

# Bayesian Estimation in mixed effects models using Monolix and mlxR

Marc Lavielle<sup>1</sup>

<sup>1</sup>Inria Saclay

Bayes 2015  
Basel, Novartis Campus

- 1 Introduction
- 2 Estimation of the population parameters using Monolix
- 3 Estimation of the individual parameters using Monolix
- 4 Mlxtran: a powerful language for complex hierarchical models

# What is a model?

A joint probability distribution!

$p(y, \psi, \theta)$  : joint probability distribution of  $y$ ,  $\psi$  and  $\theta$ .

- $y = (y_{ij}, 1 \leq i \leq N, 1 \leq j \leq n_i)$  are the observations,
- $\psi = (\psi_i, 1 \leq i \leq N)$  are the individual parameters,
- $\theta$  is the vector of population parameters.

# What is a joint probability distribution?

A hierarchical model

$$p(y, \psi, \theta) = p(y|\psi, \theta)p(\psi|\theta)p(\theta)$$

- $p(y|\psi, \theta)$ : conditional distribution of the observations,
- $p(\psi|\theta)$  population distribution of the individual parameters,
- $p(\theta)$  the distribution of  $\theta$  can represent
  - the *inter population variability*,
  - the *uncertainty* on  $\theta$ ,
  - the *prior distribution* of the population parameters.

$$p(y, \psi, \theta) = p(y|\psi, \theta)p(\psi|\theta)p(\theta)$$

## 1) Estimation of the population parameters (only $y$ is given)

- *Maximum Likelihood approach*: maximize  $p(y|\theta) = \int p(y, \psi|\theta) d\psi$
- *Bayesian approach*: compute/maximize

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)}$$

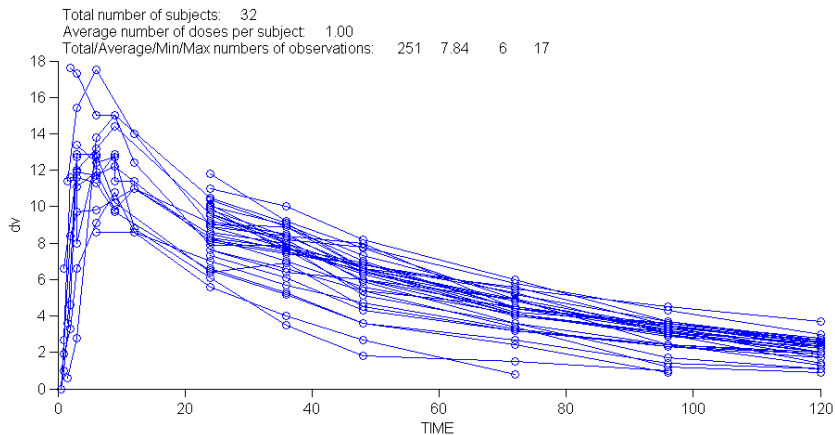
## 2) Estimation of the individual parameters ( $y$ and $\theta$ are given)

- *Individual approach*: maximize  $p(y|\psi, \theta)$
- *Population approach*: compute/maximize

$$p(\psi|y, \theta) = \frac{p(y|\psi, \theta)p(\psi|\theta)}{p(y, \theta)}$$

# A PK example

The warfarin PK data:



# A PK example

- The PK model:

$$C(t, \psi) = \frac{D k_a}{V k_a - Cl} \left( e^{-(Cl/V)t} - e^{-k_a t} \right).$$

- The model for the observed concentrations:

$$y_{ij} = C(t_{ij}, \psi_i) + a \varepsilon_{ij}$$

i.e.

$$y_{ij} | \psi_i, a \sim \mathcal{N}(C(t_{ij}, \phi_i), a^2)$$

- The model for the individual PK parameters  $\psi_i = (k_{a_i}, V_i, Cl_i)$ :

$$\log(k_{a_i}) \sim \mathcal{N}(\log(k_{a_{\text{pop}}}), \omega_{k_a}^2)$$

$$\log(V_i) \sim \mathcal{N}(\log(V_{\text{pop}}), \omega_V^2)$$

$$\log(Cl_i) \sim \mathcal{N}(\log(Cl_{\text{pop}}), \omega_{Cl}^2).$$

II

Estimation of the  
population parameters  
using Monolix



# Estimation of the population parameters

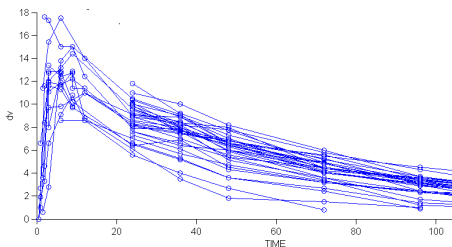
With Monolix, a given parameter can be

- **a fixed constant** if we have absolute confidence in its value or the data does not allow it to be estimated, essentially due to lack of identifiability.
- **estimated by maximum likelihood**, either because we have great confidence in the data or no information on the parameter.
- **estimated by introducing a prior** and calculating the maximum a posteriori (MAP) estimate or estimating the posterior distribution.

# Estimation of the population parameters

## ML estimation

The warfarin PK data provides a limited information about  $ka$



Estimation of the population parameters

	parameter	s.e. (lin)	r.s.e. (%)
ka_pop	: 0.587	0.12	20
V_pop	: 7.68	0.34	4
Cl_pop	: 0.135	0.0073	5
omega_ka	: 0.625	0.16	26
omega_V	: 0.221	0.036	16
omega_Cl	: 0.286	0.041	14
a	: 1.07	0.057	5

$\Rightarrow$  Let us introduce a prior for  $ka_{pop}$ :

$$\log(ka_{pop}) \sim \mathcal{N}(\log(ka^*), \gamma^2)$$

# Estimation of the population parameters

Monolix - 4.3.3- warfarinPK\_M0.mlxtran

Project Task Settings Workflow Graphics Test Tools ?

