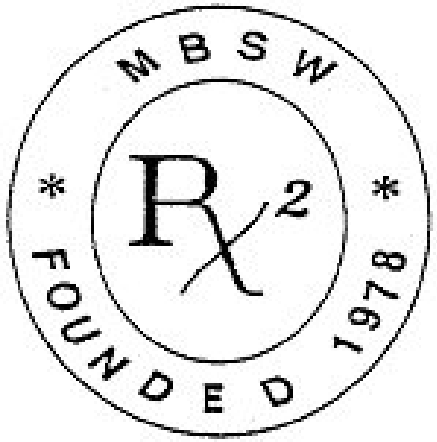
METRUM
RESEARCH GROUP

Bayesian Data Analysis Using Stan/Torsten for Pharmacometric Applications

Bill Gillespie

Metrum Research Group

15 May 2018



Bayesian data analysis using Stan/Torsten for pharmacometric applications

- Why Bayesian analysis for pharmacometrics applications?
- Why Stan?
- Torsten: Adapting Stan for typical pharmacometric modeling tasks
  - Key components
  - Examples
  - Pros & cons relative to other Bayesian PMX options
- Things to come

Why Bayesian analysis for pharmacometrics applications?

- Decision-making supported by quantitative synthesis of prior knowledge and heterogenous data.
- Calibration (and recalibration) of complex QSP models as new data accumulates.
- Bayesian framework more easily accommodates complexity in the stochastic structure of a model.

Why Bayesian?

- Pharmacometricians are often called on to leverage prior knowledge in order to interpret new data and facilitate decision-making in drug development.

Why Bayesian?

- Pharmacometricians are often called on to leverage prior knowledge in order to interpret new data and facilitate decision-making in drug development.
  - Qualitative prior knowledge is captured in the mathematical form of a model, i.e., the **likelihood function**.

Why Bayesian?

- Pharmacometricians are often called on to leverage prior knowledge in order to interpret new data and facilitate decision-making in drug development.
  - Qualitative prior knowledge is captured in the mathematical form of a model, i.e., the **likelihood function**.
  - Quantitative prior knowledge may be captured in the form of probability distributions of model parameter values, i.e., **prior distributions**.

Why Bayesian?

- Pharmacometricians are often called on to leverage prior knowledge in order to interpret new data and facilitate decision-making in drug development.
  - Qualitative prior knowledge is captured in the mathematical form of a model, i.e., the **likelihood function**.
  - Quantitative prior knowledge may be captured in the form of probability distributions of model parameter values, i.e., **prior distributions**.
- Add **data** and you have all the ingredients of Bayesian data analysis.

Why Bayesian?

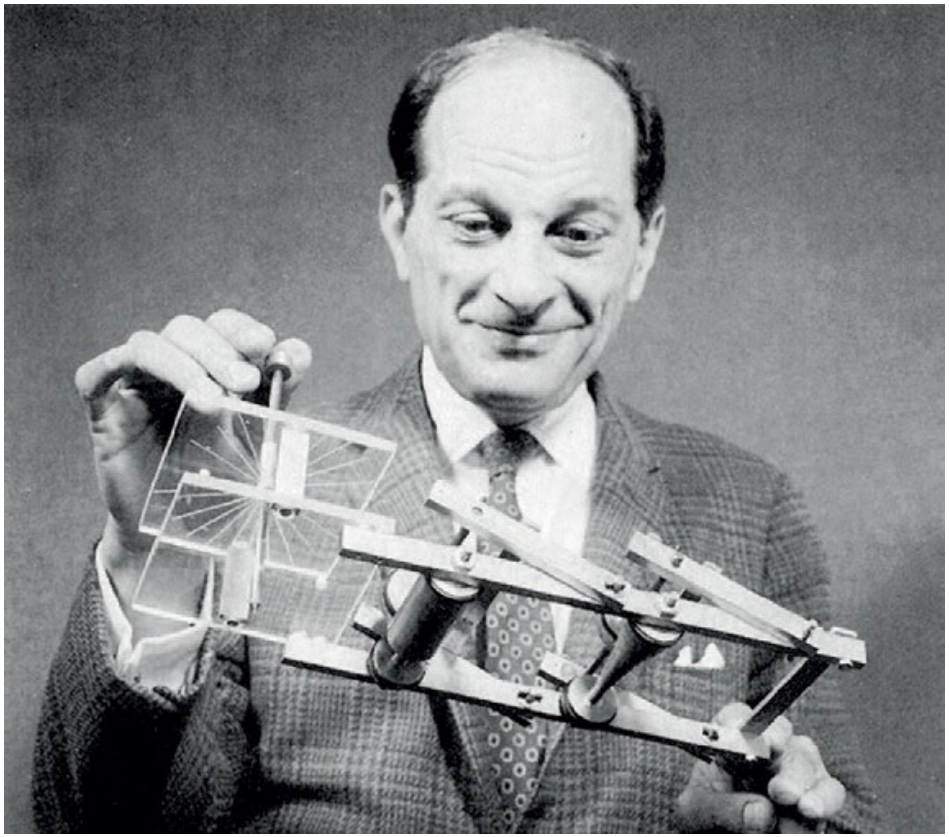
- Pharmacometricians are often called on to leverage prior knowledge in order to interpret new data and facilitate decision-making in drug development.
  - Qualitative prior knowledge is captured in the mathematical form of a model, i.e., the **likelihood function**.
  - Quantitative prior knowledge may be captured in the form of probability distributions of model parameter values, i.e., **prior distributions**.
- Add **data** and you have all the ingredients of Bayesian data analysis.
- With Bayes Rule and suitable computation tools those components are combined to yield **posterior distributions** of model parameters and predictions.
- Those distributions permit probabilistic inferences directly relevant to decision-making.

Why Stan?

Stan (`http://mc-stan.org/`) is a general purpose Bayesian modeling package [1]

- General model specification language
- Primarily uses a Hamiltonian Monte Carlo (HMC) sampler (standard HMC or NUTS (no U-turn sampler)). Other methods include:
  - Optimization for estimation of posterior modes.
  - Variational inference for approximate Bayesian inference.
- Developed by a team headed by Andrew Gelman of Columbia University
- C++ program available with several interfaces: rstan, PyStan, CmdStan, MatlabStan, Stan.jl, StataStan, ShinyStan

Stan: Why is it called that?



Stanislaw Ulam, co-inventor of Monte Carlo methods, holding an analog computer known as the FERMIAC that performed a mechanical simulation of random diffusion of neutrons (`http://fas.org/sgp/othergov/doe/lanl/pubs/00326866.pdf`).

Stan: How do I get it?

- Most Stan interfaces may be downloaded from the Stan website (`http://mc-stan.org/`).
- rstan [2] is available on CRAN (`https://cran.r-project.org/`)
- Documentation
  - Stan: `https://github.com/stan-dev/stan/releases/download/v2.17.0/stan-reference-2.17.0.pdf`
  - rstan: `https://cran.r-project.org/web/packages/rstan/vignettes/rstan.html`

Stan model specification language

Very flexible model specification language

- Imperative language: statements executed in the order in which they are written.
- Computational control structures, e.g., if-then-else, for and while loops
- Large collection of:
  - Operators
  - Built-in functions
  - Probability distributions
- User-defined functions and distributions

Stan features particularly relevant to pharmacometrics

- Functions for numerical solution of ODEs:
  - integrate_ode_rk45
    - Runge Kutta Dopri 4th/5th order algorithm with the implementation from Boost
    - Suitable for non-stiff ODEs
  - integrate_ode_bdf
    - Backward differentiation formula (BDF) method with the implementation from SUNDIALS (CVODES)
    - Designed for stiff ODEs

