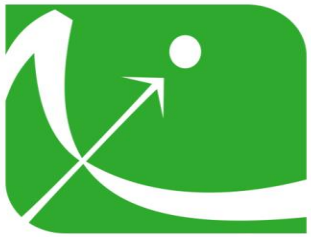
# MONOLIXSUITE 2019
## 1 MODELING LANGUAGE
## 5 APPLICATIONS

# Agenda

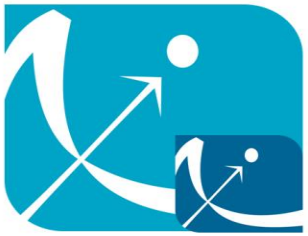
1. Introduction to MonolixSuite 2019
2. Modeling and simulation workflow on a PK/PD example
   - Data visualization
   - Step by step modeling
   - Simulations
3. Conclusions

# MonolixSuite 2019



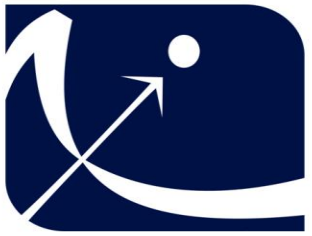
**PKanalix**

NCA and CA analysis



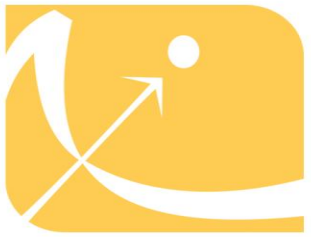
**Monolix**

Estimation, diagnosis, and run management



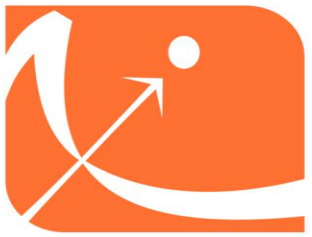
**Mlxplore**

Model exploration



**Datxplore**

Data exploration



**Simulx**

Trial simulation

# PKPD Remifentanil - Intro

## Remifentanil:

- potent, and short-acting opioid analgesic drug
- used for sedation and to relieve pain during surgery, in combination to an anaesthetic
- depth of sedation is recorded through electroencephalography

## Dataset:

- 65 healthy adults
- constant infusion rate between 1 and 8 $\mu g.kg^{-1}.min^{-1}$ for 4 to 20 minutes
- PK: dense concentration measurements during infusion and after
- PD: dense electroencephalogram (EEG) measurements

# DATA SET FORMAT

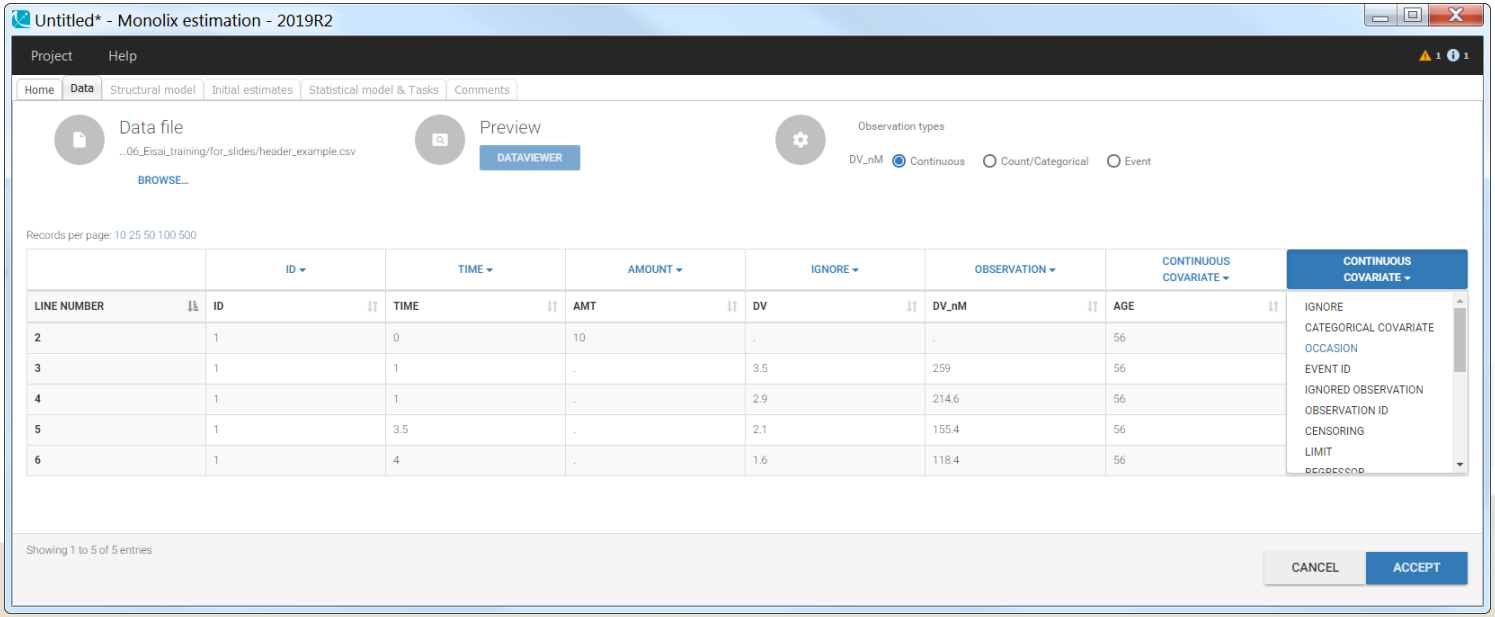
General formatting rules

# Headers

- Headers are free and can be of any length (but avoid special characters)

| ID | TIME | AMT | DV | DV_nM | AGE | WT |
|---|---|---|---|---|---|---|
| 1 | 0 | 10 | . | . | 56 | 87 |
| 1 | 1 | . | 3.5 | 259 | 56 | 87 |
| 1 | 1 | . | 2.9 | 214.6 | 56 | 87 |
| 1 | 3.5 | . | 2.1 | 155.4 | 56 | 87 |
| 1 | 4 | . | 1.6 | 118.4 | 56 | 87 |

- When loading the data set, the columns are assigned to column-types to be interpreted correctly.



# OBS ID and ADM ID columns

## NONMEM

- CMT column used for both administration and observation
- content of CMT column depends on the model

| ID | TIME | AMT | DV | EVID | CMT |
|---|---|---|---|---|---|
| 1 | 0 | 10 | 0 | 1 | 1 |
| 1 | 1 | 0 | 3.5 | 0 | 2 |
| 1 | 1 | 0 | 501.1 | 0 | 3 |
| 1 | 3.5 | 0 | 2.1 | 0 | 2 |
| 1 | 4 | 0 | 489.3 | 0 | 3 |

## MONOLIX

- OBS ID column for observations. Can be integer or string.
- ADM ID column for doses. Must be an integer.

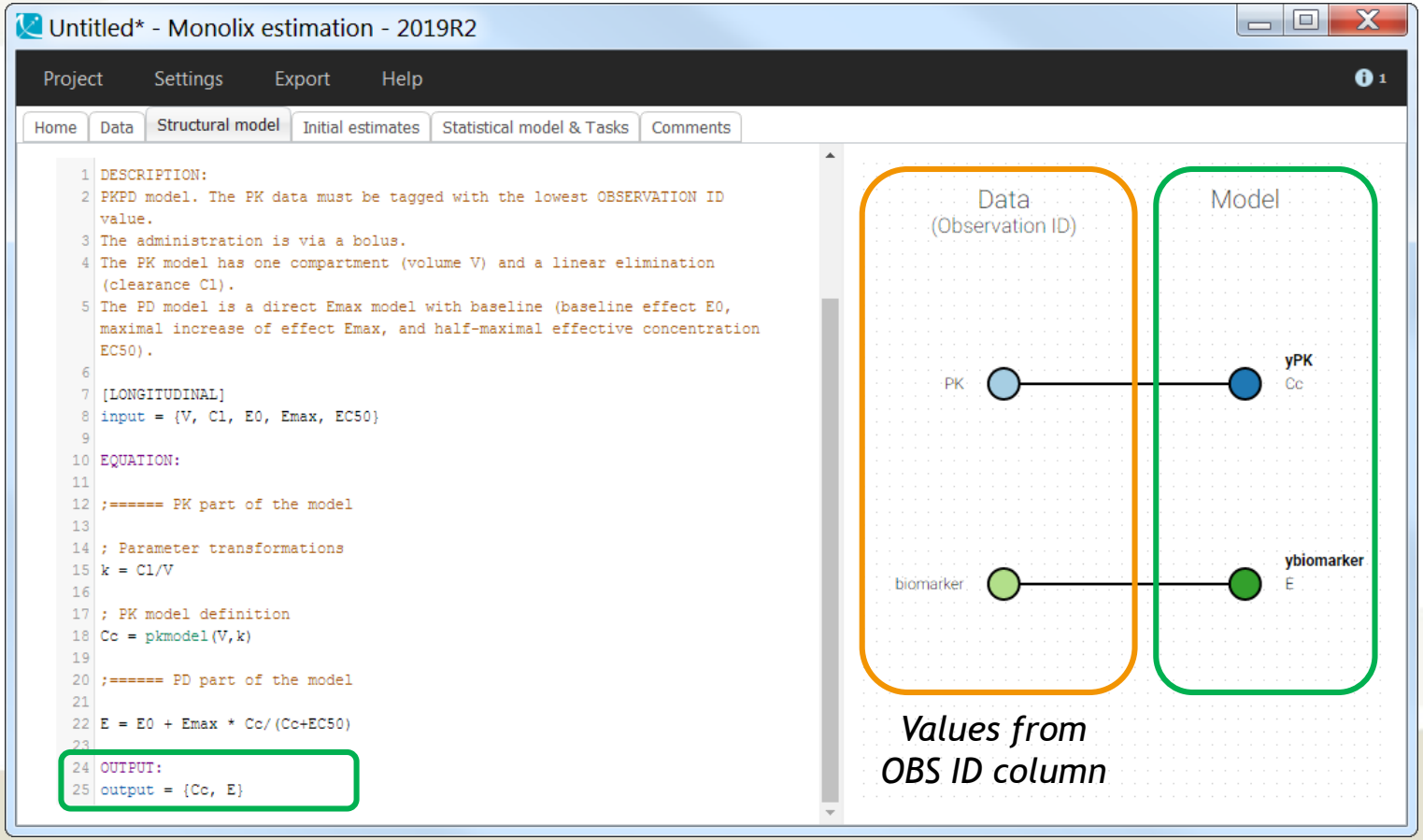
*Tagged as OBSERVATION ID*

| ID | TIME | AMT | DV | ADM | DVID |
|---|---|---|---|---|---|
| 1 | 0 | 10 | . | 1 | . |
| 1 | 1 | . | 3.5 | . | PK |
| 1 | 1 | . | 501.1 | . | biomarker |
| 1 | 3.5 | . | 2.1 | . | PK |
| 1 | 4 | . | 489.3 | . | biomarker |

*Tagged as ADMINISTRATION ID*

# OBSERVATION ID column

- The mapping between the OBS IDs and the model outputs in done in the interface.
- It is possible to have more OBS IDs than model outputs or opposite and leave some unused.



