

Prototype Model Library for Bayesian Pharmacokinetic/Pharmacodynamic (PKPD) Modeling in Winbugs William R Gillespie and Marc R Gastonguay

Metrum Institute, Tariffville, CT 06081 USA

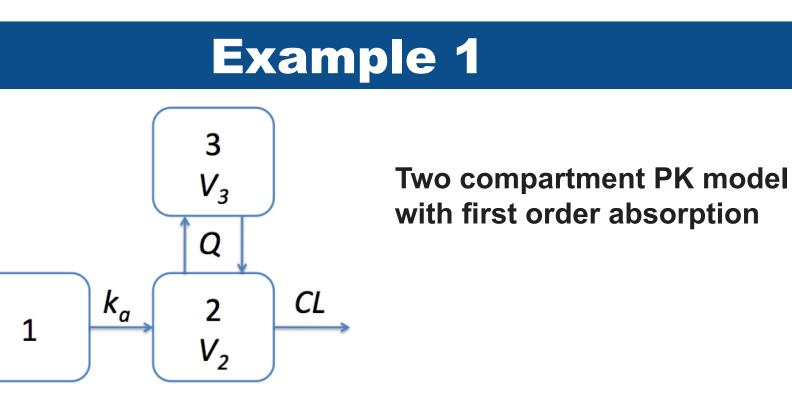
Background

- There is increasing interest in and use of Bayesian methods for PKPD modeling.
- A barrier to wider use of Bayesian PKPD modeling is the amount of custom programming required for each application.
- PKBugs and Pharmaco [1] partially address this problem by providing functions for linear 1, 2 and 3 compartment PK models and data management tools for WinBUGS [2],
- But they are limited to a small number of models and do not properly handle time-varying covariates
- There is a need for a more comprehensive model library comparable in scope to NON-MEM®/PREDPP [3].
- The combination of the BUGS language and a flexible PKPD model library would facilitate modeling and simulation tasks where the outcomes are causally related to drug exposure, e.g., models for clinical outcomes, adverse events, dropouts, and clinical trial simulations.

Objective

To support fully Bayesian PKPD modeling by developing a collection of BUGS language functions that implement or facilitate user implementation of nonlinear-mixed effects population PKPD models.

Implemented PKPD Models



BUGS language model for fitting a two compartment model.

model

for(i in 1:nsub)-

Inter-patient variatio ogtheta[i, 1:5] ~ dmnorm(logthetaMean[i, 1:5], omega.inv[1:5, 1:5]) ogthetaMean[i, 1] <- logCLHat + 0.75*log(weight[start[i]]/70) # CL 2] <- logQHat + 0.75*log(weight[start[i]]/70) # Q</p> 3] <- logV1Hat + log(weight[start[i]]/70)</p> # V1 # V2 ogthetaMean[i, 4] <- logV2Hat + log(weight[start[i]]/70)</pre> # ka - alphai ogthetaMean[i, 5] <- logDkaHat theta[i,6] <- 1 # F1 theta[i,7] <- 1 # F2 theta[i,8] <- 1 # F3 theta[i,9] <- 0 # tlag1 theta[i,10] <- 0 # tlag2 theta[i,11] <- 0 # tlag3 for(j in 1:5){

Specification of the matrix exponential form of the effect compartment model. This involves user specification of the non-zero elements of the rate constant matrix in the Component Pascal file TwoCptEffCptModel.odc.

PROCEDURE UserKMatrix(IN theta: ARRAY OF REAL: nCmt: INTEGER) POINTER TO ARRAY OF ARRAY OF REAL; VAR kMatrix: POINTER TO ARRAY OF ARRAY OF REAL; i, j: INTEGER; CL, Q, V2, V3, dka, ke0, k10, k12, k21, ksum, ka: REAL; BEGIN NEW(kMatrix,nCmt,nCmt); (* Initialize to all zeros *) FOR i := 0 TO nCmt-1 DO; FOR j := 0 TO nCmt-1 DO; kMatrix[i,j] := 0; END; END; (* 2 cpt model with 1st order absorption *) CL := theta[0];Q := theta[1]; V2 := theta[2] V3 := theta[3]; dka := theta[4] ke0 := theta[5] k10 := CL/V2; k12 := Q/V2; k21 := Q/V3; ksum := k10 + k12 + k21; ka := dka + (ksum - Math.Sqrt(ksum*ksum-4.0*k10*k21))/2.0; (* Assign nonzero rate constants *) kMatrix[0,0] := -ka; kMatrix[1,0] := ka;

