Pharmacometrics R-shiny tools for drug development

Patrick Nolain
Pharmacometrician @ Sanofi

IATDMCT 2019 Conference
25 September 2019

17th IATDMCT
2019 Congress

Motivation

One of the most powerful **analytic tools** available to solve statistical problems

Widely used many domains, especially in life sciences and biostatistics

Analyses made of **R scripts**

Results are presented/shared in a **static** format

Problem

"Can you make me this graph? And now this graph? OK... one more plot?"

Results engender **additional questions**... that cannot be answered immediately

How can we solve this?

Distribute your work and give **more knowledge** to your audience

Shiny

Open source R package developed by Joe Cheng (RStudio)

Framework for building **web applications** that can communicate with **R**

Shiny
www.rstudio.com
=
R
+
+

Create web applications around R analyses and visualizations

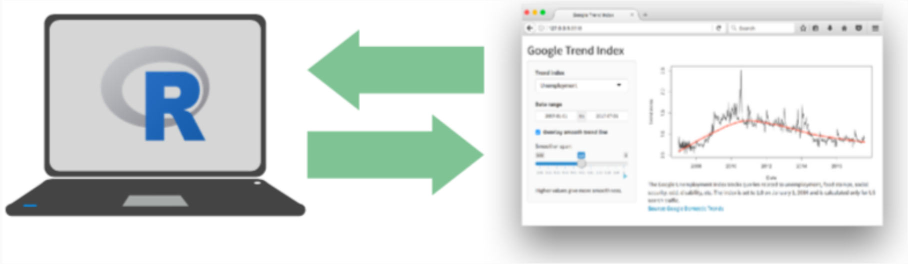
No knowledge of web programming (HTML, CSS, JavaScript...) is required

Only previous experience with R programming language is required to create a web app

Allows building interactive tools with R

What is a Shiny application?

A shiny app is a web page (connected) to a computer running a live R session



Can be viewed locally or via internet

Allows scientists/R-programmers to interactively show the output of their R program through web browsers

Visit the 🔗 Shiny Gallery

Examples: Economics forecasting, Fantasy football, Visualization of genome data, Statistics teaching, etc...

Anatomy of a Shiny app

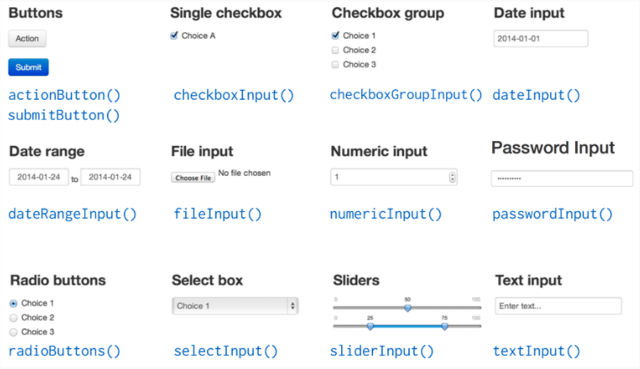
User Interface (UI)

Consists of all the graphical components needed for user interaction

Controls the **layout** and **appearance**, user-input widgets and the output to be displayed

Widgets allow users to input their choices

Variety of input options



Buttons	Single checkbox	Checkbox group	Date input

actionButton()
submitButton()
	checkboxInput()	checkboxGroupInput()	dateInput()

Date range	**File input**	**Numeric input**	**Password Input**
dateRangeInput()	fileInput()	numericInput()	passwordInput()
Radio buttons	**Select box**	**Sliders**	**Text input**
radioButtons()	selectInput()	sliderInput()	textInput()

Bootstrap 3 web library (HTML/CSS/JS) (responsive design)

UI components can be easily customized or extended

Keep it simple!