# ADMINISTRATION ID column

- The ADM IDs are used in the administration macros.

```
PK:
oral(adm=1, cmt=1, ka)
iv(adm=2, cmt=1)
```

*doses with ADM ID=1 in the data set are applied with first-order absorption*

*doses with ADM ID=2 are applied as iv boluses*

# Censored data

## NONMEM

- LOQ column
- if DV < LOQ, the data is handled in a special way:

- M1 = Ignore missing values 'MDV'
- M2 = Likelihood assumes all values are censored at LLOQ 'YLO'
- M3 = Estimate likelihood at times measurements are BLQ
- M4 = Like M3 but also assume measurements are >=0
- M5 = Replace *all* BLQ with LLOQ/2
- M6 = Replace *first* BLQ with LLOQ/2, ignore others
- M7 = Replace *all* BLQ with zero

## MONOLIX

- CENSORED column to mark censored data (CENS=1)
- LOQ value is put in OBSERVATION column
- LIMIT column to define the bound in the other direction (usually 0)

| Y | CENS | LIMIT |
|---|---|---|
| 3.5 | 0 | 0 |
| 1 | 1 | 0 |

$p(y = 3.5)$

$p(0 < y < 1)$

# Covariates

LIXOFT

## NONMEM

- Covariates values can change over time
- Missing values can be handled in the model using if/then/else

## MONOLIX

- Covariates must be constant within each individual, or within each occasion of each individual => if varying value really required, use regressors
- missing values are not allowed => infer them in the data set
- Strings allowed for categorical covariates (but avoid special characters)

# Covariates for visualization

## NONMEM

## MONOLIX

- Columns tagged as continuous or categorical covariates can be used to stratify/split the plots
  => prepare in the data set columns for future stratification (e.g DOSE)

# Amount column

## NONMEM

- A dose line can be put in only one compartment (CMT column)
  => for parallel absorptions, two dose lines are required

| ID | TIME | AMT | CMT |
|---|---|---|---|
| 1 | 0 | 100 | 1 |
| 1 | 0 | 100 | 2 |

## MONOLIX

- A dose line can be matched to several absorption macros
  => only 1 dose line required

| ID | TIME | AMT | ADM |
|---|---|---|---|
| 1 | 0 | 100 | 1 |

```
PK:
oral(adm=1,cmt=1,ka=ka1,Tlag,p=F)
oral(adm=1,cmt=1,ka=ka2,p=1-F)
```

*amount 100 split in fraction F and 1-F*

# Ignoring lines

## NONMEM

- C column and/or MDV column

## MONOLIX

- Column IGNORED OBSERVATION. Multiple IGNORED OBSERVATION columns are allowed and can be used to ignore observations for different reasons
- Column IGNORED LINE.

# Steady-state

## NONMEM

- When ADVAN model used, analytical steady-state solution is used

## MONOLIX

- additional doses are added before the SS dose
- the number of additional doses can be chosen by the user
- The additional doses can be at negative times
  => if an ODE system is used, t0 must be chosen to allow simulation of the additional doses

# Occasions

## NONMEM

- OCC column in data set and IOV handled directly in the model

```
OCC2 = 1 - OCC
CL = THETA(1) * EXP(ETA(1) + ETA(2) * OCC + ETA(3)*OCC2)

$OMEGA 1.2
$OMEGA BLOCK(1) 0.5
$OMEGA BLOCK(1) SAME
```

## MONOLIX

- OCC column defines different occasions (periods of time) but not necessarily IOV
- index of occasion doesn't matter
- time can restart from zero or not
- IOV can be added via the interface (random effects at OCC level)
- EVID=4 creates a washout and a new occasion
- covariates which vary from one occasion to the next one can be added only if the parameter has IOV (or no IIV nor IOV)

# DATXPLORE

Data visualization

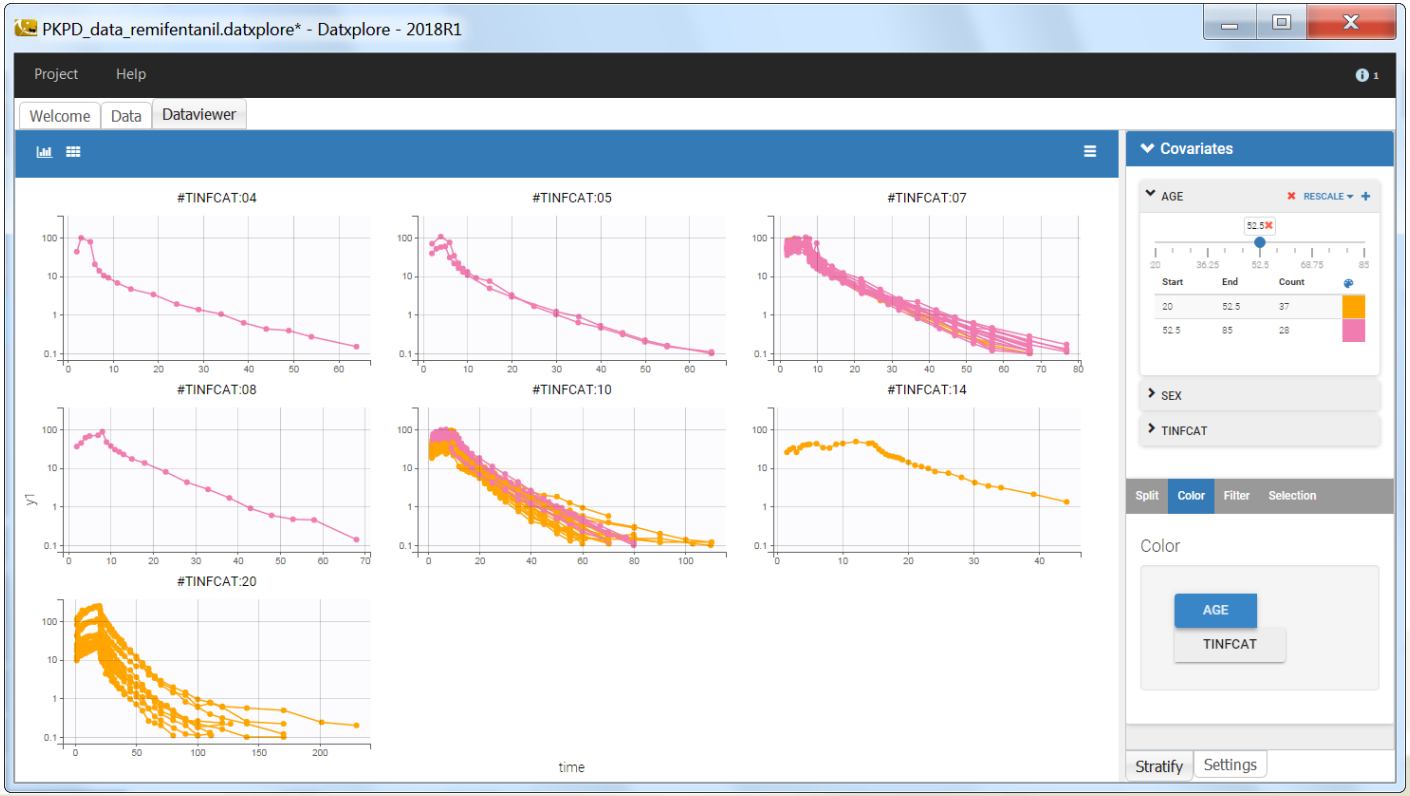
# PKPD Remifentanil - data

| ID | TIME | AMT | RATE | DV | YTYPE | MDV | AGE | SEX | LBM | TINFCAT |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | . | . | 18.85 | 2 | 0 | 30.58 | 1 | 56.5075 | 20 |
| 1 | 0 | 1439.8 | 71.99 | . | . | 1 | 30.58 | 1 | 56.5075 | 20 |
| 1 | 0.5 | . | . | 19.91 | 2 | 0 | 30.58 | 1 | 56.5075 | 20 |
| 1 | 1 | . | . | 19.51 | 2 | 0 | 30.58 | 1 | 56.5075 | 20 |
| 1 | 1.5 | . | . | 18.67 | 2 | 0 | 30.58 | 1 | 56.5075 | 20 |
| 1 | 1.5 | . | . | 9.51 | 1 | 0 | 30.58 | 1 | 56.5075 | 20 |
| 1 | 2 | . | . | 11.5 | 1 | 0 | 30.58 | 1 | 56.5075 | 20 |
| 1 | 2 | . | . | 19.03 | 2 | 0 | 30.58 | 1 | 56.5075 | 20 |

- ID: subject identifier
- TIME: time (minutes)
- AMT: dose amount (µg)
- RATE: infusion rate (µg/min)
- DV: measured Remifentanil concentration (µg/L) or spectral edge frequency (Hz)
- YTYPE: if YTYPE=1, Y is remifentanil. If YTYPE=2, Y is spectral edge frequency
- AGE: age (years)
- SEX: 1=male, 0=female
- LBM: lean body mass (kg)
- TINFCAT: infusion duration (minutes)

# Datxplore: explore your data

- Plot your data (discrete and continuous) to detect outliers, visualize trends, etc
- Split/filter/color to see covariates dependencies, etc



# MONOLIX

Population PK/PD

# Population approach in Monolix

LIXOFT

Data:

- several individuals
- longitudinal data (repeated measurements over time)

$\Rightarrow y_{ij}$ = observation for individual i at time j

Goal:

characterize the typical biological phenomena for the population, but also the inter-individual variability

hierarchical model for both the observations and the individual parameters

# Population approach in Monolix

## Model for the observations:

$$y_{ij} = f(t_{ij}, \varphi_i) + g(\xi)\varepsilon_{ij}$$

- $y_{ij}$: observation for individual i at time j
- $f(t_{ij}, \varphi_i)$: structural model
- $\varphi_i$: parameters of individual i
- $g(\xi)$: residual error model
- $\varepsilon_{ij}$: standardized random variable, $\varepsilon_{ij} \sim \mathcal{N}(0,1)$

## Model for the individual parameters:

$$\varphi_i = \varphi_{pop} + \beta c_i + \eta_i$$

- $\varphi_i$: parameters of individual i
- $\varphi_{pop}$: typical parameters in population
- $c_i$: covariate of individual i
- $\eta_i$: random effect, $\eta_i \sim \mathcal{N}(0, \omega^2)$

# 1. Population approach in Monolix

## Model for the observations:

$$y_{ij} = f(t_{ij}, \varphi_i) + g(\xi)\varepsilon_{ij}$$

- $y_{ij}$: observation for individual i at time j
- $f(t_{ij}, \varphi_i)$: structural model
- $\varphi_i$: parameters of individual i
- $g(\xi)$: residual error model
- $\varepsilon_{ij}$: standardized random variable, $\varepsilon_{ij} \sim \mathcal{N}(0,1)$

## Model for the individual parameters:

$$\varphi_i = \varphi_{pop} + \beta c_i + \eta_i$$

- $\varphi_i$: parameters of individual i
- $\varphi_{pop}$: typical parameters in population
- $c_i$: covariate of individual i
- $\eta_i$: random effect, $\eta_i \sim \mathcal{N}(0, \omega^2)$

# Example

$$y_{ij} = \frac{D}{V_i} e^{-k_i t_{ij}} + \boldsymbol{a}\varepsilon_{ij} \quad \text{with} \quad \varepsilon_{ij} \sim \mathcal{N}(0,1)$$

$$\begin{cases} V_i = \boldsymbol{V_{pop}} \left( \dfrac{\boldsymbol{WT_i}}{70} \right)^{\boldsymbol{\beta_V}} e^{\eta_{V,i}} & \text{with} \quad \eta_{V,i} \sim \mathcal{N}\left(0, \boldsymbol{\omega_V^2}\right) \\ k_i = \boldsymbol{k_{pop}} e^{\eta_{k,i}} & \text{with} \quad \eta_{k,i} \sim \mathcal{N}\left(0, \boldsymbol{\omega_k^2}\right) \end{cases}$$

# Example

$$y_{ij} = \frac{D}{V_i} e^{-k_i t_{ij}} + \boldsymbol{a}\varepsilon_{ij} \qquad \text{with} \quad \varepsilon_{ij} \sim \mathcal{N}(0,1)$$

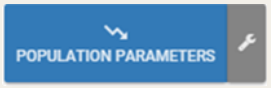
$$\begin{cases} V_i = \boldsymbol{V_{pop}} \left(\dfrac{\boldsymbol{WT_i}}{70}\right)^{\boldsymbol{\beta_V}} e^{\eta_{V,i}} & \text{with} \quad \eta_{V,i} \sim \mathcal{N}\left(0, \boldsymbol{\omega_V^2}\right) \\ k_i = \boldsymbol{k_{pop}} e^{\eta_{k,i}} & \text{with} \quad \eta_{k,i} \sim \mathcal{N}\left(0, \boldsymbol{\omega_k^2}\right) \end{cases}$$

$$\begin{cases} V_i = \boldsymbol{V_{pop}} \left(\dfrac{\boldsymbol{WT_i}}{70}\right)^{\boldsymbol{\beta_V}} e^{\eta_{V,i}} \\ k_i = \boldsymbol{k_{pop}} e^{\eta_{k,i}} \end{cases} \Leftrightarrow \begin{cases} \log(V_i) = \log\left(\boldsymbol{V_{pop}}\right) + \boldsymbol{\beta_V} \log\left(\dfrac{\boldsymbol{WT_i}}{70}\right) + \eta_{V,i} \\ \log(k_i) = \log\left(\boldsymbol{k_{pop}}\right) + \eta_{k,i} \end{cases}$$

