R-based reproduction of the estimation process hidden behind NONMEM® Part 2: First-order conditional estimation

Kyun-Seop Bae[1*] and Dong-Seok Yim[2]

[1]Asan Medical Center, University of Ulsan College of Medicine, Seoul 05505, Korea, [2]PIPET (Pharmacometrics Institute for Practical Education and Training), College of Medicine, The Catholic University of Korea, Seoul 06591, Korea.
*Correspondence: K.S. Bae; Tel: +82-2-3010-4611, Fax: +82-2-3010-4623, E-mail: ksbae@amc.seoul.kr

The first-order conditional estimation (FOCE) method is more complex than the first-order (FO) approximation method because it estimates the empirical Bayes estimate (EBE) for each iteration. By contrast, it is a further approximation of the Laplacian (LAPL) method, which uses second-order expansion terms. FOCE without INTERACTION can only be used for an additive error model, while FOCE with INTERACTION (FOCEI) can be used for any error model. The formula for FOCE without INTERACTION can be derived directly from the extension of the FO method, while the FOCE with INTERACTION method is a slight simplification of the LAPL method. Detailed formulas and R scripts are presented here for the reproduction of objective function values by NONMEM.

Introduction

The first-order conditional estimation (FOCE) method estimates the empirical Bayes estimate (EBE) at each step. Therefore, one iteration is almost the same as the first-order (FO) approximation method with the POSTHOC option, which produces EBE after finalization of parameter estimation.[1] The term 'parameter' in this tutorial means either theta (θ), omega (Ω), or sigma (Σ). Actually, NONMEM combines the theta vector and vech (half-vectorization)-operated 'Cholesky-like' decomposed omega and sigma matrix into one vector, and it differentiates simultaneously for the quasi-Newton-type minimization algorithm.[2-4] 'Cholesky-like' means that it is similar to a Cholesky decomposition, but not exactly the same, because it includes a kind of scaling for this unconstrained minimization algorithm. NONMEM 7.3 and R 3.3.1 were used for their calculation in this tutorial. Derivation of the formula and scripts for the FO method are explained in the previous article of this tutorial series.[5]

Essentially, there are two ways to derive FOCE: one is simply an expansion of the FO approximation method,[6] and the other is a further approximation of the LAPL approximation.

[7] If we want to consider the interaction between the random variables eta (η) and epsilon (ε), we need to use a second-order approximation of the Taylor series expansion or Laplacian approximation. The Taylor series expansion is adequate for the approximation of the prediction function (F), while the Laplacian approximation is suitable for the approximation of the objective function, which is usually in the form of an exponential function. Calculation of the objective function value using the second-order approximation of the prediction function (F) is too complex. Therefore, using the Laplacian approximation of likelihood (objective) function is much easier when considering the interaction of the random variables eta and epsilon, because the Laplacian method is the second-order approximation of an exponential function, such as the likelihood objective function.

FOCE without INTERACTION

The FO method can be regarded as a kind of maximum likelihood estimation (MLE) with the Taylor series expansion of the 'observation' (or prediction) function.[8,9] Observation function, or the vector of observation values will be noted here as 'Y', and the prediction function, or the vector of prediction values as 'F'. If we know the mean and variance of the observation function (Y), we can plug those into the MLE formula of normal distribution in the following:

$$L = \frac{1}{\sqrt{2\pi V(Y)}} e^{-\frac{1}{2V(Y)}(Y-E(Y))^2} \qquad \text{eq. 1}$$

where $E(Y)$ is the mean of observation, and $V(Y)$ is the variance of observation.

For the FO method, we can use $E(Y)$ and $V(Y)$ as follows:

$$E(Y) = F_0 = PRED \text{ (Typical Prediction)} \qquad \text{eq. 2}$$

$$V(Y) = G_0 \Omega G_0^T + \text{diag}(H_0 \Sigma H_0^T) = C_0 \qquad \text{eq. 3}$$

where

$$F = f_1(\theta, \eta, x) \qquad F_0 = f_1(\theta, 0, x)$$

$$Y = f_2(F, \varepsilon)$$

$$G = \frac{\partial F}{\partial \eta} = g(\theta, \eta, x) \quad G_0 = g(\theta, 0, x)$$

$$H = \frac{\partial Y}{\partial \varepsilon} = h(F, \varepsilon) \qquad H_0 = h(F_0, 0)$$

