Computer Methods and Programs in Biomedicine 162 (2018) 11–18

Contents lists available at ScienceDirect

Computer Methods and Programs in Biomedicine

journal homepage: www.elsevier.com/locate/cmpb

# Model selection for clustering of pharmacokinetic responses

Rui P. Guerra[a,c], Alexandra M. Carvalho[a,c,*], Paulo Mateus[b,c]

[a] Departamento de Engenharia Electrotécnica e de Computadores, Instituto Superior Técnico, ULisboa, Portugal
[b] Departamento de Matemática, Instituto Superior Técnico, ULisboa, Portugal
[c] Instituto de Telecomunicações, Av. Rovisco Pais, 1049-001, Lisboa, Portugal

ARTICLE INFO

*Article history:*
Received 13 December 2017
Revised 10 April 2018
Accepted 3 May 2018

*Keywords:*
Clustering
Model selection
Minimum description length
Normalised maximum likelihood
Pharmacokinetics

ABSTRACT

*Background and Objective*: Pharmacokinetics comprises the study of drug absorption, distribution, metabolism and excretion over time. Clinical pharmacokinetics, focusing on therapeutic management, offers important insights towards personalised medicine through the study of efficacy and toxicity of drug therapies. This study is hampered by subject's high variability in drug blood concentration, when starting a therapy with the same drug dosage. Clustering of pharmacokinetics responses has been addressed recently as a way to stratify subjects and provide different drug doses for each stratum. This clustering method, however, is not able to automatically determine the correct number of clusters, using an user-defined parameter for collapsing clusters that are closer than a given heuristic threshold. We aim to use information-theoretical approaches to address parameter-free model selection.

*Methods*: We propose two model selection criteria for clustering pharmacokinetics responses, founded on the Minimum Description Length and on the Normalised Maximum Likelihood.

*Results*: Experimental results show the ability of model selection schemes to unveil the correct number of clusters underlying the mixture of pharmacokinetics responses.

*Conclusions*: In this work we were able to devise two model selection criteria to determine the number of clusters in a mixture of pharmacokinetics curves, advancing over previous works. A cost-efficient parallel implementation in Java of the proposed method is publicly available for the community.



## 1. Introduction

Pharmacokinetics (PK) aims to study the evolution of drug concentration in a subject taking into account his individual patterns of drug absorption and elimination by the body [1]. Simple PK models consider that the entire human circulatory system is a single compartment with constant drug concentration in a single instant. In this model, drug elimination rate is assumed to be proportional to this concentration, and thus, the blood drug concentration per time instant (PK curve) fulfils a simple differential equation. Although simplistic, this mathematical model has been largely used with success in clinical practice [2–4].

Usually, PK is used for drug development and monitoring. With personalised medicine in mind, PK curves can also be used to adjust the drug dose to particular subjects taking into account group-dependant responses. Indeed, it is commonly the case that subjects starting with the same drug dosage present high variability in drug blood concentration, with a strong impact on drug efficacy and/or toxicity. In order to improve efficacy and diminish toxicity, subjects can be clustered in groups with similar responses and the treatment adjusted according to the average PK response of each group.

Unsupervised learning of PK curves has been proposed using an Expectation-Maximisation (EM) algorithm [5]. This method is a particular case of the nonlinear mixed-effects model proposed by Azzimonti et al. [6], where cluster-specific error variances were considered. However, the number of clusters elicited by these methods is heuristic, depending highly on user-defined parameters that avoid low-weight clusters besides merging similar ones. Finding the optimal number of clusters of an EM finite mixture is a model-selection problem, with both deterministic and stochastic solutions. Deterministic methods consider a finite range for the number of clusters $M$, from $M_{\min}$ to $M_{\max}$, and evaluate each candidate through a model selection criterion, usually a score accounting with the maximum likelihood of the model with a penalty factor. These deterministic approaches include the Laplace-Empirical Criterion [7,8], the Bayesian Information Criterion [9] and the Minimum Description Length [10], among others [11–16]. Stochastic ap-

* Corresponding author at: Instituto de Telecomunicações, Av. Rovisco Pais, 1049-001, Lisboa, Portugal.
*E-mail address:* alexandra.carvalho@tecnico.ulisboa.pt (A.M. Carvalho).

https://doi.org/10.1016/j.cmpb.2018.05.002
0169-2607/

proaches, like Markov chain Monte Carlo [17], resampling methods [18] and cross-validation approaches [19], can also be used as a model-selection criterion, incurring, however, in a high computation load when compared with deterministic ones.

In this paper, we consider two deterministic methods for model selection of mixtures of PK curves. Taking into account the PK model, we adapt two penalisation factors based on the Minimum Description Length (MDL) and the Normalised Maximum Likelihood (NML) [20,21] for mixtures of Gaussians. In this way, we are able to elicit the number of clusters with informational-theoretically considerations, improving in this way previous results [5,6]. As a benefit, the proposed model-selection criteria also offer the advantage of avoiding data overfitting. From the implementation point of view, as random initialisations required by the EM algorithm are embarrassingly parallel, we propose an algorithm that distributes the initialisations through the available cores in a cost-efficient manner. The source code is freely available at a GitHub repository in [22], together with a user manual and data used in the experiments.

The paper is organised as follows. Section 2 describes related work needed to understand the proposed method; namely, the one-compartment model, clustering of PK drug responses and model selection via MDL and NML. Section 3 presents the proposed criteria, whose implementation and experimental results are discussed in Sections 4 and 5. Finally, in Section 7 we draw some conclusions.

## 2. Background

In this section we explain basic PK concepts and describe existing solutions regarding PK clustering responses. In the end, the MDL and NML model selection criteria are presented in their general formulation.

### 2.1. Pharmacokinetic models

In order to describe and predict the effect that a drug has on a subject, it is necessary to consider a simplified representation of the human body, so-called PK model [23].

