pharmaceutics

MDPI

Article

An Algorithm for Nonparametric Estimation of a Multivariate Mixing Distribution with Applications to Population Pharmacokinetics

Walter M. Yamada [1,†], Michael N. Neely [1,2,†], Jay Bartroff [3,†], David S. Bayard [1,4,†], James V. Burke [5,†], Mike van Guilder [1,†], Roger W. Jelliffe [1,†], Alona Kryshchenko [1,6,†], Robert Leary [7,†], Tatiana Tatarinova [8,†] and Alan Schumitzky [1,3,*]

1 Laboratory of Applied Pharmacokinetics and Bioinformatics, Children's Hospital of Los Angeles, Los Angeles, CA 90027, USA; wyamada@chla.usc.edu (W.M.Y.); mneely@chla.usc.edu (M.N.N.); dbay007@earthlink.net (D.S.B.); uphill@cox.net (M.v.G.); alona.kryshchenko@csuci.edu (A.K.); schum@usc.edu (A.S.)
2 Pediatric Infectious Diseases, Children's Hospital of Los Angeles, Keck School of Medicine, University of Southern California, Los Angeles, CA 90027, USA
3 Department of Mathematics, University of Southern California, Los Angeles, CA 90089, USA; bartroff@usc.edu
4 Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109, USA
5 Department of Mathematics, University of Washington, Seattle, WA 98195, USA; burke@math.washington.edu
6 Department of Mathematics, California State University Channel Islands, University Dr, Camarillo, CA 93012, USA
7 Certara, Raleigh, NC 27606, USA; Bob.Leary@certara.com
8 Department of Biology, University of La Verne, La Verne, CA 91750, USA; ttatarinova@laverne.edu
* Correspondence: schum@usc.edu; Tel.: +1 818-249-9444
† These authors contributed equally to this work.

Citation: Yamada, W.M.; Neely, M.; Bartroff, J.; Bayard, D.S.; Burke, J.V.; van Guilder, M.; Jelliffe, R.W.; Kryshchenko, A.; Leary, R.; Tatarinova, T.; Schumitzky, A. An Algorithm for Nonparametric Estimation of a Multivariate Mixing Distribution with Applications to Population Pharmacokinetics. *Pharmaceutics* **2021**, *1*, 0. https://dx.doi.org/

Received:
Accepted:
Published:

Abstract:
Population pharmacokinetic (PK) modeling has become a cornerstone of drug development and optimal patient dosing. This approach offers great benefits for datasets with sparse sampling, such as in pediatric patients, and can describe between-patient variability. While most current algorithms assume normal or log-normal distributions for PK parameters, we present a mathematically consistent nonparametric maximum likelihood (NPML) method for estimating multivariate mixing distributions without any assumption about the shape of the distribution. This approach can handle distributions with any shape for all PK parameters. It is shown in convexity theory that the NPML estimator is discrete, meaning that it has finite number of points with nonzero probability. In fact, there are at most N points where N is the number of observed subjects. The original infinite NPML problem then becomes the finite dimensional problem of finding the location and probability of the support points. In the simplest case, each point essentially represents the set of PK parameters for one patient. The probability of the points is found by a primal-dual interior-point method; the location of the support points is found by an adaptive grid method. Our method is able to handle high-dimensional and complex multivariate mixture models. An important application is discussed for the problem of population pharmacokinetics and a nontrivial example is treated. Our algorithm has been successfully applied in hundreds of published pharmacometric studies. In addition to population pharmacokinetics, this research also applies to empirical Bayes estimation and many other areas of applied mathematics. Thereby, this approach presents an important addition to the pharmacometric toolbox for drug development and optimal patient dosing.

Keywords: mixture distribution; mixture model; high dimensional statistics; nonparametric maximum likelihood; primal-dual interior-point method; adaptive grid; population model

1. Introduction

Pharmacokinetic studies in healthy volunteers commonly collect multiple observations in each subject. These datasets often arise from Phase I clinical trials and have been traditionally analyzed by noncompartmental methods or by modeling via the standard two-stage approach. Patient datasets often contain complex dosage regimens and only a few (or one) observation(s) per patient. To analyze such sparse datasets, e.g., in pediatric or critically ill patients, population modeling is required, since noncompartmental analysis and the standard two-stage approach are not applicable [1,2]. Population modeling has been shown to estimate PK parameters without bias and with good precision for such sparse datasets [3]. Parametric population modeling algorithms are commonly used and assume typically either normal or log-normal distributions for the between subject variability of PK parameters. While this assumption is made for virtually every parametric population PK model, it is difficult to prove, especially for datasets with a small number of subjects. Nonparametric population modeling can describe a multivariate distribution of PK parameters without assuming any shape of the PK parameter distribution. This is a key advantage of the nonparametric approach that is based on the exact log-likelihood. The present work comprehensively describes the foundation of this nonparametric estimation algorithm for the first time. This algorithm has been used in hundreds of peer-reviewed papers.

Pharmacometric observations can be described statistically by a mixture model. In this case, the probability of random variable arguments (the PK population model) of the pharmacokinetic compartmental model are described by a mixing distribution. The problem of estimating the mixing distribution from a set of pharmacometric observations can be stated as follows. Let $\boldsymbol{Y}_1, ..., \boldsymbol{Y}_N$ be a sequence of independent but not necessarily identically distributed random vectors constructed from one or more observations from each of N subjects in the population. Let $\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_N$ be a sequence of independent and identically distributed random vectors that represent unknown parameter values of N subjects. The $\boldsymbol{\theta}$ belong to a compact subset Θ of Euclidean space with common but unknown distribution F. In other words, F is a distribution of the parameters in the population model. Θ represents the parameter space, and the dimension of this space is the number of parameters in the population model. The $\{\boldsymbol{\theta}_i\}$ are not observed. It is assumed that the conditional densities $p(\boldsymbol{Y}_i|\boldsymbol{\theta}_i)$ are known, for $i = 1, ..., N$. $p(\boldsymbol{Y}_i|\boldsymbol{\theta}_i)$ represents the model of observations y_i given parameter values θ_i including uncertainties of the measurement protocol. The mixing distribution of $\boldsymbol{Y}_i$ with respect to F is given by $p(\boldsymbol{Y}_i|F) = \int p(\boldsymbol{Y}_i|\boldsymbol{\theta}_i)dF(\boldsymbol{\theta}_i)$. Because of independence of the $\{\boldsymbol{Y}_i\}$, the mixing distribution of the $\{\boldsymbol{Y}_i\}$ with respect to F is given by

$$L(F) = p(\boldsymbol{Y}_1, ..., \boldsymbol{Y}_N|F) = \prod_{i=1}^{N} \int p(\boldsymbol{Y}_i|\boldsymbol{\theta}_i)dF(\boldsymbol{\theta}_i) \tag{1}$$

The mixing distribution problem is to maximize the likelihood function $L(F)$ with respect to all probability distributions F on Θ.

Remark 1. *The distribution F^{ML} that maximizes $L(F)$ is a consistent estimator of the true mixing distribution; which means that F^{ML} will converge to the true distribution if the number of subjects is large. This was proved originally by Kiefer and Wolfowitz in 1956 [4] . The consistency of F^{ML} is especially important for our application to population pharmacokinetics where F^{ML} is used as a prior distribution for Bayesian dosage regimen design [5].*

The algorithm described in this paper differs from most other published methods in a number of ways. Our algorithm allows for high dimensional parameter space Θ. Most published methods require the dimension of Θ to be small and many require the dimension of Θ to be 1 , i.e., these methods required the number of parameters in the population

model to be small. We have treated examples where the dimension of Θ is as high as 29, see Section 3.

Most published algorithms require the $\{\boldsymbol{Y}_i\}$ to be identically distributed and assume that the population model $\{p(\boldsymbol{Y}_i|\boldsymbol{\theta}_i)\}$ is rather simple, such as $p(\boldsymbol{Y}_i|\boldsymbol{\theta}_i)$ is a multivariate normal density with mean vector $\boldsymbol{\theta}_i$ and covariance matrix $\boldsymbol{\Sigma}$. Even if $\boldsymbol{\Sigma}$ is unknown and has to be estimated, the structure of this model is straightforward. However, the estimation of $\boldsymbol{\Sigma}$ has to be done carefully to avoid singularities, see Wang and Wang [6]. As will be described in Section 3, we allow $p(\boldsymbol{Y}_i|\boldsymbol{\theta}_i)$ to be calculated from a system of nonlinear ordinary differential–algebraic equations.

We now describe the details of our algorithm. It was proved by Lindsay [7] and Mallet [8] , under simple hypotheses on the population model $\{p(\boldsymbol{Y}_i|\boldsymbol{\theta}_i)\}$, that the global maximizer F^{ML} of $L(F)$ could be represented by a discrete distribution with support on at most N points, i.e., a distribution with nonzero probability located on at most N points.

Remark 2. *One way to motivate this result by Lindsay and Mallet is as follows: Suppose by some lucky chance, we knew the exact parameters for each subject. How can we package this into a distribution? Answer: The "empirical distribution" of the exact parameters. That is the discrete distribution supported at the N exact parameters with equal weights. It turns out in this case that this empirical distribution is also the nonparametric maximum likelihood (NPML) distribution of the parameters. What is remarkable is that if we only have noisy measurements* $(Y_1, ..., Y_N)$ *of the N subjects only indirectly related to the subject parameters, the structure of the NPML is the same. A discrete distribution supported at N points. Of course, in this real case, the position and weights of the N support points are unknown. Finding these positions and weights is the subject of this paper.*

This result leads immediately to a finite dimensional optimization problem for F^{ML}, namely to maximize the likelihood function

$$L(\boldsymbol{\lambda}, \boldsymbol{\phi}) = \prod_{i=1}^{N} \sum_{k=1}^{K} \lambda_k p(\boldsymbol{Y}_i|\boldsymbol{\phi}_k) \tag{2}$$

with respect to the support points $\boldsymbol{\phi} = (\boldsymbol{\phi}_1, ..., \boldsymbol{\phi}_K)$ and weights $\boldsymbol{\lambda} = (\lambda_1, ..., \lambda_K)$ such that $\boldsymbol{\phi}_k \in \Theta, \lambda_k \geq 0$ for $k = 1, ..., K$, $K \leq N$ and $\sum_{k=1}^{K} \lambda_k = 1$.

In our algorithm $l(\boldsymbol{\lambda}, \boldsymbol{\phi}) = \log L(\boldsymbol{\lambda}, \boldsymbol{\phi})$ is maximized, so that

$$l(\boldsymbol{\lambda}, \boldsymbol{\phi}) = \sum_{i=1}^{N} \log \sum_{k=1}^{K} \lambda_k p(\boldsymbol{Y}_i|\boldsymbol{\phi}_k) \tag{3}$$

and the maximization problem becomes

$$\text{maximize } l(\boldsymbol{\lambda}, \boldsymbol{\phi}) \tag{4}$$

such that $\boldsymbol{\phi} \in \Theta^K$, $\boldsymbol{\lambda} = (\lambda_1, ..., \lambda_K) \in \mathbb{R}_+^K$, $K \leq N$ and $\sum_{k=1}^{K} \lambda_k = 1$.

Although the maximization problem in Equation (4) is finite dimensional, it is still high-dimensional. The dimension of the maximization problem in Equation (4) is $N(\dim \Theta) + (N - 1)$.

The optimization problem in Equation (4) is naturally divided into two problems:

Problem 1. Given a set of support points $\{\boldsymbol{\phi}_k\}$, find the optimal weights $\{\lambda_k\}$.

Problem 2. Given the solution to Problem 1, find a better set of support points.

Problems 1 and 2 are solved cyclically until convergence, i.e., no significant improvement in $l(\boldsymbol{\lambda}, \boldsymbol{\phi})$.

