

Chi: A Python package for treatment response modelling

David Augustin ¹✉

¹ Department of Computer Science, University of Oxford, Oxford, UK ✉ Corresponding author

DOI: [10.21105/joss.05925](https://doi.org/10.21105/joss.05925)

Software

- [Review](#) 
- [Repository](#) 
- [Archive](#) 

Editor: [Antonia Mey](#)  

Reviewers:

- [@shahmoradi](#)
- [@ns-rse](#)

Submitted: 06 August 2023

Published: 31 January 2024

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).

Summary

Chi is an open source Python package for treatment response modelling with support for implementation, simulation and parameter estimation. Supported treatment response models include pharmacokinetic & pharmacodynamic (PKPD) models, physiology-based pharmacokinetic (PBPK) models, quantitative systems pharmacology (QSP) models, and nonlinear mixed effects (NLME) models. The package provides two interfaces to implement single-individual treatment response models: 1. an SBML interface, which implements models based on SBML file specifications ([Hucka et al., 2003](#)); and 2. a general purpose interface that allows users to implement their own, custom models using Python code. Models implemented using SBML files automatically provide routines to administer dosing regimens and to evaluate parameter sensitivities using the simulation engine **Myokit** ([Clerx et al., 2016](#)). These single-individual treatment response models can be extended to NLME models, making the simulation of inter-individual variability of treatment responses possible.

In **Chi**, model parameters can be estimated from data using Bayesian inference. We provide a simple interface to infer posterior distributions of model parameters from single-patient data or from population data. For the extreme case where the available population data only contains a single measurement for each individual, the package also implements filter inference, enabling the inference of NLME model parameters from such snapshot time series data ([Augustin, Lambert, Wang, et al., 2023](#)). For the purpose of model-informed precision dosing (MIPD), **Chi** can be used to find individual-specific dosing regimens that optimise treatment responses with respect to a target treatment outcome.

To sample from posterior distributions, **Chi** uses Markov chain Monte Carlo (MCMC) algorithms implemented in the Python package **PINTS** ([Clerx et al., 2019](#)). To optimise dosing regimens, different optimisation algorithms can be used, including optimisers implemented in **SciPy** ([Virtanen et al., 2020](#)) or in **PINTS** ([Clerx et al., 2019](#)).

Documentation, tutorials and install instructions are available at <https://chi.readthedocs.io>.

Statement of need

Treatment response modelling has become an integral part of pharmaceutical research ([Morgan et al., 2018](#); [Schuck et al., 2015](#)). In the early phase of drug development, treatment response models help with target and lead identification, and contribute to a mechanistic understanding of the relevant pharmacological processes. In the transition to the clinical development phase, these models provide guidance and help to identify safe and efficacious dosing regimens ([Lavé et al., 2016](#)). During clinical trials, treatment response models further facilitate the assessment of safety, efficacy and treatment response variability. More recently, treatment response models are also being used in the context of MIPD, where models help to predict individualised dosing regimens for otherwise difficult-to-administer drugs ([Augustin, Lambert, Robinson, et al.,](#)

2023).

The most widely used software packages and computer programs for treatment response modelling include NONMEM (Keizer et al., 2013), Monolix, and Matlab Simbiology (Hosseini et al., 2018). Other software packages include Scipion PKPD (Sorzano et al., 2021), PoPy, Pumas (Rackauckas et al., 2020), and a number of R libraries. These packages provide an extensive toolkit for PKPD modelling. However, most of these solutions are difficult to use for research as their source code is not publicly distributed or not subject to open-source licenses, concealing the algorithmic details, limiting the transparency of the modelling results, and hindering the methodological development. Notable exceptions are Scipion PKPD and the R libraries, which make their source code publicly available on GitHub.

Chi is an easy-to-use, open-source Python package for treatment response modelling. It is targeted at modellers on all levels of programming expertise. Modellers with a primary focus on the pharmacology can use Chi to quickly implement models and estimate their model parameters from data. Modellers with an interest in methodological research can use Chi's modular, open source framework to study the advantages and limitations of different modelling choices, and research new approaches for treatment response modelling. We hope that the open-source nature of this package will increase the transparency of treatment response models and facilitate a community effort to further develop their methodology.

