Some EM-type algorithms for incomplete data model building

Marc Lavielle

To cite this version:

Marc Lavielle. Some EM-type algorithms for incomplete data model building. 2021. hal-03512130

HAL Id: hal-03512130
https://hal.science/hal-03512130

Preprint submitted on 5 Jan 2022

Some EM-type algorithms for incomplete data model building

Marc Lavielle[a,b]

[a] *Inria, France*

[b] *CMAP, Ecole Polytechnique, CNRS, IPP, France*

Abstract

We propose an extension of the EM algorithm and its stochastic versions for the construction of incomplete data models when the selected model minimizes a penalized likelihood criterion. This optimization problem is particularly challenging in the context of incomplete data, even when the model is relatively simple. However, by completing the data, the E-step of the algorithm allows us to simplify this problem of complete model selection into a classical problem of complete model selection that does not pose any major difficulties. We then show that the criterion to be minimized decreases with each iteration of the algorithm. Examples of the use of these algorithms are presented for the identification of regression mixture models and the construction of nonlinear mixed-effects models.

Keywords: EM, SAEM, Model selection, Penalized likelihood

1. Introduction

The Expectation-Maximization (EM) algorithm is undoubtedly the most popular tool for maximum likelihood (ML) estimation in incomplete data problems of many types [1, 2]. When the model contains nonlinearities, performing the E-step is often untractable. Stochastic versions of EM have then been proposed to circumvent this difficulty by replacing the computation of this conditional expectation with a Monte Carlo approximation in the MCEM algorithm [3] or with a stochastic approximation method in the SAEM algorithm [4, 5].

In addition to the theoretical properties of convergence of these algorithms, which have been established under fairly general conditions, their practical interest has been shown in many situations, such as the identification of mixture models [2] or the estimation of parameters in nonlinear mixed effects models [6].

The ability to estimate the parameters of a model is, of course, important, but for a modeller this is only an intermediate step in the not easy process of building the model: estimating the parameters of a model consists, in fact, in choosing the ”best” model (in the sense of likelihood) within a family of models that differ from each other precisely only by the value of their parameters. However, the main challenge for the modeller is to select the ”best” family of models.

This problem of model selection has been studied extensively and various approaches have been proposed, including fully Bayesian approaches [7] and criterion-based methods [8, 9, 10]. These methods, which aim to optimize a criterion such as the AIC or the BIC, are in fact penalized maximum likelihood methods, where the penalization concerns the number of parameters of the model: we look for the model that best fits the observations with the smallest number of parameters. In turn, the lasso method, originally proposed for the construction of a linear model, uses the sum of the absolute values of the coefficients of the model as the penalty [11]. Note that because of the form of the ℓ_1-penalty, the lasso does both variable selection and shrinkage.

The performance of the selected model, both in terms of predictive quality and explanatory quality, depends of course on the criterion chosen. Nevertheless, we will not address this problem here, but rather focus on how to minimize the criterion used. Several optimization algorithms have already been proposed, but mainly for linear regression problems. When the penalty concerns the number of variables selected, among a relatively small number d of predictors available, it is always possible to fit and compare the 2^d possible models. This exhaustive search is no longer possible and can be replaced by an iterative search when the number of variables is important. In this way, stepwise regression consists of adding or removing a variable from the model at each step to reduce the criterion until no new change improves it [12]. On the other hand, least angle regression (LARS) is a very efficient algorithm for computing the lasso solution [13].

Things get seriously complicated when it comes to selecting a model with incomplete data, but the EM algorithm and its stochastic variants will prove perfectly suited to solve this problem. Indeed, the fact that the conditional distribution of the missing data can be used at each iteration to ”complete” the data in the E-step allows us to transform the problem of selecting the incomplete model into a much simpler problem of selecting the complete model in the M-step. For example, if the model for the missing data is a linear model to be constructed, then the ”classical” variable selection methods can be used with the completed data. Interestingly, we can also demonstrate that the penalized likelihood criterion decreases with each iteration of the EM algorithm.

Two examples illustrate the proposed algorithms. First, we show that this EM algorithm in its deterministic version allows the identification of a regression mixture model [14, 15, 16]. Step E consists here in computing the conditional probabilities for each observation to belong to the different classes. The completed model is then a weighted linear regression model for each class, which we can easily build in the M step, using either BIC or lasso. On the other hand, SAEM consists of generating the unobserved labels at each iteration: the data are then classified in a natural way so that we can build a regression model by class.

The second example concerns the construction of a nonlinear mixed-effects model using SAMBA (Stochastic Approximation for Model Building Algorithm) [17]. SAMBA is an extension of SAEM for constructing complex models of this type and consists at each iteration of estimating the parameters of the current model with maximum likelihood, generating the unobserved data using the estimated conditional distribution, and constructing a new model using the completed data. A Monte Carlo experiment with simulated data illustrates the good practical properties of the algorithm. Finally, an application to pharmacokinetic (PK) data shows that the algorithm is able to build a very good statistical model for these data very quickly.

2. Incomplete data model building

2.1. Incomplete data model selection

The models we are interested in here involve a set of observed variables y and a set of unobserved, or latent, variables ψ. Model selection then consists of selecting a particular model $\widehat{\mathcal{M}}$ from a (possibly very large) set of models $\mathbb{M}$. If data are available, the obvious choice is to use them for this selection, as opposed to latent data, which by definition cannot be observed and used.

In a probabilistic framework, a model $\mathcal{M}$ is a joint probability distribution $\mathrm{p}(y, \psi; \mathcal{M})$. Selection methods based on the likelihood of the model can only use the *observed data likelihood* function $\mathcal{L}(\mathcal{M}; y) \overset{\text{def}}{=} \mathrm{p}(y; \mathcal{M})$, where $\mathrm{p}(y; \mathcal{M})$ is the pdf of the observations computed under model $\mathcal{M}$. Introducing a penalty term $\mathrm{pen}(\mathcal{M})$, that favors some models and disfavors some others is a classic way to incorporate some prior information about the model. It is also an efficient way to control the complexity of the model to avoid selecting a model capable of fitting observations excessively well but with poor predictive performance. The selected model thus minimizes a penalized criterion:

$$U(\mathcal{M}; y) = -2 \log \left(\mathrm{p}(y; \mathcal{M}) \right) + \mathrm{pen}(\mathcal{M})$$

$$\widehat{\mathcal{M}} = \arg \min_{\mathcal{M} \in \mathbb{M}} U(\mathcal{M}, y)$$

Two main problems arise when implementing such a model selection method: the choice of the penalty term $\mathrm{pen}(\mathcal{M})$ and the minimization of the criterion $U(\mathcal{M}, y)$.

The problem of choosing the penalty term is basically a problem of statistical inference. Indeed, this choice must be guided by the statistical properties of the model $\widehat{\mathcal{M}}$ that one wants to select. Nevertheless, we will not deal here with the problem of the choice of $\mathrm{pen}(\mathcal{M})$ and its properties. We are only concerned with the problem of minimizing the penalized criterion $U(\mathcal{M}, y)$, which is a purely algorithmic problem. In other words: we focus here on the computation of $\widehat{\mathcal{M}}$, regardless of the properties of this model.

This optimization problem is particularly tricky when, on the one hand, the criterion is difficult to compute for a given model, and on the other hand when the number of models to be compared is very large. The natural idea to tackle such a problem in a context of incomplete data is to extend the EM algorithm and its stochastic variants.

