	16.82	24.27	18.61	21.10
Bias of $\hat{\beta}_3$					
AG quadrature ($m = 1$)	114.41	90.78	59.69	38.23	120.01
AG quadrature ($m = 3$)	46.30	91.51	52.34	36.87	87.84
AG quadrature ($m = 5$)	48.33	76.99	54.01	37.70	109.19
Cross-entropy ($N = 1000$)	28.20	26.64	36.20	27.36	26.13
Cross-entropy ($N = 2000$)	29.27	26.94	35.63	26.64	25.91
Cross-entropy ($N = 5000$)	29.10	27.04	36.03	27.04	26.28
Bias of $\hat{\sigma}$					
AG quadrature ($m = 1$)	76.27	62.46	19.31	-20.02	0.48
AG quadrature ($m = 3$)	36.01	51.43	14.62	-21.09	-41.89
AG quadrature ($m = 5$)	37.42	40.25	15.84	-20.35	-43.03
Cross-entropy ($N = 1000$)	40.12	30.98	21.35	-9.52	-58.81
Cross-entropy ($N = 2000$)	40.86	31.19	22.11	-8.94	-58.34
Cross-entropy ($N = 5000$)	40.90	31.31	22.01	-8.75	-58.11

Table SM.43: Bias for all the parameters of model (1) obtained by means of Monte Carlo simulations for various estimation techniques and various values of σ and for $G = 50$ and $n = 250$. For each value of σ , the number of replications is equal to 200. All values of the bias are multiplied by 100.

	σ				
	0.01	0.10	0.20	0.50	1.00
RMSE of $\hat{\beta}_0$					
AG quadrature ($m = 1$)	14.44	13.54	13.72	15.39	14.62
AG quadrature ($m = 3$)	14.44	13.55	13.72	15.39	14.62
AG quadrature ($m = 5$)	14.44	13.55	13.72	15.39	14.62
Cross-entropy ($N = 1000$)	14.35	13.54	13.71	15.40	14.74
Cross-entropy ($N = 2000$)	14.41	13.46	13.80	15.44	14.66
Cross-entropy ($N = 5000$)	14.39	13.52	13.71	15.42	14.58
RMSE of $\hat{\beta}_1$					
AG quadrature ($m = 1$)	19.30	17.91	18.26	19.88	19.28
AG quadrature ($m = 3$)	19.30	17.91	18.26	19.88	19.28
AG quadrature ($m = 5$)	19.30	17.91	18.26	19.88	19.28
Cross-entropy ($N = 1000$)	19.25	17.89	18.46	19.88	19.34
Cross-entropy ($N = 2000$)	19.24	17.73	18.41	19.80	19.24
Cross-entropy ($N = 5000$)	19.37	17.81	18.29	19.77	19.17
RMSE of $\hat{\beta}_2$					
AG quadrature ($m = 1$)	27.70	24.24	24.26	25.28	26.25
AG quadrature ($m = 3$)	27.70	24.24	24.26	25.28	26.25
AG quadrature ($m = 5$)	27.70	24.24	24.26	25.28	26.25
Cross-entropy ($N = 1000$)	27.65	24.05	24.29	25.56	26.40
Cross-entropy ($N = 2000$)	27.62	24.14	24.49	25.42	26.29
Cross-entropy ($N = 5000$)	27.70	24.21	24.31	25.20	26.08
RMSE of $\hat{\beta}_3$					
AG quadrature ($m = 1$)	33.34	31.11	27.87	30.47	35.05
AG quadrature ($m = 3$)	33.34	31.11	27.87	30.47	35.05
AG quadrature ($m = 5$)	33.34	31.11	27.87	30.47	35.05
Cross-entropy ($N = 1000$)	33.25	31.04	28.22	30.46	35.13
Cross-entropy ($N = 2000$)	33.33	31.00	27.99	30.37	35.15
Cross-entropy ($N = 5000$)	33.40	30.98	27.97	30.17	34.82
RMSE of $\hat{\sigma}$					
AG quadrature ($m = 1$)	12.31	12.76	17.72	44.85	94.54
AG quadrature ($m = 3$)	12.32	12.77	17.72	44.85	94.54
AG quadrature ($m = 5$)	12.32	12.77	17.72	44.85	94.54
Cross-entropy ($N = 1000$)	18.49	11.34	5.52	31.61	81.69
Cross-entropy ($N = 2000$)	18.59	11.36	5.48	31.52	81.29
Cross-entropy ($N = 5000$)	18.75	11.32	5.40	31.45	81.35
Computing time					
AG quadrature ($m = 1$)	0.73	0.62	0.61	0.62	0.61
AG quadrature ($m = 3$)	0.84	0.81	0.79	0.79	0.79
AG quadrature ($m = 5$)	0.93	0.91	0.88	0.89	0.89
Cross-entropy ($N = 1000$)	31.42	30.93	29.69	29.94	30.28
Cross-entropy ($N = 2000$)	60.48	59.52	59.11	59.44	59.52
Cross-entropy ($N = 5000$)	147.59	146.34	148.34	147.93	146.35

Table SM.44: Sample Root Mean Squared Error (RMSE) for all the parameters of model (1) and computing times obtained by means of Monte Carlo simulation for various estimation techniques and various values of σ and for $G = 8$ and $n = 1000$. For each value of σ , the number of replications is equal to 200. All the values of the RMSE are multiplied by 100; computing times are in seconds.