Problem 1 is a convex programming problem. In our algorithm, we solve this problem by the primal-dual interior-point (PDIP) method. This type of method is standard in convex optimization theory (see Boyd and Vandenberghe [9]). However, the exact implementation for a specific problem varies from problem to problem. The exact details of our implemen-

tation are described in the Appendix. See also Bell [10], Baek [11] and Yamada et al. [12]. Our PDIP implementation is fast and can easily handle thousands of variables.

Finding a better set of support points in Problem 2 is a more difficult problem. This location problem is a nonconvex global optimization problem with many local extrema and whose dimension is potentially $N \times \dim \Theta$. The details of our algorithm, called the adaptive grid (AG) method, will be described in Section 2.3 and in Algorithm 1.

Algorithm 1 Nonparametric adaptive grid (NPAG) algorithm. Input: $(\boldsymbol{Y}, \boldsymbol{\phi}^0, \boldsymbol{a}, \boldsymbol{b}, \Delta_D, \Delta_L, \Delta_F, \Delta_e, \Delta_\lambda)$, $\boldsymbol{a}$ and $\boldsymbol{b}$ are the lists of lower and upper bounds, respectively, of Θ; Δ_D is the minimum distance allowable between points in the estimated F^{ML}. Δ_x see §Section 2.7. Output: $(\boldsymbol{\phi}, \boldsymbol{\lambda}, l(\boldsymbol{\lambda}, \boldsymbol{\phi}))$.

procedure NPAG($\boldsymbol{Y}$, $\boldsymbol{\phi}^0, \boldsymbol{a}, \boldsymbol{b}, \Delta_D$) ▷ Estimate F^{ML} given $\boldsymbol{Y}$
  Initialization: $\boldsymbol{\phi} = \boldsymbol{\phi}^0$, $LogLike = -10^{30}$, $F_0 = 10^{30}$, $F_1 = 2 * F_0$, $eps = 0.2$, $\Delta_e = 10^{-4}$, $\Delta_F = 10^{-2}$, $\Delta_L = 10^{-4}$, $\Delta_\lambda = 10^{-3}$, $n = 0$
  while $eps \geq \Delta_e$ or $|F_1 - F_0| \geq \Delta_F$ **do**
    Calculate $\boldsymbol{\Psi}(\boldsymbol{\phi})$ ▷ $N \times K$ matrix $\{p(Y_i|\phi_k)\}$
    $[\hat{\boldsymbol{\lambda}}(\boldsymbol{\phi}), l(\hat{\boldsymbol{\lambda}}(\boldsymbol{\phi}), \boldsymbol{\phi})] \longleftarrow$ PDIP($\boldsymbol{\Psi}(\boldsymbol{\phi})$) ▷ Appendix A
    if (MAXCYCLES == 0) **then**
      $F^{ML}_{est} \longleftarrow l(\hat{\boldsymbol{\lambda}}(\boldsymbol{\phi}), \boldsymbol{\phi})$
      $\boldsymbol{\lambda} \longleftarrow \hat{\boldsymbol{\lambda}}(\boldsymbol{\phi})$
      return $[\boldsymbol{\phi}, \boldsymbol{\lambda}, F^{ML}_{est}]$
    end if
    $n \longleftarrow n + 1$
    $\boldsymbol{\phi}^c \longleftarrow$ CONDENSE($\boldsymbol{\phi}, \hat{\boldsymbol{\lambda}}(\boldsymbol{\phi}), \Delta_\lambda$) ▷ Alg. 3
    $[\hat{\boldsymbol{\lambda}}(\boldsymbol{\phi}^c), l(\hat{\boldsymbol{\lambda}}(\boldsymbol{\phi}^c), \boldsymbol{\phi}^c)] \longleftarrow$ PDIP($\boldsymbol{\Psi}(\boldsymbol{\phi}^c)$) ▷ PDIP returns G^n
    $NewLogLike = l(\hat{\boldsymbol{\lambda}}(\boldsymbol{\phi}^c), \boldsymbol{\phi}^c)$
    if ($n >$ MAXCYCLES) **then**
      $F^{ML}_{est} \longleftarrow l(\hat{\boldsymbol{\lambda}}(\boldsymbol{\phi}^c), \boldsymbol{\phi}^c)$
      $\boldsymbol{\lambda} \longleftarrow \hat{\boldsymbol{\lambda}}(\boldsymbol{\phi}^c)$
      return $[\boldsymbol{\phi}, \boldsymbol{\lambda}, F^{ML}_{est}]$
    end if
    if $|NewLogLike - LogLike| \leq \Delta_L$ **and** $eps > \Delta_e$ **then**
      $eps = eps/2$ ▷ Adjust precision
    end if
    if $eps \leq \Delta_e$ **then** ▷ check EXIT conditions
      $F_1 = NewLogLike$
      if $|F_1 - F_0| \leq \Delta_F$ **then**
        $F^{ML}_{est} \longleftarrow F_1$
        $\boldsymbol{\phi} \longleftarrow \boldsymbol{\phi}^c; \boldsymbol{\lambda} \longleftarrow \hat{\boldsymbol{\lambda}}(\boldsymbol{\phi}^c)$
        return $[\boldsymbol{\phi}, \boldsymbol{\lambda}, F^{ML}_{est}]$
      else
        $F_0 = F_1; eps = 0.2$ ▷ Reset Algorithm
      end if
    end if
    $\boldsymbol{\phi} \longleftarrow \boldsymbol{\phi}^e \longleftarrow$ EXPAND($\boldsymbol{\phi}^c, eps, \boldsymbol{a}, \boldsymbol{b}, \Delta_D$) ▷ Alg. 2
    $LogLike \leftarrow NewLogLike$
  end while
end procedure

Roughly speaking, Problems 1 and 2 are solved as follows. An initial large grid of possible support points is defined in the hypercube Θ. Problem 1 is solved on this large grid. After PDIP, most of the original grid points are removed due to near-zero weights, leaving a smaller high-probability grid. Problem 1 is then solved on this smaller grid. Then, the adaptive grid method for Problem 2 takes place. For each remaining grid point, up to $2 \times \dim \Theta$ new (daughter) support points are added. A daughter point outside the search space Θ or too close to a parent point is discarded. The new grid contains the current high-probability points plus the added daughter points. The algorithm is then ready for Problem 1 again. By construction, each iteration increases the value of $l(\boldsymbol{\lambda}, \boldsymbol{\phi})$. This process continues until the function $l(\boldsymbol{\lambda}, \boldsymbol{\phi})$ does not significantly change.

1.1. Comparable Methods

Because of space limitations, in this section, we only discuss NPML methods that optimize Equation (4), methods that treat multivariate distributions, and methods which allow general conditional probabilities $\{P(\boldsymbol{Y}_i, \boldsymbol{\theta}_i)\}$. As explained in this paper, any such NPML algorithm has to address two problems: locations of support points and weights of support points. NPAG does locations by an adaptive grid method and weights by the primal-dual interior-point (PDIP) method. The algorithms discussed in this seection are summarized in Table 1.

The original methods of Lindsay [7] and Mallett [8] were based on algorithms of optimal design in the style of Fedorov [13]. In Schumitzky [14], an algorithm was proposed which did both locations and weights by the expectation maximization algorithm (EM). It was stable but also slow.

In Lesperance and Kalbfleisch [15], a new method was introduced which did weights by the dual method described in Section 5 of Lindsay [7] and locations by what they called the intra-simplex direction method (ISDM). Even though the Lesperance and Kalbfleisch paper was restricted to univariate distributions, the ISDM method has been generalized to the multivariate case. To briefly describe ISDM, let $D(\boldsymbol{\theta}, F)$ be the directional derivative of $\log L(F)$ in the direction of the Dirac distribution $\delta_{\boldsymbol{\theta}}$ supported at $\boldsymbol{\theta} \in \Theta$. (This function is defined in Section 4 below.) ISDM is an iterative algorithm. At stage k, let F^k be the current estimate F^{ML}. Then, find all the local maxima of $D(\boldsymbol{\theta}, F^k)$. These local maxima are added to the current set of support points and a new F^{k+1} is calculated. If there are no new local maxima, then the algorithm is done.

In Pilla, Bartolucci, and Lindsay [16], another new method was developed where the locations were found by an initial fine grid. However, the weights were found by a dual version of the PDIP method.

In Savic, Kjellsson, and Karlsson [17], a nonparametric method (NONMEM-NP) was added to the popular NONMEM® program. NONMEM-NP is a hybrid parametric–nonparametric approach The locations of support points were found by a parametric maximum likelihood algorithm. Then, the weights were found by maximizing Equation (4) relative to the newly found support points. NONMEM-NP can handle high-dimensional and complex multivariate distributions. An extension to NONMEM-NP was developed in Savic and Karlsson [18] where additional support points are added to the original set. A comparison between NONMEM-NP and NPAG is discussed in Leary [19].

In Wang and Wang [6] , a new algorithm was developed for multivariate distributions. The locations were found by a combination of EM and a variant of ISDM. The weights were found by a family of quadratic programs. In [6], examples are performed for 8- and 13-dimensional mutivariate mixing distributions.

Table 1. Table of NPML Methods.

Author(s)/Reference/Date	Problem 1 Method	Problem 2 Method
Lindsay [7] (1983)	Convex Geometry	VDM
Mallet [8] (1986)	Optimal Design	VDM
Lesparance and Kalbfleisch [15] (1992)	Semi-Infinite Programming	ISDM
Savic, Kjellsson and Karlsson [17] (2009)	Parametric NONMEM	None
Pilla, Bartolucci and Lindsay [16] (2006)	Dual of PDIP	Adaptive Grid
Schumitzky [14] (1991)	EM	EM
X. Wang and Y. Wang [6] (2015)	Quadratic Programming	ISDM
NPAG [20] (2001)	PDIP	Adaptive Grid

Legend: VDM vertex direction method; EM expectation–maximization; ISDM intra-simplex direction method; NPML nonparametric maximum likelihood; PDIP primal-dual interior-point method.

Note: The quadratic programming algorithm (QP) of Wang and Wang [6] has an attractive feature. For a prescribed set of support points, QP finds the zero probabilities exactly. Thus, QP avoids the grid condensation step where support points from PDIP with sufficiently low probabilities are deleted. However, QP and PDIP are based on different numerical methods and a comparison of the efficiency of both algorithms has not been determined.

We finally mention that the NPML problem is a special case of a finite mixture model problem with unknown supports and weights. For a discussion of this approach, see Tatarinova and Schumitzky [21].

The algorithms which have shown by published examples to handle the highest dimensional multivariate problems are NONMEM-NP, Wang and Wang [6], and NPAG.

1.2. Benders Decomposition

For any set of grid points $\boldsymbol{\phi} = (\boldsymbol{\phi}_1, ..., \boldsymbol{\phi}_m)$ in Θ^m, let $\boldsymbol{\lambda} = \hat{\boldsymbol{\lambda}}(\boldsymbol{\phi})$ be the corresponding set of optimal weights given by the PDIP method. Then, the function $F(\boldsymbol{\phi}) = l(\hat{\boldsymbol{\lambda}}(\boldsymbol{\phi}), \boldsymbol{\phi})$ depends only on $\boldsymbol{\phi}$ and can be maximized directly. For optimization methods, this technique is called Benders decomposition. The NPAG algorithm maximizes $F(\boldsymbol{\phi})$ by an adaptive search method. In a method proposed by James Burke, $F(\boldsymbol{\phi})$ is maximized by a Newton-type method. Since the function $F(\boldsymbol{\phi})$ is not necessarily differentiable, a relaxed Newton method must be used similar to what is described in the Appendix for the primal-dual algorithm. For details of Benders decomposition as applied to our problem, see Bell [10], Baek [11] and Jordan-Squire [22].

Founded on this prior work, the present study aimed to comprehensively describe, for the first time, the nonparametric adaptive grid algorithm (NPAG). This approach uses the exact log-likelihood to solve population modeling problems and does not make any assumptions about the shape of the PK parameter distributions. We illustrated the features and capabilities of this algorithm using a population PK modeling example. This algorithm presents the computational foundation of several hundred peer-reviewed papers, to date, and is ideally suited to optimize individual patient dosage regimens. The output of the NPAG algorithm becomes the input of the BestDose™ patient dosing software which is used at the bedside in real-time [23].