Run Assessment

### The data and model

The data: warfarin\_data.bt

The structural model: oral1\_1cpt\_kaVCl

Distribution of the individual parameters

Covariance model

1	0	0
0	1	0
0	0	1

Observation model

name	type	pred	error	r
y1	continuous	Cc	constant	<input type="button" value="v"/>

There is no covariate

### The initialization

Fixed effects:

Stand. dev. of the random effects:

Residual error parameters:

### The algorithms

123456

Numbers of iterations: K1: 500  auto; K2: 200  auto

Number of chains: 2  auto; Min Size: 50

Simulated Annealing:

Monte-Carlo sizes: Pred. dist.: 100; NPDE/VPC: 500; LL: 20000

Display: 50

### The results

Results folder:  Project name: warfarinPK\_M0;  User defined:

Standard errors:  Linearization;  Stoch. Approx.

Individual parameters:  Conditional modes;  Cond. means and s.d.

Log-likelihood:  Linearization;  Importance Sampling

Graphics:

# Estimation of the population parameters

Monolix - 4.3.3- warfarinPK\_M0.mlxtran

Project Task Settings Workflow Graphics Test Tools ?

Run Assessment

### The data and model

The data: warfarin\_data.bt

Distribution of the individual parameters:

The structural model: oral1\_tcpt\_kaVCl

There is no covariate

Covariance model

1	0	0
0	1	0
0	0	1

Observation model

name	type	pred	error	r
y1	continuous	Cc	constant	<input type="button" value="v"/>

### The initialization

Fixed effects: 1

Stand. dev. of the random effects:  1 1 1

Residual error parameters: 1 0 1 0

### The algorithms

123456

Numbers of iterations: K1: 500  auto; K2: 200  auto

Number of chains: 2  auto; Min Size: 50

Simulated Annealing:

Monte-Carlo sizes: Pred. dist: 100; NPDE/VPC: 500; LL: 20000

Display: 50

### The results

Results folder: Project name: warfarinPK\_M0 ; User defined: Browse

Standard errors:  Linearization;  Stoch. Approx.

Individual parameters:  Conditional modes;  Cond. means and s.d.

Log-likelihood:  Linearization;  Importance Sampling

Graphics:

# Estimation of the population parameters

## Combining ML and MAP estimations

The screenshot displays the Monolix 4.3.3 software interface for a project named 'warfarinPK\_M0.mlxtran'. A 'Bayesian estimation' dialog box is open, showing settings for the parameter 'theta: intercept for ka'. The 'Method' section has 'M.A.P.' selected. The 'Prior distribution' section shows a 'typical value (theta)' of 1, a 'std. deviation (Z)' of 0.1, and a 'distribution' of 'log-normal'. The 'theta=' field contains the expression 'exp(Z)', with a note 'where \*Z\* is normally distributed'. The background interface is divided into several sections: 'The data and model' (data: warfarin\_data.bt, model: oral\_1cpt\_kaVCl), 'The initialization' (fixed effects: 1 1 1), 'The algorithms' (iterations: K1=500, K2=200; chains: 2; simulated annealing: checked; monte-carlo sizes: 100, 500, 20000; display: 50), and 'The results' (results folder: warfarinPK\_M0; linearization: checked; stochastic approximation: unchecked; graphics: list button).

**Bayesian estimation**

theta: intercept for ka

Method:

- M.A.P.
- posterior distribution

Prior distribution:

typical value (theta):

std. deviation (Z):

distribution:

theta=

where \*Z\* is normally distributed

**The data and model**

The data: warfarin\_data.bt

The structural model: oral\_1cpt\_kaVCl

Observation model:

name	type	pred	error	r
y1	continuous	Cc	constant	<input type="checkbox"/>

**The initialization**

Fixed effects:

**The algorithms**

New seed:

Numbers of iterations: K1:  (auto checked), K2:  (auto checked)

Number of chains:  (auto checked), Min Size:

Simulated Annealing:

Monte-Carlo sizes: Pred. dist: ; NPDE/PC: ; LL:

Display:

**The results**

Results folder:  Project name: warfarinPK\_M0;  User defined: Browse

Standard errors:  Linearization;  Stoch. Approx.

Individual parameters:  Conditional modes;  Cond. means and s.d.

Log-likelihood:  Linearization;  Importance Sampling

Graphics:

# Estimation of the population parameters

## Combining ML and MAP estimations

Split  $\theta$  into  $(\theta_E, \theta_M)$  where

- $\theta_E$  are the components of  $\theta$  to be estimated with MLE,
- $\theta_M$  are the components of  $\theta$  to be estimated with MAP.

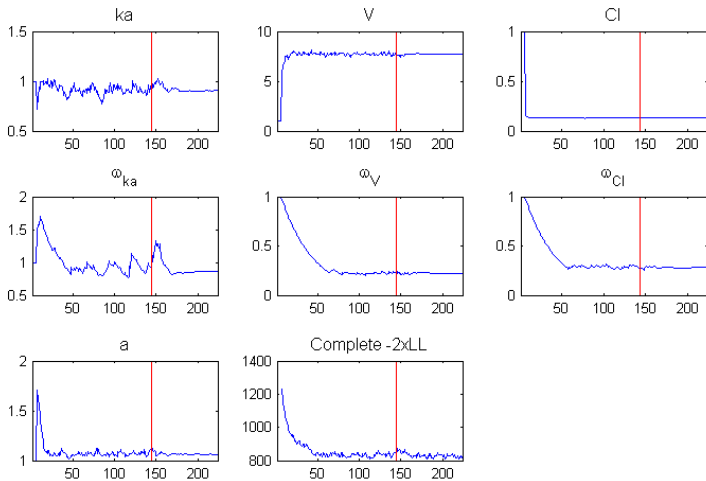
Then,  $(\hat{\theta}_E, \hat{\theta}_M)$  maximizes the penalized likelihood of  $(\theta_E, \theta_M)$ :

$$\begin{aligned}(\hat{\theta}_E, \hat{\theta}_M) &= \arg \max_{\theta_E, \theta_M} \log(p(y, \theta_M; \theta_E)) \\ &= \arg \max_{\theta_E, \theta_M} \{ \mathcal{LL}_y(\theta_E, \theta_M) + \log(p(\theta_M)) \},\end{aligned}$$

where  $\mathcal{LL}_y(\theta_E, \theta_M) \stackrel{\text{def}}{=} \log(p(y|\theta_M; \theta_E))$ .