Specification of differential equations in the Component Pascal file: TwoCptIndEff1Model.odc.

```
PROCEDURE UserDerivatives(IN theta, x: ARRAY OF REAL;
numEq: INTEGER; t: REAL; OUT dxdt: ARRAY OF REAL) ;
VAR
CL, Q, V2, V3, dka, kout, yeff0, EC50, ka, k10, k12, k21, ksum, conc: REAL;
BEGIN
CL := theta[0]
Q := theta[1];
V2 := theta[2]
V3 := theta[3]
dka := theta[4];
kout := theta[5]
yeff0 := theta[6]
EC50 := theta[7]
k10 := CL/V2;
k12 := Q/V2;
k21 := Q/V3;
ksum := k10 + k12 + k21;
ka := dka + (ksum - Math.Sqrt(ksum*ksum-4.0*k10*k21))/2.0;
(* Differential equations for the model excluding piecewise *)
```

(* constant input rates provided in the data set *)

```
dxdt[0] := -ka * x[0];
dxdt[1] := ka * x[0] - (k10 + k12) * x[1] + k21 * x[2];
dxdt[2] := k12 * x[1] - k21 * x[2];
```

The current prototype BUGS PKPD model library contains functions implementing:

- 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 compartment models described by a matrix exponential function

- General compartment models described by a system of first order ODE's.

The one and two compartment models are precompiled.

General linear or nonlinear compartment models 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 [4].

The models and data format are based on NONMEM®/NMTRAN/PREDPP conventions including:

Recursive calculation of model predictions

This permits piecewise constant covariate values

Bolus or constant rate inputs into any compartment

- Handles single dose, multiple dose and steady-state dosing histories

- Implemented NMTRAN data items include:

TIME, EVID, CMT, AMT, RATE, ADDL, II, SS

Implementation Details

Core model library procedures are programmed in Component Pascal and compiled with the BlackBox Component Builder.

• An extensible object-oriented programming approach is used to facilitate development of additional models.

The BUGS language interface to WinBUGS 1.4.3 is implemented using WBDev [5].

• One and two compartment models:

- Analytical calculations programmed in Component Pascal.

• General linear compartmental models:

Component Pascal wrapper procedures that call DLLs programmed in FOR-

log(theta[i,j]) <- logtheta[i,j]</pre>

Call to PK model library to calculate amount in each compartment at each time xhat[start[i]:end[i],1:3] <- TwoCptModel(time[start[i]:end[i]], amt[start[i]:end[i]]</pre> rate[start[i]:end[i]], ii[start[i]:end[i]], evid[start[i]:end[i]] cmt[start[i]:end[i]], addl[start[i]:end[i]], ss[start[i]:end[i]], theta[i,]]

for(i in 1:nobs)

logCobs[i] ~ dnorm(logCHat[i],tau) CHat[i] <- 1000*xhat[i,2]/theta[subject[i],3] logCHat[i] <- log(max(CHat[i],eps))</pre>

Prior distributions ogCLHat ~ dnorm(0,1.0E-6) logQHat ~ dnorm(0,1.0E-6) logV1Hat ~ dnorm(0,1.0E-6) logV2Hat ~ dnorm(0,1.0E-6) logDkaHat ~ dnorm(0,1.0E-6) log(CLHat) <- logCLHat log(QHat) <- logQHat log(V1Hat) <- logV1Hat log(V2Hat) <- logV2Hat log(DkaHat) <- logDkaHa tau <- 1/(sigma*sigma) ~ dunif(0,1000) omega.inv[1:5, 1:5] ~ dwish(omega.inv.prior[1:5, 1:5], 5) omega[1:5, 1:5] <- inverse(omega.inv[1:5, 1:5]) eps <- 1.0E-6

study 1 5 mg						study 1 10 mg					study 1 20 mg										
0 5 10 20 0 5 10 20 46 47 48 49 50			i	0 510 20 0 510 20					0 5 10 20 0 5 10 20 96 97 98 99 100					h							
	0 - 0 -	h	1	٨	1	A		300 200 100 0	٨	٨	٨	1	٨		600 400 200	L	1	h	À	٨	
		41	42	43	44	45			66	67	68	69	70	300	300 - 200 -	91	92	93	94	95	- 600
c		<u> </u>	N	<u> </u>	A	<u> </u>	- 100 - 50 - 0	c	٨	N-	N	A	A	200 100 0	c .	٨	Å	٨	A	h	- 600 - 400 - 200 - 0
atio		36	37	38	39	40	-	concentration 0 00 0 00 0 00 0 00 0 00 0 00 0 00 0	61	62	63	64	65	L.	000 tratio	86	87	88	89	90	
	0	<u> </u>	A	h	A	N			<u> </u>	<u> </u>	Å	A	٨	Ę	00000000000000000000000000000000000000	٨	A	N	<u>k</u>	٨	
00		31	32	33	34	35		0	56	57	58	59	60	± 300	0	81	82	83	84	85	- 600
		K	<u> </u>	<u> </u>	N	٨	- 100 - 50 - 0	300 200 100 0	٨	N	A	٨	۸	200 100 0		٨		٨	<u> </u>	n	- 400 - 200 - 0
	0	26	27	0 510 20	29	30			51 0 510 20	52	53	54	0 510 20		600 400 200 0	0 510 20	77	78 0 510 20	79	80	
				dose							dose							dose			
individual predictions						individual predictions				individual predictions											
study 1 40 mg						study 2 20 mg				study 2 20 mg											
		121	0 510 20	123	0 510 20	125			146	0 50100	148	0 50100	150	1		171	0 50100	173	0 50100	175	-
	- 000 - 000 - 0	116	117	118	119	120	-	600 400 200	141	142	143	143 144	145	- 600 - - 400 - - 200 - - 0 -		166	167	168	169	170	-
	-		~		L.,	K.	- 1000			142	145	144	140 F+++	- 600 - 400 - 200			107	- IOO		-	- 600 - 400 - 200
tion	-	111	112	113	114	115	- 0	tion	136	137	138	139	140	- 0	tion	161	162	163	164	165	- 0
	00 - 00 -	٨	1	1	h	~	-	Centration 005 000 000	Ann	1	~~		-		centration 005 009	~		N		A	-