2. Materials and Methods

2.1. Pmetrics

The simulations and NPAG optimizations in this paper can be duplicated in R, using programs in the Pmetrics package [24]. R and Pmetrics are free software. R is available from many download sites. Pmetrics is available from lapk.org. NPAG is run using the NPrun()

command in Pmetrics. Sample datasets and compartmental models are also available at lapk.org.

2.2. NPAG Subprograms

NPAG is a Fortran program consisting of a number of subroutines as described below. The main program performs the adaptive grid (AG) method (consisting of expansion and compression algorithms) and calls the primal-dual interior-point (PDIP) subprogram. The PDIP algorithm solves the maximization problem of Equation (4) for a fixed grid and is described precisely in the Appendix.

2.3. NPAG Implementation (NPAG—Algorithm 1)

For the purpose of this discussion, we can think of PDIP as a function $\hat{\boldsymbol{\lambda}}$ from Θ^m into the set $S^m = \{\boldsymbol{\lambda} \in \mathbb{R}_+^m : \sum_{k=1}^m \lambda_k = 1\}$ defined as follows: If $\boldsymbol{\phi} = (\boldsymbol{\phi}_1, ..., \boldsymbol{\phi}_m)$ then $\hat{\boldsymbol{\lambda}}(\boldsymbol{\phi}) = (\hat{\lambda}_1, ..., \hat{\lambda}_m)$ maximizes Equation (4) relative to the fixed set of grid points $(\boldsymbol{\phi}_1, ..., \boldsymbol{\phi}_m)$. In this case, we write $G = (\boldsymbol{\phi}, \hat{\boldsymbol{\lambda}}(\boldsymbol{\phi}))$ and $l(G) = l(\boldsymbol{\phi}, \hat{\boldsymbol{\lambda}}(\boldsymbol{\phi}))$.

In NPAG, there are two types of grids: expanded and condensed. The expanded grids are the initial grid and the grids after grid expansion (Algorithm 2). The condensed grids are generated by grid condensation (Algorithm 3). Each cycle of NPAG begins with an expanded grid. The likelihood calculation is done on the condensed grids.

Now for the adaptive grid method. Assume that Θ is a bounded Q-dimensional hyper-rectangle. Initially, we let $\boldsymbol{\phi}^0_{expanded} = (\boldsymbol{\phi}^0_1, ..., \boldsymbol{\phi}^0_M)$ be the set of M Faure grid points in Θ (see [25–27]). Alternatively, we could initially let $\boldsymbol{\phi}^0_{expanded}$ be generated by a uniform distribution on Θ or by a prior run of the program.

Remark. The Faure grid points for a hyper-rectangle Θ are a low-discrepancy set which in some sense optimally and uniformly covers Θ. In our implementation of NPAG, the Faure point sets come in discrete sizes which nest with each other. (Allowable number of points equals 2129, 5003, 10007, 20011, 40009, 80021, and multiples of 80021.) This nesting property is useful for checking the optimality of F^{ML} (see Section 4). We have found that replacing the initial Faure set with a set generated by a uniform distribution on Θ increases the time to convergence but results in the same optimal distribution.

Now set $G^0_{expanded} = (\boldsymbol{\phi}^0, \hat{\boldsymbol{\lambda}}(\boldsymbol{\phi}^0))$. Our approach is to generate a sequence of solutions G^n to Equation (4) of increasingly greater likelihood, where unless otherwise specified, G^n refers to the condensed grid at the n^{th} cycle of the algorithm. If G^n has log-likelihood negligibly different than G^{n-1}, then G^n is considered the optimal solution to Equation (4) and is relabeled F^{ML}. If not, then the process continues using the $\boldsymbol{\phi}^n$ as the new seed. This loop is repeated until F^{ML} is found.

The stopping conditions for NPAG are defined precisely in Algorithm 1. If the stopping conditions are not met prior to a set maximum number of iterations, the program will exit after writing the last calculated G^n into a file.

2.4. Grid expansion (EXPAND—Algorithm 2)

The crux of the adaptive grid method is how to go from G^0 to G^1 or, more generally, from G^n to G^{n+1}. The details of doing this are now explained roughly below and precisely in Algorithm 1.

Let Q be the dimension of Θ. Suppose at stage n we have a grid of high-probability support points $\boldsymbol{\phi}^n$. We then add $2Q$ daughter points for each support point $\boldsymbol{\phi}_k \in \boldsymbol{\phi}^n$. The daughter points are the vertices of a small hyper-rectangle centered at each $\boldsymbol{\phi}_k$ with size proportional to the original size of the hyper-rectangle defining Θ. The size of this small hyper rectangle decreases as the accuracy of the estimates increases. (See Algorithm 2.)

Let $\boldsymbol{\phi}^{n+1}_{expanded} = \boldsymbol{\phi}^n \cup$ Daughter-Points. Then the PDIP subprogram is applied to $\boldsymbol{\phi}^{n+1}_{expanded}$ resulting in the new solution set $G^{n+1}_{expanded} = (\boldsymbol{\phi}^{n+1}_{expanded}, \hat{\boldsymbol{\lambda}}(\boldsymbol{\phi}^{n+1}_{expanded}))$ (see Algorithm 1). The solution set $G^{n+1}_{expanded}$ is now ready for grid condensation.

Algorithm 2 EXPAND. Input: $\boldsymbol{\phi}$=$(\phi_1, \cdots, \phi_K)$, Δ_G, $\Theta = [a_1, b_1] \times [a_2, b_2] \times ... \times [a_Q, b_Q]$, $\boldsymbol{a} = [a_1, \cdots, a_Q]$, $\boldsymbol{b} = [b_1, \cdots, b_Q]$, Δ_D. Output: $\boldsymbol{\phi}'$=$(\phi'_1, \cdots, \phi'_M)$, where $M \leq K(1 + 2Q)$. Note: In this algorithm, $\boldsymbol{\phi}$=$(\phi_1, \cdots, \phi_K)$ is a $Q \times K$ matrix, with $Q = \dim \Theta$.

function EXPAND($\boldsymbol{\phi}$, Δ_G, $\boldsymbol{a}$, $\boldsymbol{b}$, Δ_D)
Initialize: $[Q, K] = size(\boldsymbol{\phi})$, $\mathbf{I} = \mathbf{Q} \times \mathbf{Q}$ Identity matrix, new$\boldsymbol{\phi} \longleftarrow \boldsymbol{\phi}$
for $k = 1, ..., K$ **do** ▷ $K =$ number of input support points
for $d = 1, ..., Q$ **do** ▷ $Q = \dim \Theta$
$T(d) = \Delta_G(b(d) - a(d))$
if $\phi(d, k) + T(d) \leq b(d)$ **then** ▷ Check upper boundary
$\boldsymbol{\phi}^+ = \boldsymbol{\phi}(:, k) + T(d)\mathbf{I}(:, d)$
$dist = 10^{30}$
end if
for $k_{in} = 1 :$ length(new$\boldsymbol{\phi}$) **do**
newdist $= \sum$abs$(\boldsymbol{\phi}^+ -$ new$\boldsymbol{\phi}(:, k_{in}))./(\boldsymbol{b} - \boldsymbol{a})$ ▷ x ./y done component-wise
$dist =$ min$(dist,$ newdist$)$
end for
if $dist \geq \Delta_D$ **then** ▷ Check distance to new support point
new$\boldsymbol{\phi} \longleftarrow$ [new$\boldsymbol{\phi}, \boldsymbol{\phi}^+$]
end if
if $\phi(d, k) - T(d) \geq a(d)$ **then** ▷ Check lower boundary
$\boldsymbol{\phi}^- = \boldsymbol{\phi}(:, k) - T(d)\mathbf{I}(:, d)$
$dist = 10^{30}$
end if
for $k_{in} = 1 :$ length(new$\boldsymbol{\phi}(1, :)$) **do**
newdist $= \sum$(abs$(\boldsymbol{\phi}^- -$ new$\boldsymbol{\phi}(:, k_{in}))$ $./(b - a))$ ▷ x./y done component-wise
$dist =$ min$(dist,$ newdist$)$
end for
if $dist \geq \Delta_D$ **then** ▷ Check distance to new support point
new$\boldsymbol{\phi} \longleftarrow$ [new$\boldsymbol{\phi}, \boldsymbol{\phi}^-$]
end if
end for
end for
$\boldsymbol{\phi} \longleftarrow$ new$\boldsymbol{\phi}$
end function

Algorithm 3 Condense algorithm. Input: $(\boldsymbol{\phi}, \boldsymbol{\lambda}, \Delta_\lambda)$, Output: $\boldsymbol{\phi}^c$ Note: $\boldsymbol{\phi}^c$ is considered a subset of $\boldsymbol{\phi}$

function CONDENSE($\boldsymbol{\phi}, \boldsymbol{\lambda}, \Delta_\lambda$)

ind = **find** ($\boldsymbol{\lambda} > (\max \boldsymbol{\lambda})\Delta_\lambda$) ▷ Inequality and max are performed component-wise

$\boldsymbol{\phi}^c = \boldsymbol{\phi}(:, \mathbf{ind})$

return $\boldsymbol{\phi}^c$

end function

2.5. Grid Condensation (CONDENSE—Algorithm 3)

The above solution set $G^{n+1}_{expanded}$ may have many support points with low probability. We remove all support points which have corresponding probability less than $(\max \boldsymbol{\lambda})\Delta_\lambda$, where $\boldsymbol{\lambda}$ is the vector of current probabilities and the default for Δ_λ is 10^{-3}. (Note that the remaining probabilities are not normalized at this point.) The probabilities of the remaining support points are normalized by a second call to the PDIP subprogram. This second call to PDIP is fast. The likelihood associated with these remaining support points and normalized probabilities is then used to update the program control parameters and check for convergence (Algorithm 1 and Section 2.7). If convergence is attained, then the output of this second call to PDIP provides the support points and probabilities of the final solution. If convergence is not attained, then the remaining support points are sent to the grid expansion subprogram (Algorithm 2), initializing the next cycle.

At the end of the program, the output of this second call to PDIP provides the location and weights of the final solution.

2.6. PDIP Subprogram—See Appendix A

The PDIP subprogram finds the optimal solution to Equation (4) with respect to $\boldsymbol{\lambda}$ for fixed $\boldsymbol{\phi}$. PDIP employs a primal-dual interior-point method that uses a relaxed Newton method to solve the corresponding Karush–Kuhn–Tucker equations. (See Equations (14)–(17) of Appendix A.)

For any $\boldsymbol{Y}$=($\boldsymbol{Y}_1$, ..., $\boldsymbol{Y}_N$) and any $\boldsymbol{\phi}$=($\boldsymbol{\phi}_1$, ..., $\boldsymbol{\phi}_K$) $\in \Theta^K$, the input to the PDIP subprogram is the $N \times K$ matrix $\{p(\boldsymbol{Y}_i|\boldsymbol{\phi}_k)\}$. The output consists of the optimal weights $\hat{\boldsymbol{\lambda}}(\boldsymbol{\phi})$ and the corresponding log-likelihood $l(\hat{\boldsymbol{\lambda}}(\boldsymbol{\phi}), \boldsymbol{\phi})$. An in-depth description of the PDIP algorithm and its implementation is presented in Appendix A. See also [10–12].

2.7. NPAG Stopping Conditions

As explained above, a potential solution to F^{ML} is not accepted as a global optimum until successive sequences of G^n produce final distributions evaluating to sufficiently close log-likelihood. The various upper and lower bounds Δ for NPAG control and stopping conditions are defined below and are used in Algorithms 1–3.

Δ_L Primary upper bound on the allowable difference between two successive estimated log-likelihoods; the default initialization is 10^{-4}.

