Bayesian pharmacometric modeling with BUGS, NONMEM and Stan

Bill Gillespie

Metrum Research Group

24 October 2016

Bayesian pharmacometric modeling with BUGS, NONMEM and Stan

- Why Bayesian?
- Adapting available software for typical pharmacometric modeling tasks
    - Necessary components
    - PKPD modeling software
        - NONMEM: METHOD = BAYES
    - Adapting general purpose Bayesian software
        - WinBUGS + BUGSModelLibrary
        - Stan
- Pros & cons
- Wish list

Disclosure

- Metrum Research Group is actively involved in the development of open source Stan functionality to support pharmacometrics applications.
- Supported in part by ONR STTR grant N00014-16-P-2039—a collaboration with Andrew Gelman and members of the Stan team at Columbia University.

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.

Adapting available software for typical pharmacometric modeling tasks

Common elements of pharmacometric model-based analyses

- PK and/or PD models described in terms of first order ODEs
  - Some have analytic solutions, e.g., linear 1, 2 and 3 compartment PK models,
  - But many require numerical solutions.

Adapting available software for typical pharmacometric modeling tasks

Common elements of pharmacometric model-based analyses

- PK and/or PD models described in terms of first order ODEs
  - Some have analytic solutions, e.g., linear 1, 2 and 3 compartment PK models,
  - But many require numerical solutions.
- Model calculations that depend on a sequence of events
  - Doses
  - Changes in covariate values
  - “Reset” events, e.g., zeroing out the amount in the compartment representing cumulative renal excretion when urine is collected

PKPD modeling software
NONMEM: METHOD = BAYES

NONMEM was designed to support pharmacometric applications involving nonlinear mixed effects models.

- Venerable history reaching back to 1980.
- Includes a model specification language, a variety of built-in PK models and numerical ODE solvers that permit specification of more complex PK and PD models.
- Most recent versions (7.*) also include an MCMC method (Gibbs/Metropolis-Hastings) that allows fully Bayesian analysis.
- An HMC/NUTS algorithm is implemented in version 7.4 alpha.

NONMEM: METHOD = BAYES

NONMEM is primarily designed for nonlinear mixed effects models of the form

$$
\begin{aligned}
y_{ij} &\sim p\left(\widehat{y}_{ij} | \theta_j, X_{ij}\right) \\
\widehat{y}_{ij} &= f\left(X_{ij}, \theta_j\right) \\
\theta_i &\sim N\left(\widehat{\theta}, \Omega\right)
\end{aligned}
$$

where y_{ij} is observed data for the i^{th} occasion in the j^{th} individual, p is either a normal or user-specified conditional likelihood, and X_{ij} are independent variables, e.g., time.

- Though version 7.* provides methods for more levels of nested random effects (normally distributed).
- Prior distributions are limited to normal for $\widehat{\theta}$ and inverse Wishart for Ω.

NONMEM: METHOD = BAYES

Features include:

- PREDPP component provides several built-in PK models and ODE solvers
  - Linear 1, 2 and 3 compartment models using analytic solutions
  - General linear compartmental models using numerical calculation of matrix exponential
  - General nonlinear compartmental models using numeral solution of ODEs via DVERK (5th/6th order Runge Kutta), DGEAR (Gear's method for stiff ODEs) or LSODA (automatic switching between methods for stiff and non-stiff problems)
- Flexible FORTRAN-like language for specifying the conditional likelihood

NONMEM: METHOD = BAYES

Features include:

- Event handling
  - Accommodates complicated event schedules without requiring custom programming by the user
- Parallel computation that takes advantage of the hierarchical model structure
  - Allows within chain parallelization

Adapting general purpose Bayesian software
WinBUGS + BUGSModelLibrary

BUGSModelLibrary (`https://bitbucket.org/metrumrg/bugsmodellibrary/`) is a PKPD model library for use with WinBUGS 1.4.3.

- Specific linear compartmental models:
  - One compartment model with first order absorption
  - Two compartment model with elimination from and first order absorption into central compartment
- General linear compartmental model described by a matrix exponential
- General compartmental model described by a system of first order ODEs

BUGSModelLibrary

- Event handling based on NONMEM/NMTRAN/PREDPP conventions
- Implemented NMTRAN data items include:
  - TIME, EVID, CMT, AMT, RATE, ADDL, II, SS
- Models based on general linear and nonlinear ODEs require user specification of a rate constant matrix or ODE's in a template Component Pascal procedure that must be compiled using the BlackBox Component Builder 1.5.