Bias of $\hat{\beta}_0$	σ				
	0.01	0.10	0.20	0.50	1.00
AG quadrature ($m = 1$)	0.78	0.50	0.50	0.36	0.12
AG quadrature ($m = 3$)	0.78	0.50	0.50	0.36	0.12
AG quadrature ($m = 5$)	0.78	0.50	0.50	0.36	0.12
Cross-entropy ($N = 1000$)	0.84	0.47	0.63	0.30	0.11
Cross-entropy ($N = 2000$)	0.68	0.40	0.54	0.31	0.32
Cross-entropy ($N = 5000$)	0.74	0.43	0.52	0.27	0.19
Bias of $\hat{\beta}_1$					
AG quadrature ($m = 1$)	3.92	3.17	1.36	2.57	3.13
AG quadrature ($m = 3$)	3.92	3.17	1.36	2.57	3.13
AG quadrature ($m = 5$)	3.92	3.17	1.36	2.57	3.13
Cross-entropy ($N = 1000$)	3.92	2.97	1.38	2.61	3.29
Cross-entropy ($N = 2000$)	3.90	2.80	1.46	2.51	3.28
Cross-entropy ($N = 5000$)	4.04	3.01	1.41	2.48	3.27
Bias of $\hat{\beta}_2$					
AG quadrature ($m = 1$)	5.15	5.21	2.81	3.06	4.46
AG quadrature ($m = 3$)	5.15	5.21	2.81	3.06	4.46
AG quadrature ($m = 5$)	5.15	5.21	2.81	3.06	4.46
Cross-entropy ($N = 1000$)	5.15	4.90	3.01	3.26	4.75
Cross-entropy ($N = 2000$)	5.26	4.95	3.04	3.12	4.74
Cross-entropy ($N = 5000$)	5.35	5.08	2.97	3.11	4.75
Bias of $\hat{\beta}_3$					
AG quadrature ($m = 1$)	4.97	7.35	3.47	2.09	8.12
AG quadrature ($m = 3$)	4.97	7.35	3.47	2.09	8.12
AG quadrature ($m = 5$)	4.97	7.35	3.47	2.09	8.12
Cross-entropy ($N = 1000$)	4.96	7.29	3.85	2.30	8.47
Cross-entropy ($N = 2000$)	5.12	7.02	3.81	2.22	8.53
Cross-entropy ($N = 5000$)	5.27	7.22	3.71	2.21	8.46
Bias of $\hat{\sigma}$					
AG quadrature ($m = 1$)	5.57	-1.54	-14.19	-43.51	-93.96
AG quadrature ($m = 3$)	5.58	-1.54	-14.19	-43.51	-93.96
AG quadrature ($m = 5$)	5.58	-1.54	-14.19	-43.51	-93.96
Cross-entropy ($N = 1000$)	17.71	9.82	-1.48	-31.17	-81.53
Cross-entropy ($N = 2000$)	17.79	9.81	-1.41	-31.10	-81.11
Cross-entropy ($N = 5000$)	17.96	9.88	-1.26	-31.02	-81.19

Table SM.45: Bias for all the parameters of model (1) obtained by means of Monte Carlo simulation for various estimation techniques and various values of σ and for $G = 8$ and $n = 1000$. For each value of σ , the number of replications is equal to 200. All the values of the bias are multiplied by 100.

RMSE of $\hat{\beta}_0$	σ				
	0.01	0.10	0.20	0.50	1.00
AG quadrature ($m = 1$)	14.47	15.07	13.29	13.96	15.00
AG quadrature ($m = 3$)	14.47	15.07	13.29	13.96	15.00
AG quadrature ($m = 5$)	14.47	15.07	13.29	13.96	15.00
Cross-entropy ($N = 1000$)	14.57	15.13	13.48	13.94	15.05
Cross-entropy ($N = 2000$)	14.38	15.07	13.23	13.91	14.89
Cross-entropy ($N = 5000$)	14.35	15.06	13.21	14.01	15.00
RMSE of $\hat{\beta}_1$					
AG quadrature ($m = 1$)	19.46	18.79	19.16	20.47	19.40
AG quadrature ($m = 3$)	19.46	18.79	19.16	20.47	19.40
AG quadrature ($m = 5$)	19.46	18.79	19.16	20.47	19.40
Cross-entropy ($N = 1000$)	19.46	18.94	19.33	20.54	19.51
Cross-entropy ($N = 2000$)	19.43	18.97	19.19	20.55	19.42
Cross-entropy ($N = 5000$)	19.34	18.73	19.15	20.55	19.39
RMSE of $\hat{\beta}_2$					
AG quadrature ($m = 1$)	25.81	25.52	24.21	23.90	24.29
AG quadrature ($m = 3$)	25.82	25.52	24.21	23.90	24.29
AG quadrature ($m = 5$)	25.82	25.52	24.21	23.90	24.29
Cross-entropy ($N = 1000$)	25.51	25.63	24.27	24.13	24.19
Cross-entropy ($N = 2000$)	25.43	25.45	23.93	24.07	24.28
Cross-entropy ($N = 5000$)	25.50	25.29	24.00	24.02	24.26
RMSE of $\hat{\beta}_3$					
AG quadrature ($m = 1$)	32.97	31.79	30.92	33.60	30.37
AG quadrature ($m = 3$)	32.97	31.79	30.92	33.60	30.37
AG quadrature ($m = 5$)	32.97	31.79	30.92	33.60	30.37
Cross-entropy ($N = 1000$)	32.63	31.81	30.98	33.91	30.72
Cross-entropy ($N = 2000$)	32.45	31.75	30.68	33.64	30.63
Cross-entropy ($N = 5000$)	32.48	31.56	30.83	33.66	30.52
RMSE of $\hat{\sigma}$					
AG quadrature ($m = 1$)	14.74	12.59	17.58	44.57	93.00
AG quadrature ($m = 3$)	14.76	12.60	17.58	44.57	93.00
AG quadrature ($m = 5$)	14.76	12.60	17.58	44.57	93.00
Cross-entropy ($N = 1000$)	19.29	10.67	4.41	31.54	80.91
Cross-entropy ($N = 2000$)	19.34	10.56	4.30	31.54	80.81
Cross-entropy ($N = 5000$)	19.26	10.73	4.24	31.43	80.72
Computing time					
AG quadrature ($m = 1$)	0.64	0.63	0.64	0.63	0.64
AG quadrature ($m = 3$)	0.81	0.80	0.81	0.81	0.81
AG quadrature ($m = 5$)	0.92	0.91	0.92	0.90	0.91
Cross-entropy ($N = 1000$)	31.70	31.00	31.45	31.19	32.14
Cross-entropy ($N = 2000$)	61.13	61.23	60.50	62.37	62.01
Cross-entropy ($N = 5000$)	153.70	153.65	152.07	153.65	155.82

