OCS LIFE SCIENCES
A DIVISION OF OCS CONSULTING

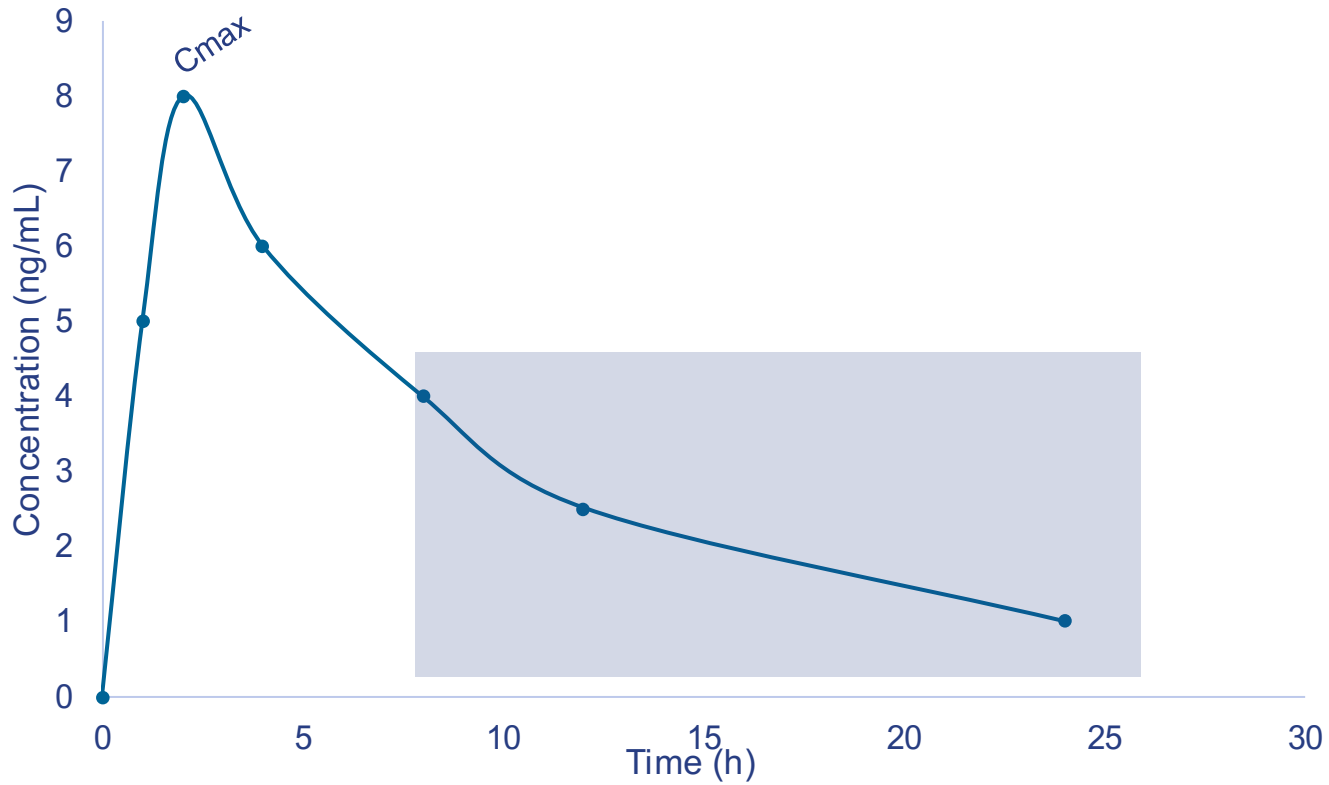
# From Prompt to PK: AI-Driven Development of an Automated NCA Optimization Tool

**Ramon Jakobs**
Statistical Programmer

# Agenda

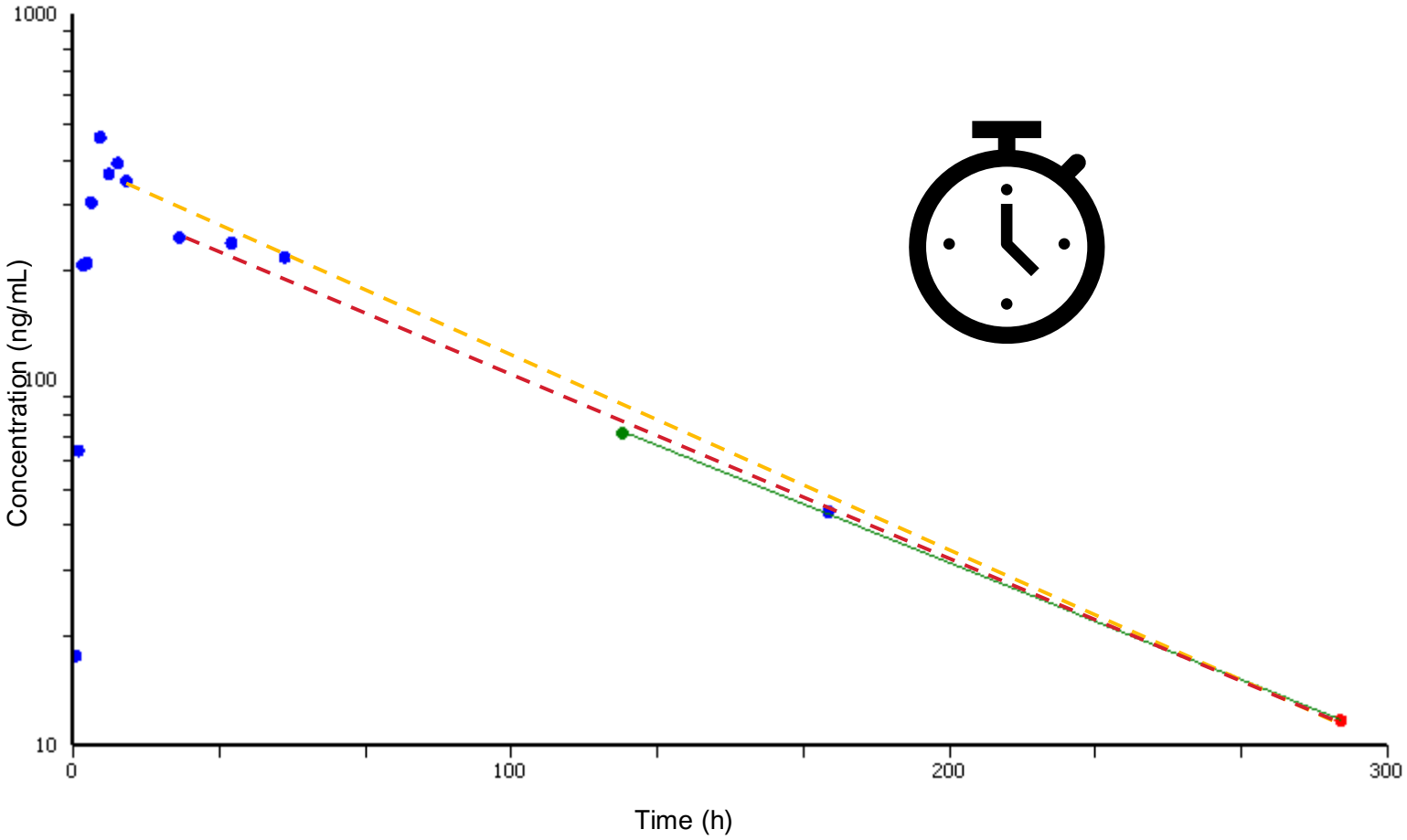
1. Pharmacokinetics and the challenge
2. From idea to solution: the journey
3. Working with AI: expectations vs reality
4. What worked well, and what didn’t
5. Lessons learned and key takeaways