# Estimation of the population parameters

Combining ML and MAP estimations



# Estimation of the population parameters

## Combining ML and MAP estimations

The MAP estimate of  $\theta = (ka_{\text{pop}}, V_{\text{pop}}, \dots)$  is the solution of the following minimization problem:

$$\hat{\theta}^{\text{MAP}} = \arg \min_{\theta} \left\{ -2\mathcal{L}\mathcal{L}_y(\theta) + \frac{1}{\gamma^2}(\log(ka_{\text{pop}}) - \log(ka^*))^2 \right\}.$$

We can compute the MAP estimate of  $ka_{\text{pop}}$  for various values of  $\gamma$  and  $ka^* = 1$ :

$\gamma$	0	0.02	0.03	0.2	0.3	$+\infty$
$\hat{ka}_{\text{pop}}^{\text{MAP}}$	1	0.991	0.963	0.809	0.623	0.587



# Estimation of the population parameters

## Combining ML and posterior distribution estimation

The screenshot displays the Monolix 4.3.3 interface for a project named 'warfarinPK\_M3'. The main window is divided into several sections: 'The data and model', 'The initialization', 'The algorithms', and 'The results'. A 'Bayesian estimation' dialog box is open in the center, showing settings for the parameter 'theta: intercept for ka'. The dialog box includes a 'Method' section with 'posterior distribution' selected, a 'Prior distribution' section with 'typical value (theta): 1', 'std. deviation (Z): 0.1', and 'distribution: log-normal'. The 'theta=' field is set to 'exp(Z)', with a note that 'Z' is normally distributed. The background interface shows 'The data and model' section with 'warfarin\_data.txt' as the data source and 'ora1\_1cpt\_kaVCI' as the structural model. 'The initialization' section shows 'Fixed effects' with values 1, 1, 1. 'The algorithms' section shows 'Numbers of iterations' for K1 (300), K2 (200), and Min Size (50). 'The results' section shows 'Results folder' as 'warfarinPK\_M3' and 'Standard errors' as 'Stoch. Approx.'. The 'Bayesian estimation' dialog box has 'Accept' and 'Cancel' buttons at the bottom.

**Bayesian estimation**

Method

M.A.P.

posterior distribution

Prior distribution

typical value (theta):

std. deviation (Z):

distribution:

theta =  where "Z" is normally distributed

# Estimation of the population parameters

Combining ML and posterior distribution estimation

Split  $\theta$  into  $(\theta_E, \theta_R)$  where

- $\theta_E$  are the components of  $\theta$  to be estimated with MLE,
- $\theta_R$  are the components of  $\theta$  whose posterior distribution is estimated.

Then,

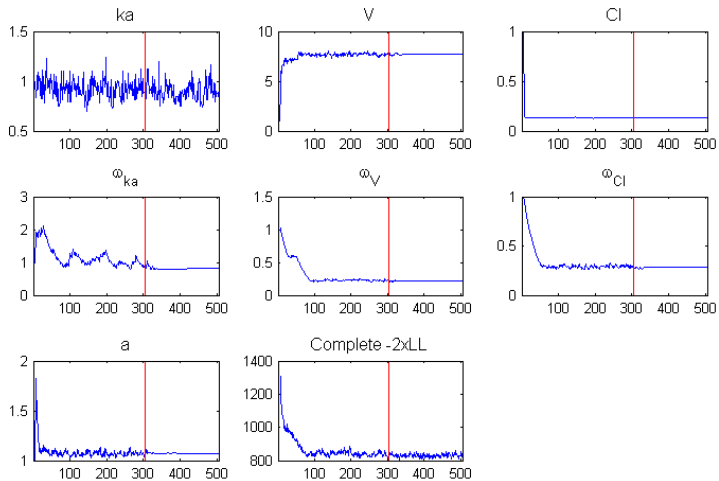
- i) Compute the maximum likelihood of  $\theta_E$ :

$$\begin{aligned}\hat{\theta}_E &= \arg \max_{\theta_E} p(y; \theta_E) \\ &= \arg \max_{\theta_E} \int p(y, \theta_R; \theta_E) d\theta_R.\end{aligned}$$

- ii) Estimate the conditional distribution  $p(\theta_R | y; \hat{\theta}_E)$ .

# Estimation of the population parameters

Combining ML and posterior distribution estimation



# Estimation of the population parameters

Combining ML and posterior distribution estimation

The screenshot displays the Monolix software interface. A dialog box titled "List of graphics" is open, showing a selection of various plots and reports. The "Posterior distribution" option is highlighted. The background shows the main project configuration screen for "warfarinPK\_M3.mlxtran".

**Monolix - 4.3.3- warfarinPK\_M3.mlxtran**

Project Task Settings Workflow Graphics Test Tools ?