Table SM.46: Sample Root Mean Squared Error (RMSE) for all the parameters of model (1) and computing times obtained by means of Monte Carlo simulation for various estimation techniques and various values of σ and for $G = 12$ and $n = 1000$. For each value of σ , the number of replications is equal to 200. All the values of the RMSE are multiplied by 100; computing times are in seconds.

Bias of $\hat{\beta}_0$	σ				
	0.01	0.10	0.20	0.50	1.00
AG quadrature ($m = 1$)	1.97	0.86	0.75	1.40	0.78
AG quadrature ($m = 3$)	1.97	0.86	0.75	1.40	0.78
AG quadrature ($m = 5$)	1.97	0.86	0.75	1.40	0.78
Cross-entropy ($N = 1000$)	1.86	0.85	0.93	1.40	0.66
Cross-entropy ($N = 2000$)	1.82	0.81	0.73	1.36	0.75
Cross-entropy ($N = 5000$)	1.85	0.80	0.66	1.36	0.67
Bias of $\hat{\beta}_1$					
AG quadrature ($m = 1$)	1.72	2.55	2.31	2.54	2.76
AG quadrature ($m = 3$)	1.72	2.55	2.31	2.54	2.76
AG quadrature ($m = 5$)	1.72	2.55	2.31	2.54	2.76
Cross-entropy ($N = 1000$)	1.63	2.50	2.48	2.64	2.63
Cross-entropy ($N = 2000$)	1.44	2.50	2.37	2.48	2.72
Cross-entropy ($N = 5000$)	1.46	2.44	2.29	2.59	2.61
Bias of $\hat{\beta}_2$					
AG quadrature ($m = 1$)	4.23	2.28	3.38	4.92	4.15
AG quadrature ($m = 3$)	4.23	2.28	3.38	4.92	4.15
AG quadrature ($m = 5$)	4.23	2.28	3.38	4.92	4.15
Cross-entropy ($N = 1000$)	4.20	2.32	3.60	5.13	4.14
Cross-entropy ($N = 2000$)	3.99	2.19	3.53	4.86	4.20
Cross-entropy ($N = 5000$)	4.00	2.15	3.43	4.98	3.98
Bias of $\hat{\beta}_3$					
AG quadrature ($m = 1$)	4.55	3.77	6.20	7.40	5.40
AG quadrature ($m = 3$)	4.56	3.77	6.20	7.40	5.40
AG quadrature ($m = 5$)	4.56	3.77	6.20	7.40	5.40
Cross-entropy ($N = 1000$)	4.51	3.89	6.55	7.68	5.43
Cross-entropy ($N = 2000$)	4.28	3.77	6.33	7.36	5.59
Cross-entropy ($N = 5000$)	4.26	3.63	6.30	7.54	5.23
Bias of $\hat{\sigma}$					
AG quadrature ($m = 1$)	7.29	-1.56	-13.28	-43.21	-92.20
AG quadrature ($m = 3$)	7.30	-1.55	-13.27	-43.20	-92.19
AG quadrature ($m = 5$)	7.30	-1.55	-13.27	-43.20	-92.19
Cross-entropy ($N = 1000$)	18.77	9.72	-1.14	-31.24	-80.79
Cross-entropy ($N = 2000$)	18.80	9.54	-0.97	-31.23	-80.68
Cross-entropy ($N = 5000$)	18.77	9.76	-0.99	-31.12	-80.60

Table SM.47: Bias for all the parameters of model (1) obtained by means of Monte Carlo simulation for various estimation techniques and various values of σ and for $G = 12$ and $n = 1000$. For each value of σ , the number of replications is equal to 200. All the values of the bias are multiplied by 100.