# Pharmacokinetics (PK) in Clinical Studies




- Pharmacokinetics (PK) describes what the body does to the drug
- Non-compartmental analysis (NCA) summarizes PK using observed data
- Terminal phase determines elimination rate and half-life

# Manual slope selection comes with limitations



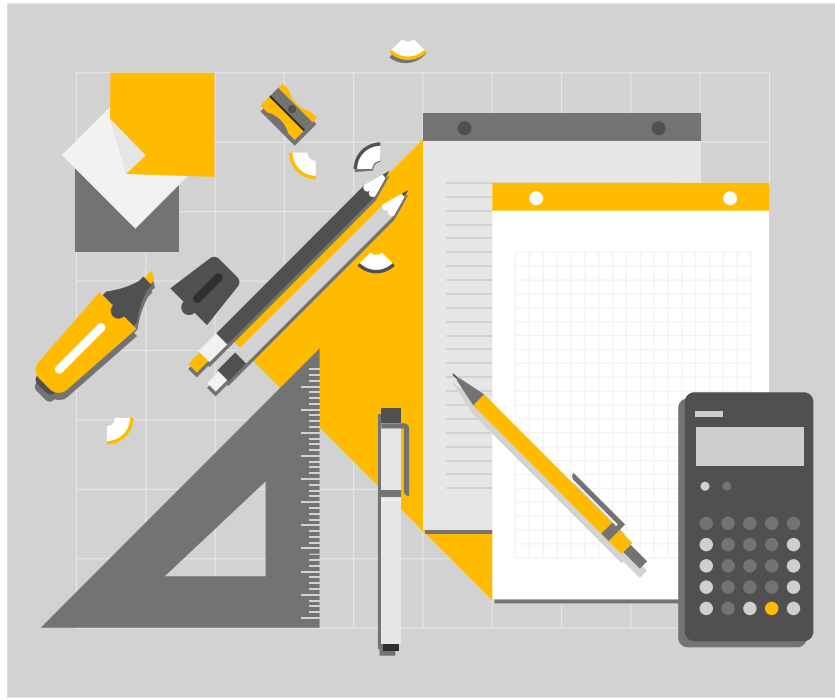

- Phoenix WinNonlin is the industry-standard tool for NCA
- Terminal slope selection manually checked

- Limitations
  - Time consuming
  - Error prone
  - Subjective → variability across analysts

# What is the idea?

## Create a tool that

- Automates terminal slope selection
- Reduces subjectivity and variability
- Limits resources spent
- Explores AI-assisted development

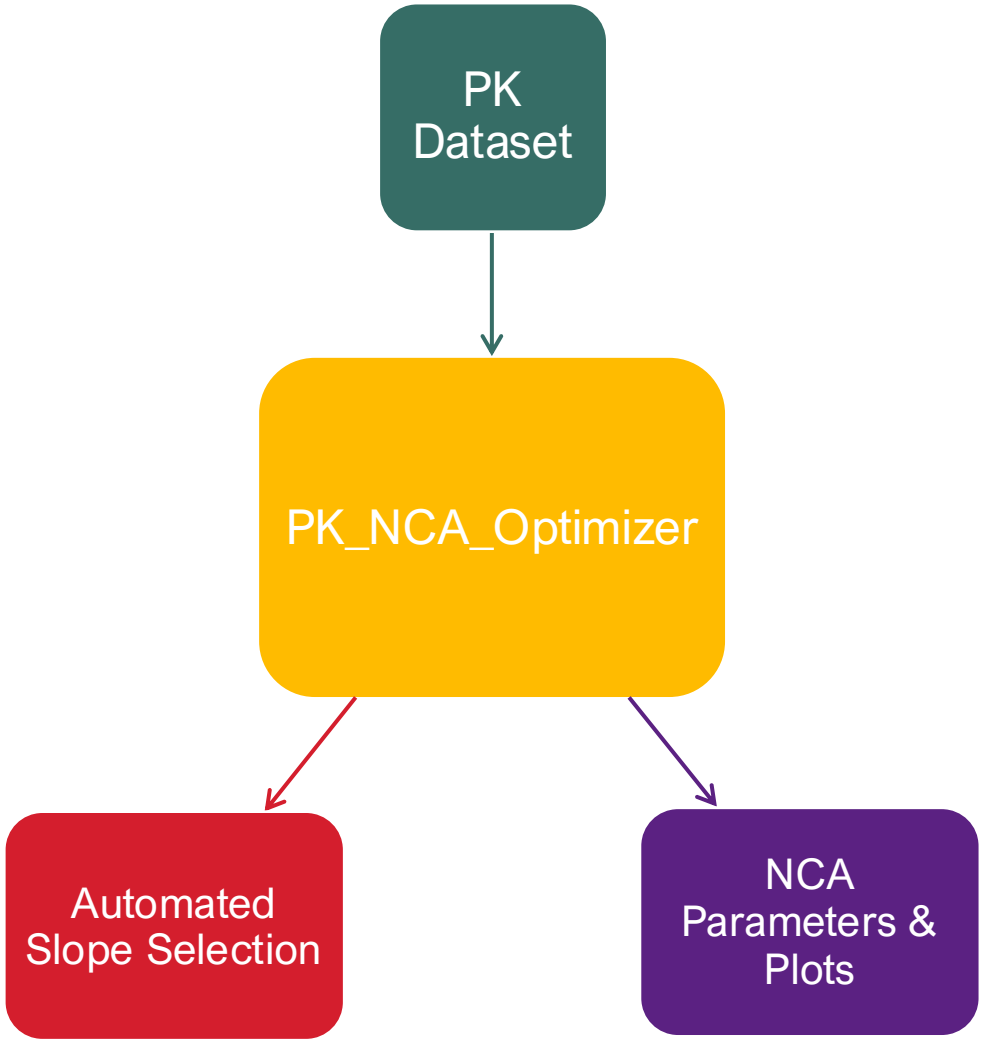


# The Journey: What do we expect?

- Support standard PK data formats (sas7bdat, XPT, xlsx)
- Allow flexible parameter settings (e.g. SPAN, number of points)
- Align with Phoenix WinNonlin slope selection principles
- Provide a simple and accessible user interface

