Simple, Automatic Noncompartmental Analysis: The PKNCA R Package

William S. Denney1, Sridhar Duvvuri1, Clare Buckeridge1

1Clinical Pharmacology, Pfizer, Cambridge, Massachusetts

Objective: Noncompartmental analysis (NCA) of pharmacokinetic (PK) data has typically been the realm of specialized software that does not integrate easily with data workflows. The goal of this project was to build an NCA analysis engine in R that can perform required calculations, plotting, and summarization.

Methods: Standard methods for NCA were implemented with the algorithms in Gabrielsson and Weiner [1]. For each subject in each period, the tool automatically determines the NCA parameters appropriate for calculation based on concentration-time and dosing data.

The information used for automatic determination of the correct parameters include selections based on route of administration, single or multiple dosing, the dosing interval (if multiple dosing), and if steady-state has been achieved. The NCA calculated during any given interval including $C_{\text{max}}$, $T_{\text{max}}$, AUC and AUMC with linear or linear-up/log-down integration, $t_{1/2}$, $T_{ss}$ (time to steady-state) with either a monoexponential rise or the stepwise linear method [2], CL or CL/F, $V_{ss}$, $T_{\text{last}}$, $C_{\text{last}}$, and interpolated or extrapolated concentrations.

All settings, data cleaning, and summarization functions can be customized by the NCA analyst. Also, a NONMEM-ready dataset can be easily created after loading the data.

Results: The PKNCA R package has been integrated into an internal tool used for rapid, automated data analysis of Phase 1 studies, AMP (Automated Monitoring of Phase 1). Calculation results are identical when compared with validated internal and external software. The integration of NCA calculations into the dose-escalation study monitoring workflow has improved speed and reproducibility of the data for review.

Conclusions: The PKNCA R package was developed and will be made available to the public on CRAN. The package targets 100% test coverage of the code, and as open source in R is easily integrated in almost any workflow.

References: