simpar: an R Package for Parameter Uncertainty Simulations in Pharmacometric Modeling

Shen Cheng[1], Seth Green[1], Ryan Marinelli[1], Kyle Baron[1]
[1]*Metrum Research Group, Tariffville, CT, USA*

Introductions

- Parameter uncertainty is frequently included in simulations to support model-based decision-making in pharmacometrics.
- The recently-deprecated *metrumrg* package for R previously included functionality for simulating both fixed effect and random effect parameters.
- The objective of this work was to spin off a new R package (*simpar*) that preserved this important functionality and extended the set of features to better support incorporating parameter uncertainty in pharmacometric simulations.

Methods

- *simpar* is a freely available open-source R package available on GitHub [1] for the simulation of parameter uncertainty in pharmacometric simulations.
- Development of the *simpar* package started with the relevant functionalities in the now-deprecated R package, *metrumrg*.
- New features were incorporated to expand the applicability of the package.

Results A

Parameter Uncertainty Simulations in *simpar*

- Given a set of parameter estimates and the associated variance-covariance matrix output by a pharmacometric or statistical model (i.e., a mixed effects model), *simpar* allows users to:
  - Sample fixed effect parameters (*THETA*) assuming a multivariate normal distribution.
  - Sample interindividual variability random-effect (*OMEGA*) matrix and residual variability random-effect (*SIGMA*) matrix assuming inverse Wishart distributions if the matrix dimension is 2×2 or more.
  - Sample interindividual variability random-effect (*OMEGA*) matrix and residual variability random-effect (*SIGMA*) matrix assuming inverse chi-square distributions if the matrix dimension is 1×1.
- New functionalities were developed to expand the traditional behavior of *simpar* (**Results C**).

Figure 1: Example *simpar* input.

```
> th
       THETA1  THETA2  THETA3  THETA4  THETA5   THETA6     THETA7   THETA8
[1,] 0.419241 4.13054 1.18047 4.20429 1.25826 0.514715 -0.0439246 0.534128
> covar
              THETA1       THETA2       THETA3       THETA4       THETA5       THETA6       THETA7       THETA8
THETA1  3.25470e-03  8.87769e-04  6.73136e-04 -2.46387e-04 -8.13271e-04  5.44303e-05  0.000334468 -8.64523e-05
THETA2  8.87769e-04  7.22709e-04  4.70059e-04 -9.42680e-05 -3.57934e-04  2.23537e-04  0.000333134 -4.25997e-05
THETA3  6.73136e-04  4.70059e-04  9.11747e-04  4.01476e-05 -1.04550e-04  4.94790e-04  0.000848500  9.49418e-04
THETA4 -2.46387e-04 -9.42680e-05  4.01476e-05  4.00267e-04  1.03630e-04  8.41551e-05  0.000176186  2.60625e-04
THETA5 -8.13271e-04 -3.57934e-04 -1.04550e-04  1.03630e-04  1.07653e-03  1.73405e-05 -0.000246674 -3.72187e-04
THETA6  5.44303e-05  2.23537e-04  4.94790e-04  8.41551e-05  1.73405e-05  2.46656e-03 -0.000187453  7.63530e-04
THETA7  3.34468e-04  3.33134e-04  8.48500e-04  1.76186e-04 -2.46674e-04 -1.87453e-04  0.008706680 -2.39090e-03
THETA8 -8.64523e-05 -4.25997e-05  9.49418e-04  2.60625e-04 -3.72187e-04  7.63530e-04 -0.002390900  1.63798e-02
> om
[[1]]
          [,1]      [,2]     [,3]
[1,] 0.155197 0.0000000 0.000000
[2,] 0.000000 0.0693031 0.000000
[3,] 0.000000 0.0000000 0.112695

> sg
[[1]]
           [,1]    [,2]
[1,] 0.0399988 0.00000
[2,] 0.0000000 1.05918
```

- **th**: vector of fixed effect parameter estimates.
- **covar**: fixed effect parameters variance-covariance matrix.
- **om**: first level random effects variance-covariance matrices.
- **sg**: second level random effects variance-covariance matrices.

Figure 2: Example code for parameter uncertainty simulations using *simpar*.

```
set.seed(12345)
uc <- simpar(
  nsim = 1000,
  theta = th,
  covar = covar,
  omega = om,
  odf = 200, # >= nid
  sigma = sg,
  sdf = 4000 # >= nobs
) %>% as.data.frame()
```

- **nsim**: scalar numeric specifying the number of sets to attempt.
- **theta**: *vector* of point estimates of fixed effect parameters.
- **covar**: variance-covariance *matrix* for fixed effect parameters.
- **omega**: list of variance-covariance *matrices* for first level random effects.
- **odf**: vector of omega degrees of freedom, one per matrix (typically $\geq$ number of individuals in the data set).
- **sigma**: list of variance-covariance *matrices* for second level random effects.
- **sdf**: vector of sigma degrees of freedom, one per matrix (typically $\geq$ number of observations in the data set).