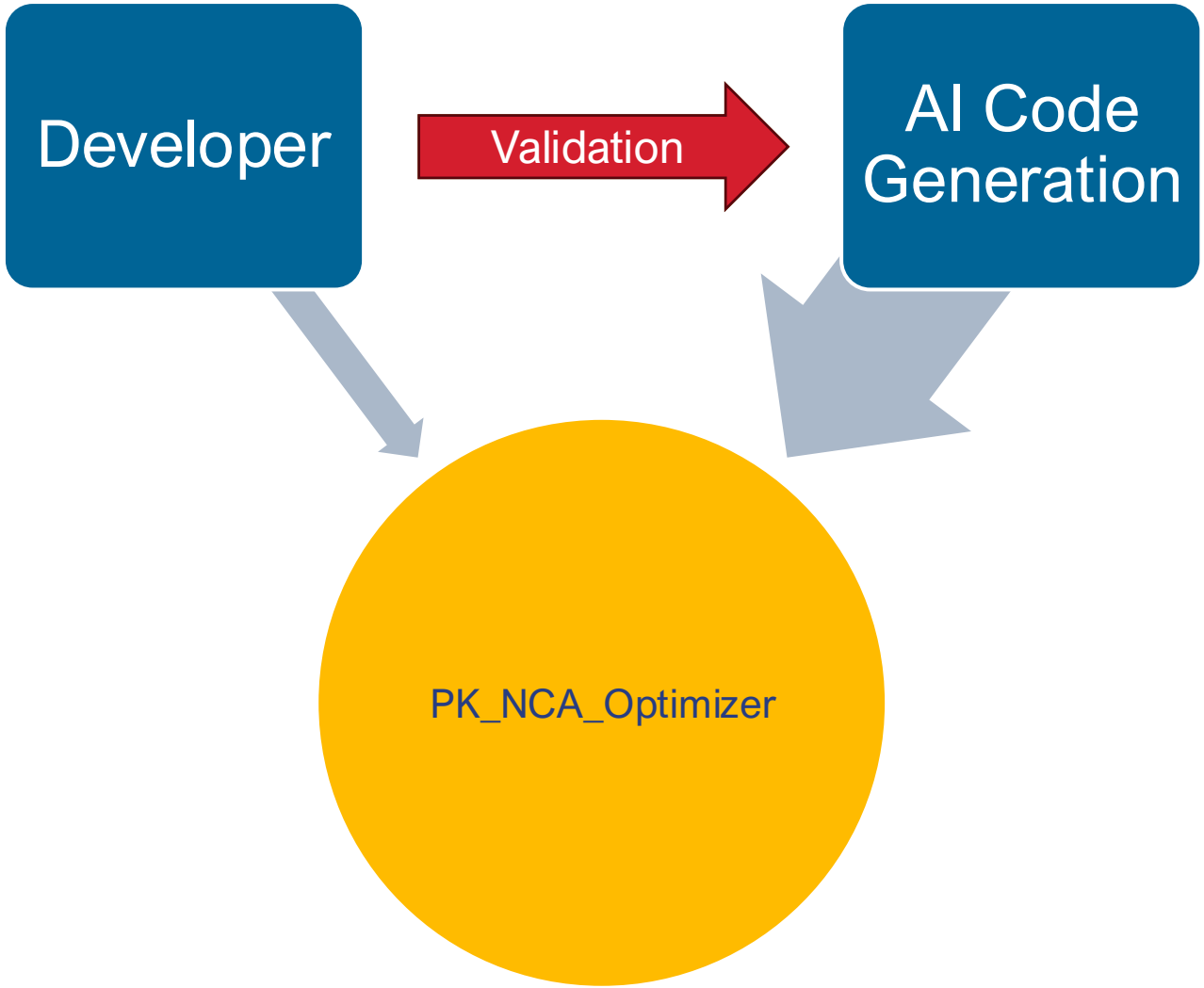
***Serve as a guidance, not as a replacement for WinNonlin***

# The Journey: What do we expect?




- Accepts clinical PK dataset as input
- Applies automated slope selection algorithm aligned with Phoenix WinNonlin
- Outputs NCA parameters and visualization plots
- Tool will NOT use AI

# AI collaboration approach




- Tool developed primarily through human–AI collaboration
- Visual Studio Code + Roo Code used for code generation
- AI translated PK domain knowledge into R Shiny application
- Focused on automating slope selection while following standard PK principles

# The Journey: Working with the AI

**Task**

I want you to create a tool which determines the best possible fit for the conc-time graphs in terms of data points for a PK NCA. this tool need to include a set of parameters which the user can change (min SPAN, min points used, min R2, maybe more). The tool imports a dataset created by NCA in WinNonlin. The problem with WinNonlin is that it always select the best fit based on R2 and you have to manually selects datapoint in Slope Selector if you want to select the best fit based on for example R2 and SPAN, which is very time consuming and I want to sort of automate this with a script. Preferably in R or SAS

**Roo has a question**

What is the typical format of the WinNonlin NCA dataset that you'll be importing? Is it a CSV, Excel file, or another format?

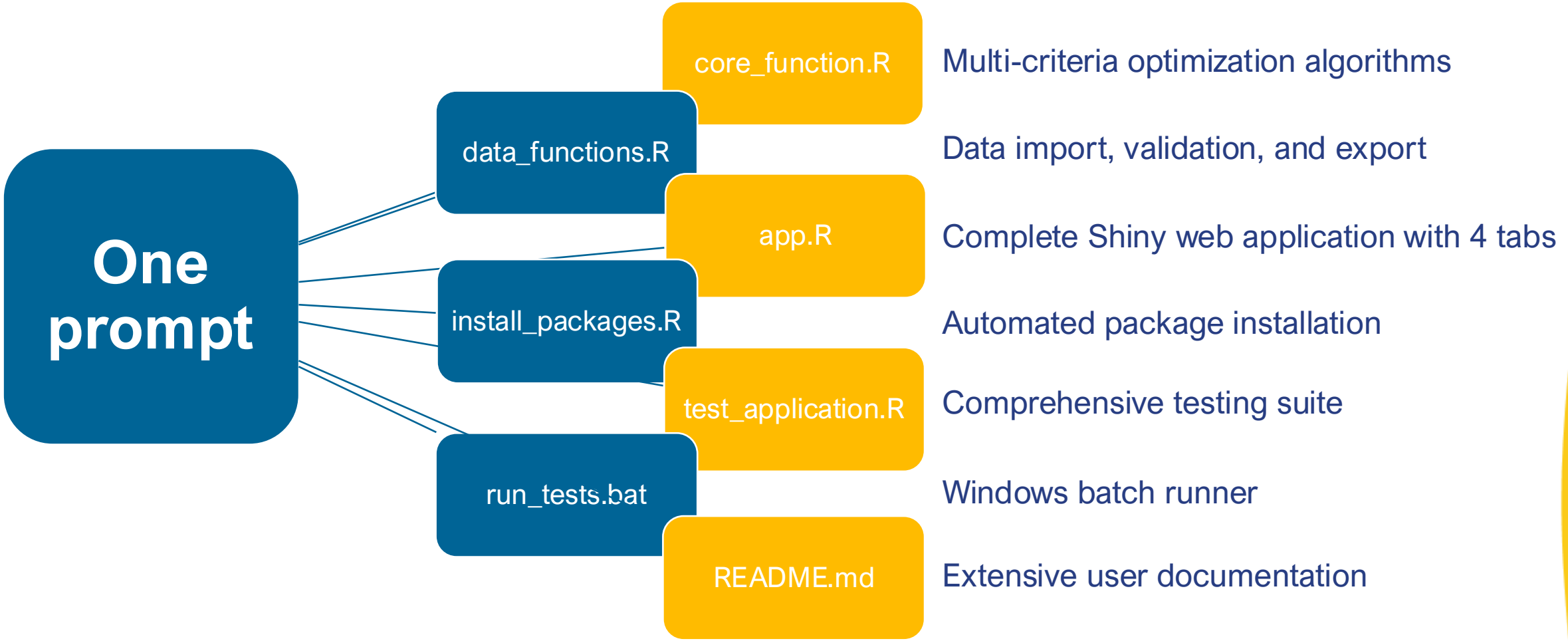
- CSV file with columns for Subject ID, Time, Concentration, and other parameters
- Excel file (.xlsx) with multiple sheets for different subjects or studies
- Tab-delimited text file with standard NCA output format
- SAS dataset (.sas7bdat) or other specialized format

**Roo has a question**

Do you have a preference for the user interface of this tool? Should it be a command-line script, an R Shiny web application, or a simple R function with parameters?