Server Instructions

Backbone of the application

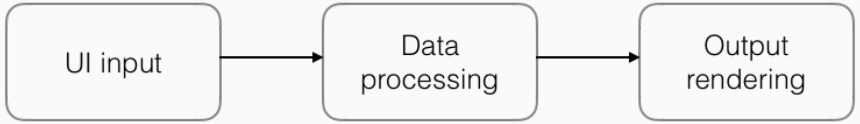
Where **computational logic** lives

Provides the reactive environment for the interactive UI

Controls the processing of user-input to display output to the user-interface

Accepts any R code

Shiny-specific code



`reactive` expressions

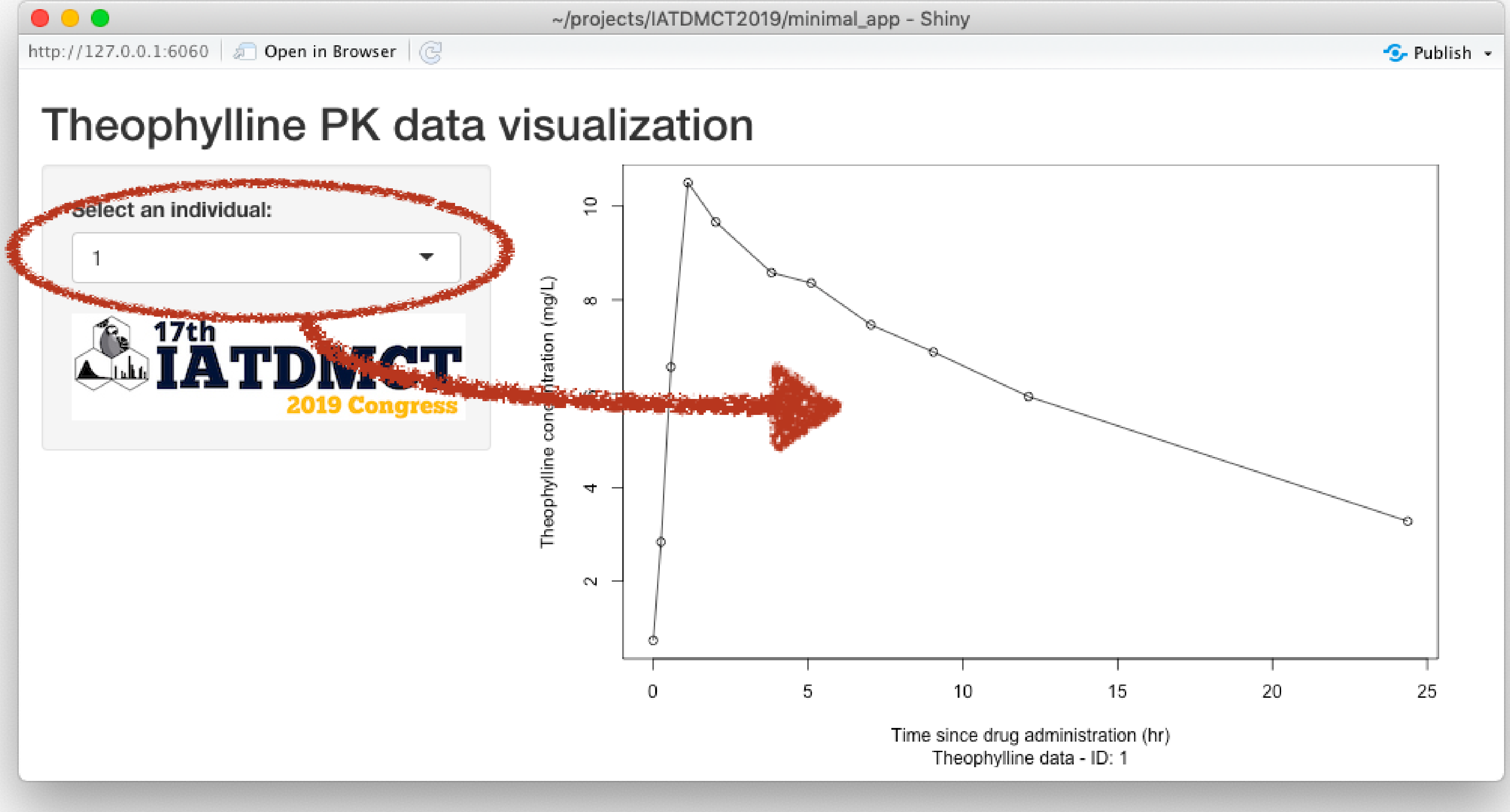
"objects" that can change based on the user input

`render*()` functions

renderText(), renderPlot(), renderTable(), etc...