kMatrix[1,1]	:= -(k10+k12);	
kMatrix[1,2]	:= k21;	
kMatrix[2,1]	:= k12;	
kMatrix[2,2]	:= -k21;	
kMatrix[3,1]	:= ke0;	
kMatrix[3,3]	:= -ke0;	

RETURN kMatrix;

END UserKMatrix;

PROCEDURE (m: MatExpModel) InitModel*; BEGIN

m.nParameter := 12; m.F1Index := 6; m.tlag1Index := 10; m.nCmt := 4; END InitModel

	study 1 5 mg	study 1 10 mg	study 1 20 mg				
	0 510 20 0 510 20 46 47 48 49 50	0 510 20 0 510 20 71 72 73 74 75	0 5 10 20 0 5 10 20 96 97 98 99 100				
	<u>41 42 43 44 45</u> 60 20	66 67 68 69 70 60 20	<u>91 92 93 94 95</u> 100 50				
se	-20 <u>36</u> 37 <u>38</u> <u>39</u> <u>40</u> 60	-20					
response	20 -20 -31 -32 -33 -34 -35						
	26 27 28 29 30 60 20	51 52 53 54 55 60 20	100 - 76 77 78 79 80 50 - 77 78 79 80				
	-20 -20		0 510 20 0 510 20 0 510 20				
	dose	dose	dose				
	individual predictions	individual predictions	individual predictions				
	study 1 40 mg	study 2 20 mg	study 2 20 mg				
	0 510 20 0 510 20	0 50100 0 50100 146 147 148 149 150					
	100 - 121 122 123 124 125 50 - 50 - 50 - 50 - 50 - 50 - 50 - 50						
			0 - 166 167 168 169 170				
			100				
	00						
æ							
sponse	111 112 113 114 115 100 50						
response	100 100 0 106 107 108 109 110 109 100 100 100 100 100	9 100 - - - 0 136 137 138 139 140 0 50 - - - - 131 132 133 134 135 -	$\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 $				
response	111 112 113 114 115 100 50 0 	0 0 100 0 0 0 0 0 0 0 0 0 0 0 0	100 100 50 0 0 0 0 0 0 0 0 0 0 0 0 0				
	100 111 112 113 114 115 100 0 106 107 108 109 110 100 50	100 136 137 138 139 140 100 131 132 133 134 135 100 131 132 133 134 135	$\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 $				
	$\begin{array}{c} 100 \\ 101 \\ 100 \\ 0 \\ 0 \\ 106 \\ 107 \\ 108 \\ 109 \\ 101 \\ 102 \\ 103 \\ 104 \\ 105 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\$	$\begin{array}{c} 32 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ $	$\begin{array}{c} 30\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0$				
	$\begin{array}{c} 100 \\ 111 \\ 112 \\ 113 \\ 114 \\ 115 \\ 100 \\ 0 \\ 106 \\ 107 \\ 108 \\ 109 \\ 110 \\ 100 \\ 50 \\ 101 \\ 102 \\ 103 \\ 104 \\ 105 \\ 104 \\ 105 \\ 104 \\ 105 \\ 100 \\ 50 \\ 0 \\ 100 \\$	$\begin{array}{c} 3 \\ 3 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\$	$\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 $				
	$\begin{array}{c} 0 \\ 111 \\ 112 \\ 113 \\ 114 \\ 115 \\ 100 \\ 0 \\ 106 \\ 107 \\ 108 \\ 109 \\ 110 \\ 100 \\ 100 \\ 100 \\ 101 \\ 102 \\ 103 \\ 104 \\ 105 \\ 0 \\ 0 \\ 50 \\ 0 \\ 0 \\ 50 \\ 0 \\ 0 \\ 50 \\ 0 \\ $	$\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 $	$\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 $				
	100 111 112 113 114 115 0 0 106 107 108 109 110 100 100 100 100 100 100	9 100 136 137 138 139 140 0 100 131 132 133 134 135 100 100 131 132 133 134 135 100 100 131 132 133 134 135 100 100 100 131 132 133 134 135 100 100 100 131 132 133 134 135 100 100 100 100 131 132 133 134 135 100 100 100 100 100 131 132 133 134 135 100 100 100 100 100 100 100 10	$\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 $				
	100 101 102 103 104 105 104 105 100 105 100 100 100 100 100	900 100 100 100 100 100 100 100	$\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 $				
	100 111 112 113 114 115 0 0 106 107 108 109 110 100 100 100 100 100 100	9 9 100 136 137 138 139 140 0 100 131 132 133 134 135 100 100 126 127 128 129 130 0 50 100 100 100 100 100 100	9 9 9 9 9 9 9 9 9 9 9 9 9 9				
	100 111 112 113 114 115 0 0 106 107 108 109 110 100 100 100 100 100 100	9 9 100 100 100 100 100 100 100	$\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 $				