2.2. *EM-type algorithms*

2.2.1. *EM algorithm for model building*

Let us define a penalized version of the *complete log-likelihood* function:

$$V(\mathcal{M}; y, \psi) = -2\log\left(\mathrm{p}(y, \psi; \mathcal{M})\right) + \mathrm{pen}(\mathcal{M}). \tag{1}$$

We can then design the following *Expectation-Maximization* algorithm:

- An initial model $\mathcal{M}_0$ is chosen
- At iteration k
  - E-step: for any $\mathcal{M} \in \mathbb{M}$, let

    $$Q(\mathcal{M}, \mathcal{M}_{k-1}) = \mathbb{E}\left(V(\mathcal{M}, y, \psi) \,|\, y, \mathcal{M}_{k-1}\right)$$

  - M-step: compute

    $$\mathcal{M}_k = \arg\min_{\mathcal{M}\in\mathbb{M}} Q(\mathcal{M}, \mathcal{M}_{k-1})$$

Proposition 1 *The sequence of observed criteria* $(U(\mathcal{M}_k, y), k \geq 0)$ *is a decreasing sequence. Furthermore, if the sequence* $(U(\mathcal{M}_k, y), k \geq 0)$ *is bounded, then* $U(\mathcal{M}_k, y)$ *converges to some* $U^\star < \infty$.

The proof of this proposition is in the Appendix. It is a straightforward extension of the proof of convergence of the standard EM algorithm [1].

2.2.2. *SAEM algorithm for model building*

When the calculation of the conditional expectation cannot be performed in a closed form, it can be replaced by a stochastic approximation and/or a Monte-Carlo approximation. The algorithm in its most general form then consists of the following three steps at iteration k:

- **Simulation step**: R realizations $\psi^{(k,1)}, \ldots, \psi^{(k,R)}$ are drawn from the conditional distribution $\mathrm{p}(\psi|y; \mathcal{M}_{k-1})$.
- **Expectation step**: $Q(\mathcal{M}, \mathcal{M}_{k-1})$ is approximated by

  $$Q^{(k)}(\mathcal{M}) = Q^{(k-1)}(\mathcal{M}) + \gamma_k \left(\frac{1}{R}\sum_{r=1}^{R} V(\mathcal{M}, y, \psi^{(k,r)}) - Q^{(k-1)}(\mathcal{M})\right).$$

- **Maximization step**:

  $$\mathcal{M}_k = \arg\min_{\mathcal{M}\in\mathbb{M}} Q^{(k)}(\mathcal{M})$$

The sequence (γ_k) is such that $\sum_{k\geq 1}\gamma_k = +\infty$ and $\sum_{k\geq 1}\gamma_k^2 < +\infty$. Convergence of SAEM has been established under quite general conditions in the context of maximum likelihood estimation [4, 5].

Combining the stochastic approximation with a Monte-Carlo approximation permits to reduce the variance of the simulation and to better approximate the conditional expectation $\mathbb{E}\left(V(\mathcal{M}, y, \psi) \,|\, y, \mathcal{M}_{k-1}\right)$. Note that the use of multiple draws is usually quite easy to implement. Indeed, if an MCMC algorithm is used to generate these draws, it is possible to build multiple chains in parallel and also extract multiple draws from the same chain.

3. Parametric incomplete data model

3.1. Model selection and parameter estimation

A parametric model $\mathcal{M}_\theta$ assumes that the joint distribution of y and ψ is a parametric distribution $\mathrm{p}(y, \psi; \theta)$, where θ is a vector of parameters.

The problem of model selection then combines with a problem of parameter estimation:

- For a given family of models $\mathfrak{M} = \{\mathcal{M}_\theta\ ,\ \theta \in \Theta\}$, *estimation* consists in selecting a particular element θ in Θ. For instance, maximum likelihood (ML) estimation consists in selecting the element θ that maximizes the observed likelihood $\mathcal{L}_{\mathfrak{M}}(\theta; y)$.
- Let $\mathbb{M} = \{\mathfrak{M}^{(\ell)}\ ,\ 1 \leq \ell \leq L\}$ be a (finite) collection of families of models where $\mathfrak{M}^{(\ell)} = \{\mathcal{M}_\theta\ ,\ \theta \in \Theta^{(\ell)}\}$ is a family of parametric models. Then, *model selection* consists in selecting a family of models $\mathfrak{M}^{(\ell)}$ in $\mathbb{M}$ and a particular element θ in $\Theta^{(\ell)}$. In this context, model selection via penalized likelihood optimization consists in selecting a model $\widehat{\mathfrak{M}}$ and a vector of parameters $\hat{\theta}$ by minimizing a penalized criteria:

$$(\widehat{\mathfrak{M}}, \hat{\theta}) = \arg \min_{(\mathfrak{M}^{(\ell)} \in \mathbb{M}\ ,\ \theta \in \Theta^{(\ell)})} \left\{-2\log(\mathcal{L}_{\mathfrak{M}^{(\ell)}}(\theta; y)) + \mathrm{pen}(\mathfrak{M}^{(\ell)}, \theta)\right\} \tag{2}$$

Let's see how to design the EM and SAEM algorithms presented Section 2.2 to minimize such criterion.

3.2. EM for parametric model building

Here, EM consists in defining a sequence of families of models $(\mathfrak{M}_k \in \mathbb{M})$, i.e. a sequence of indexes $(l_k \in \{1, 2, \ldots, L\})$ where $\mathfrak{M}_k = \mathfrak{M}^{(l_k)}$, and a sequence $(\theta_k \in \Theta^{(l_k)})$ such that

$$(\mathfrak{M}_k, \theta_k) = \arg \min_{(\mathfrak{M}^{(\ell)} \in \mathbb{M}\ ,\ \theta \in \Theta^{(\ell)})} \left\{-2\mathbb{E}\left(\log\left(\mathrm{p}(y, \psi; \theta)\right) | \mathfrak{M}_{k-1}\right) + \mathrm{pen}(\mathfrak{M}^{(\ell)}, \theta)\right\}$$

The way this minimization problem is solved at each iteration of EM depends on the model and the penalization criterion chosen. For example, in the case of a mixture of regression models, we will see in Section 4.1 that different algorithms can be used to solve this problem, depending on whether the penalty criterion takes into account the number of variables in the model or their norm.

3.3. The SAMBA algorithm

The version of EM proposed above assumes that a new model must be selected at each iteration of the algorithm. This can be particularly costly if a large number of iterations are required to ensure convergence of an algorithm such as SAEM.

SAMBA (Stochastic Approximation for Model Building Algorithm) is a SAEM algorithm for parametric model building that allows to significantly reduce model updates by devoting most iterations to parameter estimation and very few iterations to model updating. Indeed, SAMBA exploits the fact that the optimization problem (2) can be decomposed into a parameter estimation problem and a model family selection problem. Indeed, for $\ell = 1, 2, \ldots L$, let

$$\hat{\theta}^{(\ell)} = \arg \min_{\theta \in \Theta^{(\ell)}} \left\{ -2 \log(\mathcal{L}_{\mathfrak{M}^{(\ell)}}(\theta; y)) + \mathrm{pen}(\mathfrak{M}^{(\ell)}, \theta) \right\} \tag{3}$$

be the penalized ML estimator of θ when the model is an element of the family $\mathfrak{M}^{(\ell)}$. Then, the selected family of models is

$$\begin{aligned} \widehat{\mathfrak{M}} &= \mathfrak{M}^{(\hat{\ell})} \\ &= \arg \min_{\mathfrak{M}^{(\ell)} \in \mathbb{M}} \{ -2 \log(\mathcal{L}_{\mathfrak{M}^{(\ell)}}(\hat{\theta}^{(\ell)}; y)) + \mathrm{pen}(\mathfrak{M}^{(\ell)}, \hat{\theta}^{(\ell)}) \} \end{aligned}$$

and the selected model is the element $\mathcal{M}_{\hat{\theta}(\hat{\ell})}$ in $\mathfrak{M}^{(\hat{\ell})}$.