Here, it is assumed that eta and epsilon are independent random variables, i.e. COV(eta, epsilon) = 0, and there is no correlation among residuals. The assumption of no correlation among epsilons is not necessary, but it is natural for the nonlinear structural model combined with the heteroscedastic statistical model. If we observe correlation among residuals, it means that the structural and/or statistical models are not adequate and the modeling process needs to be continued. Sheiner and Beal discarded the assumption of normal distribution after the building of the objective function. They called this procedure the 'Extended Least Square (ELS)' method instead of the MLE method.[10-12]

Meanwhile, the FOCE method uses the following $E(Y)$ and $V(Y)$ for the MLE plug-in as follows:

$$E(Y) = F_1 - G_1 \eta_t = IPRED - G_1 \eta_t \qquad \text{eq. 4}$$

$$V(Y) = G_1 \Omega G_1^T + \text{diag}(H_0 \Sigma H_0^T) = C_x \qquad \text{eq. 5}$$

$$V(Y) = G_1 \Omega G_1^T + \text{diag}(H_1 \Sigma H_1^T) = C_1 \qquad \text{eq. 6}$$

where

$$F = f_1(\theta, \eta, x) \qquad F_1 = f_1(\theta, \eta_t, x)$$

$$Y = f_2(F, \varepsilon)$$

$$G = \frac{\partial F}{\partial \eta} = g(\theta, \eta, x) \quad G_1 = g(\theta, \eta_t, x)$$

$$H = \frac{\partial Y}{\partial \varepsilon} = h(F, \varepsilon) \qquad H_0 = h(F_0, 0) \qquad H_1 = h(F_1, 0)$$

F_0 is called the typical prediction (PRED) with EBE=0, while F_1 is called the individual prediction (IPRED or IPRE) with EBE. Subscript 0 denotes that 0 was used as a true eta value and

1 denotes that EBE was used as a true eta value (η_t). The function for Y, f_2 may contain theta (θ) in real situations, i.e. one can include thetas (θ) within the $ERROR clause of NONMEM. However, theta (θ) is not included in f_2 for the clarity. One may also use a Y function, such as Y=f($\theta,\eta,x,\varepsilon$). For the MLE plug-in, $V(Y)$ in equation 6 seems more natural but NONMEM® uses $V(Y)$ in equation 5 strangely.

Here, F is the predictive function, theta (θ) is a parameter (constant) vector within the predictive function, and eta (η) is a random variable vector indicating interindividual random variability with the mean of zero vector and the variance–covariance matrix of omega (Ω). Epsilon (ε) is also a random variable vector used to describe unexplainable residual variability between prediction (F) and observation (Y).

The primary work of NONMEM software is to estimate parameters (θ, Ω, and Σ), while the secondary work is to estimate the empirical Bayes estimate (EBE) of the random variable eta for each subject. Therefore, NONMEM has two objective functions to optimize: one for the parameters (θ, Ω and Σ), and the other for EBE. To estimate parameters (θ, Ω and Σ), EBEs are regarded as true values, while to estimate EBE, parameters (θ, Ω and Σ) are considered to be true. Parameters (θ, Ω and Σ) are constant for the entire population, while EBEs are constant for a certain subject or dosing occasion.

For the additive error model, H_1 is identical to H_0 as 1.

$$Y = F + \varepsilon \qquad \text{eq. 7}$$

$$H = \frac{\partial Y}{\partial \varepsilon} = 1 \qquad \text{eq. 8}$$

In the case of the following combined error model:

$$Y = F + F \cdot \varepsilon_1 + \varepsilon_2 \qquad \text{eq. 9}$$

$$\frac{\partial Y}{\partial \varepsilon_1} = F \qquad \text{eq. 10}$$

$$\frac{\partial Y}{\partial \varepsilon_2} = 1 \qquad \text{eq. 11}$$

thus, $H = (F, 1)$.