conc := x[1]/V2; dxdt[3] := kout * (yeff0 - (1 - conc/(EC50+conc))*(x[3]+yeff0)) (* x[3] = yeff - yeff0 *)

END UserDerivatives;

Predicted (posterior median and 90% credible intervals) and observed response 2 measurements.

study 1 0 mg	study 1 5 mg	study 1 10 mg						
0 510 20 0 510 20	0 510 20 0 510 20	0 510 20 0 510 20						
200 21 22 23 24 25	46 47 48 49 50	71 72 73 74 75						
		250 - 150						
50 16 17 18 19 20	41 42 43 44 45	50 66 67 68 69 70						
- 200	- 150	E 250						
100 - 50	- 100 - 50	- 150 50						
11 12 13 14 15	9 36 37 38 39 40 9 •	61 62 63 64 65						
	50 - <u>31 32 33 34 35</u>	2 50 <u>56 57 58 59 60</u>						
- 200 - 150	- 150	250						
	26 27 28 29 30	51 52 53 54 55 50						
200		250						
0 510 20 0 510 20 0 510 20	0 510 20 0 510 20 0 510 20	0 510 20 0 510 20 0 510 20						
dose	dose	dose						
individual predictions	individual predictions	individual predictions						
study 1 20 mg	study 1 40 mg	study 2 20 mg						
0 5 10 20 0 5 10 20 96 97 98 99 100	0 510 20 0 510 20	0 50100 0 50100						
350 -	500							

91 92 93 94 95 350	<u>116 117 118 119 120</u>	141 142 143 144 145						
250								
86 87 88 89 90 50	111 112 113 114 115	126 127 128 120 140						
9 350 60 87 88 89 90 50 50 50 50 50 50 50 50 50 50 50 50 50 5								
81 82 83 84 85	0 100 - 106 107 108 109 110	80 131 132 133 134 135						
350								
76 77 78 79 80	101 102 103 104 105							
350 250								
0 5 10 20 0 5 10 20 0 5 10 20	0 5 10 20 0 5 10 20 0 5 10 20	0 50100 0 50100 0 50100						
dose individual predictions	dose individual predictions	dose individual predictions						
study 2 20 mg	study 2 20 mg	study 2 20 mg						
0 50100 0 50100	0 50100 0 50100	0 50100 0 50100						
171 172 173 174 175	196 197 198 199 200	221 222 223 224 225						
P88	PR8	216 217 218 219 220						
1 588	E 1888							
2 588	2 488	211 212 213 214 215 2 588						
9 500 9 500 9 500 9 500 156 157 158 159 160								
156 157 156 155 166	181 182 183 184 185	200 207 208 209 210						
151 152 153 154 155	176 177 178 179 180	201 202 203 204 205						
· · · · · · · · · · · · · · · · · · ·								
0 50100 0 50100 0 50100 dose	0 50100 0 50100 0 50100 dose	0 50100 0 50100 0 50100 dose						
individual predictions	individual predictions	individual predictions						

- TRAN.
- The DLLs are generated from Expokit [6] for matrix exponential calculations a LAPACK [7] for matrix calculations.
- General compartmental models:
 - Numerical solution of ODEs via Runge-Kutta 4th/5th order method.
 - Uses ODE solver code provided (but not documented) in WinBUGS.
 - Steady-state calculations based on solution of boundary value problem. Requi numerical solution of algebraic equations. Uses procedures from LibSolve [8].
 - Programmed in Component Pascal.

Simulated Examples

Phase 1 study in healthy volunteers

- Parallel dose-escalation design
- 25 subjects per dose arm
- Single doses

Placebo, 5, 10, 20 and 40 mg

- **—** PK: plasma concentrations of parent drug (*c*)
- PD: 2 different responses

Response 1: Emax function of effect compartment concentration (F

Response 2: Indirect effect model with drug effect on kout (inhibitor Emax) (R_2)

and R_{2}).