Elements that are re-executed on every change of a widget as they are dependent on widget input

Minimal Shiny application



~/projects/IATDMCT2019/minimal_app - Shiny
http://127.0.0.1:6060
Open in Browser
Publish
17th IATDMCT 2019 Congress
10
8
4
2
0
5
10
15
20
25
Theophylline concentration (mg/L)

Theophylline PK data visualization

Select an individual:

1

Time since drug administration (hr)
Theophylline data - ID: 1

Minimal Shiny application

ui.R

User Interface (UI)

```r
# Display adjusts automatically to the browser dimensions
ui <- fluidPage(
  titlePanel("Theophylline PK data visualization"),
  sidebarLayout(
    sidebarPanel(
      # Create a select drop-down
      selectInput("subject_id", "Select an individual:", choices = 1:12),
      img(src = "iatdmct2019.jpg")),
    mainPanel(
      # Place-holder for a plot to draw
      plotOutput("pk_profile")
    )
  )
)
```

Series of nested shiny functions controlling the layout of the content

server.R

Server Instructions

```r
server <- function(input, output) {
  # Filter ID data
  individual_data <- reactive({
    # Subset the Theophylline dataset
    subset(Theoph, Subject == input$subject_id)
  })

  # Generate a plot named "pk_profile"
  output$pk_profile <- renderPlot({
    cc_data <- individual_data()

    plot(conc ~ Time, data = cc_data,
         xlab = "Time since drug administration (hr)",
         ylab = "Theophylline concentration (mg/L)",
         sub  = paste0("Theophylline data - ID: ", input$subject_id),
         type = "o")
  })
}
```

Run !

```r
shiny::runApp(ui = ui, server = server)
```

Shiny for pharmacometrics

Pharmacometrics

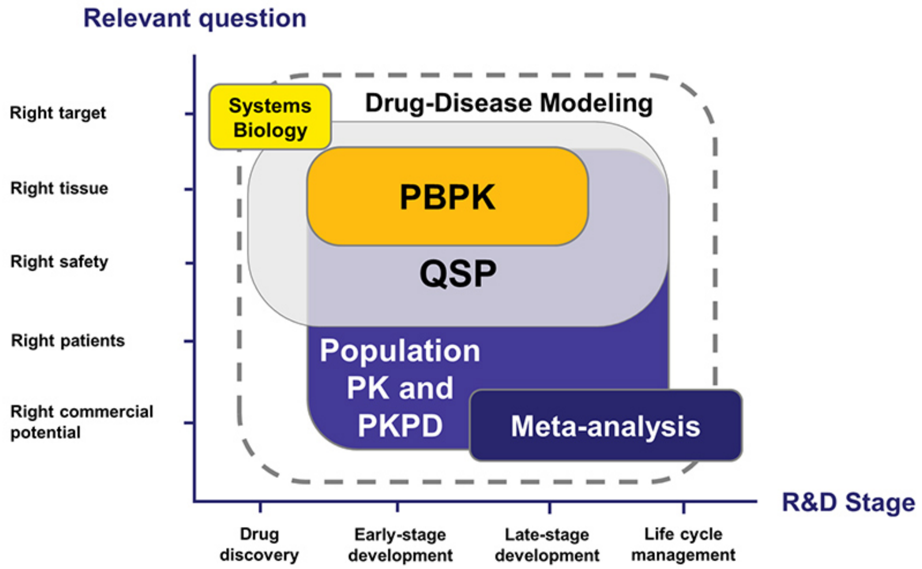
Mathematical equations to summarize the behavior of complex biological systems

Supports: exposure-response, proof of concept, trial design, trial simulation...

Population Pharmacokinetic and/or Pharmacodynamic models describe:

- Underlying mechanisms responsible for observed data
- Variability in a data set

Method of choice for quantifying relationships between pharmacokinetics and pharmacodynamics



Drug-disease modeling. QSP: Quantitative Systems Pharmacology; PBPK: Physiologically-Based Pharmacokinetics.

Reference: Helmlinger G et al. Eur J Pharm Sci. 2017 Nov 15;109S:S39-S46

Shiny for pharmacometrics

Demonstrative purpose

- Visualization of interactive simulations:
  - Communicate efficiently model's features
  - Visualize parameters impact
  - Help design future studies

Examples of applications

- Interactive model simulations
- Model post-processing analysis
- Therapeutic Drug Monitoring