	σ				
	0.01	0.10	0.20	0.50	1.00
RMSE of $\hat{\beta}_0$					
AG quadrature ($m = 1$)	13.63	14.98	14.96	14.61	14.06
AG quadrature ($m = 3$)	13.63	14.98	14.96	14.61	14.06
AG quadrature ($m = 5$)	13.63	14.98	14.96	14.61	14.06
Cross-entropy ($N = 1000$)	13.65	15.05	14.88	14.58	14.11
Cross-entropy ($N = 2000$)	13.52	14.90	14.79	14.64	14.00
Cross-entropy ($N = 5000$)	13.58	15.02	14.86	14.57	14.02
RMSE of $\hat{\beta}_1$					
AG quadrature ($m = 1$)	18.31	22.00	19.77	21.44	18.56
AG quadrature ($m = 3$)	18.31	22.01	19.77	21.45	18.56
AG quadrature ($m = 5$)	18.31	22.01	19.77	21.45	18.56
Cross-entropy ($N = 1000$)	18.21	21.69	19.81	21.19	18.57
Cross-entropy ($N = 2000$)	18.20	21.62	19.61	20.95	18.38
Cross-entropy ($N = 5000$)	18.28	21.65	19.65	20.86	18.39
RMSE of $\hat{\beta}_2$					
AG quadrature ($m = 1$)	23.64	25.12	28.53	27.92	25.83
AG quadrature ($m = 3$)	23.65	25.13	28.53	27.93	25.83
AG quadrature ($m = 5$)	23.65	25.13	28.54	27.93	25.83
Cross-entropy ($N = 1000$)	23.42	24.90	28.16	27.67	25.49
Cross-entropy ($N = 2000$)	23.52	24.89	28.14	27.43	25.25
Cross-entropy ($N = 5000$)	23.43	24.82	28.23	27.32	25.33
RMSE of $\hat{\beta}_3$					
AG quadrature ($m = 1$)	29.56	32.52	34.94	34.80	34.00
AG quadrature ($m = 3$)	29.56	32.52	34.94	34.80	34.00
AG quadrature ($m = 5$)	29.56	32.52	34.94	34.81	34.00
Cross-entropy ($N = 1000$)	29.59	32.14	34.70	34.01	33.53
Cross-entropy ($N = 2000$)	29.42	31.95	34.39	33.88	33.45
Cross-entropy ($N = 5000$)	29.57	32.06	34.59	33.86	33.37
RMSE of $\hat{\sigma}$					
AG quadrature ($m = 1$)	17.33	14.20	18.71	41.63	89.21
AG quadrature ($m = 3$)	17.37	14.23	18.72	41.61	89.19
AG quadrature ($m = 5$)	17.37	14.23	18.72	41.61	89.19
Cross-entropy ($N = 1000$)	19.30	10.22	3.37	29.98	79.79
Cross-entropy ($N = 2000$)	19.20	10.11	3.24	29.83	79.66
Cross-entropy ($N = 5000$)	19.26	10.29	3.19	29.66	79.57
Computing time					
AG quadrature ($m = 1$)	0.67	0.66	0.67	0.67	0.68
AG quadrature ($m = 3$)	0.84	0.86	0.87	0.85	0.86
AG quadrature ($m = 5$)	0.96	0.96	0.97	0.97	0.98
Cross-entropy ($N = 1000$)	37.56	37.63	38.18	38.25	37.95
Cross-entropy ($N = 2000$)	73.76	74.02	74.62	75.40	74.79
Cross-entropy ($N = 5000$)	182.87	181.81	183.77	181.28	183.17

Table SM.48: Sample Root Mean Squared Error (RMSE) for all the parameters of model (1) obtained by means of Monte Carlo simulations for various estimation techniques and various values of σ and for $G = 25$ and $n = 1000$. For each value of σ , the number of replications is equal to 200. All values of the RMSE are multiplied by 100.