Δ_F Secondary upper bound on the allowable difference between two successive estimated log-likelihoods of potential F^{ML}; the default initialization is 10^{-2}.

Δ_e Sets an upper bound on the accuracy variable *eps* of Algorithm 1. The default initialization for Δ_e is 10^{-4}. The default initialization for *eps* is 0.2 and is stepped down until $eps \leq \Delta_e$ Δ_F and Δ_e define the two stopping conditions for Algorithm 1.

Δ_D Sets a lower bound on how close two support points can get; the default initialization is 10^{-4}.

Δ_λ Sets a lower bound factor on the probabilities of the weights λ; the default initialization is 10^{-3}.

2.8. Calculation of $p(\boldsymbol{Y}_i \mid \phi_k)$

Given observations $\boldsymbol{Y}_i$, $i = 1, ..., N$ and grid points $\boldsymbol{\phi}_k$, $k = 1, ..., K$, the PDIP subprogram only depends on the $N \times K$ matrix $\{p(\boldsymbol{Y}_i|\boldsymbol{\phi}_k)\}$. NPAG can be used for any problem once this matrix is defined. However, the default setting of NPAG is for the problem of population pharmacokinetics. For a good background of population pharmacokinetics, see Davidian and Giltinan [28,29].

In population pharmacokinetics, generally $\boldsymbol{Y}_i = (\boldsymbol{y}_{i,1}, ..., \boldsymbol{y}_{i,M})$ is a matrix of vector observations for the ith subject. Since NPAG allows multiple outputs, each $\boldsymbol{y}_{i,m}$ is itself a q-dimensional vector $\boldsymbol{y}_{i,m} = (y_{i,m,1}, \cdots, y_{i,m,q})$. The observations $y_{i,m,j}$, are then typically given by a regression equation of the form:

$$\begin{aligned} &y_{i,m,j} = f_{i,m,j}(\boldsymbol{\theta}_i) + \nu_{i,m,j},\ j = 1, \cdots, q \\ &\nu_{i,m,j} \sim N(0, (\sigma_{i,m,j}(\boldsymbol{\theta}_i))^2) \\ &\quad \boldsymbol{\theta}_i \text{ are unobserved parameters specific for } \boldsymbol{Y}_i \end{aligned} \tag{5}$$

In the above Equation (5), $f_{i,m,j}$ is a known nonlinear function depending on the model structure, the dosage regimen, the sampling schedule, all covariates and of course the subject-specific parameter vector $\boldsymbol{\theta}_i$. Except for simple models, $f_{i,m,j}$ requires the solution of (possibly nonlinear) ordinary differential equations.

In the current implementation of NPAG, it is assumed that the $(\boldsymbol{y}_{i,1}, ..., \boldsymbol{y}_{i,M})$ are independent. Then

$$p(\boldsymbol{Y}_i|\boldsymbol{\phi}_k) = \frac{\exp\left(-\frac{1}{2}\sum_{m=1}^{M}(\boldsymbol{y}_{i,m} - \boldsymbol{f}_{i,m}(\boldsymbol{\phi}_k))\boldsymbol{\Sigma}_{i,m}^{-1}(\boldsymbol{\phi}_k)(\boldsymbol{y}_{i,m} - \boldsymbol{f}_{i,m}(\boldsymbol{\phi}_k))^T\right)}{\prod_{m=1}^{M}\sqrt{(2\pi)^q \det \boldsymbol{\Sigma}_{i,m}(\boldsymbol{\phi}_k)}} \tag{6}$$

where $\boldsymbol{f}_{i,m} = (f_{i,m,1}, ..., f_{i,m,q})$ and $\boldsymbol{\Sigma}_{i,m} = diag(\sigma_{i,m,1}^2, ..., \sigma_{i,m,q}^2)$. For the purposes of matrix multiplication in Equation (6) ,we think of $\boldsymbol{y}_{i,m}$ and $\boldsymbol{f}_{i,m}$ as q-dimensional row vectors.

To complete the description of Equation (6) we need to model the standard deviation terms $\sigma_{i,m,j}$ of the assay noise. In our implementation of NPAG, four different models are allowed. Let

$$\alpha_{i,m,j}(\boldsymbol{\phi}_k) = c_0 + c_1 f_{i,m,j}(\boldsymbol{\phi}_k) + c_2 f_{i,m,j}^2(\boldsymbol{\phi}_k) + c_3 f_{i,m,j}^3(\boldsymbol{\phi}_k) \tag{7}$$

and set

$$\sigma_{i,m,j} = \begin{cases} \alpha_{i,m,j} & \text{assay error polynomial only} \\ \gamma\alpha_{i,m,j} & \text{multiplicative error} \\ \sqrt{\alpha_{i,m,j}^2 + \gamma^2} & \text{additive error} \\ \gamma & \text{constant level of error} \end{cases} \tag{8}$$

The parameter γ in Equation (8) is a variance factor. Artificially increasing the variance during the first several cycles of NPAG increases the likelihood for each $\boldsymbol{\phi}$, allowing the algorithm to use these cycles to find a better initial state from which to begin optimization. NPAG also has an option to "optimize" γ. This changes NPAG from a nonparametric method to a "semiparametric" method and will not be discussed here. The interested reader can consult [12].

Next, if $c_0 = 0$ in Equation (7), then $\alpha_{i,m,j}$ can become small for certain values of $\boldsymbol{\phi}$ that in early iterations can be far from optimal. This, in turn, causes numerical problems as the likelihood is infinite if $\sigma_{i,m,j} = 0$. One way to avoid this problem is to take $\sigma_{i,m,j} = constant$. Another way would be to assume that $\alpha_{i,m,j}$ is known and is given by

$$\alpha_{i,m,j} = c_0 + c_1 y_{i,m,j} + c_2 y_{i,m,j}^2 + c_3 y_{i,m,j}^3 \tag{9}$$

That is, to approximate σ by using a polynomial of the observed values rather than model predicted values. In our experience with NPAG, the approximation of Equation (9) is useful for ensuring computational stability (especially during the early cycles of the algorithm). However, from a theoretical perspective, this change violates the conditions of maximum likelihood and will not be discussed here. Again, the interested reader can consult [12].

2.9. Convergence

For a given initial grid $\boldsymbol{\phi}^0$, the NPAG algorithm is only guaranteed to find a local maximum of $L(F)$. More precisely, if $\boldsymbol{\phi}^*$ is the final grid of NPAG starting from $\boldsymbol{\phi}^0$, then $\hat{\boldsymbol{\lambda}}(\boldsymbol{\phi}^*)$ is a global maximum on $\boldsymbol{\phi}^*$ but the support points $\boldsymbol{\phi}^*$ may be only a local maximum.

Global convergence of a nonparameteric maximum likelihood method for estimation of a multivariate mixing distribution is difficult. For one-dimensional distributions the problem is straightforward. The idea of proof goes back to at least Fedorov [13] in 1972, which involves the use of directional derivatives.

Let F be any distribution on Θ. Then, the directional derivative of $\log L(F)$ in the direction of the Dirac distribution $\delta_{\boldsymbol{\theta}}$ supported at $\boldsymbol{\theta}$ is defined by

$D(\boldsymbol{\theta}, F)$=$[\sum_{i=1}^{N} P(\boldsymbol{Y}_i|\boldsymbol{\theta})/P(\boldsymbol{Y}_i|F)] - N$, $\boldsymbol{\theta} \in \Theta$, where $p(\boldsymbol{Y}_i|F) = \int p(\boldsymbol{Y}_i|\boldsymbol{\theta})dF(\boldsymbol{\theta})$. Let F_k be the current NPML estimate at iteration k. The Fedorov method involves maximizing $D(\boldsymbol{\theta}, F_k)$ for $\boldsymbol{\theta} \in \Theta$, at every iteration. Then, the point at which the maximum occurs is added in an optimal way to F_k to give F_{k+1}. Under the assumptions of regularity, Fedorov shows that $L(F_k)$ converges to $L(F^{ML})$, see Fedorov [13], (Theorem 2.5.3). Many improvements to this method have been made. In Lesperance and Kalbfleisch [15] and Wang and Wang [6], instead of just adding the point at which $D(\boldsymbol{\theta}, F_k)$ occurs, all the points where local maxima occur are added in an optimal way. Again, under the assumptions of regularity, convergence as above is proved. In one dimension, these methods are efficient. In higher dimensions, these methods are not computationally practical.

We now suggest a method to check whether the final distribution of NPAG is globally optimal and, if not optimal, how close it is to the optimal. It also involves the use of the directional derivative $D(\boldsymbol{\theta}, F)$, but only at the last iteration of NPAG. Now define

$$D(F) = \max_{\boldsymbol{\theta} \in \Theta} D(\boldsymbol{\theta}, F)$$

Note that the *max* in the above expression is only over Θ and not over Θ^N. It is proved in Lindsay [7] that F^* is a global maximum of $L(F)$, i.e., F^*=F^{ML}, if and only if $D(F^*) = 0$. Even if $D(F^*) \neq 0$, it is useful to make this computation as it is also proved in Lindsay [7] that $L(F^{ML}) - L(F^*) \leq D(F^*)$, so this last expression gives an estimate of the accuracy of the final NPAG result.

Now, even though we said above it is not practical to calculate $D(F)$ at every iteration of an algorithm, we are just suggesting to make this calculation at the end of the algorithm. This calculation can be performed by a deterministic or stochastic optimization algorithm.

3. Examples

First of all, the NPAG program has been used successfully in high-dimensional and very complex pharmacokinetic–pharmacodynamic models. In Ramos-Martin et al. [30], the NPAG program was used for a population model of the pharmacodynamics of vancomycin for coagulase-negative staphylococci (CoNS) infection in neonates. Vancomycin is an antibiotic used to treat a number of serious bacterial infections. CoNS are the most commonly isolated pathogens in the neonatal intensive care unit. This model had 7 nonlinear differential equations and 11 random parameters. The population was a combination of 300 experimental and animal subjects. In Drusano et al. [31], the NPAG program was used for a population model of two drugs for the treatment of tuberculosis. This model had 5 nonlinear differential equations, 3 nonlinear algebraic equations, 1671 observations from 6

outputs and 29 random parameters. In the algebraic equations, the state variables were only defined implicitly and had to be solved for by an iterative method.

The above two examples are too complex to use for simulation purposes. Consequently, we present here a simpler model which has an analytic solution and which can be checked by other algorithms. Nevertheless, the estimation of parameters in this model is not trivial. We consider a three-compartment PK model with a continuous IV infusion into the central compartment and a bolus input into the absorption compartment. The individual subject model is described by the following differential equations:

$$\frac{dx_1}{dt} = -K_a x_1, \quad x_1(t) = \begin{cases} 0 & \text{for } 0 \le t < 5 \\ b & \text{if } t = 5 \end{cases}$$
$$\frac{dx_2}{dt} = K_a x_1 - (K_e + K_{cp}) x_2 + K_{pc} x_3 + r(t), \quad x_2(0) = 0$$
$$\frac{dx_3}{dt} = K_{cp} x_2 - K_{pc} x_3, \quad x_3(0) = 0$$

and output equation

$y_1(t) = x_2(t)/V_c + w(t), w(t) \sim N(0, \sigma^2),\ \sigma = 5.5$

The inputs are a bolus $b = 2000$ mg at $t = 5$ Hr and a continuous infusion $r(t) = 500$ Hr^{-1}, for $0 < t < 16$ Hr. This model has 5 random parameters (V, K_a, K_e, K_{cp}, K_{pc}). A diagram of this model is given in Figure A1. It is known that this model is structurally identifiable, see Godfrey [32]. However, we have found that for a continuous IV infusion, the parameters K_{cp} and K_{pc} can be difficult to estimate in a noisy environment.

The details of the simulation are as follows. There were 300 simulated subjects. The random variables $(V,\ K_a,\ K_{cp},\ K_{pc})$ were independently simulated from normal distributions with means respectively equal to $(1.2,\ 0.8,\ 2.0,\ 0.2)$ and standard deviations equal to 25% coefficient of variation.