Run Assessment

**List of graphics**

Selection: All Reduced Simulation Other

**Reduced**

Item	Count
Project Summary	1
Spaghetti Plot	1
Individual Fits	1
Obs. vs Pred.	1
Covariates	1
Parameters Distribution	1
Random Effects (boxplot)	1
Random Effects (joint dist.)	1
Convergence SAEM	1

**Simulation**

Residuals	1
VPC	1
Time to Event data	1
NPC	1
BLQ	1
Prediction Distribution	1

**Other**

Categorized Data	1
Transition Probabilities	1
Posterior distribution	1
Contribution to likelihood	1

Save OK Cancel

**Main Project Configuration**

Parameters: oral1\_1cpt\_kaVCl

Observation model

name	type	pred	error	r
y1	continuous	Cc	constant	

Residual error parameters

1	0	1	0
---	---	---	---

Monte-Carlo sizes

Pred. dist.	NPDE/VPC	LL	Display
100	500	20000	50

Results folder: warfarinPK\_M3

Standard errors: Stoch. Approx.

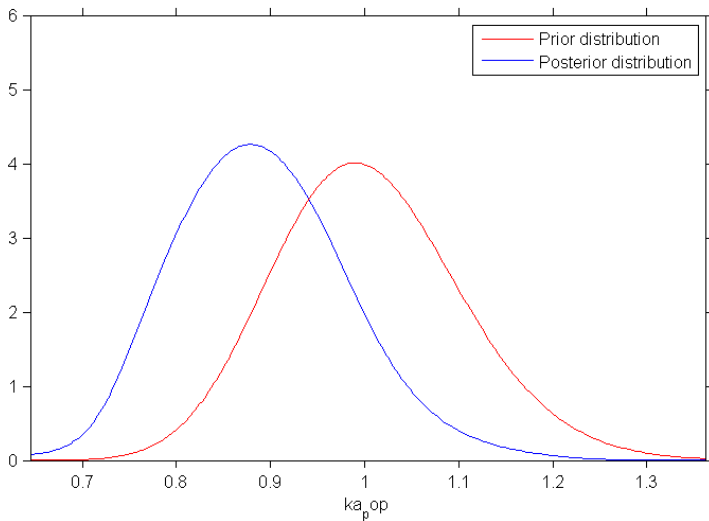
Individual parameters: Cond. means and s.d.

Log-likelihood: Importance Sampling

Graphics: List

# Estimation of the population parameters

Combining ML and posterior distribution estimation



III

Estimation of the  
individual parameters  
using Monolix

# Estimation of the individual parameters

Once  $\theta$  has been estimated, Monolix allows one:

- to compute the maximum a posteriori (MAP) estimate of  $\psi_i$ , for  $i = 1, 2, \dots, N$

$$\hat{\psi}_i^{\text{mode}} = \arg \max_{\psi_i} p(\psi_i | y_i; \hat{\theta})$$

- to estimate the conditional distribution  $p(\psi_i | y_i; \hat{\theta})$ , for  $i = 1, 2, \dots, N$  using a Metropolis Hastings (MH) algorithm and estimate the conditional mean

$$\begin{aligned}\hat{\psi}_i^{\text{mean}} &= \mathbb{E}(\widehat{\psi_i | y_i; \hat{\theta}}) \\ &= \frac{1}{K} \sum_{k=1}^K \psi_i^{(k)}.\end{aligned}$$

# Estimation of the individual parameters

## Conditional mode and conditional distribution

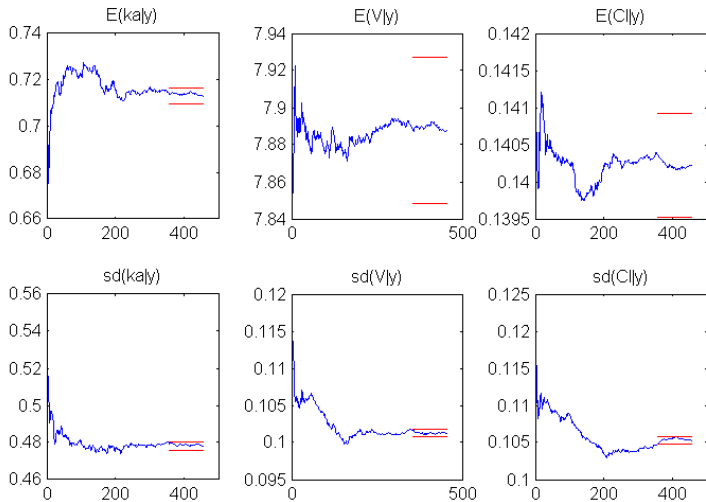
The screenshot displays the Monolix 4.3.3 interface for the project 'warfarinPK\_M0.mlxtran'. The interface is organized into several sections:

- The data and model:**
  - The data: warfarin\_data.bt
  - The structural model: oral1\_1cpt\_kaVCI
  - The covariate model: wt, with values 0 | 0 | 0
  - Distribution of the individual parameters: Diagonal (selected) / Full
  - Covariance model:  $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
  - Observation model: name y1, type continuous, pred Cc, error constant, r
- The initialization:**
  - Fixed effects: 1 | 1 | 1
  - Stand. dev. of the random effects: 1 | 1 | 1
  - Residual error parameters: 1 | 0 | 1 | 0
  - Buttons: Check initial fixed effects, Use the last estimates
- The algorithms:**
  - New seed: 123456
  - Numbers of iterations: K1 500 (auto), K2 200 (auto)
  - Number of chains: 2 (auto), Min Size 50
  - Simulated Annealing: checked
  - Monte-Carlo sizes: Pred. dist. 100, NPDE/VPC 500, LL 20000
  - Display: 50
- The results:**
  - Results folder: Project name warfarinPK\_M0
  - Standard errors: Linearization (checked), Stoch. Approx.
  - Individual parameters: Conditional modes (checked), Cond. means and s.d. (checked)
  - Log-likelihood: Linearization (checked), Importance Sampling
  - Graphics: List



