Efficient preconditioned stochastic gradient descent for estimation in latent variable models

Charlotte Baey [*1] **Maud Delattre** [*2] **Estelle Kuhn** [*2] **Jean-Benoist Leger** [*3] **Sarah Lemler** [*4]

Abstract

Latent variable models are powerful tools for modeling complex phenomena involving in particular partially observed data, unobserved variables or underlying complex unknown structures. Inference is often difficult due to the latent structure of the model. To deal with parameter estimation in the presence of latent variables, well-known efficient methods exist, such as gradient-based and EM-type algorithms, but with practical and theoretical limitations. In this paper, we propose as an alternative for parameter estimation an efficient preconditioned stochastic gradient algorithm. Our method includes a preconditioning step based on a positive definite Fisher information matrix estimate. We prove convergence results for the proposed algorithm under mild assumptions for very general latent variables models. We illustrate through relevant simulations the performance of the proposed methodology in a nonlinear mixed effects model and in a stochastic block model.

1. Introduction

Latent variable models are widely used in many fields to describe complex phenomena whose mechanisms are indirectly observed and whose consideration in the model requires the use of unobserved variables. One can mention, for instance, mixture models (McLachlan & Basford, 1988) or stochastic block models (Abbe, 2018; Lee & Wilkinson, 2019) that are respectively used to describe the existence of an unknown group structure in a population and in an interaction network, or mixed-effects models whose latent structure is intended to describe some inter-individual variability (Pinheiro & Bates, 2000b; Lavielle, 2014). While bringing a certain level of detail to the modeling, the use of latent variables leads to more complex inference in general, as the observed likelihood often does not have an explicit form. This has led to the development of specific numerical methods for parameter estimation in latent variable models. Among the most common approaches, one can find the EM algorithm (Dempster et al., 1977) and its variants such as the Stochastic Approximation EM (SAEM) algorithm (Delyon et al., 1999) or even the Variational EM (VEM) algorithm (Bernardo et al., 2003). There are also gradient-based methods (see *e.g* chapters 10-11 in Cappé et al. (2005)), in particular some stochastic versions of the gradient descent algorithm have been specifically developed for latent variable models (Gu & Kong, 1998; Cai, 2010; Fang & Li, 2021).

EM-type algorithms are popular for their ease of implementation in curved exponential models where only operations on the sufficient statistics of the model are required at each iteration. These algorithms can still be applied in more general latent variable models but the methodology is not generic and requires new developments for each new model considered. To face this restrictive assumption, Debavelaere & Allassonnière (2021) suggested to use the exponentialization trick which consists in performing inference in an extended model belonging to the curved exponential family instead of in the initial model. However this approach has limitations in practice, mainly due to difficult algorithmic settings and tuning. Besides, the theoretical properties of EM-type algorithms have been the subject of many contributions. To our knowledge, the existing convergence theorems however all assume that the model belongs to the curved exponential family and none of them brings any guarantee beyond this framework (see Wu (1983) for EM and Delyon et al. (1999) for SAEM).

Gradient-based methods are an attractive alternative when EM-type algorithms are not easy to implement since their main ingredient is the log-likelihood gradient which is easily available in any latent variable model or at least approximated by combining the Fisher's identity (see Cappé et al.

[*]Equal contribution [1]Univ. Lille, CNRS, UMR 8524 - Laboratoire Paul Painlevé, F-59000 Lille, France [2]Université Paris-Saclay, INRAE, MaIAGE, 78350, Jouy-en-Josas, France. [3]Université de technologie de Compiègne, CNRS, Heudiasyc, Compiègne, France [4]Université Paris-Saclay, CentraleSupélec, Mathématiques et Informatique pour la Complexité et les Systèmes, 91190, Gif-sur-Yvette, France.. Correspondence to: Charlotte Baey <charlotte.baey@univ-lille.fr>, Estelle Kuhn <estelle.kuhn@inrae.fr>.

Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

(2005)) with Monte-Carlo methods. Among these, we find in particular the algorithms of Cai (2010) and Fang & Li (2021). The former has no theoretical guarantee and the latter, although supported by a convergence theorem, applies only to models that do not contain parameters to be estimated in the distribution of the latent variables. In recent stochastic gradient algorithms, some attention is also given to variance control for the full gradient estimation when it is obtained by processing mini-batches of the original sample (see *e.g.* (Johnson & Zhang, 2013; Reddi et al., 2016; Fang et al., 2018)). For all the above mentioned algorithms, although often supported by theoretical convergence guarantees, the practical performance is strongly affected by the choice of the learning rate which is a difficult setting. To overcome this difficulty and speed up the convergence, some people propose to precondition the gradient, but as suggested in Li (2017), one also has to be careful with the definition of the preconditioner so as not to degrade the behavior of the algorithm.

In this paper, our contribution consists in presenting a new stochastic gradient algorithm for maximum likelihood parameter estimation in general latent variable models. The proposed algorithm distinguishes itself from other stochastic gradient-based algorithms by including an easily available and structurally positive definite preconditioner based on Delattre & Kuhn (2023), which is also an estimate of the Fisher information matrix (FIM) in independent data models. As the FIM corresponds to the Hessian matrix of the objective function, it is therefore a natural choice regarding the second-order approximation (see Li, 2017). In addition, as the FIM estimate proposed by Delattre & Kuhn (2023) has the nice structural property of being symmetric positive definite, our preconditioning step allows to scale the different directions of the parameter space, homogenizing the evolution of the algorithm, and ensures that the search direction corresponds to a descent direction. Since the algorithm entails updating the estimate of the Fisher information matrix at each iteration, asymptotic confidence intervals can also be easily computed for all parameters as a by-product of the algorithm. Theoretical convergence results are provided that, unlike the existing results for competing EM-type algorithms, are also valid beyond the curved exponential family. The algorithm implementation is straightforward in practice, as it only requires the computation of the gradients of the log-densities of the latent variables and the gradients of the conditional log-densities of the observations given the latent, both of which being readily available quantities for the classical stochastic gradient descent. In addition, we propose a generic warming procedure which simplifies tuning and improves the algorithm efficiency in practice. Finally, it can also be easily extended to Bayesian maximum a posteriori estimation as well as to regularized estimation.

The paper is organized as follows. Section 2 introduces latent variable models. Section 3 presents our new Fisher-preconditioned stochastic gradient algorithm called Fisher-SGD and provides theoretical analysis. Some details on algorithmic settings are also given. Numerical results are presented in Section 4 that show the good performances of the algorithm. Some concluding remarks are given in Section 5.

2. Maximum Likelihood Estimation in Latent Variable Models

2.1. Description of Latent Variable Models

Let us consider observed random variables denoted by Y taking value in $\mathcal{Y}$ and latent random variables denoted by Z taking value in $\mathcal{Z}$ which are not observed. We assume that the couple (Y, Z) admits a parametric density f parameterized by θ taking value in $\Theta \subset \mathbb{R}^d$, where d is a non-zero positive integer. We denote by y and z realizations of the random variables Y and Z respectively. We denote by $p_\theta(\cdot \mid y)$ the density of the posterior distribution, i.e. the conditional distribution of Z given y.

Popular examples are mixture models, mixed effects models (Davidian & Giltinan, 1995), hidden Markov models (Cappé et al., 2005), stochastic block models Nowicki & Snijders (2001), or frailty models (Duchateau & Janssen, 2008).

Estimation of model parameters is not trivial in these models due to the presence of the latent structure and the unobserved variables Z. Namely one has to estimate θ only using the observed values of variables Y denoted by y.

2.2. Examples

In this section, we provide several examples of latent variable models, that will be used in the numerical experiments.

2.2.1. Mixed-effect Models

Mixed-effects models are commonly used when repeated data are available for each observational unit, e.g. in longitudinal studies or population models. They allow to account for both intra- and inter-individual variabilities through the use of fixed and random effects, the former being common to all the individuals while the latter vary from one individual to the other. These models can be described hierarchically, with a first layer giving the marginal distribution of the latent variables Z, and a second layer specifying the conditional distribution of the observations Y given the latent variables Z. More specifically, denoting by Y_{ij} the j-th observation of individual i, with $j = 1, \ldots, J$ and $i = 1, \ldots, N$, we consider the following model:

$$\begin{cases} Z_i \sim \mathcal{N}(\beta, \Gamma) \\ Y_{ij} = h(\alpha, Z_i, X_{ij}) + \varepsilon_{ij}, \quad \varepsilon_{ij} \sim \mathcal{N}(0, \sigma^2) \end{cases}$$

where α is a vector of fixed effects, X_{ij} is a set of known covariates, h is a nonlinear function representing the intra-individual variability and ε_{ij} is a random error term. The random effects are the latent variables Z_i, and are assumed to be mutually independent. The sequences (Z_i) and (ε_i), with $\varepsilon_i = (\varepsilon_{i1}, \ldots, \varepsilon_{iJ})^T$ are also assumed mutually independent. The parameters to be estimated are α, β, Γ and σ^2.

We consider now the specific case of the logistic growth curve model, which is commonly used in the nonlinear mixed-effect models community (see e.g. Pinheiro & Bates (2000a) and their famous example of orange trees growth), where function h is given by:

$$h(\alpha, Z_i, X_{ij}) = \frac{Z_{i1}}{1 + \exp\left(-\frac{X_{ij} - Z_{i2}}{\alpha}\right)}, \tag{1}$$

where $Z_i = (Z_{i1}, Z_{i2})^T \sim \mathcal{N}(\beta, \Gamma)$, with $\beta = (\beta_1, \beta_2)^T$ and Γ a 2×2 symmetric positive definite matrix. The model parameters are $\beta \in \mathbb{R}_+^* \times \mathbb{R}$, $\alpha \in \mathbb{R}_+^*$, $\Gamma \in \mathbb{S}_2^{++}$, where $\mathbb{S}_p^{++}$ is the set of symmetric, positive definite matrices of size $p \times p$ and $\sigma \in \mathbb{R}^+$.

Due to the presence of the fixed effect α, the joint density of (Y_i, Z_i) does not belong to the curved exponential family as defined in Delyon et al. (1999). Note that the implementation of stochastic versions of the EM algorithm are not trivial in such cases, involving complex terms evaluated by induction.

2.2.2. Stochastic Block Models

The Stochastic Block Model (SBM) is a common model in graph analysis, introduced by Holland et al. (1983) and Nowicki & Snijders (2001). For a directed graph of N nodes, the SBM assumes a latent unknown node classification (here Z), and assumes edge presences as independent and identically distributed conditionally on the cluster of nodes with a probability distribution which depends only on node clusters. We note Y the adjacency matrix of the graph, K the number of groups, Z_i the latent class indicator (one hot encoding) of node i. The SBM is formulated as follow:

$$\begin{cases} Z_i \sim \mathcal{M}(1; \alpha) & \text{with } \alpha \in {\mathbb{R}_+^*}^K,\ \sum_k \alpha_k = 1 \\ Y_{ij} | Z_{ik} Z_{jl} = 1 \sim \mathcal{B}(p_{kl}) & \text{with } \forall k, l,\ p_{kl} \in (0,1) \end{cases} \tag{2}$$

In this formulation, matrix Y is the observed variable and Z the latent variable. The parameters are $\alpha \in {\mathbb{R}_+^*}^K$ s.t. $\sum_k \alpha_k = 1$ and $p \in (0,1)^{K \times K}$.

Please note the non independence of $(Z_i)_{i:1 \leq i \leq N}$ conditionally on Y, thus the log-likelihood will be not splittable in terms involving only one Z_i each.

2.3. Maximum Likelihood Estimation in Latent Variable Models

We consider the marginal density of Y denoted by g and defined by

$$g(y; \theta) = \int_{\mathcal{Z}} f(y, z; \theta)\, dz$$

The maximum likelihood estimate (MLE) for parameter θ, denoted by $\hat{\theta}$, is defined as:

$$\hat{\theta} = \arg\max_{\theta} g(y; \theta)$$

This estimate is very popular in statistics since it has very nice asymptotic properties in a wide class of statistical models. Assuming mild conditions, as the number of observations goes to infinity, the MLE is consistent, asymptotically Gaussian and efficient (Van der Vaart, 2000). Therefore one can build asymptotic confidence intervals for model parameters as soon as an estimate of the Fisher information matrix is available.

However, computing this estimate in latent variable models through the maximization of the marginal likelihood g often requires numerical tools. Indeed the marginal density does not usually admit an explicit expression. Most popular tools are EM-like algorithms (Ng et al., 2012), used to maximize the density g with respect to θ.