# Tasks

- Estimate population parameters $\theta=(V_{pop},k_{pop},\omega_V, \omega_k,\beta_V, a)$ using maximum likelihood
- Estimation of the uncertainty of the population parameters
- Estimate the individual parameters $\varphi_i=(k_i,V_i)$ for each individual
  - The most probable one (EBEs)
  - The conditional distribution
- Simulations using the model

# Tasks in Monolix




Goal:
- Estimate population parameters with SAEM

Results:
- Population parameters
- Approximate individual parameter (mean of conditional distribution)

Usage:
- Graphical report of convergence


Goal:
- Estimate individual parameters (EBEs)

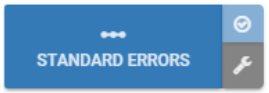
Results:
- Modes of individual conditional distributions (EBEs)

Usage:
- Used for individual fits


Goal:
- Sample from individual conditional distributions with MCMC

Results:
- Samples drawn from distribution
- Mean and standard deviation

Usage:
- Improves the performance of diagnostic plots
- Statistical tests (with plots)




Goal:
- Standard errors for population parameters with linearization or stochastic approximation

Results:
- Standard errors and relative standard errors
- Correlation matrix of the estimates
- Wald test for covariate parameters

Usage:
- Uncertainty of population parameters


Goal:
- Compute log-likelihood with linearization or stochastic approximation

Results:
- -2*LL, AIC, BIC

Usage:
- Compare goodness of fit to other models


Goal:
- Generate diagnostic plots

Results:
- Interactive plots in interface
- Saved images
- Charts data

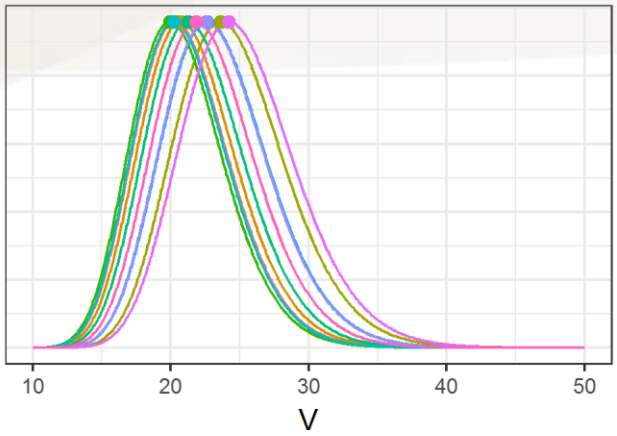
Usage:
- Identification of misspecifications

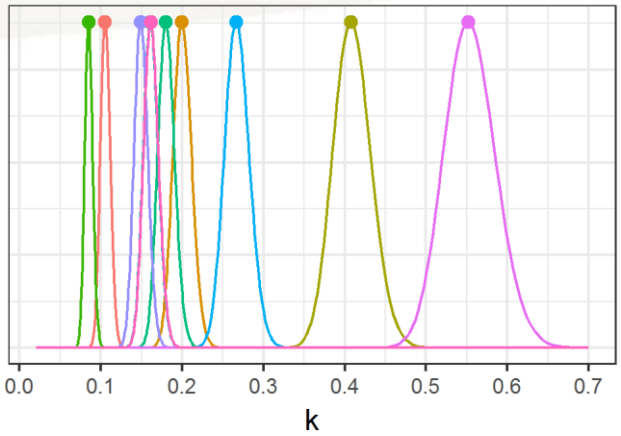
# Shrinkage

**Volume of distribution V**
V has high shrinkage

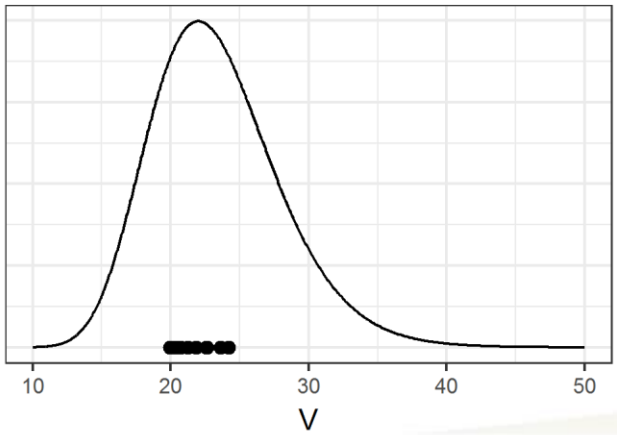
**Elimination rate k**
k has very low shrinkage

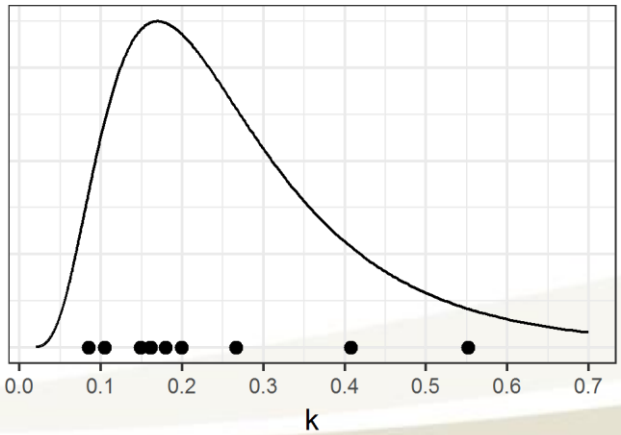
Individual conditional distributions




Population distribution



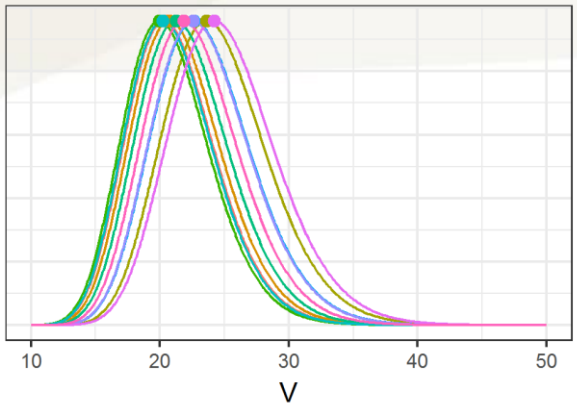



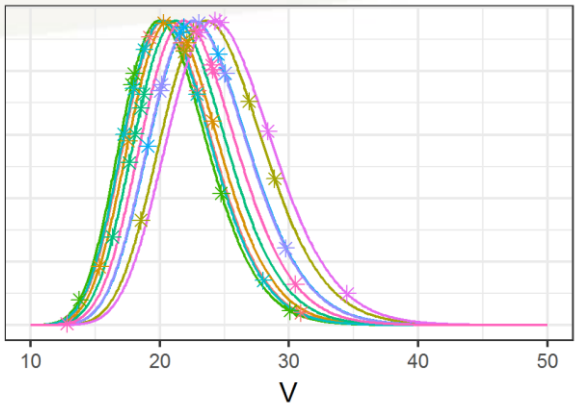
# Circumventing shrinkage

**Using the mode of the conditional distribution**

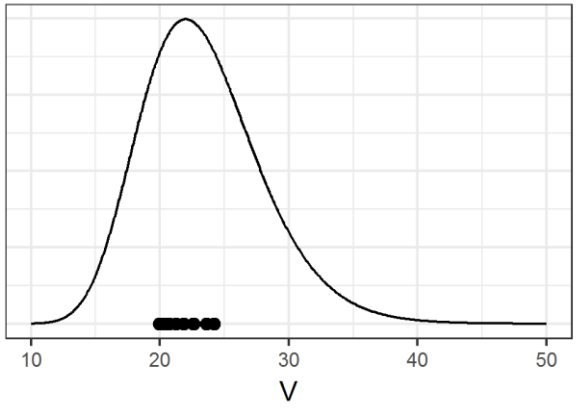
**Using several samples from the conditional distribution**

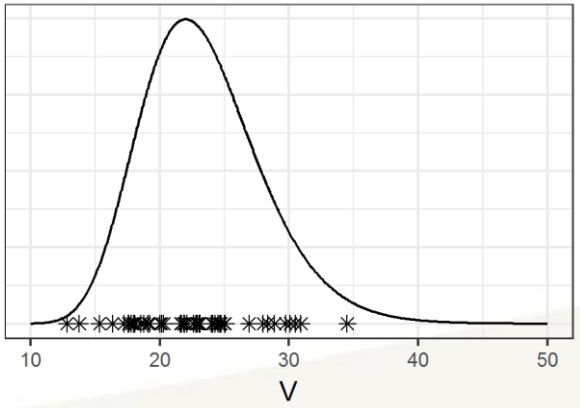
Individual conditional distributions




Population distribution



# Transforming covariates

$$\log(Cl) = \log\left(Cl_{pop}\right) + \beta_{Cl} \times AGE + \eta_{Cl}$$

$$\Longleftrightarrow \quad Cl = Cl_{pop}\ e^{\beta_{Cl} \times AGE}\ e^{\eta_{Cl}}$$

*Exponential relationship*

$$Cl = Cl_{pop} \left(\frac{AGE}{45}\right)^{\beta_{Cl}} e^{\eta_{Cl}}$$

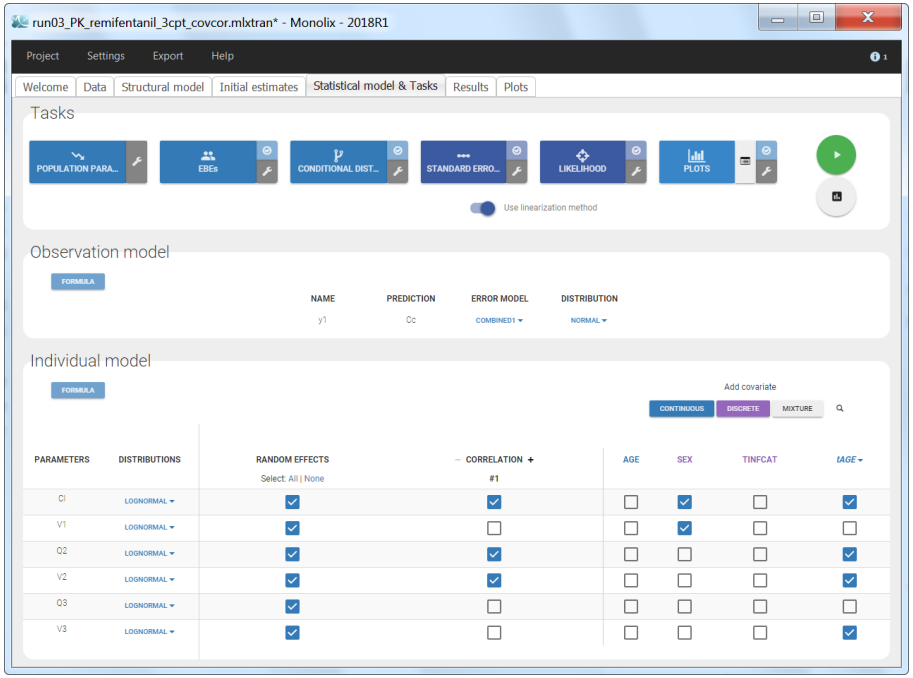
*Power law relationship*

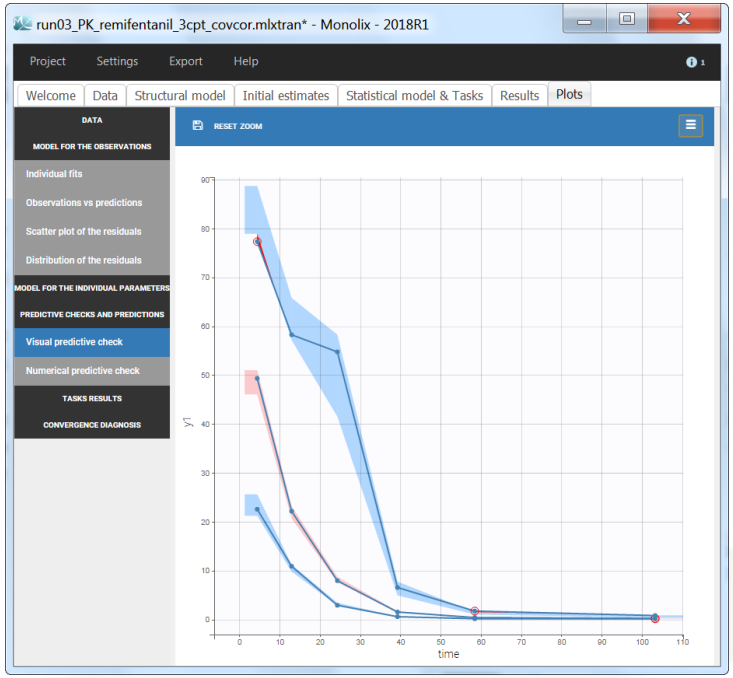
$$\Longleftrightarrow \quad \log(Cl) = \log\left(Cl_{pop}\right) + \beta_{Cl} \times \underbrace{\log\left(\frac{AGE}{45}\right)}_{tAGE} + \eta_{Cl}$$