Figure 3: Example *simpar* output. Typical output is a **data frame**, with column names indicating parameters, and row names indicating set number.

```
> head(uc)
    TH.1  TH.2  TH.3  TH.4  TH.5   TH.6     TH.7   TH.8  OM1.1     OM2.1   OM2.2     OM3.1     OM3.2  OM3.3   SG1.1     SG2.1 SG2.2
1 0.3976 4.135 1.224 4.206 1.287 0.5255  0.12360 0.4989 0.1340  0.014310 0.06696 -0.010140  0.004318 0.12190 0.03976  0.002024 1.036
2 0.4964 4.130 1.172 4.175 1.248 0.4900 -0.01883 0.4472 0.1782  0.002769 0.06239  0.004625  0.006629 0.11470 0.04122 -0.003349 1.061
3 0.3829 4.123 1.144 4.190 1.272 0.4926 -0.11370 0.5324 0.1722  0.011320 0.07087 -0.004886 -0.015070 0.12050 0.03987  0.002034 1.024
4 0.4579 4.175 1.222 4.215 1.251 0.5115 -0.14270 0.5670 0.1513 -0.001885 0.05289  0.009952 -0.000877 0.09929 0.03874  0.000607 1.014
5 0.4731 4.179 1.238 4.182 1.265 0.5620 -0.07351 0.4392 0.1504 -0.004950 0.07703  0.006071  0.007768 0.12620 0.04052  0.000872 1.046
6 0.3898 4.127 1.170 4.182 1.269 0.5714 -0.27090 0.7159 0.1600 -0.011500 0.07894  0.017300 -0.012980 0.13440 0.03999  0.001545 1.045
```

Results B

Pharmacometric Simulations Incorporating Parameter Uncertainty

- Users can request simulated outputs in multiple formats, including:
  - *format* = "*df*": Default. The traditional R data frame output with rows defining the sample index and columns defining the parameter names.
  - *format* = "*list*": An R list output containing simulated *THETA* vectors, and *OMEGA* and *SIGMA* matrices, which are structured for use directly in simulations with *mrgsolve*.

Figure 4: Simulations incorporating parameter uncertainties using *simpar* data frame (df) output.

```
set.seed(12345)
uc <- simpar(
  nsim = 1000,
  theta = th,
  covar = covar,
  omega = om,
  odf = 200,
  sigma = sg,
  sdf = 4000
) %>% as.data.frame()

names(uc) <- gsub("[[:punct:]]", "", names(uc))
names(uc) <- gsub("TH", "THETA", names(uc))
names(uc) <- gsub("OM", "OMEGA", names(uc))
names(uc) <- gsub("SG", "SIGMA", names(uc))
```

```
   THETA1 THETA2 THETA3 THETA4 THETA5 THETA6   THETA7 THETA8
1  0.4820  4.172  1.155  4.247  1.215 0.4278 -0.04554 0.3078
2  0.4692  4.166  1.184  4.211  1.214 0.5185  0.18080 0.2899
3  0.4058  4.127  1.189  4.183  1.275 0.5303 -0.10120 0.7277
4  0.4445  4.142  1.191  4.169  1.259 0.5346 -0.01400 0.5226
5  0.4444  4.162  1.194  4.198  1.230 0.4411 -0.02781 0.4798
6  0.3536  4.116  1.169  4.205  1.305 0.4324  0.03610 0.5018
7  0.4568  4.136  1.200  4.228  1.259 0.4743 -0.02697 0.6101
8  0.3944  4.140  1.205  4.248  1.269 0.4971  0.08313 0.5241
9  0.3439  4.118  1.188  4.221  1.274 0.4569 -0.13650 0.7177
10 0.4235  4.121  1.198  4.180  1.237 0.4757 -0.07307 0.5833
11 0.4572  4.144  1.208  4.222  1.184 0.4793  0.09039 0.3958
12 0.4293  4.129  1.181  4.216  1.211 0.5583 -0.09148 0.3778
13 0.3136  4.145  1.143  4.194  1.298 0.5646 -0.11680 0.5632
14 0.4228  4.160  1.207  4.210  1.223 0.5723 -0.01580 0.4603
15 0.4583  4.147  1.208  4.198  1.222 0.5014 -0.16640 0.6578
16 0.3786  4.141  1.165  4.184  1.219 0.4006 -0.04433 0.5691
```

```
simfunc <- function(i){

  mod <- param(mod, uc[i,])
  mod <- omat(mod, as_bmat(uc[i,], "OMEGA"))
  mod <- smat(mod, as_bmat(uc[i,], "SIGMA"))

  mrgsim(mod,
         data = simpop,
         output = "df",
         recover = "DOSE",
         obsonly = TRUE,
         quiet  = TRUE) %>%
    mutate(isim = i)
}


set.seed(12315)
sims <- lapply(1:1000, simfunc) %>% bind_rows()
```