Another way to compute the MLE $\hat{\theta}$ in latent variable models, often omitted but well discussed in Cappé et al. (2005), consists in searching the zeros of the derivative of $\log g$ using gradient-based methods. Indeed, assuming regularity conditions on f, we get the Fisher identity (Cappé et al., 2005) which states that for all $\theta \in \Theta$:

$$\nabla_\theta \log g(y; \theta) = \mathrm{E}\left(\nabla_\theta \log f(y, Z; \theta) \,|\, y; \theta\right) \tag{3}$$

Solving a function defined as an expectation can be done using stochastic gradient algorithms. Note however that the quantity θ is involved twice in the Fisher identity, namely in the derivative of the log-likelihood and in the posterior distribution of Z.

Therefore we will consider in the following section a stochastic gradient type algorithm to solve the Fisher identity (3).

3. Efficient Stochastic Gradient Algorithm with Preconditioning Step

In this section, we present our algorithm Fisher-SGD and some convergence results. We first present the algorithm in the case of independent observations and give in a second time a more general version for non-independent observations.

3.1. Description of the Algorithm in the Independent Case

One of the main ideas of our algorithm is to pre-condition, in the stochastic gradient descent at iteration k, by the positive definite estimate of the Fisher information $\widehat{I}(\theta)$ proposed by Delattre & Kuhn (2023) and detailed in Appendix B. The algorithm is described in Algorithm 1 (more details can be found in Algorithm 3 in the Appendix).

Algorithm 1 Fisher-SGD in the independent case

Input: $z_0, \theta_0, y_1, \ldots, y_N, r$
for $k = 1, \ldots, K$ **do**
  for $i = 1, \ldots, N$ **do**
    $z_i^k \sim q$ where q is either the posterior $p_{\theta_{k-1}}(\cdot \mid y_i)$ or a Markov kernel $\Pi_{\theta_{k-1}}(\cdot \mid \cdot, y_i)$
    Compute $\Delta_i^k = (1-\gamma_k)\Delta_i^{k-1} + \gamma_k \nabla_\theta \log f\left(y_i, z_i^k; \theta_{k-1}\right)$
  end for
  Compute $v_k = \frac{1}{N}\sum_{i=1}^N \nabla_\theta \log f\left(y_i, z_i^k; \theta_{k-1}\right)$
  Compute $I_k = \dfrac{1}{N}\sum_{i=1}^N \Delta_i^k \left(\Delta_i^k\right)^T$
  Set $\theta_k = \theta_{k-1} + \gamma_k I_k^{-1} v_k$
end for
Output: Estimated parameter θ_k, estimated Fisher information matrix of the whole sample NI_k, r-quasi-sample of latent variables in the posterior distribution $\left(z^{K-r+1} \cdots z^K\right)$.

Remark 3.1.

1. The proposed algorithm computes the MLE in latent variable models without requiring that the joint density belongs to the curved exponential family.
2. The proposed algorithm allows to compute asymptotic confidence intervals for free since the MLE and a Fisher matrix estimate are available as a by-product of the algorithm.
3. The proposed algorithm can be run either using exact simulation of the latent variables according to their conditional distribution given the observations whenever possible, or using the transition kernel of an ergodic Markov Chain having the posterior distribution as invariant distribution. Note that one common practical choice for such MCMC sampling scheme is the Metropolis-Hastings or the Metropolis-within-Gibbs algorithm (Robert et al., 1999).
4. The proposed algorithm can be extended to compute maximum a posteriori (MAP) estimate in Bayesian settings or regularized estimates, by adding to the criterion to be maximized a term corresponding to the prior distribution or the regularization term, and then considering an additional proximal step as in Atchadé et al. (2017).
5. The stochasticity of the stochastic gradient in our setting is due to the sampling of the latent random variable which allows to compute the target criterion defined as an expectation on the latent space.
6. In settings where the number of independent observations is large, usual minibatch techniques can be easily included in the proposed algorithm to deal with the high dimension of the observations (Schmidt et al. (2017)).

3.2. Theoretical Results

We now present two convergence results for the algorithm, depending on how the realizations of Z are generated at each iteration of the algorithm, either from the posterior distribution or from the transition kernel of a Markov Chain Monte Carlo algorithm. For the sake of simplicity in the next section we drop the index $1 \leq i \leq N$ in the notations. We first need some general regularity assumptions on the statistical model and on the sequence of step sizes (γ_k).

Assumption 3.2.

1. The joint density f is twice differentiable for $\theta \in \Theta$.
2. For all $y \in \mathcal{Y}$ the observed log-likelihood $\log g(y, \theta)$ is continuously differentiable on Θ and
$$\nabla_\theta g(y, \theta) = \int_{\mathcal{Z}} \nabla_\theta f(y, z, \theta) dz.$$
3. The sequence of step-sizes $(\gamma_k)_k$ satisfies for all $k \geq 0$, $0 \leq \gamma_k \leq 1$, $\sum_{k=1}^{+\infty} \gamma_k = +\infty$ and $\sum_{k=1}^{+\infty} \gamma_k^2 < +\infty$.

All these assumptions are classical for maximum likelihood estimation in latent variable models (Delyon et al., 1999; Kuhn & Lavielle, 2004).

We omit in the sequel the dependency of several quantities in y since it is considered as fixed. Let us define the objective function to minimize $F(\theta) = -\log g(y, \theta)$. Therefore solving the Fisher identity (3) is equivalent to solving $\nabla_\theta F(\theta) = 0$.

3.2.1. First Case: Simulating from the Posterior

To obtain the convergence of the algorithm in this context, we need additional assumptions.

Assumption 3.3.

1. The gradient of F is L-Lipschitz on Θ.
2. There exists $C > 0$ such that for all y and θ, $\mathrm{E}\left[\|\nabla_\theta \log f(y, Z; \theta)\|^2\right] \leq C$.
3. There exist $\mu_m > 0$ and $\mu_M > 0$ such that for all k, $\forall \mu \in \mathrm{Eig}(I_k)$, $\mu_m < \mu < \mu_M$, where $\mathrm{Eig}(A)$ denotes the set of eigenvalues (the spectrum) of matrix A.

The first two assumptions are classical when proving the convergence of a stochastic gradient algorithm. The third one is specific to our pre-conditioning using a positive definite estimate of the Fisher information. Note that in all regular models where the FIM is positive definite, this last assumption is satisfied for N and k large enough since the FIM estimate proposed by Delattre & Kuhn (2023) is convergent when the sample size N goes to infinity and the algorithm is convergent when the number of iterations k goes to infinity.

Theorem 3.4. *Under Assumptions 3.2 and 3.3, the iterates $(\theta_k)_k$ defined in Algorithm 1 with q equals to the posterior distribution $p_{\theta_{k-1}}$ at iteration k satisfy the convergence guarantee*

$$\mathrm{E}\left[\min_{0\leq l\leq k}\|\nabla_\theta F(\theta_l)\|^2\right] \leq \frac{2(F(\theta_0)-\min F)}{2\mu_m\sum_{l=0}^k\gamma_l} + \frac{\mu_M^2 CL\sum_{l=0}^k\gamma_l^2}{2\mu_m\sum_{l=0}^k\gamma_l}.$$

Proof: See Appendix B.

The control bound in Theorem 3.4 goes well to 0 when k goes to infinity and is similar to those obtained by, e.g., Bottou et al. (2018) or Ghadimi & Lan (2013) for the standard stochastic gradient algorithm.

3.2.2. Second Case: Simulating from a Kernel

In this context, we need additional assumptions on the Markov chain. Let us introduce the following notations: for a measurable function $V:\mathcal{Z}\to[1,+\infty)$, a measure μ on the σ-field of $\mathcal{Z}$ and a function $f:\mathcal{Z}\to\mathbb{R}$, we define

$$|f|_V=\sup_{z\in\mathcal{Z}}\frac{|f(z)|}{V(z)},\quad \|\mu\|_V=\sup_{f,|f|_V\leq 1}\left|\int f d\mu\right|.$$

Assumption 3.5.

1. For all $\theta\in\Theta$, the transition kernel Π_θ is a Markov kernel with invariant distribution the posterior p_θ.
2. There exist $0<\lambda<1$, $0<b$, $p\geq 2$ and a measurable function $W:\mathcal{Z}\to[1,+\infty)$ such that $\sup_{\theta\in\Theta}|\nabla_\theta\log f(\cdot,y;\theta)|_W<\infty$ and $\sup_{\theta\in\Theta}\Pi_\theta W^p\leq\lambda W^p+b$.
3. For any $0<\nu<p$, there exist a constant C and $0<\rho<1$ such that for any $z\in\mathcal{Z}$, $\sup_{\theta\in\Theta}\|\Pi_\theta^m(z,\cdot)-p_\theta\|_{W^\nu}\leq C\rho^m W^\nu(z)$.
4. There exists a constant C such that for any $(\theta,\theta')\in\Theta^2$,
$$|\nabla_\theta\log f(\cdot,y;\theta)-\nabla_\theta\log f(\cdot,y;\theta')|_W\leq C\|\theta-\theta'\|$$
$$|p_\theta-p_{\theta'}|_W\leq C\|\theta-\theta'\|$$
$$\sup_z\frac{\|\Pi_\theta(z,\cdot)-\Pi_{\theta'}(z,\cdot)\|}{W(z)}\leq C\|\theta-\theta'\|.$$

These assumptions are standard sufficient conditions in the literature to ensure the uniform ergodicity of the Markov chain with respect to θ and the existence of a solution to the Poisson equation associated to function $\nabla_\theta\log f(\cdot,y;\theta)$ (see for example Allassonniere & Kuhn (2015); Fort et al. (2011)). Concerning the logistic growth nonlinear mixed effects model defined in section 2.2.1 using (1), the function W can be chosen equal to $W(z)=1+\|z\|^2$.

Assumption 3.6. The sequence of step sizes (γ_k) satisfies $\sum|\gamma_{k+1}-\gamma_k|<\infty$.

This step size assumption is classical when controlling stochastic approximation and is satisfied in particular when the step size sequence is decreasing.

Theorem 3.7. *Under Assumptions 3.2, 3.3, 3.5, 3.6 and assuming that Θ is bounded, the iterates $(\theta_k)_k$ defined in Algorithm 1 with q equal to a transition kernel $\Pi_{\theta_{k-1}}$ satisfy the convergence guarantee*

$$\mathrm{E}\left[\min_{0\leq l\leq k}\|\nabla_\theta F(\theta_l)\|^2\right] \leq \frac{2(F(\theta_0)-\min F)}{2\mu_m\sum_{l=0}^k\gamma_l} + \frac{\mu_M^2 CL\sum_{l=0}^k\gamma_l^2+C}{2\mu_m\sum_{l=0}^k\gamma_l}.$$

Proof: See Appendix B.

This result extends that of Theorem 3.4 to the setting often used in practice when it is not possible to generate the latent variables from the posterior distribution. We obtain a similar bound as the one in Theorem 3.4 with an additional residual term that appears because of the MCMC procedure used to simulate z^k. The proof is postponed to the supplementary material and relies on technical tools involving Poisson equation solution for the control of Markov chain where the parameter evolves simultaneously.

3.3. Description of the Algorithm in the Non-independent Case

In the non-independent case, the algorithm should be adapted as the log-likelihood is not separable into terms involving only one Z_i. The proposed methodology uses the global criterion $f(y,z;\theta_k)$. The adapted algorithm is provided in Algorithm 2. There is no general formula to write the estimation of the Fisher information matrix in this case. See the adaptation for the SBM in Section 2.2.2 for an example.

Remark 3.8. Theorems 3.4 and 3.7 can be immediately extended to the non-independent setting.

3.4. Practical Implementation

The implementation of the algorithm should be done in practice with some precautions, in particular at the beginning of

Algorithm 2 Fisher-SGD in the non-independent case

Input: z_0, θ_0, y, r
for k=1,...,K **do**
  $z^k \sim q$ where q is either the posterior $p_{\theta_{k-1}}(\cdot \mid y)$ or a Markov kernel $\Pi_{\theta_{k-1}}(\cdot \mid \cdot, y)$
  Compute $v_k = \nabla_\theta \log f\big(y, z^k; \theta_{k-1}\big)$
  Compute I_k with a custom method.
  Set $\theta_k = \theta_{k-1} + \gamma_k I_k^{-1} v_k$
  if $k > K - r$ **then**
    Compute $\nabla_\theta^2 \log f\big(y, z^k; \theta_k\big)$
  end if
end for
Output: Estimated parameter: θ_k, estimated Fisher information matrix of the whole sample: $\frac{1}{r}\sum_{k=K-r+1}^{K} \nabla_\theta^2 \log f\big(y, z^k; \theta_k\big)$, sample of latent in the posterior distribution: $\big(z^{K-r+1} \cdots z^K\big)$.

the algorithm for the learning step γ_k and the calculation of the preconditioning matrix I_k.