$$tAGE = \log\left(\frac{AGE}{45}\right)$$

# Monolix: parameter estimation

- User-friendly, fast and robust parameter estimation
- Cutting-edge statistical methods (SAEM)
- Efficient model diagnostic using interactive graphics

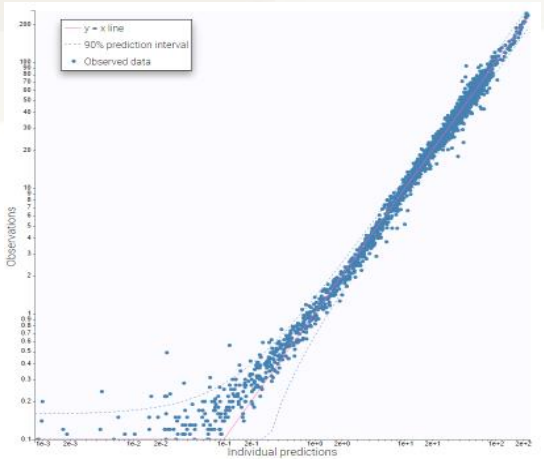


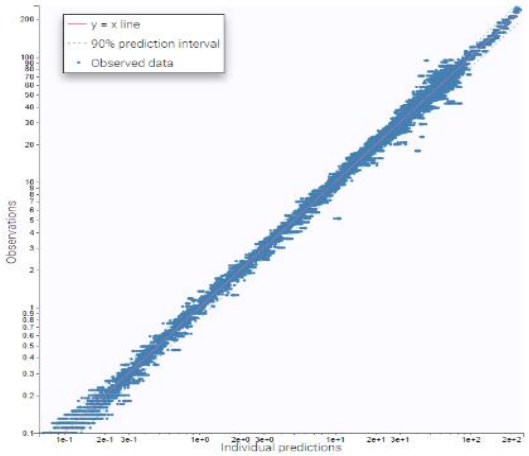
# PK Remifentanil - Monolix

**2 compartment model**
=> low concentrations are not well captured

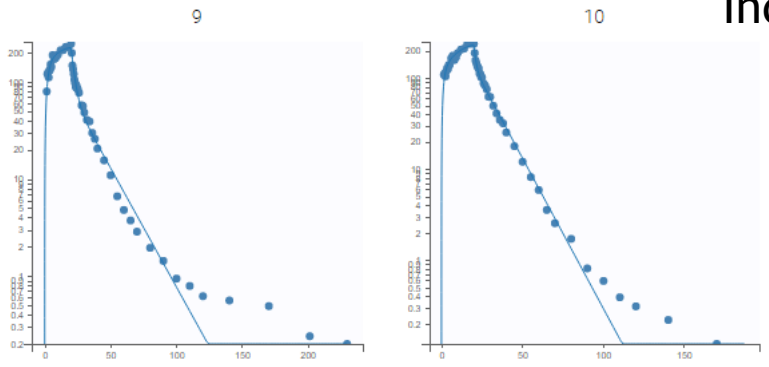
**3 compartment model**
=> No misspecification detected

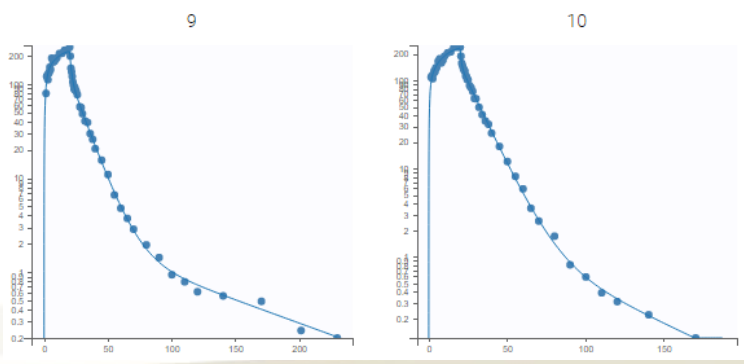



Obs. Versus Pred



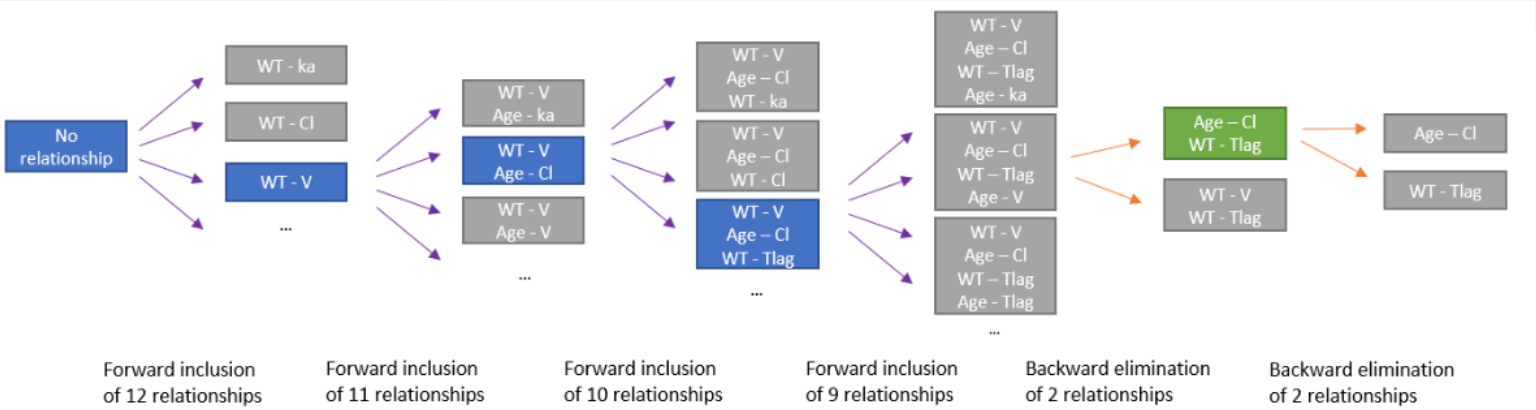

Individual fits



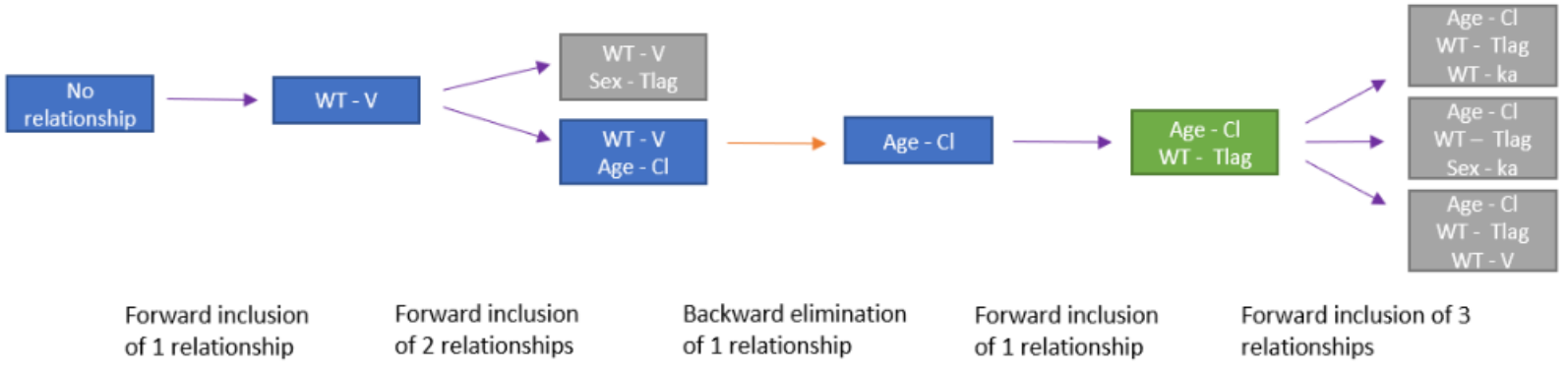



# Automatic covariate search

- SCM: all covariate-parameter relationships are tested at each step

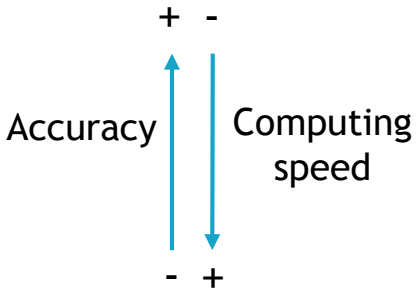



- COSSAC: only the most promising (given the statistical tests) covariate-parameter relationship is tested at each step



# PK-PD modeling approaches

| | Population PK parameters | Individual PK parameters | PD parameters (population + individual) |
|---|---|---|---|
| Joint | **estimated** | **estimated** | **estimated** |
| Intermediate | fixed (from PK analysis) | **estimated** | **estimated** |
| Sequential | - | fixed (from PK analysis) | **estimated** |



# PK-PD model

## Joint approach

PK-PD dataset

PK-PD model


## Intermediate approach

PK-PD dataset

PK-PD model

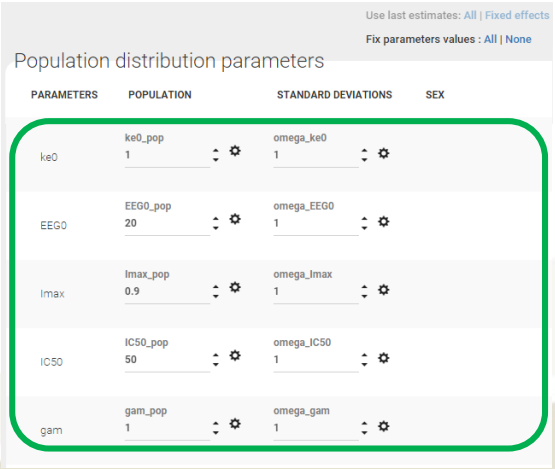

## Sequential approach

PK-PD dataset + **PK parameters**

| ID | TIME | AMT | RATE | DV | YTYPE | MDV | Cl | V1 | Q2 | V2 | Q3 | V3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | . | . | 18.85 | 2 | 0 | 2.85 | 8.02 | 1.29 | 12.22 | 0.14 | 7.09 |
| 1 | 0 | 1439.8 | 71.99 | . | . | 1 | 2.85 | 8.02 | 1.29 | 12.22 | 0.14 | 7.09 |
| 1 | 0.5 | . | . | 19.91 | 2 | 0 | 2.85 | 8.02 | 1.29 | 12.22 | 0.14 | 7.09 |
| 1 | 1 | . | . | 19.51 | 2 | 0 | 2.85 | 8.02 | 1.29 | 12.22 | 0.14 | 7.09 |
| 1 | 1.5 | . | . | 18.67 | 2 | 0 | 2.85 | 8.02 | 1.29 | 12.22 | 0.14 | 7.09 |
| 1 | 1.5 | . | . | 9.51 | 1 | 1 | 2.85 | 8.02 | 1.29 | 12.22 | 0.14 | 7.09 |