The random variable K_e was independently simulated from a bimodal mixture of two normal distributions with means respectively equal to 0.5 and 1.5, with standard deviations equal to 10% coefficient of variation, and with weights equal to 0.2 and 0.8. This distribution would apply to an elimination rate constant with a bimodal distribution where 80% of the subjects have a mean of 1.5, and only 20% have a mean of 0.5. The power of the nonparametric method allows the detection of the 20% group.

Eleven observations were taken at times $t = 0.25,\ 1.0,\ 4.98,\ 5.25,\ 5.5,\ 6.0,\ 7.0,\ 8.5, 10.0,$ $13.0, 16.0$.

These sampling times were chosen in an ad hoc fashion and are not to be considered optimal. In Figure A2, we show the profiles of the 300 noisy model outputs y_1. These profiles are plotted as piecewise linear functions with nodes at the observation times.

The initial Faure set had $80,321$ support points dispersed in the volume

$$(K_a, V, K_e, K_{cp}, K_{pc}) \in [(0.01, 2.0), (0.01, 2.5), (0.0001, 2.0), (0.0, 4.0), (0.0001, 2.0)]$$

The assay error (Equation (8)) is not always known. An approximate assay error polynomial can be inferred from literature and Pmetrics includes a routine to estimate the assay error polynomial from the data. Another approach to analyzing data with unknown measurement error is to run successive NPAG optimizations using decreasing error magnitude for each new run. An advantage of this approach is that model development is faster. The first cycle of NPAG, which begins with a relatively large measurement error, can be initialized using a relatively small number of support points. Each NPAG solution is used as a prior to skip the first (and most computationally burdensome) step in the next NPAG run. We demonstrate this approach can converge to the correct solution on this simulated data.

Convergence for this problem was accomplished after applying NPAG four times. For the first application, $\sigma = 0.025 Y_{simulated}$. The output distribution of this first application was used as a prior to start NPAG again, this time with $\sigma = 7.96 + 0.0125 Y_{simulated}$. The

output of this second application was used as a prior to start a third NPAG run, this time with $\sigma = 7.96 + 0.0065Y_{simulated}$. Finally, the output density of this third run was used as a prior to run NPAG a fourth time, this time with $\sigma = 5.5$, the same as for the simulation. The step down in assumed observation error happened at convergence for each previous error levels: at cycles 4513, 5972, and 6791. Final convergence happened at cycle 8012. There are 284 support points in the final density.

The simulated and estimated marginal distributions are shown in Figures A3 and A4. It is seen that the estimated marginal distributions are similar to the simulated histograms. In particular, the bimodal shape of K_e was uncovered. Similarity is tested using the R routine mtsknn.eq(..., k = 3), which returns a $pval = 0.5809$. mtsknn.eq applies a $K-$nearest neighbors approach to estimate the probability that two non-parametric distributions arise from the same distribution.

NPAG is designed to estimate the whole joint distribution of the parameters. As mentioned earlier, the estimate F^{ML} is especially important for our application to population pharmacokinetics where F^{ML} is used as a prior distribution for Bayesian dosage regimen design. However, F^{ML} is a consistent estimator of the true mixing distribution and consequently, the moments of F^{ML} should be consitent estimators of the true moments. Means and variances of parameter estimates for F^{ML} can be easily obtained by integrating the corresponding marginal distributions. So as a check of this fact, in Table A1, the comparisons of estimated versus simulated means and variances are shown. Again, results are quite accurate (see Tables A1 and A2).

Finally, in Figure A5 we include a graph of Predicted versus Observed values which shows the all around good fit of the data. The predicted (right panel: Predicted Bayesian) values are gotten as follows: For each subject, the Bayesian mean estimate of the parameters are found using the final NPAG distribution as a prior and that subject's observations. Then, based on these parameter means, the subject's concentration profile is calculated. The predicted (left panel: Predicted Population) is the weighted average of the evaluation of all support points for subject model.

4. Conclusions

We have provided the first comprehensive description of the NPAG algorithm for estimating multivariate mixing distributions. This algorithm can describe between subject variability without assuming any shape for the distribution of PK parameters and is excellently suited for optimizing patient dosing [33–35]; NPAG is based on an iterative algorithm employing the Primal-Dual Interior-Point method and an Adaptive Grid method. This approach can handle pharmacokinetic, pharmacodynamic and other models with a high number of estimated model parameters and is based on the exact log-likelihood. A detailed description of NPAG is provided along with an application for a common two-compartment PK models. Finally, the NPAG algorithm is arguably the most efficiently parallelizable population modeling algorithm, since it can parallelize over both subjects and support points. This allows one to readily implement this algorithm on supercomputers as our laboratory has done on research projects. In addition to population pharmacokinetics, this research also applies to empirical Bayes estimation, see Koenker and Mizera [36] and to many other areas of applied mathematics, see Banks et al. [37]. Overall, the NPAG algorithm provides an important addition to the pharmacometric toolbox for drug development and optimal patient dosing at the interface of applied mathematics, biomedical scientists, and clinicians.

Author Contributions: Conceptualization, J.V.B., R.L. and A.S.; data curation, M.v.G., M.N.N. and W.M.Y.; formal analysis, J.V.B. and A.S.; funding acquisition, J.V.B., R.W.J. and M.N.N.; investigation, M.N.N. and R.W.J.; methodology, R.L., T.T. and A.S.; project administration, A.S.; resources, M.N.N. and R.J; software, M.v.G., A.K. and W.M.Y.; supervision, A.S.; validation, J.B., D.S.B., J.V.B., A.K., R.L., T.T., A.S. and W.M.Y.; visualization, W.M.Y.; writing—original draft preparation, W.M.Y.; writing—review and editing, J.B., D.S.B., J.V.B., A.K., R.L., T.T., A.S. and W.M.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by grants from NIH: RR11526, GM65619, GM068968, EB005803, EB001978, HD070886. JB was supported in part by NSF/DMS-0505712.

Data Availability Statement: Data is available on request from `contact@lapk.org`.

Acknowledgments: The NPAG program was developed at the USC Laboratory of Applied Pharmacokinetics. James Burke (University of Washington) developed the Primal-Dual Interior-Point method discussed in the Appendix. Robert Leary (Pharsight Corporation) developed the Adaptive Grid method and wrote the original Fortran program for NPAG. Michael Neely, MD (USC Children's Hospital of Los Angeles) developed the program package Pmetrics which contains NPAG as a subprogram. Pmetrics is an R package for nonparametric and parametric population modeling and simulation and is available at `www.lapk.org`, see Neely et al. [24].

5. Supplementary Material

5.1. Recent References Using Npag

As a means for readers to more quickly assess the applicability of NPAG in a wide variety of pharmacometric studies, we refer here to the first 10 references on a recent PubMed search for papers that use NPAG . Note that all population PK studies using Pmetrics employ NPAG.

1. Pharmacodynamics of Posaconazole in Experimental Invasive Pulmonary Aspergillosis: Utility of Serum Galactomannan as a Dynamic Endpoint of Antifungal Efficacy. Gastine S, Hope W, Hempel G, Petraitiene R, Petraitis V, Mickiene D, Bacher J, Walsh TJ, Groll AH. Antimicrob Agents Chemother. 2020 Nov 9:AAC.01574-20. doi: 10.1128/AAC.01574-20. Online ahead of print. PMID: 33168606
2. Cerebrospinal fluid penetration of ceftolozane/tazobactam in critically ill patients with an indwelling external ventricular drain. Sime FB, Lassig-Smith M, Starr T, Stuart J, Pandey S, Parker SL, Wallis SC, Lipman J, Roberts JA. Antimicrob Agents Chemother. 2020 Oct 19:AAC.01698-20. doi: 10.1128/AAC.01698-20. Online ahead of print. PMID: 33077655
3. Population Pharmacokinetics of Continuous-Infusion Meropenem in Febrile Neutropenic Patients with Hematologic Malignancies: Dosing Strategies for Optimizing Empirical Treatment against Enterobacterales and P. aeruginosa. Cojutti PG, Candoni A, Lazzarotto D, Filì C, Zannier M, Fanin R, Pea F. Pharmaceutics. 2020 Aug 19;12(9):785. doi: 10.3390/pharmaceutics12090785. PMID: 32825109 Free PMC article.
4. Caspofungin Weight-Based Dosing Supported by a Population Pharmacokinetic Model in Critically Ill Patients. Märtson AG, van der Elst KCM, Veringa A, Zijlstra JG, Beishuizen A, van der Werf TS, Kosterink JGW, Neely M, Alffenaar JW. Antimicrob Agents Chemother. 2020 Aug 20;64(9):e00905-20. doi: 10.1128/AAC.00905-20. Print 2020 Aug 20. PMID: 32660990 Free PMC article.
5. Ethionamide Population Pharmacokinetic Model and Target Attainment in Multidrug-Resistant Tuberculosis. Al-Shaer MH, Märtson AG, Alghamdi WA, Alsultan A, An G, Ahmed S, Alkabab Y, Banu S, Houpt ER, Ashkin D, Griffith DE, Cegielski JP, Heysell SK, Peloquin CA. Antimicrob Agents Chemother. 2020 Aug 20;64(9):e00713-20. doi: 10.1128/AAC.00713-20. Print 2020 Aug 20. PMID: 32631828
6. Development and validation of a dosing nomogram for amoxicillin in infective endocarditis. Rambaud A, Gaborit BJ, Deschanvres C, Le Turnier P, Lecomte R, Asseray-Madani N, Leroy AG, Deslandes G, Dailly É, Jolliet P, Boutoille D, Bellouard R, Gregoire M; Nantes Anti-Microbial Agents PK/PD (NAMAP) study group. J Antimicrob Chemother. 2020 Oct 1;75(10):2941-2950. doi: 10.1093/jac/dkaa232. PMID: 32601687
7. Population Pharmacokinetics and Target Attainment of Cefepime in Critically Ill Patients and Guidance for Initial Dosing. Al-Shaer MH, Neely MN, Liu J, Cherabuddi K, Venugopalan V, Rhodes NJ, Klinker K, Scheetz MH, Peloquin CA. Antimicrob Agents Chemother. 2020 Aug 20;64(9):e00745-20. doi: 10.1128/AAC.00745-20. Print 2020 Aug 20. PMID: 32601155

8. Baclofen self-poisoning: Is renal replacement therapy efficient in patient with normal kidney function? Brunet M, Léger M, Billat PA, Lelièvre B, Lerolle N, Boels D, Le Roux G. Anaesth Crit Care Pain Med. 2020 Oct 14:S2352-5568(20)30230-7. doi: 10.1016/j.accpm.2020.07.021. Online ahead of print. PMID: 33068797
9. Ceftriaxone dosing in patients admitted from the emergency department with sepsis. Heffernan AJ, Curran RA, Denny KJ, Sime FB, Stanford CL, McWhinney B, Ungerer J, Roberts JA, Lipman J. Eur J Clin Pharmacol. 2020 Sep 24. doi: 10.1007/s00228-020-03001-z. Online ahead of print. PMID: 32974748
10. Population Pharmacokinetic Models of Anti-Tuberculosis Drugs in Patients: a Systematic Critical Review. Otalvaro JD, Hernandez B E AM, Rodriguez CA, Zuluaga AF. Ther Drug Monit. 2020 Sep 18. doi: 10.1097/FTD.0000000000000803. Online ahead of print. PMID: 32956238

5.2. Recent Studies to Which NPAG Can Be Applied

The references below use excellent parametric methods. All of these methods have the same mathematical structure as our nonparametric NPAG. The main difference is that in the parametric methods, it is assumed that the population distribution is multivariate normal with unknown mean vector and unknown covariance matrix. In NPAG, we do not make any parametric assumptions about the population distribution, as discussed in our paper. Otherwise, the population analysis problem is the same. [38] and [39] use the method S-ADAPT. S-ADAPT is based on the ADAPT package developed by D'Argenio, Wang and Schumitzky. [40] and [41] use the industry standard parametric version of NONMEM. [42] uses the stochastic approximation expectation maximization method (SAEM). NPAG can run any problem that is run by any of these programs.

