Hierarchical ODE Models Using Stan
20.05.2015

Bayes Pharma 2015
Jan Serroyen, Nele Goeyvaerts, Tom Haber
Overview

- ExaScience Life Lab
- Stan
- Application: PK-PD model
- Stan-CVODE
- Truncating the derivative
PRE-COMPETITIVE RESEARCH
OPPORTUNITIES FOR COMPUTING

Better simulations and statistics
• E.g. Biostatistics, systems biology, PBPK, molecular dynamics

Better predictions
• E.g. Chemogenomics, toxicology

Analysis of very large amounts of data
• E.g. Genomics, large compound libraries, sensor data

Aims:
• Improve **disease knowledge** and **effectiveness** of medicines
  • Pharma provide the **challenges** and the **data**
• Leverage **broad computing expertise** to do this
• **Pool resources** to improve capabilities
What is Stan?

• “Probabilistic programming language implementing full Bayesian statistical inference”
  – MCMC sampling (Hamiltonian MC, NUTS)
  – Maximum likelihood estimation (BFGS)
• Coded in C++ and runs on all major platforms
• Open-source software (+ maintained): http://mc-stan.org/
• Standalone software, or interfaces with R, Python, Matlab, Julia
• HMC uses gradient information → less affected by correlations between parameters than random walk MC
Application: Pre-Clinical PK-PD Experiment

- Screening experiment on enzyme blocker

- Data:
  - 12 animals
  - 3 dose groups (oral)
  - 7 PK measurements over time per animal (drug concentration)
  - 3 PD measurements over time per animal (enzyme concentration)

- PK-PD model:
  - Pharmacokinetics: first-order one-compartment model
  - Pharmacodynamic: turnover model
Pharmacokinetic Model

First-order one-compartment model:

\[
\begin{align*}
\frac{dC_g}{dt} &= -k_a C_g \\
\frac{dC_p}{dt} &= \left( k_a C_g - k_e C_p \right) / V_f
\end{align*}
\]

Where

- \( C_g \) is the drug concentration in the GI tract (latent)
- \( C_p \) is the drug concentration in the blood plasma (observed)
- \( k_a \) is the absorption rate
- \( k_e \) is the elimination rate
- \( V_f \) is the (apparent) volume of distribution
- Initial conditions (\( t_0 \)): \( C_g(0) = \text{dose} \), \( C_p(0) = 0 \)
Pharmacodynamic Model

Turnover model:

\[
\frac{dR}{dt} = k_{in} \left( 1 - \frac{I_{max}C_p}{IC_{50}C_p} \right) - k_{out}R
\]

Where
- \( R \) is the enzyme concentration (observed)
- \( k_{in} \) is the production rate
- \( k_{out} \) is the elimination rate
- \( I_{max} \) is the maximal inhibition (fixed to 100%)
- \( C_p \) corresponds to the plasma concentration
- \( IC_{50} \) is the plasma concentration required to obtain 50% inhibition
- \( k_{in}/k_{out} \) is animal (species) dependent \( \rightarrow \) historical data
Bayesian PK-PD Model

- Model parameters:
  - Fixed effects (6): $\mu_{Ka}$, $\mu_{Ke}$, $\mu_{Vf}$, $\mu_{Kin}$, $\mu_{Kout}$, $\mu_{IC50}$
  - Random (subject-specific) effects (6x12)
  - Variance components (6+2): random effect variances + resid. error

- Priors:
  - Fixed effects: Normal(0, 100)
  - Random effects: Normal(Fixed, VarComp)
  - Variance components: Cauchy(0, 2.5)
  - Note: priors are automatically truncated by Stan based on user-specified bounds on parameters
First Attempts in Stan

• Problem: Stan seems to easily get stuck (not updating) when fitting ODE models

• Attempted solutions that didn’t work:
  – Specify “well-chosen” starting values for model parameters
  – Put upper bounds on model parameters (besides lower bounds)
  – Modify NUTS tuning parameter settings

• Underlying issue: no stopping criterion for step-halving
  → current ODE solver (Boost) is unstable
Contributions from ExaScience Lab

• More complex models

• Bug fixes:
  – Memory leak (later incorporated into Stan 2.6)
  – Initial condition ODE (t0): removed restriction (timepoints ≠ t0)

• Implemented better ODE solver: **CVODE** (Sundials)
  – Currently in Stan: only Runge-Kutta (simple/non-stiff)
  – CVODE: can deal with difficult (stiff/unstable) models
  – Jacobian: built using the auto-diff system of Stan

• Stan development team (Daniel Lee) is currently looking at Stan-CVODE implementation
Results and Impressions

• Correlations induced by hierarchical setting, combined with

• Non-linearity of ODE system:
  – Difficult to navigate parameter space
    → HMC: small step sizes & many leapfrog steps (e.g. 1023 steps)
    → Many calls to ODE solver per iteration
  – Long warmup phase needed

• Elapsed time :
  ~4 hours for 650 iterations (including 150 warm-up iterations)
  Limited options for speedup: parallel chains
### Posterior: Summary Statistics