References

- Augustin, D., Lambert, B., Robinson, M., Wang, K., & Gavaghan, D. (2023). Simulating clinical trials for model-informed precision dosing: Using warfarin treatment as a use case. *Frontiers in Pharmacology*, 14. <https://doi.org/10.3389/fphar.2023.1270443>
- Augustin, D., Lambert, B., Wang, K., Walz, A.-C., Robinson, M., & Gavaghan, D. (2023). Filter inference: A scalable nonlinear mixed effects inference approach for snapshot time series data. *PLOS Computational Biology*, 19(5), 1–29. <https://doi.org/10.1371/journal.pcbi.1011135>
- Clerx, M., Collins, P., De Lange, E., & Volders, P. G. (2016). Myokit: A simple interface to cardiac cellular electrophysiology. *Progress in Biophysics and Molecular Biology*, 120(1-3), 100–114. <https://doi.org/10.1016/j.pbiomolbio.2015.12.008>
- Clerx, M., Robinson, M., Lambert, B., Lei, C. L., Ghosh, S., Mirams, G. R., & Gavaghan, D. J. (2019). Probabilistic inference on noisy time series (PINTS). *Journal of Open Research Software*. <https://doi.org/10.5334/jors.252>
- Hosseini, I., Gajjala, A., Bumbaca Yadav, D., Sukumaran, S., Ramanujan, S., Paxson, R., & Gadkar, K. (2018). gPKPDSim: A SimBiology®-based GUI application for PKPD modeling in drug development. *Journal of Pharmacokinetics and Pharmacodynamics*, 45, 259–275. <https://doi.org/10.1007/s10928-017-9562-9>
- Hucka, M., Finney, A., Sauro, H. M., Bolouri, H., Doyle, J. C., Kitano, H., Arkin, A. P., Bornstein, B. J., Bray, D., Cornish-Bowden, A., & others. (2003). The systems biology markup language (SBML): A medium for representation and exchange of biochemical network models. *Bioinformatics*, 19(4), 524–531. <https://doi.org/10.1093/bioinformatics/btg015>
- Keizer, R. J., Karlsson, M., & Hooker, A. (2013). Modeling and simulation workbench for NONMEM: Tutorial on pirana, PsN, and xpose. *CPT: Pharmacometrics & Systems Pharmacology*, 2(6), 1–9. <https://doi.org/10.1038/psp.2013.24>
- Lavé, T., Caruso, A., Parrott, N., & Walz, A. (2016). Translational PK/PD modeling to increase probability of success in drug discovery and early development. *Drug Discovery Today: Technologies*, 21-22, 27–34. <https://doi.org/10.1016/j.ddtec.2016.11.005>

- Morgan, P., Brown, D. G., Lennard, S., Anderton, M. J., Barrett, J. C., Eriksson, U., Fidock, M., Hamrén, B., Johnson, A., March, R. E., Matcham, J., Mettetal, J., Nicholls, D. J., Platz, S., Rees, S., Snowden, M. A., & Pangalos, M. N. (2018). Impact of a five-dimensional framework on r&d productivity at AstraZeneca. *Nature Reviews Drug Discovery*, 17(3), 167–181. <https://doi.org/10.1038/nrd.2017.244>
- Rackauckas, C., Ma, Y., Noack, A., Dixit, V., Mogensen, P. K., Byrne, S., Maddhashiya, S., Santiago Calderón, J. B., Nyberg, J., Gobburu, J. V., & others. (2020). Accelerated predictive healthcare analytics with pumas, a high performance pharmaceutical modeling and simulation platform. *BioRxiv*, 2020–2021. <https://doi.org/10.1101/2020.11.28.402297>
- Schuck, E., Bohnert, T., Chakravarty, A., Damian-Iordache, V., Gibson, C., Hsu, C.-P., Heimbach, T., Krishnatry, A. S., Liederer, B. M., Lin, J., Maurer, T., Mettetal, J. T., Mudra, D. R., Nijssen, M. J., Raybon, J., Schroeder, P., Schuck, V., Suryawanshi, S., Su, Y., ... Wong, H. (2015). Preclinical pharmacokinetic / pharmacodynamic modeling and simulation in the pharmaceutical industry: An IQ consortium survey examining the current landscape. *The AAPS Journal*, 17(2), 462–473. <https://doi.org/10.1208/s12248-014-9716-2>
- Sorzano, C., Fonseca-Reyna, Y., & Cruz-Moreno, M. P. de la. (2021). Scipion PKPD: An open-source platform for biopharmaceutics, pharmacokinetics and pharmacodynamics data analysis. *Pharmaceutical Research*, 38(7), 1169–1178. <https://doi.org/10.1007/s11095-021-03065-1>
- Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., ... SciPy 1.0 Contributors. (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17, 261–272. <https://doi.org/10.1038/s41592-019-0686-2>