We can construct two types of residuals as follows:

$$R_0 = C_0^{-1/2}(Y - F_0) = WRES \qquad \text{eq.12}$$

$$R_1 = C_x^{-1/2}(Y - F_1) = CWRES \qquad \text{eq. 13}$$

$$R_1 = C_1^{-1/2}(Y - F_1) = CWRES \qquad \text{eq. 14}$$

R_0 is called the weighted residual (WRES) in NONMEM, which is distinct from the weighted residual in statistics textbooks. Therefore, the sign of WRES could be opposite to that of the residual, RES (= Y – F), in NONMEM. By contrast, R_1 in equation 13 is called a conditional weighted residual (CWRES).

However, using R_I in equation 14 seems more natural in authors' opinion. C_0, C_X and C1 are defined in equations 3, 5 and 6, respectively. The calculation of the matrix to the power of -0.5 is explained as the R function of mat.sqrt.inv() in the previous article [5].

If we use the variance (V(Y) or C), and the weighted residual (WRES or R) for the objective function, NONMEM's objective function is as follows:

$$O_i = log|C_i| + \vec{R}_i^T \vec{R}_i \qquad \text{eq. 15}$$

$$O = \sum_{i=1}^{n_i} O_i \qquad \text{eq. 16}$$

The total objective function value (O) is simply the summation of individual objective function values (O_i).

The reason that we use "G11 G21 G31 H11" (Fig. 1) for the extraction of G matrix and H matrix is that NONMEM (up to version 6.2) stores those values in the internal matrices as follows:

$$G = \begin{bmatrix} \dfrac{\partial F}{\partial \eta_1} & \dfrac{\partial^2 F}{\partial \eta_1 \partial \eta_1} & [\text{blank}] & [\text{blank}] \\ \dfrac{\partial F}{\partial \eta_2} & \dfrac{\partial^2 F}{\partial \eta_2 \partial \eta_1} & \dfrac{\partial^2 F}{\partial \eta_2 \partial \eta_2} & [\text{blank}] \\ \dfrac{\partial F}{\partial \eta_3} & \dfrac{\partial^2 F}{\partial \eta_3 \partial \eta_1} & \dfrac{\partial^2 F}{\partial \eta_3 \partial \eta_2} & \dfrac{\partial^2 F}{\partial \eta_3 \partial \eta_3} \end{bmatrix}$$

$$H = \begin{bmatrix} \dfrac{\partial Y}{\partial \varepsilon_1} & \dfrac{\partial^2 Y}{\partial \varepsilon_1 \partial \eta_1} & \dfrac{\partial^2 Y}{\partial \varepsilon_1 \partial \eta_2} & \dfrac{\partial^2 Y}{\partial \varepsilon_1 \partial \eta_3} \\ \dfrac{\partial Y}{\partial \varepsilon_2} & [\text{blank}] & [\text{blank}] & [\text{blank}] \end{bmatrix}$$

The second subscripts in "G11, G21, G31, H11" mean the column numbers of the internal storage matrix, while the first subscripts denote the row numbers, i.e. G21 means row 2, column 1 of the G matrix. NONMEM version 7.x also supports printing the first column elements, such as G11, G21, G31, H11, and H21, but not other elements. Usually, we need the values of the first column only. If you want to reduce the time needed to calculate unnecessary derivatives, you can use the "$ABBREVI-ATED DERIV2=NO" clause in the NONMEM control file. If you use the PREDPP library (ADVAN subroutines), G and H matrices are renamed as GG and HH, respectively.

A slight difference in objective function values between R (Fig. 2) and NONMEM may be caused while calculating many matrices. Matrix calculation results often differ among different software programs. The first seven digits were identical in our example.

```
$PROB  THEOPHYLLINE ORAL FILENAME =THEOFOCE
$INPUT ID AMT = DROP  TIME  DV BWT
$DATA  THEO
$PRED
   DOSE = 320
   V      = THETA(1) * EXP(ETA(1))
   K      = THETA(2) * EXP(ETA(2))
   KA     = THETA(3) * EXP(ETA(3))
   F      = DOSE / V  * KA / (KA − K) * (EXP(−K * TIME)−EXP(−KA * TIME))
   IPRE   = F
   Y      = F + EPS(1)

$THETA  (0, 50, 500) (0, 1, 10) (0, 5, 50)
$OMEGA BLOCK(3)
   0.2
   0.1      0.2
   0.1      0.1      0.2
$SIGMA  0.1
$EST METHOD = COND
$TAB  ID    TIME    IPRE  G11  G21  G31  H11  CWRES   FILE = sdtabTHEOFOCE
NOPRINT  ONEHEADER
```