3.4.1. Evolution of the Learning Step

The authors propose a strategy in three steps, similar to the warm-up strategies used for example by Loshchilov & Hutter (2017) and Smith & Topin (2019):

pre-heating: first, at the very beginning of the algorithm, the learning step is gradually increased following an exponential growth, starting from a very small value until reaching 1.

heating: then, the step is kept at 1 during a certain time, constituting the heating period.

decreasing steps: after stabilization, the steps are decreasing (such that $\sum_k \gamma_k = +\infty$ and $\sum_k \gamma_k^2 < +\infty$).

We thus obtain:

$$\gamma_k = \begin{cases} \gamma_0^{\left(1 - \frac{k}{K_{\text{pre-heating}}}\right)} & \text{if } k \leq K_{\text{pre-heating}} \\ 1 & \text{else if } k \leq K_{\text{heating}} \\ (k - K_{\text{heating}})^{-\alpha} & \text{else} \end{cases}$$

The authors propose to use $K_{\text{pre-heating}} = 1000$ and $\gamma_0 = 10^{-4}$. The proposed setting seems to be conservative for most situations, but $K_{\text{pre-heating}}$ and γ_0 can be respectively increased and decreased to improve stabilization of the pre-heating phase.

Concerning the choice of K_{heating}, the authors propose to use an adaptive method, averaging the norms of the gradients calculated with a third order filter of constant $\frac{1}{1000}$ and to stop the heating phase when the norm of the averaged gradient does not decrease anymore.

Concerning the choice of α, to ensure $\sum_k \gamma_k = +\infty$ and $\sum_k \gamma_k^2 < +\infty$, the authors propose to use $\alpha = 2/3$.

The complete algorithm using the preheating and the adaptive length heating is given in the Appendix in Algorithm 3.

3.4.2. Preconditioning Matrix

At the beginning of the algorithm, a bad preconditioning can lead to unstable behavior. Thus during the whole preheating period, instead of using I_k as defined in algorithm 1, we use a stabilized version. Let us introduce

$$\widehat{I}_k^* = \frac{1}{N}\sum_{i=1}^{N} \Delta_i^k \big(\Delta_i^k\big)^T$$

and define the new preconditioner by:

$$I_k = \begin{cases} (1-\gamma_k) r_k Id + \gamma_k \widehat{I}_k^* & \text{if } k < K_{\text{pre-heating}} \\ \widehat{I}_k^* & \text{otherwise.} \end{cases}$$

We can choose $r_k = 1$ or, to further avoid instabilities we can choose $r_k = \max\left(1, \operatorname{tr}\left(\widehat{I}_k^*\right)\right)$.

The complete algorithm using this stabilization is given in the Appendix in Algorithm 3.

3.4.3. Auto-differentiation and Parametrization

Nowadays, there is high interest in using automatic differentiation over analytical calculation of gradients. The authors propose to compute all gradients by means of automatic differentiation.

The parameters of the statistical models are rarely all parameters in $\mathbb{R}^d$, but the algorithm is presented in the framework of $\Theta = \mathbb{R}^d$. Using the functional invariance property of maximum likelihood, the authors propose to systematically reparametere to $\mathbb{R}^d$, through a bijective reparametrization. For obvious reasons of derivatives use, these reparametrizations must be diffeomorphisms and must be differentiable by automatic differentiation to be used transparently in the framework proposed here. The parametrization Cookbook (Leger, 2023) introduces a set of classical reparametrizations in statistics that verify these properties. The obtained maximum likelihood properties (and, in particular, the confidence intervals) can then be transferred to the initial space using a delta method (Van der Vaart, 2000).

4. Numerical Experiments

4.1. Logistic Growth Mixed-effects Model

We generated data according to the logistic growth model presented in Section 2.2, with $N = 1000$ individuals,

with the same vector of observation times for each individual, defined as a vector of $m = 20$ equally spaced values between 100 and 1500, and using the following parameter values: $\beta = (200, 500)^T, \Gamma = \text{diag}\,(40, 100)$, $\alpha = 150$ and $\sigma = 10$. Using a reparametrization, Algorithm 1 was run in the reparameterized space $\mathbb{R}^d$ with $d = 7$. The code is available in the Git repository `https://github.com/baeyc/fisher-sgd-nlme`.

To evaluate the performance and the robustness of our approach with respect to maximum likelihood estimate and confidence regions, we compared our algorithm to the competing MCMC-SAEM algorithm (Kuhn & Lavielle, 2004). This algorithm is, to the best of our knowledge, the only other one providing theoretical convergence guarantees towards the MLE when the model belongs to the curved exponential family. Since we introduced a fixed effect in the model, the considered model does no longer belong to the exponential family. Therefore, we used a specific implementation of the MCMC-SAEM algorithm (Comets et al., 2017) which rely on an exponentialization trick for models that do not belong to the curved exponential family, but is only usable for some specific nonlinear mixed-effects models. It is also noteworthy to mention that this algorithm relies on a block-diagonal estimate of the FIM, which has no particular reason to be block-diagonal in general.

The Fisher-SGD algorithm (we used the version given in Algorithm 1 using a Metropolis-within-Gibbs sampler) was run on M=1000 datasets generated using the same parameter values. The competing MCMC-SAEM algorithm was run on the same simulated datasets using the R package `saemix`.

Table 1 gives the root mean squared errors (RMSE) associated with each parameter and the empirical coverages computed as the proportion of the M datasets for which the true parameter value used to generate the data felt into the 95% confidence region. The confidence regions were built using automatic differentiation and the delta method (see Appendix C for details). Our approach performed better than the MCMC-SAEM algorithm, especially for variance components parameters, i.e. for Γ_{11}, Γ_{12} and Γ_{22}. Since the RMSE are similar with both algorithms, these results suggest that the FIM estimate obtained with Fisher-SGD is more accurate than the one obtained with the `saemix` package.

As an illustration, Figure 1 gives the evolution of one of the M trajectories, in the original parameter space. We can see that the algorithm reaches a neighborhood of the true value at the end of the pre-heating phase, stabilizes itself around this true value during the heating phase and reaches convergence during the last phase. Figure 3 in Appendix D.1 gives the evolution of the diagonal of the estimated FIM, along with the evolution of the learning step across the iterations.

Table 1. Root mean squared error (RMSE) and empirical coverage of confidence regions built at the nominal level of 0.95 using the FIM and the parameter estimates, for a total of $M = 1000$ repetitions. The first line corresponds to the vector of all parameters θ, and thus the coverage is associated to the confidence region in $\mathbb{R}^7$. The simulated values for the parameters are $\beta_1 = 200, \beta_2 = 500$, $\alpha = 150, \Gamma_{11} = 40, \Gamma_{12} = 0, \Gamma_{22} = 100$ and $\sigma^2 = 100$.

TYPE	FISHER-SGD		MCMC-SAEM	
	RMSE	COVERAGE	RMSE	COVERAGE
θ	15.13	0.952 ± 0.014	17.24	0.935 ± 0.015
β_1	0.234	0.942 ± 0.012	0.236	0.941 ± 0.015
β_2	0.586	0.958 ± 0.010	0.625	0.941 ± 0.015
α	0.414	0.972 ± 0.013	0.416	0.968 ± 0.011
Γ_{11}	2.221	0.951 ± 0.013	2.241	0.949 ± 0.014
Γ_{12}	4.156	0.948 ± 0.014	4.334	0.935 ± 0.015
Γ_{22}	14.324	0.948 ± 0.014	16.492	0.905 ± 0.018
σ^2	1.005	0.957 ± 0.012	1.010	0.951 ± 0.013

4.2. Stochastic Block Model

We generated 2000 simulated networks according to the Stochastic Block Model presented above in Section 2.2, with $K = 4$ groups and with $N = 100$ nodes or $N = 200$ nodes.

All networks are generated from the same set of parameters, we choose:

$$\alpha = (1/4, 1/4, 1/4, 1/4)$$
$$p = \begin{bmatrix} 2/3 & 2/3 & 1/3 & 2/3 \\ 2/3 & 2/3 & 2/3 & 1/3 \\ 1/3 & 2/3 & 2/3 & 2/3 \\ 2/3 & 1/3 & 2/3 & 2/3 \end{bmatrix}$$

As $\alpha^T p$ and $p\alpha$ are constant vectors, expected inner and outer degrees of node do not depend on clusters, therefore this simulation setting is not an easy case where naive algorithm can be applied, *e.g.* Channarond et al. (2012).

As described in Section 2.2, parameters are in a constrained space. To handle these constraints, a reparametrization is applied (see Appendix C for details). The complete algorithm used for estimation in Stochastic Block Model is a particular case of Algorithm 2 where the preconditionning matrix is computed with the same idea than for the independent case and the sampling is made with a Gibbs sampler. The complete algorithm is given in Algorithm 5 in the appendix. However, this algorithm is not sufficient to handle all the cases and for some initializations the sampling and the algorithm behavior leads to empty classes: in these cases the algorithm is restarted from the beginning with a new random initialization. The code is available in the Git repository `https://gitlab.com/jbleger/sbm_with_`

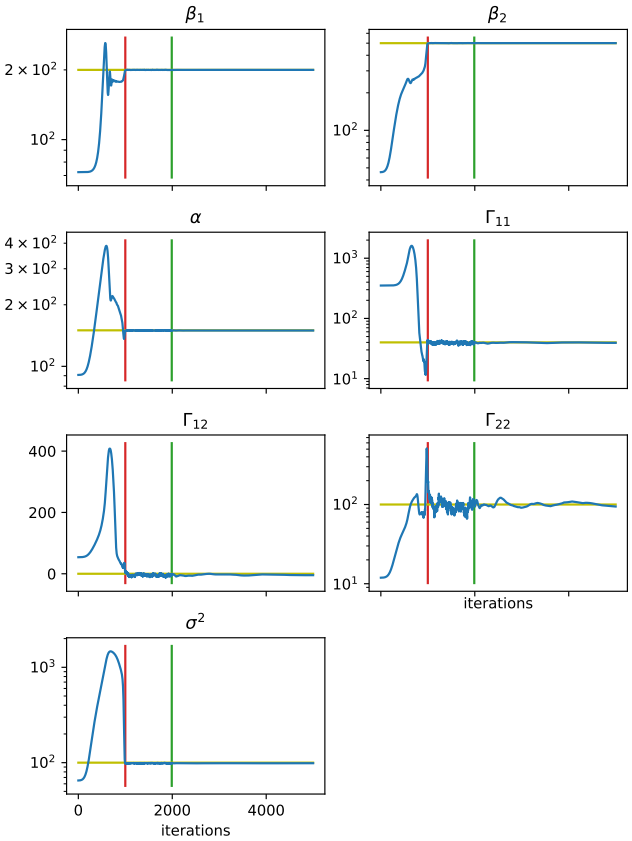

Figure 1. Evolution of the parameter estimates across the iterations in the logistic growth model. Yellow line: simulated value. The red line is the end of the pre-heating, and the green line is the end of the heating.

`fisher-sgd`.

As other model-based classification methods, SBM is subject to label-switching and the parameter is identifiable up to a permutation of classes. To compare the estimation to the simulated value, classes are permuted to maximize the congruence between the posterior of Z and the simulated value of Z.

The main advantage of our method is that we obtain an estimate of the FIM. As we have the asymptotic normality property (Bickel et al., 2013), we can compute asymptotic confidence interval of parameters. Then we illustrate our method by evaluating the estimation error with root mean squared error (RMSE) and by evaluating the coverage of 95% confidence interval obtained with the parameter estimate and the Fisher Information Matrix estimate. See Appendix C for details.

We chose not to compare ourselves to other methods, since to the authors' knowledge there is no method computing the MLE for not small SBM networks, disallowing computation of asymptotic confidence intervals.