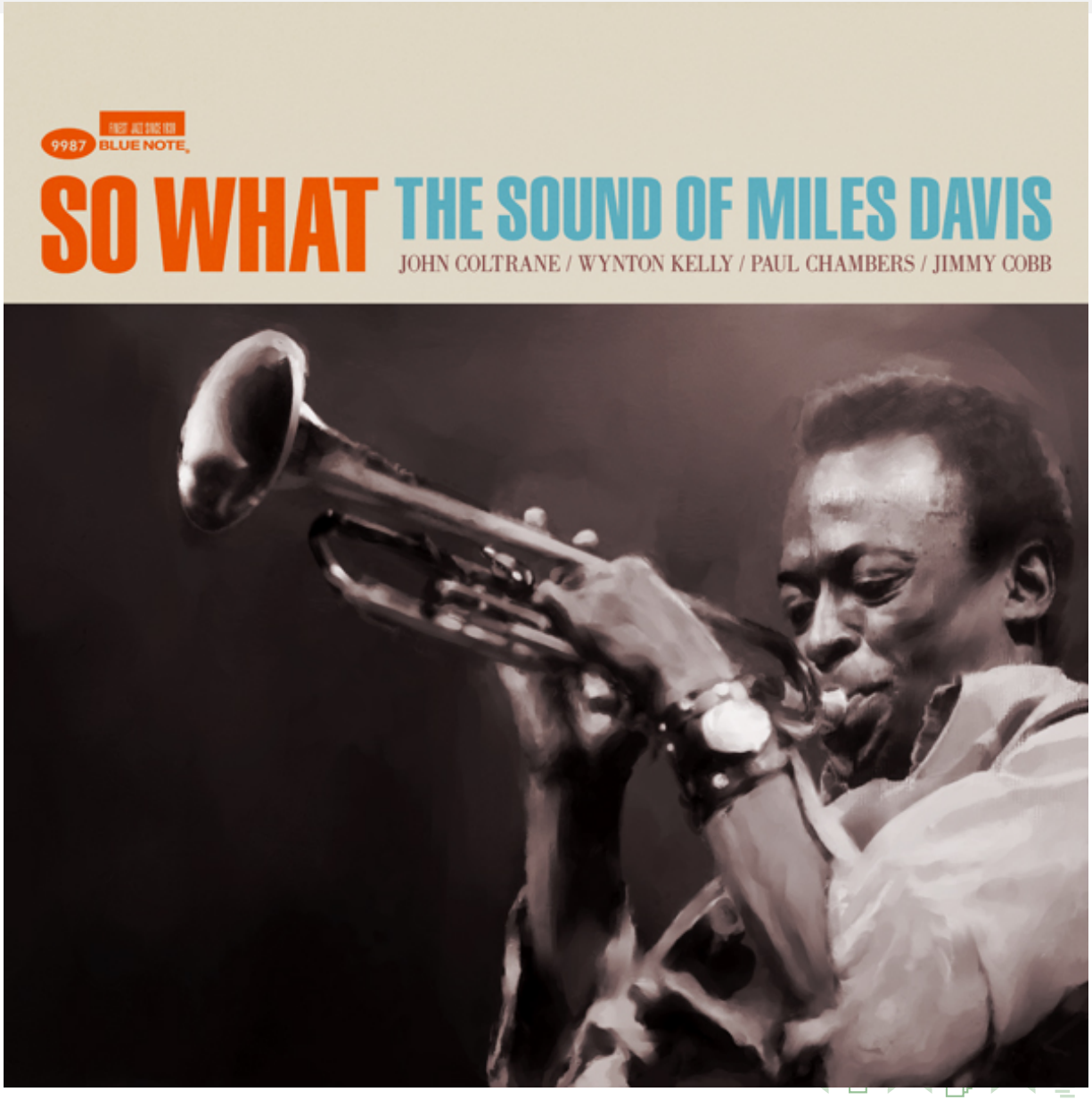
Stan features particularly relevant to pharmacometrics

- Functions for numerical solution of ODEs:
  - integrate_ode_rk45
    - Runge Kutta Dopri 4th/5th order algorithm with the implementation from Boost
    - Suitable for non-stiff ODEs
  - integrate_ode_bdf
    - Backward differentiation formula (BDF) method with the implementation from SUNDIALS (CVODES)
    - Designed for stiff ODEs
- There are no built-in handlers for PKPD event schedules—requires user programming.

Stan features particularly relevant to pharmacometrics

- Functions for numerical solution of ODEs:
  - integrate_ode_rk45
    - Runge Kutta Dopri 4th/5th order algorithm with the implementation from Boost
    - Suitable for non-stiff ODEs
  - integrate_ode_bdf
    - Backward differentiation formula (BDF) method with the implementation from SUNDIALS (CVODES)
    - Designed for stiff ODEs
- There are no built-in handlers for PKPD event schedules—requires user programming.
- HMC/NUTS more efficiently samples the complex, high-dimensional joint posterior distributions resulting from nonlinear PMX models.

OK I get it. Stan is flexible. So what?



Let's take a look at an example that exploits that flexibility.

Joint modeling of individual patient data (IPD) and aggregate data (AD)

BAYESIAN AGGREGATION OF AVERAGE DATA: AN APPLICATION IN DRUG DEVELOPMENT

By Sebastian Weber[‡] Andrew Gelman[§,*] Daniel Lee[¶]
Michael Betancourt[§,*]
Aki Vehtari[‖,†] and Amy Racine-Poon[‡]

Novartis Pharma AG[‡], *Columbia University*[§], *Stan Group*[¶], *Aalto University*[‖]

`http://www.stat.columbia.edu/~gelman/research/published/extrap_paper_aoas.pdf` (accepted by Annals of Applied Statistics) [5]

Why combine individual and aggregate data?

Addition of AD to enhance/extend inferences from IPD analysis

- Good reasons
  - Indirect comparisons of treatment effects
    - Particularly when comparators are only available in AD
  - Quantifying effects of other group-level covariates (when AD is available for the relevant groups)
  - Quantifying inter-trial variability
  - Improving precision of some model parameter estimates
- Not-so-good reasons
  - Quantifying effects of patient-level covariates

Why combine individual and aggregate data?

Addition of IPD to enhance/extend inferences from AD analysis

- IPD required to inform correlations among individual-level outcomes and covariates
- IPD required to quantify effects of patient-level covariates

Problem faced by Weber et al

- Limited clinical data (IPD) for a new anti-VEGF agent for treating wet AMD.

Problem faced by Weber et al

- Limited clinical data (IPD) for a new anti-VEGF agent for treating wet AMD.
- Want a model suitable for simulating longitudinal data for individual patients and making inferences about:
  - Treatment regimens.
  - Clinical trial designs.

Problem faced by Weber et al

- Limited clinical data (IPD) for a new anti-VEGF agent for treating wet AMD.
- Want a model suitable for simulating longitudinal data for individual patients and making inferences about:
  - Treatment regimens.
  - Clinical trial designs.
- Want to enhance the precision and generalizability of the model by leveraging past data from trials of other anti-VEGF agents.
  - Ranibizumab: IPD from 3 trials
  - Aflibercept: published AD from 2 trials