- R Shiny web application for interactive parameter adjustment and visualization
- Command-line R script that can be run from terminal/batch files
- R function/package that can be called from other R scripts
- Both a function and a simple GUI interface

# The Journey: Working with the AI



# The Journey: Working with the AI

```r
# AI-Generated Advanced Statistical Calculations
fit_terminal_phase <- function(time_range, log_conc_range) {
  # Calculate lambda_z (elimination rate constant)
  lambda_z <- abs(slope)

  # Calculate half-life from lambda_z
  half_life <- log(2) / lambda_z

  # Calculate SPAN (time range covered relativ
  time_span <- max(time_range) - min(time_rang
  span_ratio <- time_span / half_life

  # Calculate residuals for diagnostic purpose
  residuals <- residuals(fit)
  residual_runs <- calculate_runs_test(residua

  # Calculate confidence intervals for the fit
  conf_int <- predict(fit, interval = "confide

  return(list(
    lambda_z = lambda_z, half_life = half_life
    span_ratio = span_ratio, r_squared = r_squ
    equation = paste("log(C) =", round(interce
  ))
}
```

```r
#' Apply EXACT WinNonlin selection algorithm (Phoenix WinNonlin 8.x)
filter_and_rank_fits_exact_winnonlin <- function(fits, min_points = 3,
                                                 adj_r2_tolerance = 0.0001) {

  # STEP 1: Apply WinNonlin's CORE filtering criteria
  core_filtered_fits <- list()

  for (fit in fits) {
    # 1. Must have negative slope (elimination phase)
    slope_ok <- !is.na(fit$slope) && fit$slope < 0
    # 2. Must have minimum number of points
    points_ok <- fit$n_points >= min_points
    # 3. Must have valid R² values
    r2_ok <- !is.na(fit$r_squared) && !is.na(fit$adj_r_squared)
  }

  # WinNonlin Algorithm Step 1: Find maximum adjusted R²
  max_adj_r2 <- max(adj_r2_values, na.rm = TRUE)

  # WinNonlin Algorithm Step 2: Apply tolerance rule
  within_tolerance_idx <- which(abs(adj_r2_values - max_adj_r2) <= adj_r2_tolerance)
}
```