One representation commonly used in practice is the one-compartment model. In this very simple case, the entire human circulatory system is considered as a single compartment with a constant volume $V$, usually measured in litres, and a time-variant quantity of drug $Q(t)$ within that volume, measured in milligrams. This time variance is caused either by absorption or elimination of the drug from the body. These processes are ruled by two constants that depend on the subject, namely the absorption rate constant ($k_a$) and the elimination rate constant ($k_e$). Considering the absorption of the drug by the body as a function of time $I(t)$, with initial condition $I(0)$ given by $I(0) = Dose \times F$, where $Dose$ is the initial dosage and $F$ is a constant related to the bioavailability of the subject, such that $I'(t) = -k_a I(t)$. It is possible to write a differential equation representing the quantity of drug in the body given by

$$Q'(t) = -k_e Q(t) + k_a I(t). \tag{1}$$

The concentration $C(t)$ of the drug along time in the single compartment is used to more consistently compare the reaction of the drug in different subjects. This can simply be computed using the expression

$$C(t) = \frac{Q(t)}{V}. \tag{2}$$

### 2.2. Unsupervised learning of PK curves

An Expectation-Maximisation (EM) algorithm was recently proposed for clustering PK responses [5]. To introduce notation, we sketch this algorithm in what follows.

By solving the system of equations given by (1) and (2), the expression for the drug concentration is given by

$$C(t) = \alpha(e^{-\beta_1 t} - e^{-\beta_2 t}), \tag{3}$$

where

$$\alpha = \frac{k_a Dose \times F}{V(k_a - k_e)},\ \beta_1 = k_e \text{ and } \beta_2 = k_a. \tag{4}$$

The variables $\alpha$, $\beta_1$ and $\beta_2$ are the free parameters of the PK curves, modelling the drug responses of the subjects. Subjects are then grouped into clusters, depending on their responses. The drug concentration over time for cluster $l$ is described by

$$C_l(t) = \alpha_l(e^{-\beta_{1l} t} - e^{-\beta_{2l} t}). \tag{5}$$

Each subject $i$, belonging to cluster $l$, has a concentration $y_{il}$ at time $j$ given by

$$y_{il} = C_l(t_j) + \epsilon_{ijl}, \tag{6}$$

where $\epsilon_{ijl}$ is Gaussian error with zero mean and variance $\upsilon_l$.

Two assumptions are made to establish the EM algorithm. First, measurements over all $N$ subjects are performed over the same $n$ time instants $\boldsymbol{t} = (t_1, \ldots, t_n)$. Second, measurement errors at different time instants are independent. In this case, the probability density function of $\boldsymbol{y}_i = (y_{i1}, \ldots, y_{in})$ for subject $i$ belonging to cluster $l$ is given by

$$p_l(\boldsymbol{y}_i) = \frac{1}{(2\pi\upsilon_l)^{\frac{n}{2}}} e^{\frac{-1}{2\upsilon_l}\sum_{j=1}^{n}(y_{ij} - C_l(t_j))^2}. \tag{7}$$

Let $\omega_l$ be the probability of a subject belonging to cluster $l$. In addition, denote by $\boldsymbol{W} = (W_1, \ldots, W_N)$ the random vector where each $W_i$ describes the cluster to whom subject $i$ belongs; in this case, we have

$$P(W_i = l) = \omega_l \text{ for } 1 \leq l \leq M, \tag{8}$$

where $M$ is the number of clusters.

Under the previous assumptions, the EM algorithm estimates the parameters

$$\boldsymbol{\theta} = \{\alpha_l, \beta_{1l}, \beta_{2l}, \upsilon_l, \omega_l\}_{l \in 1,\ldots,M} \tag{9}$$

that best fit the observed data $Y = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N)$ of size $N \times n$, corresponding to the responses of $N$ subjects during $n$ sampling instants.

Let $\boldsymbol{w}$ be a realisation of the random vector $\boldsymbol{W}$, that is, $\boldsymbol{w} = (l_1, \ldots, l_N)$ where $l_i$ is the cluster to whom subject $i$ belongs to. The working hypothesis $H = (\boldsymbol{w}, \boldsymbol{\theta})$ allows to write the probability of observing data $Y$ as

$$p(Y|H) = p(Y|\boldsymbol{w}, \boldsymbol{\theta}) = \prod_{i=1}^{N} \omega_{li} p_l(\boldsymbol{y}_i). \tag{10}$$

The EM algorithm is an iterative process divided in two steps: the Expectation (E) step and the Maximisation (M) step [24]. The E step consists of computing an objective function $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(k)})$ whose maximisation corresponds to the maximisation of the likelihood of the data. This function is defined as

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(k)}) = \sum_{l=1}^{M}\sum_{i=1}^{N} X_{il}^{(k)} \log(\omega_l p_l(\boldsymbol{y}_i)), \tag{11}$$

where $X_{il}^{(k)}$ is the probability of observation $\boldsymbol{y}_i$ be described by cluster $l$ in the $k$-th step, given by

$$X_{il}^{(k)} = \frac{\omega_l^{(k)} p_l^{(k)}(\boldsymbol{y}_i)}{\sum_{r=1}^{M} \omega_r^{(k)} p_r^{(k)}(\boldsymbol{y}_i)}. \tag{12}$$

The M step consists in finding a new set of parameters $\boldsymbol{\theta}^{(k+1)}$ that maximizes the objective function given by Eq. (11). The weight and variance parameters, $\omega_l$ and $\upsilon_l$, can be maximised based on the canonical expressions for Gaussian mixtures. However, the model parameters $\alpha_l$, $\beta_{1l}$ and $\beta_{2l}$ are more complex to maximise, requiring numerical methods. We refer the interested reader to Tomás et al. [5].

One of the main drawbacks of the EM algorithm is the requirement to fix the number of clusters $M$ a priori. This shortcoming was previously approached in [5] by disregarding negligible clusters and merging similar ones. However, there is no canonical notion of similarity among clusters, and so, the elicited number of clusters highly depends on user-defined parameters. Model selection provides a sound alternative justified at the light of information theory.

### 2.3. Model selection

The MDL principle is a model selection method proposed by Rissanen [25] and has been successfully used in a variety of learning tasks [26,27]. It states that the best description of the data is the one that manages to compress it the most, according to its regularity. This is what MDL strives for, finding an hypothesis to explain the data that is at the same time, as simple as possible, and compresses the data, as much as possible.

The first and simplest implementation of the MDL principle is one that divides an objective function in two parts [28], crudely given by

$$L(H) + L(Y|H)\,, \tag{13}$$

where $L(H)$ corresponds to the length, in bits, needed to describe an hypothesis $H$ and $L(Y|H)$ is the length, in bits, of the description of data $Y$ according to hypothesis $H$.