Let us now look at how this property can be exploited to redesign SAEM. Assume that model family $\mathfrak{M}_k = \mathfrak{M}^{(\ell_k)}$ is selected at iteration k. Then, the three following steps are performed:

- **Estimation step**: Compute the penalized ML estimate of θ

$$\begin{aligned} \theta_k &= \hat{\theta}^{(\ell_k)} \\ &= \arg \min_{\theta \in \Theta^{(\ell_k)}} \{ -2 \log(\mathcal{L}_{\mathfrak{M}_k}(\theta; y)) + \mathrm{pen}(\mathfrak{M}_k, \theta) \} \end{aligned}$$

- **Simulation step**: Draw R realizations $\psi^{(k,1)}, \ldots, \psi^{(k,R)}$ of the conditional distribution $\mathrm{p}(\psi | y; \theta_k)$.
- **Selection step**: select a new model family $\mathfrak{M}_{k+1}$ using the R sets of completed data $(y, \psi^{(k,1)}), \ldots, (y, \psi^{(k,R)})$:

$$\mathfrak{M}_{k+1} = \arg \min_{\mathfrak{M}^{(\ell)} \in \mathbb{M}} \left\{ \min_{\theta \in \Theta^{(\ell)}} \{ -2 \Lambda_k(\theta, \mathfrak{M}^{(\ell)}) + \mathrm{pen}(\mathfrak{M}^{(\ell)}, \theta) \} \right\}$$

where

$$\Lambda_k(\theta, \mathfrak{M}^{(\ell)}) = \Lambda_{k-1}(\theta, \mathfrak{M}^{(\ell)}) + \gamma_k \left(\frac{1}{R} \sum_{r=1}^{R} \log(\mathcal{L}_{\mathfrak{M}^{(\ell)}}(\theta; y, \psi^{(k,r)})) - \Lambda_{k-1}(\theta, \mathfrak{M}^{(\ell)}) \right)$$

The main interest of this method lies in the fact that the selection step is generally very easy to implement once the data is complete and requires little computational effort.

On the other hand, the various numerical experiments we have conducted have shown that a practical and efficient stopping rule is to consider SAMBA has converged as soon as $\mathfrak{M}_{k+1} = \mathfrak{M}_k$. With this stopping rule, SAMBA usually converges in very few iterations with a constant step sequence $\gamma_k = 1$.

Finally, note that the estimation step can be performed using the standard SAEM algorithm for ML estimation [4, 5].

4. Illustrations

4.1. Mixture of linear regression models

4.1.1. The model

Finite mixture models aim to identify population heterogeneity through a finite set of latent classes. Within this framework, regression mixture models specifically seek to identify differences in the effect of a set of predictors on an outcome. These models are therefore quite widely used in various fields such as the social and behavioral sciences [14].

Here, we consider univariate observations y_1, y_2 ... y_n resulting from a mixture of G linear Gaussian models:

$$y_i \sim \sum_{g=1}^{G} \pi_g \, \mathcal{N}\left(X_i \boldsymbol{\beta}_g \, , \, \sigma_g^2\right)$$

where X_i is vector of d individual predictor variables and where $\boldsymbol{\beta}_g$ is a vector of coefficients that differ between populations.

The set of parameters of the model is $\theta = (\beta_{1,1}, \beta_{1,2} \ldots, \beta_{G,d}, \sigma_1^2, \ldots, \sigma_G^2, \pi_1, \ldots, \pi_G)$ and the probability density function of the observations is

$$\mathrm{p}(y \, ; \, \theta) = \prod_{i=1}^{n} \left(\sum_{g=1}^{G} \frac{\pi_g}{\sqrt{2\pi\sigma_g^2}} \exp\left\{ -\frac{1}{2\sigma_g^2} \left(y_i - X_i \boldsymbol{\beta}_g\right)^2 \right\} \right).$$

Among the various estimation and selection methods that have already been proposed for these models, we can mention, for example, a method for robust parameter estimation [18] and a method for selecting the number of components [14].

The regression models we want to build are models with a reduced number of predictors. In other words, we assume that several components of $\boldsymbol{\beta}_g$ are null in each subpopulation and that the list of null coefficients may differ between subpopulations. For each subpopulation, therefore, the problem is both one of model selection (selecting the relevant predictors) and estimation (estimating the non-zero coefficients).

This specific problem of variable selection in a regression mixture model can be addressed by several methods. In a Bayesian framework, a reversible jump Markov chain Monte Carlo has recently been proposed to model each component as a sparse regression model [15]. Alternatively, model selection and parameter estimation problems can be solved simultaneously by optimizing a penalized likelihood criterion of the form:

$$U(\theta\ ;\ y) = -2\log(\mathrm{p}(y\ ;\ \theta)) + \mathrm{pen}(\theta).$$

Namely, the selected family of models is implicitly defined by the list of non-zero estimated coefficients. The role of the penalization is to select the regression variables in each subpopulation. It is therefore only concerned with the vectors of coefficients $\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_G$.

A joint selection of the number of components and variables is performed in [19] using an extension of the Akaike Information Criteria (AIC). The method consists in fitting different models using EM and then comparing them with the proposed criterion. Such an approach is difficult to consider when there are a large number of predictors, as the number of models to be fitted can be very large. Several penalties based on the on the ℓ_1 norm of the $\boldsymbol{\beta}_g$'s are compared in [16]. Again, EM is used to fit the model, but a quadratic approximation of the penalty term is introduced to perform the M-step of the algorithm.

We will see that variable selection using a penalized criterion in this context can indeed be easily integrated into the EM and SAEM algorithms.

4.1.2. EM and SAEM algorithms

For this kind of mixture model, it is convenient to introduce a sequence of label variables ψ_1, ψ_2 ... ψ_n where $\psi_i \in \{1, 2, \ldots, G\}$ denotes the subpopulation from which y_i comes. Then, the complete regression model writes

$$y_i = \sum_{g=1}^{G} (X_i\, \boldsymbol{\beta}_g \ +\ \sigma_g\, e_i)\, 1\!\!1_{\psi_i = g}.$$

where $e_i \sim_{\text{i.i.d.}} \mathcal{N}(0, 1)$

It is assumed here that the penalization term only affects the regression coefficients and decomposes as $\mathrm{pen}(\theta) = \sum_{g=1}^{G} \mathrm{pen}(\boldsymbol{\beta}_g)$. Then, the completed objective function defined in (1) writes:

$$V(\theta; y, \psi) = \sum_{g=1}^{G} \left(\sum_{i=1}^{n} \left(-2\log(\pi_g) + \log(2\pi\sigma_g^2) + \frac{1}{\sigma_g^2}(y_i - X_i\, \boldsymbol{\beta}_g)^2 \right) 1\!\!1_{\psi_i = g} + \mathrm{pen}(\boldsymbol{\beta}_g) \right)$$

At iteration k of EM, E-step consists in computing

$$\begin{aligned} t_{i,g,k} &= \mathbb{E}\left(1\!\!1_{\psi_i=g} | y_i; \theta_{k-1}\right) \\ &= \frac{(\pi_{g,k-1}/\sigma_{g,k-1}) \exp\left\{-\frac{1}{2\sigma^2_{g,k-1}}\left(y_i - X_i\,\boldsymbol{\beta}_{g,k-1}\right)^2\right\}}{\sum_{h=1}^{G}(\pi_{h,k-1}/\sigma_{h,k-1}) \exp\left\{-\frac{1}{2\sigma^2_{h,k-1}}\left(y_i - X_i\,\boldsymbol{\beta}_{h,k-1}\right)^2\right\}} \end{aligned}$$

and M-step requires to compute

$$\begin{aligned} \pi_{g,k} &= \frac{1}{n}\sum_{i=1}^{n} t_{i,g,k} \\ \boldsymbol{\beta}_{g,k} &= \arg\min_{\boldsymbol{\beta}\in\mathbb{R}^d}\left(\sum_{i=1}^{n} t_{i,g,k}(y_i - X_i\,\boldsymbol{\beta})^2 + \text{pen}(\boldsymbol{\beta})\right) \\ \sigma^2_{g,k} &= \frac{\sum_{i=1}^{n} t_{i,g,k}(y_i - X_i\,\boldsymbol{\beta}_{g,k})^2}{\sum_{i=1}^{n} t_{i,g,k}} \end{aligned}$$

The only difficulty here is in updating the vectors of the coefficients $\boldsymbol{\beta}_{1,k}, \ldots, \boldsymbol{\beta}_{G,k}$. The problem that now arises, however, is much simpler than the original problem. Namely, it involves selecting the variables of a linear regression model for each group and estimating its parameters. There are well-known algorithms that can be used for this step, including least angle regression (LARS) to minimize the lasso loss function or stepwise variable selection for BIC or AIC. If the number d of predictor variables is not too large, it is even possible to fit the 2^d models for each subgroup and select the best one according to the chosen criterion.