# The Journey: Working with the AI

```
#### Professional Project
```
pk_nca_optimizer/
├── app.R
├── core_functions.R
├── data_functions.R
├── tests/
│   ├── syntax_check.R
│   ├── winnonlin_compari
│   └── test_enhanced_fea
├── docs/
└── examples/
```
```

```
## 📈 Performance

- **Processing Speed**: ~50-100 subjects/minute (depending on complexity)
- **Memory Usage**: <500MB for typical datasets (1000+ subjects)
- **Scalability**: Tested with >10,000 subject-period combinations

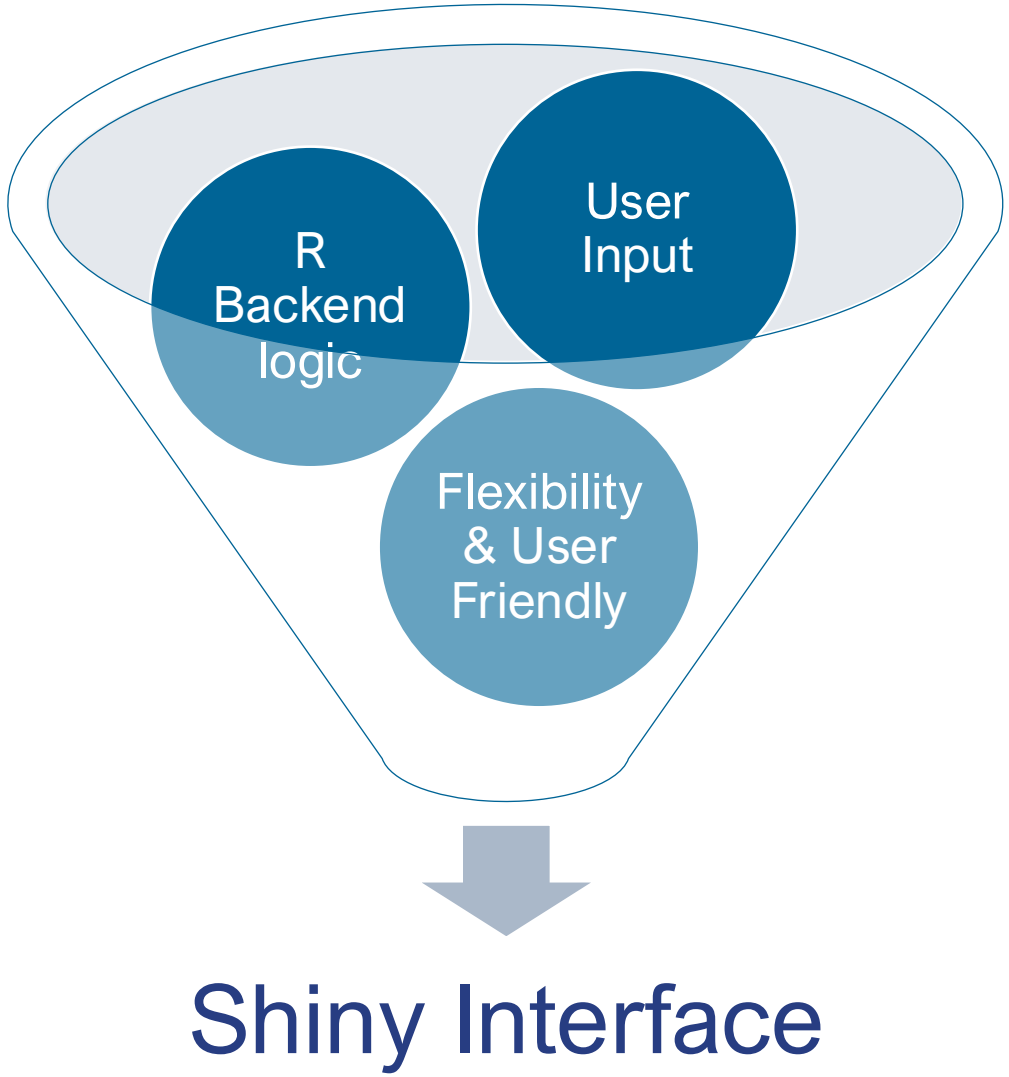
## 📋 Requirements

### R Packages (Auto-installed)
```r
shiny, shinydashboard, DT, plotly, dplyr, readr,
shinycssloaders, shinyWidgets, broom, stringr,
haven, readxl
```

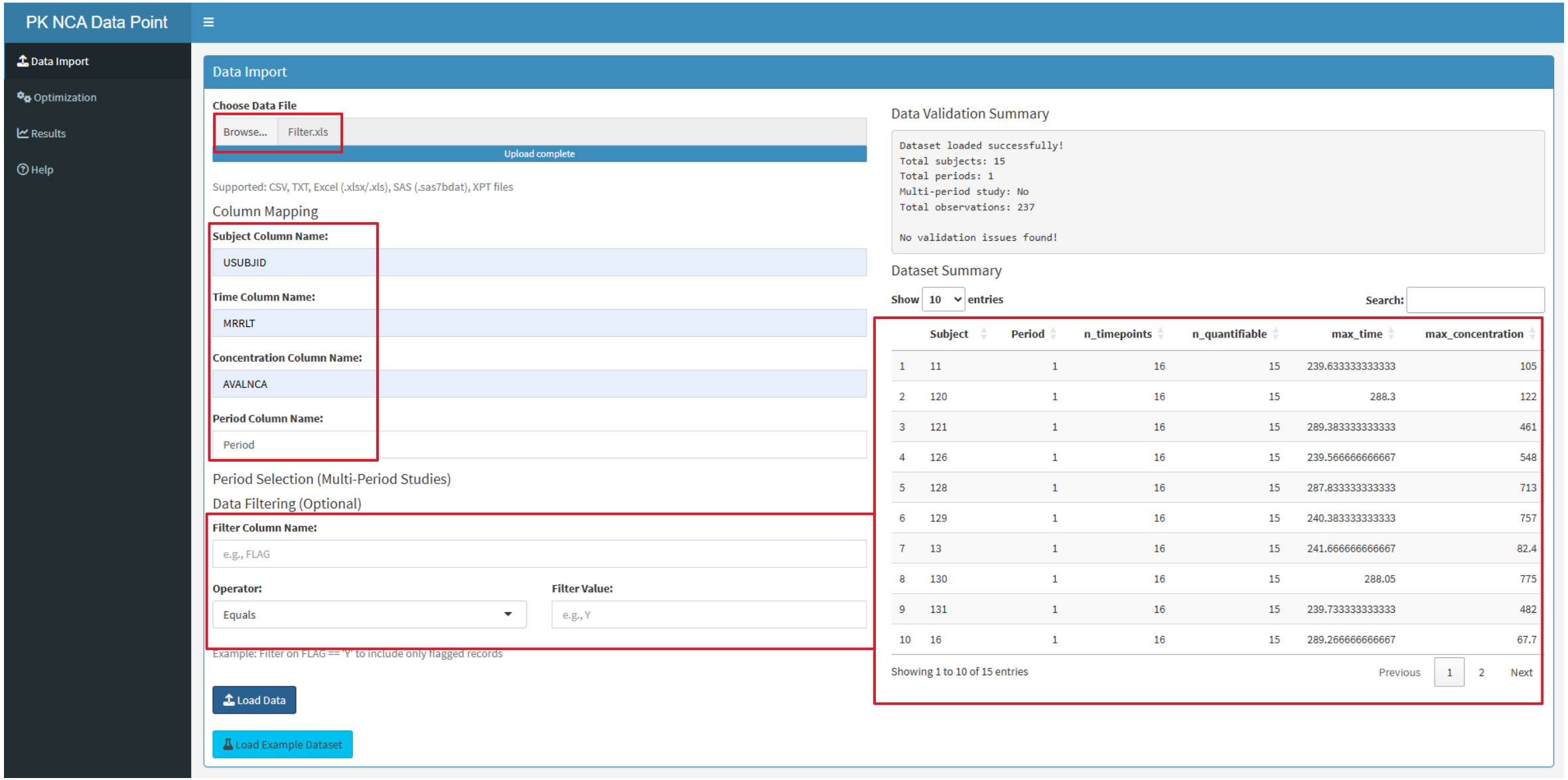
### System Requirements
- **OS**: Windows, macOS, Linux
- **RAM**: 4GB+ recommended
- **Storage**: 100MB for application + data
- **Browser**: Chrome, Firefox, Safari, Edge
```

# The Journey: Shiny App

- **What did we expect from the Shiny App?**
  - Importing
  - Data manipulation
  - Parameter selection
  - Visualisation
  - Interactive
  - Validation



# The Journey: Shiny App



# The Journey: Shiny App

PK NCA Data Point

- Data Import
- Optimization
- Results
- Help

## Optimization Parameters

### Core PK NCA Rules

- Slope must be negative (lambda_z > 0)
- Sample at tmax excluded from fitting
- At least 3 data points required
- Range ≥ 1.5x estimated half-life
- Largest adjusted $R^2$ preferred
- More points chosen if adj-$R^2$ within 0.0001

### Fitting Criteria

**Minimum $R^2$ Threshold:** 0.85

**Minimum SPAN Ratio:** 1.5

**Minimum Number of Points:** 3

### Advanced Options

**Adjusted $R^2$ Tolerance (WinNonlin):** 0.0001

WinNonlin tolerance for adjusted $R^2$ comparison. Try 0.0001, 0.0002, or 0.0005 if selection differs from WinNonlin.

- [ ] Apply AIC Filter
- [ ] Apply Runs Test

### Plot Controls

- [x] Show Observed Data Line
- [x] Show Terminal Phase Fit

### WinNonlin Compatibility

- [x] Use Exact WinNonlin Algorithm

**Uses Phoenix WinNonlin 8.x exact selection algorithm for maximum compatibility**

- [ ] Apply Regulatory Filters ($R^2$, SPAN)

▶ Run Optimization

### Progress

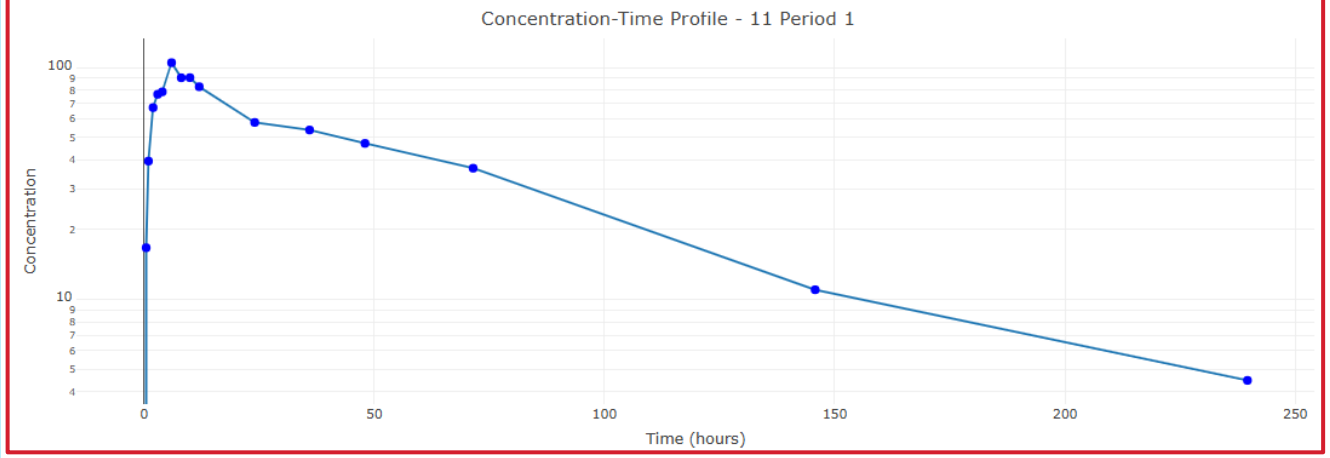
Click 'Run Optimization' to start processing...

