

The following bugs have been detected in NONMEM 7.4.1

- 1) When using the FAST option, and any Mu referenced thetas with OMEGA 0.0 FIXED are not at the end of the list of thetas to be estimated, then those thetas will not be moved during estimation. As a work-around, either move all thetas associated with OMEGA 0.0 FIXED to the end of the list of to-be-estimated thetas, or set OMEGA 1.0E-10 FIXED.
- 2) If no thetas are to be estimated, and only OMEGAS and/or SIGMAS are to be estimated, then an error in the gradient setup occurs, and the estimation fails. As a work-around use the SLOW option.
- 3) Because of inappropriate memory allocation, when applying the SIR algorithm, if the number of THETAS in the model is greater than the total number of parameters to be estimated (non-fixed thetas, omegas, and sigmas), an error can occur in assessing the variance from the SIR analysis. As a work-around, replace a sufficient number of the fixed thetas with variables assigned hard-coded values.
- 4) ADVAN14 and ADVAN15 can have memory allocation problems, or it may report that initial advance from t to tout is too small. To avoid the memory allocation problem, uncomment the following lines in subroutine FCVINIT of file cvodeu.f90, or FIDAINIT routine in idau.f90:

```
NBASE=NTOTAL  
NSENSE=0
```

To avoid the t, tout error message, modify your data set so that two closely spaced time events differ by a relative amount of at least 1.0E-04.

Updated CVODE.f90 and IDA.f90 that correct these problems are available in <https://nonmem.iconplc.com/nonmem741/extra/pr>

Download these files into the ..\pr directory of your installed NONMEM, and compile them as follows:

Linux gfortran compiler:

```
gfortran CVODE.f90 -c -O3 -ffast-math -I../resource -J../resource  
gfortran IDA.f90 -c -O3 -ffast-math -I../resource -J../resource
```

Windows gfortran compiler:

```
gfortran CVODE.f90 -c -O3 -ffast-math -I../resource -J../resource  
gfortran IDA.f90 -c -O3 -ffast-math -I../resource -J../resource
```

(or use your particular compiler options).

Linux intel fortran compiler:

```
ifort CVODE.f90 -c -Gs -Ob1ti -4Yportlib -prec-div -I../resource  
ifort IDA.f90 -c -Gs -Ob1ti -4Yportlib -prec-div -I../resource
```

Windows intel fortran compiler:

```
ifort CVODE.f90 -c -Gs -Ob1ti -4Yportlib -Qprec-div -I../resource
```

ifort IDA.f90 -c -Gs -Ob1ti -4Yportlib -Qprec-div -I..\resource

(or use your particular compiler options).

- 5) An access violation sometimes occurs if there are more than one SIGMA, and the first estimation method is IMP, SAEM, or BAYES. To avoid this, let the first estimation be

\$EST METHOD=ITS NITER=0

To cause the appropriate initializations to occur. This bug has occurred since NONMEM 7.2

- 6) For ITS, IMP, SAEM, BAYES, and NUTS problems, if a theta is MU modeled but fixed, ordinary differential equations (ODE) problems may run inefficiently because it evaluates an unneeded derivative for it. In such cases you may get speed improvement if you set \$EST MUM=N(x), where x is the theta number.

For ITS, MAP, SAEM, and BAYES, if the theta is MU modeled and estimated (not fixed), but has a 0 valued OMEGA associated with it, then the gradient is properly calculated. However by default, it will turn on analytical first derivative assessment for all thetas, which may become much less efficient than evaluating by finite difference for the few thetas that are needed, as was done in earlier NONMEM versions. When there are only a few thetas that have no OMEGAS or zero valued OMEGAS associated with them, it may be more efficient to force finite difference derivative evaluation for these few thetas, as was done in earlier versions of NONMEM. To do this, turn off the MU reference, with \$EST ... MUM=N(x), where x is the theta number, or list of theta numbers (such as MUM=N(5-8,12). For NUTS, you should not turn off MU referencing of estimated thetas with 0 OMEGAS, as analytical derivatives are needed for all thetas for this algorithm.

- 7) The NPD and NPDE diagnostics do not perform properly on BQL data. This error has occurred since NONMEM 7.3. There is no work-around.
- 8) During jobs that are parallelized, the CWRES and CWRESI values for some subjects may be those of WRES and WRESI. This is due to some of the worker processes not having available the empirical Bayes estimates for some subjects, necessary for evaluating CWRES and CWRESI. If you notice some of the subjects having considerable numbers of extreme values (>3.0 or <-3.0), it is best to have the table diagnostics executed in single CPU mode by setting the option PARAFILE on the first \$TABLE record to off:

\$TABLE ... PARAFILE=OFF

This will allow the other steps (\$EST, \$COV) to be still executed in parallel. Alternatively, you may set PARSE_TYPE in the pnm file to 1, although this can reduce the efficiency of the parallelization process for all steps.

- 9) Data files with records longer than 1000 characters may prevent NMTRAN from completing the data input. This error has been present since NONMEM 7.1. A work-around is to provide a FORMAT statement that matches the layout of the data file, such as:

\$DATA mydata.csv (100(1X,1PE11.4))

- 10) In defining the compartments in \$MODEL, if the first compartment is initially set to OFF, such as:

\$MODEL

COMP=(One,INITIALOFF,NODOSE)

COMP=(PKCENT)

NONMEM may issue an error statement:

0DATA REC 1: COMPARTMENT ASSOCIATED WITH THE PREDICTION
IS OFF

The remedy is to specify the default observation and dose compartments:

\$MODEL

COMP=(One,INITIALOFF,NODOSE)

COMP=(PKCENT,DEFDOSE,DEFOBS)

This error has been present since NONMEM 7.1

- 11) Left-strings appearing in \$ABBR REPLACE may not be replaced if there is a space preceding or trailing the left-string in the abbreviated code. For example:

\$ABBR REPLACE THINDEX=1

\$ABBR REPLACE THBASE=0.0

...

\$PK

...

CL=THETA (THINDEX) + THBASE

THINDEX and THBASE in the abbreviated code will not be replaced. However, when the space is removed:

CL=THETA (THINDEX) +THBASE

THINDEX will be replaced with the right_string 1, and THBASE will be replaced with the right string 0.0.

The leading space bug has been present since NONMEM 7.3, the trailing space bug has been present since NONMEM 7.4.0, and these have been fixed for NONMEM 7.4.2.