PK-PD model + **regressors**

Cl = {use = regressor}
V1 = {use = regressor}
Q2 = {use = regressor}
V2 = {use = regressor}
Q3 = {use = regressor}
V3 = {use = regressor}




PK parameters

PD parameters

# MLXTRAN LANGUAGE

General language rules

# Mlxtran

**Mlxtran**: human-readable language

- Very natural syntax
- Unified language for all applications
- Support of continuous, categorical, count and time-to-event models (and any combination of them)
- Model definition via macros or equations (ODEs, DDEs)
- Extensive documentation (online or PDF)

# Mlxtran - general structure

LIXOFT

```
[LONGITUDINAL]
input={ … }
```
*(mandatory)* — input parameters

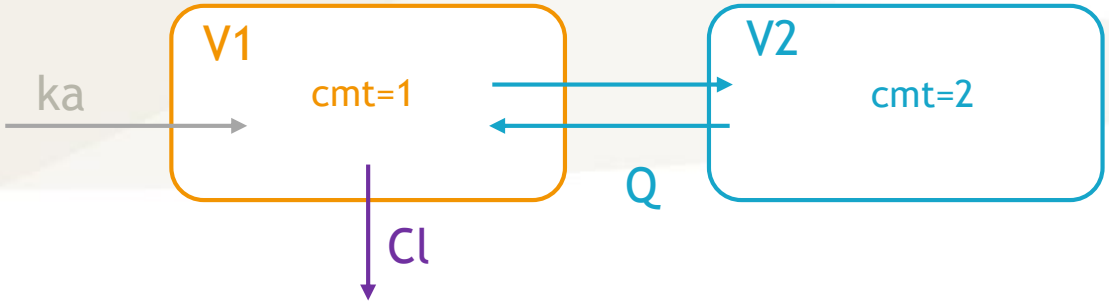
```
PK:
```
*(optional)* — section to define a model via macros *(except the global pkmodel() macro that can be used in EQUATION: too)*

```
EQUATION:
```
*(optional)* — section to define a model via ODEs (and other equations if needed)

```
DEFINITION:
```
*(optional)* — section to define random variables for non-continuous (time-to-event, count or categorical)

```
OUTPUT:
output = { … }
```
variable(s) that will be mapped to the data set observations

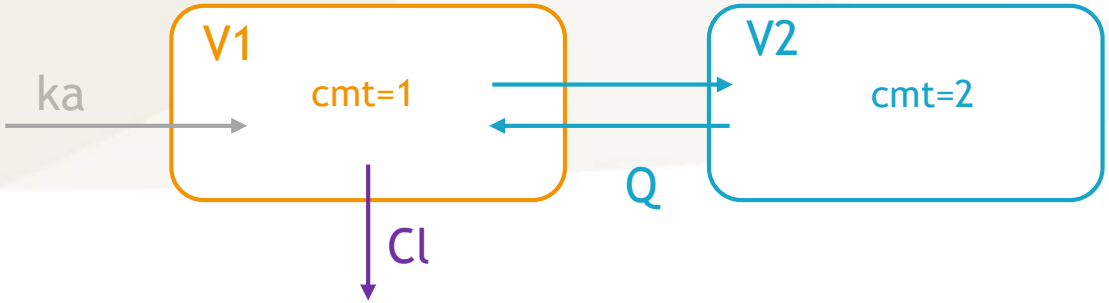
# Example model




This model can be used in Monolix via 4 different ways:

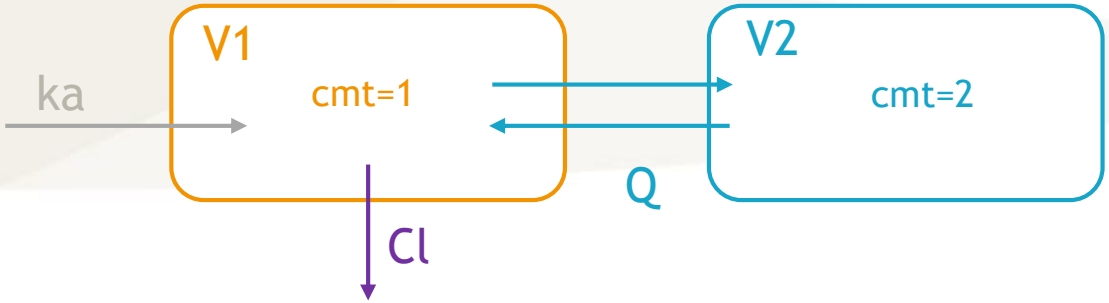
- Select the model from the PK library
- Write a model using the pkmodel() macro
- Write a model using a set a unitary macros
- Write a model using a set of ODEs

# With the pkmodel() macro




```
[LONGITUDINAL]
input = {ka, V1, Cl, Q, V2}

PK:
Cc = pkmodel(ka, V=V1, k=Cl/V1, k12=Q/V1, k21=Q/V2)

OUTPUT:
output = Cc
```

# With a set of macros




```
[LONGITUDINAL]
input = {ka, V1, Cl, Q, V2}

PK:
compartment(cmt=1, volume=V1, concentration=Cc)
oral(cmt=1, ka)
peripheral(k12=Q/V1,k21=Q/V2)
elimination(cmt=1,Cl)

OUTPUT:
output = Cc
```

# Mlxtran - macros

- peripheral(k12,k21) implicitly defines the second compartment
- for a second peripheral compartment, with compartment number 3: peripheral(k13,k31)
- k12 and k21 are recognized keywords. To use other names or other parameterizations:
  peripheral(k12=Q/V, k21=Q/V2)
- iv(adm=1,cmt=1): adm refers to the ADM column of the data set, which permits to distinguish several types of administrations
- in case of several outputs, output={Cc,Effect}, the match is done by order with the YTYPE column of the data set

# Mlxtran - macros

## Administration macros

**depot(**type/adm=1, target=Ad, Tlag, p=F,
Tk0, ← for zero-order absorption
ka, Ktr, Mtt) ← for first-order absorption

**absorption(**type/adm, cmt, Tlag, p,
Tk0, ← for zero-order absorption
ka, Ktr, Mtt) ← for first-order absorption

**iv(**type/adm, cmt, Tlag, p**)**

## Compartment macros

**compartment(**cmt=1,
amount=Ac,
concentration=Cc,
volume=V**)**

**peripheral(**kij, kji,
amount=Ap,
volume=Vp,
concentration=Cp**)**

**effect(**cmt, ke0, concentration=Ce**)**

**transfer(**from=i, to=j, kt**)**

## Elimination macro

**elimination(**cmt, V,
k, Cl , ← for linear elimination
Vm, Km**)** ← for Michaelis-Menten elimination

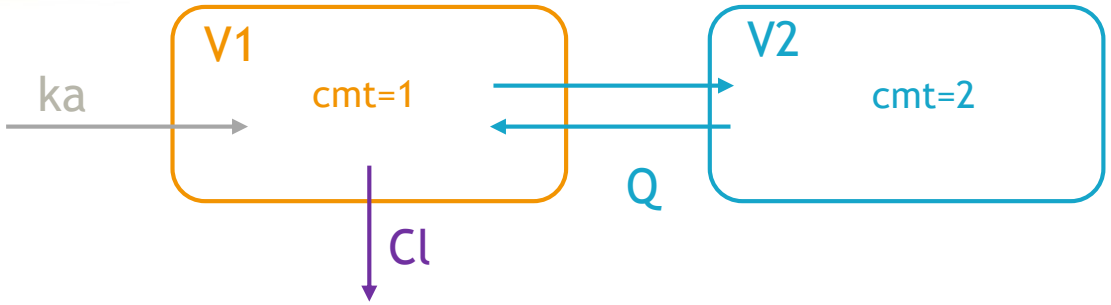
## pkmodel macro

**pkmodel(**Tlag, p,
Tk0, ← zero-order absorption
ka, Ktr, Mtt, ← first-order absorption
k/Cl**,** ← linear elimination
Vm, Km, ← MM elimination
(k12, k21), ← transfer rates
ke0**)** ← effect compartment transfer rate

Doc for PK macros:
http://mlxtran.lixoft.com/pk/
+ Mlxtran cheatsheet

# With a set of ODEs

```
[LONGITUDINAL]
input = {ka, V1, Cl, Q, V2}

PK:
depot(adm=1, target=Ac)
depot(adm=2, target=Ad)

EQUATION:
t_0 = 0
Ac_0 = 0
Ap_0 = 0

k=Cl/V1
k12=Q/V1
k21=Q/V2

ddt_Ad = -ka*Ad
ddt_Ac = -k*Ac - k12*Ac + k21*Ap + ka*Ad
ddt_Ap =         k12*Ac - k21*Ap

Cc = Ac/V1

OUTPUT:
output = Cc
```




Doc for ODEs:
http://mlxtran.lixoft.com/longitudinal/how-do-i-model-an-ode/
+ Mlxtran cheatsheet

# Mlxtran syntax for ODEs

- depot macro to link data set and model.
  - Bolus, zero-order and first-order absorptions can be used.
  - Target must be an amount (not a concentration).
  - For first-order absorption, write `depot(target=Ac,ka)`. No need for a depot compartment.

- define inital time and initial values with "_0", for instance "t_0=" or "Ac_0="

- define ODEs with "ddt_" for instance "ddt_Ac="

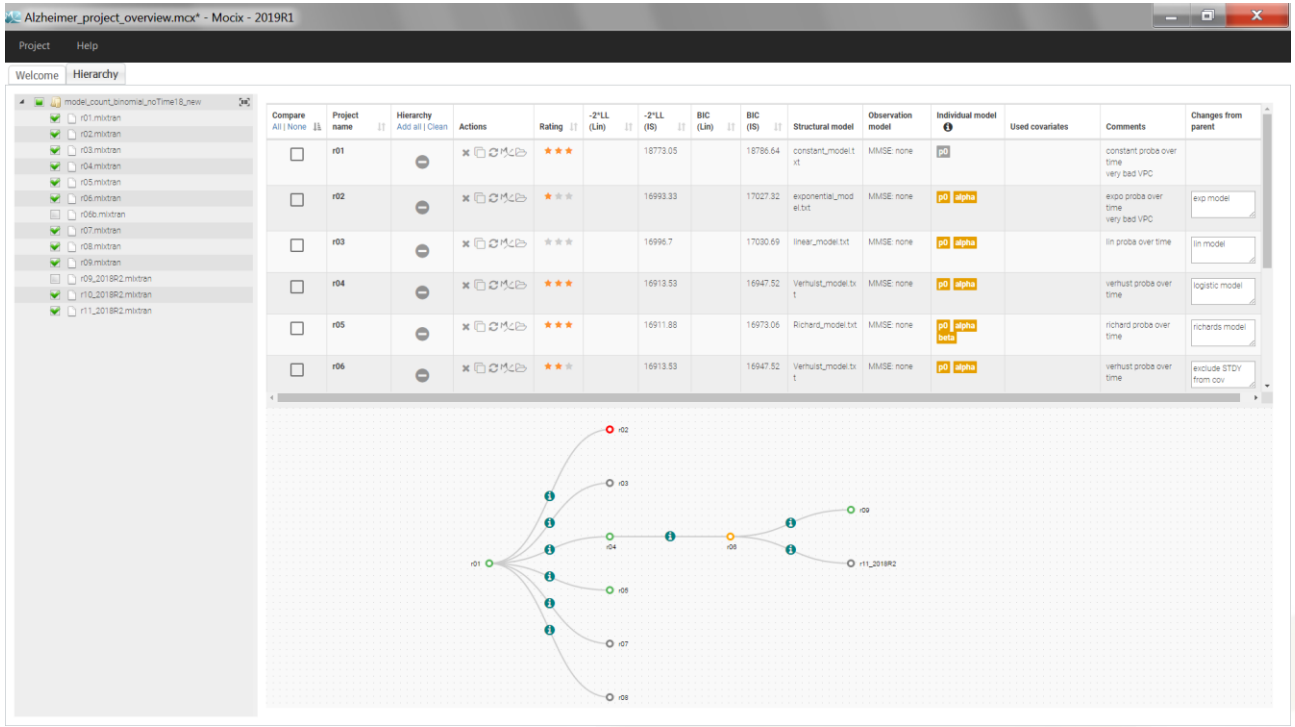
- ODEs in block EQUATION:  (while macros are in block PK:)

# SYCOMORE

Workbench for Monolix

# Sycomore: workbench

- overview of Monolix runs in table and graphical format
- comparison of runs side by side
- run projects in batch mode



# Rsmlx R package

Workbench for Monolix

# Rsmlx: R speaks Monolix

- covariateSearch: SCM or COSSAC method
- bootmlx: case bootstrap (stratification possible)
- confintmlx: confidence intervals using bootstrap, profile likelihood or the Fisher Information matrix

➢ The code is open and commented. It can be easily adapted and modified to your needs.