## Subject and Period Selection

### Select Subject for Visualization

**Subject:** 11

### Concentration-Time Plot




### Terminal Phase Candidates

Click on any row to visualize that specific fit in the plot above.

### Selection Summary

# The Journey: Shiny App

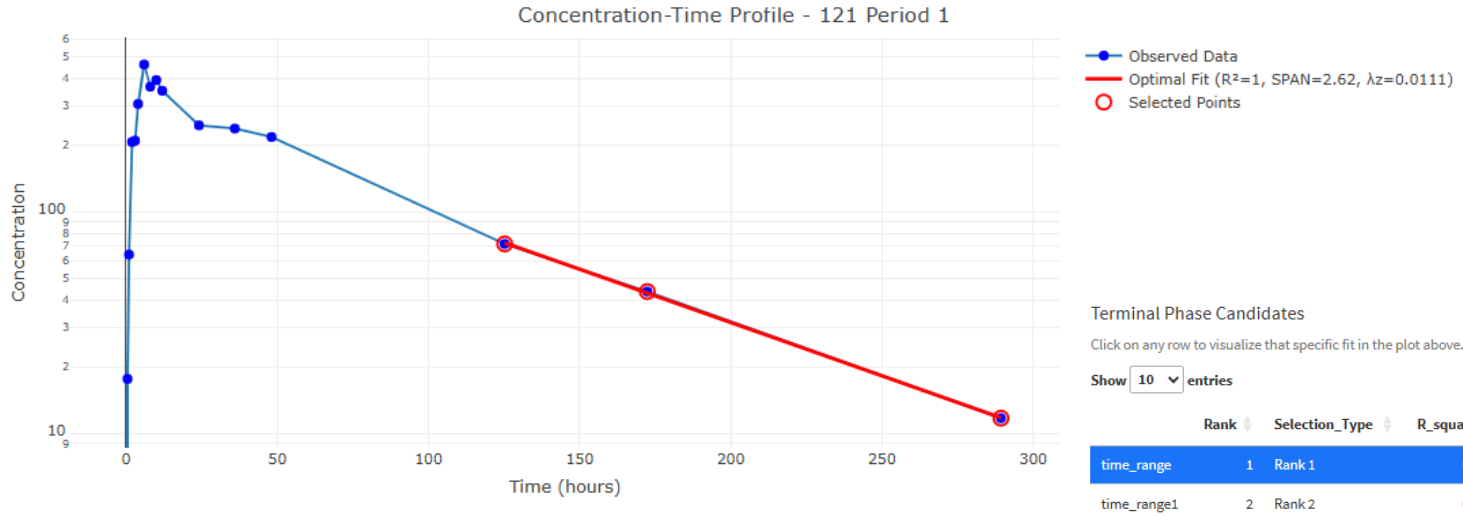
Subject and Period Selection

Select Subject for Visualization

**Subject:**

121

Concentration-Time Plot




Terminal Phase Candidates

Click on any row to visualize that specific fit in the plot above.

Show 10 entries Search:

| | Rank | Selection_Type | R_squared | Adj_R_squared | SPAN_ratio | SPAN_Status | Points | AIC | Half_life | Lambda_z | Time_range |
|---|---|---|---|---|---|---|---|---|---|---|---|
| time_range | 1 | Rank 1 | 0.9998 | 0.9995 | 2.62 | PASS | 3 | -12.23 | 62.62 | 0.0111 | 125.2 - 289.4 |
| time_range1 | 2 | Rank 2 | 0.9963 | 0.9953 | 4.54 | PASS | 6 | -9.23 | 58.44 | 0.0119 | 24.1 - 289.4 |
| time_range2 | 3 | Rank 3 | 0.9965 | 0.9953 | 4.39 | PASS | 5 | -6.96 | 57.65 | 0.012 | 36 - 289.4 |
| time_range3 | 4 | Rank 4 | 0.9951 | 0.9941 | 4.83 | PASS | 7 | -9.49 | 57.38 | 0.0121 | 12 - 289.4 |
| time_range4 | 5 | Rank 5 | 0.9955 | 0.9933 | 2.53 | PASS | 4 | -6.88 | 53.89 | 0.0129 | 36 - 172.4 |
| time_range5 | 6 | Rank 6 | 0.9965 | 0.9929 | 1.77 | PASS | 3 | -6.05 | 50.33 | 0.0138 | 36 - 125.2 |
| time_range6 | 7 | Rank 7 | 0.9951 | 0.9927 | 4.16 | PASS | 4 | -3.56 | 57.97 | 0.012 | 48.1 - 289.4 |
| time_range7 | 8 | Rank 8 | 0.9929 | 0.9917 | 4.96 | PASS | 8 | -8.59 | 56.31 | 0.0123 | 10 - 289.4 |
| time_range8 | 9 | Rank 9 | 0.9918 | 0.9891 | 2.65 | PASS | 5 | -7.16 | 55.97 | 0.0124 | 24.1 - 172.4 |
| time_range9 | 10 | Rank 10 | 0.9906 | 0.9883 | 2.95 | PASS | 6 | -8.45 | 54.43 | 0.0127 | 12 - 172.4 |