Adapting general purpose Bayesian software
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 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

Stan features 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 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 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.

Stan pharmacometrics resources

- Torsten: Prototype library of PKPD functions for Stan
  - Set of Stan functions that replicates the functionality of NONMEM's PREDPP library
  - For details see our poster T-09 "Stan Functions for Bayesian Pharmacometric Modeling" by Charles Margossian & William R Gillespie.
  - Current version of Torsten is available at: `https://github.com/charlesm93/example-models/blob/feature/issue-70-PKPDexamples-torsten/PKPD/torsten/README.md`

Stan pharmacometrics resources

- PMXStan
  - By Yuan Xiong & Wenping Wang, Novartis
  - Similar objectives to Torsten
  - R package + Stan functions
  - Uses a modified version of LSODA for numerical solution of ODEs
  - http://discuss.go-isop.org/t/introduction-to-pmxstan-an-r-library-to-facilitate-pkpd-modeling-with-stan/554
  - Not yet publicly available (but I understand they're working on it)
- Examples of models written in Stan language
  - by Sebastian Weber, Novartis
  - https://github.com/stan-dev/example-models/tree/feature/issue-72-stan-pkpdlib/misc/pkpd

Torsten: Prototype library of PKPD functions for Stan

Functions in current prototype:

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

Torsten: Prototype library of PKPD functions for Stan

Functions in current prototype:

- 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: Prototype library of PKPD functions for Stan

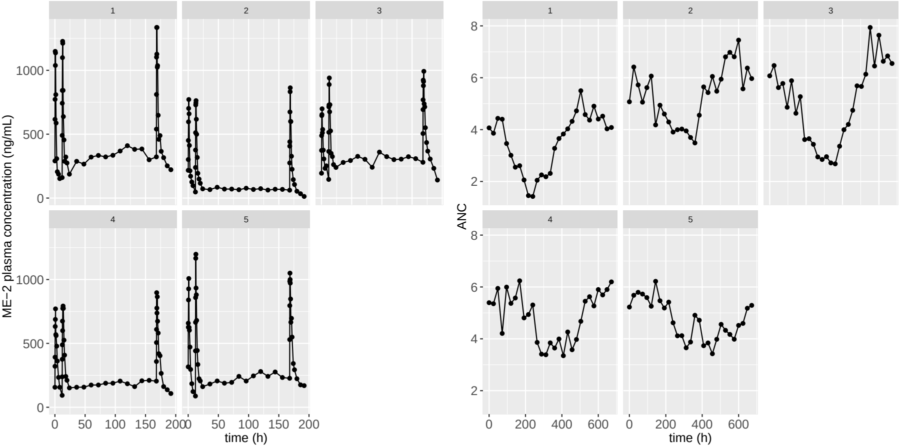
Functions in current prototype:

- 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 PMX functions

- Uses NONMEM/PREDPP conventions for data specification and event handling
- Recursive calculation: For each event time calculate the amount in each compartment given the compartment amounts plus doses at the previous event time.
- Steady-state (SS) currently implemented only for PKModelOneCpt, PKModelTwoCpt and linCptModel.
- A work in progress—more features to come.

Torsten example: PKPD model of drug-induced neutropenia

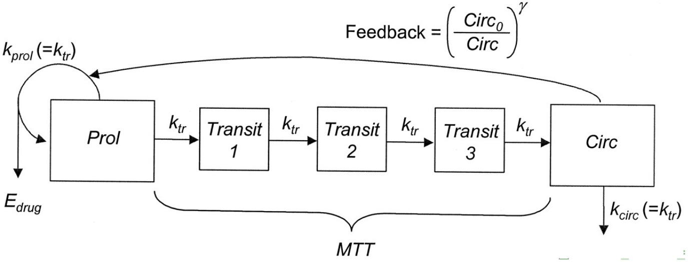


Friberg-Karlsson semi-mechanistic model for drug-induced myelosuppression

- 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 [2, 3, 4, 5, 6, 7]