Inference for Stan model: stanPKPD.
2 chains, each with iter=650; warmup=150; thin=1;
post-warmup draws per chain=500, total post-warmup draws=1000.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mean</th>
<th>SE Mean</th>
<th>n_eff</th>
<th>Rhat</th>
</tr>
</thead>
<tbody>
<tr>
<td>muKa</td>
<td>0.770</td>
<td>0.012</td>
<td>241.090</td>
<td>1.012</td>
</tr>
<tr>
<td>muKe</td>
<td>1.061</td>
<td>0.010</td>
<td>228.917</td>
<td>1.014</td>
</tr>
<tr>
<td>muVf</td>
<td>9.374</td>
<td>0.089</td>
<td>213.036</td>
<td>1.017</td>
</tr>
<tr>
<td>muKin</td>
<td>84.419</td>
<td>1.022</td>
<td>158.511</td>
<td>1.009</td>
</tr>
<tr>
<td>muKout</td>
<td>0.167</td>
<td>0.001</td>
<td>225.177</td>
<td>1.010</td>
</tr>
<tr>
<td>muIC50</td>
<td>22.575</td>
<td>0.474</td>
<td>263.675</td>
<td>1.000</td>
</tr>
<tr>
<td>sigmaEps[1]</td>
<td>67.044</td>
<td>0.188</td>
<td>1000.000</td>
<td>1.001</td>
</tr>
<tr>
<td>sigmaEps[2]</td>
<td>2.331</td>
<td>0.040</td>
<td>198.920</td>
<td>1.007</td>
</tr>
<tr>
<td>sigmaKa</td>
<td>0.334</td>
<td>0.010</td>
<td>232.765</td>
<td>1.000</td>
</tr>
<tr>
<td>sigmaKe</td>
<td>0.307</td>
<td>0.025</td>
<td>43.428</td>
<td>1.034</td>
</tr>
<tr>
<td>sigmaVf</td>
<td>2.530</td>
<td>0.076</td>
<td>218.520</td>
<td>1.009</td>
</tr>
<tr>
<td>sigmaKin</td>
<td>13.444</td>
<td>2.789</td>
<td>21.563</td>
<td>1.109</td>
</tr>
<tr>
<td>sigmaKout</td>
<td>0.049</td>
<td>0.002</td>
<td>144.723</td>
<td>1.005</td>
</tr>
<tr>
<td>sigmaIC50</td>
<td>11.457</td>
<td>0.706</td>
<td>160.463</td>
<td>1.000</td>
</tr>
</tbody>
</table>

**Fixed effects**

**Variance components**
Trace Plots: PK model (fixed effects)

- Trace of muka
- Trace of muke
- Trace of muVf

Iterations (without warmup)
Trace Plots: PD model (fixed effects)
PK Model: Fitted Curves

Subject: {42}  Subject: {44}  Subject: {45}  Subject: {46}

Subject: {49}  Subject: {50}  Subject: {51}  Subject: {52}

Subject: {53}  Subject: {54}  Subject: {55}  Subject: {56}

PK: drug concentration

Time (hours)
PD Model: Fitted Curves

subject: {42}  subject: {44}  subject: {45}  subject: {46}

subject: {49}  subject: {50}  subject: {51}  subject: {52}

subject: {53}  subject: {54}  subject: {55}  subject: {56}

PD enzyme concentration

Time (hours)
Installation Script RStan-CVODE (Linux)

git clone https://github.com/stan-dev/rstan.git
cd rstan
git checkout -b cvode_rstan 7d6bf44c5b45b061e8e0f0b32f0a81050f37d4fb
git submodule update --init
cd stan
git remote add tomh https://github.com/tomhaber/stan.git
git fetch tomh
git checkout cvode
cd ../rstan
 Make

# Install Sundials library (Ubuntu/Debian)
sudo apt-get install libsundials-serial libsundials-serial-dev

# Start RStudio with CVODE preloaded
LD_PRELOAD=/usr/lib/lib sundials_cvode.so:/usr/lib/lib sundials_nvecserial.so rstudio
Alternative Fix: Truncating the Derivative

- Aim: stabilize ODE solution
- \( \text{dy}^* = \min(\max(\text{dy}, -a), a) \)
- Truncating ("clipping") is more straightforward than constraining all parameters individually

• Best used in combination with upper bounds on parameters
Alternative Fix: Truncating the Derivative

- Stan functions:

```stan
real clip(real dy, real a) {
    real dyclipped;
    dyclipped <- (fabs(dy + a) - fabs(dy - a)) / 2;
    return dyclipped;
}

real[] ode(real t, real[] y, real[] theta, real[] x_r, int[] x_i) {
    real dydt[3];
    dydt[1] <- clip(-theta[1] * y[1], 10);
    return dydt;
}
```
Summary and Conclusion

• Bayesian hierarchical ODE models were successfully fitted in Stan
• Default ODE solver is not stable
  → replaced by CVODE
  → Alternative: truncate the derivatives
• Our PK-PD model needs further refinement
  e.g. non-centered parametrization
• Checking quality and robustness of model fit for complex ODE models is not straightforward (and time consuming)
Acknowledgements

- Tom Jacobs, Janssen R&D
- Jeroen Tolboom, Janssen R&D
Thank you!