- PK and PD measured at 0, 0.125, 0.25, 0.5, 0.75, 1, 2, 3, 4, 6, 8, 12, 18 and 24 hours after dose.

Phase IIa trial in patients

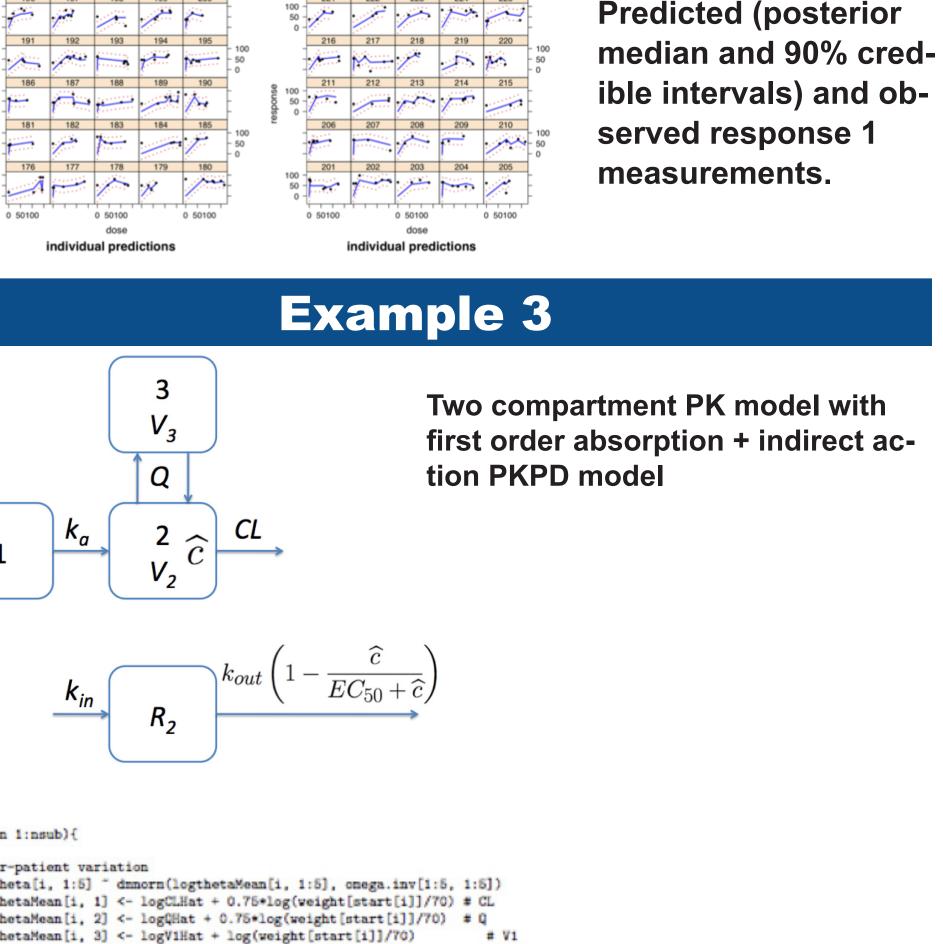
Multiple doses

20 mg bid (q12h) x 7 days

100 patients per treatment arm

- Sparse PK data (3-6 samples/patient)