Bias of $\hat{\beta}_0$	σ				
	0.01	0.10	0.20	0.50	1.00
AG quadrature ($m = 1$)	0.79	1.43	2.89	3.68	1.70
AG quadrature ($m = 3$)	0.79	1.43	2.89	3.68	1.71
AG quadrature ($m = 5$)	0.79	1.43	2.89	3.68	1.71
Cross-entropy ($N = 1000$)	0.43	1.12	2.73	3.32	1.51
Cross-entropy ($N = 2000$)	0.60	1.23	2.57	3.41	1.37
Cross-entropy ($N = 5000$)	0.57	1.27	2.63	3.45	1.35
Bias of $\hat{\beta}_1$					
AG quadrature ($m = 1$)	2.09	4.09	4.16	6.17	3.68
AG quadrature ($m = 3$)	2.09	4.09	4.16	6.18	3.69
AG quadrature ($m = 5$)	2.09	4.09	4.16	6.18	3.69
Cross-entropy ($N = 1000$)	1.53	3.71	3.85	5.75	2.96
Cross-entropy ($N = 2000$)	1.57	3.81	3.70	5.50	3.10
Cross-entropy ($N = 5000$)	1.74	3.72	3.74	5.54	2.97
Bias of $\hat{\beta}_2$					
AG quadrature ($m = 1$)	2.61	5.10	6.59	9.62	6.71
AG quadrature ($m = 3$)	2.62	5.10	6.59	9.63	6.71
AG quadrature ($m = 5$)	2.62	5.10	6.59	9.63	6.72
Cross-entropy ($N = 1000$)	2.21	4.57	6.02	8.64	5.68
Cross-entropy ($N = 2000$)	2.13	4.68	5.96	8.64	5.80
Cross-entropy ($N = 5000$)	2.17	4.68	6.05	8.63	5.81
Bias of $\hat{\beta}_3$					
AG quadrature ($m = 1$)	3.80	7.50	9.97	12.24	9.19
AG quadrature ($m = 3$)	3.80	7.51	9.97	12.25	9.20
AG quadrature ($m = 5$)	3.80	7.51	9.98	12.25	9.20
Cross-entropy ($N = 1000$)	3.20	6.88	9.23	11.00	8.01
Cross-entropy ($N = 2000$)	3.15	6.85	9.18	11.00	8.13
Cross-entropy ($N = 5000$)	3.31	6.99	9.38	11.02	7.92
Bias of $\hat{\sigma}$					
AG quadrature ($m = 1$)	9.22	-1.02	-10.15	-38.31	-87.83
AG quadrature ($m = 3$)	9.24	-1.00	-10.13	-38.29	-87.80
AG quadrature ($m = 5$)	9.24	-1.00	-10.13	-38.28	-87.80
Cross-entropy ($N = 1000$)	19.05	9.68	0.00	-29.80	-79.73
Cross-entropy ($N = 2000$)	18.98	9.65	0.08	-29.67	-79.61
Cross-entropy ($N = 5000$)	19.07	9.87	0.17	-29.49	-79.51

Table SM.49: Bias for all the parameters of model (1) obtained by means of Monte Carlo simulation for various estimation techniques and various values of σ and for $G = 25$ and $n = 1000$. For each value of σ , the number of replications is equal to 200. All the values of the bias are multiplied by 100.

RMSE of $\hat{\beta}_0$	σ				
	0.01	0.10	0.20	0.50	1.00
AG quadrature ($m = 1$)	13.84	14.26	13.83	13.20	15.69
AG quadrature ($m = 3$)	13.85	14.27	13.83	13.21	15.70
AG quadrature ($m = 5$)	13.85	14.27	13.83	13.21	15.70
Cross-entropy ($N = 1000$)	13.60	13.95	13.75	13.16	15.37
Cross-entropy ($N = 2000$)	13.52	13.77	13.64	13.05	15.42
Cross-entropy ($N = 5000$)	13.62	13.84	13.56	13.00	15.39
RMSE of $\hat{\beta}_1$					
AG quadrature ($m = 1$)	20.56	21.40	19.85	20.23	20.30
AG quadrature ($m = 3$)	20.57	21.41	19.86	20.23	20.31
AG quadrature ($m = 5$)	20.57	21.42	19.86	20.24	20.31
Cross-entropy ($N = 1000$)	19.61	20.68	19.02	19.49	19.88
Cross-entropy ($N = 2000$)	19.78	20.60	19.01	19.23	19.85
Cross-entropy ($N = 5000$)	19.72	20.49	19.07	19.24	19.85
RMSE of $\hat{\beta}_2$					
AG quadrature ($m = 1$)	25.09	26.65	24.69	25.18	27.61
AG quadrature ($m = 3$)	25.10	26.67	24.71	25.19	27.62
AG quadrature ($m = 5$)	25.10	26.67	24.71	25.19	27.62
Cross-entropy ($N = 1000$)	23.19	25.27	23.72	24.48	27.39
Cross-entropy ($N = 2000$)	23.63	25.39	23.70	24.41	27.01
Cross-entropy ($N = 5000$)	23.55	25.51	23.75	24.41	26.98
RMSE of $\hat{\beta}_3$					
AG quadrature ($m = 1$)	32.33	35.02	32.60	31.61	32.65
AG quadrature ($m = 3$)	32.35	35.05	32.62	31.63	32.67
AG quadrature ($m = 5$)	32.35	35.06	32.62	31.64	32.67
Cross-entropy ($N = 1000$)	30.59	33.18	31.08	29.85	32.05
Cross-entropy ($N = 2000$)	30.63	33.37	31.23	29.78	31.91
Cross-entropy ($N = 5000$)	30.75	33.43	31.33	29.88	31.91
RMSE of $\hat{\sigma}$					
AG quadrature ($m = 1$)	21.85	18.72	18.68	42.38	89.51
AG quadrature ($m = 3$)	21.94	18.81	18.73	42.36	89.47
AG quadrature ($m = 5$)	21.95	18.83	18.74	42.36	89.47
Cross-entropy ($N = 1000$)	19.03	10.43	2.40	30.21	79.75
Cross-entropy ($N = 2000$)	19.33	10.55	2.32	30.10	79.84
Cross-entropy ($N = 5000$)	19.43	10.54	2.24	29.99	79.65
Computing time					
AG quadrature ($m = 1$)	0.71	0.72	0.71	0.71	0.71
AG quadrature ($m = 3$)	0.91	0.90	0.90	0.90	0.89
AG quadrature ($m = 5$)	1.02	1.01	1.03	1.01	1.02
Cross-entropy ($N = 1000$)	49.01	45.23	45.13	45.38	44.43
Cross-entropy ($N = 2000$)	97.30	88.74	88.18	88.49	88.56
Cross-entropy ($N = 5000$)	224.44	221.45	222.62	219.23	222.36

Table SM.50: Sample Root Mean Squared Error (RMSE) for all the parameters of model (1) obtained by means of Monte Carlo simulations for various estimation techniques and various values of σ and for $G = 50$ and $n = 1000$. For each value of σ , the number of replications is equal to 200. All values of the RMSE are multiplied by 100.