Figure 1. NONMEM control file for FOCE without INTERACTION

```
setwd("D:/NM/THEO")  # Set this to your working directory
CtlName = "THEOFOCE" # Control file name without extension
EXT     = as.matrix(read.table(paste0(CtlName, ".ext"), skip=1, header=TRUE))
OM      = ltv2mat(EXT[2, c("OMEGA.1.1.", "OMEGA.2.1.", "OMEGA.2.2.",
          "OMEGA.3.1.", "OMEGA.3.2.", "OMEGA.3.3.")]) # Omega matrix
SG      = matrix(EXT[2, "SIGMA.1.1."], nrow=1, ncol=1) # Sigma matrix

EBE     = as.matrix(read.table(paste0(CtlName, ".phi"), skip=1, header=TRUE))
DATA    = as.matrix(read.table("sdtab", skip=1, header=TRUE))

IDs     = unique(DATA[,"ID"])
nID     = length(IDs)

CWRES   = vector()
OFVi    = cbind(IDs, rep(NA, nID))

for (i in 1:nID) {
  cID   = IDs[i]
  EBEi  = EBE[EBE[,"ID"] == cID, c("ETA.1.", "ETA.2.", "ETA.3.")]
  DATi  = DATA[DATA[,"ID"] == cID, ]
  Yi    = DATi[, "DV"]
  Fi    = DATi[, "IPRE"]
  Gi    = DATi[, c("G11", "G21", "G31")]
  Hi    = DATi[, "H11"]
  Ci    = Gi %*% OM %*% t(Gi) + diag(diag(Hi %*% SG %*% t(Hi)))
  Ri    = mat.sqrt.inv(Ci) %*% (Yi - Fi + Gi %*% EBEi)
  CWRES = append(CWRES, Ri)
  OFVi[i,2] = log(det(Ci)) + t(Ri) %*% Ri
}

sum(OFVi[,2]) # 103.4961 vs. 103.4947
all.equal(EBE[,"OBJ"], OFVi[,2]) # Mean relative difference: 2.673653e-05
all.equal(DATA[,"CWRES"], CWRES) # Mean relative difference: 4.893101e-05
```

Figure 2. R script for FOCE without INTERACTION

About SIGDIG in NONMEM

SIGDIG (significant digits) can be defined as follows:

$$\text{SIGDIG} = -\log_{10} \left| \frac{\theta - \theta_{true}}{\theta_{true}} \right| \qquad \text{eq. 17}$$

In a scaled and transformed parameter (STP, now it is renamed as UCP, unconstrained parameter) space, the standardization (dividing by θ_{true}) may not be very meaningful because it was already standardized in a certain way. NONMEM starts minimization with the value of 0.1 in UCP space.

If we calculate the following example, the process may be clearer. Notice that the values are in a real number domain, not in an integer domain.

$$-\log_{10} \left| \frac{10.001 - 10.000}{10.000} \right| = 4$$

However, we do not know the true value of parameters. Therefore, we need to use the following instead:

$$\widehat{\text{SIGDIG}} = -\log_{10} \left| \frac{\theta_i - \theta_{i-1}}{\theta_i} \right| \qquad \text{eq. 18}$$

Here, θ_i denotes the i-th iterated value during the minimization of objective function. During the minimization process, differences between adjacent iterated values become smaller. Therefore, SIGDIG increases as the iterations proceed. NONMEM calculates SIGDIG a little differently because it also con-

siders the gradients. Other software programs usually regards the minimization process as successful if differences in objective function values or parameter values between adjacent iterations are smaller than a specified amount. However, NONMEM includes one more criterion to judge minimization success or failure – the SIGDIG. Where the minimization process cannot reach the predefined SIGDIG criterion (Default is 3), it ends with a 'rounding error', which is considered as 'success' by other software. Because this SIGDIG is calculated in the scaled and transformed parameter (STP) space, there are usually more actual significant digits in the original scale. One simple explanation for this could be that the modeler specifies in the control file as constrained (bounded) optimization of which the maximum bound is (−1e6,1e6), but in the background optimization process, it uses unconstrained minimization of which the bound is (−inf, +inf). Therefore, even if one uses SIGDIG = 1, one usually obtains more significant digits in the original scale. This means that during the model building process, one does not need to place too much emphasis on SIGDIG, and it could even be reduced to 1. One more tip for covariate screening is that the OMEGA matrix could even be fixed during the covariate screening and this does not cause much difference in covariate selection.