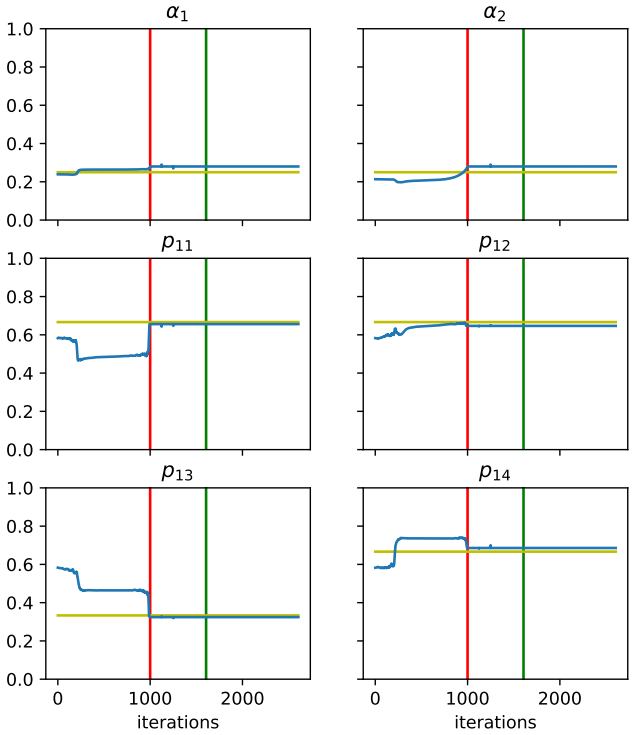

Figure 2. Evolution of the parameter estimates across the iterations with $N = 100$ in the stochastic block model. Yellow line: simulated value. The red line is the end of the pre-heating, and the green line is the end of the heating. Results for all parameters are given in appendix Figures 6 and 7.

In Figure 2, we show the evolution of parameter estimates (after transformation in the original parameter space). We see here the practical importance of the pre-heating phase: when latent variable are not acceptable, rapid evolution of parameters can lead to non convergent algorithm.

Results are presented for $N = 100$ for a subset of the original parameters in Table 2. Results for all parameters for $N = 100$ and $N = 200$ are given in the appendix in Table 3. We deduce that computed confidence ellipsoid for θ and confidence intervals for α and p are correct, which validate the MLE and Fisher Information Matrix estimation provided by our algorithm.

5. Conclusion

In this article, we consider parameter estimation in latent variable models. We propose an efficient stochastic gradient algorithm that includes a preconditioning step to scale the different directions of the parameter space, homogenizing the evolution of the algorithm. The preconditioner we use corresponds to a positive definite estimate of the Fisher

Table 2. Results of Fisher-SGD applied on SBM with 2000 replications. Root mean squared error (RMSE) and empirical coverage of confidence regions built at the nominal level of 0.95 using the FIM and the parameter estimates, for a total of $M = 2000$ repetitions. The first line corresponds to the vector of parameters θ, and thus the coverage is associated to the confidence region in $\mathbb{R}^{K^2+K-1}$. See Table 3 in Appendix for complete results with $N = 100$ and $N = 200$.

PARAMETER	SIMULATED	$N = 100$	
		RMSE	COVERAGE
θ		0.648	0.936 ± 0.011
α_1	0.250	0.044	0.943 ± 0.010
α_2	0.250	0.044	0.939 ± 0.010
$p_{1,1}$	0.667	0.023	0.940 ± 0.010
$p_{1,2}$	0.667	0.019	0.947 ± 0.010
$p_{1,3}$	0.333	0.022	0.948 ± 0.010
$p_{1,4}$	0.667	0.019	0.947 ± 0.010

information matrix in independent data models, which allows to get for free asymptotic confidence intervals for the parameters as a by-product of the algorithm. Theoretical convergence results are provided under mild assumptions for very general latent variables models, without assuming that the density belongs to the curved exponential density family. Using simulations, we show that our new algorithm performs satisfactorily and gives similar to better performances compare to competing methods. As we also propose a warming procedure, the method is generic enough to be easily implemented in very general latent variable models.

Funding

This work was funded by the Stat4Plant project ANR-20-CE45-0012.

Acknowledgment

The authors are grateful to Calimero for helpful discussion.

References

Abbe, E. Community detection and stochastic block models: Recent developments. *Journal of Machine Learning Research*, 18(177):1–86, 2018.

Allassonniere, S. and Kuhn, E. Convergent stochastic expectation maximization algorithm with efficient sampling in high dimension. application to deformable template model estimation. *Computational Statistics & Data Analysis*, 91:4–19, 2015.

Atchadé, Y. F., Fort, G., and Moulines, E. On perturbed proximal gradient algorithms. *The Journal of Machine Learning Research*, 18(1):310–342, 2017.

Bernardo, J., Bayarri, M., Berger, J., Dawid, A., Heckerman, D., Smith, A., West, M., et al. The variational bayesian em algorithm for incomplete data: with application to scoring graphical model structures. *Bayesian statistics*, 7 (453-464):210, 2003.

Bickel, P., Choi, D., Chang, X., and Zhang, H. Asymptotic normality of maximum likelihood and its variational approximation for stochastic blockmodels. *The Annals of Statistics*, 41(4):1922–1943, 2013.

Bottou, L., Curtis, F. E., and Nocedal, J. Optimization methods for large-scale machine learning. *SIAM Review*, 60(2):223–311, 2018.

Cai, L. Metropolis-hastings robbins-monro algorithm for confirmatory item factor analysis. *Journal of Educational and Behavioral Statistics*, 35(3):307–335, 2010.

Cappé, O., Moulines, E., and Rydén, T. *Inference in Hidden Markov Models*. Springer, 2005.

Channarond, A., Daudin, J.-J., and Robin, S. Classification and estimation in the stochastic blockmodel based on the empirical degrees. *Electronic Journal of Statistics*, 6: 2574–2601, 2012.

Comets, E., Lavenu, A., and Lavielle, M. Parameter estimation in nonlinear mixed effect models using saemix, an r implementation of the saem algorithm. *Journal of Statistical Software*, 80(3):1–41, 2017. doi: 10.18637/jss.v080.i03.

Davidian, M. and Giltinan, D. M. *Nonlinear models for repeated measurement data*. Chapman & Hall, 1995.

Debavelaere, V. and Allassonnière, S. On the curved exponential family in the stochastic approximation expectation maximization algorithm. *ESAIM: Probability & Statistics*, 25, 2021.

Delattre, M. and Kuhn, E. Estimating fisher information matrix in latent variable models based on the score function. *arXiv preprint arXiv:1909.06094v2*, 2023.

Delyon, B., Lavielle, M., and Moulines, E. Convergence of a stochastic approximation version of the em algorithm. *Annals of statistics*, pp. 94–128, 1999.

Dempster, A. P., Laird, N. M., and Rubin, D. B. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.

Duchateau, L. and Janssen, P. *The frailty model*. Springer, 2008.

Fang, C., Li, C. J., Lin, Z., and Zhang, T. Spider: Near-optimal non-convex optimization via stochastic path-integrated differential estimator. *Advances in Neural Information Processing Systems*, 31, 2018.

Fang, G. and Li, P. On estimation in latent variable models. In *International Conference on Machine Learning*, pp. 3100–3110. PMLR, 2021.

Fort, G., Moulines, E., and Priouret, P. Convergence of adaptive and interacting markov chain monte carlo algorithms. *The Annals of Statistics*, 39(6):3262–3289, 2011.

Ghadimi, S. and Lan, G. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization, ii: Shrinking procedures and optimal algorithms. *SIAM Journal on Optimization*, 23(4):2061–2089, 2013.

Goymann, W., Safari, I., Muck, C., and Schwabl, I. Sex roles, parental care and offspring growth in two contrasting coucal species. *Royal Society Open Science*, 3(10), 2016.

Gu, M. G. and Kong, F. H. A stochastic approximation algorithm with markov chain monte-carlo method for incomplete data estimation problems. *Proceedings of the National Academy of Sciences*, 95(13):7270–7274, 1998.

Hall, P. and Heyde, C. C. *Martingale limit theory and its application*. Academic press, 2014.

Holland, P. W., Laskey, K. B., and Leinhardt, S. Stochastic blockmodels: First steps. *Social networks*, 5(2):109–137, 1983.

Johnson, R. and Zhang, T. Accelerating stochastic gradient descent using predictive variance reduction. *Advances in neural information processing systems*, 26, 2013.

Kuhn, E. and Lavielle, M. Coupling a stochastic approximation version of em with an mcmc procedure. *ESAIM: Probability and Statistics*, 8:115–131, 2004.

Lavielle, M. *Mixed effects models for the population approach: models, tasks, methods and tools*. CRC Press, 2014.

Lee, C. and Wilkinson, D. A review of stochastic block models and extensions for graph clustering. *Appl Netw Sci*, 4, 2019.

Leger, J.-B. Parametrization cookbook: A set of bijective parametrizations for using machine learning methods in statistical inference, 2023.

Li, X.-L. Preconditioned stochastic gradient descent. *IEEE transactions on neural networks and learning systems*, 29 (5):1454–1466, 2017.

Loshchilov, I. and Hutter, F. Sgdr: Stochastic gradient descent with warm restarts. International Conference on Learning Representations (ICLR), 2017.

McLachlan, G. J. and Basford, K. E. *Mixture models: Inference and applications to clustering*, volume 38. M. Dekker New York, 1988.

Ng, S. K., Krishnan, T., and McLachlan, G. J. The em algorithm. In *Handbook of computational statistics*, pp. 139–172. Springer, 2012.

Nowicki, K. and Snijders, T. A. B. Estimation and prediction for stochastic blockstructures. *Journal of the American statistical association*, 96(455):1077–1087, 2001.

Pinheiro, J. and Bates, D. *Mixed-Effects Models in S and S-PLUS*. Springer-Verlag, 2000a.

Pinheiro, J. and Bates, D. *Mixed-Effects Models in S and S-PLUS*. Springer, 2000b.

Reddi, S. J., Hefny, A., Sra, S., Póczos, B., and Smola, A. Stochastic variance reduction for nonconvex optimization. In *International conference on machine learning*, pp. 314–323. PMLR, 2016.

Robert, C. P., Casella, G., and Casella, G. *Monte Carlo statistical methods*, volume 2. Springer, 1999.

Schmidt, M., Le Roux, N., and Bach, F. Minimizing finite sums with the stochastic average gradient. *Mathematical Programming*, 162:83–112, 2017.

Smith, L. N. and Topin, N. Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial intelligence and machine learning for multi-domain operations applications*, volume 11006, pp. 369–386. SPIE, 2019.

Van der Vaart, A. W. *Asymptotic statistics*, volume 3. Cambridge university press, 2000.

Wu, C. F. J. On the Convergence Properties of the EM Algorithm. *The Annals of Statistics*, 11(1):95 – 103, 1983.

A. Algorithms

A.1. Algorithms in the independent and non independent case with details

Algorithms 3 and 4 includes practical details as warming and soft-start described in Section 3.4 in respectively the independent case and the non-independent case. The authors uses the following parameters:

- $\gamma_0 = 10^{-4}$,
- $K_{\text{pre-heating}} = 1000$,
- $C_{\text{heating}} = \frac{1}{1000}$,
- $\alpha = \frac{2}{3}$,

Algorithm 3 Fisher-SGD in the independent case with details

Input: $z_0, \theta_0, y_1, \ldots, y_N$
Algorithm parameters: $\gamma_0, K_{\text{pre-heating}}, C_{\text{heating}}, \alpha$
Initialize a 3-order mean filter with constant C_{heating} to compute mean grad.
Set heating as not finished
for k=1,...,K **do**
  {*The following loop is operated with vector calculus without explicit loop.*}
  for i=1,...,N **do**
    $z_i^k \sim \Pi_{\theta_{k-1}}\left(\cdot \,\middle|\, z_i^{k-1}, y_i\right)$ where $\Pi_\theta(\cdot \mid \cdot, y_i)$ is a transition kernel of a MCMC having $p_\theta(\cdot \mid y_i)$ as stationary distribution for all θ
  end for
  if $k < K_{\text{pre-heating}}$ **then**
    Set $\gamma_k = \exp\left((1 - k/K_{\text{pre-heating}}) \log \gamma_0\right)$
  else if heating not finished **then**
    Set $\gamma_k = 1$
  else
    Set $\gamma_k = (k - K_{\text{end-heating}})^{-\alpha}$
  end if
  {*The following loop is computed without explicit loop, all the gradients are computed in one step as the jacobian of the vector of criterion, and Δ with vector calculus.*}
  for i=1,...,N **do**
    Compute $J_i^k = \nabla_\theta \log f\left(y_i, z_i^k; \theta_{k-1}\right)$
    Compute $\Delta_i^k = (1 - \gamma_k)\Delta_i^{k-1} + \gamma_k J_i^k$
  end for
  Compute $v_k = \frac{1}{N} \sum_{i=1}^N J_i^k$
  {*The following loop is computed with vector calculus*}
  Compute $I_k = \dfrac{1}{N} \sum_{i=1}^N \Delta_i^k \left(\Delta_i^k\right)^T$
  if $k < K_{\text{pre-heating}}$ **then**
    Set $P_k = (1 - \gamma_k) \max\left(1, \operatorname{tr}(I_k)\right) Id + \gamma_k I_k$
  else
    Set $P_k = I_k$
  end if
  if $k > K_{\text{pre-heating}}$ and heating not finished **then**
    Update 3-order mean filter with v_k
    if norm-2 of 3-order mean of gradient is increasing **then**
      Set heating finished
      Set $K_{\text{end-heating}} = k$
    end if
  end if
  Set $\theta_k = \theta_{k-1} + \gamma_k P_k^{-1} v_k$
end for
Output: Estimated parameter: θ_k, estimated Fisher information matrix of the whole sample: $N I_k$, sample of latent in the posterior distribution: $\left(z^{K-r+1} \cdots z^K\right)$.

