

DiffMEM: an open-source package for rapid pharmacometric model estimation

Tom Haber¹, Tom Jacobs², Xavier Woot de Trixhe² and Frank van Reeth¹

¹Hasselt University, Belgium

²Janssen Research & Development, Belgium

Introduction

DiffMEM is an open-source package under active development, freely available on Bitbucket [1], for rapid optimization of non-linear mixed effect models using ordinary and partial differential equations (ODE/PDE). During parameter estimation, these ODE/PDE's are frequently integrated numerically as the values of the parameters change from individual to individual and across iterations. This can be computationally expensive and results in long runtimes for model fitting that can become a serious limitation for building, testing and using such models. Through heavy use of parallelism, extensive low-level optimizations and algorithmic innovation [2], DiffMEM is able to rapidly fit such complex models. This allows for shorter duty-cycles while building a model, and enables adaptive/optimal trial design typically requiring a lot of simulation-estimation steps.

Objectives

- Examine the estimation properties of the SAEM routines implemented in DiffMEM with several other methods and implementations based on the methodology from Plan et al. [3]
- Compare the performance between DiffMEM and NONMEM in several real-life use-cases

Methods

Plan et al. [3] considered a sigmoid Emax model (Eq. 1) with 100 individuals and observations at four doses (rich design) and two doses (sparse design) to study nine approaches for ML estimation.

$$E_i = E_{0_i} + \frac{E_{max_i} d^\gamma}{ED_{50_i}^\gamma + d^\gamma} \quad (1)$$

One hundred simulated datasets were generated for each of eight scenarios. Gaussian random components with normal zero-mean distribution were assumed for all individual parameters except γ . The residual error model was assumed to be additive or proportional. We tested the SAEM estimation routine in DiffMEM on the same datasets to compare its results to the other nine approaches.

DiffMEM was used for the parameter estimation on the ODE based PK/PD models from real-life use-cases by Dunne et al. [4] and de Winter et al. [5]. The estimation was performed on the original models without alterations. The former model consists of a two-compartment PK model and a turn-over PD model. Dunne et al. [4] used a sequential estimation method: PK parameters are estimated in a first step and then fixed while fitting PD parameters in the second step. In our comparison we will only focus on this second step. The latter is a population PK model based on a turn-over model for HbA1c consisting of 1347 type-2 diabetic patients. In both settings, we compared the performance of DiffMEM with parallel NONMEM.