FOCE with INTERACTION

To consider the interaction of eta and epsilon, we need to use at least a second-order approximation of observation function (Y) or likelihood function (accurate objective function).

The Taylor series expansion of observation function (Y) for the calculation of E(Y) and V(Y) is too complex. Because they are no longer normally distributed, we cannot use the MLE formula for normal distribution. Therefore, it is better to use the Laplacian approximation, which is the second-order Taylor series approximation of the likelihood (an exponential function).

$$O_i = \Phi_i + \hat{\eta}_i^T \Omega^{-1} \hat{\eta}_i + \log\left|\Omega\right| + \log\left|\Omega^{-1} + \frac{\hat{\Delta}_i}{2}\right| \qquad \text{eq. 19}$$

$$O_i = \Phi_i + \hat{\eta}_i^T \Omega^{-1} \hat{\eta}_i + \log\left|\Omega\right| + \log\left|\Omega^{-1} + G_i^T V^{-1} G_i\right| \qquad \text{eq. 20}$$

$$O_{\hat{\eta}_i} = \Phi_i + \hat{\eta}_i^T \Omega^{-1} \hat{\eta}_i \qquad \text{eq. 21}$$

where

$$\Phi_i = -2LL_i - n_i \log(2\pi) = \log\left|V_i\right| + (Y_i - F_i)^T V_i^{-1} (Y_i - F_i)$$

$$\hat{\Delta}_i = \left.\frac{\partial^2 \Phi_i}{\partial \eta^2}\right|_{\eta=\eta_t}$$

```
$PROB  THEOPHYLLINE ORAL
$INPUT ID AMT = DROP TIME DV BWT
$DATA  THEO
$PRED
   DOSE = 320
   V      = THETA(1) * EXP(ETA(1))
   K      = THETA(2) * EXP(ETA(2))
   KA     = THETA(3) * EXP(ETA(3))
   F      = DOSE / V  * KA / (KA − K) * (EXP(−K * TIME)−EXP(−KA * TIME))
   IPRE  = F
   Y      = F + F * EPS(1) + EPS(2)

$THETA (0, 50, 500) (0, 1, 10) (0, 5, 50)
$OMEGA BLOCK(3)
   0.2
   0.1     0.2
   0.1     0.1     0.2
$SIGMA  0.1     0.1
$EST METHOD = COND INTER
$TAB  ID  TIME  IPRE  G11 G21 G31 H11 H21 CWRES   FILE = sdtabTHEOFOCEI
NOPRINT  ONEHEADER
```