nd	$ \frac{9}{600} - \frac{100}{100} - \frac{100}{126} - \frac{110}{127} - \frac{1128}{128} - \frac{129}{130} - \frac{100}{100} - \frac{100}{126} - \frac{1100}{100} - \frac{100}{100} -$	⁸ ⁹ ¹⁵⁶ ¹⁵⁷ ¹⁵⁸ ¹⁵⁹ ¹⁶⁰ ⁶⁰⁰ ²⁰⁰ ⁰⁰⁰ ¹⁵¹ ¹⁵² ¹⁵³ ¹⁵⁴ ¹⁵⁴ ¹⁵⁵ ¹⁵⁵ ¹⁵⁵ ¹⁵⁵ ¹⁵⁰ ⁰⁰⁰ ²⁰⁰ ⁰⁰⁰ ²⁰⁰ ⁰⁰⁰ ⁰⁰⁰ ²⁰⁰ ⁰⁰⁰ ¹⁵¹ ¹⁵¹ ¹⁵² ¹⁵³ ¹⁵⁴ ¹⁵⁵ ¹⁵⁵ ¹⁵⁰ ¹⁵¹	a b	90 90 90 90 90 90 90 90 90 90
ires		Predicted (posterior me- dian and 90% credible intervals) and observed plasma drug concentra- tions.	$\begin{bmatrix} 3 \\ V_3 \\ Q \\ Q \\ Q \\ CL \end{bmatrix} CL$	Example Two co first or tion Pk
	V_3 first ord Q compare 1 k_a 2 CL	mpartment PK model with der absorption + effect rtment PKPD model	$1 \qquad k_{in} \qquad k_{out} \qquad k$	$\left(1 - \frac{\widehat{c}}{EC_{50} + \widehat{c}}\right)$
R₁) ry 24	$\widehat{R}_{1} = \frac{E_{\max} c_{e}}{EC_{50} + e}$		<pre>for(i in 1:nsub){ # Inter-patient variation logtheta[i, 1:5] ~ dnnorn(logthetaMean logthetaMean[i, 1] <- logCLHat + 0.75*1 logthetaMean[i, 2] <- logQHat + 0.75*1 logthetaMean[i, 3] <- logV1Hat + log(v logthetaMean[i, 4] <- logV2Hat + log(v logthetaMean[i, 5] <- logDkaHat logtheta[i, 6:8] ~ dnnorn(logthetaMean logthetaMean[i, 6] <- logKoutHat logthetaMean[i, 7] <- logYeff0Hat logthetaMean[i, 8] <- logEC50Hat </pre>	<pre>elog(weight[start[i]]/70) # CL log(weight[start[i]]/70) # Q weight[start[i]]/70) # V1 weight[start[i]]/70) # V2 # 1</pre>
	<pre># Inter-patient variation logtheta[i, 1:5] ~ dnnorn(logthetaMean[i, 1:5], onega.inv[1:5, 1:5]) logthetaMean[i, 1] <- logCLHat + 0.75*log(weight[start[i]]/70) # CL logthetaMean[i, 2] <- logQHat + 0.75*log(weight[start[i]]/70) # V1 logthetaMean[i, 3] <- logV1Hat + log(weight[start[i]]/70) # V2 logthetaMean[i, 4] <- logV2Hat + log(weight[start[i]]/70) # V2 logthetaMean[i, 5] <- logDkaHat # ka - alpha1 logtheta[i,6] ~ dnorn(logKeOHat,tauKeO) # ke0 theta[i,7] <- 1 # F1 theta[i,8] <- 1 # F2 theta[i,9] <- 1 # F3 theta[i,10] <- 0 # tlag1</pre>		<pre>theta[i,9] <- 1 # F1 theta[i,10] <- 1 # F2 theta[i,11] <- 1 # F3 theta[i,12] <- 1 # F4 theta[i,13] <- 0 # tlag1 theta[i,14] <- 0 # tlag2 theta[i,16] <- 0 # tlag3 theta[i,16] <- 0 # tlag4 for(j in 1:8){</pre>	BUC a tw with an i



V1

V2

ka - alphal

BUGS language model for fitting

a two compartment PK model

with first order absorption and

an indirect action PKPD model.

ndividual predictions ndividual prediction

Development Plans

ndividual predictions

Revise prototype library using better programming practices to enhance extensibility, maintainability, reliability and efficiency. Specific changes include:

- Use of factory design pattern.
- Adaptive allocation of arrays to eliminate limits imposed by fixed size arrays while also avoiding inefficiencies due to overuse of dynamic allocation.
- Implement additional ODE solvers, e.g., LSODA, CVODE, etc.
- Improve method used for general compartmental model steady-state calculations.
- Implement additional models to be determined in collaboration with potential users.
- Develop documentation for both users and developers
- Develop OpenBUGS interface for the model library.
- Execute rigorous and well-documented testing of the model library components.
- Conduct beta testing

BUGSModelLibrary Distribution

- BUGSModelLibrary is an active, open-source project of Metrum Institute (http://www.metruminstitute.org).
- BUGSModelLibrary may be downloaded from http://bugsmodellibrary.googlecode.com.