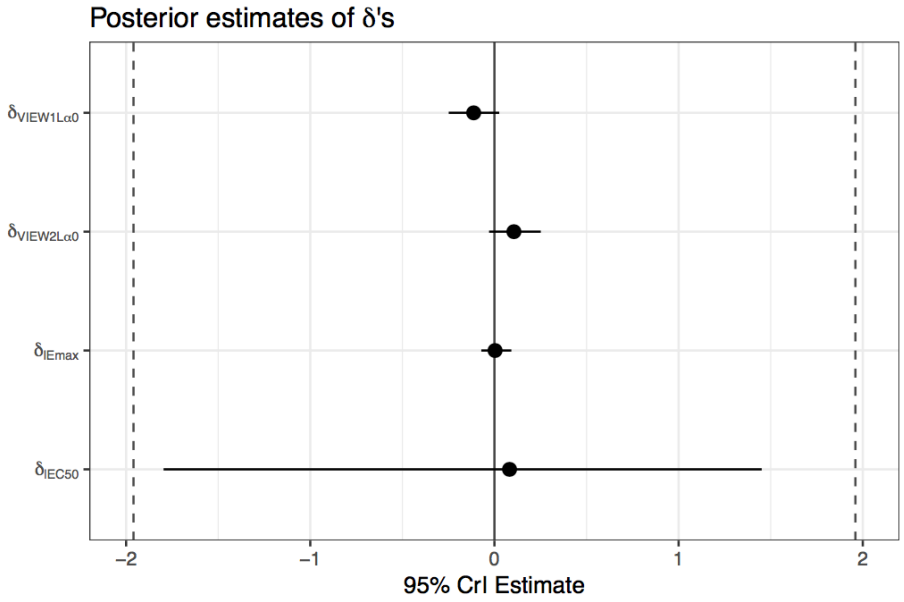
Problem faced by Weber et al

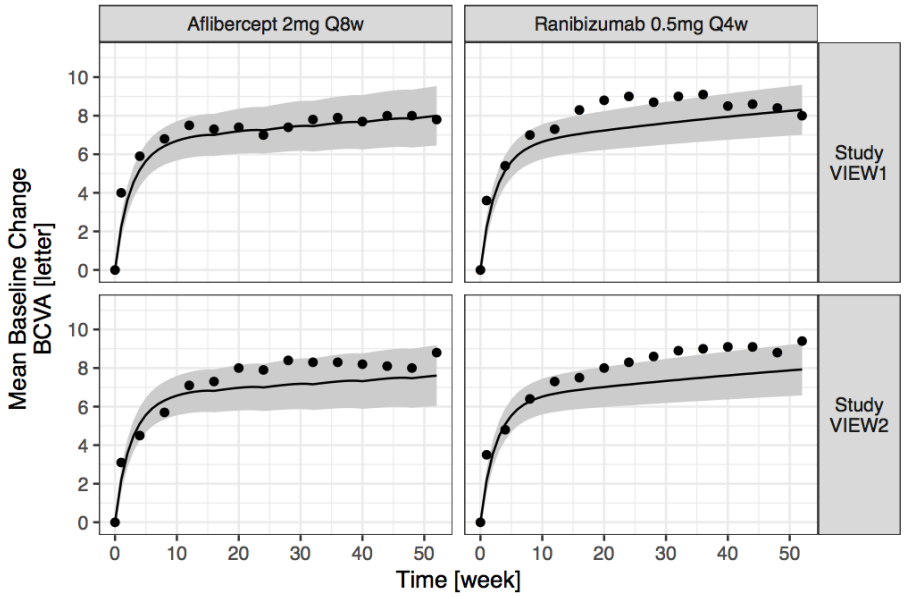
- Limited clinical data (IPD) for a new anti-VEGF agent for treating wet AMD.
- Want a model suitable for simulating longitudinal data for individual patients and making inferences about:
  - Treatment regimens.
  - Clinical trial designs.
- Want to enhance the precision and generalizability of the model by leveraging past data from trials of other anti-VEGF agents.
  - Ranibizumab: IPD from 3 trials
  - Aflibercept: published AD from 2 trials
- Nonlinear longitudinal model for IPD
  - Indirect action KPD model for BCVA (best corrected visual acuity) requiring numerical solution of ODEs
  - Hierarchical (inter-trial, inter-individual and residual variability)

Tackling the problem with Stan

- The IPD likelihood is readily implemented in Stan.
- An approximate AD likelihood is imputed via simulations of individual data [6, 7, 8, 9].
- For each treatment arm suppose you have a set of means $\overline{y}_i, i = 1, 2, \ldots, n_T$ of longitudinal data for N individuals.
- Impute the joint likelihood of the $\overline{y}_i$'s by:
  - Simulating individual data for a large number of individuals,
  - Calculating the mean vector M_s and covariance matrix Σ_s of the simulated values,
  - Approximating the joint likelihood of $\overline{y}_i, i = 1, 2, \ldots, n_T$ as multivariate normal: $N\left(M_s, \frac{\Sigma_s}{N}\right)$

Stan is flexible enough to support this approach





The model specification language permitted implementation of all aspects of the model:

- 3 level stochastic hierarchy,
- Numerical solution of ODEs,
- Simulation to impute AD likelihood,
- Fully Bayesian data analysis.

Torsten: Library of PKPD functions for Stan

A set of Stan functions that provides functionality similar to NONMEM's PREDPP library

Core functions in the current version:

- One & two compartment PK models with 1st order absorption
  - Analytical solutions

Torsten: Library of PKPD functions for Stan

A set of Stan functions that provides functionality similar to NONMEM's PREDPP library

Core functions in the current version:

- One & two compartment PK models with 1st order absorption
  - Analytical solutions
- Linear compartment model specified as a rate constant matrix
  - Semi-analytical solution based on matrix exponential

Torsten: Library of PKPD functions for Stan

A set of Stan functions that provides functionality similar to NONMEM's PREDPP library

Core functions in the current version:

- One & two compartment PK models with 1st order absorption
  - Analytical solutions
- Linear compartment model specified as a rate constant matrix
  - Semi-analytical solution based on matrix exponential
- General compartmental model specified as a system of 1st order ODEs
  - Numerical solutions
  - Non-stiff solver: Runge Kutta Dopri 4th/5th order algorithm with the implementation from Boost
  - Stiff solver: Backward differentiation formula (BDF) method with the implementation from SUNDIALS (CVODES)

Torsten Teorell

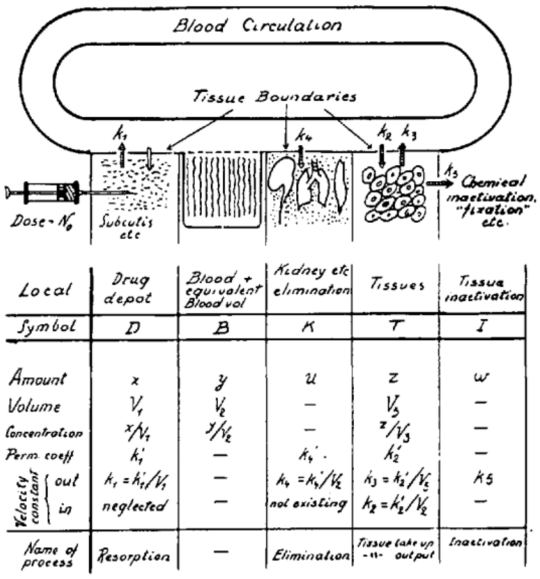


FIG. 1

Scheme of the Concept of Drug Distribution used in this paper. Instead the *injection* pictured in the figure, the administration of the drug depot can be made *per os*, *per rectum*, *by inhalation*, etc.

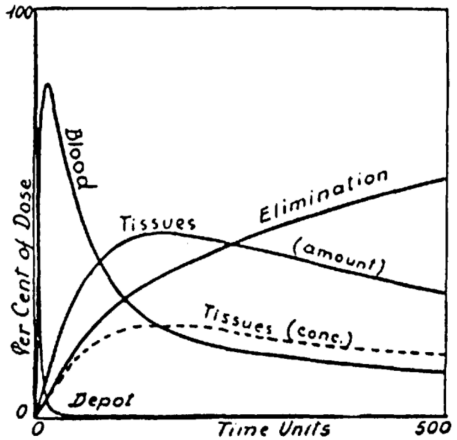


FIG. 3

Typical Case of Extravascular Administration in the absence of tissue inactivation.

($k_1 = 0.2$; $k_2 = 0.01$; $k_3 = 0.005$; i.e. "blood" volume/"tissue" volume is 1 : 2; $k_4 = 0.005$; $k_5 = 0$).

T. Teorell. Kinetics of distribution of substances administered to the body. I. The extravascular modes of administration. Arch Int Pharmacodyn et Ther 57: 205-225, 1937.