Figure 3. NONMEM control file for FOCE with INTERACTION method

```
    CtlName = "THEOFOCEI" # Control file name without extension
    EXT     = as.matrix(read.table(paste0(CtlName, ".ext"), skip=1, header=TRUE))
    OM      = ltv2mat(EXT[2, c("OMEGA.1.1.", "OMEGA.2.1.", "OMEGA.2.2.",
              "OMEGA.3.1.", "OMEGA.3.2.", "OMEGA.3.3.")]) # Omega matrix
    SG      = diag(EXT[2, c("SIGMA.1.1.", "SIGMA.2.2.")]) # Sigma matrix

    EBE     = as.matrix(read.table(paste0(CtlName, ".phi"), skip=1, header=TRUE))
    DATA    = as.matrix(read.table(paste0("sdtab",CtlName), skip=1, header=TRUE))

    IDs     = unique(DATA[,"ID"])
    nID     = length(IDs)

    Term4   = determinant(OM, logarithm=TRUE)$modulus[[1]]
    invOM   = solve(OM)

    CWRES   = vector()
    OFVi    = cbind(IDs, rep(NA, nID), rep(NA, nID))

    for (i in 1:nID) {
      cID     = IDs[i]
      DATi    = DATA[DATA[,"ID"]==cID, ]
      EBEi    = EBE[EBE[,"ID"]==cID, c("ETA.1.", "ETA.2.", "ETA.3.")]
      Yi      = DATi[,"DV"]
      Fi      = DATi[,"IPRE"]
      Gi      = DATi[, c("G11", "G21", "G31")]
      Hi      = DATi[, c("H11", "H21")]
      Vi      = diag(diag(Hi %*% SG %*% t(Hi)))
      invVi   = solve(Vi)

      Term1 = determinant(Vi, logarithm=TRUE)$modulus[[1]] # log(det(Vi))
      Term2 = t(Yi - Fi)%*% invVi %*%(Yi - Fi)
      Term3 = t(EBEi) %*% invOM %*% EBEi
    # Term4 = determinant(OM, logarithm=TRUE)$modulus[[1]]
      Term5 = log(det(invOM + t(Gi) %*% invVi %*% Gi))
      OFVi[i,2] = Term1 + Term2 + Term3 + Term4 + Term5

      Ci      = Gi %*% OM %*% t(Gi) + diag(diag(Hi %*% SG %*% t(Hi)))
      CWRESi  = mat.sqrt.inv(Ci) %*% (Yi - Fi + Gi %*% EBEi)
      OFVi[i,3] = log(det(Ci)) + t(CWRESi) %*% CWRESi
      CWRES   = append(CWRES, CWRESi)
    }

    CWRES
    OFVi

    sum(OFVi[,2]) # 92.22962 vs. 92.83087
    sum(OFVi[,3]) # 88.18747 using the formula and script in Fig. 2.
    all.equal(EBE[,"OBJ"], OFVi[,2]) # Mean relative difference: 0.006000912
    all.equal(EBE[,"OBJ"], OFVi[,3]) # Mean relative difference: 0.04634464
    all.equal(DATA[,"CWRES"], CWRES) # Mean relative difference: 4.541912e-05
```
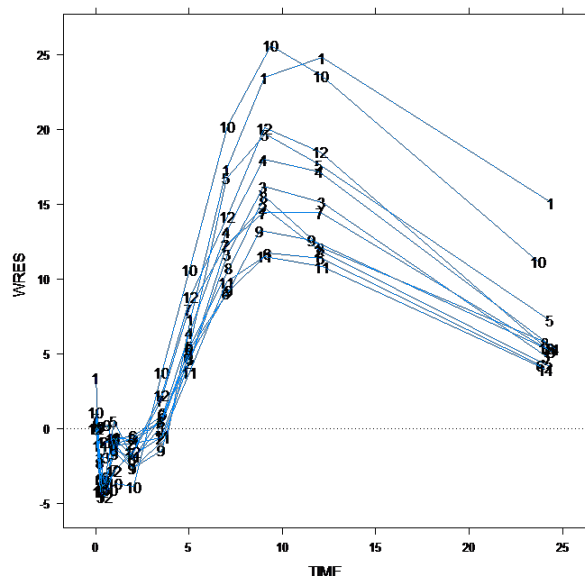
Figure 4. R script for FOCE with INTERACTION method

(a) FOCE without INTERACTION

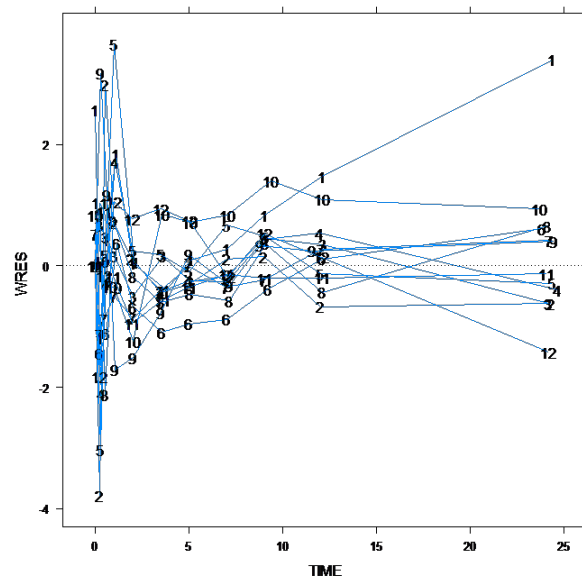(b) FOCE with INTERACTION

Figure 5. Comparison of WRES vs time plot in the proportional error model

The objective function for LAPL is equation 19.[6,7] Additionally, its approximation is equation 20, which can be used for the objective function of FOCE with or without INTERACTION. A more complete explanation of equation 20 will be made in the next article of this tutorial series.