A stochastic approximation version of this EM algorithm is straightforward to derive. Indeed, the simulation step of SAEM at iteration k consists in computing the conditional probabilities $(t_{i,g,k}\ , 1 \leq i \leq n, 1 \leq g \leq G)$ as we did for the EM algorithm and generating R sequences of labels $\psi^{(k,1)}, \ldots, \psi^{(k,R)}$ with these probabilities. The expectation step then reduces to updating the weights as follows:

$$w_{i,g,k} = w_{i,g,k-1} + \gamma_k\left(\frac{1}{R}\sum_{r=1}^{R} 1\!\!1_{\psi_i^{(k,r)}=g} - w_{i,g,k-1}\right)$$

The maximization step is then identical to that of the EM algorithm, with the conditional probabilities $(t_{i,g,k})$ replaced by the weights $(w_{i,g,k})$.

4.1.3. Numerical experiment

The simulation of the data and the implementation of the algorithms were carried out using R version 4.0.3.

We consider $G = 2$ subgroups of respectively $n_1 = 80$ and $n_2 = 120$ observations. For each individual, $d = 10$ regression variables were generated from a standard normal distribution. All the coefficients were set to 0 except $\beta_{1,3} = 2$ and $\beta_{1,5} = -1$ for the first group and $\beta_{2,2} = 2$ and $\beta_{2,4} = -1$ for the second group. The standard deviation of the residual error is the same in both groups: $\sigma_1 = \sigma_2 = 2$.

Figure A.1 shows the convergence of EM when BIC is used, i.e. setting $\text{pen}(\boldsymbol{\beta}_g) = \log(200)\ \sum_{j=1}^{10} \mathbb{1}_{\beta_{g,j} \neq 0}$ for $g = 1, 2$. Initial covariate model is an empty model for $g = 1$ and a full model for $g = 2$.

We can observe that EM converges very quickly in this example, finding the optimal model (which happens to be the "true" model used for the simulation here) in only five iterations. The next ten iterations permit to improve the estimation of the parameters.

The convergence of SAEM is now shown in Figure A.2, using the same initial estimate and criterion. It can be seen that the behavior of SAEM is similar to that of EM, but with random fluctuations that decrease during the iterations as the step size γ_k decreases.

Of course, the convergence of the algorithm does not always go so well... The next example, based on the use of lasso, will then show us that very simple extensions to the basic algorithm can be used to improve both the convergence of the algorithm and the quality of the solution obtained.

Instead of a ℓ_0 norm for BIC, lasso regularization is a ℓ_1 norm: $\text{pen}(\boldsymbol{\beta}_g) = \lambda \sum_{j=1}^{10} |\beta_{g,j}|$ for $g = 1, 2$. Parameter λ controls the number of non-zero coefficients in the regression models.

Figure A.3-A shows the convergence of the regression coefficients of the first subgroup only when $\lambda = 15$. We can see that a spurious variable is added to the first regression model since $\hat{\beta}_{1,6} = -0.063$. The estimated value of this coefficient seems very small and would lead us instead to eliminate the 6th covariate from the model. Such a filtering can be automatically performed in several ways. A particularly efficient method is to keep only those variables that are significantly correlated with the data. For each subgroup, we can then compute at each iteration of EM the correlation between each covariate and the weighted data (using the $(t_{i,g,k}\ , 1 \leq i \leq n, 1 \leq g \leq G)$ as weights), and eliminate those with a small correlation. We see Figure A.3-B that the 6th covariate is eliminated when a minimum absolute correlation of 0.2 is required. One could also think of increasing the value of λ to eliminate this covariate. Nevertheless, Figure A.3-C shows that EM converges poorly with $\lambda = 18$ instead of $\lambda = 15$: This is indeed an example where EM converges to a local minimum of the criterion. Convergence of EM is displayed Figure A.3-D when the regularization coefficient λ changes during the iterations using the following scheme: $\lambda_k = 15 + 3k/K$. Progressively increasing the penalty thus makes it possible to improve the convergence of the algorithm in this example.

Figure A.3-A shows the convergence of the regression coefficients of the first subgroup only when $\lambda = 15$. We see that a spurious variable has been added to the first regression model, as $\hat{\beta}_{1,6} = -0.063$. The estimated value of this coefficient appears to be very small and would lead us to eliminate the 6th covariate from the model. Such filtering can be performed automatically in a number of ways. One particularly efficient method is to keep only those variables that are significantly correlated with the data. For each subgroup, we can then compute the correlation between each covariate and the weighted data at each iteration of EM (using $(t_{i,g,k}\ , 1 \leq i \leq n, 1 \leq g \leq G)$ as weights) and eliminate those with low correlation. We see in Figure A.3-B that the 6th covariate is eliminated when a minimum absolute correlation of 0.2 is required. One could also think of increasing the value of λ to eliminate this covariate. However, Figure A.3-C shows that EM converges poorly with $\lambda = 18$ instead of $\lambda = 15$: This is indeed an example of EM converging to a local minimum of the criterion. The convergence of EM is shown in Figure A.3-D when the regularization coefficient λ changes during the iterations according to the following scheme: $\lambda_k = 15 + 3k/K$. Thus, the fact that the penalty is progressively increased makes it to improve the convergence of the algorithm.

These variants of EM, which we have introduced here (selecting variables only among those that are statistically significant, using a non-constant penalty parameter during iterations), seem to improve the convergence of the algorithm. There is no doubt that other modifications could also improve this convergence. A more comprehensive study on this topic would certainly be worthwhile.

4.2. Nonlinear mixed-effects models

4.2.1. The model

Let y_i be the n_i-vector of measurements for individual i, $1 \leq i \leq N$, where N is the number of individuals. To account for variability among individuals, we assume that the model for y_i depends on a vector ψ_i of individual parameters and possibly a vector of population parameters ξ. For example, a model for continuous longitudinal data writes

$$y_{ij} = f(t_{ij}, \psi_i) + g(t_{ij}, \psi_i, \xi)\varepsilon_{ij} \ \ , \ 1 \leq i \leq N \ \ , \ \ 1 \leq j \leq n_i. \tag{4}$$

were y_{ij} is the observation obtained from subject i at time t_{ij}. The residual errors (ε_{ij}) are assumed to be standard normal random variables. The residual error model is defined by function g in model (4). A limited number of possible error models will be considered in the numerical examples: constant ($g = a$), proportional ($g = bf$), combined1 ($g = a+bf$) and combined2 ($g = \sqrt{a^2 + b^2 f^2}$).

On the other hand, we assume a linear Gaussian model for the vector of individual parameters ψ_i. More precisely, we assume that there exists a one-to-one transformation h, a vector of typical parameter values $\psi_{\rm pop}$, a vector of coefficients β and a covariance matrix Ω such that

$$h(\psi_i) \sim \mathcal{N}(h(\psi_{\rm pop}) + C_i\beta \ , \ \Omega) \ \ , \ 1 \leq i \leq N. \tag{5}$$

where matrix C_i is formed from individual covariates that are supposed to explain some of the variability of the ψ_i's.

Building a model in this context is particularly complex, as it is a matter of selecting the structural model f, the residual error model g, the transformation h, the covariate model, i.e. the structure of matrix C_i, and the correlation model, i.e. the structure of matrix Ω.

Initial methods for constructing the covariate and correlation models are proposed in [20]. The vast majority of the methods developed thereafter mainly concern the covariate model [21, 22, 23]. We will see here how to use SAMBA to build the covariate model, the correlation model, and the residual error model, given the structural model f and the transformation h.