Showing 1 to 10 of 27 entries Previous 1 2 3 Next

# The Journey: Shiny App

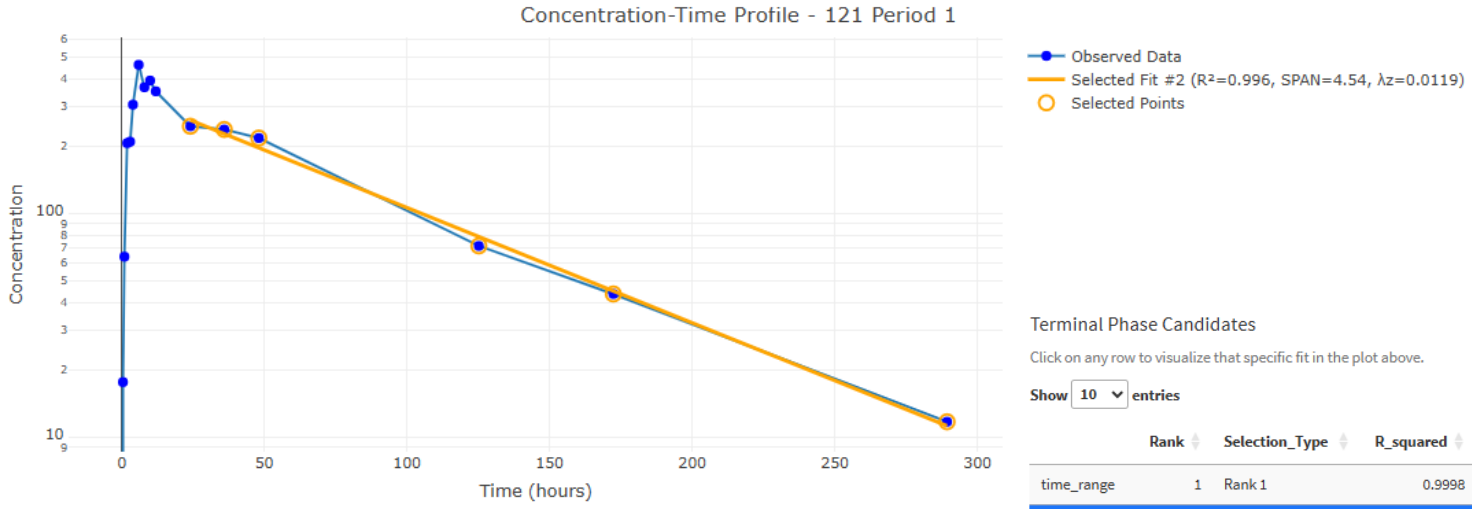
Subject and Period Selection

Select Subject for Visualization

**Subject:**

121

Concentration-Time Plot




Terminal Phase Candidates

Click on any row to visualize that specific fit in the plot above.

Show 10 entries

Search:

| | Rank | Selection_Type | R_squared | Adj_R_squared | SPAN_ratio | SPAN_Status | Points | AIC | Half_life | Lambda_z | Time_range |
|---|---|---|---|---|---|---|---|---|---|---|---|
| time_range | 1 | Rank 1 | 0.9998 | 0.9995 | 2.62 | PASS | 3 | -12.23 | 62.62 | 0.0111 | 125.2 - 289.4 |
| time_range1 | 2 | Rank 2 | 0.9963 | 0.9953 | 4.54 | PASS | 6 | -9.23 | 58.44 | 0.0119 | 24.1 - 289.4 |
| time_range2 | 3 | Rank 3 | 0.9965 | 0.9953 | 4.39 | PASS | 5 | -6.96 | 57.65 | 0.012 | 36 - 289.4 |
| time_range3 | 4 | Rank 4 | 0.9951 | 0.9941 | 4.83 | PASS | 7 | -9.49 | 57.38 | 0.0121 | 12 - 289.4 |
| time_range4 | 5 | Rank 5 | 0.9955 | 0.9933 | 2.53 | PASS | 4 | -6.88 | 53.89 | 0.0129 | 36 - 172.4 |
| time_range5 | 6 | Rank 6 | 0.9965 | 0.9929 | 1.77 | PASS | 3 | -6.05 | 50.33 | 0.0138 | 36 - 125.2 |
| time_range6 | 7 | Rank 7 | 0.9951 | 0.9927 | 4.16 | PASS | 4 | -3.56 | 57.97 | 0.012 | 48.1 - 289.4 |
| time_range7 | 8 | Rank 8 | 0.9929 | 0.9917 | 4.96 | PASS | 8 | -8.59 | 56.31 | 0.0123 | 10 - 289.4 |
| time_range8 | 9 | Rank 9 | 0.9918 | 0.9891 | 2.65 | PASS | 5 | -7.16 | 55.97 | 0.0124 | 24.1 - 172.4 |
| time_range9 | 10 | Rank 10 | 0.9906 | 0.9883 | 2.95 | PASS | 6 | -8.45 | 54.43 | 0.0127 | 12 - 172.4 |