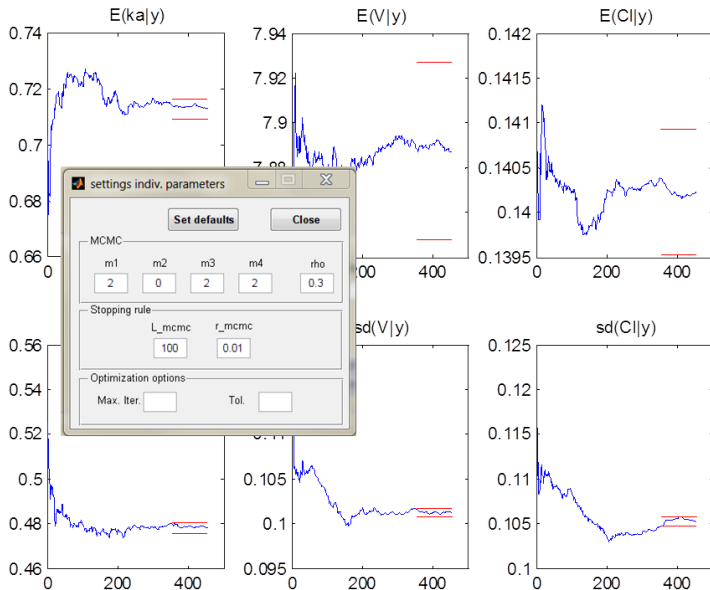
# The Metropolis Hastings algorithm

Transitions kernels and stopping rule



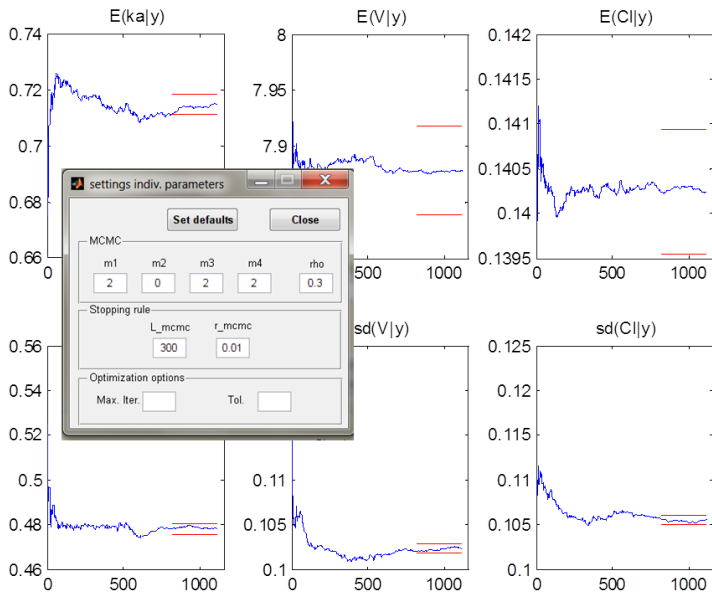
# The Metropolis Hastings algorithm

## Transitions kernels and stopping rule



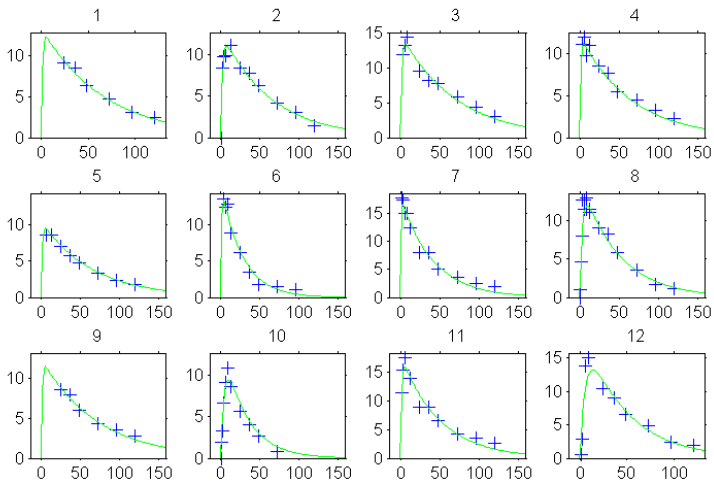
# The Metropolis Hastings algorithm

Transitions kernels and stopping rule



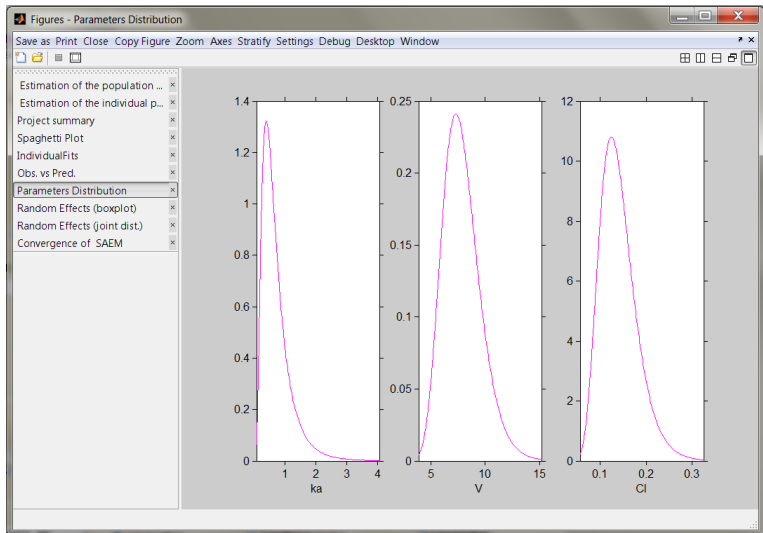
# Some diagnostic plots

## Individual fits



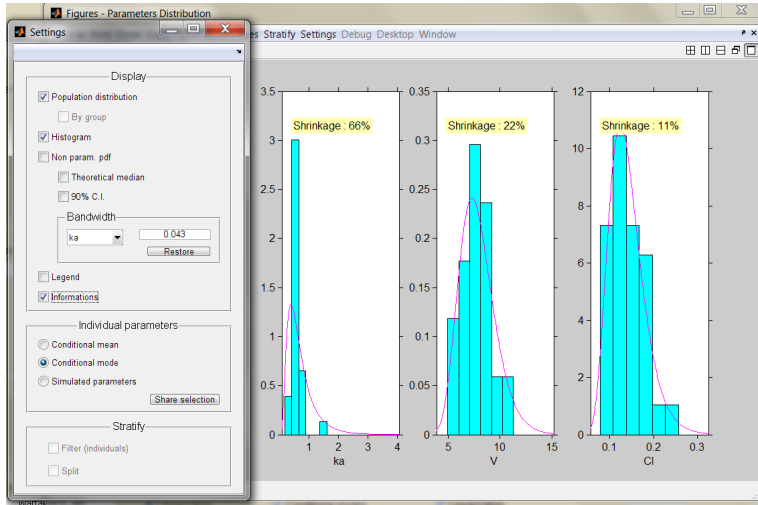
# Some diagnostic plots

## Distribution of the individual parameters



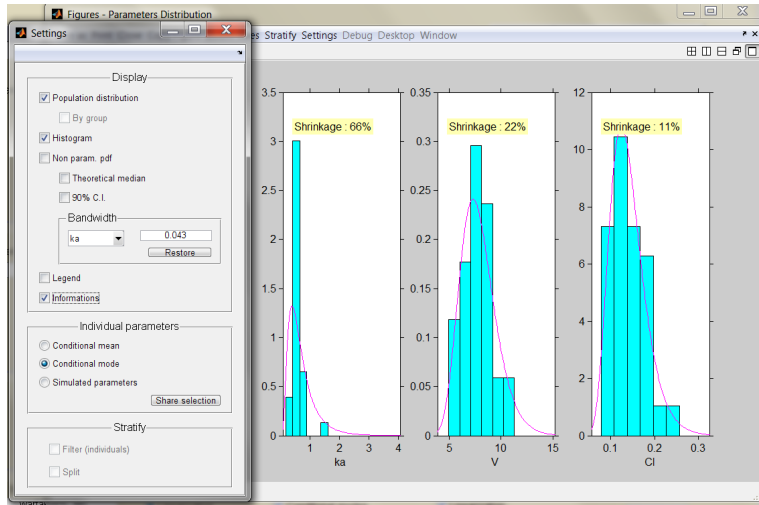
# Some diagnostic plots

## Distribution of the individual parameters



# Some diagnostic plots

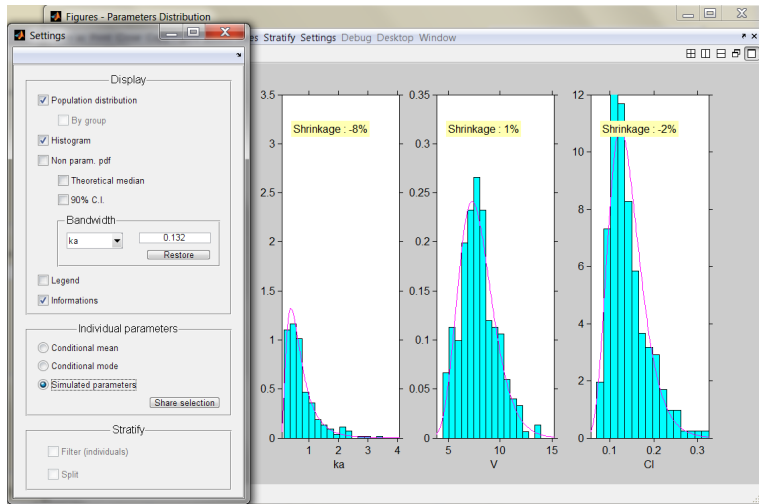
## Distribution of the individual parameters



⇒ Don't use the MAP/conditional mode/posthoc/EBEs for diagnostic!