We propose to use the corrected version of the BIC proposed in [24] as a model selection criterion. This criterion, denoted BICc, penalizes differently the different components of the vector $\theta = (\psi_{\rm pop}, \beta, \Omega, \xi)$.

Let $n_{\rm tot} = \sum_i n_i$ be the total number of observations, $d_{\psi_{\rm pop}}$ be the number of parameters in the structural model f, d_β be the total number of coefficients in the covariate model, and d_Ω be the number of non-zero variances and correlations in Ω. Then for each model family $\mathfrak{M}^{(\ell)}$ and each $\theta \in \Theta^{(\ell)}$,

$$\text{pen}_{\rm BICc}(\mathfrak{M}^{(\ell)}, \theta) = \log(n_{\rm tot})(d_{\psi_{\rm pop}} + d_\xi) + \log(N)(d_\beta + d_\Omega)$$

4.2.2. SAMBA for nonlinear mixed-effects models

First, we can note that the criterion BICc depends only on the non-zero elements of the components of θ, not on their values. Then, for a given family of models $\mathfrak{M}^{(\ell)}$, the penalization used in (3) to define the penalized ML estimate of θ is the same for each $\theta \in \Theta^{(\ell)}$. Consequently,

$$\hat{\theta}^{(\ell)} = \arg \min_{\theta \in \Theta^{(\ell)}} \left\{-2\log(\mathcal{L}_{\mathfrak{M}^{(\ell)}}(\theta; y))\right\}.$$

At each iteration of SAMBA, the *Estimation step* and the *Simulation step* can be performed using the SAEM algorithm for ML estimation and a Markov chain Monte Carlo algorithm, respectively, as described in [25]. These algorithms have already proven their worth. We will not go into further detail here, considering that we have the ML estimate θ_k and R realizations of the conditional distribution $\mathrm{p}(\psi|y;\theta_k)$ available at iteration k.

However, the new problem we face is the one related to the *Selection step*. We will take advantage of the fact that the joint distribution of y and ψ naturally decomposes into a product of two distributions:

$$\mathrm{p}(y, \psi\ ;\ \theta) = \mathrm{p}(\psi\ ;\zeta)\mathrm{p}(y|\psi\ ;\ \xi)$$

where $\zeta = (\psi_{\rm pop}, \beta, \Omega)$. We can then split the model selection problem into two subproblems: the selection of the linear Gaussian model $\mathfrak{M}_\psi$ on the one hand, and the selection of the conditional model $\mathfrak{M}_{y|\psi}$ on the other. Note that for the continuous data defined in (4), the selection of the conditional model is reduced to the selection of the residual error model when the structural model f is fixed.

For any $k > 0$, let $\alpha_k = \sum_{m=1}^{k}(1-\gamma_m)/R$. Then,

$$\mathfrak{M}_{\psi,k+1} = \arg\min_{\mathfrak{M}_\psi^{(\ell)} \in \mathbb{M}_\psi} \left\{\min_{\zeta \in Z^{(\ell)}} \left\{-2\Lambda_{\psi,k}(\zeta, \mathfrak{M}_\psi^{(\ell)}) + \log(N)(d_\beta + d_\Omega)\right\}\right\} \quad (6)$$

$$\mathfrak{M}_{y|\psi,k+1} = \arg\min_{\mathfrak{M}_{y|\psi}^{(\ell)} \in \mathbb{M}_{y|\psi}} \left\{\min_{\xi \in \Xi^{(\ell)}} \left\{-2\Lambda_{y|\psi,k}(\xi, \mathfrak{M}_{y|\psi}^{(\ell)}) + \log(n_{\rm tot})d_\xi\right\}\right\} \quad (7)$$

where

$$\Lambda_{\psi,k}(\zeta, \mathfrak{M}_{\psi}^{(\ell)}) = \sum_{m=1}^{k} \sum_{r=1}^{R} \alpha_k \log(\mathcal{L}_{\mathfrak{M}_{\psi}^{(\ell)}}(\zeta; \psi^{(m,r)})) \tag{8}$$

$$\Lambda_{y|\psi,k}(\xi, \mathfrak{M}_{y|\psi}^{(\ell)}) = \sum_{m=1}^{k} \sum_{r=1}^{R} \alpha_k \log(\mathcal{L}_{\mathfrak{M}_{y|\psi}^{(\ell)}}(\xi; y, \psi^{(m,r)})) \tag{9}$$

The first optimization problem (6) is to select the linear Gaussian model that maximizes the penalized weighted likelihood defined in (8). This is a fairly classical problem that does not present any particular difficulties [26, 27].

The second optimization problem (7) can be easily solved by directly computing and comparing the selection criterion for the different possible residual error models.

4.2.3. Numerical experiment

This numerical experiment is based on 100 simulations of the same experiment. For each simulation, data were generated from a two-compartment pharmacokinetics (PK) model:

$$\begin{aligned} \frac{dA_c}{dt}(t) &= -\frac{Q}{V_c}A_c(t) + \frac{Q}{V_p}A_p(t) - \frac{Cl}{V_c}A_c(t) \\ \frac{dA_p}{dt}(t) &= \frac{Q}{V_c}A_c(t) - \frac{Q}{V_p}A_p(t) \end{aligned} \tag{10}$$

Here $A_c(t)$ and $A_p(t)$ are, respectively, the amounts of drug in the central and peripheral compartments at time t while $C_c(t) = A_c(t)/V_c$ is the concentration in the central compartment. The parameters of the model are the central and peripheral volumes V_c and V_p and the central and intercompartmental clearances Cl and Q.

$N = 100$ vectors of 50 individual covariates $C_i = (C_{1,i}, \ldots, C_{50,i})$ were drawn from standard normal distributions. Then, $N = 100$ vectors of individual parameters $\psi_i = (V_{c,i}, V_{p,i}, Cl_i, Q_i)$ were obtained from log-normal distributions:

$$\begin{aligned} &\log(V_{c,i}) \sim \mathcal{N}(\log(6) + 0.2C_{1,i} + 0.3C_{2,i}\,,\, 0.2^2) \quad , \\ &\log(V_{p,i}) \sim \mathcal{N}(\log(10)\,,\, 0.3^2) \quad , \\ &\log(Cl_i) \sim \mathcal{N}(\log(20) + 0.3C_{3,i}\,,\, 0.4^2) \quad , \\ &\log(Q_i) \sim \mathcal{N}(\log(5) + 0.4C_{1,i}\,,\, 0.4^2) \quad , \\ &\mathrm{Cor}(\log(V_{c,i}), \log(Cl_i)) = 0.6 \quad . \end{aligned}$$

Observed drug concentrations were then simulated for the $N = 100$ individuals at times (0.1h, 0.25h, 0.75h, 1h, 2h, 4h, 8h) when a single dose of 1000mg is administrated by intravenous bolus at $t = 0$ and assuming a proportional error model:

$$y_{ij} \sim \mathcal{N}\left(C_c(t_j\,;\,\psi_i)\,,\, 0.2^2 C_c^2(t_j\,;\,\psi_i)\right)$$

We then used SAMBA with each of the 100 simulated trials to build the statistical model. The initial model assumes that there are no relationships between covariates and individual parameters, no correlation between random effects, and a constant error model.

Each time, the algorithm selected a model that closely resembled the ”true” model used for the simulation. Indeed, on average over the 100 replicates, 3.92 of the 4 existing relationships between covariates and individual parameters were correctly detected (true positives), while only 2.44 of the 196 nonexisting relationships were falsely considered to exist (false positives). The existing correlation between $\log(Cl)$ and $\log(V_c)$ was correctly detected in 84 of the 100 replicates, while a total of 8 false correlations were detected. Finally, the correct error model was identified in 94% of the cases, while a combined model was selected in the remaining 6% of the simulations.