5.3. Comparison of NPAG to Classical Population Analysis Programs

Several studies have compared NPAG to a parametric algorithm. Bustad et al. [43] compared the statistical consistency and efficiency of ITS and two older NONMEM® routines, FO and FOCE to that of NPEM and NPAG on simulated datasets; the nonparametric methods were more consistent and efficient. Prémaud et al. [44] also compared NPAG to NONMEM® FOCE, but the two algorithms converged to distinctly different results. NPAG converged to an estimator with better predictive performance allowing for its use in therapeutic drug monitoring. More recently, de Velde et al. [45] compared NONMEM® FOCE-I to NPAG, with both methods converging to similar parameter estimates. In each of the above studies, NPAG converged to a distribution with greater variance.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

AG	Adaptive grid
CoNS	Coagulase-negative *Staphylococci*
EM	Expectation–maximization algorithm
ISDM	Intra-simplex direction method
NPAG	Nonparametric adaptive grid algorithm
NPML	Nonparametric maximum likelihood
PDIP	Primal-dual interior-point method
QP	Quadratic programming

Appendix A. A Primal-Dual Interior-Point Algorithm (PDIP)

To make this paper self-contained, we outline here the PDIP algorithm which was written by James Burke. This algorithm is a FORTRAN subroutine of NPAG. The descrip-

tion below is based on the Matlab and C++ codes found in Bradley Bell's website, see [10]. Definition of general terms and theorems can be found in Boyd and Vandenberghe [9].

Appendix A.1. Duality Theory and the Basic Problem

Given a set of support points $\{\phi_k\}$, the problem of finding the optimal weights $\{\lambda_k\}$ in Equation (4) can be posed as the following optimization problem

$$\mathcal{P} \quad \min \Phi(\boldsymbol{\Psi}\boldsymbol{\lambda}) \text{ s.t. } 0 \leq \boldsymbol{\lambda},\ \boldsymbol{e}^\intercal \boldsymbol{\lambda} = 1,$$

where $\boldsymbol{\Psi} \in \mathbb{R}^{n \times m}$ is the matrix whose (i, j) entry is $p(y_i \mid \phi_j)$ and where in general, the function $\Phi : \mathbb{R}^k \mapsto \mathbb{R} \cup \{+\infty\}$ is given by

$$\Phi(z) = \begin{cases} -\sum_{i=1}^{k} \log z_i & , 0 < z, \text{ and} \\ +\infty & , \text{otherwise.} \end{cases} \tag{A1}$$

The symbol $\boldsymbol{e}$ is always to be interpreted as the vector of all ones of the appropriate dimension.

The problem $\mathcal{P}$ is a convex programming problem since the objective function Φ is convex and the constraining region is a convex set. The Fenchel–Rockafellar dual of the convex program $\mathcal{P}$ is the problem

$$\mathcal{D} \quad \min \Phi(\boldsymbol{\omega}) \quad \text{s.t. } \boldsymbol{L}^\intercal \boldsymbol{\omega} \leq m\boldsymbol{e}.$$

From Boyd, we obtain the following Karush–Kuhn–Tucker (KKT) equations relating the solutions to the problem $\mathcal{P}$ and $\mathcal{D}$.

$$m\boldsymbol{e} = \boldsymbol{\Psi}^\intercal \boldsymbol{w} + \boldsymbol{y} \tag{A2}$$
$$\boldsymbol{e} = \boldsymbol{W}\boldsymbol{\Psi}\boldsymbol{\lambda} \tag{A3}$$
$$0 = \boldsymbol{\Lambda}\boldsymbol{Y}\boldsymbol{e} \tag{A4}$$

where for any vector $\boldsymbol{x}$, we define $\boldsymbol{X}$ to be the diagonal matrix having $\boldsymbol{x}$ along the diagonal.

Appendix A.2. An Interior-Point Path-Following Algorithm

The relaxed KKT is given by

$$m\boldsymbol{e} = \boldsymbol{\Psi}^\intercal \boldsymbol{w} + \boldsymbol{y} \tag{A5}$$
$$\boldsymbol{e} = \boldsymbol{W}\boldsymbol{\Psi}\boldsymbol{\lambda} \tag{A6}$$
$$\mu\boldsymbol{e} = \boldsymbol{\Lambda}\boldsymbol{Y}\boldsymbol{e} \tag{A7}$$
$$0 \leq \boldsymbol{\lambda},\ 0 \leq \boldsymbol{w},\ 0 \leq \boldsymbol{y}, \tag{A8}$$

for $\mu > 0$. (μ is the relaxation parameter.) A damped Newton's method is used to solve the above system.

Consider the function $F : \mathbb{R}^{2m+n} \mapsto \mathbb{R}^{2m+n}$ given by

$$F(\boldsymbol{\lambda}, \boldsymbol{w}, \boldsymbol{y}) = \begin{bmatrix} \boldsymbol{\Psi}^\intercal \boldsymbol{w} + \boldsymbol{y} \\ \boldsymbol{W}\boldsymbol{\Psi}\boldsymbol{\lambda} \\ \boldsymbol{\Lambda}\boldsymbol{Y}\boldsymbol{e} \end{bmatrix}.$$

A triple $(\boldsymbol{\lambda}, \boldsymbol{w}, \boldsymbol{y})$ solves Equations (A5) to (A8) if and only if

$$F(\boldsymbol{\lambda}, \boldsymbol{w}, \boldsymbol{y}) = \begin{pmatrix} m\boldsymbol{e} \\ \boldsymbol{e} \\ \mu\boldsymbol{e} \end{pmatrix} \tag{A9}$$

and $0 \leq \boldsymbol{\lambda}$, $0 \leq \boldsymbol{\omega}$, and $0 \leq \boldsymbol{y}$. Path-following algorithms attempt to solve A9 by applying Newton's method for progressively smaller values of the relaxation parameter μ. We first need the derivative of F. It follows

$$F'(\boldsymbol{\lambda}, \boldsymbol{\omega}, \boldsymbol{y}) = \begin{bmatrix} 0 & \boldsymbol{\Psi}^\intercal & I \\ \boldsymbol{W}\boldsymbol{\Psi} & \boldsymbol{Z} & 0 \\ \boldsymbol{Y} & 0 & \boldsymbol{\Lambda} \end{bmatrix}$$

where $\boldsymbol{z} = \boldsymbol{\Psi}\boldsymbol{\lambda}$.

At the kth iteration of the algorithm, the Newton step is given by the solution to the nonsingular linear system

$$F\left(\boldsymbol{\lambda}^k, \boldsymbol{w}^k, \boldsymbol{y}^k\right) + F'\left(\boldsymbol{\lambda}^k, \boldsymbol{w}^k, \boldsymbol{y}^k\right) * \left[\boldsymbol{\Lambda}^k, \boldsymbol{W}^k, \boldsymbol{Y}^k\right]^\intercal = \left[\boldsymbol{e}_m, \boldsymbol{e}_n, \mu^k \boldsymbol{e}_m\right]^\intercal \tag{A10}$$

where $\boldsymbol{y}$ is constrained to satisfy the first KKT condition $\boldsymbol{y}^k = \boldsymbol{e}_m - \boldsymbol{\Psi}^\intercal \boldsymbol{w}^k$.

The above set of equations can be reduced by standard techniques. It follows:

$$\Delta \boldsymbol{w} = \boldsymbol{H}^{-1} \boldsymbol{r}_2 \tag{A11}$$

$$\Delta \boldsymbol{y} = -\boldsymbol{\Psi} \Delta \boldsymbol{\omega} \tag{A12}$$

$$\Delta \boldsymbol{\lambda} = \boldsymbol{r}_1 - \boldsymbol{\lambda} - \boldsymbol{D}_1 \Delta \boldsymbol{y} \tag{A13}$$

where $\boldsymbol{H} = \boldsymbol{D}_2 - \boldsymbol{\Psi} \boldsymbol{D}_1 \boldsymbol{\Psi}^\intercal$, $\boldsymbol{D}_2 = \boldsymbol{Z}\boldsymbol{W}^{-1}$, $\boldsymbol{D}_1 = \boldsymbol{\Lambda}\boldsymbol{Y}^{-1}$, $\boldsymbol{r}_1 = \mu \boldsymbol{Y}^{-1} \boldsymbol{e}$, $\boldsymbol{r}_2 = \boldsymbol{W}^{-1}\boldsymbol{e} - \boldsymbol{\Psi}\boldsymbol{r}_1$ where the superscript k is suppressed for simplicity.

Appendix A.3. The Algorithm

To describe the algorithm, we need to define the variables: $q = \frac{1}{m}\sum_{i=1}^{m} \lambda_i y_i$ $\rho = \|\boldsymbol{e} - \boldsymbol{W}\boldsymbol{Z}\boldsymbol{e}\|_\infty$ and the scaled duality gap $\gamma = \frac{|\Phi(\boldsymbol{\omega}) + \Phi(\boldsymbol{\Psi}\boldsymbol{\lambda})|}{1 + |\Phi(\boldsymbol{\Psi}\boldsymbol{\lambda})|}$.

Appendix A.3.1. Initialization

Initially choose $\boldsymbol{\lambda}^0 = \boldsymbol{e}_m / m$, $\boldsymbol{w}^0 = \boldsymbol{e}_n / \boldsymbol{\Psi}\boldsymbol{\lambda}^0$, and $\boldsymbol{y}^0 = \boldsymbol{e}_m - \boldsymbol{\Psi}^\intercal \boldsymbol{w}^0$. (Division of two vectors is performed component-wise.) Set $\varepsilon = 10^{-8}$.

Appendix A.3.2. Iteration

At iteration $k+1$, set
$\mu^{k+1} = \sigma^k q^k$
where the reduction factor σ is defined by

$$\sigma = \begin{cases} 1 & \text{, if } \mu \leq \varepsilon \text{ and } \rho > \varepsilon, \\ \min(0.3, (1-\delta_1)^2), (1-\delta_2)^2, \frac{|\rho - \mu|}{\rho + 100\tau} & \text{, otherwise.} \end{cases}$$

The next iterates are given by $\boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^k + \delta_1[\Delta \boldsymbol{\lambda}^k]$, $\boldsymbol{\omega}^{k+1} = \boldsymbol{\omega}^k + \delta_2[\Delta \boldsymbol{\omega}^k]$ and $\boldsymbol{y}^{k+1} = \boldsymbol{y}^k + \delta_2[\Delta \boldsymbol{y}^k]$, where the "damping" factors δ_1 and δ_2 are defined by

$$\delta_{1,0} = -\left[\min(\min(\boldsymbol{\Lambda}^{-1}\Delta\boldsymbol{\lambda}), -\frac{1}{2})\right]^{-1}$$

$$\delta_{2,0} = -\left[\min(\min(\boldsymbol{Y}^{-1}\Delta\boldsymbol{y}), \min(\boldsymbol{W}^{-1}\Delta\boldsymbol{w}), -\frac{1}{2})\right]^{-1}$$

$$\delta_1 = \min(1, 0.99995\delta_{1,0})$$

$$\delta_2 = \min(1, 0.99995\delta_{2,0})$$

Appendix A.3.3. Exit Conditions

Iterate Equations (A11)–(A13) until $\mu \leq \varepsilon$ and $\rho \leq \varepsilon$ and $\gamma \leq \varepsilon$.

If these conditions are not satisfied after a set number of iterations, then write "PDIP did not converge in the given number of iterations".