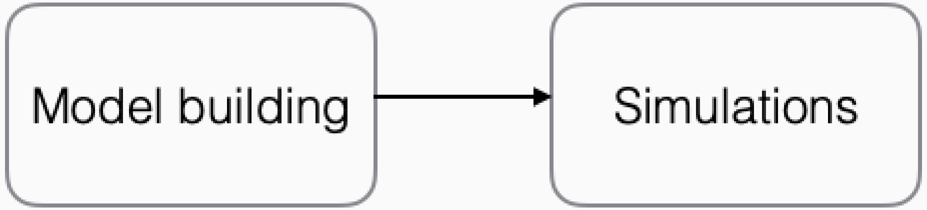
Explorative purpose

- Model development
  - Exploratory data analysis
  - Front-end around command line interfaces
  - Post-processing analysis
- Visualization of model dynamics
  - Understanding parameter roles/effects and ranges (eg. mechanistic PK/PD models)

Interactive model simulations

Modeling & simulation workflow

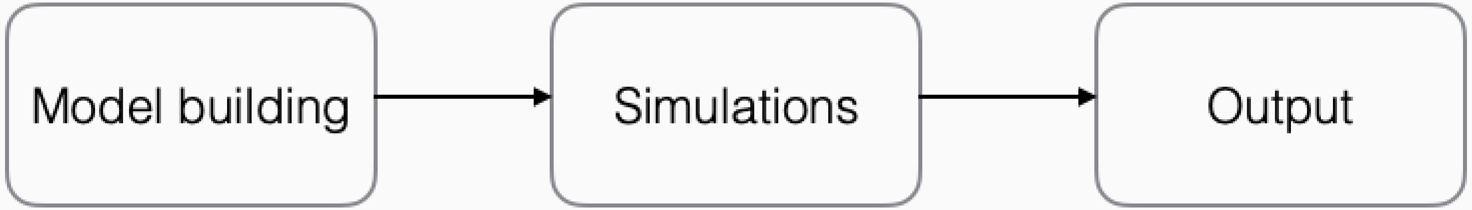
How do we use models?



- From a model
- Simulate an administration scenario

Modeling & simulation workflow

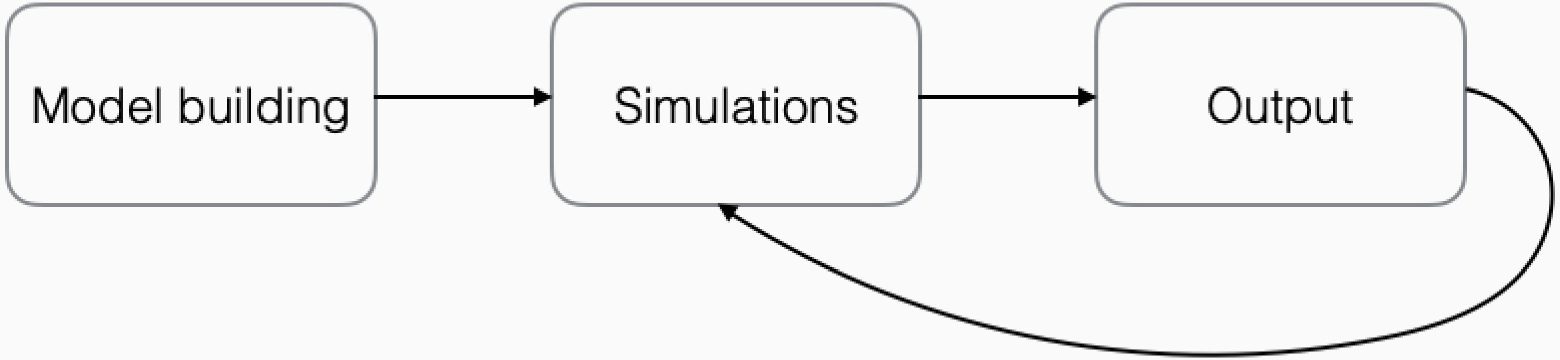
How do we use models?



- From a model
- Simulate an administration scenario
- Generate an output summarizing the model outcome
    - Concentration/effect vs time profile
    - Exposure summary statistics (AUC, Cmax, probability of target attainment...)

Modeling & simulation workflow

How do we use models?