 $\log(c_{ij}) \sim N(\log(\widehat{c}_{ij}), \sigma^2)$ $\widehat{c}_{ij} = f_{2cpt}(t_{ij}, D_j, \tau_j, CL_j, Q_j, V_{1j}, V_{2j}, k_{aj})$ $\log(CL_j, Q_j, V_{ssj}, k_{aj}) \sim$ $N\left(\log\left(\widehat{CL}\left(\frac{bw_j}{70}\right)^{0.75}, \widehat{Q}\left(\frac{bw_j}{70}\right)^{0.75}, \widehat{V}_{ss}\left(\frac{bw_j}{70}\right), \widehat{k}_a\right), \Omega\right)$ $V_{1j} = f_{V_1}V_{ssj}$ $V_{2j} = (1 - f_{V_1}) V_{ssj}$ $\left(\widehat{CL}, \widehat{Q}, \widehat{V}_{ss}, \widehat{k}_{a}, f_{V_{1}}\right) = (10 \text{ L/h}, 15 \text{ L/h}, 140 \text{ L}, 2 \text{ h}^{-1}, 0.25)$ $\Omega = \begin{pmatrix} 0.25^2 & 0 & 0 & 0 \\ 0 & 0.25^2 & 0 & 0 \\ 0 & 0 & 0.25^2 & 0 \\ 0 & 0 & 0 & 0.25^2 \end{pmatrix} \quad \sigma = 0.1$ $R_{1ij} \sim N\left(\hat{R}_{1ij}, \sigma_{R_1}^2\right)$ $\widehat{R}_{1ij} = rac{E_{max}c_{eij}^{\gamma}}{EC_{50j}^{\gamma}+c_{eij}^{\gamma}}$ $c_{e\cdot j}' = k_{e0j} (c_{\cdot j} - c_{e\cdot j})$ $\log (EC_{50j}, k_{e0j}) \sim N \left(\log \left(\widehat{EC}_{50}, \widehat{k}_{e0} \right), \Omega_{R_1} \right)$ $\left(E_{max}, \widehat{EC}_{50}, \gamma, \widehat{k}_{e0}\right) = (100, 100.7, 1.07, 1)$ $\Omega_{R_1} = \left(egin{array}{cc} 0.2^2 & 0 \ 0 & 0.25^2 \end{array}
ight) \quad \sigma_{R_1} = 10$ $\log (R_{2ij}) \sim N \left(\log \left(\widehat{R}_{2ij} \right), \sigma_{R_2}^2 \right)$ $\widehat{R}'_{2\cdot j} = k_{out,j} \left(R_{20j} - \left(1 - rac{\widehat{c}_{\cdot j}}{EC^{R_2}_{50j} + \widehat{c}_{\cdot j}} \right) \widehat{R}_{2\cdot j}
ight)$ $\log \left(R_{20j}, EC_{50j}^{R_2}, k_{out,j} \right) \sim N \left(\log \left(\widehat{R}_{20}, \widehat{EC}_{50}^{R_2}, \widehat{k}_{out} \right), \Omega_{R_2} \right)$ $\left(\widehat{R}_{20}, \widehat{EC}_{50}^{R_2}, \widehat{k}_{out}\right) = (100, 200, 0.5)$ $\Omega_{R_2} = egin{pmatrix} 0.25^2 & 0 & 0 \ 0 & 0.25^2 & 0 \ 0 & 0 & 0.25^2 \end{pmatrix} \sigma_{R_2} = 0.15$

theta[1,10] <- 0 # tlag1 theta[1,11] <- 0 # tlag2 theta[1,12] <- 0 # tlag3 logEC50[i] dnorn(logEC50Hat,tauEC50) log(EC50[i]) <- logEC50[i] for(j in 1:6){ log(theta[i,j]) <- logtheta[i,j]</pre> for(i in 1:nobs){ Model equations used to logCobs[i] ~ dnorn(logCHat[i],tauC) CHat[i] <- 1000*xhat[i,2]/theta[subject[i],3] simulate the plasma drug logCHat[i] <- log(max(CHat[i],eps))</pre> concentrations (c) and fxa[i] ~ dnorn(fxaMean[i],tauFxa) the two PD responses (R_{\star} CeHat[i] <- 1000*xhat[i,4]/theta[subject[i],3] logCeHat[i] <- log(nax(CeHat[i],eps))</pre> # Prior distributions logCLHat ~ dnorm(0,1.0E-6) logQHat ~ dnorm(0,1.0E-6) logV1Hat ~ dnorm(0,1.0E-6) logV2Hat ~ dnorm(0,1.0E-6) logDkaHat ~ dnorm(0,1.0E-6) log(CLHat) <- logCLHat log(QHat) <- logQHat log(V1Hat) <- logV1Hat log(V2Hat) <- logV2Hat log(DkaHat) <- logDkaHat logKeOHat ~ dnorm(0,1.0E-6) logEC50Hat ~ dnorm(0,1.0E-6) log(keOHat) <- logKeOHat log(EC50Hat) <- logEC50Hat tauC <- 1/(sigmaC*sigmaC) sigmaC ~ dunif(0,1000) tauFxa <- 1/(signaFxa*signaFxa)</pre> sigmaFxa ~ dunif(0,1000) omega[1:5, 1:5] <- inverse(omega.inv[1:5, 1:5]) omegaKe0 ~ dunif(0,1000) tauKe0 <- 1/(omegaKe0*omegaKe0) omegaEC50 ~ dunif(0,1000) tauEC50 <- 1/(omegaEC50*omegaEC50 eps <- 1.0E-6 Emax <- 100