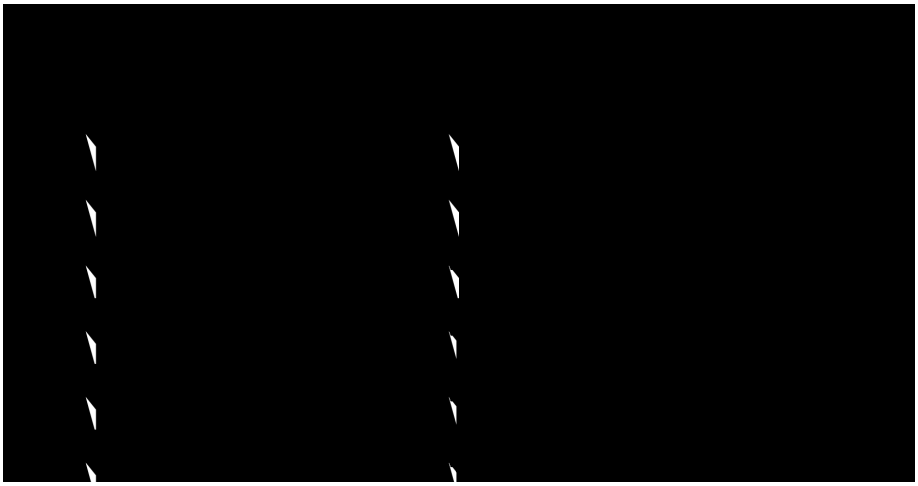
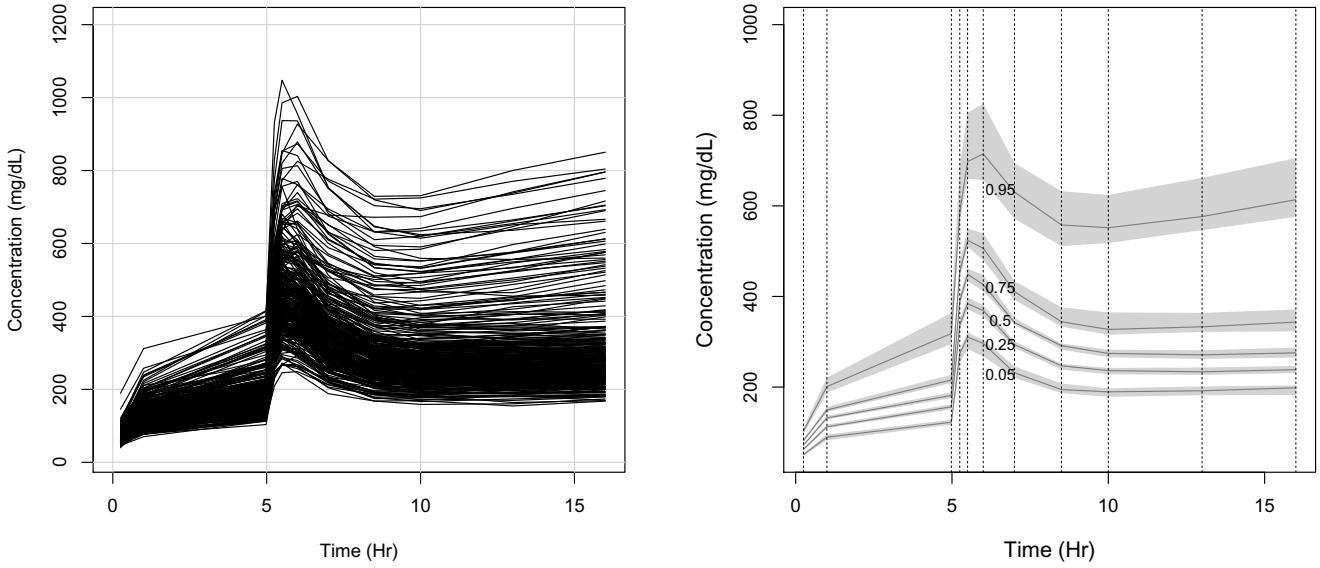


Figure A1. Model.



(**a**) Simulated model profiles.

(**b**) Percent tiles of the simulated model profiles at the observation times.

Figure A2. The 300 simulated observed profiles, frame a, are generated in Pmetrics using the function SIMrun(). The 0.05, 0.25, 0.5, 0.75, and 0.95 percent tiles of the profiles are plotted in frame b, with the $CI_{95\%}$ in grey. The observation times are marked by a vertical dotted line in frame b. Other details are in the text. Of note: inspection of these profiles does not suggest a bimodal elimination parameter.

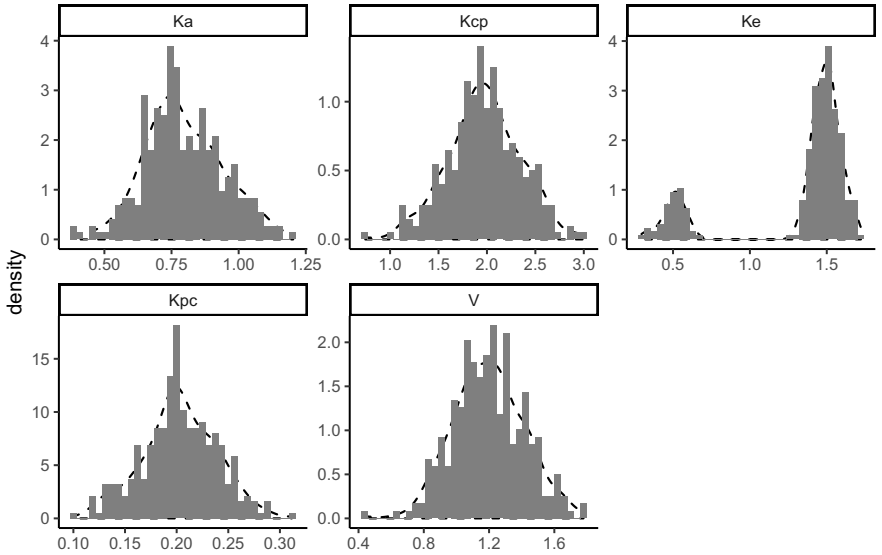


Figure A3. Kernel densities of simulated PK parameters. All K are in Hr^{-1}, Volume is in dL. Kernel density (dashed line) of the distribution is over-plotted on the histogram (grey), which are normalized to the total observations. Simulated support points are generated in Pmetrics using the function SIMrun(). Details are in the text. Kernel densities are calculated in R using the (S3) generic function density().

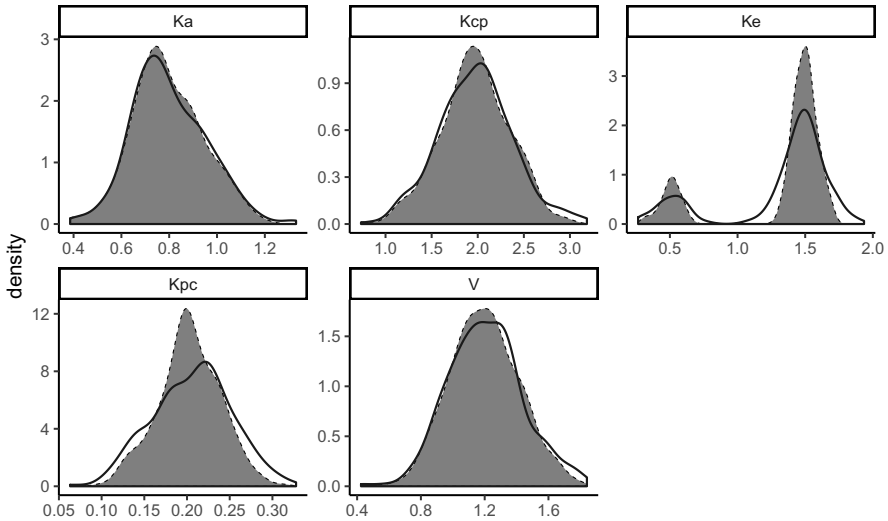


Figure A4. Kernel densities of NPAG estimated PK parameters (black line) vs. Simulated r.v.s (dashed with grey fill). K are in Hr^{-1}, volume is in dL. NPAG estimation is calculated using the NPrun() function in Pmetrics. Similarity of these two distributions is verified in R using the function mtsknn.eq(). See text for further detail.

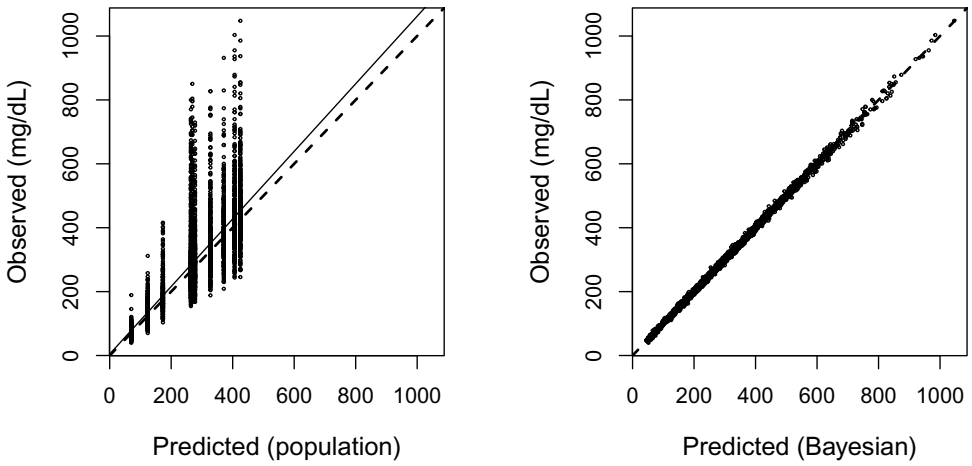


Figure A5. Predicted vs. Observed. Population Fit: $r^2 = 0.569$, intercept $= 3.8(CI_{95\%} - 5.85, 13.5)$, slope $= 1.12(CI_{95\%}1.08, 1.15)$,bias $= -6.45$, imprecision $= 366$. Individual Bayesian Posterior Fit: $r^2 = 0.999$, intercept $= 0.168(CI_{95\%} - 0.267, 0.603)$, slope $= 0.998(CI_{95\%}0.997, 1)$, bias $= 0.0612$, imprecision $= 1.15$.

Table A1. Simulation versus optimization. Row 1: True simulated means for each parameter. Row 2: NPAG estimates of corresponding means.

	Ka	**Vc**	**Ke**	**Kcp**	**Kpc**
μ_{SIM}	0.7948080	1.1982732	1.3162403	1.9700861	0.2028168
μ_{NPAG}	0.8003521	1.2054767	1.3108326	1.9780514	0.2059419

Table A2. Parameter Covariance Matrices.

Simulation	**Ka**	**V**	**Ke**	**Kcp**	**Kpc**
Ka	0.0215919795				
V	0.0011916465	0.0468421011			
Ke	−0.0015398993	−0.0005839048	0.1563336061		
Kcp	0.0009717487	−0.0032613149	−0.0019679155	0.1391046100	
Kpc	−0.0000572936	0.0004080801	0.0007402169	0.0004695783	0.0013055173
NPAG	**Ka**	**V**	**Ke**	**Kcp**	**Kpc**
Ka	0.0241232043				
V	0.0038827461	0.0537290654			
Ke	−0.0009243608	−0.0075706783	0.1686441632		
Kcp	−0.0017343788	−0.0083046624	−0.0022665464	0.1617627081	
Kpc	0.0006769795	0.0005601072	0.0027358931	0.0007782649	0.0021142178

References

1. Sheiner, L.; Beal, S. Evaluation of Methods for Estimating Population Pharmacokinetic Parameters. I. Biexponential Model and Experimental Pharmacokinetic Data. *J. Pharmacokinet. Biopharm.* **1980**, *8*, 553–571.
2. Sheiner, L.; Beal, S. Evaluation of Methods for Estimating Population Pharmacokinetic Parameters. II. Michaelis–Menten Model: Routine Clinical Pharmacokinetic Data. *J. Pharmacokinet. Biopharm.* **1981**, *9*, 635–651.
3. Bauer, R.; Guzy, S.; Ng, C. A Survey of Population Analysis Methods and Software for Complex Pharmacokinetic and Pharmacodynamic Models with Examples. *AAPS J.* **2007**, *9*, E60–E83. doi:10.1208/aapsj0901007.