Algorithm 4 Fisher-SGD in the non independent case with details

Input: z_0, θ_0, y
Algorithm parameters: $\gamma_0, K_{\text{pre-heating}}, C_{\text{heating}}, \alpha$
Initialize a 3-order mean filter with constant C_{heating} to compute mean grad.
Set heating as not finished
for k=1,...,K **do**
  $z^k \sim \Pi_{\theta_{k-1}}\left(\cdot \,\middle|\, z^{k-1}, y\right)$ where $\Pi_\theta(\cdot \mid \cdot, y)$ is a transition kernel of a MCMC having $p_\theta(\cdot \mid y)$ as stationary distribution for all θ
  if $k < K_{\text{pre-heating}}$ **then**
    Set $\gamma_k = \exp\left((1 - k/K_{\text{pre-heating}}) \log \gamma_0\right)$
  else if heating not finished **then**
    Set $\gamma_k = 1$
  else
    Set $\gamma_k = (k - K_{\text{end-heating}})^{-\alpha}$
  end if
  Compute $v_k = \nabla_\theta \log f\left(y, z^k; \theta_{k-1}\right)$
  Compute I_k with custom method.
  if $k < K_{\text{pre-heating}}$ **then**
    Set $P_k = (1 - \gamma_k) \max\left(1, \operatorname{tr}\left(I_k\right)\right) Id + \gamma_k I_k$
  else
    Set $P_k = I_k$
  end if
  if $k > K_{\text{pre-heating}}$ and heating not finished **then**
    Update 3-order mean filter with v_k
    if norm-2 of 3-order mean of gradient is increasing **then**
      Set heating finished
      Set $K_{\text{end-heating}} = k$
    end if
  end if
  Set $\theta_k = \theta_{k-1} + \gamma_k P_k^{-1} v_k$
  if $k > K - r$ **then**
    Compute $\nabla_\theta^2 \log f\left(y, z^k; \theta_k\right)$
  end if
end for
Output: Estimated parameter: θ_k, estimated Fisher information matrix of the whole sample: $\frac{1}{r} \sum_{k=K-r+1}^{K} \nabla_\theta^2 \log f\left(y, z^k; \theta_k\right)$, sample of latent in the posterior distribution: $\left(z^{K-r+1} \cdots z^K\right)$.

Algorithm 5 Fisher-SGD for the Stochastic Block Model with details

Input: z_0, θ_0, y
Algorithm parameters: $\gamma_0, K_{\text{pre-heating}}, C_{\text{heating}}, \alpha$
Initialize a 3-order mean filter with constant C_{heating} to compute mean grad.
Set heating as not finished
for k=1,...,K **do**
  {*Here, a Gibbs sampler is used*}
  for i in random permutation of 1,...,N **do**
    $z_i^k \sim f\left(\cdot \,\middle|\, \left(z_j^{k-1}\right)_{j\neq i}, y; \theta_{k-1}\right)$
  end for
  if $k < K_{\text{pre-heating}}$ **then**
    Set $\gamma_k = \exp\left(\left(1 - k/K_{\text{pre-heating}}\right)\log\gamma_0\right)$
  else if heating not finished **then**
    Set $\gamma_k = 1$
  else
    Set $\gamma_k = \left(k - K_{\text{end-heating}}\right)^{-\alpha}$
  end if
  {*The following loops is computed without explicit loops, all the gradients are computed in one step as the jacobian of the vector of criterion, and Δ^{obs} with vector calculus.*}
  for i in 1,...,N **do**
    for j in 1,...,N **do**
      Compute $J_{ij}^{k,\text{obs}} = \nabla_\theta \log f\left(y_{ij} | z_i^k, z_j^k; \theta_{k-1}\right)$
      Compute $\Delta_{ij}^{k,\text{obs}} = (1-\gamma_k)\Delta_{ij}^{k-1,\text{obs}} + \gamma_k J_{ij}^{k,\text{obs}}$
    end for
  end for
  {*The following loop is computed without explicit loop, all the gradients are computed in one step as the jacobian of the vector of criterion, and Δ^{lat} with vector calculus.*}
  for i in 1,...,N **do**
    Compute $J_i^{k,\text{lat}} = \nabla_\theta \log f\left(z_i^k; \theta_{k-1}\right)$
    Compute $\Delta_i^{k,\text{lat}} = (1-\gamma_k)\Delta_i^{k-1,\text{lat}} + \gamma_k J_i^{k,\text{lat}}$
  end for
  Compute $v_k = \sum_{ij} J_{ij}^{k,\text{obs}} + \sum_i J_i^{k,\text{lat}}$
  {*The following computation is computed efficiently with Einstein summation*}
  Compute $I_k = \sum_{ij} \Delta_{ij}^{k,\text{obs}} \left(\Delta_{ij}^{k,\text{obs}}\right)^T + \sum_i \Delta_i^{k,\text{lat}} \left(\Delta_i^{k,\text{lat}}\right)^T$
  if $k < K_{\text{pre-heating}}$ **then**
    Set $P_k = (1-\gamma_k)\max\left(1, \operatorname{tr}\left(I_k\right)\right) Id + \gamma_k I_k$
  else
    Set $P_k = I_k$
  end if
  if $k > K_{\text{pre-heating}}$ and heating not finished **then**
    Update 3-order mean filter with v_k
    if norm-2 of 3-order mean of gradient is increasing **then**
      Set heating finished
      Set $K_{\text{end-heating}} = k$
    end if
  end if
  Set $\theta_k = \theta_{k-1} + \gamma_k P_k^{-1} v_k$
end for
Output: Estimated parameter: θ_k, estimated Fisher information matrix of the whole sample: $\frac{1}{r}\sum_{k=K-r+1}^{K} \nabla_\theta^2 \log f\left(y, z^k; \theta_k\right)$, sample of latent in the posterior distribution: $\left(z^{K-r+1} \cdots z^K\right)$.

B. Proofs of the theorems

Let us first recall the expression of the FIM estimate proposed by Delattre & Kuhn (2023):

$$\widehat{I}(\theta) = \frac{1}{N}\sum_{i=1}^{N} \mathrm{E}\left(\nabla_\theta \log f(y_i, Z_i; \theta) \mid y_i; \theta\right) \mathrm{E}\left(\nabla_\theta \log f(y_i, Z_i; \theta) \mid y_i; \theta\right)^T.$$

B.1. First setting: realizations of latent are generated from the posterior distribution

We adapt the proof of the convergence of the standard stochastic gradient algorithms (see, e.g. Bottou et al. (2018) or Ghadimi & Lan (2013)) to our algorithm. The specificity in our proof comes from the control of the preconditioner term.

Proof of Theorem 3.4. By Taylor-Lagrange inequality and under Assumption 3.3.1, we get for all k

$$\begin{aligned} F(\theta_{k+1}) \leq & F(\theta_k) + \langle \nabla_\theta F(\theta_k), \theta_{k+1} - \theta_k \rangle + \frac{L}{2}\|\theta_{k+1} - \theta_k\|^2 \\ \leq & F(\theta_k) + \gamma_k \langle \nabla_\theta F(\theta_k), \widehat{I}(\theta_k)^{-1} \nabla_\theta \log f(y, Z^{k+1}; \theta_k) \rangle \\ & + \gamma_k^2 \frac{L}{2} \|\widehat{I}(\theta_k)^{-1} \nabla_\theta \log f(y, Z^{k+1}; \theta_k)\|^2 \end{aligned}$$

We introduce some notations to simplify the writing of the proof. Let $H_\theta(Z) = -\nabla_\theta \log f(y, Z; \theta)$. We introduce also the following notation E_k for the expectation taking with respect to the posterior distribution p_{θ_k}.

Taking the conditional expectation in the previous inequality and noting that $\mathrm{E}_k(H_{\theta_k}(Z^{k+1})) = \nabla_\theta F(\theta_k)$, we get:

$$\begin{aligned} \mathrm{E}_k[F(\theta_{k+1})] \leq & F(\theta_k) - \gamma_k \langle \nabla_\theta F(\theta_k), \widehat{I}(\theta_k)^{-1} \mathrm{E}_k(H_{\theta_k}(Z^{k+1})) \rangle \\ & + \gamma_k^2 \frac{L}{2} \mathrm{E}_k[\|\widehat{I}(\theta_k)^{-1} H_{\theta_k}(Z^{k+1})\|^2] \\ \leq & F(\theta_k) - \gamma_k \|\hat{I}(\theta_k)^{-1/2} \nabla_\theta F(\theta_k)\|^2 \\ & + \gamma_k^2 \frac{L}{2} \mathrm{E}_k[\|\widehat{I}(\theta_k)^{-1} H_{\theta_k}(Z^{k+1})\|^2] \end{aligned}$$

Under Assumption 3.3.2 and 3, taking the total expectation and the sum for l between 0 and k, we obtain

$$\mu_m \mathrm{E}\left(\sum_{l=0}^{k} \gamma_l \|\nabla_\theta F(\theta_l)\|^2\right) \leq F(\theta_0) - \mathrm{E}(F(\theta_{k+1})) + \mu_M^2 \frac{CL}{2} \sum_{l=0}^{k} \gamma_l^2.$$

We finally obtain the result by noticing that for all $0 \leq l \leq k$ $\|\nabla F(\theta_l)\|^2 \geq \min_{0 \leq l' \leq k} \|\nabla F(\theta_{l'})\|^2$ for all l and $\mathrm{E}[F(\theta_{k+1})] \geq \min F$. □

B.2. Second setting: realizations of latent are generated from a transition kernel from an ergodic Markov chain having the posterior distribution as stationary distribution

Proof of Theorem 3.7. We consider now the setting where at iteration k the realization Z^k is sampled from a transition kernel of a Markov chain having the posterior distribution as stationary distribution. To state the convergence proof in this setting let us introduce for all integer k the $\sigma-$algebra $\mathcal{F}_k = \sigma(\theta_0, Z_l, 0 \leq l \leq k)$.

In this case the expectation $\mathrm{E}(H_{\theta_k}(Z^{k+1})|\mathcal{F}_k)$ is not equal to $\nabla_\theta F(\theta_k)$, leading to the presence of a supplementary term due to the use of a MCMC in the simulation task. The proof begins the same way as in the previous case.

By Taylor-Lagrange inequality and under Assumption 3.3.1, we get for all k

$$\begin{aligned} F(\theta_{k+1}) \leq & F(\theta_k) + \langle \nabla_\theta F(\theta_k), \theta_{k+1} - \theta_k \rangle + \frac{L}{2}\|\theta_{k+1} - \theta_k\|^2 \\ \leq & F(\theta_k) + \gamma_k \langle \nabla_\theta F(\theta_k), \widehat{I}(\theta_k)^{-1} \nabla_\theta \log f(y, Z^{k+1}; \theta_k) \rangle \\ & + \gamma_k^2 \frac{L}{2} \|\widehat{I}(\theta_k)^{-1} \nabla_\theta \log f(y, Z^{k+1}; \theta_k)\|^2 \end{aligned}$$

Taking first the expectation conditionally to the $\sigma-$algebra $\mathcal{F}_k$ and introducing $\nabla_\theta F(\theta_k)$, we get:

$$\begin{aligned}\mathrm{E}(F(\theta_{k+1})|\mathcal{F}_k) \leq & F(\theta_k) - \gamma_k \langle \nabla_\theta F(\theta_k), \widehat{I}(\theta_k)^{-1} \nabla_\theta F(\theta_k)\rangle \\ & + \gamma_k \langle \nabla_\theta F(\theta_k), \widehat{I}(\theta_k)^{-1} (\nabla_\theta F(\theta_k) - \mathrm{E}(H_{\theta_k}(Z^{k+1})|\mathcal{F}_k))\rangle \\ & + \gamma_k^2 \frac{L}{2} \mathrm{E}(\|\widehat{I}(\theta_k)^{-1} H_{\theta_k}(Z^{k+1})\|^2|\mathcal{F}_k)\end{aligned}$$

The difficulty here is to control the additional term $B_k = \langle \nabla_\theta F(\theta_k), \widehat{I}(\theta_k)^{-1} (\nabla_\theta F(\theta_k) - \mathrm{E}(H_{\theta_k}(Z^{k+1})|\mathcal{F}_k))\rangle$ in the second line raised up by the MCMC procedure used for the simulation of Z^k.