# Some diagnostic plots

## Distribution of the individual parameters



⇒ Use individual parameters simulated with the conditional distribution  $p(\psi_i | y_i, \hat{\theta})!$



# Some diagnostic plots

Using simulated individual parameters

Indeed, whenever data becomes sparse or uninformative, the EBE distribution will shrink towards the population value.

A particularly effective solution is to draw individual parameters  $(\psi_i^{(k)}, 1 \leq k \leq K)$  with the conditional distribution  $p(\psi_i|y_i, \hat{\theta})$  rather than taking the mode.

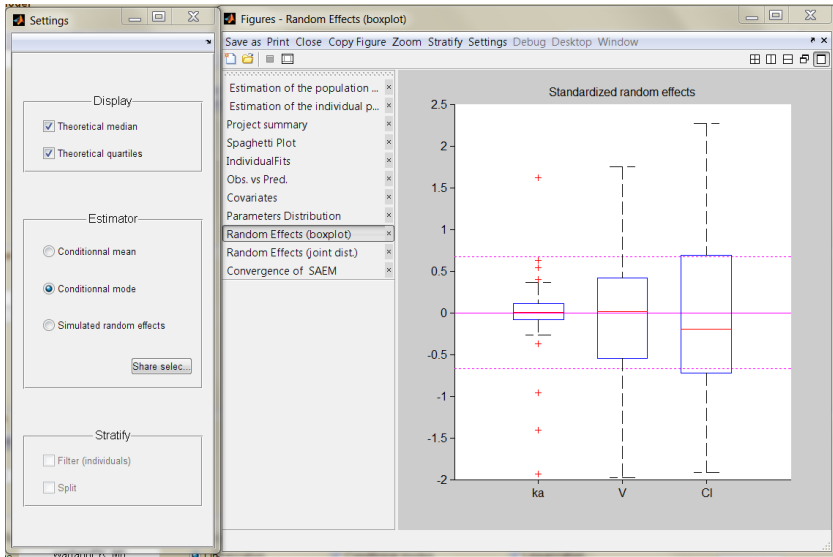
The resulting estimator is unbiased under  $H_0$  in the following sense:

$$\begin{aligned} p(\psi_i) &= \int p(\psi_i|y_i)p(y_i)d y_i \\ &= \mathbb{E}_{y_i}(p(\psi_i|y_i)). \end{aligned}$$

This relationship is a fundamental one when considering inverse problems, incomplete data models, mixed effects models, etc.