➢ These functions use functions from the Monolix API (R package called lixoftConnectors)

# MlxConnectors: Monolix API

- functions that allow to create/modify/run a Monolix project from R

- Example:

```
library(lixoftConnectors)
initializeLixoftConnectors(software = "monolix")

loadProject("r02.mlxtran")
popparam <- getEstimatedPopulationParameters()
if(popparam['a']<1e-6){
   setErrorModel(DV = "proportional")
   setPopulationParameterInformation(b=list(initialValue=0.3, method="MLE"))
   saveProject("r02bis.mlxtran")
   runScenario()
}
getEstimatedPopulationParameters()
```

# SIMULX (mlxR package)

Simulations

# Simulx: mlxR package

**Simulx**: an application for advanced simulations

- A powerful and flexible simulator
- Available via an R library
- Integrated with Monolix
- A tool for decisions

# Overview of possibilities

- compare different treatments
- compare the outcome in different populations (which have different covariates, or inclusion/exclusion criteria)
- simulate clinical trials with different number of patients or different designs (crossover, parallel, adaptive designs), do replicates and calculate the power
- reuse covariates from existing data bases (as text files)
- take into account the uncertainty of population parameters
- take into account non-adherence to a treatment
- dose individualization after therapeutic drug monitoring
- generate customized VPCs
- …

*VPC: visual predictive check*

# Simulx: flexibility

**Model defined in Mlxtran**
*From Monolix project*
*Or User defined*

**Covariates**
*From data set*
*Or User defined*

**Nb of subjects**
*From data set*
*Or User defined*

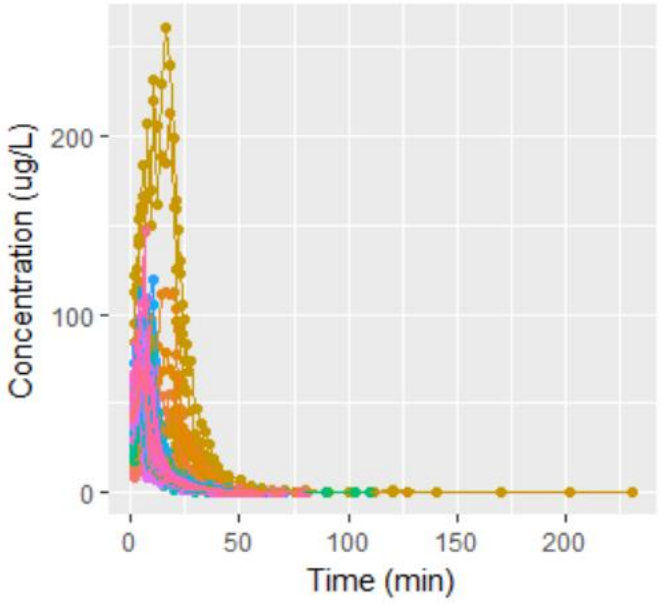
**Pop. parameters**
*From Monolix outputs*
*Or User defined*

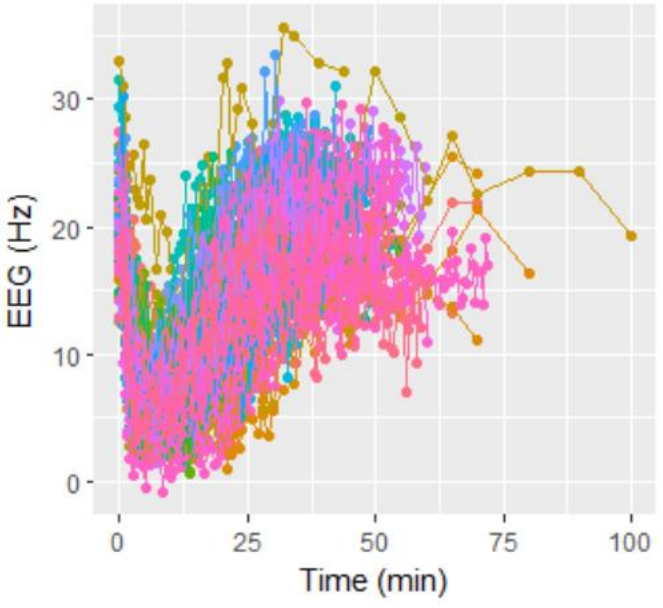
**Outputs**
*From data set*
*Or User defined*

**Treatments**
*From data set*
*Or User defined*

# Simulx: resimulate data set

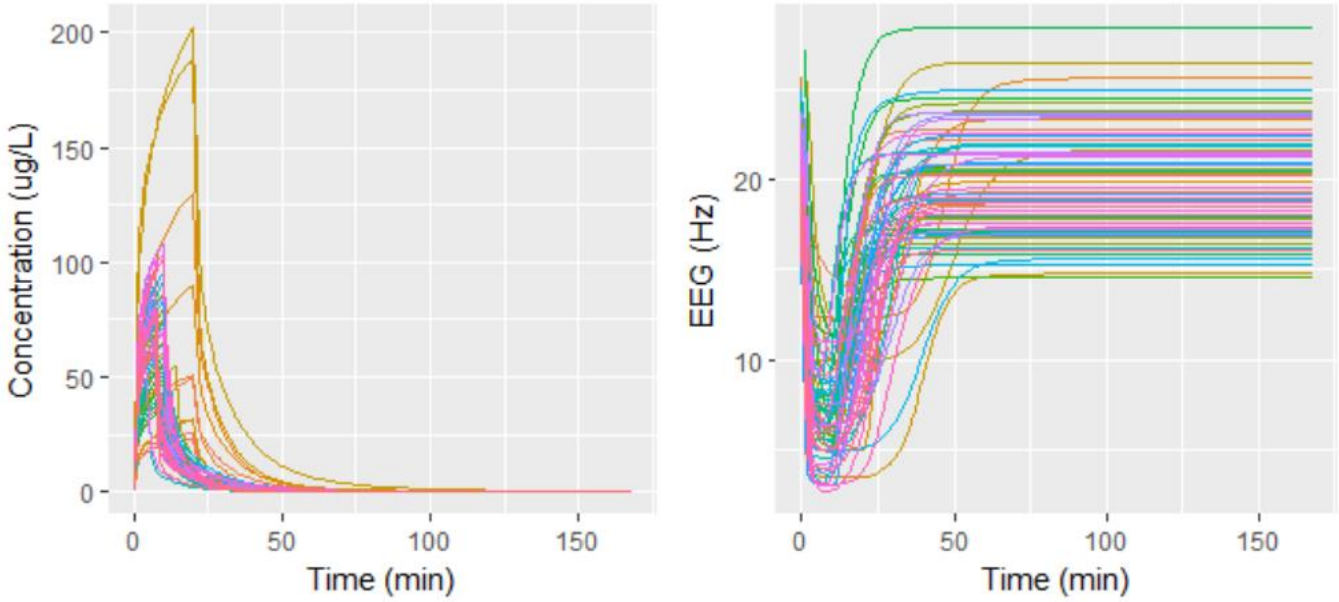
1. Resimulate a data set:
   - same covariates, same design (administration and output)
   - estimated population parameters
   - random individual parameters and residual error



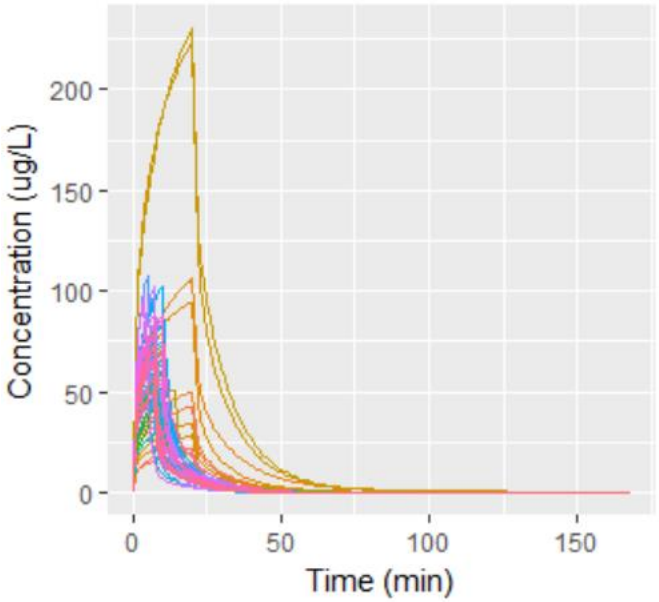
# Simulx: resimulate data set

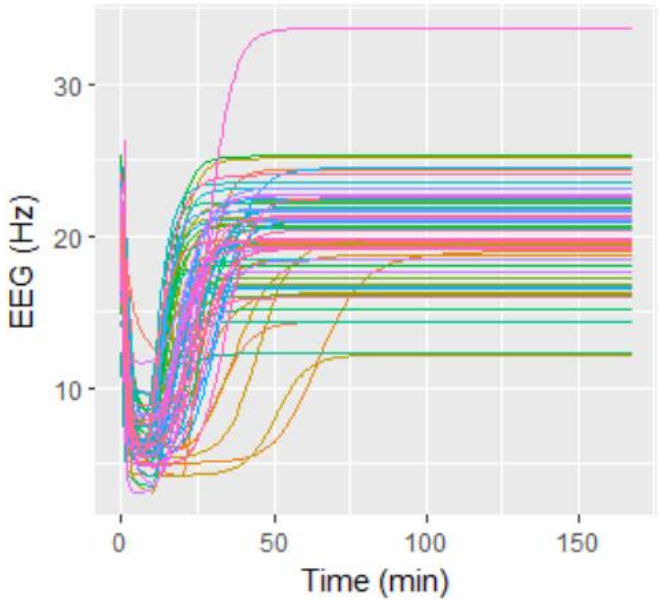
1. Resimulate a data set:
   - output predictions instead of simulated observations



# Simulx: resimulate data set

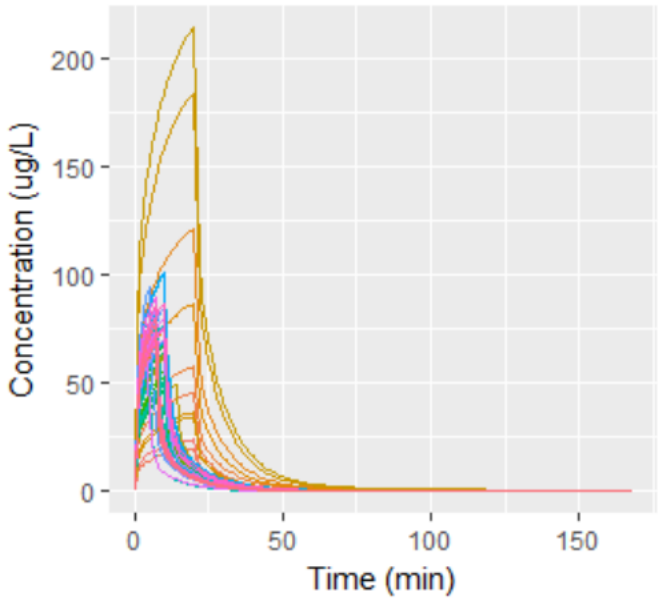
1. Resimulate a data set:
   - simulating with EBEs