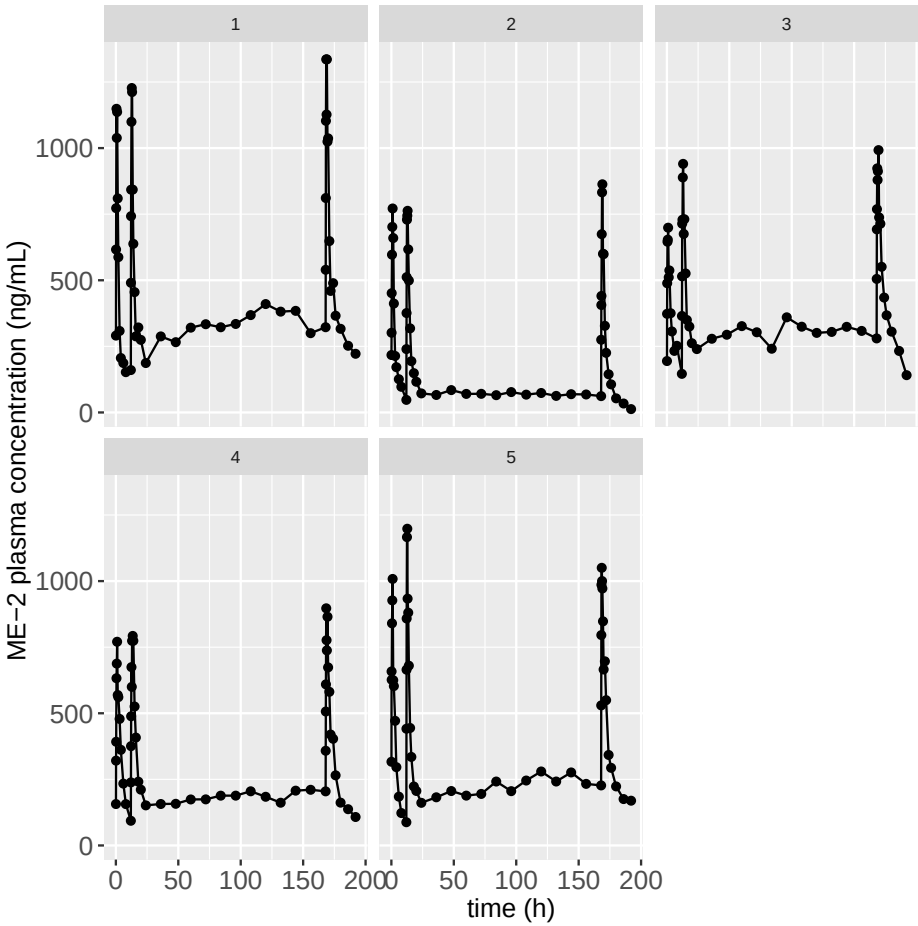
Torsten PMX functions

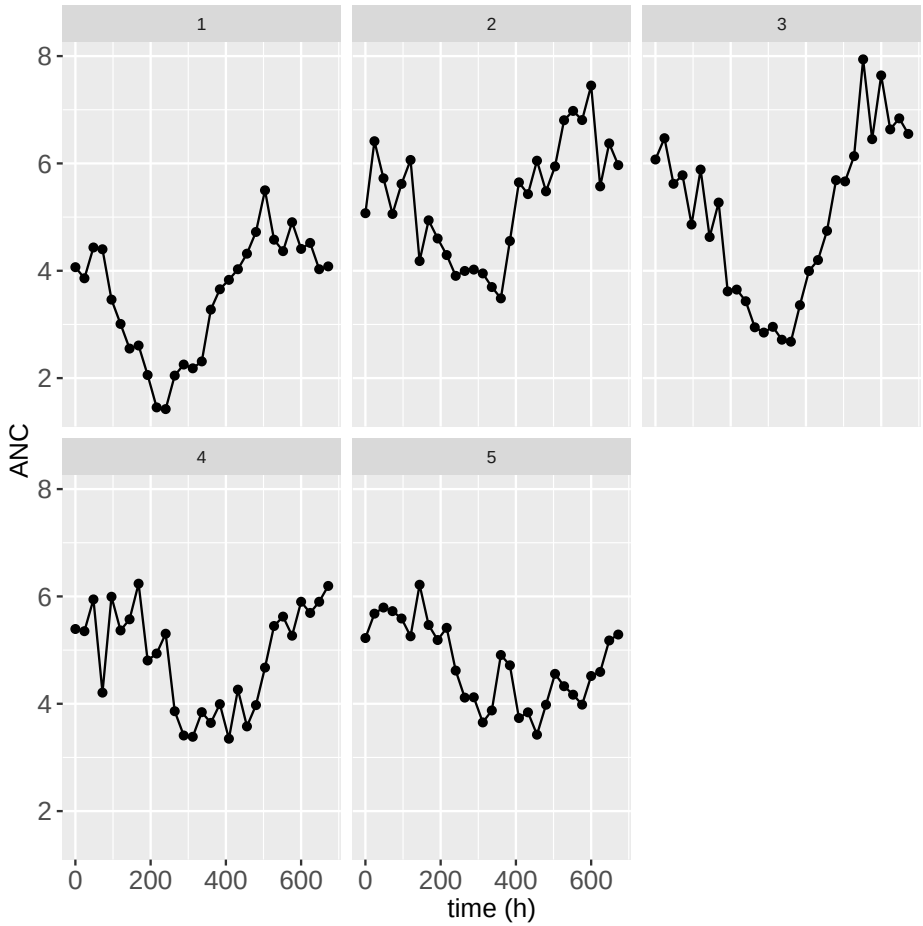
- Uses NONMEM/PREDPP conventions for data specification and event handling
- Data format: Time-ordered event records for each individual à la NONMEM
- Implemented NONMEM data types: TIME, CMT, AMT, RATE, EVID, II, ADDL, SS
- Recursive calculation: For each event time calculate the amount in each compartment given the compartment amounts plus doses at the previous event time.
- Allows for time-varying (piece-wise constant) parameter values.

Torsten: Prototype library of PKPD functions for Stan

- Current version of Torsten is available at: `https://github.com/metrumresearchgroup/Torsten`
- Includes installation instructions for use with CmdStan and RStan.
- Documentation: `https://github.com/metrumresearchgroup/Torsten/blob/master/docs/torsten_manual.pdf`

Torsten example: PKPD model of drug-induced neutropenia



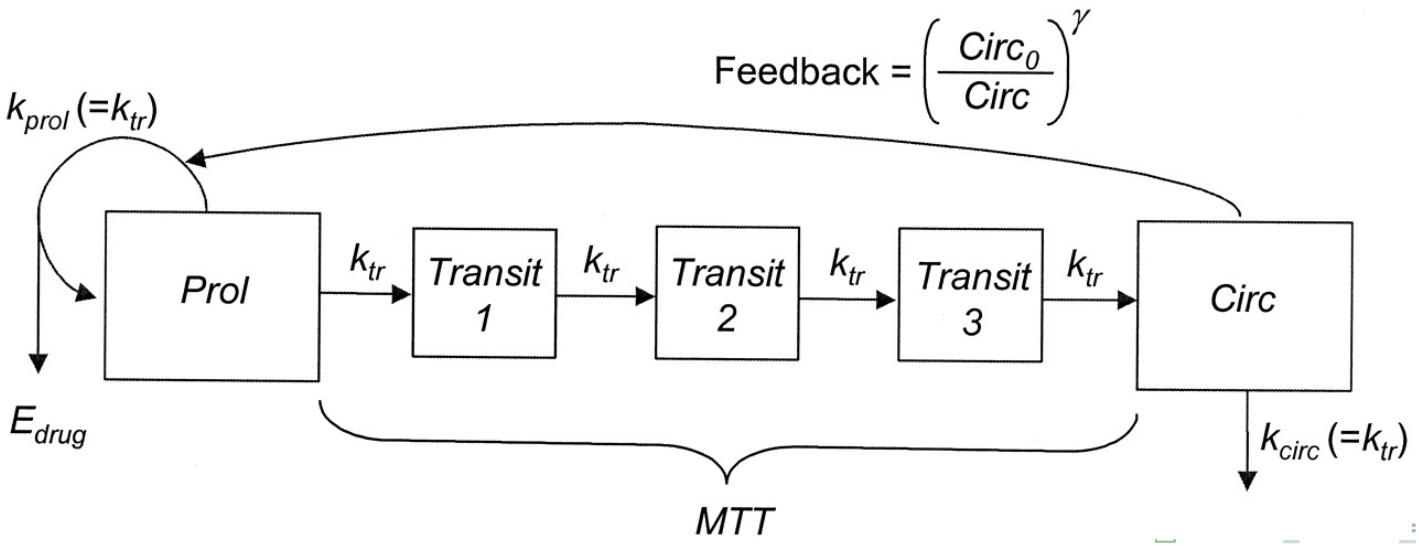


Friberg-Karlsson semi-mechanistic model for drug-induced myelosuppression

- PK model: Two compartment model with first order absorption describing plasma drug concentration on the i^{th} occasion in the j^{th} subject as a function of time, dose and body weight:

$$\begin{aligned} \log\left(c_{ij}\right) &\sim N\left(\log\left(\hat{c}_{ij}\right), \sigma\right) \\ \hat{c}_{ij} &= f_{2cpt}\left(t_{ij}, D_j, \tau_j, CL_j, Q_j, V_{1j}, V_{2j}, k_{aj}\right) \end{aligned}$$

- Friberg-Karlsson semi-mechanistic model for drug-induced myelosuppression [10, 11, 12, 13, 14, 15]



Friberg-Karlsson semi-mechanistic model for drug-induced myelosuppression

$$\frac{dProl}{dt} = k_{prol} Prol\,(1 - E_{drug}) \left(\frac{Circ_0}{Circ}\right)^{\gamma} - k_{tr} Prol$$

$$\frac{dTransit1}{dt} = k_{tr} Prol - k_{tr} Transit1$$

$$\frac{dTransit2}{dt} = k_{tr} Transit1 - k_{tr} Transit2$$

$$\frac{dTransit3}{dt} = k_{tr} Transit2 - k_{tr} Transit3$$

$$\frac{dCirc}{dt} = k_{tr} Transit3 - k_{circ} Circ$$

$$\widehat{c} \equiv \text{plasma drug concentration}$$

$$Circ \equiv \text{absolute neutrophil count (ANC)}$$

Parameters in red are *system* parameters, i.e., drug-independent.

$$E_{drug} = \alpha \widehat{c}$$

$$k_{prol} = k_{circ} = k_{tr}$$

$$MTT = \frac{n+1}{k_{tr}}$$

IIV and prior distributions

- Inter-individual variation

$$\log\left(CL_j, Q_j, V_{1j}, V_{2j}, k_{aj}, MTT_j, Circ_{0j}, \alpha_j\right) \sim N\left(\log\left(\widehat{CL}\left(\frac{bw_j}{70}\right)^{0.75}, \widehat{Q}\left(\frac{bw_j}{70}\right)^{0.75}, \widehat{V}_1\left(\frac{bw_j}{70}\right), \widehat{V}_2\left(\frac{bw_j}{70}\right), \widehat{k}_a, \widehat{MTT}, \widehat{Circ_0}, \widehat{\alpha}\right), \Omega\right)$$

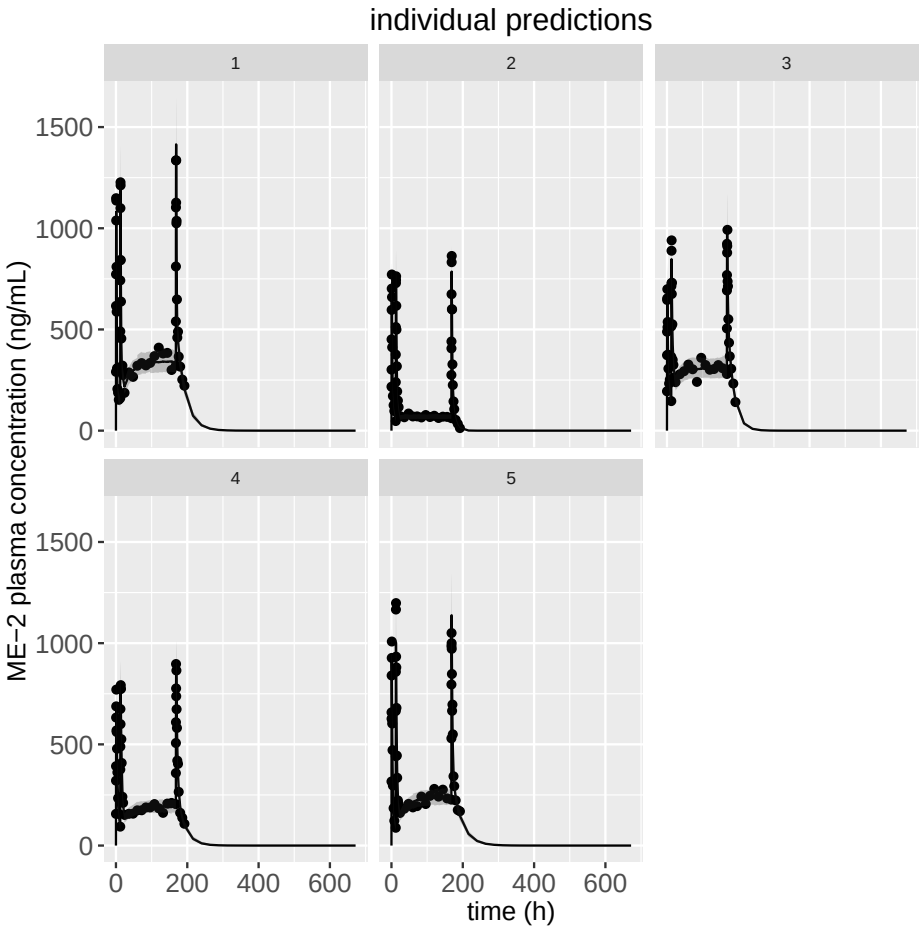
- Prior distributions: moderately informative for PK, strongly informative for system parameters, weakly informative for drug effect