DiffMEM Usage

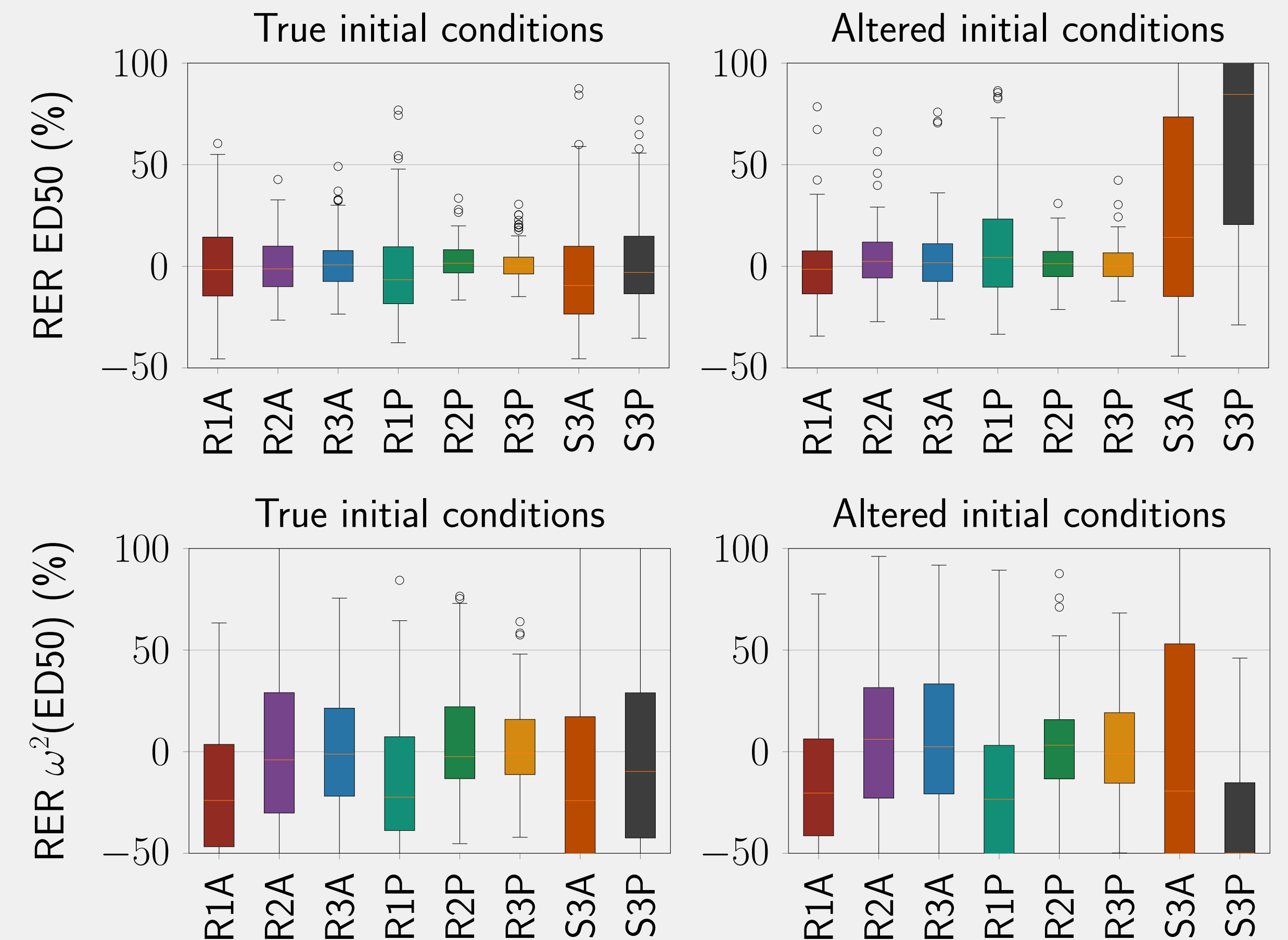
DiffMEM is implemented as a (C++) toolbox consisting of several optimization algorithms and Bayesian samplers, accessible from both R and Python.

- Common models are available for direct use,
- Models can be implemented in a custom high-level language (similar to WinBUGS) or in C++,
- R functions can be used as models directly while retaining the ability to evaluate them in parallel (experimental)

In order to shield users from problems such as software installation, hardware configuration and performance tuning, we have built a **web-service** around DiffMEM: users can submit model specifications and accompanying data to this service and the backend will automatically fit the model given the available hardware.

Results

The SAEM routine in DiffMEM produces consistently comparable results to other SAEM approaches investigated by Plan et al. [3].