Figure 5: Simulations incorporating parameter uncertainties using *simpar* list output.

```
set.seed(12345)
uc <- simpar(
  nsim = 1000,
  theta = th,
  covar = covar,
  omega = om,
  odf = 200,
  sigma = sg,
  sdf = 4000,
  format = "list"
)
```

```
$param
  THETA1 THETA2 THETA3 THETA4 THETA5 THETA6 THETA7 THETA8
1 0.3976  4.135  1.224  4.206  1.287 0.5255 0.1236 0.4989

$omega
$omega[[1]]
         [,1]     [,2]      [,3]
[1,]  0.13400 0.014310 -0.010140
[2,]  0.01431 0.066960  0.004318
[3,] -0.01014 0.004318  0.121900


$sigma
$sigma[[1]]
         [,1]     [,2]
[1,] 0.039760 0.002024
[2,] 0.002024 1.036000
```

```
simfunc <- function(i){

  mod <- update(mod, data = uc[[i]])

  mrgsim(mod,
         data = simpop,
         output = "df",
         recover = "DOSE",
         obsonly = TRUE,
         quiet  = TRUE) %>%
    mutate(isim = i)
}


set.seed(12315)
sims <- lapply(1:1000, simfunc) %>% bind_rows()
```

Results C

Simulate Diagonal Matrices in *simpar*

- Users can request simulated off-diagonal *OMEGA* and *SIGMA* elements be fixed to zero when they are zero in the input matrices (diagonal matrices).
- In this case, on-diagonal *OMEGA* or *SIGMA* elements are simulated as a series of independent inverse chi-squared simulations regardless of dimensionality.

Figure 6: Simulate Block Matrices.

```
set.seed(12345)
uc1 <- simpar(
  nsim = 1000,
  theta = th,
  covar = covar,
  omega = om,
  odf = 200,
  sigma = sg,
  sdf = 4000,
  format = "list"
)
```

Input/Output

```
> uc1[[1]]$omega
[[1]]
         [,1]     [,2]      [,3]
[1,]  0.13400 0.014310 -0.010140
[2,]  0.01431 0.066960  0.004318
[3,] -0.01014 0.004318  0.121900

> uc1[[1]]$sigma
[[1]]
         [,1]     [,2]
[1,] 0.039760 0.002024
[2,] 0.002024 1.036000
```

Figure 7: Simulate Diagonal *OMEGA* Matrices.

```
set.seed(12345)
uc2 <- simpar(
  nsim = 1000,
  theta = th,
  covar = covar,
  omega = om,
  odf = 200,
  sigma = sg,
  sdf = 4000,
  format = "list",
  omega_diag = TRUE
)
```

Input/Output

```
> uc2[[1]]$omega
[[1]]
         [,1]    [,2]   [,3]
[1,] 0.1299 0.00000 0.0000
[2,] 0.0000 0.06663 0.0000
[3,] 0.0000 0.00000 0.1181

> uc2[[1]]$sigma
[[1]]
         [,1]     [,2]
[1,] 0.041060 0.001717
[2,] 0.001717 1.012000
```

Figure 8: Simulate Diagonal *OMEGA* and *SIGMA* Matrices.

```
set.seed(12345)
uc3 <- simpar(
  nsim = 1000,
  theta = th,
  covar = covar,
  omega = om,
  odf = 200,
  sigma = sg,
  sdf = 4000,
  format = "list",
  omega_diag = TRUE,
  sigma_diag = TRUE
)
```

Input/Output

```
> uc3[[1]]$omega
[[1]]
         [,1]    [,2]   [,3]
[1,] 0.1299 0.00000 0.0000
[2,] 0.0000 0.06663 0.0000
[3,] 0.0000 0.00000 0.1181

> uc3[[1]]$sigma
[[1]]
         [,1]  [,2]
[1,] 0.04106 0.000
[2,] 0.00000 1.085
```

Conclusions

simpar R package provides a flexible tool for users to perform pharmacometric simulations with parameter uncertainty.

References

[1] simpar on Github. `https://github.com/metrumresearchgroup/simpar`. Accessed: 2023-10-16.

QR code

Presented at the American Conference on Pharmacometrics; 5 November - 8 November 2023