4. Kiefer, J.; Wofowitz, J. Consistency of the Maximum Likelihood Estimator in the Presence of Infinitely many Incidental Parameters. *Ann. Math. Statist.* **1956**, *27*, 887–906.
5. Jelliffe, R.; Bayard, D.; Milman, M.; Guilder, M.V.; Schumitzky, A. Achieving Target Goals most Precisely using Nonparametric Compartmental Models and 'Multiple Model' Design of Dosage Regimens. *Ther. Drug Monit.* **2000**, *22*, 346–353.
6. Wang, X.; Wang, Y. Nonparametric multivariate density estimation using mixtures. *Stat. Comput.* **2015**, *25*, 33–43.
7. Lindsay, B.G. The Geometry of Mixture Likelihoods: A general theory. *Ann. Statist.* **1983**, *11*, 86–94.
8. Mallet, A. A Maximum Likelihood Estimation Method for Random Coefficient Regression Models. *Biometrika* **1986**, *73*, 645–656.
9. Boyd, S.; Vandenberghe, L. *Convex Optimization*; Cambridge University Press: Cambridge, UK, 2004.
10. Bell, B. Non-Parametric Population Analysis. Personal communication, Seattle, Washington, 2012.
11. Baek, Y. An Interior Point Approach to Constrained Nonparametric Mixture Models. Ph.D. dissertation, Department of Mathematics, University of Washington, Seattle, Washington, USA, 2006.
12. Yamada, W.; Bartroff, J.; Bayard, D.; Burke, J.; van Guilder, M.; Jelliffe, R.; Leary, R.; Neely, M.; Kryshchenko, A.; Schumitzky, A. *The Nonparametric Adaptive Grid Algorithm for Population Pharmacokinetic Modeling*; Technical Report TR-2014-1; Children's Hospital Los Angeles: Los Angeles, CA, USA, 2014.
13. Fedorov, V.V. *Theory of Optimal Experiments*; Studden, W.J., Klimko, E.M., Eds.; Academic Press: New York, New York, USA, 1972.
14. Schumitzky, A. Nonparametric EM Algorithms For Estimating Prior Distributions. *Appl. Math. Comput.* **1991**, *45*, 143–157.
15. Lesperance, M.L..; Kalbfleisch, J.D. An algorithm for computing the nonparametric MLE of a mixing distribution. *J. Am. Stat. Assoc.* **1992**, *87*, 120–126.
16. Pilla, R.S.; Bartolucci, F.; Lindsay, B.G. Model building for semiparametric mixtures. *arXiv* **2006**, arXiv:math/0606077
17. Savic, R.M.; Kjellsson, M.C.; Karlsson, M.O. Evaluation of the nonparametric estimation method in NONMEM VI. *Eur. J. Pharm. Sci.* **2009**, *37*, 27–35.
18. Savic, R.M.; Karlsson, M.O. Evaluation of an extended grid method for estimation using nonparametric distributions. *AAPS J.* **2009**, *11*, 615–627.
19. Leary, R. An overview of nonparametric estimation methods used in population analysis. In *Abstracts of the Annual Meeting of the Population Approach Group in Europe 2017*; Number Abstract 7383; Population Analysis Group Europe (PAGE); https://www.page-meeting.org; p. 26.
20. Leary, R.; Jelliffe, R.; Schumitzky, A.; Guilder, M.V. An Adaptive Grid Non-Parametric Approach to Pharmacokinetic and Dynamic (PK/PD) Population Models. In Proceedings of the 14th IEEE Symposium on Computer-Based Medical Systems (CBMS'01), Bethesda, MD, USA, 26–27 March 2001; p. 0389. doi:10.1109/CBMS.2001.941750.
21. Tatarinova, T.; Schumititzky, A. *Nonlinear Mixture Models: A Bayesian Approach*; Imperial College Press: London, UK, 2015.
22. Jordan-Squire, C. Convex Optimization over Probability Measures. Ph.D. Thesis, Department of Mathematics, University of Washington, Washington, DC, USA, 2015.
23. Neely, M.; Philippe, M.; Rushing, T.; Fu, X.; van Guilder, M.; Bayard, D.; Schumitzky, A.; Bleyzac, N.; Goutelle, S. Accurately Achieving Target Busulfan Exposure in Children and Adolescents With Very Limited Sampling and the BestDose Software. *Ther. Drug Monit.* **2016**, *38*, 332–342. doi:10.1097/FTD.0000000000000276.
24. Neely, M.; van Guilder, M.; Yamada, W.; Schumitzky, A.; Jelliffe, R. Accurate Detection of Outliers and Subpopulations with Pmetrics: a non-parametric and parametric pharmacometric package for R. *Ther. Drug Monit.* **2012**, *34*, 467–476.
25. Faure, H. Discrépance de suites associées á un système de numération (en dimension s). *Acta Arith.* **1982**, *41*, 337–351.
26. Bratley, P.; Fox, B.L. Algorithm 659: Implementing Sobol's Quasirandom Sequence Generator. *ACM Trans. Math. Softw.* **1988**, *14*, 88–100.
27. Fox, B.L. Algorithm 647: Implementation and Relative Efficiency of Quasirandom Sequence Generators. *ACM Trans. Math. Softw.* **1986**, *12*, 362–376.
28. Davidian, M.; Giltinan, D.M. *Nonlinear Models for Repeated Measurement Data*; Chapman and Hall/CRC Press: Boca Raton, Florida, USA 1995.
29. Davidian, M.; Giltinan, D.M. Nonlinear Models for Repeated Measurement Data: An overview and update. *J. Agric. Biol. Environ. Stat.* **2003**, *8*, 387–419.
30. Ramos-Martin, V.; Johnson, A.; Livermore, J.; McEntee, L..; Goodwin, J.; Whalley, F..; Docobo-Perez, F.; Felton, T.W.; Zhao, W.; Jacqz-Aigrain, E.; Sharland, M.; Turner, M.; Hope, W.W. Pharmacodynamics of vancomycin for CoNS infection: experimental basis for optimal use of vancomycin in neonates. *J. Antimicrob. Chemother.* **2016**, *71*, 992–1002.
31. Drusano, G.; Neely, M.; van Guilder, M..; Schumitzky, A.; Brown, D.; Fikes, S..; Peloquin, C.; Louie, A. Analysis of combination drug therapy to develop regimens with shortened duration treatment for tuberculosis. *PLoS ONE* **2014**, *9*, e101311.
32. Godfrey, K.R. The identifiability of parametric models used in biomedicine. *Math. Model.* **1986**, *7*, 1195–1214.
33. Åsberg, A.; Midtvedt, K.; van Guilder, M.; Størset, E.; Bremer, S.; Bergan, S.; Jelliffe, R.; Hartmann, A.; Neely, M. Inclusion of CYP3A5 genotyping in a nonparametric population model improves dosing of tacrolimus early after transplantation. *Transpl. Int. Off. J. Eur. Soc. Organ Transplant.* **2013**, *26*, 1198–1207. doi:10.1111/tri.12194.
34. Størset, E.; Åsberg, A.; Skauby, M.; Neely, M.; Bergan, S.; Bremer, S.; Midtvedt, K. Improved Tacrolimus Target Concentration Achievement Using Computerized Dosing in Renal Transplant Recipients–A Prospective, Randomized Study. *Transplantation* **2015**, *99*, 2158–2166. doi:10.1097/TP.0000000000000708.

35. D.S. Bayard, D.; Neely, M. Experiment design for nonparametric models based on minimizing Bayes Risk: application to voriconazole. *J. Pharmacokinet. Pharmacodyn.* **2017**, *44*, 95–111.
36. Koenker, R.; Mizera, I. Convex optimization, shape constraints, compound decisions, and empirical Bayes rules. *J. Am. Stat. Assoc.* **2014**, *109*, 674–6 85.
37. Banks, H.T.; Kenz, Z.R.; Thompson, W.C. A Review of Selected Techniques in Inverse Problem Nonparametric Probability Distribution Estimation. *J. Inverse -Ill-Posed Probl.* **2012**, *20*, 429–460.
38. Bulitta, J.B.; Jiao, Y.; Landersdorfer, C.B.; Sutaria, D.S.; Tao, X.; Shin, E.; H´ohl, R.; Holzgrabe, U.; Stephan, U.; Sorgel, F. Comparable Bioavailability and Disposition of Pefloxacin in Patients with Cystic Fibrosis and Healthy Volunteers Assessed via Population Pharmacokinetics. *Pharmaceutics* **2019**, *11*, 323–338. doi:10.3390/pharmaceutics11070323.
39. Shah, N.R.; Bulitta, J.B.; Kinzig, M.; Landersdorfer, C.B.; Jiao, Y.; Sutaria, D.S.; Tao, X.; H´ohl, R.; Holzgrabe, U.; Kees, F.; Stephan, U.; S´orgel, F. Novel population pharmacokinetic approach to explain the differences between cystic fibrosis patients and healthy volunteers via protein binding. *Pharmaceutics* **2019**, *11*, 286. doi:10.3390/pharmaceutics11060286.
40. Ishihara, N.; Nishimura, N.; Ikawa, K.; Karino, F.; Miura, K.; Tamaki, H.; Yano, T.; Isobe, T.; Morikawa, N.; Naora, K. Population Pharmacokinetic Modeling and Pharmacodynamic Target Attainment Simulation of PiperacillinTazobactam for Dosing Optimization in Late Elderly Patients with Pneumoni. *Antibiotics* **2020**, *9*, 113. doi:10.3390/antibiotics9030113.
41. Soraluce, A.; Barrasa, H.; Asín-Prieto, E.; Sánchez-Izquierdo, J.Á.; Maynar, J.; Isla, A.; Rodríguez-Gascón, A. Novel Population Pharmacokinetic Model for Linezolid in Critically Ill Patients and Evaluation of the Adequacy of the Current Dosing Recommendation. *Pharmaceutics* **2020**, *12*, 54. doi:10.3390/pharmaceutics12010054.
42. Allard, Q.; Djerada, Z.; Pouplard, C.; Repessé, Y.; Desprez, D.; Galinat, H.; Frotscher, B.; Berger, C.; Harroche, A.; Ryman, A.; Flaujac, C.; Chamouni, P.; Guillet, B.; Volot, F.; Szymezak, J.; Nguyen, P.; Cazaubon, Y. Real Life Population Pharmacokinetics Modelling of Eight Factors VIII in Patients with Severe Haemophilia A: Is It Always Relevant to Switch to an Extended Half-Life? *Pharmaceutics* **2020**, *12*, 380. doi:10.3390/pharmaceutics12040380.
43. Bustad, A.; Terziivanov, D.; Leary, R.; Port, R.; Schumitzky, A.; Jelliffe, R. Parametric and Nonparametric Population Methods: Their comparative performance in analysing a clinical dataset and two Monte Carlo simulation studies. *Clin. Pharmacokinet.* **2006**, *45*, 365–383.
44. Prémaud, A.; Weber, L.T.; Tönshoff, B.; Armstrong, V.; Oellerich, M.; Urien, S.; Marquet, P.; Rousseau, A. Population pharmacokinetics of mycophenolic acid in pediatric renal transplant patients using parametric and nonparametric approaches. *Pharmacol. Res.* **2011**, *63*, 216–224.
45. de Velde, F.; de Winter, B.C.M.; Neely, M.N.; Yamada, W.M.; Koch, B.C.P.; Harbarth, S.; von Dach, E.; van Gelder, T.; Huttner, A.; Mouton, J.W. Population Pharmacokinetics of Imipenem in Critically Ill Patients: A Parametric and Nonparametric Model Converge on CKD-EPI Estimated Glomerular Filtration Rate as an Impactful Covariate. *Clin. Pharmacokinet.* **2020**, *59*, 885–898. doi:10.1007/s40262-020-00859-1.