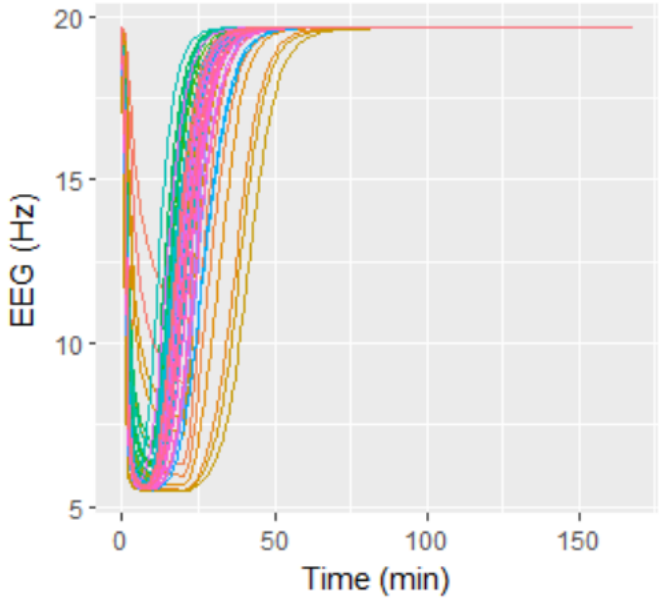


# Simulx: resimulate data set

1. Resimulate a data set:
   - simulating with population parameters

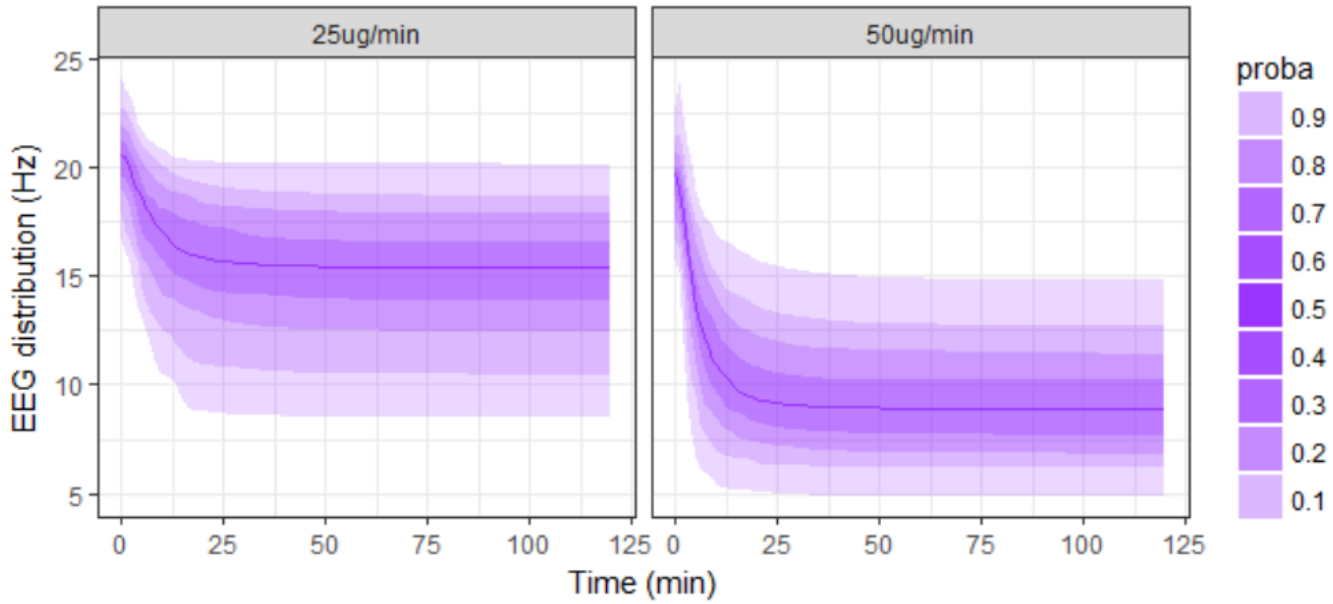


# Simulx: new treatment

## 2. Simulate two different treatments:

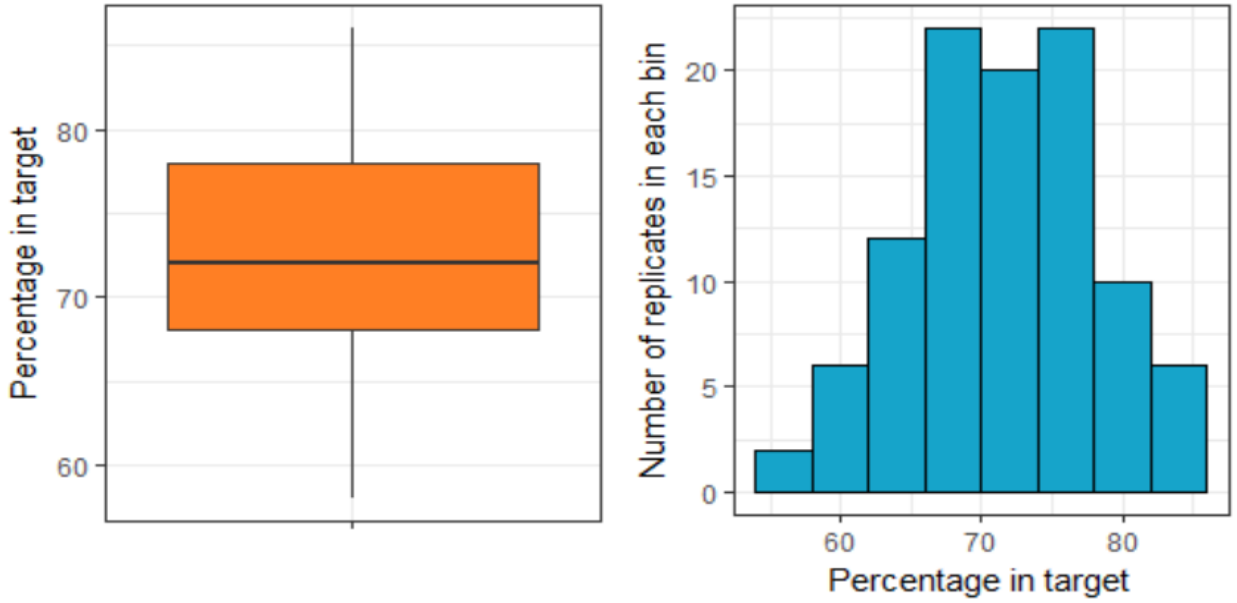
- reuse a monolix project
- change the treatment applied to the individuals: 25 or 50 ug/min



# Simulx: replicates

## 3. Uncertainty of an endpoint:

- Define an endpoint: percentage of individuals with 5 Hz < EEG < 15 Hz
- Do replicates to estimate the uncertainty of the endpoint ('nrep' argument)



# Simulx: clinical trials

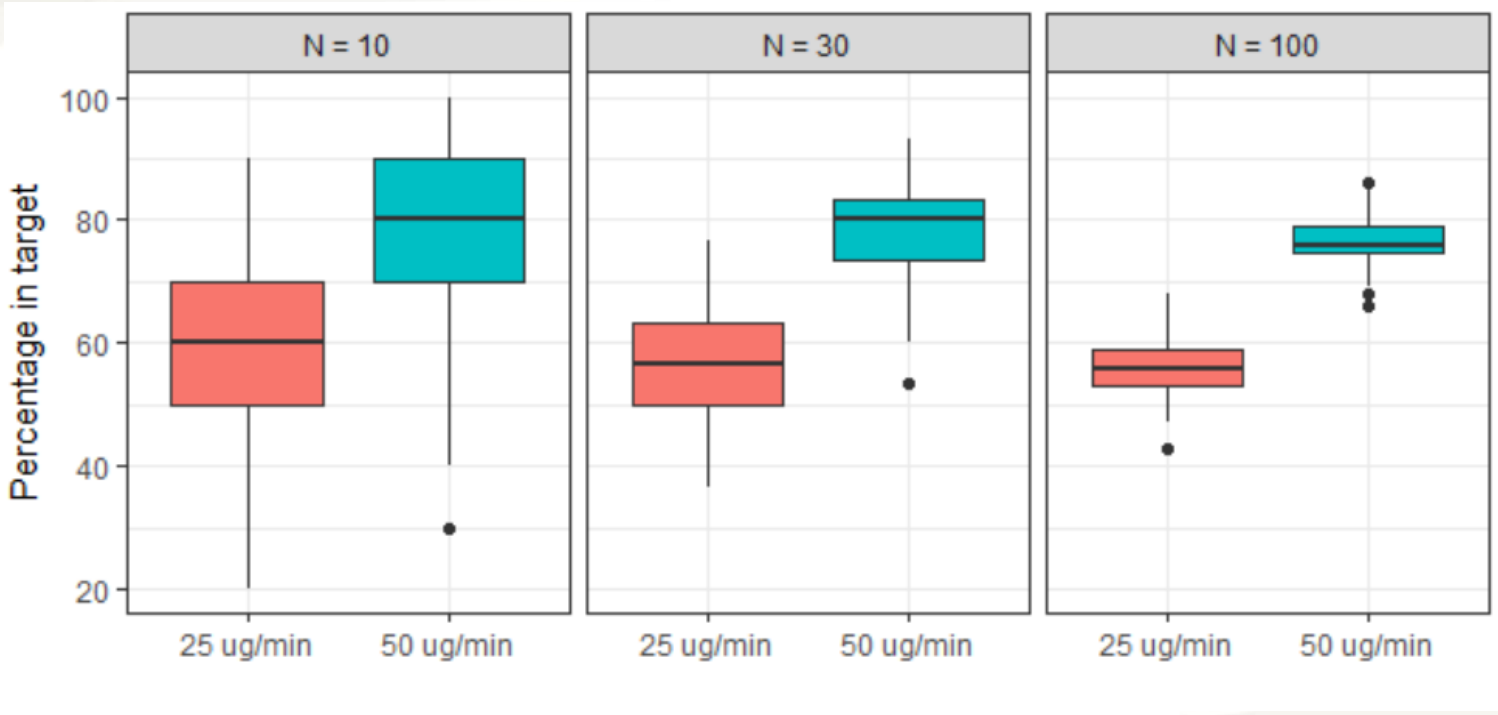
4. Clinical trial simulations with different numbers of individuals:
   - N = 10, 30, and 100 patients
   - repeated administrations every 4 weeks with 50 ug/min or 25 ug/min

*=> we simulate M=100 replicates of each trial design,*
*i.e. a total of (10+30+100) × 2 ×100 = 48 000 simulations.*