In order to define the second part of the expression, $L(Y|H)$, it is possible to use the fact that the hypothesis $H$ defines a probability distribution for the data. Considering that the length is measured in bits, the codelength of data $Y$ while using hypothesis $H$ can be given by the *Shannon-Fano* code [29], and so it is possible to write

$$L(Y|H) = -\log p(Y|H)\,, \tag{14}$$

where $p(Y|H)$ is the probability density of data $Y$ according to hypothesis $H$. This expression shows the parallelism between finding the shortest length code and finding the distribution with the highest log-likelihood.

It is visible that the other part of the code, $L(H)$, depends only on the hypothesis at hand and not on the observed data. An asymptotical result derived by Rissanen [30,31], which coincides with a Bayesian criterion proposed by Schwarz [32], known as the *Bayesian Information Criterion*, takes $L(H)$ to represent the extra amount of bits needed to describe the data given as a function of the number of parameters. This term depends on the number of observations $N$ and the total number of parameters in the model $K$, as well as, the number of observations $h_\ell$ that belongs to each cluster $\ell$. For $M$ clusters, $L(H)$ is given by

$$L(H) = \frac{1}{2}\sum_{\ell=1}^{M} \log h_\ell + \frac{1}{2} K \log N\,. \tag{15}$$

The first term of this expression has commonly a small weight, therefore, it is often disregarded for the sake of simplicity.

An alternative method to define the model complexity $L(H)$ is by using the NML codelength [33]. For this purpose we recall the concept of the regret [34] of a model for a set of hypothesis $\mathcal{H}$.

In terms of codelengths, the regret of distribution $p$ against a set of hypothesis $\mathcal{H}$ can be seen as the additional length, in bits, needed to encode the data $Y$ using the code associated with $p$, in comparison with the bits needed by the "optimal" distribution within $\mathcal{H}$. Given data $Y$, the optimal distribution in $\mathcal{H}$, denoted by $\hat{H}$, corresponds to the distribution that maximises the likelihood of the data. A good measure to check the quality of a model against the set of hypothesis $\mathcal{H}$ consists in checking its regret in the worst case. This measure resumes to find the maximum possible regret for all data of fixed size $N$. Accordingly to the NML principle, the best probability distribution $p_{\text{NML}}$ is the one that minimises the maximum regret over all possible data $Y$, of a fixed size $N$, as in

$$p_{\text{NML}}(\cdot|\mathcal{H}) = \min_{p}\max_{Y}(-\log p(Y) + \log p(Y|\hat{H}))\,. \tag{16}$$

The solution to this minimax problem is achieved by the NML distribution, also known as the Shtarkov distribution [35], given by

$$p_{\text{NML}}(Y|\mathcal{H}) = \frac{p(Y|\hat{H})}{\mathcal{C}(\mathcal{H}, N)}\,, \tag{17}$$

where the denominator for the continuous case can be computed as

$$\mathcal{C}(\mathcal{H}, N) = \int_Y p(Y|\hat{H})dY. \tag{18}$$

We recall that the integration in Eq. (18) ranges over all data $Y$ of size $N$, and moreover, that the NML distribution given by Eq. (17) does not need to belong to $\mathcal{H}$.

By applying a logarithm to the NML distribution, it is possible to obtain the NML codelength, or stochastic complexity, given by

$$-\log p_{\text{NML}}(Y|\mathcal{H}) = -\log p(Y|\hat{H}) + \log \mathcal{C}(\mathcal{H}, N)\,. \tag{19}$$

From this expression it is easy to understand that the first term corresponds, again, to the goodness-of-fit term (the symmetric of the log-likelihood), which is similar to the MDL case, and the second term corresponds to the complexity of the class of models in $\mathcal{H}$, so-called the parametric complexity.

The value of the parametric complexity can be difficult to compute for certain probability distributions. However, for Gaussian mixture models, an expression has been derived in [36] that can be adapted for mixtures of PK responses. We address this issue in the next section.

## 3. Model selection for mixtures of PK curves

To address model selection for mixtures of PK curves using the EM algorithm described in Section 2.2, two penalisations were derived: one using using the MDL principle and another with the NML codelength, both described in Section 2.3.

Adapting the MDL principle is relatively simple, since the goodness-of-fit term corresponds to maximising the likelihood of the data, which is given by Eq. (10). In information-theoretical terms, this is equivalent to minimising the (symmetric of the) log-likelihood of the data, whose maxima coincides with that of $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(k)})$, given by Eq. (11). Moreover, the model complexity term is given by Eq. (15); following the canons of the literature, only the second term will be used. In this case, the number of degrees of freedom of the model is five times $M$ minus one. This follows from the five cluster parameters ($\alpha$, $\beta_1$, $\beta_2$, $\upsilon$ and $\omega$) and from the fact that one cluster weight $\omega$ is linearly dependant on the others.

Thus, the modified EM algorithm with MDL model-selection scheme is set to optimise the following expression

$$-\log(p(Y|\boldsymbol{w}, \boldsymbol{\theta})) + \frac{1}{2}\log(N)(5M - 1)\,. \tag{20}$$

Note that, since the penalisation does not depend on the fitted parameters $\boldsymbol{w}$ and $\boldsymbol{\theta}$, the EM algorithm runs for the number of clusters $M$, ranging within $M_{\min}$ and $M_{\max}$, and then it elicits the number of clusters $M_{\text{MDL}}$ that minimise Eq. (20).