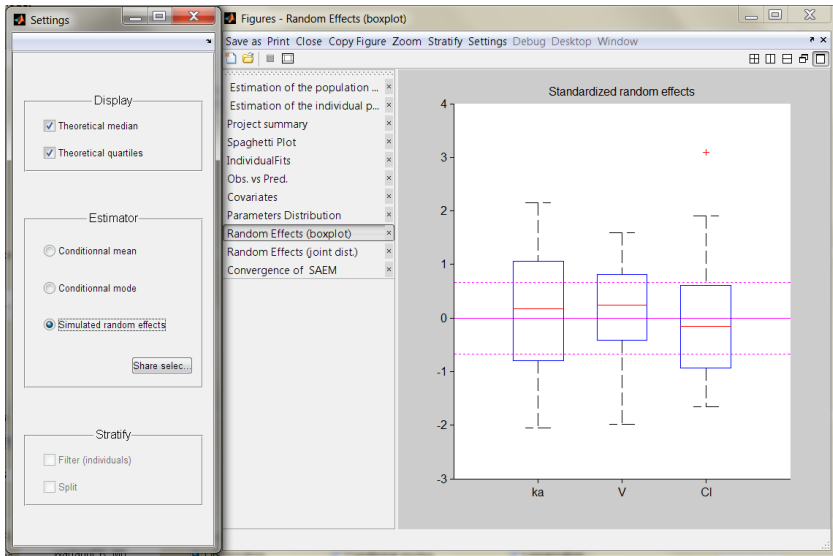
# Diagnostic plots

## Marginal distributions of the random effects



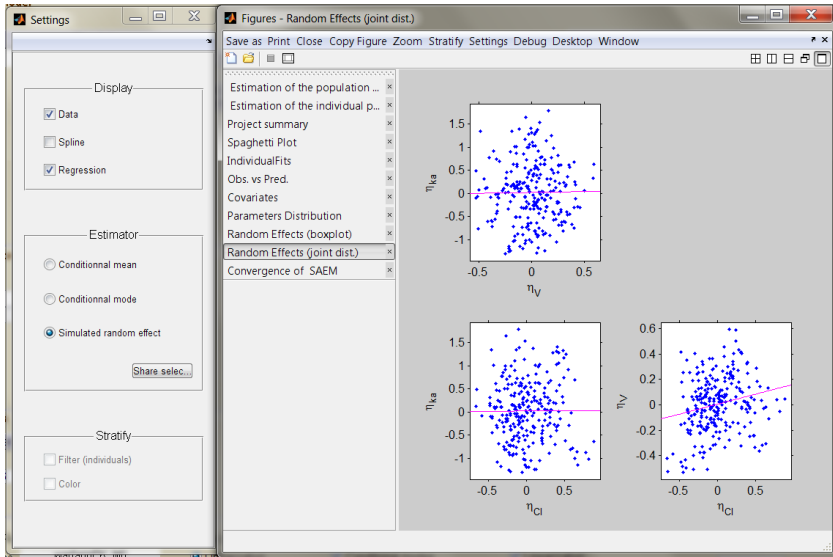
# Diagnostic plots

## Marginal distributions of the random effects



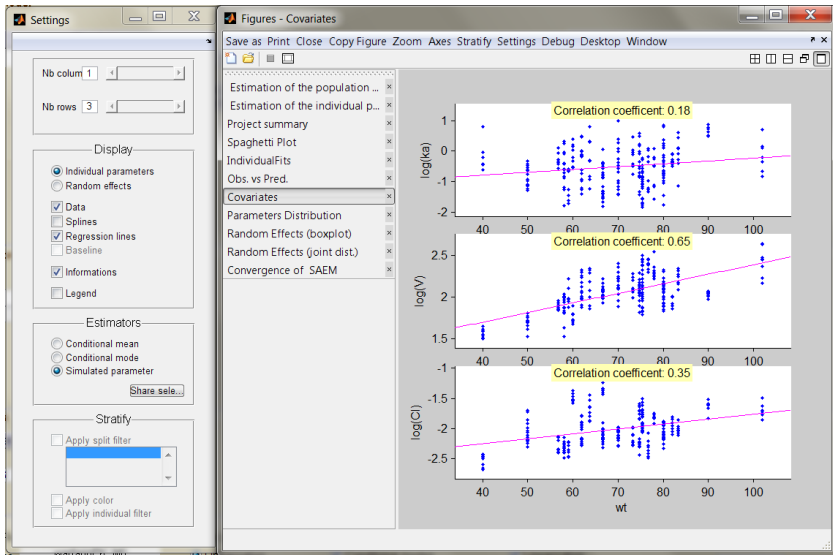
# Diagnostic plots

## Joint distributions of the random effects



# Diagnostic plots

## Relationship between parameters and covariates



## IV

Mlxtran: a powerful  
language for complex  
hierarchical models

# A hierarchical model

- The PK model:

$$C(t, \psi) = \frac{D k_a}{V k_a - Cl} \left( e^{-(Cl/V)t} - e^{-k_a t} \right).$$

- The model for the observed concentrations:

$$y_{ij} = C(t_{ij}, \psi_i) + a\varepsilon_{ij}$$

i.e.