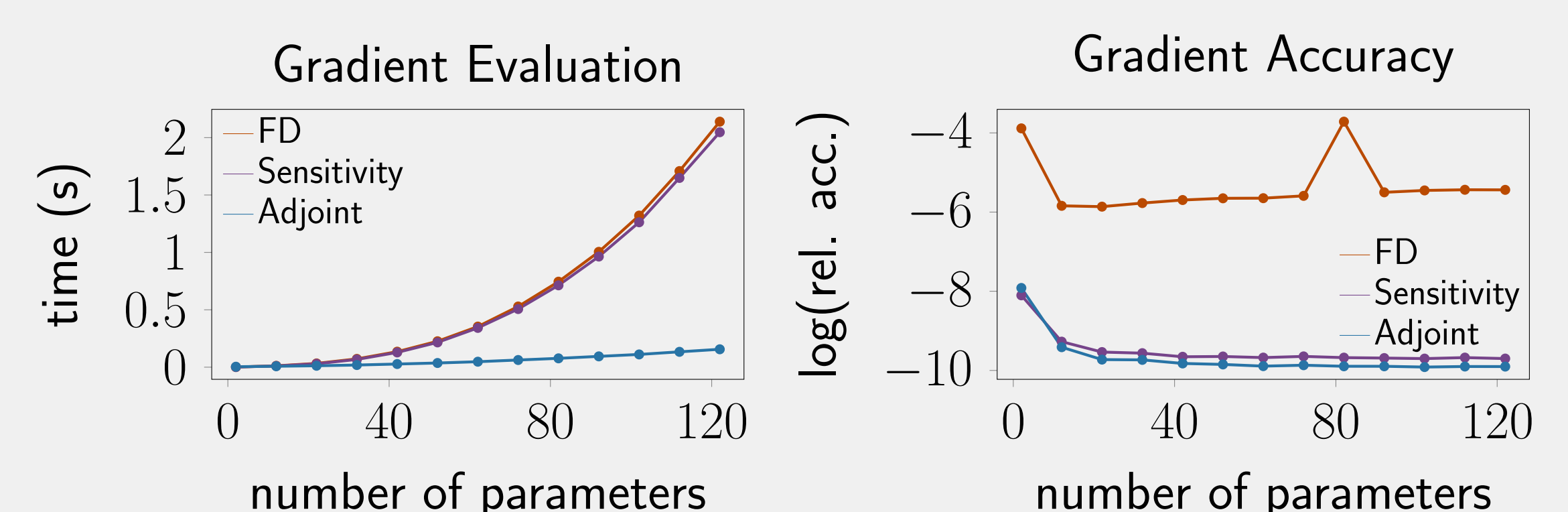
For the use-cases from Dunne et al. [4] and de Winter et al. [5], DiffMEM is able to produce similar results as NONMEM but with much lower runtime: model fitting using parallel NONMEM (24 cores) takes respectively **3 hours** and **8 hours** whereas, using DiffMEM, the runtime is reduced to **10** and **20 minutes**.

Conclusion and Future Work

The study suggests that DiffMEM is a viable open-source alternative for parameter estimation of non-linear mixed effect models while being an order of magnitude faster in delivering results.

Adjoint Method

For a model defined by differential equations, we employ the adjoint state method for efficient computation of the first and the second derivatives [2]. Essentially, the gradient can be computed with a cost (measured in model evaluations) that is independent of the number of ODE model parameters and the Hessian with a linear cost in the number of the parameters instead of the quadratic one.



References

- [1] Tom Haber et al. *DiffMEM*. 2018. url: <https://bitbucket.org/tomhaber/diffmem>.
- [2] Valdemar Melicher, Tom Haber, and Wim Vanroose. "Fast Derivatives of Likelihood Functionals for ODE Based Models Using Adjoint-state Method". In: *Comput. Stat.* 32.4 (Dec. 2017).
- [3] Elodie Plan et al. "Performance Comparison of Various Maximum Likelihood Nonlinear Mixed-Effects Estimation Methods for Dose-Response Models". In: 14 (Apr. 2012).
- [4] Adrian Dunne et al. "The method of averaging applied to pharmacokinetic/pharmacodynamic indirect response models". In: 42 (July 2015).
- [5] Willem de Winter et al. "Dynamic population pharmacokinetic-pharmacodynamic modelling and simulation supports similar efficacy in glycosylated haemoglobin response with once or twice-daily dosing of canagliflozin". In: *British Journal of Clinical Pharmacology* 83 (2017).

Acknowledgments:

Part of the work presented in this paper was funded by Johnson & Johnson.