The NML codelength for multidimensional Gaussian mixtures with location vector $\vec{\mu}$ and covariance matrix $\Sigma$ was recently derived in [36]. Given the number of clusters $M$ and the data size $N$, the parametric complexity for such models is given by

$$\mathcal{C}(\mathcal{H}(M), N) = \sum_{h_1+\cdots+h_M=N} \frac{N!}{h_1!\cdots h_M!} \times \prod_{\ell=1}^{M} \left(\frac{h_\ell}{N}\right)^{h_\ell} \times I(h_\ell)\,, \quad (21)$$

where

$$I(h_\ell) = B(n, \lambda_{min}, R)\left(\frac{h_\ell}{2e}\right)^{\frac{nh_\ell}{2}} \frac{1}{\Gamma_m\left(\frac{h_\ell-1}{2}\right)}\,, \quad (22)$$

$$B(n, \lambda_{min}, R) = \frac{2^{n+1}R^{\frac{n}{2}} \prod_{j=1}^{n} \lambda_{min}^{(j)\,-\frac{n}{2}}}{n^{n+1}\Gamma\left(\frac{n}{2}\right)}\,, \quad (23)$$

$\lambda_{\text{min}}^{(j)}$ is a lower bound for the $j$th eigenvalue of the covariance matrix $\Sigma$, $R$ is an upper bound on the square of the norm of the location vector $\vec{\mu}$, $n$ is the dimension of the Gaussian distribution, $\Gamma$ is the Gamma function, and $\Gamma_m$ is the multivariate Gamma function [37] given by

$$\Gamma_m(x) = \pi^{\frac{n(n-1)}{4}} \prod_{j=1}^{n} \Gamma\left(x + \frac{1-j}{2}\right). \quad (24)$$

The case of PK responses is simpler than the multivariate case described above. The consequence is that, by adapting the above formulation, we obtain an upper-bound for the NML codelength of PK responses.

Concretely, each subject has $n$ independent measurements (over time) of drug concentrations, corresponding to a vector of Gaussian variables. Due to independence of each measurement, the covariance matrix for cluster $l$ is a diagonal matrix with constant value $\lambda = v_l$, the variance of cluster $l$. Moreover, a lower bound to $\lambda_{\text{min}}$ can be obtained from the precision of the concentrations stored in the data, in other words, the concentrations should have been collected with a device with error at most $\pm\sqrt{\lambda_{\text{min}}}$. So, Eq. (23) can be simplified to

$$B(n, \lambda_{min}, R) = \frac{2^{n+1}R^{\frac{n}{2}} \lambda_{min}^{-\frac{n^2}{2}}}{n^{n+1}\Gamma\left(\frac{n}{2}\right)}\,. \quad (25)$$

Finally, $R$ is upper-bounded by $n \times C_{\text{max}}$ where $C_{\text{max}}$ is a maximum concentration of the drug in the blood, which is also a physical constant that depends on the drug. We stress that, although both $\lambda_{\text{min}}$ and $R$ are parameters required to compute the NML, they are not user-defined parameters, or heuristics, they have concrete physical meaning and can be extrapolated by knowing how the data was collected and what kind of drug is being measured.

Even considering the simplifications above, the expression in Eq. (21) can still be quite difficult to compute. However, this computation can be simplified by performing a recursive algorithm described in Algorithm 1, as suggested in [36], with a computational time complexity of $\mathcal{O}(N^2 \times M)$.

**Algorithm 1** NML parametric complexity.

1: Set $\mathcal{C}(\mathcal{H}(M), 0) = 1$;
2: Compute $\mathcal{C}(\mathcal{H}(1), j) = I(j)$ for $j = 1, \ldots, N$;
3: **for** $k = 2$ to $M$ **do**
4: **for** $j = 1$ to $N$ **do**
5: Compute $\mathcal{C}(\mathcal{H}(k), j) = \sum_{r_1+r_2=j} \binom{j}{r_1} \left(\frac{r_1}{j}\right)^{r_1} \left(\frac{r_2}{j}\right)^{r_2} \times \mathcal{C}(\mathcal{H}(k-1), r_1) I(r_2)$;
6: **end for**
7: **end for**

Using this algorithm it is possible to immediately obtain the parametric complexity term for every possible number of clusters $M$ by using the value stored in $\mathcal{C}(\mathcal{H}(M), N)$.

The resulting expression that is used for performing model selection based on the NML codelength is given by

$$-\log(p(Y|\boldsymbol{w}, \boldsymbol{\theta})) + \log \mathcal{C}(\mathcal{H}(M), N)\,, \quad (26)$$

and it can be optimised ranging over the number of clusters, from $M_{\text{min}}$ to $M_{\text{max}}$, similarly to the MDL case.

## 4. Implementation details

In this section, we describe the overall EM procedure to cluster PK responses using the proposed model-selection schemes described in the previous section.

For convenience, we let the number of random initialisations be a user-defined parameter. In addition, it is up to the user to define the minimum and maximum number of clusters ($M_{\text{min}}$ and $M_{\text{max}}$), and the scoring criterion for the learning procedure (MDL or NML).

The pseudocode presented in Algorithm 2 summarises the execution of a single random initialisation for each possible numbers of clusters $M$, from $M_{\text{min}}$ to $M_{\text{max}}$. The *score* refers either to MDL or NML.

**Algorithm 2** Expectation-Maximisation algorithm.

1: **function** RUNEM
2: **for** each possible number of clusters $M$ from $M_{\text{min}}$ to $M_{\text{max}}$ **do**
3: randomly generate initial cluster parameters for $M$ clusters;
4: compute concentration of each cluster;
5: compute log-likelihood of each subject data to belong to each cluster;
6: compute degree of belonging of each subject to each cluster;
7: update cluster weights $\omega$ and variances $\upsilon$ for each cluster;
8: **while** maximum iterations is not reached and algorithm has not converged **do**
9: update cluster parameters $\alpha$, $\beta_1$ and $\beta_2$ for each cluster;
10: compute concentration of each cluster;
11: compute log-likelihood of each subject data to belong to each cluster;
12: compute degree of belonging of each subject to each cluster;
13: update cluster weights $\omega$ and variances $\upsilon$ for each cluster;
14: **end while**
15: assign each subject to the cluster with the highest probability of belonging;
16: compute *score* value;
17: **if** *score* is higher than current best *score* value **then**
18: save clustering output;
19: **end if**
20: **end for**
21: return best clustering output;
22: **end function**

In order to improve performance we considered a parallel implementation. The parallelisation method used in the current implementation of the program is the one that divides the load such that each processor takes care of approximately the same number of random initialisations for each possible numbers of clusters. With this process, it is possible to guarantee the best possible

**Table 1**
Detailed description of the 12 synthetic datasets. The values described in the Scenario column are the number of observations that belong to each of the generated clusters; for instance, in the imbalanced data number 7, the first cluster contains 50 subjects, whereas the second and third clusters contain each only 5 subjects. The within-group variance is described by the *Sum of Squares within* (SSw) normalised by the number $N \times n$ of generated concentrations.