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

$$
\begin{aligned}
\frac{dProl}{dt} &= k_{prol} Prol\,(1 - E_{drug})\left(\frac{Circ_0}{Circ}\right)^{\gamma} - k_{tr} Prol \\
\frac{dTransit1}{dt} &= k_{tr} Prol - k_{tr} Transit1 \\
\frac{dTransit2}{dt} &= k_{tr} Transit1 - k_{tr} Transit2 \\
\frac{dTransit3}{dt} &= k_{tr} Transit2 - k_{tr} Transit3 \\
\frac{dCirc}{dt} &= k_{tr} Transit3 - k_{circ} Circ
\end{aligned}
$$

$$
\begin{aligned}
\widehat{c} &\equiv \text{plasma drug concentration} \\
Circ &\equiv \text{absolute neutrophil count (ANC)}
\end{aligned}
$$

$$
\begin{aligned}
E_{drug} &= \alpha \widehat{c} \\
k_{prol} &= k_{circ} = k_{tr} \\
MTT &= \frac{n+1}{k_{tr}}
\end{aligned}
$$

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

IIV and prior distributions

- Inter-individual variation

$$\log\left(CL_j, Q_j, V_{1j}, V_{2j}, k_{aj}, MTT_j, Circ_{0j}, \alpha_j\right)$$
$$\sim \quad N\left(\log\left(\widehat{CL}\left(\frac{bw_j}{70}\right)^{0.75}, \widehat{Q}\left(\frac{bw_j}{70}\right)^{0.75}, \widehat{V}_1\left(\frac{bw_j}{70}\right), \widehat{V}_2\left(\frac{bw_j}{70}\right), \widehat{k}_a, \widehat{MTT}, \widehat{Circ}_0, \widehat{\alpha}\right), \Omega\right)$$

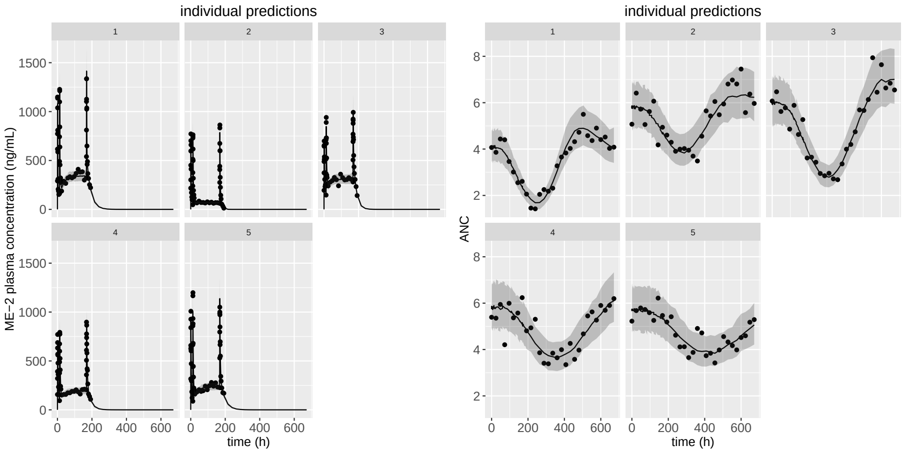
- Prior distributions: moderately informative for PK, strongly informative for system parameters, weakly informative for drug effect

$$\begin{aligned}
\widehat{CL} &\sim \log N\left(\log(10), 0.5\right) \quad \widehat{Q} \sim \log N\left(\log(15), 0.5\right) \quad \widehat{V}_1 \sim \log N\left(\log(35), 0.5\right) \\
\widehat{V}_2 &\sim \log N\left(\log(105), 0.5\right) \quad \widehat{k}_a \sim \log N\left(\log(2), 0.5\right) \\
\widehat{MTT} &\sim \log N\left(\log(125), 0.2\right) \quad \widehat{Circ}_0 \sim \log N\left(\log(5), 0.2\right) \quad \gamma \sim \log N\left(\log(0.17), 0.2\right) \\
\widehat{\alpha} &\sim \log N\left(\log(3 \times 10^{-4}), 1\right) \quad \sigma \sim \text{half-Cauchy}\left(0, 1\right) \\
\Omega &= \text{diag}\left(\omega\right) P \,\text{diag}\left(\omega\right) \\
\omega_i &\sim \text{half-Cauchy}\left(0, 1\right), i \in \{1, 2, \ldots, 8\} \qquad P \sim \text{LKJCorr}\left(1\right)
\end{aligned}$$

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)



Pros & cons: NONMEM

- Pros
  - Flexible model specification language for the conditional likelihood of an observation
  - Built-in handlers for event schedules encountered in PKPD data
  - Good numerical ODE solvers
  - Support for parallel computations within chain
  - Steady-state calculations even for ODE-based models
  - Optimization for estimation of posterior modes