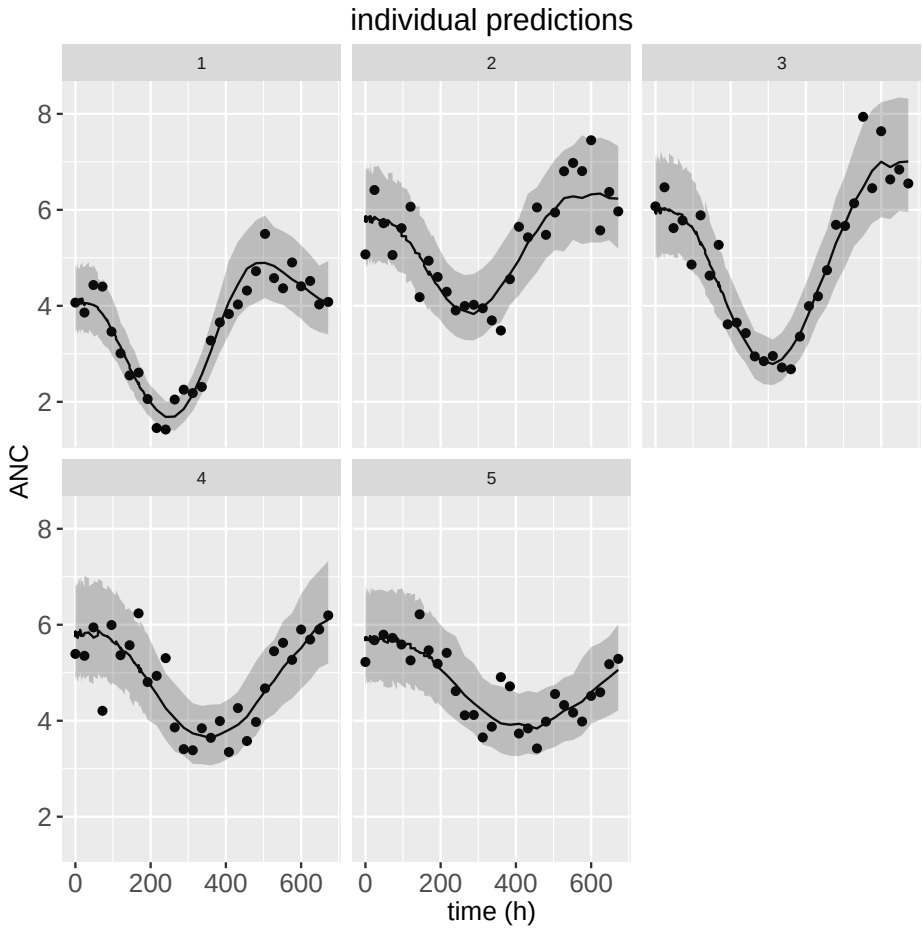
$$\begin{aligned}
\widehat{CL} &\sim \log N(\log(10), 0.5) \quad \widehat{Q} \sim \log N(\log(15), 0.5) \quad \widehat{V}_1 \sim \log N(\log(35), 0.5) \\
\widehat{V}_2 &\sim \log N(\log(105), 0.5) \quad \widehat{k}_a \sim \log N(\log(2), 0.5) \\
\widehat{MTT} &\sim \log N(\log(125), 0.2) \quad \widehat{Circ_0} \sim \log N(\log(5), 0.2) \quad \gamma \sim \log N(\log(0.17), 0.2) \\
\widehat{\alpha} &\sim \log N\left(\log(3 \times 10^{-4}), 1\right) \quad \sigma \sim \text{half-Cauchy}(0, 1) \\
\Omega &= \text{diag}(\omega)\, P\, \text{diag}(\omega) \\
\omega_i &\sim \text{half-Cauchy}(0, 1), i \in \{1, 2, \ldots, 8\} \quad P \sim \text{LKJCorr}(1)
\end{aligned}$$

Good convergence and mixing with only 4 chains of 100 warmup and 100 post-warmup samples/chain

parameter	mean	sd	95% CI	n_eff	Rhat
CLHat	$1.30e+01$	$2.51e+00$	$(8.48e+00, 1.82e+01)$	400	0.996
QHat	$1.76e+01$	$4.92e+00$	$(9.75e+00, 2.88e+01)$	400	0.997
V1Hat	$4.51e+01$	$9.29e+00$	$(2.90e+01, 6.52e+01)$	400	1.000
V2Hat	$1.06e+02$	$1.61e+01$	$(7.81e+01, 1.38e+02)$	400	0.995
kaHat	$2.30e+00$	$4.75e-01$	$(1.48e+00, 3.37e+00)$	324	1.004
sigma	$9.73e-02$	$4.62e-03$	$(8.90e-02, 1.06e-01)$	349	1.003
alphaHat	$3.06e-04$	$2.99e-05$	$(2.46e-04, 3.66e-04)$	308	0.997
mttHat	$1.22e+02$	$1.76e+01$	$(9.22e+01, 1.61e+02)$	400	1.002
circ0Hat	$5.35e+00$	$4.72e-01$	$(4.44e+00, 6.32e+00)$	400	0.993
gamma	$1.94e-01$	$1.53e-02$	$(1.66e-01, 2.27e-01)$	303	1.009
sigmaNeut	$9.92e-02$	$5.59e-03$	$(8.93e-02, 1.10e-01)$	400	1.003
omega[1]	$5.00e-01$	$2.57e-01$	$(2.27e-01, 1.11e+00)$	400	1.003
omega[2]	$7.30e-01$	$3.08e-01$	$(3.67e-01, 1.59e+00)$	400	1.011
omega[3]	$5.83e-01$	$2.69e-01$	$(2.68e-01, 1.18e+00)$	400	0.998
omega[4]	$3.85e-01$	$1.58e-01$	$(1.86e-01, 7.22e-01)$	335	1.005
omega[5]	$5.33e-01$	$2.49e-01$	$(2.18e-01, 1.07e+00)$	400	1.005
omega[6]	$4.10e-01$	$1.77e-01$	$(2.08e-01, 8.29e-01)$	310	1.003
omega[7]	$2.13e-01$	$9.64e-02$	$(1.08e-01, 4.95e-01)$	319	1.005
omega[8]	$1.77e-01$	$1.17e-01$	$(3.95e-02, 5.02e-01)$	336	0.996

Model fits (posterior median & 90 % CI)





Why Stan/Torsten?

Flexibility

- Flexible w.r.t. stochastic structure
  - Any number of levels variability
  - Large selection of built-in probability distributions
  - Permits sub-models with very different stochastic hierarchies

Why Stan/Torsten?

Flexibility

- Flexible w.r.t. stochastic structure
  - Any number of levels variability
  - Large selection of built-in probability distributions
  - Permits sub-models with very different stochastic hierarchies
- Flexible w.r.t. deterministic structure
  - Control structures: if-then-else, for and while loops
  - Large collection of built-in functions
  - Operators and functions for vector and matrix calculations

Why Stan/Torsten?

Flexibility

- Flexible w.r.t. stochastic structure
  - Any number of levels variability
  - Large selection of built-in probability distributions
  - Permits sub-models with very different stochastic hierarchies
- Flexible w.r.t. deterministic structure
  - Control structures: if-then-else, for and while loops
  - Large collection of built-in functions
  - Operators and functions for vector and matrix calculations

Computational efficiency

- Typically faster than Gibbs or Metropolis-Hastings
  - Measured in terms of $\frac{\text{time}}{\text{effective sample size}}$

Why Stan/Torsten?

Flexibility

- Flexible w.r.t. stochastic structure
  - Any number of levels variability
  - Large selection of built-in probability distributions
  - Permits sub-models with very different stochastic hierarchies
- Flexible w.r.t. deterministic structure
  - Control structures: if-then-else, for and while loops
  - Large collection of built-in functions
  - Operators and functions for vector and matrix calculations

Computational efficiency

- Typically faster than Gibbs or Metropolis-Hastings
  - Measured in terms of $\frac{\text{time}}{\text{effective sample size}}$
- Also includes optimization and variational inference methods for rapid approximate Bayesian analysis

Why Stan/Torsten?

Some alternative general purpose modeling tools with Bayesian analysis capabilities

- BUGS variants (WinBUGS, OpenBUGS, JAGS)
- SAS (Proc MCMC + SAS/IML)
- Pharmacometrics s/w: NONMEM & Monolix

attribute	Stan/Torsten	WinBUGS	OpenBUGS	JAGS	SAS Proc MCMC	NONMEM	Monolix
Flexibility wrt stochastic structure	++	++	++	++	+	−	−
Flexibility wrt deterministic structure	++	−	−	−	++	+	+
Built-in distributions	++	+	+	+	+	−	−
Discrete parameters	−	+	+	+	+	−	−
Support for PK models							
ODE solver(s)	+	+	±	−	+	+	+
Event schedules, e.g., multiple dosing	+	+	−	−	−	+	+
Within chain parallel computation	−	−	−	−	?	+	?
Active development program	+	−	−	+	+	+	+
Portability	++	−	−	++	+	++	++
Open source	+	−	+	+	−	−	−
Cost	+	+	+	+	−−−	−−	−−

Current role of Stan/Torsten for PMX applications

- Very flexible platform for fully Bayesian analyses that cannot be implemented in standard PMX platforms, e.g., NONMEM or Monolix, without substantial compromises.