It is important to emphasise that in the cases where the true model was not selected, the final model was very often better than the true model in terms of corrected BIC. Indeed, the difference in BICc between these two models was between -2 and -30 in 71% of the cases and between $+2$ and $+9$ in only 6% of the cases. These replicates correspond to runs where the algorithm failed to converge to the global minimum of the chosen criterion.

Finally, we note that the average time of a run was 75s (sd=24s) on a standard laptop.

4.2.4. Application to tranexamic PK data

Tranexamic acid (TXA) is an antifibrinolytic agent that controls bleeding. In [28], a population-based pharmacokinetic study conducted to quantify TXA exposure is described. Data were obtained from a double-blind, parallel-arm, randomized study: 165 patients received an intravenous bolus of TXA 1 g followed by a continuous infusion of either placebo (group A) or TXA 1 g (group B) for 8 hours. A total of 811 TXA plasma concentrations were measured (see Figure A.4).

The structural model is the two-compartment PK model described in (10) and already used for the simulation study. A log-normal distribution is used for the four individual PK parameters. There are 12 individual covariates, including 11 physiological covariates (age, sex, height, weight, body mass index (BMI), body surface area (BSA), lean body weight (LBW), glomerular filtration rate (GFR), creatinine clearance (CrCl), CKD-EPI, Cockroft, and treatment group as an additional categorical covariate. All continuous covariates were log-transformed to account for linear relationships between log-parameters and log-covariates.

The initial model contains no relationship between parameters and covariates and no correlation between random effects. It then took only 3 iterations and 220” for SAMBA to converge and propose a statistical model for these data. The estimated parameters for this model are shown in Table A.1.

The covariate model shows an effect of weight on central volume, BSA on peripheral volume, GFR on clearance, and of LBW on intercompartmental clearance. In addition, the model indicates that the route of administration has an influence on peripheral volume and clearance.

Positive correlations are found between clearance and the two volumes. Note that the correlation model constructed necessarily has a block structure. The correlation between the two volumes ρ_{V_c,V_p} is therefore estimated, although it is not significant and can therefore be considered zero.

The value of the selection criterion BICc is 5560 for this selected model and 5874 for the initial model. Note that the model proposed in [28] contains only one relationship between Crcl and clearance and one between weight and central volume, along with a unique correlation between clearance and central volume. The value of BICc for this model is 5815.

Finding the ”best” model in terms of BICc does not necessarily mean that this model is a ”good” model for fitting these data. It is therefore important to validate this model choice by ensuring, on the one hand, that the various hypotheses assumed are not rejected and that the model has good predictive performance. In this regard, it is worth noting that all components of the selected model are statistically significant, which allows us to confirm the hypotheses established regarding the relationships between covariates and parameters, as well as between parameters. Visual predictive check (VPC) is displayed in Figure A.5. This diagnostic tool allows visual comparison of multiple quantiles of the empirical distribution of the data with prediction intervals estimated using Monte Carlo simulation [6]. Thus, we see that the observed data are quite consistent with the predictions of the model, which confirms the very good predictive capabilities of the model. In conclusion, there is no statistical reason to reject the model built by SAMBA.

References

[1] A. P. Dempster, N. M. Laird, D. B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, Journal of the Royal Statistical Society. Series B 39 (1) (1977) 1–38.

[2] G. J. McLachlan, T. Krishnan, The EM algorithm and extensions, Vol. 382, John Wiley & Sons, 2007.

[3] G. C. Wei, M. A. Tanner, A Monte Carlo implementation of the EM algorithm and the poor man’s data augmentation algorithms, Journal of the American statistical Association 85 (411) (1990) 699–704.

[4] B. Delyon, M. Lavielle, E. Moulines, Convergence of a stochastic approximation version of the EM algorithm, Annals of Statistics 27 (1) (1999) 94–128.

[5] E. Kuhn, M. Lavielle, Coupling a stochastic approximation version of EM with an MCMC procedure, ESAIM: Probability and Statistics 8 (2004) 115–131.

[6] M. Lavielle, Mixed effects models for the population approach: models, tasks, methods and tools, CRC press, 2014.

[7] T. Ando, Bayesian model selection and statistical modeling, CRC Press, 2010.

[8] K. P. Burnham, D. R. Anderson, Multimodel inference: understanding AIC and BIC in model selection, Sociological methods & research 33 (2) (2004) 261–304.

[9] D. Anderson, K. Burnham, Model selection and multi-model inference, Second. NY: Springer-Verlag 63 (2020) (2004) 10.

[10] G. Claeskens, N. L. Hjort, et al., Model selection and model averaging, Cambridge Books (2008).

[11] R. Tibshirani, Regression shrinkage and selection via the lasso, Journal of the Royal Statistical Society: Series B (Methodological) 58 (1) (1996) 267–288.

[12] P. Bruce, A. Bruce, Practical statistics for data scientists: 50 essential concepts, " O'Reilly Media, Inc.", 2017.

[13] B. Efron, T. Hastie, I. Johnstone, R. Tibshirani, Least angle regression, The Annals of statistics 32 (2) (2004) 407–499.

[14] A. E. Lamont, J. K. Vermunt, M. L. Van Horn, Regression mixture models: Does modeling the covariance between independent variables and latent classes improve the results?, Multivariate behavioral research 51 (1) (2016) 35–52.

[15] K.-J. Lee, M. Feldkircher, Y.-C. Chen, Variable selection in finite mixture of regression models with an unknown number of components, Computational Statistics & Data Analysis 158 (2021) 107180.

[16] A. Khalili, J. Chen, Variable selection in finite mixture of regression models, Journal of the american Statistical association 102 (479) (2007) 1025–1038.

[17] M. Prague, M. Lavielle, SAMBA: a novel method for fast automatic model building in nonlinear mixed-effects models, CPT: Pharmacometrics and Systems Pharmacology (in press).

[18] X. Bai, W. Yao, J. E. Boyer, Robust fitting of mixture regression models, Computational Statistics & Data Analysis 56 (7) (2012) 2347–2359.

[19] P. A. Naik, P. Shi, C.-L. Tsai, Extending the Akaike information criterion to mixture regression models, Journal of the American Statistical Association 102 (477) (2007) 244–254.

[20] J. C. Pinheiro, D. M. Bates, M. J. Lindstrom, Model building for nonlinear mixed effects models, University of Wisconsin, Department of Biostatistics Madison, WI, 1995.

[21] G. Ayral, J.-F. Si Abdallah, C. Magnard, J. Chauvin, A novel method based on unbiased correlations tests for covariate selection in nonlinear mixed effects models–the COSSAC approach, CPT: Pharmacometrics & Systems Pharmacology (2021).

[22] K. G. Kowalski, M. M. Hutmacher, Efficient screening of covariates in population models using Wald's approximation to the likelihood ratio test, Journal of pharmacokinetics and pharmacodynamics 28 (3) (2001) 253–275. doi:10.1023/a:1011579109640.

[23] M. Yuan, Z. Zhu, Y. Yang, M. Zhao, K. Sasser, H. Hamadeh, J. Pinheiro, X. S. Xu, Efficient algorithms for covariate analysis with dynamic data using nonlinear mixed-effects model, Statistical Methods in Medical Research 30 (1) (2021) 233–243. doi:10.1177/0962280220949898.

[24] M. Delattre, M. Lavielle, M.-A. Poursat, A note on BIC in mixed-effects models, Electronic journal of statistics 8 (1) (2014) 456–475.

[25] E. Kuhn, M. Lavielle, Maximum likelihood estimation in nonlinear mixed effects models, Computational statistics & data analysis 49 (4) (2005) 1020–1038.

[26] G. James, D. Witten, T. Hastie, R. Tibshirani, An introduction to statistical learning, Vol. 112, Springer, 2013.

[27] A. M. Variyath, A. Brobbey, Variable selection in multivariate multiple regression, Plos one 15 (7) (2020) e0236067.

[28] J. Lanoiselée, P. J. Zufferey, E. Ollier, S. Hodin, X. Delavenne, et al., Is tranexamic acid exposure related to blood loss in hip arthroplasty? A pharmacokinetic–pharmacodynamic study, British journal of clinical pharmacology 84 (2) (2018) 310–319.