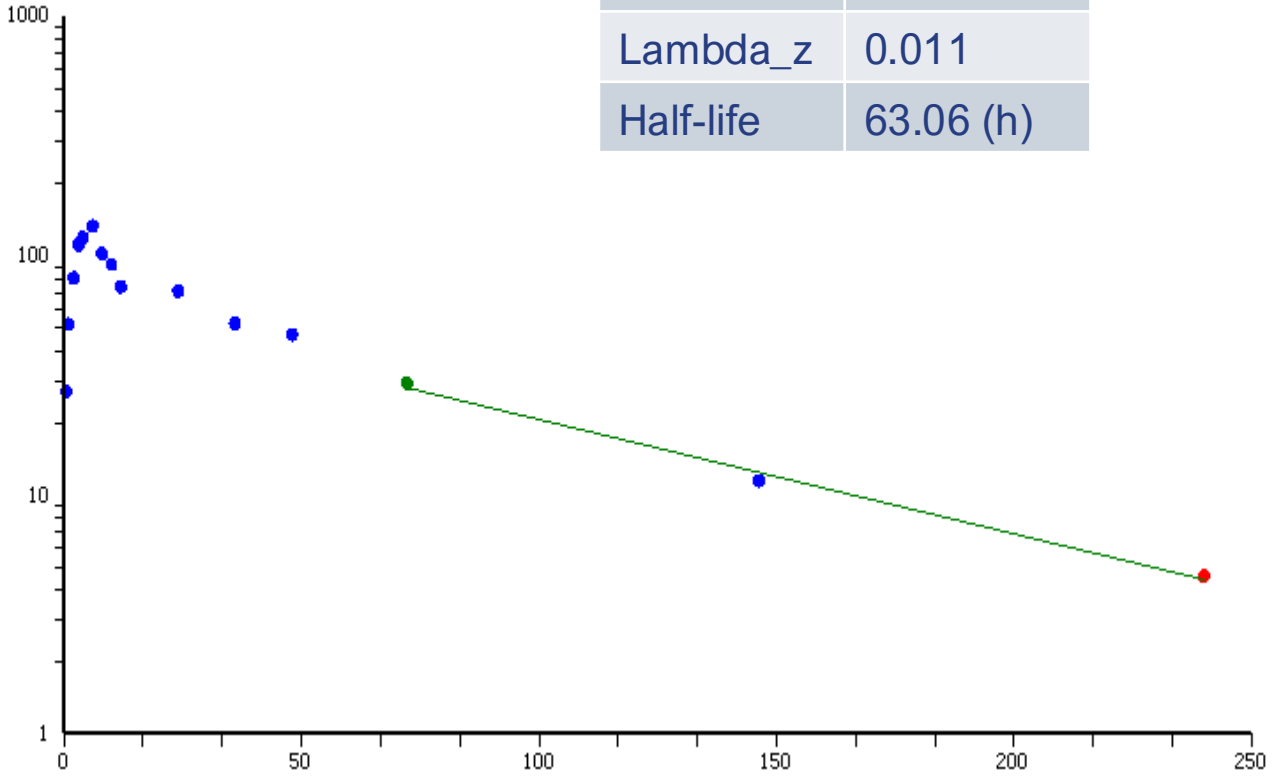
Showing 1 to 10 of 27 entries

Previous 1 2 3 Next

# WinNonlin vs PK_NCA_Optimizer

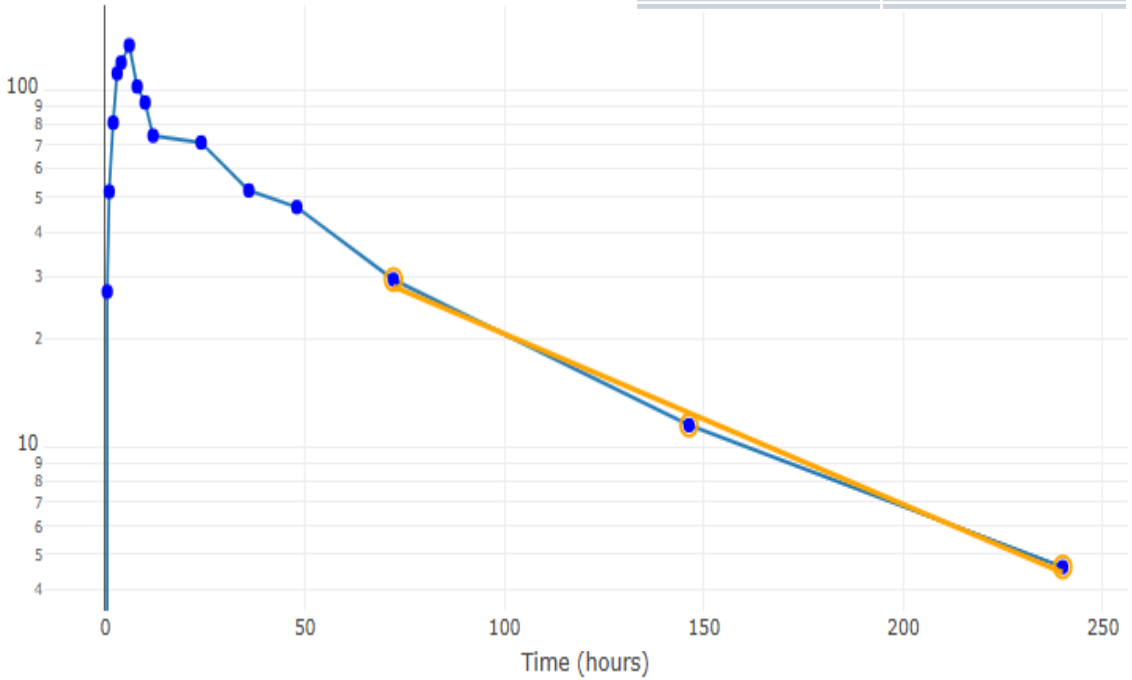
**WinNonlin**
Best Fit

| | |
|---|---|
| $R^2$ | 0.9888 |
| # of points | 3 |
| SPAN | 2.66 |
| Lambda_z | 0.011 |
| Half-life | 63.06 (h) |




**PK_NCA_Optimizer**
Best Fit

| | |
|---|---|
| $R^2$ | 0.9888 |
| # of points | 3 |
| SPAN | 2.66 |
| Lambda_z | 0.011 |
| Half-life | 63.06 (h) |



# Where AI excels (in its own words)

- Automates terminal slope selection, reducing manual effort
- Provides consistent and reproducible results across datasets
- Aligns with established PK principles (e.g., Phoenix WinNonlin)
- Integrates data processing and visualization in one workflow
- Extensive documentation
- Thorough error handling and debugging
- Very flexible

# The downsides

- Hallucination and confidence is a problem
- Sensitivity to data quality and terminal phase definition
- AI-generated code requires careful validation and review
- Hard to keep track of changes and additions
- Dependent on AI models and big tech

# Lessons learned

- Creating tools with AI is fun and impressive
- Prompting is (too?) easy, code gets overwhelming very quickly
- Generated code and documentation is impressive and extensive
- Hallucination is real
- Software development has never been this easy and quick
  - Tool is not used in practice yet

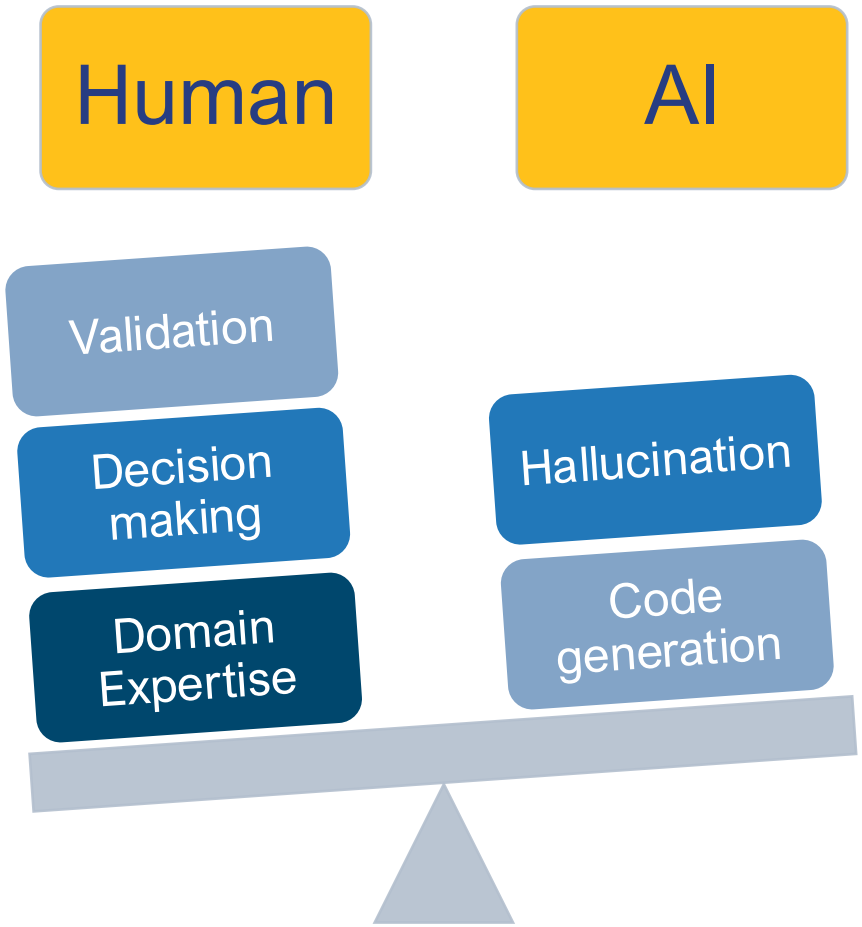
**But:**
**Is this a good or bad thing?**

# Key takeaways

- **Automated slope selection can improve consistency in NCA workflows**
- **AI can support development of specialized analytical tools, but has limitations**
- **Human expertise remains essential for validation and decision-making**

# Key takeaways

***"AI will not take over our jobs, but it will change how we do them"***



# Do you have any further questions?

OCS LIFE SCIENCES
A DIVISION OF OCS CONSULTING

## Thank you for your attention!


**Ramon Jakobs**

Statistical Programmer

ramon.jakobs@ocs-consulting.com

Ruwekampweg 2 g • 5222 AT 's-Hertogenbosch • The Netherlands • info.nl@ocs-consulting.com