Current role of Stan/Torsten for PMX applications

- Very flexible platform for fully Bayesian analyses that cannot be implemented in standard PMX platforms, e.g., NONMEM or Monolix, without substantial compromises.
- You can do more routine popPK and popPKPD analyses with Stan, particularly with the Torsten extensions, but
  - Computation times make it non-optimal for such applications

Current role of Stan/Torsten for PMX applications

- Very flexible platform for fully Bayesian analyses that cannot be implemented in standard PMX platforms, e.g., NONMEM or Monolix, without substantial compromises.
- You can do more routine popPK and popPKPD analyses with Stan, particularly with the Torsten extensions, but
  - Computation times make it non-optimal for such applications
- Bottom line: For the moment save it for problems where
  - Fully Bayesian methods are particularly useful, e.g., use of informative priors.
  - A more flexible model specification language is needed.

Barriers to routine use of Stan for PMX applications

- Computation time required for adequate MCMC sampling
- Programming time required to implement typical popPKPD models


THE SHAPE OF
STAN JAZZ TO COME
ORNETTE COLEMAN

Stan & Torsten development plans

- Fast approximate Bayesian methods
  - Gradient-based marginal optimization (GMO) for marginal maximum penalized likelihood estimation
  - Data-streaming variational Bayes via stochastic automatic differentiation variational inference (ADVI)
  - Data-parallel variational Bayes via expectation propagation (EP)

Stan & Torsten development plans

- Fast approximate Bayesian methods
  - Gradient-based marginal optimization (GMO) for marginal maximum penalized likelihood estimation
  - Data-streaming variational Bayes via stochastic automatic differentiation variational inference (ADVI)
  - Data-parallel variational Bayes via expectation propagation (EP)
- Within chain parallel computation

Stan & Torsten development plans

- Fast approximate Bayesian methods
  - Gradient-based marginal optimization (GMO) for marginal maximum penalized likelihood estimation
  - Data-streaming variational Bayes via stochastic automatic differentiation variational inference (ADVI)
  - Data-parallel variational Bayes via expectation propagation (EP)
- Within chain parallel computation
- DAEs
- PDEs
- SDEs

Stan & Torsten development plans

- Fast approximate Bayesian methods
  - Gradient-based marginal optimization (GMO) for marginal maximum penalized likelihood estimation
  - Data-streaming variational Bayes via stochastic automatic differentiation variational inference (ADVI)
  - Data-parallel variational Bayes via expectation propagation (EP)
- Within chain parallel computation
- DAEs
- PDEs
- SDEs
- R package to simplify implementation of pharmacometrics models
- R package for specialized visualization and reporting of PKPD model analyses

Stan & Torsten development plans

- Fast approximate Bayesian methods
  - Gradient-based marginal optimization (GMO) for marginal maximum penalized likelihood estimation
  - Data-streaming variational Bayes via stochastic automatic differentiation variational inference (ADVI)
  - Data-parallel variational Bayes via expectation propagation (EP)
- Within chain parallel computation
- DAEs
- PDEs
- SDEs
- R package to simplify implementation of pharmacometrics models
- R package for specialized visualization and reporting of PKPD model analyses

Addition of those features will make Stan/Torsten a superior open source alternative to (your favorite PMX platform here).

Stan & Torsten development plans

- Fast approximate Bayesian methods
  - Gradient-based marginal optimization (GMO) for marginal maximum penalized likelihood estimation
  - Data-streaming variational Bayes via stochastic automatic differentiation variational inference (ADVI)
  - Data-parallel variational Bayes via expectation propagation (EP)
- Within chain parallel computation
- DAEs
- PDEs
- SDEs
- R package to simplify implementation of pharmacometrics models
- R package for specialized visualization and reporting of PKPD model analyses

They will also make it plausible to use Stan for more complex PBPKPD and QSP modeling applications.

Credit where it is due...

Metrum Research Group

- Charles Margossian
- Yi Zhang
- Marc Gastonguay

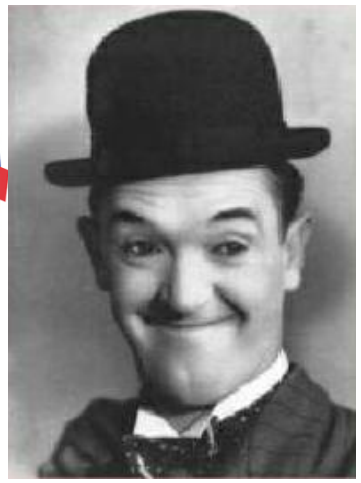
Stan developers

- Andrew Gelman, Columbia Univ.
- Bob Carpenter, Columbia Univ.
- Mike Betancourt, Columbia Univ.
- Daniel Lee, Columbia Univ.
- Sebastian Weber, Novartis
- + contributions from many more

Funding sources:

- Bill & Melinda Gates Foundation
- Office of Naval Research STTR Program
  - contract no. N00014-16-P-2039
  - contract no. N6833518C0110

A few of my favorite Stans



References I

[1] Stan Modeling Language Users Guide and Reference Manual, Version 2.17.1, 2018.
`http://mc-stan.org/`.

[2] RStan: the R interface to Stan, Version 2.7.0, 2015.
`http://mc-stan.org/rstan.html`.

[3] Radford M. Neal.
MCMC using hamiltonian dynamics.
In Steve Brooks, Andrew Gelman, Galin L. Jones, and Xiao-Li Meng, editors, *Handbook of Markov Chain Monte Carlo*, chapter 5, pages 113–162. Chapman & Hall/CRC, Boca Raton, FL, 2011.

[4] Matthew D Hoffman and Andrew Gelman.
The no-U-turn sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo.
The Journal of Machine Learning Research, 15(1):1593–1623, 2014.

[5] Sebastian Weber, Andrew Gelman, Daniel Lee, Michael Betancourt, Aki Vehtari, and Amy Racine-Poon.
Bayesian aggregation of average data: An application in drug development.
Annals of Applied Statistics `http://www.stat.columbia.edu/~gelman/research/published/extrap_paper_aoas.pdf`, 2018.

References II

[6] Yusuke Yamaguchi, Wataru Sakamoto, Shingo Shirahata, and Masashi Goto.
A meta-analysis method based on simulated individual patient data.
In *Bulletin of the ISI 58th World Statistics Congress of the International Statistical Institute*, pages 4702–4707, 2011.

[7] Yusuke Tamaguchi.
Meta-Analysis Methods Based on Simulated Individual Patient Data.
PhD thesis, THE GRADUATE SCHOOL OF ENGINEERING SCIENCE, OSAKA UNIVERSITY, Osaka, Japan, March 2013.

[8] Sebastian Weber, Andrew Gelman, Bob Carpenter, Daniel Lee, Michael Betancourt, Aki Vehtari, and Amy Racine.
Hierarchical expectation propagation for bayesian aggregation of average data.
arXiv preprint arXiv:1602.02055, 2016.

[9] S Weber, A Gelman, B Carpenter, D Lee, F Bois, and A Racine.
Bayesian aggregation of average data in hierarchical models.
Bayes 2015, Basel, Switzerland, May 19–21 2015.

[10] L. E. Friberg, A. Henningsson, H. Maas, L. Nguyen, and M. O. Karlsson.
Model of chemotherapy-induced myelosuppression with parameter consistency across drugs.
J Clin Oncol, 20(24):4713–21, 2002.

References III

[11] L. E. Friberg and M. O. Karlsson.
Mechanistic models for myelosuppression.
Invest New Drugs, 21(2):183–194, 2003.

[12] J. E. Latz, M. O. Karlsson, J. J. Rusthoven, A. Ghosh, and R. D. Johnson.
A semimechanistic-physiologic population pharmacokinetic/pharmacodynamic model for neutropenia following pemetrexed therapy.
Cancer Chemotherapy and Pharmacology, 57(4):412–426, 2006.

[13] I. F. Troconiz, M. J. Garrido, C. Segura, J. M. Cendros, P. Principe, C. Peraire, and R. Obach.
Phase i dose-finding study and a pharmacokinetic/pharmacodynamic analysis of the neutropenic response of intravenous diflomotecan in patients with advanced malignant tumours.
Cancer Chemother Pharmacol, 57(6):727–35, 2006.