Appendix A. Proof of Proposition 1

Using the fact that, for any models $\mathcal{M}$ and $\mathcal{M}'$ in $\mathbb{M}$,

$$\begin{aligned}\log\left(\mathrm{p}(y;\mathcal{M})\right) &= \mathbb{E}\left(\log\left(\mathrm{p}(y;\mathcal{M})\right)|y;\mathcal{M}'\right)\\ &= \mathbb{E}\left(\log\left(\mathrm{p}(y,\psi;\mathcal{M})\right)|y;\mathcal{M}'\right) - \mathbb{E}\left(\log\left(\mathrm{p}(y,\psi|\psi;\mathcal{M})\right)|y;\mathcal{M}'\right)\end{aligned}$$

we have that

$$\begin{aligned}&\mathrm{U}(\mathcal{M}_k,y) - \mathrm{U}(\mathcal{M}_{k-1},y)\\ =& -2\log\left(\mathrm{p}(y;\mathcal{M}_k)\right) + \mathrm{pen}(\mathcal{M}_k) + 2\log\left(\mathrm{p}(y;\mathcal{M}_{k-1})\right) - \mathrm{pen}(\mathcal{M}_{k-1})\\ =& -2\mathbb{E}\left(\log\left(\mathrm{p}(y,\psi;\mathcal{M}_k)\right)|y;\mathcal{M}_{k-1}\right) + 2\mathbb{E}\left(\log\left(\mathrm{p}(y,\psi|y;\mathcal{M}_k)\right)|y;\mathcal{M}_{k-1}\right)\\ &+ 2\mathbb{E}\left(\log\left(\mathrm{p}(y,\psi;\mathcal{M}_{k-1})\right)|y;\mathcal{M}_{k-1}\right) - 2\mathbb{E}\left(\log\left(\mathrm{p}(y,\psi|y;\mathcal{M}_{k-1})\right)|y;\mathcal{M}_{k-1}\right)\\ &+ \mathrm{pen}(\mathcal{M}_k) - \mathrm{pen}(\mathcal{M}_{k-1})\\ =& \,\mathbb{E}\left(V(y,\psi,\mathcal{M}_k)|y;\mathcal{M}_{k-1}\right) - \mathbb{E}\left(V(y,\psi,\mathcal{M}_{k-1})|y;\mathcal{M}_{k-1}\right)\\ &+ 2\mathbb{E}\left(\log\left(\mathrm{p}(y,\psi|y;\mathcal{M}_k)\right)|y;\mathcal{M}_{k-1}\right) - 2\mathbb{E}\left(\log\left(\mathrm{p}(y,\psi|y;\mathcal{M}_{k-1})\right)|y;\mathcal{M}_{k-1}\right)\end{aligned}$$

By construction,

$$\mathbb{E}\left(V(y,\psi,\mathcal{M}_k)|y;\mathcal{M}_{k-1}\right) \leq \mathbb{E}\left(V(y,\psi,\mathcal{M}_{k-1})|y;\mathcal{M}_{k-1}\right)$$

On the other hand,

$$\mathbb{E}\left(\log\left(\mathrm{p}(y,\psi|y;\mathcal{M}_{k-1})\right)|y;\mathcal{M}_{k-1}\right) - \mathbb{E}\left(\log\left(\mathrm{p}(y,\psi|y;\mathcal{M}_k)\right)|y;\mathcal{M}_{k-1}\right)$$

is a Kullback-Leibler divergence and is therefore positive (it is null if and only if $\mathcal{M}_k = \mathcal{M}_{k-1}$).

We deduce that $\mathrm{U}(\mathcal{M}_k,y) < \mathrm{U}(\mathcal{M}_{k-1},y)$ if $\mathcal{M}_k \neq \mathcal{M}_{k-1}$ and the sequence $(\mathrm{U}(\mathcal{M}_k,y))$ is a decreasing sequence. □

	value	s.e.	p.value
$V_{c,\text{pop}}$	6.36	0.18	
$V_{p,\text{pop}}$	10.78	0.28	
Q_{pop}	23.46	0.90	
Cl_{pop}	5.08	0.12	
$\beta_{V_c,\text{weight}}$	0.69	0.12	$< 10^{-5}$
$\beta_{V_p,\text{BSA}}$	1.30	0.15	$< 10^{-5}$
$\beta_{V_p,\text{group}}$	-0.30	0.05	$< 10^{-5}$
$\beta_{Q,\text{LBW}}$	0.91	0.16	$< 10^{-5}$
$\beta_{Cl,\text{GFR}}$	0.56	0.04	$< 10^{-5}$
$\beta_{Cl,\text{group}}$	0.12	0.03	$2\ 10^{-4}$
ω_{V_c}	0.32	0.02	
ω_{V_p}	0.17	0.03	
ω_{Q}	0.17	0.04	
ω_{Cl}	0.22	0.01	
$\rho_{V_c,Cl}$	0.45	0.08	$< 10^{-5}$
$\rho_{V_p,Cl}$	0.72	0.15	$< 10^{-5}$
ρ_{V_c,V_p}	0.04	0.19	0.80
b	0.11	0.00	

Table A.1: Estimated population parameters and their standard errors of the tranexamix PK model. The p-value is that of the t-test used to test if a parameter is null.