*=> runs in 4 minutes on a regular laptop*

# Simulx: clinical trials

4. Clinical trial simulations with different numbers of individuals:



# Simulx: clinical trials

## 4. Clinical trial simulations: power of the study

- success of a one clinical trial if proportion in target significantly higher for 50 ug/min compared to 25 ug/min
- count the number of successes to estimate the power of the study

| | N = 10 | N = 30 | N=100 |
|---|---|---|---|
| power | 0.18 | 0.55 | 0.98 |

# Simulx: uncertainty

## 5. Uncertainty of population parameters

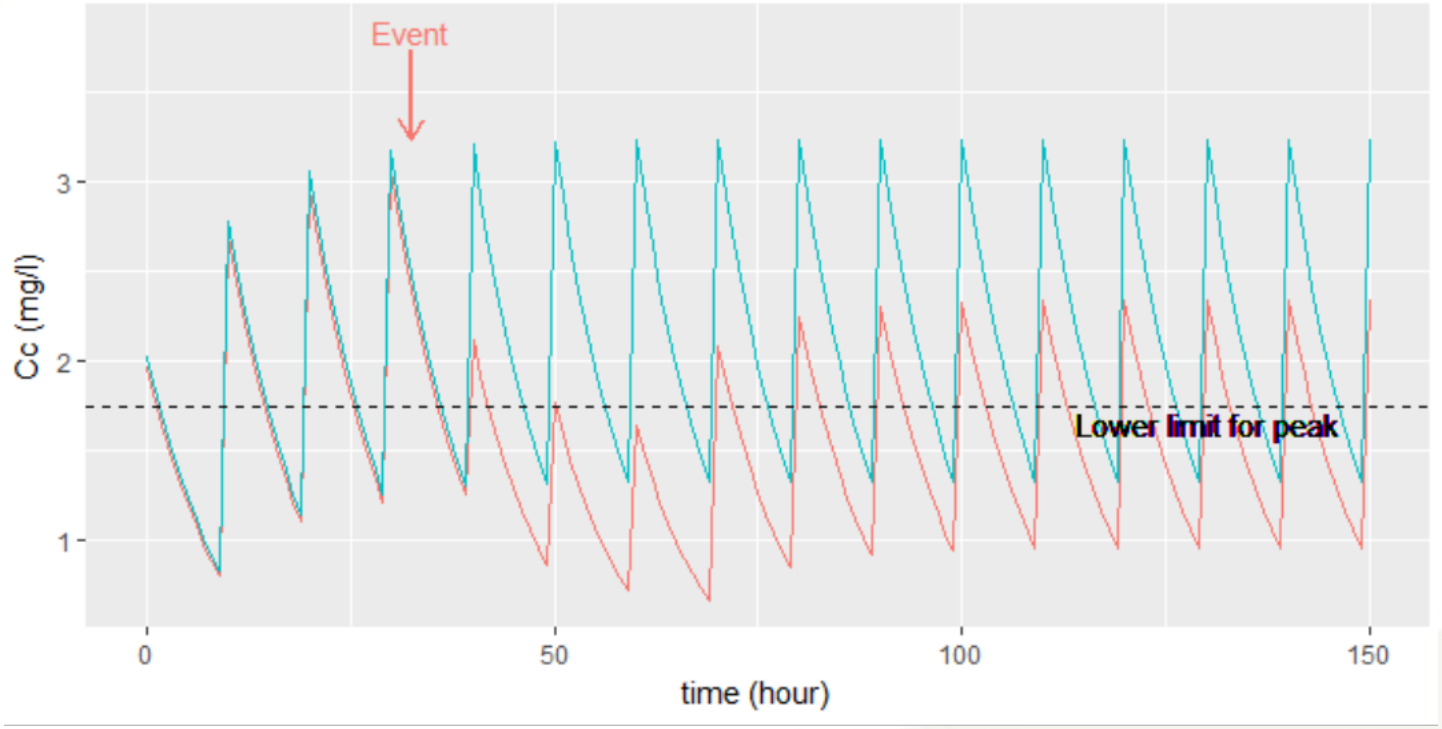
- replace nrep by npop
- use s.e of population parameters to draw population parameters from their uncertainty distribution

| id | pop | IC50_pop | IC50 |
|---|---|---|---|
| 1 | 1 | 12.486 | 16.174447 |
| 2 | 1 | 12.486 | 12.708641 |
| 3 | 1 | 12.486 | 9.237041 |
| 1 | 2 | 13.712 | 13.106824 |
| 2 | 2 | 13.712 | 9.198109 |
| 3 | 2 | 13.712 | 20.120062 |

# Simulx: dose adaptation

## 6. Dose adaptation using therapeutic drug monitoring

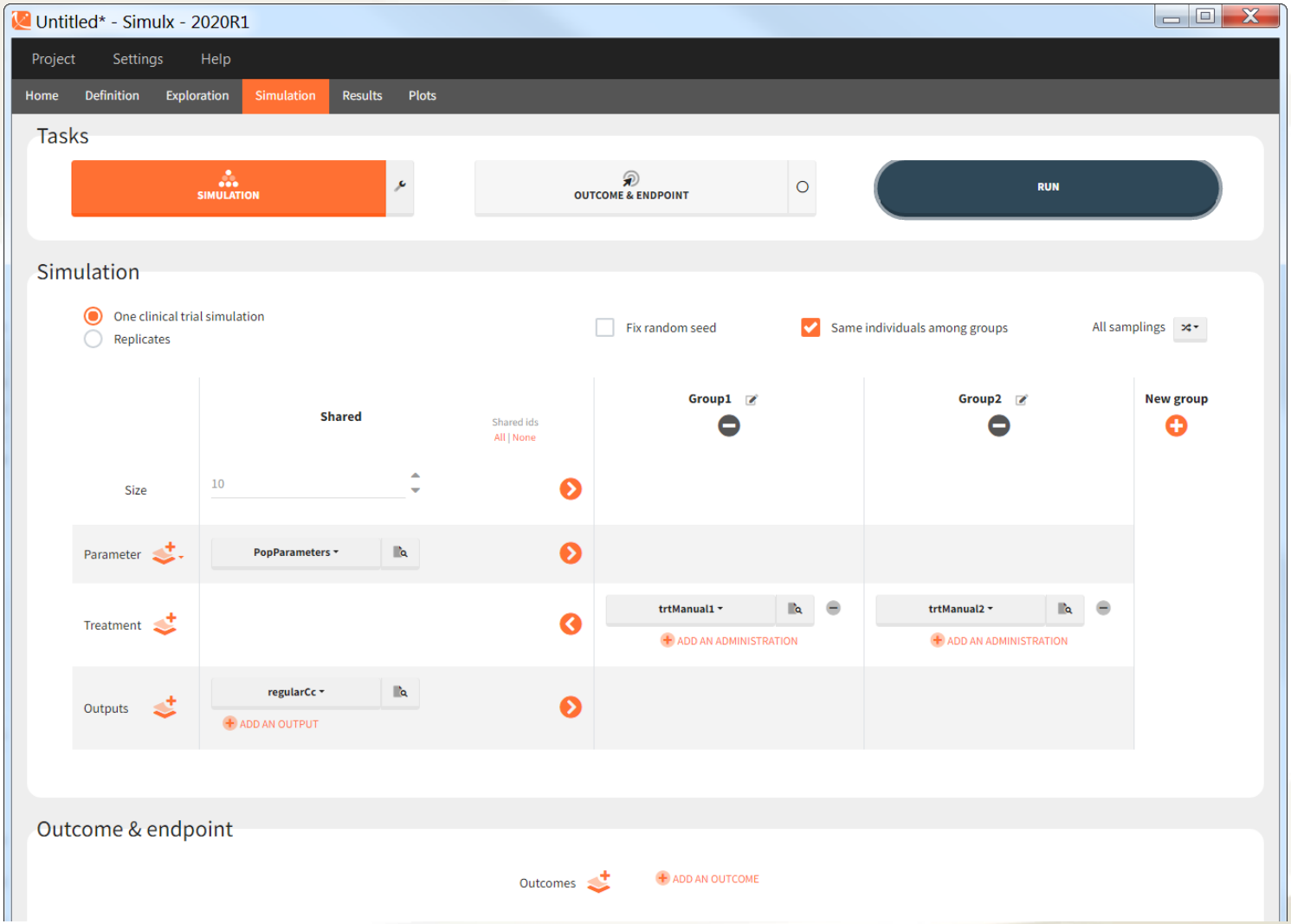
- titration function to define a target and how dose can be changed



# SIMULX-GUI

Simulations

# Simulx-GUI

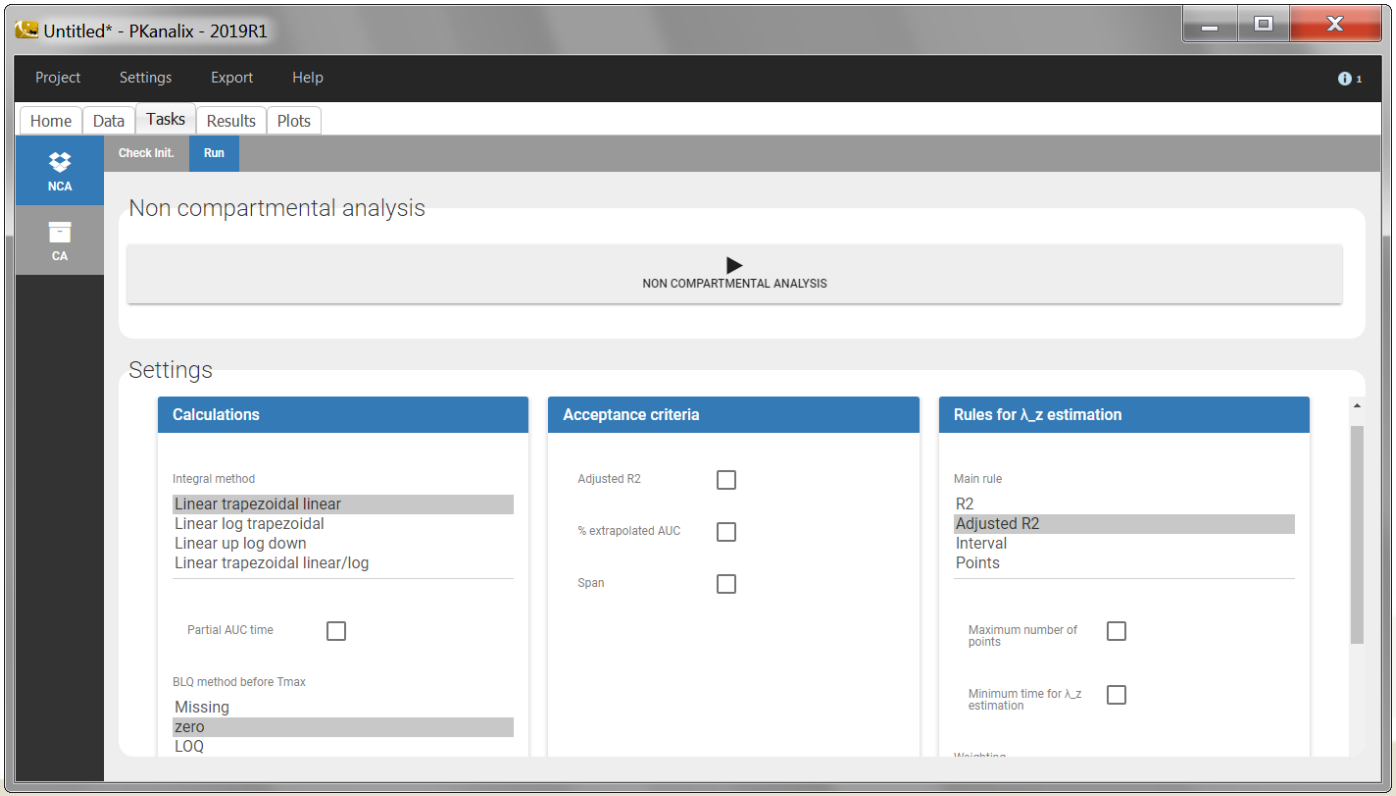

Untitled* - Simulx - 2020R1
Project
Settings
Help
Home
Definition
Exploration
Simulation
Results
Plots
Tasks
SIMULATION
OUTCOME & ENDPOINT
RUN
Simulation
One clinical trial simulation
Replicates
Fix random seed
Same individuals among groups
All samplings
Shared
Shared ids
All | None
Group1
Group2
New group
Size
10
Parameter
PopParameters
Treatment
trtManual1
trtManual2
ADD AN ADMINISTRATION
ADD AN ADMINISTRATION
Outputs
regularCc
ADD AN OUTPUT
Outcome & endpoint
Outcomes
ADD AN OUTCOME

# PKanalix

NCA and CA analysis

# PKanalix: NCA & CA analysis

- fast and user-friendly
- out-of-the-box and report-ready plots and tables
- Industry-standard calculations



# Conclusion

- MonolixSuite2019R1: 1 language – 5 applications
  - State-of-the-art statistical methods
  - Increased productivity and quality
  - Ease of use
  - Lixoft's export support
- Contact us:
  - support@lixoft.com
- www.lixoft.com