Interestingly, the first two terms of equations 19 and 20 are the same as those in the objective function for EBE (eq. 21).

To show the calculation results of objective function values from the FOCE with INTERACTION method in NONMEM, the combined error model (epsilon 1 for proportional error and epsilon 2 for additive error) was chosen as shown in Figure 3.

The difference in objective function values between NONMEM and R (Fig. 4) becomes larger, which might be the result of a more complex calculation of matrices. One can also see larger differences in individual objective function values in OFV,[3] in the code. If one uses the INTERACTION option in the FOCE method in NONMEM and the combination of formulae in equations 1, 4, and 5 in R, one will get a different objective function value: 88.187 in R compared with 92.831 in NONMEM. This shows that NONMEM uses equation 20 for the FOCE with INTERACTION method, and not a direct extension of the FO method. A theoretical proof and numerical examples are provided in Wang's article.[7] The method of calculating CWRES in FOCE with INTERACTION is known to be the same as FOCE without INTERACTION method.

If one does not use the INTERACTION option when the error model is other than additive, a residual plot could show a peculiar trend as seen in Figure 5.

Conclusion

Mathematical formulae and R scripts for FOCE with or without INTERACTION are shown in this article. For an additive error model, we can use the FOCE without INTERACTION method, while for other error models, we should use the FOCE with INTERACTION method. The FOCE formula that was obtained by the approximation of LAPL usually gives results very similar to those using the LAPL method. Therefore, we recommend using FOCE with INTERACTION method as a workhorse in population modeling.

Acknowledgement

Conflict of Interests

The authors declare that there is no conflict of interests.

References

1. Kang D, Bae KS, Houk BE, Savic RM, Karlsson MO. Standard Error of Empirical Bayes Estimate in NONMEM® VI. Korean J Physiol Pharmacol 2012;16:97-106. doi: 10.4196/kjpp.2012.16.2.97.

2. Sheiner LB, Beal SL. Evaluation of methods for estimating population pharmacokinetics parameters. I. Michaelis-Menten model: routine clinical pharmacokinetic data. J Pharmacokinet Biopharm 1980;8:553-571.

3. Sheiner LB, Beal SL. Evaluation of methods for estimating population pharmacokinetic parameters. II. Biexponential model and experimental pharmacokinetic data. J Pharmacokinet Biopharm 1981;9:635-651.

4. Sheiner LB, Beal SL. Evaluation of methods for estimating population phar-

macokinetic parameters. III. Monoexponential model: routine clinical pharmacokinetic data. J Pharmacokinet Biopharm 1983;11:303-319.

5. Kim MG, Yim DS, Bae KS. R-based reproduction of the estimation process hidden behind NONMEM® Part 1: first-order approximation method. Transl Clin Pharmacol 2015;23:1-7. doi: 10.12793/tcp.2015.23.1.1.

6. Beal SL, Sheiner LB, Boeckmann AJ, Bauer R. NONMEM user's guides. Ellicott city, MD, USA 1989-2011.

7. Wang Y. Derivation of various NONMEM estimation methods. J Pharmacokinet Pharmacodyn 2007;34:575-593. doi: 10.1007/s10928-007-9060-6.

8. Beal S, Sheiner L. The NONMEM System. Am Stat 1980;34:118-119. doi: 10.2307/2684123.

9. Plan EL, Maloney A, Mentre F, Karlsson MO, Bertrand J. Performance comparison of various maximum likelihood nonlinear mixed-effects estimation methods for dose-response models. AAPS J 2012;14:420-432. doi: 10.1208/s12248-012-9349-2.

10. Peck CC, Beal SL, Sheiner LB, Nichols AI. Extended least squares nonlinear regression: a possible solution to the "choice of weights" problem in analysis of individual pharmacokinetic data. J Pharmacokinet Biopharm 1984;12:545-558.

11. Sheiner LB, Beal SL. Pharmacokinetic parameter estimates from several least squares procedures: superiority of extended least squares. J Pharmacokinet Biopharm 1985;13:185-201.

12. Peck CC, Sheiner LB, Nichols AI. The problem of choosing weights in nonlinear regression analysis of pharmacokinetic data. Drug Metab Rev 1984;15:133-148. doi: 10.3109/03602538409015060.