- From a model
- Simulate an administration scenario
- Generate an output summarizing the model outcome
  - Concentration/effect vs time profile
  - Exposure summary statistics (AUC, Cmax, probability of target attainment…)

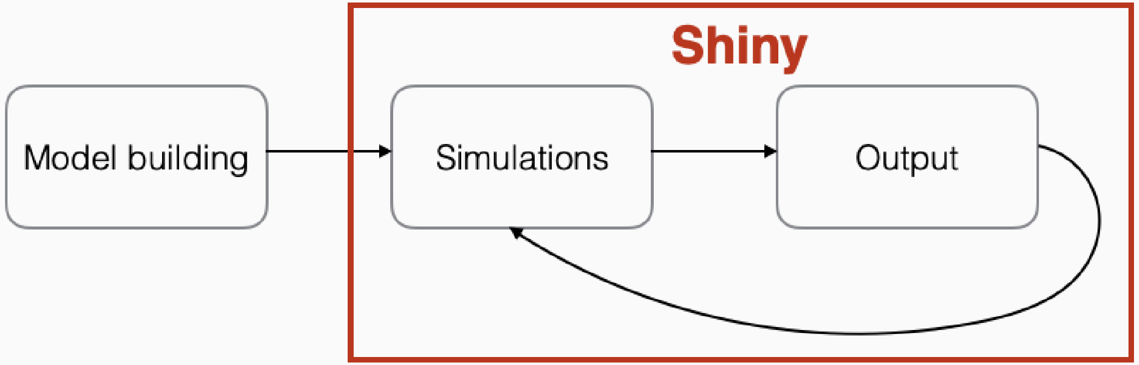
Repeat for each **N scenarios**

Time-consuming process requiring expertise and **speciliazed software** (NONMEM®, Monolix®…)

Not interactive, need to rationalize the number of hypotheses or regimen to test

Model simulations

Build an interactive application for model simulations



Smooth the process

Evaluation of “what-if” scenarios by means of simulations

- Change in doses / dosing regimen,
- Impact of covariates (age, gender, weight, comedications...)

Strengthen the role of PMX

Model simulations

Objective: Evaluation of “what-if” scenarios by means of simulations

Target audience: Clinical pharmacologists/pharmacokineticists

USER INTERFACE

Input

Administration scenario (dose, route, dosing interval)

Individual characteristics (covariates, parameters, variability)

Output

Visualizing PK/PD profiles

Computing (and exporting) exposure, metrics

SERVER

Simulation computations

Model implementation

- Analytical model
- ODE model: dedicated R packages

→ `deSolve`, `mrgsolve`, `RxODE`, `mlxR`

Typical simulations

Simulations with variability/uncertainty

Model simulations : Concrete case

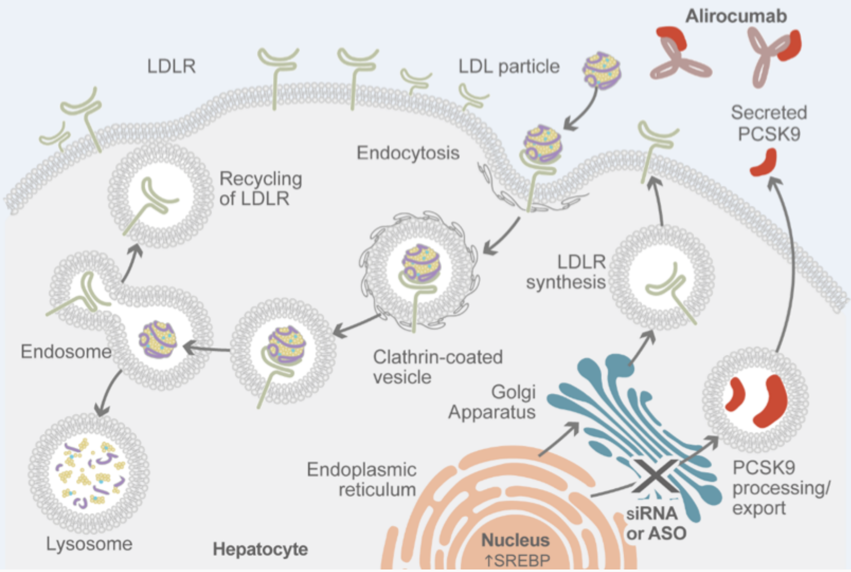
Alirocumab (Praluent®) : anti-PCSK9 monoclonal antibody - Lipid-lowering therapy

Indications

- Treatment of adults with primary hyperlipidemia (including heterozygous familial hypercholesterolemia) to reduce low-density lipoprotein cholesterol (LDL-C)
- Reduction of the risk of myocardial infarction, stroke, and unstable angina requiring hospitalization in adults with established cardiovascular disease