Let us introduce the notation $\eta_k = H_{\theta_k}(Z^{k+1}) - \nabla_\theta F(\theta_k)$.

Taking full expectation of the previous inequality, we get

$$\begin{aligned}\mathrm{E}(F(\theta_{k+1})) \leq & \mathrm{E}(F(\theta_k)) - \gamma_k \mathrm{E}(\langle \nabla_\theta F(\theta_k), \widehat{I}(\theta_k)^{-1} \nabla_\theta F(\theta_k)\rangle) \\ & - \gamma_k \mathrm{E}(\langle \nabla_\theta F(\theta_k), \widehat{I}(\theta_k)^{-1} \mathrm{E}(\eta_k|\mathcal{F}_k)\rangle) \\ & + \gamma_k^2 \frac{L}{2} \mathrm{E}(\|\widehat{I}(\theta_k)^{-1} H_{\theta_k}(Z^{k+1})\|^2)\end{aligned}$$

Therefore reordering the terms we get

$$\begin{aligned}\gamma_k \mathrm{E}(\langle \nabla_\theta F(\theta_k), \widehat{I}(\theta_k)^{-1} \nabla_\theta F(\theta_k)\rangle) \leq & \mathrm{E}(F(\theta_k)) - \mathrm{E}(F(\theta_{k+1})) \\ & - \gamma_k \mathrm{E}(\langle \nabla_\theta F(\theta_k), \widehat{I}(\theta_k)^{-1} \mathrm{E}(\eta_k|\mathcal{F}_k)\rangle) \\ & + \gamma_k^2 \frac{L}{2} \mathrm{E}(\|\widehat{I}(\theta_k)^{-1} H_{\theta_k}(Z^{k+1})\|^2)\end{aligned}$$

Under Assumption 3.3.2 and 3.3.3, suming for l between 0 and k, we obtain

$$\begin{aligned}\mu_m \mathrm{E}\left(\sum_{l=0}^k \gamma_l \|\nabla_\theta F(\theta_l)\|^2\right) \leq & F(\theta_0) - \mathrm{E}(F(\theta_{k+1})) + \mu_M^2 \frac{CL}{2} \sum_{l=0}^k \gamma_l^2 \\ & - \mathrm{E}(\sum_{l=0}^k \gamma_l \langle \nabla_\theta F(\theta_l), \widehat{I}(\theta_l)^{-1} \mathrm{E}(\eta_l|\mathcal{F}_l)\rangle)\end{aligned}$$

To control the last term, we apply the result of Lemma B.3 below which proves that $\sum \gamma_k \langle \nabla_\theta F(\theta_k), \widehat{I}(\theta_k)^{-1} \eta_k\rangle$ converges almost surely. Therefore we get:

$$\mu_m \mathrm{E}\left(\sum_{l=0}^k \gamma_l \|\nabla_\theta F(\theta_l)\|^2\right) \leq F(\theta_0) - \mathrm{E}(F(\theta_{k+1})) + \mu_M^2 \frac{CL}{2} \sum_{l=0}^k \gamma_l^2 + C$$

Finally we get the result by dividing the inequality by $\sum_{l=0}^k \gamma_l$.

□

Before stating Lemma B.3, we first establish several preliminary results needed for the proof. In particular we first introduce the Poisson equation as done for example in Allassonniere & Kuhn (2015); Atchadé et al. (2017) and establish several technical lemmas derived below. Recall $H_\theta(Z) = -\nabla_\theta \log f(y, Z; \theta)$ and $\eta_k = H_{\theta_k}(Z^{k+1}) - \nabla_\theta F(\theta_k)$.

Lemma B.1. *Assume 3.5. Then there exists a measurable function $(\theta, z) \to \hat{H}_\theta(z)$ such that $\sup_\theta |\hat{H}_\theta|_W < \infty$ and for any $(\theta, z) \in \Theta \times \mathcal{Z}$,*

$$\hat{H}_\theta(z) - \Pi_\theta \hat{H}_\theta(z) = H_\theta(z) - \int H_\theta(z) p_\theta(z) dz \tag{4}$$

Moreover there exists a constant C such that for any $(\theta, \theta') \in \Theta^2$,

$$\|\Pi_\theta \hat{H}_\theta(z) - \Pi_{\theta'} \hat{H}_{\theta'}(z)\|_W \leq C\|\theta - \theta'\|$$

Proof of Lemma B.1. The proof is established in details in Lemma 4.2 of (Fort et al., 2011). □

Lemma B.2. *Under Assumptions 3.5.1,3.5.2, we have* $\sup_k \mathrm{E}(W^p(Z^k)) < \infty$.

Proof of Lemma B.2. Since Z_{k+1} is generated from the transition kernel $\Pi_{\theta_k}(\cdot|Z^k, y)$, we get :

$$\mathrm{E}(W^p(Z^{k+1})) = \mathrm{E}(\mathrm{E}(W^p(Z^{k+1})|\mathcal{F}_k)) = \mathrm{E}(\Pi_{\theta_k} W^p(Z^k))$$

Applying the drift inequality of Assumption 3.5.2, we get

$$\mathrm{E}(W^p(Z^{k+1})) \leq \lambda\, \mathrm{E}(W^p(Z^k)) + b$$

The result is then obtained by induction. □

Lemma B.3. *Assume Assumptions 3.5.1, 3.5.2, 3.5.3, 3.6 and* Θ *is bounded. Then* $\sum \gamma_k \langle \nabla_\theta F(\theta_k), \widehat{I}(\theta_k)^{-1}\eta_k\rangle$ *converges almost surely.*

Proof of Lemma B.3. Applying Lemma B.1, we get that there exist a function $\hat{H}_{\theta_k}$ satisfying equation (4). Therefore we get $\eta_k = \hat{H}_{\theta_k}(Z^{k+1}) - \Pi_{\theta_k}\hat{H}_{\theta_k}(Z^{k+1})$. Let us denote by $M_k = \hat{H}_{\theta_k}(Z^{k+1}) - \Pi_{\theta_k}\hat{H}_{\theta_k}(Z^k)$, $R_k = \Pi_{\theta_k}\hat{H}_{\theta_k}(Z^k) - \Pi_{\theta_{k+1}}\hat{H}_{\theta_{k+1}}(Z^{k+1})$, and $R'_k = \Pi_{\theta_{k+1}}\hat{H}_{\theta_{k+1}}(Z^{k+1}) - \Pi_{\theta_k}\hat{H}_{\theta_k}(Z^{k+1})$ such that $\eta_k = M_k + R_k + R'_k$. We will prove successively that the three sums $\mathrm{E}(\sum \gamma_k \|\langle \nabla_\theta F(\theta_k), \widehat{I}(\theta_k)^{-1} M_k\rangle\|)$, $\mathrm{E}(\sum \gamma_k \|\langle \nabla_\theta F(\theta_k), \widehat{I}(\theta_k)^{-1} R_k\rangle\|)$, $\mathrm{E}(\sum \gamma_k \|\langle \nabla_\theta F(\theta_k), \widehat{I}(\theta_k)^{-1} R'_k\rangle\|)$ are finite with probability one.

Let us first note that the term $\gamma_k \langle \nabla_\theta F(\theta_k), \widehat{I}(\theta_k)^{-1} M_k\rangle$ is a martingale increment with respect to the filtration $(\mathcal{F}_k)$. By Lemma B.1 and under Assumption 3.5.2, we get that there exists C such that with probability one for all k $\|M_k\| \leq C(W(Z^{k+1}) + W(Z^k))$. Applying classical result on martingales (Hall & Heyde, 2014), we get that $\sum \gamma_k \langle \nabla_\theta F(\theta_k), \widehat{I}(\theta_k)^{-1} M_k\rangle$ converges almost surely if $\sum \gamma_k^2 \|\widehat{I}(\theta_k)^{-1}\nabla_\theta F(\theta_k)\|^2 \|M_k\|^2 < \infty$ almost surely. By Assumption 3.2, $\|\widehat{I}(\theta_k)^{-1}\nabla_\theta F(\theta_k)\|^2 \leq C\mu_M^2$ and $\|M_k\|^2 \leq 2C(W^2(Z^{k+1}) + W^2(Z^k))$ leading to $\mathrm{E}(\sum \gamma_k^2 \|\widehat{I}(\theta_k)^{-1}\nabla_\theta F(\theta_k)\|^2 \|M_k\|^2) < \infty$ which gives the control of the second sum.

Concerning the second term, applying the Abel transformation leads to

$$\begin{aligned}\sum_{k=0}^{K} \gamma_k \langle \nabla_\theta F(\theta_k), \widehat{I}(\theta_k)^{-1} R_k\rangle &= \sum_{k=1}^{K} \langle \gamma_k \widehat{I}(\theta_k)^{-1}\nabla_\theta F(\theta_k) - \gamma_{k-1}\widehat{I}(\theta_{k-1})^{-1}\nabla_\theta F(\theta_{k-1}), \Pi_{\theta_k}\hat{H}_{\theta_k}(Z^k)\rangle \\ &+ \gamma_0 \langle \widehat{I}(\theta_0)^{-1}\nabla_\theta F(\theta_0), \Pi_{\theta_0}\hat{H}_{\theta_0}(Z^0)\rangle - \gamma_K \langle \widehat{I}(\theta_K)^{-1}\nabla_\theta F(\theta_K), \Pi_{\theta_{K+1}}\hat{H}_{\theta_{K+1}}(Z^{K+1})\rangle\end{aligned}$$

Let us denote by $\xi_k = \gamma_k \widehat{I}(\theta_k)^{-1}\nabla_\theta F(\theta_k) - \gamma_{k-1}\widehat{I}(\theta_{k-1})^{-1}\nabla_\theta F(\theta_{k-1})$. Therefore we get

$$\begin{aligned}\|\xi_k\| &\leq |\gamma_k - \gamma_{k-1}| \|\widehat{I}(\theta_k)^{-1}\nabla_\theta F(\theta_k)\| + \gamma_{k-1}\|\widehat{I}(\theta_k)^{-1}\nabla_\theta F(\theta_k) - \widehat{I}(\theta_k)^{-1}\nabla_\theta F(\theta_{k-1})\| \\ &+ \gamma_{k-1}\|\widehat{I}(\theta_k)^{-1}\nabla_\theta F(\theta_{k-1}) - \widehat{I}(\theta_{k-1})^{-1}\nabla_\theta F(\theta_{k-1})\|\end{aligned}$$

Under Assumption 3.2, there exists $M > 0$ such that for all k $\|\widehat{I}(\theta_k)^{-1} - \widehat{I}(\theta_{k-1})^{-1}\| \leq M\|\theta_k - \theta_{k-1}\|$ therefore we get

$$\begin{aligned}\|\xi_k\| &\leq |\gamma_k - \gamma_{k-1}| \|\widehat{I}(\theta_k)^{-1}\nabla_\theta F(\theta_k)\| + \gamma_{k-1}\mu_M \|\nabla_\theta F(\theta_k) - \nabla_\theta F(\theta_{k-1})\| \\ &+ \gamma_{k-1} CM \|\theta_k - \theta_{k-1}\| \\ &\leq |\gamma_k - \gamma_{k-1}| C\mu_M + \gamma_{k-1}\mu_M L\|\theta_k - \theta_{k-1}\| + \gamma_{k-1} CM\|\theta_k - \theta_{k-1}\|\end{aligned}$$

Taking the expectation, we get

$$\mathrm{E}(\|\xi_k\|) \leq |\gamma_k - \gamma_{k-1}| C\mu_M + \gamma_{k-1}^2 C(\mu_M L + CM)$$

By Lemma B.1 and Assumptions 3.5.2 and 3.5.3 there exists C such that with probability one for all k $\|\Pi_{\theta_k}\hat{H}_{\theta_k}(Z^k)\| \leq CW(Z^k)$. Moreover by Lemma B.2, we get $\sup_k \mathrm{E}(W^2(Z^k)) < \infty$ thus we get $\sum \mathrm{E}(\|\gamma_k \langle \nabla_\theta F(\theta_k), \widehat{I}(\theta_k)^{-1} R_k\rangle\|) < \infty$ which control the second sum.