Bias of $\hat{\beta}_0$	σ				
	0.01	0.10	0.20	0.50	1.00
AG quadrature ($m = 1$)	2.07	2.64	2.48	1.97	1.37
AG quadrature ($m = 3$)	2.08	2.65	2.49	1.98	1.37
AG quadrature ($m = 5$)	2.08	2.65	2.49	1.98	1.37
Cross-entropy ($N = 1000$)	1.44	2.08	1.74	1.51	0.89
Cross-entropy ($N = 2000$)	1.53	2.00	1.90	1.37	0.86
Cross-entropy ($N = 5000$)	1.59	2.12	1.94	1.41	0.90
Bias of $\hat{\beta}_1$					
AG quadrature ($m = 1$)	1.71	5.38	5.42	3.92	4.82
AG quadrature ($m = 3$)	1.72	5.39	5.43	3.93	4.83
AG quadrature ($m = 5$)	1.72	5.39	5.43	3.93	4.83
Cross-entropy ($N = 1000$)	0.65	4.35	4.44	2.93	3.97
Cross-entropy ($N = 2000$)	0.83	4.39	4.25	3.04	3.77
Cross-entropy ($N = 5000$)	0.81	4.32	4.33	2.82	3.73
Bias of $\hat{\beta}_2$					
AG quadrature ($m = 1$)	4.88	7.28	4.67	5.14	7.12
AG quadrature ($m = 3$)	4.89	7.30	4.69	5.15	7.13
AG quadrature ($m = 5$)	4.90	7.30	4.69	5.16	7.14
Cross-entropy ($N = 1000$)	3.13	5.62	3.10	3.61	6.00
Cross-entropy ($N = 2000$)	3.39	5.60	3.15	3.67	5.88
Cross-entropy ($N = 5000$)	3.47	5.85	3.18	3.63	5.68
Bias of $\hat{\beta}_3$					
AG quadrature ($m = 1$)	6.74	10.00	8.80	5.57	8.17
AG quadrature ($m = 3$)	6.76	10.03	8.83	5.59	8.19
AG quadrature ($m = 5$)	6.77	10.04	8.83	5.60	8.19
Cross-entropy ($N = 1000$)	4.68	7.99	6.81	3.65	6.60
Cross-entropy ($N = 2000$)	4.81	8.17	6.79	3.73	6.50
Cross-entropy ($N = 5000$)	5.04	8.07	6.96	3.70	6.42
Bias of $\hat{\sigma}$					
AG quadrature ($m = 1$)	11.60	2.95	-6.02	-38.15	-87.70
AG quadrature ($m = 3$)	11.65	3.02	-5.95	-38.10	-87.64
AG quadrature ($m = 5$)	11.66	3.03	-5.94	-38.09	-87.63
Cross-entropy ($N = 1000$)	18.87	10.10	0.21	-30.10	-79.71
Cross-entropy ($N = 2000$)	19.16	10.28	0.41	-30.00	-79.81
Cross-entropy ($N = 5000$)	19.29	10.28	0.47	-29.89	-79.63

Table SM.51: Bias for all the parameters of model (1) obtained by means of Monte Carlo simulations for various estimation techniques and various values of σ and for $G = 50$ and $n = 1000$. For each value of σ , the number of replications is equal to 200. All values of the bias are multiplied by 100.

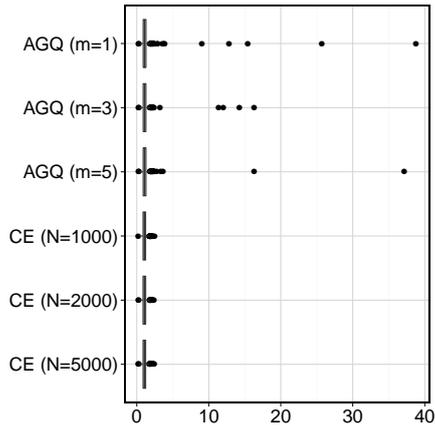


Figure SM.1: Boxplots of $\hat{\beta}_1$ for models with $G = 50$ and $\sigma = 1$ according to the estimation method.

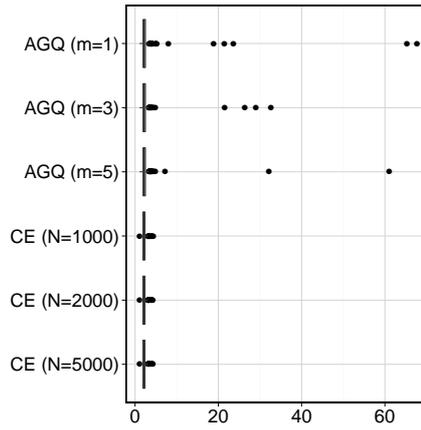


Figure SM.2: Boxplots of $\hat{\beta}_2$ for models with $G = 50$ and $\sigma = 1$ according to the estimation method.