Mechanism of action

Decreases LDL-C levels by blocking the interaction between PCSK9 and LDL-R



Model simulations : Concrete case

Population PK

Population Pharmacokinetic Analysis of Alirocumab in Healthy Volunteers or Hypercholesterolemic Subjects Using a Michaelis–Menten Approximation of a Target-Mediated Drug Disposition Model—Support for a Biologics License Application Submission: Part I

Martinez et al. 🔗 doi: 10.1007/s40262-018-0669-y

Population PKPD

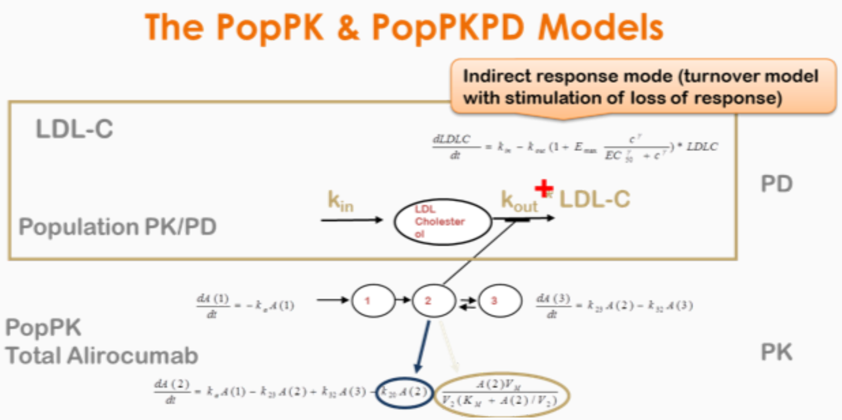
Population Pharmacokinetic/Pharmacodynamic Analysis of Alirocumab in Healthy Volunteers or Hypercholesterolemic Subjects Using an Indirect Response Model to Predict Low-Density Lipoprotein Cholesterol Lowering: Support for a Biologics License Application Submission: Part II

Nicolas et al. 🔗 doi: 10.1007/s40262-018-0670-5

Target-Mediated Drug Disposition PK

Target-Mediated Drug Disposition Population Pharmacokinetics Model of Alirocumab in Healthy Volunteers and Patients: Pooled Analysis of Randomized Phase I/II/III Studies

Djebli et al. 🔗 doi: 10.1007/s40262-016-0505-1



The PopPK & PopPKPD Models

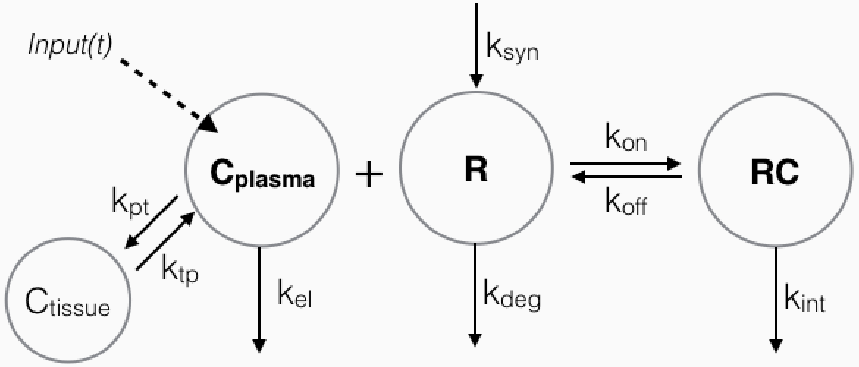
Indirect response mode (turnover model with stimulation of loss of response)

$$\frac{dLDLC}{dt} = k_{in} - k_{out}(1 + E_{max}\frac{c^{\gamma}}{EC_{50}^{\gamma} + c^{\gamma}}) * LDLC$$

$$\frac{dA(1)}{dt} = -k_a A(1)$$

$$\frac{dA(3)}{dt} = k_{23}A(2) - k_{32}A(3)$$

$$\frac{dA(2)}{dt} = k_a A(1) - k_{23}A(2) + k_{32}A(3) - k_{20}A(2) \; \frac{A(2)V_M}{V_2(K_M + A(2)/V_2)}$$



Model simulations : Concrete case

Evaluate and compare, on the fly, multiple design and/or subject scenarios

- Facilitates PK/PD understanding
- Makes model more easy to use
- Saves a lot of time compared to previous procedure