| Number | $M$ | Balance | Scenario | Variance | Standard deviation | $\frac{SSw}{N \times n}$ |
|---|---|---|---|---|---|---|
| 1 | 3 | Yes | 20-20-20 | Low | 0.3-0.6-0.2 | 0.15 |
| 2 | 4 | Yes | 15-15-15-15 | Low | 0.3-0.3-0.3-0.3 | 0.07 |
| 3 | 5 | Yes | 12-12-12-12-12 | Low | 0.3-0.3-0.3-0.3-0.3 | 0.08 |
| 4 | 3 | Yes | 20-20-20 | High | 1.3-1.1-0.9 | 1.27 |
| 5 | 4 | Yes | 15-15-15-15 | High | 0.6-0.6-0.6-0.6 | 0.28 |
| 6 | 5 | Yes | 12-12-12-12-12 | High | 0.6-0.6-0.6-0.6-0.6 | 0.28 |
| 7 | 3 | No | 50-5-5 | Low | 0.3-0.6-0.2 | 0.10 |
| 8 | 4 | No | 45-5-5-5 | Low | 0.3-0.3-0.3-0.3 | 0.07 |
| 9 | 5 | No | 40-5-5-5-5 | Low | 0.3-0.3-0.3-0.3-0.3 | 0.07 |
| 10 | 3 | No | 50-5-5 | High | 1.3-1.1-0.9 | 0.93 |
| 11 | 4 | No | 45-5-5-5 | High | 0.6-0.6-0.6-0.6 | 0.30 |
| 12 | 5 | No | 40-5-5-5-5 | High | 0.6-0.6-0.6-0.6-0.6 | 0.31 |

**Table 2**
Scores of a single initialisation using the input data 3 from Table 1.

| No. of clusters | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| MDL | 1472 | 1093 | 884 | 829 | **211** | 621 | 642 | 927 | 680 | 303 |
| NML | 1536 | 1190 | 996 | 766 | **333** | 769 | 413 | 455 | 406 | 390 |

cost efficiency [38], with the load balance being perfect if the user-defined number of random initialisations is divisible by the number of available processors. In this case, a computer with $p$ processors is able to run the program approximately $p$ times faster than if it was run sequentially.

An implementation using Java is publicly available at the GitHub repository in [22]. A graphical user interface, with an application user guide, is available therein, along with the synthetic data used in the experiments.

## 5. Results

In order to confirm that the program performs correctly and produces consistent results, we conducted a series of tests with varied input data. This section aims at describing these results.

### 5.1. Synthetic data

Synthetic data was generated using 60 subjects, each being sampled at the same eight time instants. Different number of clusters and balanced scenarios were considered; each cluster with its own error variance. These datasets, available at the GitHub repository [22], are summarised in Table 1 and depicted in Fig. 1. Clusters are presented in different colours, each line representing a subject.

The program was run using 200 random initialisations with $M_{\min} = 1$ and $M_{\max} = 10$. The maximum number of clusters $M_{\max}$ was chosen to be higher than the probable number of clusters in the data. The number of random initialisations was chosen as a lower bound that was consistently able to find the correct solution. Both scoring schemes, MDL and NML, were used. We conclude that the proposed algorithm performed well, being able to obtain the correct cluster functions and correctly assign the subjects to the original clusters from where they were generated, even in the difficult scenarios of imbalanced and high variance data (datasets 10–12 in Table 1).

To exemplify that the program makes the correct decisions while comparing different outputs, Table 2 shows as an example the resulting scores of a single initialisation of the algorithm applied to the dataset number 3 (see Table 1), using both the MDL and NML criteria.

**Table 3**
Execution times for different numbers of processors $p$ using the dataset 3 from Table 1, with both MDL and NML criteria.

| $p$ | MDL Time [s] | NML Time [s] |
|---|---|---|
| 4 | 80.219 | 90.233 |
| 2 | 161.358 | 170.751 |
| 1 | 272.366 | 277.542 |

#### 5.1.1. Behaviour across initializations

In addition to correct clusters retrieval, we wanted to study the number of times the proposed algorithm converge to the correct number of clusters among all random initialisations. For this, we used 10 synthetic datasets with 3 clusters, all with the same cluster parameters, except for the cluster-specific variances. Results are presented in Fig. 2.

We noted that small standard deviations cause the proposed EM algorithm to elicit several single-subject clusters. Nevertheless, even when there is a small numbers of initialisations that converge to the true clusters, the algorithm still converges to the correct output, empirically attesting for the good behaviour of the scoring schemes. With higher values for the standard deviation almost half of the initialisations converge to the correct number of clusters. Comparing the behaviour of the proposed scoring criteria, NML seems to be slightly superior, although both MDL and NML have similar patterns in unveiling the correct number of clusters in the data throughout the 200 initialisations.

#### 5.1.2. Cost-efficient implementation

To empirically validate the cost efficiency of the implemented algorithm, the execution times, using different numbers of processors $p$, for both MDL and NML criteria, are shown in Table 3 for the dataset 3 described in Table 1. It is visible that the execution time approximately doubles as the number of processors are cut in half; similar results were obtained when four processors were used.