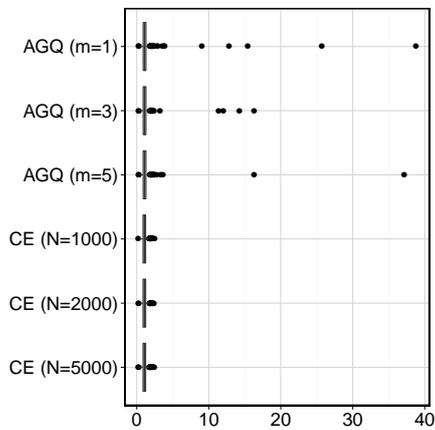


Figure SM.3: Boxplots of $\hat{\beta}_3$ for models with $G = 50$ and $\sigma = 1$ according to the estimation method.

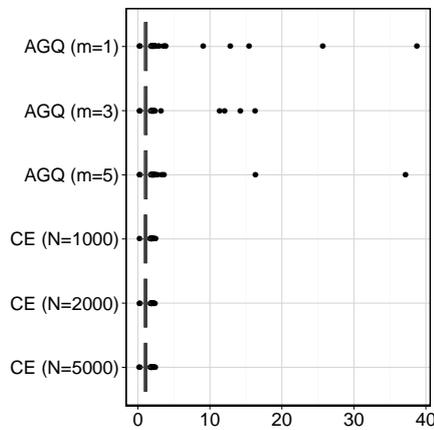


Figure SM.4: Boxplots of $\hat{\beta}_4$ for models with $G = 50$ and $\sigma = 1$ according to the estimation method.

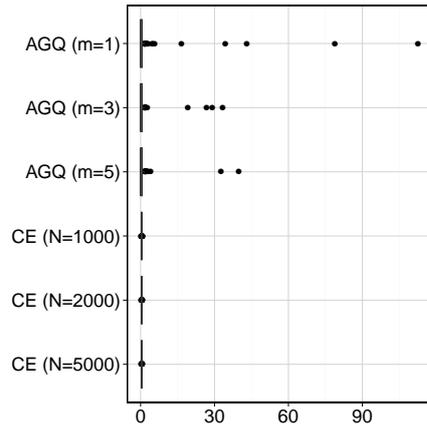


Figure SM.5: Boxplots of $\hat{\sigma}$ for models with $G = 50$ and $\sigma = 1$ according to the estimation method.

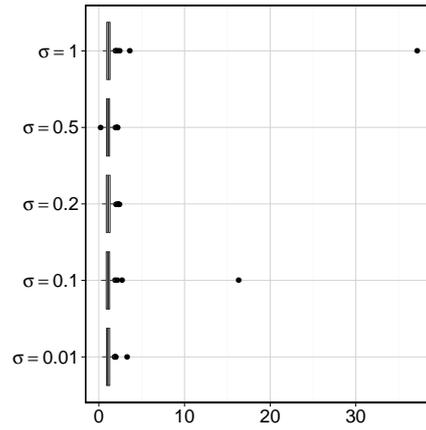


Figure SM.6: Boxplots of $\hat{\beta}_1$ for models with $G = 50$ and various values of σ , estimated by means of AGQ with $m = 5$.

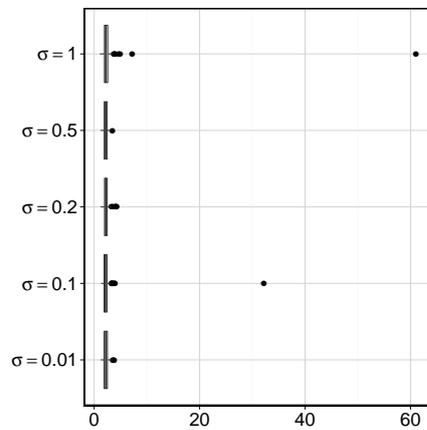


Figure SM.7: Boxplots of $\hat{\beta}_2$ for models with $G = 50$ and various values of σ , estimated by means of AGQ with $m = 5$.

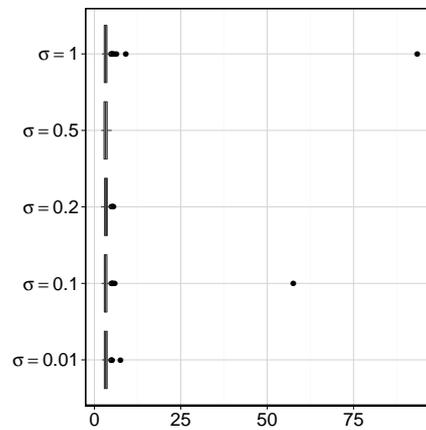


Figure SM.8: Boxplots of $\hat{\beta}_3$ for models with $G = 50$ and various values of σ , estimated by means of AGQ with $m = 5$.

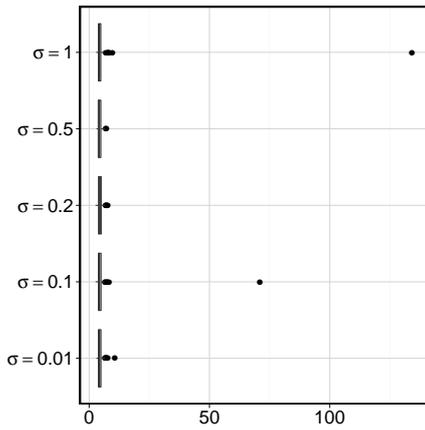


Figure SM.9: Boxplots of $\hat{\beta}_4$ for models with $G = 50$ and various values of σ , estimated by means of AGQ with $m = 5$.