$$y_{ij} | \psi_i, a \sim \mathcal{N}(C(t_{ij}, \phi_i), a^2)$$

- The model for the individual PK parameters  $\psi_i = (k_{a_i}, V_i, Cl_i)$ :

$$\log(k_{a_i}) \sim \mathcal{N}(\log(k_{a_{\text{pop}}}), \omega_{k_a}^2)$$

$$\log(V_i) \sim \mathcal{N}(\log(V_{\text{pop}}), \omega_V^2)$$

$$\log(Cl_i) \sim \mathcal{N}(\log(Cl_{\text{pop}}), \omega_{Cl}^2).$$

- The distribution for  $k_{a_{\text{pop}}}$ :  $\log(k_{a_{\text{pop}}}) \sim \mathcal{N}(\log(k_{a^*}), \gamma^2)$

# Mlxtran code for hierarchical model

```
[LONGITUDINAL]
input = {ka, V, Cl, a}
EQUATION:
C = pkmodel(ka, V, Cl)
DEFINITION:
y = {distribution = normal, prediction = C, sd = a}
;-----

[INDIVIDUAL]
input = {ka_pop, omega_ka, V_pop, omega_V, Cl_pop, omega_Cl}
DEFINITION:
ka = {distribution = lognormal, mean = log(ka_pop), sd = omega_ka}
V = {distribution = lognormal, mean = log(V_pop), sd = omega_V}
Cl = {distribution = lognormal, mean = log(Cl_pop), sd = omega_Cl}
;-----

[POPULATION]
input = {ka_star, gamma_ka}
DEFINITION:
ka_pop = {distribution = lognormal, mean = log(ka_star), sd = gamma_ka}
```



# R code using Simulx

```
p <- c(ka_star=0.5, gamma_ka=0.1, omega_ka=0.2,
      V_pop=10,    omega_V=0.2,
      Cl_pop=1,    omega_Cl=0.15,
      a=0.5)

f <- list(name='C', time=seq(0, 30, by=0.1))
obs <- list(name='y', time=seq(1, 30, by=3))
ind <- list(name=c('ka','V', 'Cl'))
pop <- list(name='ka_pop')

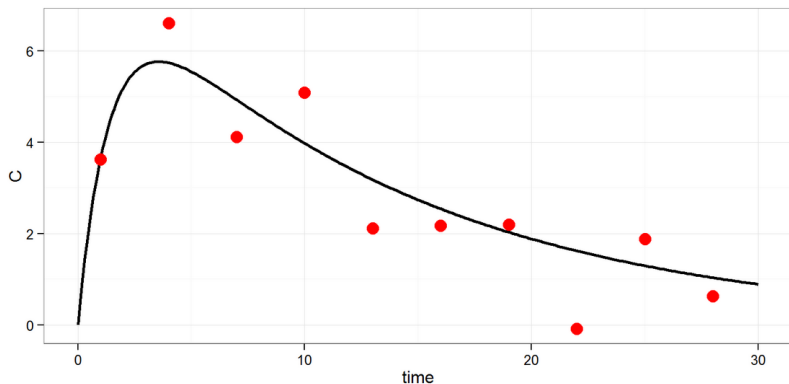
res <- simulx(model      = 'mlxtran1.txt',
              parameter = p,
              treatment = list(time=0, amount=100),
              output     = list(pop, ind, f, obs))

print(res$parameter)
```

```
##      ka_pop      ka      V      Cl
## 1 0.6039348 0.7340071 10.46294 1.327486
```

# R code using Simulx

```
print(ggplotmlx() + geom_line(data=res$C, aes(x=time, y=C), size=1) +  
      geom_point(data=res$y, aes(x=time, y=y), colour='red', size=4))
```



# R code using Simulx

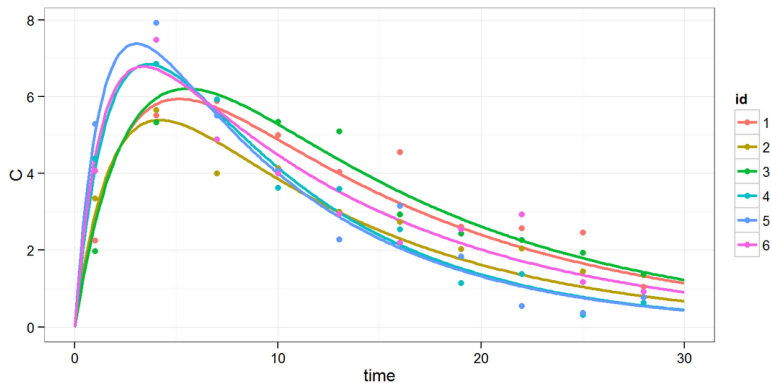
```
res <- simulx(model      = 'mlxtran1.txt',
              parameter = p,
              treatment = list(time=0, amount=100),
              output    = list(pop, ind, f, obs),
              group     = list(size=c(2,3), level=c('population','individual')))

print(res$parameter)
```

```
##   id   ka_pop      ka      V      Cl
## 1  1 0.4505751 0.4021775 11.465519 0.8554521
## 2  2 0.4505751 0.5028199 12.809041 1.1290427
## 3  3 0.4505751 0.3519262 10.550282 0.8050163
## 4  4 0.5421385 0.5509877  9.718898 1.0911209
## 5  5 0.5421385 0.7338368  9.671072 1.0744846
## 6  6 0.5421385 0.7349852 11.236472 0.8981349
```

# R code using Simulx

```
print(ggplotmlx() +  
  geom_line( data=res$C, aes(x=time, y=C, colour=id), size=1) +  
  geom_point(data=res$y, aes(x=time, y=y, colour=id), size=2))
```



# Simulx

A R function of the mlxR package  
for computing predictions and sampling longitudinal data  
from Mlxtran and PharmML models.

[User Guide](#) [Videos](#) [Case studies](#) [Simulx & Shiny](#) [mlxR](#) [Notes](#) [Installation](#)



## User Guide

Learn how to use `simulx` with many illustrative examples



## Case Studies

Discover several examples of practical use of `simulx`, including clinical trial simulation, modelling and simulation workflow...



## Shiny

See how to combine `simulx` with Shiny and produce web applications and training material