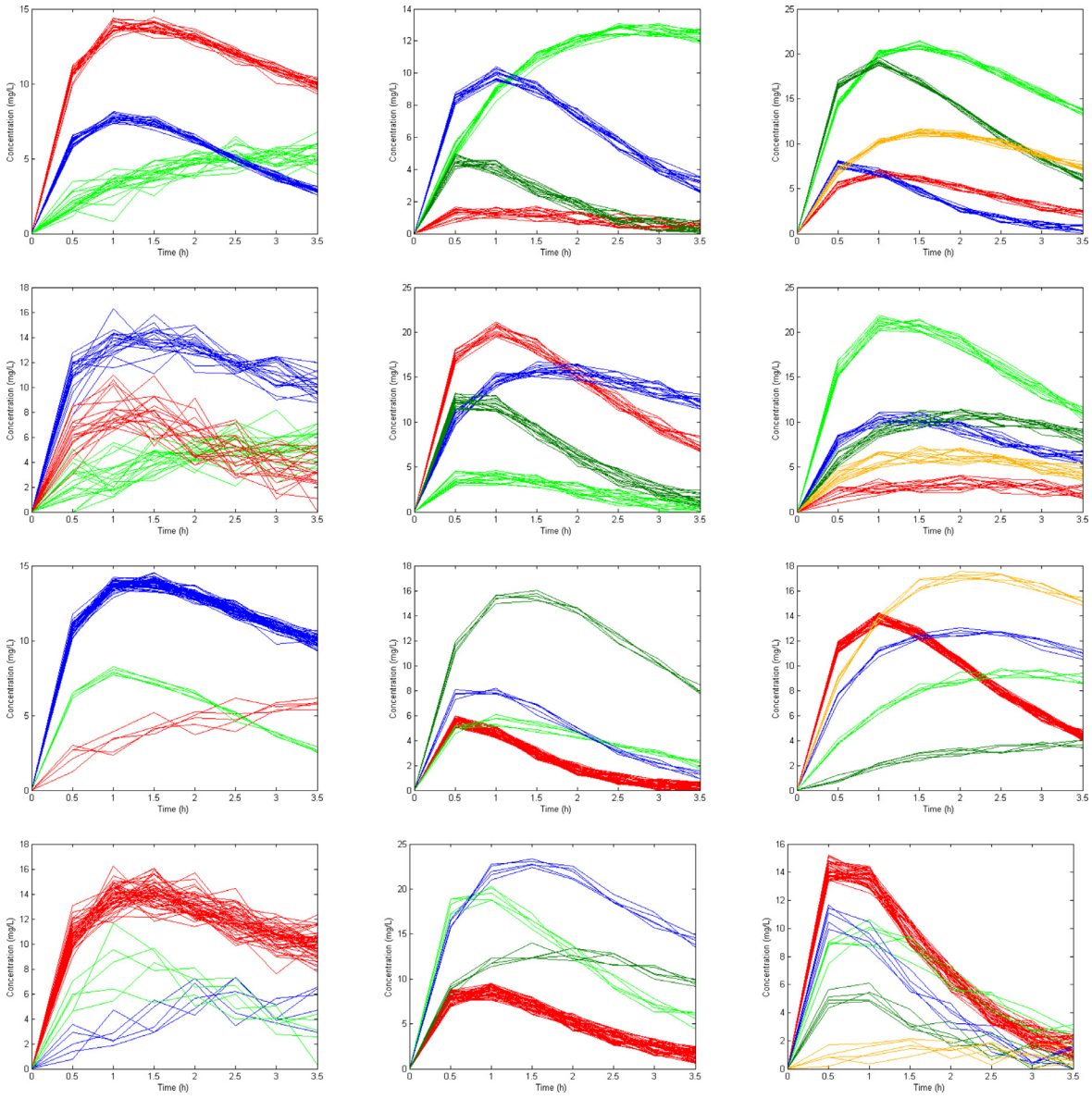


**Fig. 1.** Synthetic datasets summarised in Table 1, ordered from top-left to bottom-right. Each cluster is in a different color and each line represents a subject.

### 5.2. Real data

To test the algorithm in a real scenario, a dataset with blood concentrations of an anti-asthmatic drug was used. The clinical data is available in R [39], where twelve patients were given oral doses of the theophylline drug and had their serum measured over the following 25 h (at 11 time instants). The data is shown on Fig. 3 (left). The PK curves were unveiled by running the algorithm with exactly the same conditions as for the synthetic data: 200 random initializations, with $M_{\min} = 1$ and $M_{\max} = 10$, for both MDL and NML criteria. The algorithm found six different clusters, as color-coded in Fig. 3 (right).

Six clusters from a dataset of 12 subjects might seem excessive. Our interpretation is that this is precisely due to the small sample size, as the influence of the parametric complexity terms is directly related to the number of subjects $N$. Nonetheless, the output should be interpreted by experts with domain knowledge about the drug under study. After that, we should be able to relate subject features with the elicited groups and predict to which cluster a new subject belongs.

## 6. Discussion

In pharmokinetics (PK), the time course of drug concentrations in the body is usually described with compartment models [40–42]. Such models define functions of the drug concentration through time among patients. Each compartment represents a group of similar tissues, an organ or a fluid.

Drug concentration in the compartments is fitted mostly using nonlinear mixed effects (NLME) models [43]; data is measured usually in easily sampled fluids, like blood and urine. NLME statistical models contain both fixed (entire population) and random (subject specific) effects. Their main shortcoming is the hardness of integrating out random effects to compute maximum likelihood estimates. Expectation-Maximisation (EM) and stochastic approximations [44–46] are known techniques to overcome this issue, which can be found, for instance, in the Monolix software packages [47].

The above-mentioned methods ignore the subject-specific effects, and extract a single cluster only with average population effects. Meanwhile, an EM method considering both fixed and random effects with Gaussian noise to the measurements was pro-

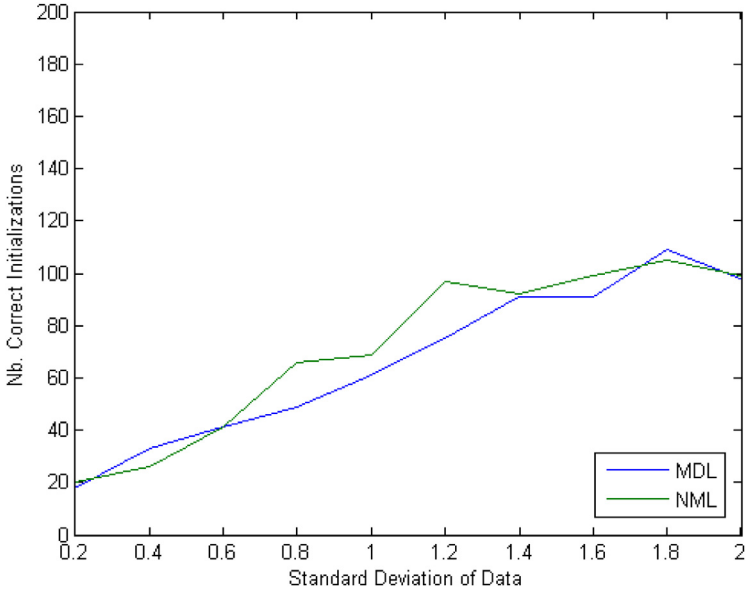


**Fig. 2.** Number of random initialisations out of 200 where the algorithm found the correct number of clusters for different data standard deviations.