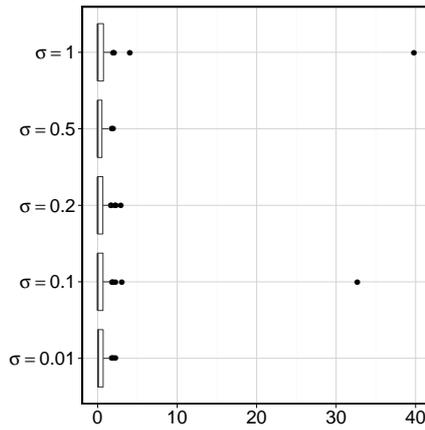


Figure SM.10: Boxplots of $\hat{\sigma}$ for models with $G = 50$ and various values of σ , estimated by means of AGQ with $m = 5$.

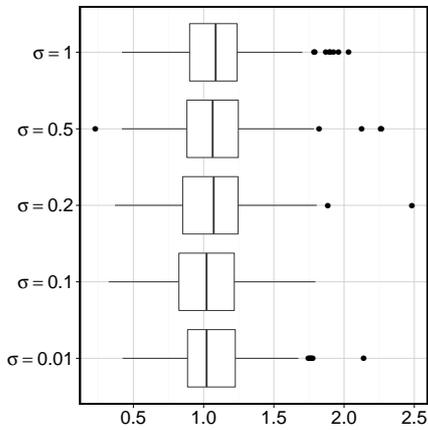


Figure SM.11: Boxplots of $\hat{\beta}_1$ for models with $G = 50$ and various values of σ , estimated by means of the cross-entropy algorithm with $N = 1000$.

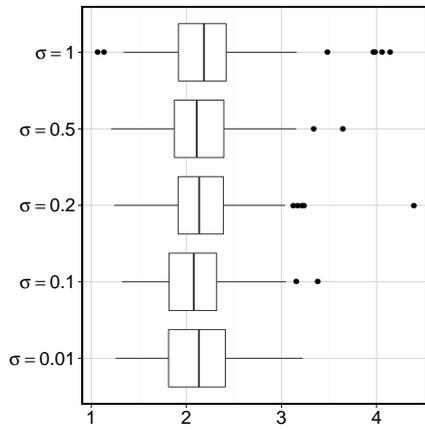


Figure SM.12: Boxplots of $\hat{\beta}_2$ for models with $G = 50$ and various values of σ , estimated by means of the cross-entropy algorithm with $N = 1000$.

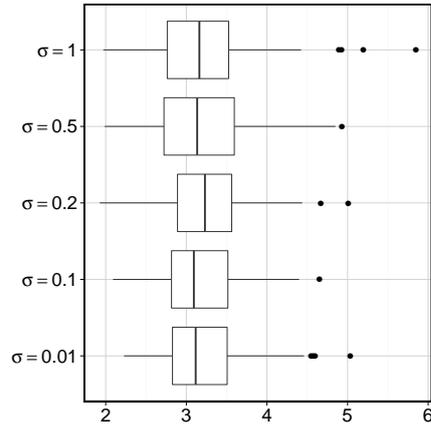


Figure SM.13: Boxplots of $\hat{\beta}_3$ for models with $G = 50$ and various values of σ , estimated by means of the cross-entropy algorithm with $N = 1000$.

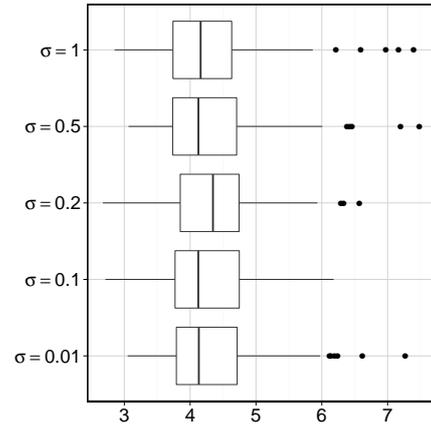


Figure SM.14: Boxplots of $\hat{\beta}_4$ for models with $G = 50$ and various values of σ , estimated by means of the cross-entropy algorithm with $N = 1000$.

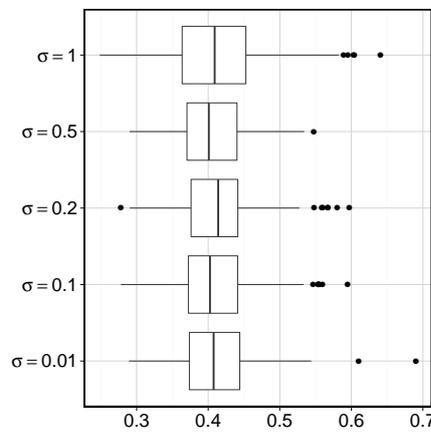


Figure SM.15: Boxplots of $\hat{\delta}$ for models with $G = 50$ and various values of σ , estimated by means of the cross-entropy algorithm with $N = 1000$.