Building Automated Pharmacometrics Analysis Workflows in R with NMsim

Boris Grinshpun[1], Samer Mouksassi[1], Philip Delff[2]

[1] Certara USA, Radnor, PA

[2] Vertex Pharmaceuticals Incorporated, Boston, MA, USA

Introduction

The new R package `NMsim` provides the capability to perform NONMEM simulations directly from R, without need of model reimplementation. This functionality allows a pharmacometrics simulation to be performed without having to transcribe and validate models using different software platforms and modeling syntax. As a result, efficient automated workflows can be developed to perform simulation based analyses and visualize key results.

The following automated pharmacometrics simulation workflows are described:

- Covariate effects analysis: Multiple simulations are performed with parameter uncertainty using `NMsim` and visualized as a forest plot using the `coveffectsplot` package.
- Visual predictive check: Estimation data is reused for simulation using `NMsim` and VPC plots are produced via the `tidyvpc` R package.

Workflow

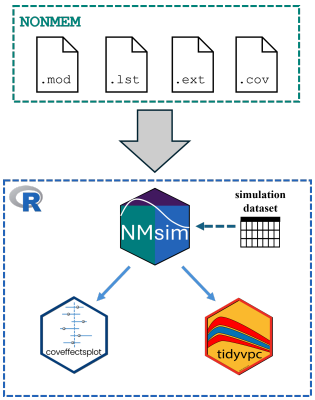

Figure 1: Schematic of NMsim Driven Analysis Workflow

Pre-requisite: A successful NONMEM model run with output files.
1. Generate a simulation NONMEM dataset.
2. Simulate from the NONMEM model – simulation type will depend on analysis.
3. Postprocess simulation outputs for compatibility with other analysis packages.
4. Run `coveffectsplot` and/or `tidyvpc`.

NMsim Functions

`NMsim` includes built in functions to quickly generate a simulation dataset.

NMsim: Simulates from a NONMEM model.
NMcreateDoses: Generates dosing records.
addEVID2: Add a sampling scheme to dosing records.
expandCovs: Generate simulation data set with quantiles and labels for one covariate at a time, keeping others at reference. Suitable for a forest plot simulation.

Supplementary Function – Scan the QR code for details.
forestSummarize: Prepare forest plot ready input data frame

Optional Function
NMscanData: Automatically find and read in NONMEM input dataset (from the `NMdata` package)

NMsim Set Up

```
library(NMdata) # version 0.1.7
library(NMsim) # version 0.1.3.941

NMdataConf(as.fun="data.table" # data.table outputs
           ,path.nonmem="C:/nm75g64/run/nmfe75.bat" # NONMEM path (Windows OS)
           ,dir.sims="simtmp" # location of sim tmp files
           ,dir.res="simres" # location of sim results
)

file.mod = "models/xgxr134.mod" # example dataset
```

Forest Plot Automation

Preparing the Simulation Dataset

```
res <- NMdata::NMscanData(file.mod, quiet=T) # read in NONMEM dataset (file.mod)
dosing = NMsim::NMcreateDoses(# create a dosing table
  TIME=0, AMT=100, addl=list(ADDL=29,II=12),CMT=1)

dt.covs <- NMsim::expandCovs(
   WEIGHTB=list(ref=70,values=c(40,60,80,100),label="Bodyweight (kg)")
  ,AGE=list(ref=median,quantiles=c(10,25,75,90)/100,label="Age (years)")
  ,MALEN=list(ref=c(Female=0),values=c(Male = 1, Female = 0),label = "Sex")
  ,data=res)

# combine simulation dataset and covariates
dosing <- dosing[,dt.covs[],by=dosing]
dosing[,ID:=.GRP,by=.(type,covvar,covval)]

# add a sampling scheme
time.sim <- data.table::data.table(TIME=c(seq(0,24,1), seq(336,360,1)))
time.sim[TIME<=24,period:="Day 1"]
time.sim[TIME>=336&TIME<=360,period:="Steady-State"]
dt.sim <- NMsim::addEVID2(dosing,time.sim=time.sim,CMT=1)
```

Simulating with Parameter Uncertainty

The method `NMsim_VarCov` is used to sample from the variance-covariance matrix defined in the .cov nonmem output file.

```
simres.forest <- NMsim::NMsim(file.mod # path to NONMEM model
                ,data=dt.sim, # simulation dataset
                ,name.sim="pars_covars_varcov" # output name suffix
                ,method.sim=NMsim_VarCov # sampling method
                ,typical=TRUE # FALSE to include BSV
                ,table.vars=c(PRED,IPRED) # output table variables
                ,method.update.inits="nmsim" # update parameters from ext
                ,nsims=250 # number of simulations
                ,seed.R=342 # seed for reproducibility
                ,sge=FALSE # TRUE if submitting to a cluster
                ,reuse.results=TRUE # TRUE to load existing results
                ,quiet = TRUE) # FALSE to view all output messages
```