Provide dynamic reports to decision makers

Supported a Supplemental Biologics License Agreement filing to the FDA

Any published model from the literature can be implemented

➔ Compare *in-development* product with a *standard-of-care* or *competitor*

Interactive model development

Model development

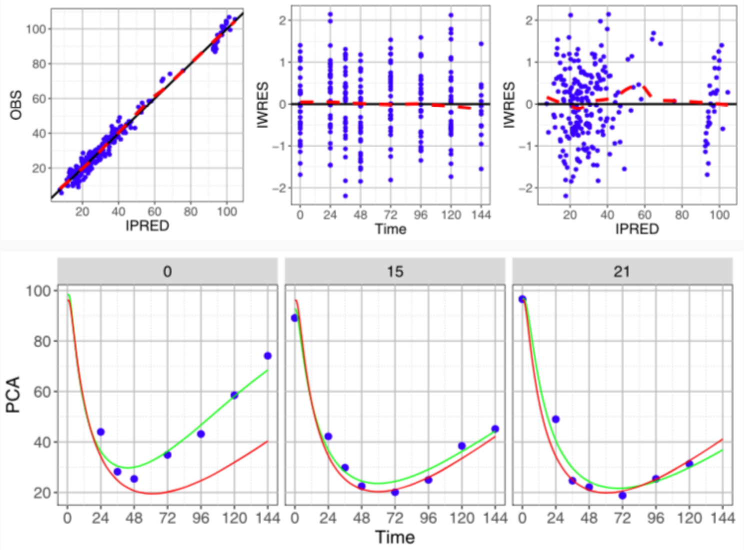
Population PK/PD modeling is a well-established method that can **quantify** and explain the **variability** in drug concentrations and effects among individuals

Softwares: NONMEM®, Monolix®, Phoenix® NLME™, Pmetrics™

Standard diagnostic methods for Non-Linear Mixed Effects Models

Model Evaluation of Continuous Data Pharmacometric Models: Metrics and Graphics Nguyen et al. 🔗 doi: 10.1002/psp4.12161 (CPT: Pharmacometrics & Systems Pharmacology)

Goodness-of-fit plots

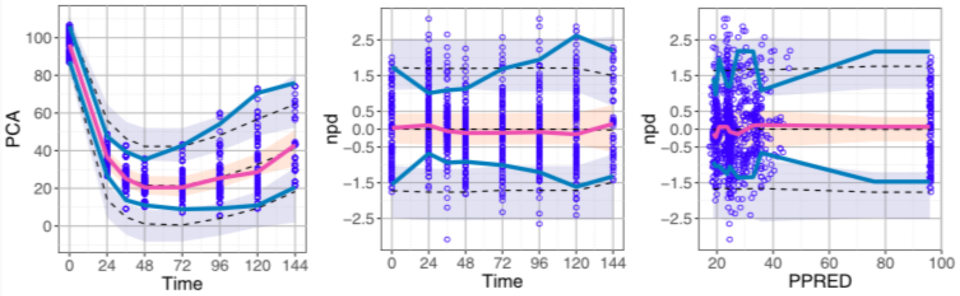


Empirical Bayes Estimates

Distributions, correlations, correlation vs covariates

Simulation-based methods

VPC, NPC, NPDE



Model development

Objective: Bring interactivity in the process of population model development

Target audience: NONMEM® users

NONMEM Toolbox app for post-processing analysis: model diagnostics, qualification, comparison…

Dynamically generates typical plots/tables and corresponding R code

Part of the `popkinr` package:

Aiming at facilitating the development of non-linear mixed effects models with NONMEM

- `pmxploit`: Post-processing of NONMEM runs
- `pmxplore`: Exploratory Data Analysis
- `pmxecute`: NONMEM run launcher front-end

🔗 github.com/pnolain/popkinr

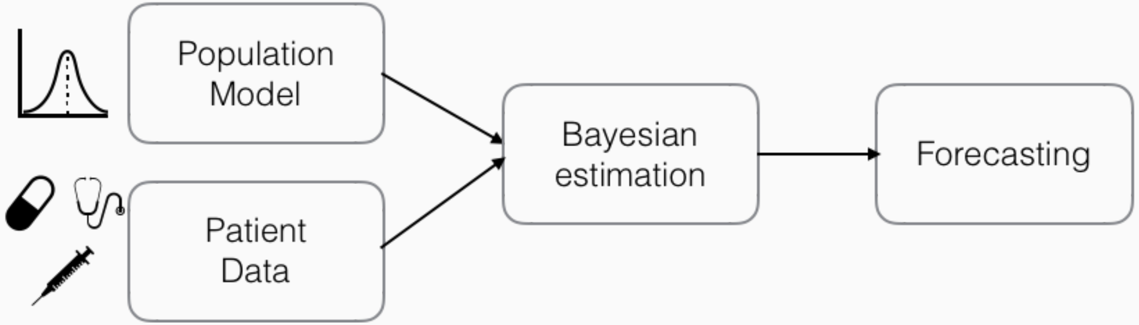
🔗 github.com/pnolain/pmxploit

Therapeutic Drug Monitoring

Therapeutic Drug Monitoring

Model-based TDM and precision dosing

Suitable for clinical pratice in real life and during drug development



Objective: Real-time Bayesian dosing recommendations

Target audience: Clinicians / Modelers

USER INTERFACE

Load/enter patient data (dosing, labs, covariates)

Model selection

Visualizing estimated/predicted individual profiles

Computing (and exporting) exposure, metrics

SERVER

Model implementation (analytical/ODE) → `mrgsolve`, `RxODE`, `mlxR`

Optimization routine to calculate MAP Bayes estimates → `optim`, `minqa`

$$O(\vec{\eta}_i) = -2LL(\vec{\eta}_i) = \sum_j \left[log(\sigma_{ij}^2) + \frac{(Y_{ij} - F_{ij})^2}{\sigma_{ij}^2} \right] + \vec{\eta}_i^{\intercal} \Omega^{-1} \vec{\eta}_i$$

Reproducibility

Reproducibility

"Reproducibility crisis" in science

Can we trust Shiny applications?

Every data analysis should be:

- **Traceable**
- **Reproducible**
- **Transparent**

Share source code files?...

...or use the `shinymeta` package

Capture logic in a Shiny app and **expose it as code** for running outside the app

Generates code for **relevant parts** of an analysis

Make things extensible and more transparent for yourself and others

Reproducibility

shinymeta syntax

User Interface (UI)

```
ui <- fluidPage(
    titlePanel("Theophylline PK data visualization - Reproducible"),
  sidebarLayout(
    sidebarPanel(
      selectInput("subject_id", "Select an individual:", choices = 1:12),
      img(src = "iatdmct2019.jpg")),
    mainPanel(
      outputCodeButton(plotOutput("pk_profile"))
    )
  )
)
```

Server Instructions

```
server <- function(input, output) {
    individual_data <- metaReactive({
    subset(Theoph, Subject == ..(input$subject_id))
  })

  output$pk_profile <- metaRender(renderPlot, {
    cc_data <- ..(individual_data())

    plot(conc ~ Time, data = cc_data,
         xlab = "Time since drug administration (hr)",
         ylab = "Theophylline concentration (mg/L)",
         sub  = paste0("Theophylline data - ID: ", ..(input$subject_id)),
         type = "o")
  })

  observeEvent(input$pk_profile_output_code, {
    code <- expandChain(output$pk_profile())
    displayCodeModal(code, wordWrap = TRUE)
  })
}
```

```
shiny::runApp(ui = ui, server = server)
```

Summary

Challenge of communication of model-based information

Shiny makes pharmacometric models accessible to a larger audience → Can accelerate model development

Elegant and flexible interfaces to models can be produced relatively quickly once the basics of Shiny are mastered

Learning tool

Explore data, explain analysis, teach methods...

Communication medium

Interdisciplinary knowledge sharing

Collaboration enhancer

Improves collaboration and promote your activities

Thank you!

Slides and apps

🔗 github.com/pnolain/IATDMCT2019

Additional resources

- Shiny Developer Center - http://shiny.rstudio.com
- Shiny Gallery - https://shiny.rstudio.com/gallery/
- Building Shiny apps: an interactive tutorial - https://deanattali.com/blog/building-shiny-apps-tutorial/
- Shiny Developer Series videos - https://shinydevseries.com/
- Model simulations
  - `mrgsolve` package - https://mrgsolve.github.io
  - `RxODE` package - https://github.com/nlmixrdevelopment/RxODE
  - *Interactive Pharmacometric Applications Using R and the Shiny Package* J Wojciechowski et al. (doi: 10.1002/psp4.21, CPT: Pharmacometrics & Systems Pharmacology)
- Model simulations with `mlxR` - http://webpopix.org/shiny/ShinyExamples.html
- `shinymeta` - https://rstudio.github.io/shinymeta