Pros & cons: NONMEM

- Pros
  - Flexible model specification language for the conditional likelihood of an observation
  - Built-in handlers for event schedules encountered in PKPD data
  - Good numerical ODE solvers
  - Support for parallel computations within chain
  - Steady-state calculations even for ODE-based models
  - Optimization for estimation of posterior modes
- Cons
  - Restricted stochastic model structure
  - Very restricted choice of prior distributions
  - Relatively expensive and not open source

Pros & cons: WinBUGS + BUGSModelLibrary

- Pros
  - Flexible model specification language
  - Many built-in functions and distributions
  - Built-in handlers for event schedules encountered in PKPD data
  - Good numerical ODE solvers
  - Steady-state calculations even for ODE-based models
  - Freely available

Pros & cons: WinBUGS + BUGSModelLibrary

- Pros
  - Flexible model specification language
  - Many built-in functions and distributions
  - Built-in handlers for event schedules encountered in PKPD data
  - Good numerical ODE solvers
  - Steady-state calculations even for ODE-based models
  - Freely available
- Cons
  - Limited portability: Windows app. Requires Wine or similar to run on *nix platforms.
  - ODE models require writing/compiling a Component Pascal model
  - Lack of control structures like true loops and if-then-else in BUGS language
  - BUGSModelLibrary has not been ported to OpenBUGS or JAGS.
  - WinBUGS 1.4.3 is not open source
  - Little or no continued development of BUGS and the BlackBox Component Builder

Pros & cons: Stan

- Pros
  - HMC/NUTS sampler often performs better than the Gibbs/Metropolis samplers in NONMEM and BUGS
  - Very flexible imperative model specification language (vs BUGS declarative language)
    - Many built-in functions and distributions
    - Easy to create user-defined functions
    - Control structures like for loops, while loops, if-then-else
    - Vector and matrix operators and functions
    - Can directly specify likelihood without resorting to tricks
    - Good ODE solvers
  - Active development program
  - Freely available and open source

Pros & cons: Stan

- Pros
  - HMC/NUTS sampler often performs better than the Gibbs/Metropolis samplers in NONMEM and BUGS
  - Very flexible imperative model specification language (vs BUGS declarative language)
    - Many built-in functions and distributions
    - Easy to create user-defined functions
    - Control structures like for loops, while loops, if-then-else
    - Vector and matrix operators and functions
    - Can directly specify likelihood without resorting to tricks
    - Good ODE solvers
  - Active development program
  - Freely available and open source
- Cons
  - No built-in PMX models
  - No built-in handlers for PKPD event schedules
  - Steady-state calculations for ODE models not readily implemented
  - These features will be available soon (already available in prototypes).

Wish list

Flexible general purpose Bayesian software with:

- Numerical solvers for
  - Systems of algebraic equations (root solver)
  - Differential algebraic equations
  - Delay differential equations
  - Stochastic differential equations
  - Partial differential equations
- Approximate Bayesian method(s) that permits parallel computation, e.g., expectation propagation
- Within chain parallel computation for some classes of hierarchical models

Fan mail from some frequentists

Bayesian (bey' -zh*uh*n) *n.* **1.** Result of breeding a statistician with a clergyman to produce the much sought after "honest statistician"[a]. **2.** One who asks you what you think before a clinical trial in order to tell you what you think afterwards[b]. **3.** One who, vaguely expecting a horse, and catching a glimpse of a donkey, strongly believes he has seen a mule[c].

[a]anonymous

[b]S Senn. Statistical Issues in Drug development, 2nd Ed. Wiley, 2008. p. 51.

[c]ibid. p. 46

Fan mail from some frequentists

Bayesian (bey' -zh*uh*n) *n.* **1.** Result of breeding a statistician with a clergyman to produce the much sought after "honest statistician"[a]. **2.** One who asks you what you think before a clinical trial in order to tell you what you think afterwards[b]. **3.** One who, vaguely expecting a horse, and catching a glimpse of a donkey, strongly believes he has seen a mule[c].

---

[a] anonymous

[b] S Senn. Statistical Issues in Drug development, 2nd Ed. Wiley, 2008. p. 51.

[c] ibid. p. 46

A posterior distribution

References I

[1] Stan Modeling Language Users Guide and Reference Manual, Version 2.8.0, 2015.
`http://mc-stan.org/`.

[2] 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.

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

[4] 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.

[5] 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.

References II

[6] 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.

[7] 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.