Call to PK model library to calculate amount in each compartment at each time xhat[start[i]:end[i],1:4] <- TwoCptEffCptModel(time[start[i]:end[i]]</pre> ant[start[i]:end[i]], rate[start[i]:end[i]], ii[start[i]:end[i]], evid[start[i]:end[i]], cnt[start[i]:end[i]], addl[start[i]:end[i]], ss[start[i]:end[i]], theta[i,]) **BUGS** language model for fxaMean[i] <- Enax*CeHat[i]/(EC50[subject[i]]+CeHat[i])</pre> fitting a two compartment PK model with first order absorption and an effect compartment PKPD model. onega.inv[1:5, 1:5] ~ dwish(onega.inv.prior[1:5, 1:5], 5)

Call to PK model library to calculate amount in each compartment at each time xhat[start[i]:end[i],1:4] <- TwoCptIndEff1Model(time[start[i]:end[i]],</pre> ant[start[i]:end[i]], rate[start[i]:end[i]], ii[start[i]:end[i]], evid[start[i]:end[i]], cnt[start[i]:end[i]], addl[start[i]:end[i]], ss[start[i]:end[i]], theta[i,]) for(i in 1:nobs){ logCobs[i] ~ dnorn(logCHat[i],tauC) CHat[i] <- 1000*xhat[i,2]/theta[subject[i],3] logCHat[i] <- log(max(CHat[i],eps))</pre> logYeff[i] ~ dnorn(logYeffHat[i],tauYeff) YeffHat[i] <- 1000*(xhat[i,4] + theta[subject[i],7])</pre> logYeffHat[i] <- log(max(YeffHat[i],eps))</pre> logCLHat ~ dnorm(0,1.0E-6) logQHat ~ dnorm(0,1.0E-6) logV1Hat ~ dnorm(0,1.0E-6) logV2Hat ~ dnorm(0,1.0E-6) logDkaHat ~ dnorm(0,1.0E-6) log(CLHat) <- logCLHat log(QHat) <- logQHat log(V1Hat) <- logV1Hat log(V2Hat) <- logV2Hat log(DkaHat) <- logDkaHat logKoutHat ~ dnorm(0,1.0E-6) logYeff0Hat ~ dnorm(0,1.0E-6) logEC50Hat ~ dnorm(0,1.0E-6) log(koutHat) <- logKoutHat log(yeff0Hat) <- logYeff0Hat log(EC50Hat) <- logEC50Hat tauC <- 1/(sigmaC*sigmaC) sigmaC ~ dunif(0,1000) tauYeff <- 1/(sigmaYeff*sigmaYeff)</pre> signaYeff ~ dunif(0,1000) omega.inv[1:5, 1:5] ~ dwish(omega.inv.prior[1:5, 1:5], 5) onega[1:5, 1:5] <- inverse(onega.inv[1:5, 1:5])</pre> onegaPD.inv[1:3, 1:3] ~ dwish(onegaPD.inv.prior[1:3, 1:3], 3) omegaPD[1:3, 1:3] <- inverse(omegaPD.inv[1:3, 1:3]) eps <- 1.0E-6

log(theta[i,j]) <- logtheta[i,j]</pre>

log(thetaPred[i,j]) <- logthetaPred[i,j]</pre>

References

- 1. DJ Lunn, J Wakefield, A Thomas, N Best, D Spiegelhalter. PKBugs User Guide, Dept. Epidemiology & Public Health, Imperial College School of Medicine, London (1999). (http:// www.winbugs-development.org.uk/pkbugs/home.html & http://www.winbugs-development org.uk/pharmaco.html)
- 2. DJ Lunn, A Thomas, N Best, D Spiegelhalter. WinBUGS -- a Bayesian modelling framework: concepts, structure, and extensibility. Statistics and Computing, 10:325-337 (2000). (http://www.mrc-bsu.cam.ac.uk/bugs/winbugs/contents.shtml)
- 3. SL Beal, LB Sheiner NONMEM users guide (1998) (http://www.icondevsolutions.com/nonmem.htm)
- 4. Oberon Microsystems. BlackBox Component Builder 1.5 Users Manual (2005) (http://www. oberon.ch/blackbox.html)
- 5. DJ Lunn. WinBUGS Development Interface (WBDev). ISBA Bulletin, 10(3): 10-11 (2003) (http://www.winbugs-development.org.uk/wbdev.html)
- 6. RB Sidje. Expokit: A Software Package for Computing Matrix Exponentials. ACM Trans. Math. Softw., 24(1):130-156, 1998 (http://www.maths.uq.edu.au/expokit/)

7. E Anderson, Z Bai, C Bischof, S Blackford, J Demmel, J Dongarra, J Du Croz, A Greenbaum, S Hammarling, A McKenney, D Sorensen. LAPACK Users' Guide. SIAM (1999) (http://www.netlib.org/lapack/)

8. RD Campbell. LibSolve (http://www.zinnamturm.eu/downloadsIN.htm#Lib)

©2009 Metrum Research Group LLC copies of this poster are available at www.metruminstitute.com/publications