Summarizing Exposures

The **forestSummarize** function can generate exposure metrics stratified by covariate groups, and summarize them in a table compatible with the `coveffectsplot` package.

```
coveff.df <- forestSummarize(
                data = simres.forest, # NMsim output
                metrics=c("AUC","Cmax"), #"AUC", "Cmax", "Cmin"
                period="Steady-State") # defined during dataset construction
```

period	paramname	covname	label	lower	mid	upper	MEANVAL	LOWCI	UPCI	LABEL
Steady-State	Cmax	Age (years)	34	0.90	1.10	1.4	1.10	0.90	1.4	1.1 [0.9-1.4]
Steady-State	AUC	Age (years)	34	0.80	1.20	1.8	1.20	0.80	1.8	1.2 [0.8-1.8]
Steady-State	Cmax	Age (years)	45	0.86	1.00	1.3	1.00	0.86	1.3	1 [0.86-1.3]
Steady-State	AUC	Age (years)	45	0.75	1.10	1.6	1.10	0.75	1.6	1.1 [0.75-1.6]
Steady-State	Cmax	Age (years)	65	0.82	0.97	1.2	0.97	0.82	1.2	0.97 [0.82-1.2]
Steady-State	AUC	Age (years)	65	0.66	0.94	1.4	0.94	0.66	1.4	0.94 [0.66-1.4]
Steady-State	Cmax	Age (years)	73	0.80	0.95	1.2	0.95	0.80	1.2	0.95 [0.8-1.2]
Steady-State	AUC	Age (years)	73	0.62	0.90	1.3	0.90	0.62	1.3	0.9 [0.62-1.3]
Steady-State	Cmax	Sex	F	0.84	1.00	1.3	1.00	0.84	1.3	1 [0.84-1.3]
Steady-State	AUC	Sex	F	0.71	1.00	1.5	1.00	0.71	1.5	1 [0.71-1.5]

Constructing the Forest Plot

Forest plots are constructed using the forest_plot function within `coveffectsplot`.

```
fp <- coveffectsplot::forest_plot(data = coveff.df # input data frame
  ,facet_formula = "covname ~ paramname" # stratification
  ,facet_scales = "free_y"
  ,facet_space = "free_y"
  ,table_text_size = 5
  ,plot_table_ratio = 1.5
  ,x_label_text_size = 14
  ,y_label_text_size = 14
  ,table_title_size = 16L
  ,base_size=14
  ,strip_placement = "outside"
  ,legend_order = c("pointinterval", "ref", "area")
  ,x_range = c(.2,1.8))
```

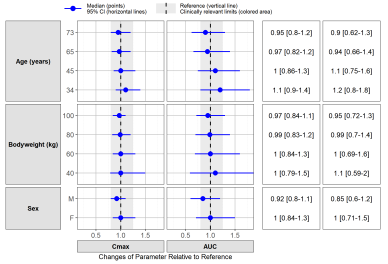

Figure 2: Forest plot generated using simulations from NMsim

Visual Predictive Check Automation

Re-using Estimation Data For Simulation

If the data argument is not provided `NMsim` will re-use estimation data.

```
simres.vpc <- NMsim::NMsim(file.mod # path to NONMEM model,
                ,table.vars=c("PRED","IPRED", "Y") # output table variables
                ,name.sim="vpc_result", # output name suffix
                ,subproblems=250 # SUBPROBLEMS argument to $SIMULATION block
                ,seed.R = 342 # seed for reproducibility
                ,sge=FALSE # TRUE if submitting to a cluster
                ,reuse.results=TRUE # TRUE to load existing results
                ,quiet = TRUE) # FALSE to view all output messages
```

Assessing Model Predictions With a VPC plot

Visual predictive checks can be generated using the `tidyvpc` package.

```
# read in NONMEM input data
res <- NMdata::NMscanData(file.mod,as.fun="data.table",quiet=TRUE)
res$DOSEC = factor(res$DOSE, levels = res$DOSE,
                   labels = paste(res$DOSE, "mg")) # for labeling purposes
res.obs <- subset(res,EVID==0) # keep only observations

# keep only observations from simulated data
simres.vpc  <- subset(simres.vpc,EVID==0)

# check that this ratio matches the number of subproblems
nrow(simres.vpc)/nrow(res.obs)
#> [1] 250

# generate the vpc
vpc <-
  tidyvpc::observed(res.obs, x = TIME, y = DV) |>
  tidyvpc::simulated(simres.vpc, y = Y) |>
  tidyvpc::stratify(~DOSEC) |>
  tidyvpc::binning(bin = "ntile", nbins = 9) |>
  tidyvpc::vpcstats()
```

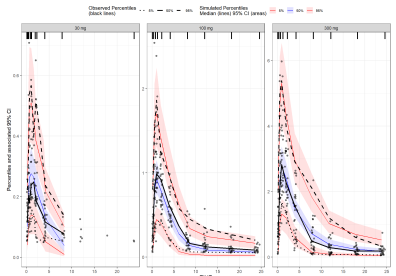

Figure 3: Visual predictive check generated using simulations from NMsim

Conclusion

By combining the simulation capabilities of NMsim with other pharmacometrics packages in R, it is possible to build automated analysis pipelines with minimal code. We hope these examples will aid pharmacometricians in producing quick, clean, and efficient workflows, and reduce coding burden.

See Also

See the NMsim website for vignettes and news.

Related posters at ACoP 2024

- NMsim - Seamless NONMEM Simulation Platform in R (T32)
- Simulation of clinical trial predictions with model uncertainty using NMsim (T110)
- Simulate modified NONMEM models using NMsim (T19)
- A Model-Based Simulation Workflow Enables Automated and Accurate Generation of Clinical Pharmacology Summary Statistics (T103)

References

[1] Delff P. 2024. *NMsim: Seamless 'Nonmem' Simulation Platform*. https://philipdelff.github.io/NMsim/

[2] Marier J-F, Teuscher N, Mouksassi M-S. Evaluation of covariate effects using forest plots and introduction to the coveffectsplot R package. CPT Pharmacometrics Syst Pharmacol. 2022; 11: 1283-1293.

[3] Barriere O, Rich B, Craig J, Mouksassi S (2024). *tidyvpc: VPC Percentiles and Prediction Intervals*. https://github.com/certara/tidyvpc