

## Advancing pharmacometrics with a new open-source framework for pharmacokinetic modelling

Otalvaro, J.D.<sup>1</sup>, Hovd, M.<sup>2,3</sup>, Yamada, W.M.<sup>1</sup>, Kryshchenko, A.<sup>1,4</sup>, Schumitzky, A.<sup>1</sup>, Neely, M.<sup>1,5</sup>

<sup>1</sup> Laboratory of Applied Pharmacokinetics and Bioinformatics, Children's Hospital Los Angeles, CA, USA

<sup>2</sup> Department of Transplantation Medicine, Oslo University Hospital, Oslo, Norway

<sup>3</sup> Department of Pharmacy, University of Oslo, Oslo, Norway

<sup>4</sup> Department of Mathematics, California State University Channel Islands, Camarillo, CA, USA

<sup>5</sup> Keck School of Medicine, University of Southern California, Los Angeles, CA, USA

### Background

Pharmacometrics is a clinically useful tool for describing and predicting drug disposition at the individual and population level. Our laboratory has previously developed Pmetrics, a pharmacometrics package for R implementing our non-parametric adaptive grid (NPAG) algorithm for parameter estimation. The NPAG algorithm was initially written in the Fortran language (legacy), which is computationally efficient, but lacks the features of modern programming languages. As such, we made significant effort to rewrite NPAG and embed it in a new framework in Rust, a memory-safe, fast, and modern language. We here present that new framework, "PMcore", and compare it to legacy NPAG.

### Methods

We used three previously published population pharmacokinetic models. Model A was a simulated one-compartment model with IV infusion and bimodal elimination. Model B was developed on real data, consisted of two compartments, and with IV bolus dosing. Model C was also developed on real data, consisted of 3 compartments, and with age-dependent bioavailability, and Michaelis-Menten elimination after oral doses and IV infusions. We fitted the same datasets and model priors with both PMcore and legacy.

### Results

For the simplest Model A, PMcore was slower than legacy by one second (9s versus 10s). However, for the more complex Models B and C with larger datasets, PMcore was 73% (1136s versus 4137s) and 33% (2497s versus 3711s) faster than legacy. Additionally, PMcore achieved lower objective functions ( $-2 \times \log$ -likelihood) than legacy, i.e. more likely solutions, for Model A (-646 vs -397), Model B (9275 vs 9911), and Model C (507 vs 602).

### Conclusions

We have successfully developed and validated PMcore, a new framework for pharmacokinetic modelling. NPAG in PMcore outperformed legacy Fortran with more likely solutions found as or more quickly. PMcore is integrated within the Pmetrics package for R and requires no additional user training. The architecture of PMcore permits unparalleled easy swapping and performance comparisons between model fitting algorithms, both parametric and nonparametric, and even hybrid population and machine learning algorithms, as they are added to the package.