Finally let us consider the third sum. By Lemma B.1 there exists C such that with probability one for all k

$$\begin{aligned}\|\langle \nabla_\theta F(\theta_k), \widehat{I}(\theta_k)^{-1} R'_k \rangle\| &\leq C\mu_M \|\theta_k - \theta_{k+1}\| W(Z^{k+1}) \\ &\leq C\mu_M^2 \gamma_{k+1} W^2(Z^{k+1})\end{aligned}$$

By Lemma B.2 we get $\sum \gamma_k^2 \, \mathrm{E}(W^2(Z^{k+1})) < \infty$ which implies that $\mathrm{E}(\sum \gamma_k \|\langle \nabla_\theta F(\theta_k), \widehat{I}(\theta_k)^{-1} R'_k \rangle\|)$ is finite almost surely.

This concludes the proof of Lemma B.3. □

C. Reparametrization of models and confidence intervals

C.1. Reparametrization

Models in numerical experiments use a constrained parameter space. In particular, we have:

Logistic growth mixed-effects model: parameters are $\beta \in \mathbb{R}_+^* \times \mathbb{R}$, $\alpha \in \mathbb{R}_+^*$, $\Gamma \in \mathbb{S}_{++}^2$, where $\mathbb{S}_p^{++}$ is the set of symmetric, positive definite matrices of size $p \times p$ and $\sigma \in \mathbb{R}_+^*$. Then, the original parameter space is $\mathbb{R}_+^* \times \mathbb{R} \times \mathbb{R}_+^* \times \mathbb{S}_2^{++} \times \mathbb{R}_+^*$ where S_2^{++} is the set of 2×2 positive definite matrix.

Stochastic Block Model: parameters are $\alpha \in \mathring{\mathcal{S}}_{K-1}$ and $p \in (0,1)^{K \times K}$ where $\mathring{\mathcal{S}}_{K-1} \subset \mathbb{R}^K$ is the $K-1$ dimentionnal open unit simplex.

Handling directly this parameter space in our algorithm could lead to constraint violations. To handle constraints we propose to use a bijective differentiable mapping from the constrained parameter space to $\mathbb{R}^d$.

Reparametrization are build using the Parametrization Cookbook (Leger, 2023), and practically using the Python module `parametrization_cookbook`.

We introduce the original parameter as a function of $\theta \in \Theta = \mathbb{R}^d$.

Logistic growth mixed-effects model: $\Theta = \mathbb{R}^d$ with $d = 7$, and we use α_θ, β_θ, Γ_θ and σ_θ.

Stochastic Block Model: $\Theta = \mathbb{R}^d$ with $d = K^2 + K - 1$, and we use α_θ and p_θ.

C.2. Confidence ellipsoid

We obtain $\widehat{\theta}$ an estimate of θ_0 and $\widehat{I}_{\text{whole}}\left(\widehat{\theta}\right)$ the estimation of the Fisher Information Matrix of the whole sample. Therefore we have asymptotically:

$$\widehat{I}_{\text{whole}}\left(\widehat{\theta}\right)^{1/2}\left(\widehat{\theta} - \theta_0\right) \xrightarrow[n \to +\infty]{(d)} \mathcal{N}(0, I_d).$$

So we compute the asymptotic confidence ellipsoid on θ_0 at confidence level $1-a$:

$$\left\{\theta \in \mathbb{R}^d \,:\, \left(\theta - \widehat{\theta}\right)^T \left[\widehat{I}_{\text{whole}}\left(\widehat{\theta}\right)\right]\left(\theta - \widehat{\theta}\right) \leq \chi^2_{d;1-a}.\right\}$$

C.3. Confidence interval on original parameter

With η an original parameter of the model before reparametrization (see section above for original parameter in logistic growth mixed-effects model or in stockastic block model). After reparametrization this original parameter is a function of θ, noted η_θ.

Applying the reparametrization, from $\widehat{\theta}$, we obtain $\eta_{\widehat{\theta}}$ an estimate of η_0.

Applying delta-method (Van der Vaart, 2000), we obtain the asymptotic distribution of $\eta_{\widehat{\theta}}$:

$$\frac{\eta_{\widehat{\theta}} - \eta_0}{\sqrt{g_{\eta_{\widehat{\theta}}}^T \left[\widehat{I}_{\text{whole}}\left(\widehat{\theta}\right)\right]^{-1} g_{\eta_{\widehat{\theta}}}}} \xrightarrow[n \to +\infty]{(d)} \mathcal{N}(0,1),$$

with

$$g_{\eta_{\widehat{\theta}}} = \left.\frac{\mathrm{d}\,\eta_\theta}{\mathrm{d}\,\theta}\right|_{\theta = \widehat{\theta}},$$

and $g_{\eta_{\widehat{\theta}}}$ is computed with automatic differentiation.

Then we obtain the asymptotic confidence interval on η_0 at confidence level $(1-a)$:

$$\left[\eta_{\widehat{\theta}} \pm u_{1-a/2}\sqrt{g_{\eta_{\widehat{\theta}}}^T \left[\widehat{I}_{\text{whole}}\left(\widehat{\theta}\right)\right]^{-1} g_{\eta_{\widehat{\theta}}}}\right].$$

D. Supplementary numerical results

D.1. Logistic growth mixed-effects model

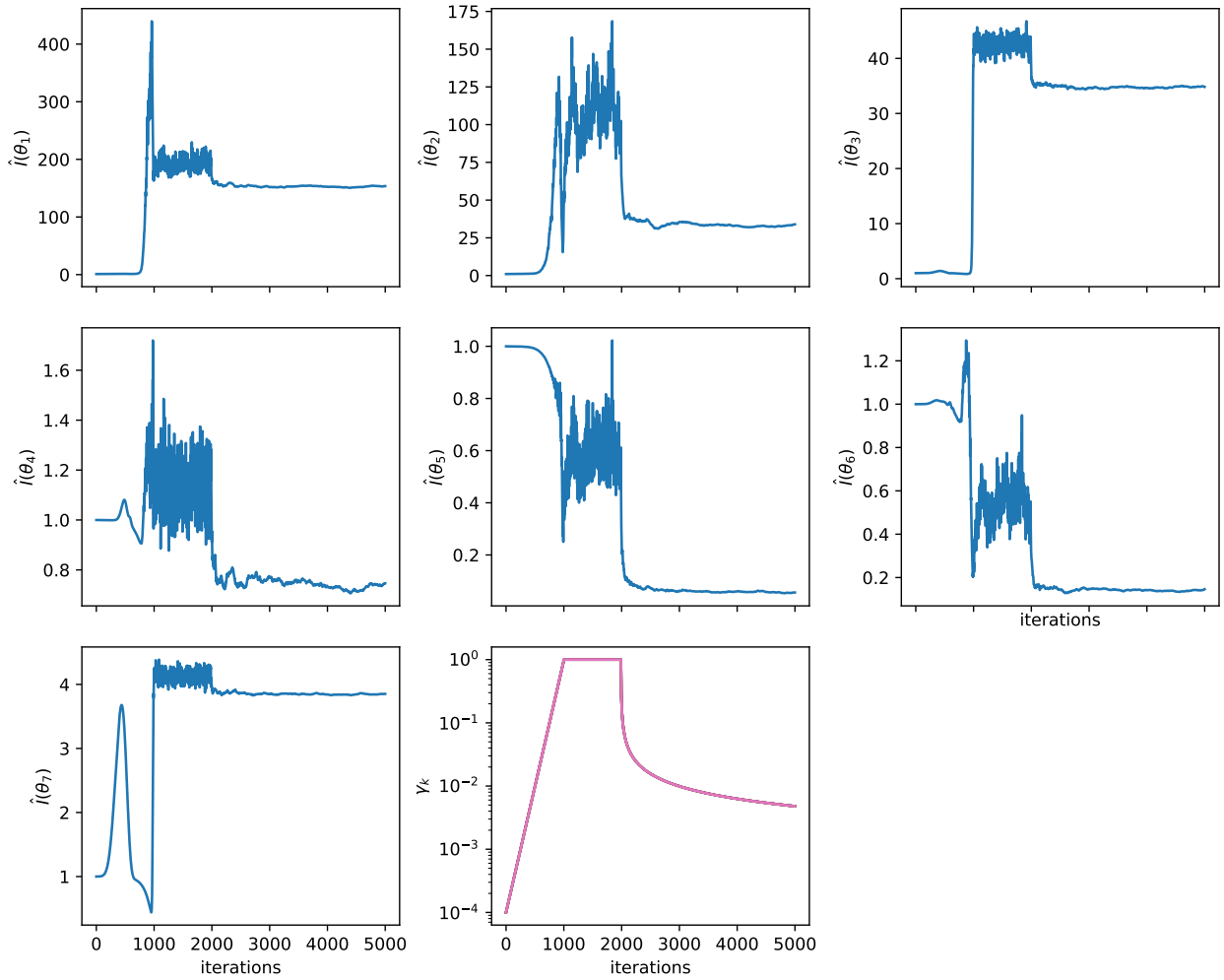

Figure 3. Evolution of the diagonal elements of the FIM estimate across the iterations (in blue), along with the evolution of the learning step (in red).

D.2. Real data analysis

We applied our algorithm to a real dataset from a study on coucal growth rates (Goymann et al., 2016). In this study, body weights of $N = 259$ birds were collected from their hatching date until they left the nest (see Figure 4). The number of measurements per bird ranges from 1 to 9.

The logistic growth mixed model defined in Section D.1 was fitted to the dataset, with the asymptotic weight and the inflexion point (i.e. the age at which bird i reaches half its asymptotic body mass) as random effects. The tuning parameters of the algorithm were set as follows: $K_{pre-heating} = 2000$, $K = 10000$, $C_{heating} = 100$, $\alpha = 2/3$, $\lambda_0 = 10^{-4}$, and the algorithm was initialized at a random value for θ. Results are given in Figure 5. The final estimates were $\hat{\beta}_1 = 97.15$, $\hat{\beta}_2 = 6.50$, $\hat{\alpha} = 2.80$, $\hat{\Gamma}_{11} = 271.00$, $\hat{\Gamma}_{12} = 7.75$, $\hat{\Gamma}_{22} = 1.10$ and $\hat{\sigma}^2 = 19.80$. Our results are consistent with those provided by the `semix` package implemented in `R`, which performs maximum likelihood estimation using the SAEM algorithm for which theoretical guarantees only exist in the exponential family setting. However in our case, due to the presence of a fixed effect, the model does not belong to the curved exponential family as explained in the main body of the paper.