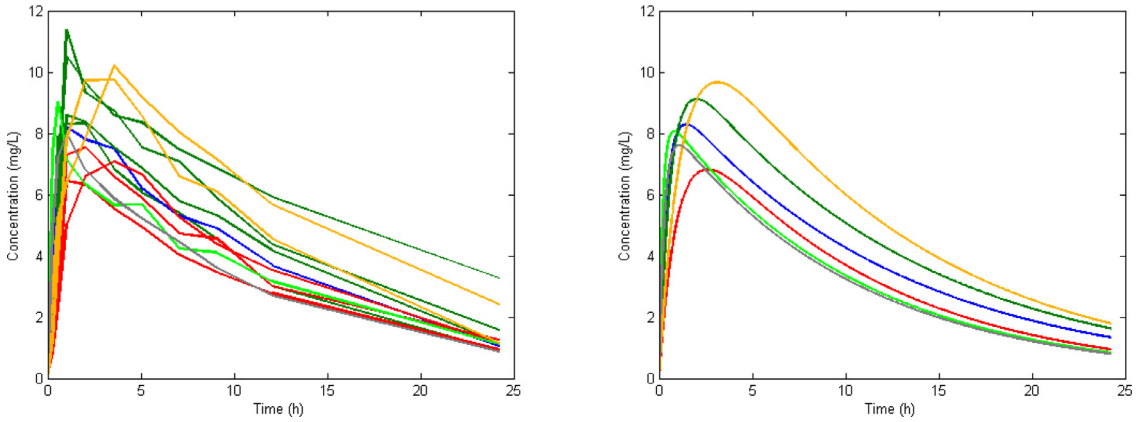


**Fig. 3.** On the left, the real data from the theophylline drug administration. On the right, the clusters elicited with the proposed algorithm; both MDL and NML criteria resulted in the same clusters.

posed [6]. As a particular case, subject-specific groups of PK curves has been addressed, unravelling clusters of subjects with similar drug responses [5]. The envisaged application of this method is to allow personalised medicine, delineating a treatment for each cluster.

A drawback of the current available methods is that they are not parameters-free, and the clusters unveiled depend highly upon user-defined parameters. In this work we addressed this issue and were able to devise two model selection criteria, based on information theory, to determine the number of clusters in a mixture of pharmacokinetics curves, advancing significantly over previous works.

Our first proposed criterion was based on the Minimum Description Length (MDL) principle, as developed by Rissanen [25]. MDL allows to balance the description of a model with its complexity, obtaining the best clustering outputs in a parameter-free manner. The second solution was based on the Normalised Maximum Likelihood (NML) coding, adapted from the case of Gaussian mixture model as derived in [36].

A current limitation of both EM algorithms in which this work is based [5,6] is that they can only address a constant variance of the measurement noise. In PK, however, a constant coefficient of variation model is commonly used. Concerning EM, this would make the M-Step intractable, as variance would depend deeply on the remaining parameters. A possible solution is to use a variant of EM, where the variance at the current step is proportional to the average concentration computed at the previous step. Unfortunately, in this case, as far as the authors know, there is no analytical guarantee that EM reaches a local maximum of the likelihood.

In closing, we stress that only the one compartment model, with first-order absorption, was considered; the impact of specifying the wrong model was not assessed. We leave for future work addressing other PK models and evaluating which of these best fits the data. Another possible topic to pursue is to consider NLME models accounting for correlations within individual data. In this case, the number of parameters in the MDL and NML criteria needs to be updated to account for the entries of the covariance matrix, at the expense of a much heavier optimisation algorithm.

## 7. Conclusion

In this work we were able to devise two model selection criteria to determine the number of clusters in a mixture of pharmacokinetics curves, advancing over previous works. Experimental results showed the ability of the proposed model-selection schemes to unravel the correct number of clusters in synthetic data.

A parallel implementation was made by assigning to each processor identical amounts of work in the form of the number of random initialisations of the algorithm. By doing so, it was possible to guarantee its cost efficiency. The source code in Java, along with an user guide and the synthetic datasets used in the experiments, was made available at the GitHub repository [22].

## Competing interests

The authors declare they have no competing interests.

## Acknowledgements

This work was supported by national funds through FCT, Fundação para a Ciência e a Tecnologia, under contract IT (UID/EEA/50008/2013), and by projects PERSEIDS (PTDC/EMS-SIS/0642/2014), NEUROCLINOMICS2 (PTDC/EEI-SII/1937/2014), and internal IT projects QBigData and RAPID.