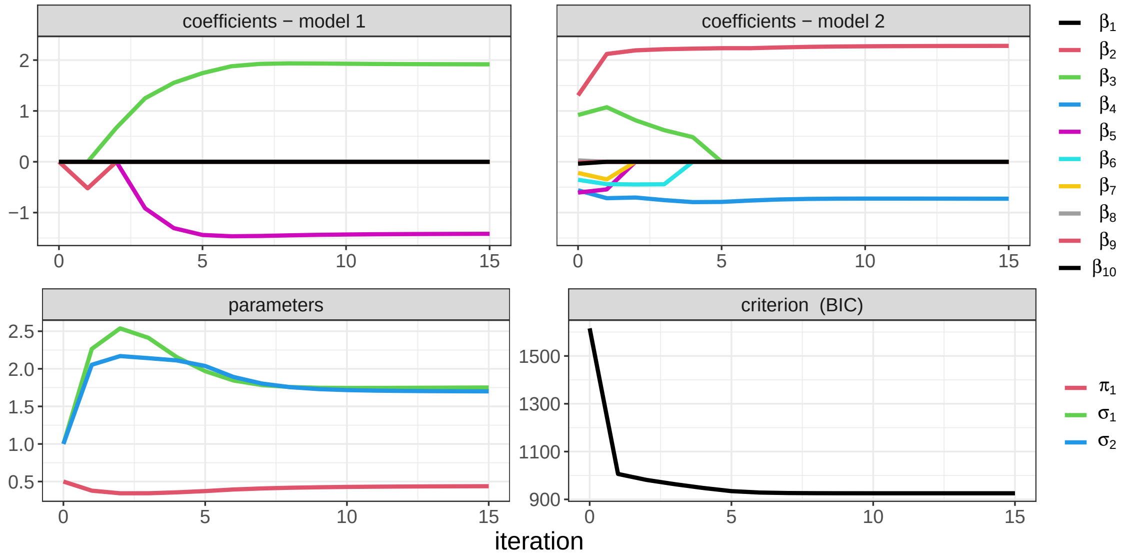

Figure A.1: Convergence of EM using BIC

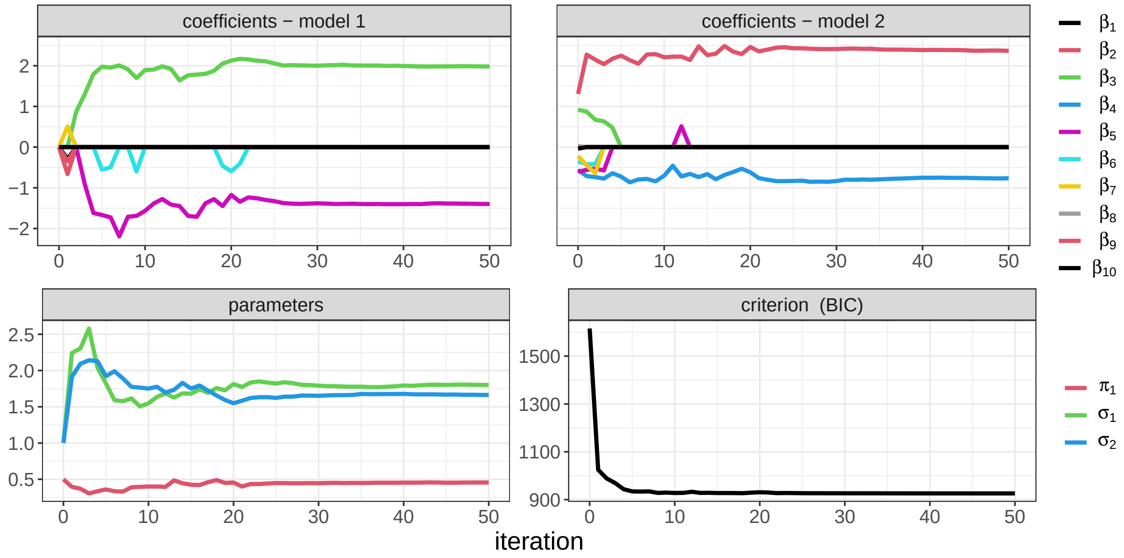

Figure A.2: Convergence of SAEM using BIC

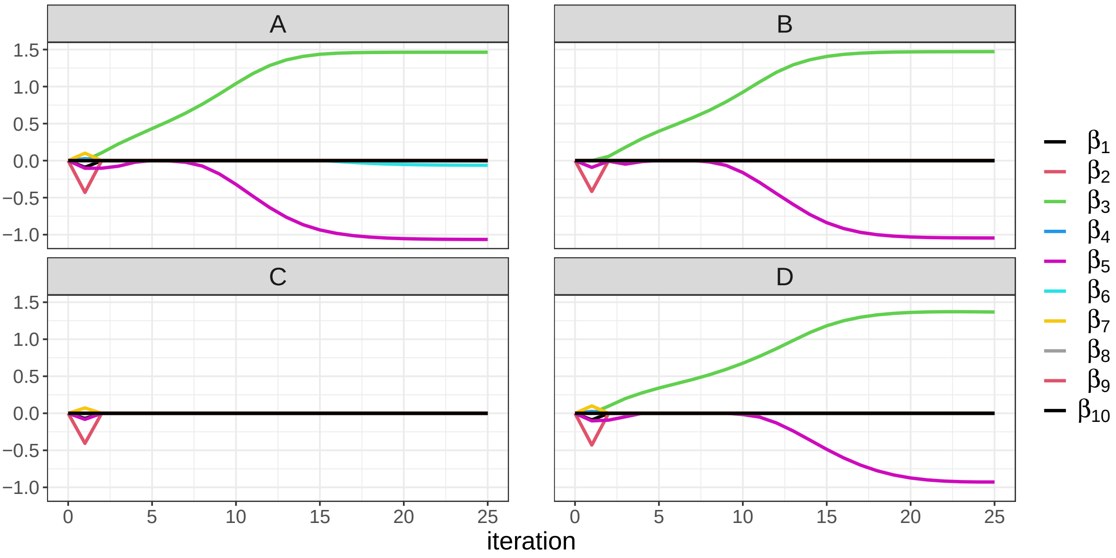

Figure A.3: Convergence of EM using lasso. A: with $\lambda = 15$; B: with $\lambda = 15$ and $\rho_{\min} = 0.2$; C: with $\lambda = 18$; D: with $\lambda = 15 + 3k/25$

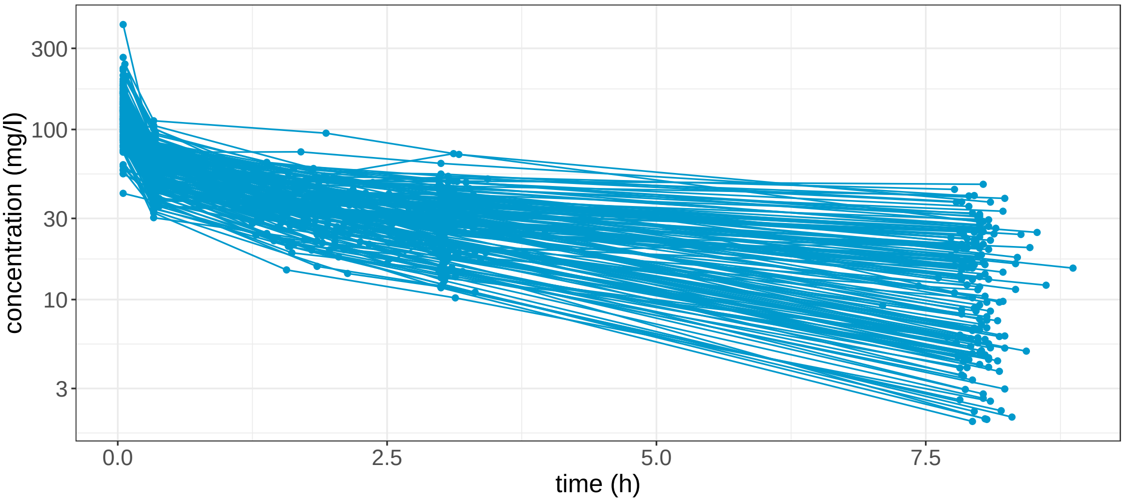

Figure A.4: tranexamic pharmacokinetic data

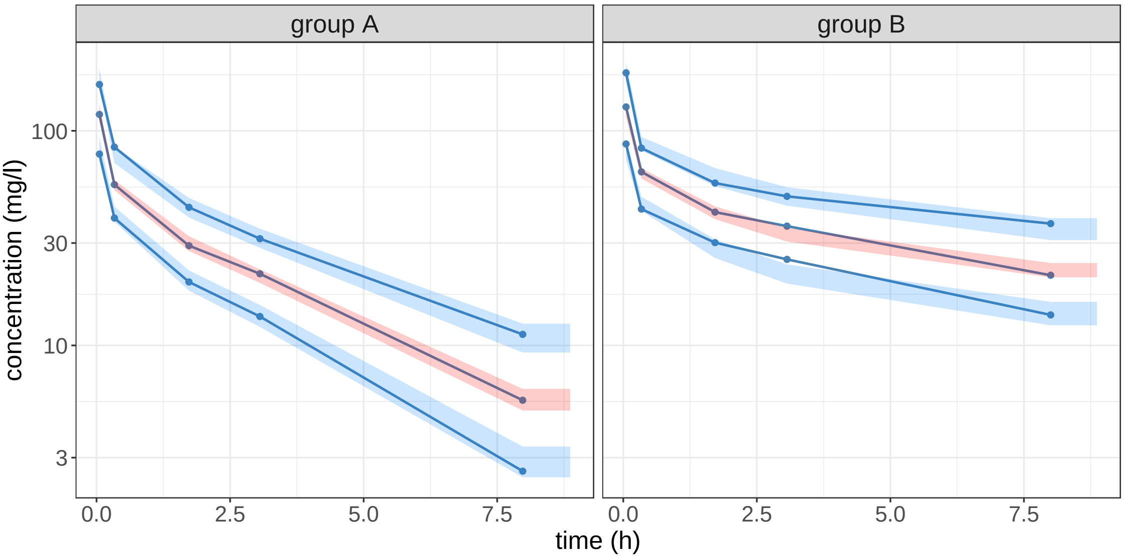

Figure A.5: Visual predictive check for the two treatment groups. Observed quantiles of order 10%, 50% and 90% are displayed (solid lines) with their respective prediction intervals estimated by Monte Carlo simulation under the model built by SAMBA.