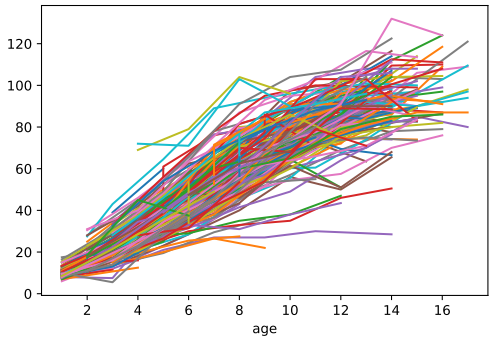

Figure 4. Evolution of body weight (in g) as a function of age (in days from hatching)

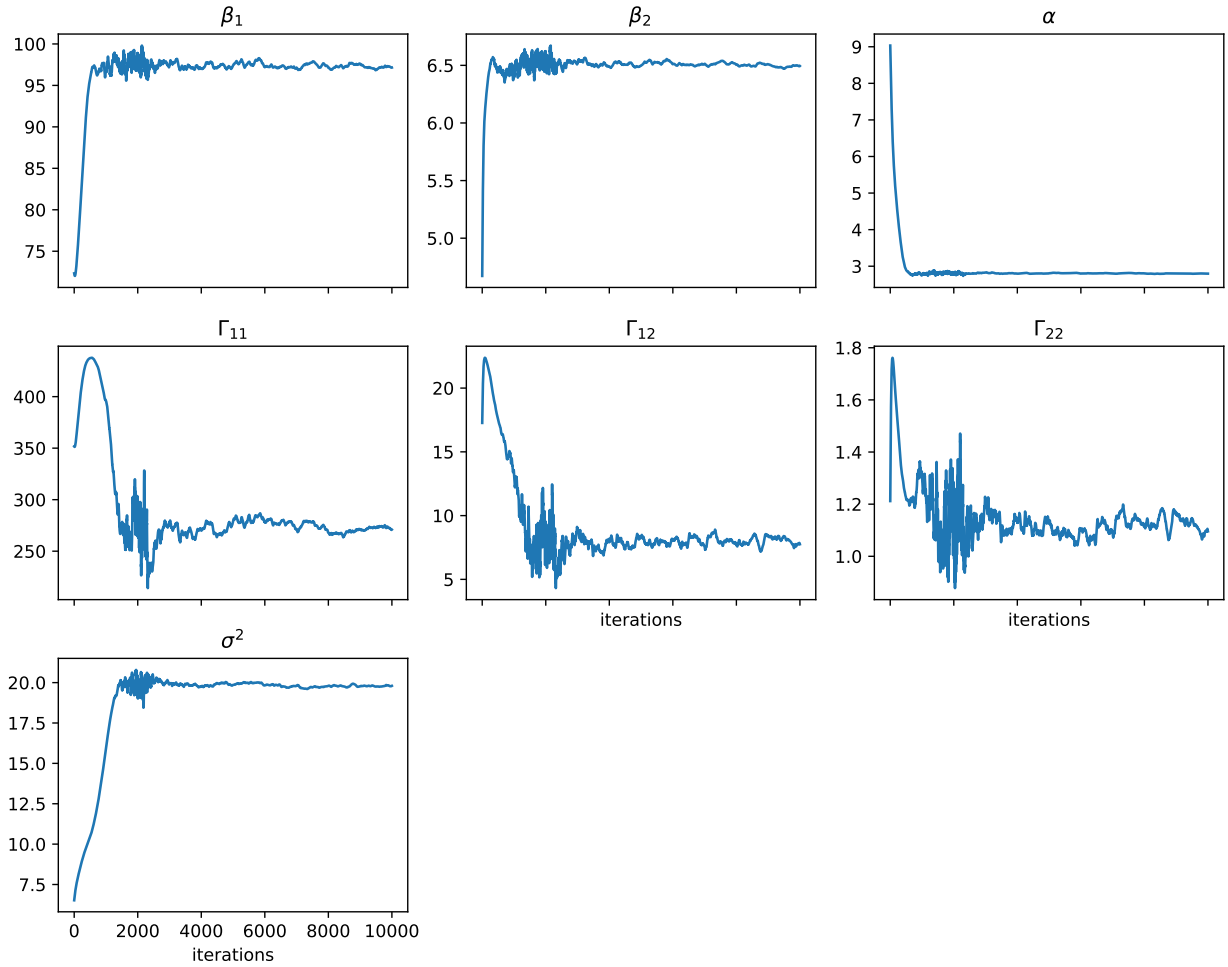

Figure 5. Evolution of the parameter estimates across the iterations on the real dataset.

D.3. Stochastic Block Model

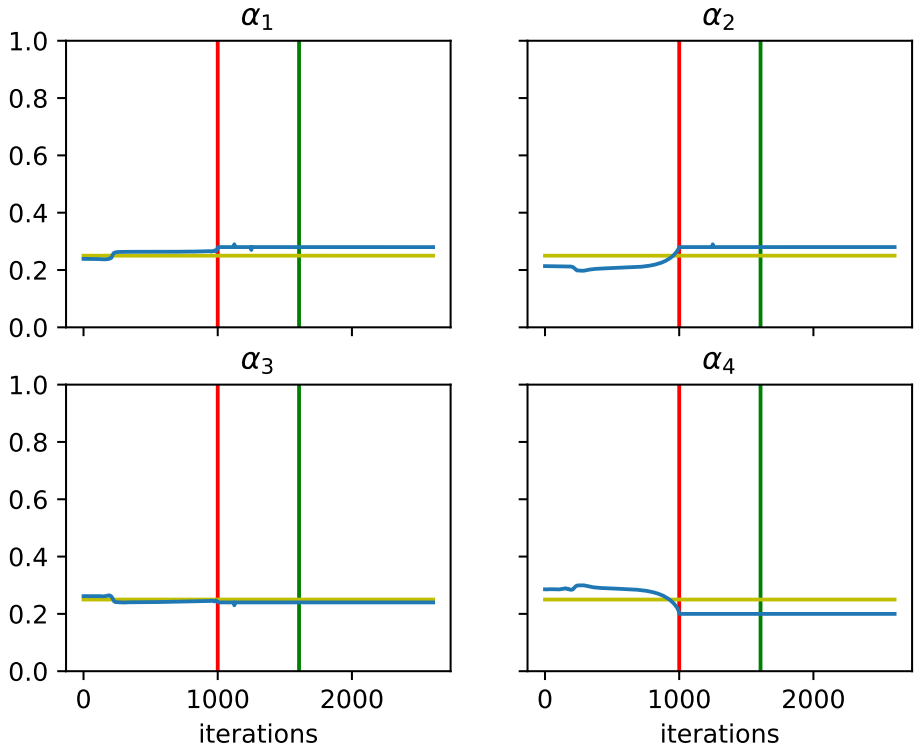

Figure 6. Evolution of the α estimates across the iterations with $N = 100$ and $K = 4$. Yellow line: simulated value. The red line is the end of the pre-heating, and the green line is the end of the heating.

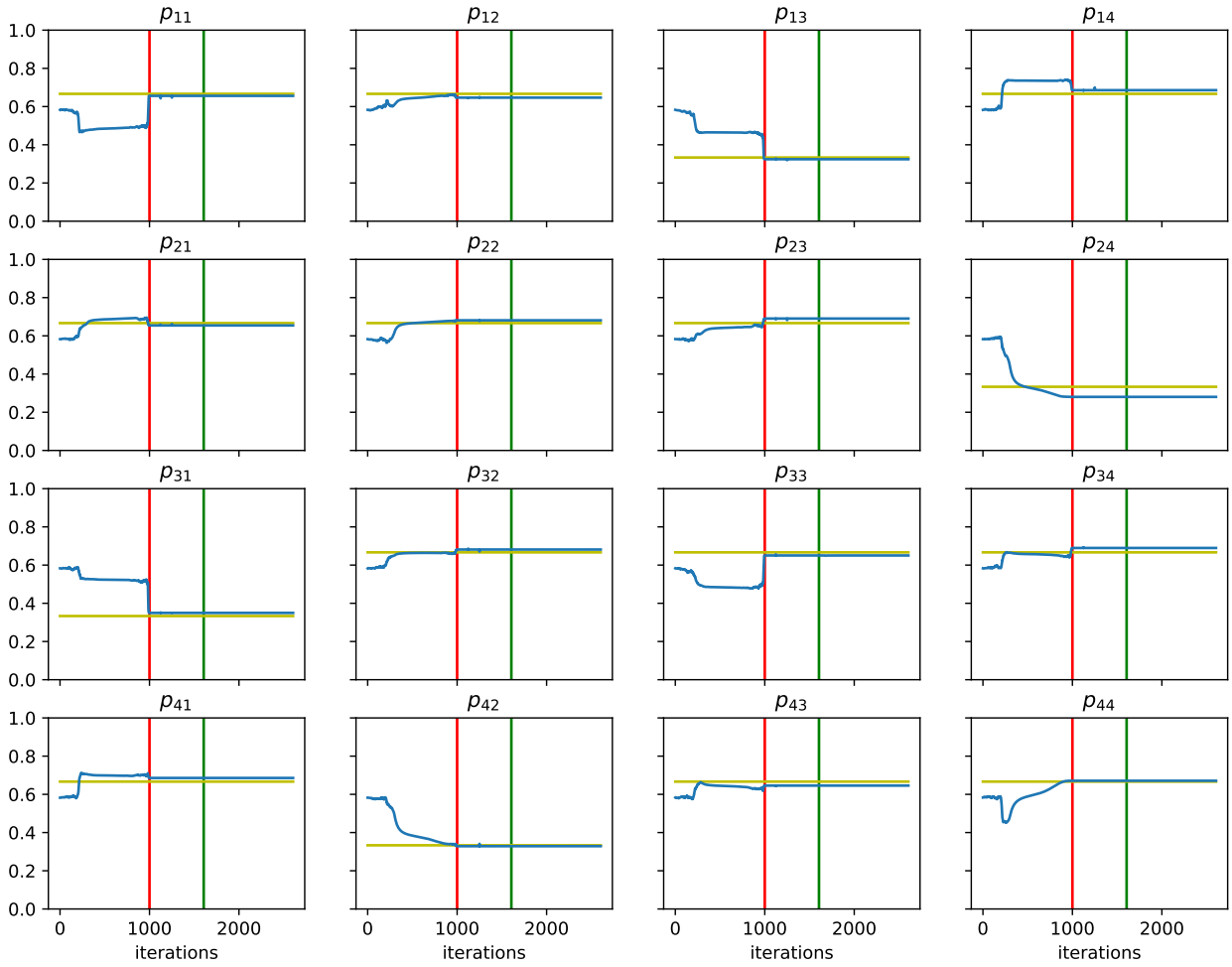

Figure 7. Evolution of the p estimates across the iterations with $N = 100$ and $K = 4$. Yellow line: simulated value. The red line is the end of the pre-heating, and the green line is the end of the heating.

		$N = 100$		$N = 200$	
Parameter	Simulated value	RMSE	Empirical coverage	RMSE	Empirical coverage
Global θ		0.648	0.936 ± 0.011	0.415	0.956 ± 0.009
α_1	0.250	0.044	0.943 ± 0.010	0.031	0.940 ± 0.010
α_2	0.250	0.044	0.939 ± 0.010	0.031	0.943 ± 0.010
α_3	0.250	0.044	0.941 ± 0.010	0.031	0.931 ± 0.011
α_4	0.250	0.044	0.945 ± 0.010	0.031	0.940 ± 0.010
$p_{1,1}$	0.667	0.023	0.940 ± 0.010	0.012	0.949 ± 0.010
$p_{1,2}$	0.667	0.019	0.947 ± 0.010	0.010	0.949 ± 0.010
$p_{1,3}$	0.333	0.022	0.948 ± 0.010	0.012	0.953 ± 0.009
$p_{1,4}$	0.667	0.019	0.947 ± 0.010	0.010	0.948 ± 0.010
$p_{2,1}$	0.667	0.020	0.944 ± 0.010	0.010	0.947 ± 0.010
$p_{2,2}$	0.667	0.022	0.945 ± 0.010	0.012	0.952 ± 0.009
$p_{2,3}$	0.667	0.020	0.940 ± 0.010	0.010	0.953 ± 0.009
$p_{2,4}$	0.333	0.020	0.942 ± 0.010	0.011	0.951 ± 0.010
$p_{3,1}$	0.333	0.023	0.943 ± 0.010	0.013	0.943 ± 0.010
$p_{3,2}$	0.667	0.019	0.950 ± 0.010	0.010	0.939 ± 0.011
$p_{3,3}$	0.667	0.023	0.941 ± 0.010	0.013	0.955 ± 0.009
$p_{3,4}$	0.667	0.020	0.943 ± 0.010	0.009	0.950 ± 0.010
$p_{4,1}$	0.667	0.020	0.939 ± 0.010	0.009	0.957 ± 0.009
$p_{4,2}$	0.333	0.020	0.949 ± 0.010	0.011	0.947 ± 0.010
$p_{4,3}$	0.667	0.019	0.955 ± 0.009	0.010	0.952 ± 0.009
$p_{4,4}$	0.667	0.024	0.946 ± 0.010	0.011	0.954 ± 0.009

Table 3. Detailed result for numerical experiments on SBM with 2000 replications. The RMSE given for θ is the empirical mean of $\|\widehat{\theta} - \theta_0\|^2$ and the coverage is the coverage of the confidence ellipsoid in $\mathbb{R}^{Q^2+Q-1}$ built at the nominal level of 0.95. For all original parameter, the RMSE is computed after transformation of $\widehat{\theta}$ in original parameter space, and the confidence interval built at the nominal level of 0.95 is obtained by applying delta method with the Fisher Information Matrix estimate.