## References

[1] J.T. DiPiro, W.J. Spruill, W.E. Wade, R.A. Blouin, J.M. Pruemer, Concepts In Clinical Pharmacokinetics, fourth ed., American Society of Health-System Pharmacists, 2005.
[2] S. Tang, Y. Xiao, One-compartment model with Michaelis-Menten elimination kinetics and therapeutic window: an analytical approach, J. Pharmacokinet. Pharmacodyn. 34 (6) (2007) 807–827.
[3] C. Csajka, D. Verotta, Pharmacokinetic–pharmacodynamic modelling: history and perspectives, J. Pharmacokinet. Pharmacodyn. 33 (3) (2006) 227–279.
[4] J.G. Wagner, A modern view of pharmacokinetics, J. Pharmacokinet. Biopharm. 1 (5) (1973) 363–401.
[5] E. Tomás, S. Vinga, A.M. Carvalho, Unsupervised learning of pharmacokinetic responses, Comput. Stat. 32 (2017) 409–428.
[6] L. Azzimonti, F. Ieva, A.M. Paganoni, Nonlinear nonparametric mixed-effects models for unsupervised classification, Comput. Stat. 28 (2013) 1549–1570.
[7] S.J. Roberts, D. Husmeier, I. Rezek, W. Penny, Bayesian approaches to gaussian mixture modeling, IEEE Trans. Pattern Anal. Mach. Intell. 20 (1998) 0–1142.
[8] G. McLachlan, D. Peel, Finite Mixture Models, Wiley Series in Probability and Statistics, Wiley-Interscience, 2000.
[9] A. Dasgupta, A.E. Raftery, Detecting features in spatial point processes with clutter via model-based clustering, J. Am. Stat. Assoc. 93 (1998) 294–302.
[10] J. Rissanen, Stochastic Complexity in Statistical Inquiry, World Scientific Series in Computer Science, World Scientific, 1989.
[11] M.A.T. Figueiredo, A.K. Jain, Unsupervised learning of finite mixture models, IEEE Trans. Pattern Anal. Mach. Intell. 24 (2002) 381–396.
[12] C. Biernacki, G. Celeux, G. Govaert, Assessing a mixture model for clustering with the integrated completed likelihood, IEEE Trans. Pattern Anal. Mach. Intell. 22 (2000) 719–725.
[13] C. Biernacki, G. Celeux, G. Govaert, An improvement of the NEC criterion for assessing the number of clusters in a mixture model, Pattern Recognit. Lett. 20 (1999) 267–272.
[14] C. Biernacki, G. Govaert, Using the classification likelihood to choose the number of clusters, Comput. Sci. Stat. 29 (1997) 451–457.
[15] J.D. Banfield, A.E. Raftery, Model-based gaussian and non-gaussian clustering, Biometrics 49 (1993) 803–821.
[16] M.P. Windham, A. Cutler, Information ratios for validating mixture analyses, J. Am. Stat. Assoc. 87 (420) (1992) 1188–1192.
[17] H. Bensmail, G. Celeux, A.E. Raftery, C.P. Robert, Inference in model-based cluster analysis, Stat. Comput. 7 (1997) 1–10.
[18] G.J. McLachlan, On bootstrapping the likelihood ratio test stastistic for the number of components in a normal mixture, J. R. Stat. Soc. 36 (1987) 318–324.
[19] P. Smyth, Model selection for probabilistic clustering using cross-validated likelihood, Stat. Comput. 10 (2000) 63–72.
[20] A. Barron, J. Rissanen, B. Yu, A modern view of pharmacokinetics, IEEE Trans. Inf. Theory 44 (6) (1998) 2743–2760.
[21] J. Rissanen, Fisher information and stochastic complexity, IEEE Trans. Inf. Theory 42 (1996) 40–47.
[22] R. Guerra, Clustering of pharmacokinetic responses, 2017, (https://rjri.github.io/pkclusteringmdl/).
[23] A. Rescigno, Foundations Of Pharmacokinetics, First ed., Springer, 2003.
[24] J. Bilmes, A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models, Technical Report, International Computer Science Institute, 1998.
[25] P.D. Grünwald, J. Rissanen, The Minimum Description Length Principle (Adaptive Computation and Machine Learning), Adaptive Computation and Machine Learning, The MIT Press, 2007.
[26] A.M. Carvalho, P. Adão, P. Mateus, Hybrid learning of bayesian multinets for binary classification, Pattern Recognit. 47 (10) (2014) 2438–3450.
[27] J.L. Monteiro, S. Vinga, A.M. Carvalho, Polynomial-time algorithm for learning optimal tree-augmented dynamic bayesian networks, in: M. Meila, T. Heskes (Eds.), Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence, UAI, AUAI Press, 2015, pp. 622–631.
[28] P.D. Grünwald, A tutorial introduction to the minimum description length principle, CoRR (2004).
[29] C.E. Shannon, A mathematical theory of communication, Bell Syst. Tech. J. 27 (3) (1948) 379–423.
[30] J. Rissanen, Stochastic complexity and modeling, Ann. Stat. 14 (3) (1986) 1080–1100.
[31] M.H. Hansen, B. Yu, Model selection and the principle of minimum description length, J. Am. Stat. Assoc. 96 (454) (2001) 746–774.
[32] G. Schwarz, Estimating the dimension of a model, Ann. Stat. 6 (1978) 461–464.
[33] J.I. Myung, D.J. Navarro, M.A. Pitt, Model selection by normalized maximum likelihood, J. Math. Psychol. 50 (2) (2006) 167–179.
[34] D.E. Bell, Regret in decision making under uncertainty, Oper. Res. 30 (1982).
[35] Y.M. Shtarkov, Universal sequential coding of individual messages., (translated from) Prob. Inf. Transm. 23 (3) (1987) 3–17.
[36] S. Hirai, K. Yamanishi, Efficient computation of normalized maximum likelihood codes for gaussian mixture models with its applications to clustering, IEEE Trans. Inf. Theory 59 (2013) 7718–7727.
[37] J. Wishart, The generalised product moment distribution in samples from a normal multivariate population, Biometrika 20A (1928) 32–52.
[38] I. Foster, Designing and Building Parallel Programs: Concepts and Tools for Parallel Software Engineering, Addison Wesley, 1995.
[39] S.L. Beal, L.B. Sheiner, A.J. Boeckmann, NONMEM Users Guide, Technical Report, University of California, San Francisco, 1993.
[40] H. Derendorf, L.J. Lesko, P. Chaikin, W.A. Colburn, P. Lee, R. Miller, R. Powell, G. Rhodes, D. Stanski, J. Venitz, Pharmacokinetic/pharmacodynamic modeling in drug research and development, J. Clin. Pharmacol. 40 (12 Pt 2) (2000) 1399–1418.
[41] D.E. Mager, E. Wyska, W.J. Jusko, Diversity of mechanism-based pharmacodynamic models, Drug Metab. Dispos. 31 (5) (2003) 510–518.
[42] I. Gueorguieva, K. Ogungbenro, G. Graham, S. Glatt, L. Aarons, A program for individual and population optimal design for univariate and multivariate response pharmacokinetic-pharmacodynamic models, Comput. Methods Programs Biomed. 86 (1) (2007) 51–61.
[43] M. Davidian, D.M. Giltinan, Nonlinear models for repeated measurement data: an overview and update, J. Agric. Biol. Environ. Stat. 8 (2003) 387–419.
[44] L. Wu, A joint model for nonlinear mixed-effects models with censoring and covariates measured with error, with application to AIDS studies, J. Am. Stat. Assoc. 97 (2002) 955–964.
[45] L. Wu, Exact and approximate inferences for nonlinear mixed-effects models with missing covariates, J. Am. Stat. Assoc. 99 (2004) 700–709.
[46] E. Kuhn, M. Lavielle, Maximum likelihood estimation in nonlinear mixed effects models, Comput. Stat. Data Anal. 49 (2005) 1020–1038.
[47] Monolix version 2016R1, 2016. Antony, France: Lixoft SAS.