[14] S. J. Kathman, D. H. Williams, J. P. Hodge, and M. Dar.
A bayesian population pk-pd model of ispinesib-induced myelosuppression.
Clin Pharmacol Ther, 81(1):88–94, 2007.

References IV

[15] Steven J Kathman, Daphne H Williams, Jeffrey P Hodge, and Mohammed Dar.
A bayesian population pk-pd model for ispinesib/docetaxel combination-induced myelosuppression.
Cancer Chemother Pharmacol, 63(3):469–476, 2009 Feb.

Additional slides

HMC performance

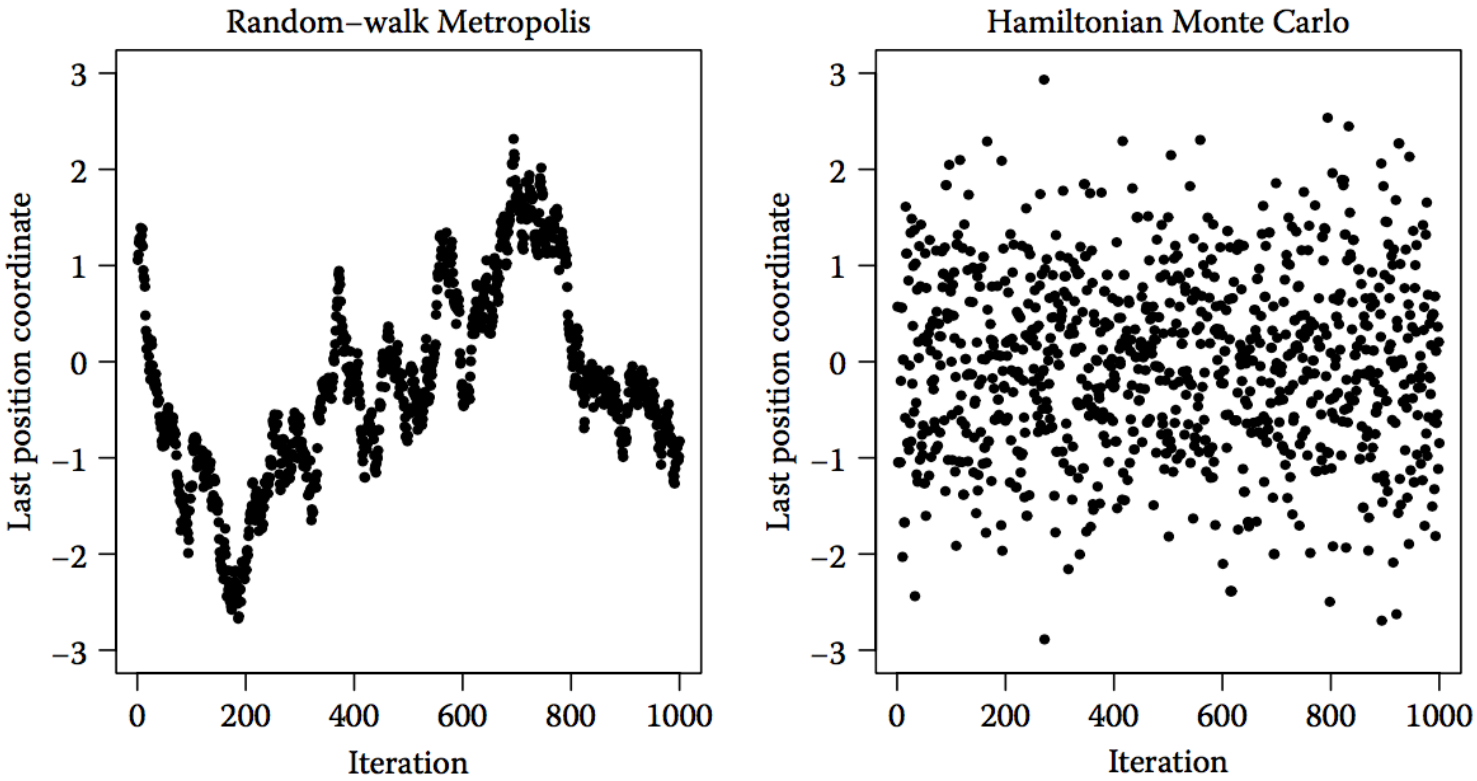


FIGURE 5.6
Values for the variable with largest standard deviation for the 100-dimensional example, from a random-walk Metropolis run and an HMC run with $L = 150$. To match computation time, 150 updates were counted as one iteration for random-walk Metropolis.

from RM Neal. MCMC Using Hamiltonian Dynamics (2011) [3]

HMC performance

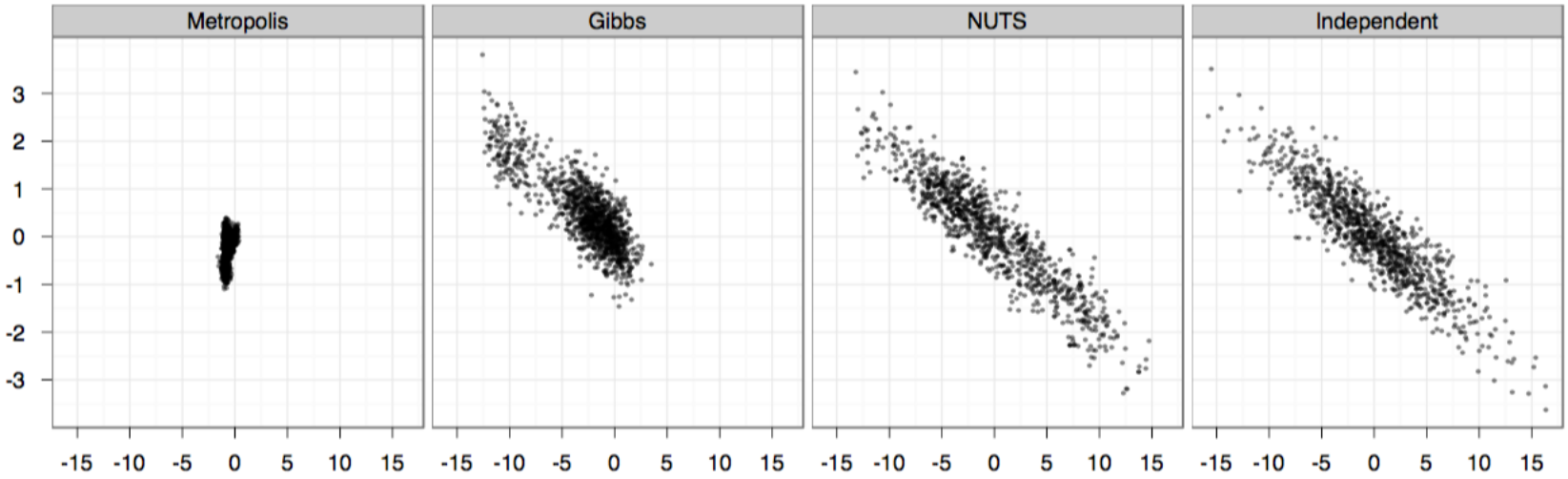


Figure 7: *Samples generated by random-walk Metropolis, Gibbs sampling, and NUTS. The plots compare 1,000 independent draws from a highly correlated 250-dimensional distribution (right) with 1,000,000 samples (thinned to 1,000 samples for display) generated by random-walk Metropolis (left), 1,000,000 samples (thinned to 1,000 samples for display) generated by Gibbs sampling (second from left), and 1,000 samples generated by NUTS (second from right). Only the first two dimensions are shown here.*

from MD Hoffman and A Gelman. The no-U-turn sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo (2014) [4]

HMC issues/limitations

- Requires calculation of the gradient $\frac{d \log(p(\theta|Y))}{d\theta}$
- Suitable for sampling of continuous parameters only
  - Cannot sample discrete parameters
  - Discrete data is OK as long as the likelihood depends only on continuous parameters.
  - Models with discrete parameters, e.g., finite mixture models, can often be implemented by marginalizing out the discrete parameters.