

Edison 을 통한 비선형 약동학 회귀분석과 simulation.

윤석규[†], 한성필, 조용순, 김형섭, 배균섭*

서울아산병원 임상약리학과, 울산대학교병원, 서울특별시 송파구 올림픽로 43 길 88

skyoon@acp.kr, ksbae@acp.kr*

abstract: 약동/약력학 분석을 위해서 약 30 년 전부터 비선형 회귀분석(Nonlinear regression analysis)이 오래전부터 사용되어 왔다. 주로 NONMEM[®] 이라고 하는 소프트웨어를 통해 활용되었으나, 이번 경진대회를 통해 비선형 회귀분석(Nonlinear regression analysis)방법을 Edison system 에 적용하여 여러 약동학 상황에서의 데이터에 대한 분석을 실시하였다. 각각의 데이터에 대해 Edison 앱 내에서, 비선형 회귀분석(Nonlinear regression analysis)을 사용할 수 있음을 확인하였다. 이번에 분석한 약동학 모형 이외에 다양한 약동학적 모형에 대한 비선형 회귀분석(Nonlinear regression analysis)을 통한 모델링을 Edison system 안에서 구현하고 활용할 수 있을 것으로 기대된다.

Introduction

약동/약력학 분석을 위해서 약 30년 전부터 비선형 회귀분석(Nonlinear regression analysis)이 사용 되어 왔고 현재 이 방식은 집당 약동/약력학 분석에서 가장 필수적인 플랫폼으로 자리 잡은 상태이다. 이 때 Phoenix WNL®과 같은 여러가지 소프트웨어들이 사용되고 있지만, 현재 이 비선형 회귀분석(Nonlinear regression analysis)을 적용시키기 위한 가장 널리쓰이는 도구는 NONMEM®이라고 하는 소프트웨어이다. 이것은 FORTRAN 코드로 프로그램 되었으며 현재 버전 7.4까지 지속적으로 업데이트 되고 있는 소프트웨어로, 현재 계량약리학 분야에서 가장 표준적인 도구로 생각되고 있다. 하지만 NONMEM® software는 정기적으로 사용하기 위해 상당부분의 비용을 지불하여야 하며, 일반인들에게 접근이 쉽지 않았다는 단점을 가지고 있다. 따라서 아산병원 임상약리학과는 여러 패키지들을 R®이라는 무료 배포 소프트웨어(R®은 통계 및 수학적 계산에 적합한 소프트웨어 이며 간단하지만 견고한 시스템을 가지고 있다. 이때 다양한 패키지를 추가하여 특정 문제나 상황에 적합한 계산을 수행할 수 있으며, 이러한 R®의 표준 기능은 모두 R® 언어로 코딩되어 있기 때문에, 그들의 알고리즘의 흐름을 쉽게 따라갈 수 있도록 한다.)에 등록하였으며, 이를 통해 비선형 회귀 분석 방법(Nonlinear regression)을 사용하여 다양한 약동/약력학 분석을 진행 할 수 있게 되었다. 이번 논문에서는 생성한 Package의 원리를 설명하며 이 package를 통해 시행한 다양한 약동학 모델을 실제 에디슨을 통해 시행 결과를 보여주려고 한다.

NONMEM® software의 비선형 회귀 분석

NONMEM®의 경우 비선형 회귀분석(Nonlinear regression analysis)을 시행하기 위해서 최대 우도 추정법(Maximum likelihood estimation)을 활용하게 된다. 이 최대 우도 추정법은 sample 데이터에서 Y가 현재 측정된 값, E(Y)는 model에 의해 각각의 관측 지점에서의 예상되는 값 그리고 V(Y)를 그 모델의 분산이라고 표현하면 모델에 주어진 관측값에 대한 likelihood는 다음과 같이 표현될 수 있다.

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

이때 이 일련의 관찰 값에 대한 우도는 개개인 관찰값의 확률을 통해 나오게 된다. 따라서 n개의 관찰값이 있으면 그 종합된 우도는 n개의 개개인별 확률이 되고 subject에 대한 우도는 다음과 같이 계산된다.

$$L_i = \prod_{j=1}^{n_i} \frac{1}{\sqrt{2\pi V(Y_{ij})}} e^{-\frac{1}{2V(Y_{ij})}(Y_{ij}-E(Y_{ij}))^2}$$

이것은 -2 로그 우도로(-2 log likelihood) 전환될 수 있으며, 이것은 다음과 같이 표현된다.

$$-2\log(L_i) = n_i \log(2\pi) + \sum_{j=1}^n \left(\log(V(Y_{ij})) + \frac{(Y_{ij} - E(Y_{ij}))^2}{V(Y_{ij})} \right)$$

결국 NONMEM은 이 -2 로그 우도의 값을 최소화 하는 방식으로 표현된다. 이 때, 우측 수식에서 앞에있는 $n_i \log(2\pi)$ 는 상수이기 때문에

실제 I번째 대상에서의 objection function은 다음과 같이 나타난다.

$$OBJ_i = n_i \log V(Y_i) + \sum_{j=1}^{n_i} \frac{(Y_{ij} - E(Y_{ij}))^2}{V(Y_i)}; (j = 1, \dots, n_i)$$

실제 여기에서 derivation 을 정규 분포에만 적용한 것이 최대 우도 측정법 (maximum likelihood estimation method)라면, 정규 분포라는 가정 없이도 적용한 것이 NONMEM 소프트웨어이며 이 방식을 '확장된 최소 제곱법'(extended least square method)이라고 한다. 이번 Edison system에서는 NONMEM® 의 확장된 최소 제곱법을 적용한 'nlr' function을 사용하였다.

지금까지는 Edison에 NONMEM의 비선형 회귀분석(Nonlinear regression analysis)을 설명하기 위한 이론적 배경이었으며, 이제 각각의 예제들을 통해 실제 계산결과가 Edison을 통해 어떻게 도출되는지를 확인해 보도록 하겠다.

Method

이번 Edison 과제를 통해 NONMEM을 Edison에 적용하기 위해서 두가지 package를 적용하였는데 이 중 하나는 'deSolve'이며 이것은 Karlene Soetaert, Thomas Petzoldt, R. Woodrow Setzer가 만든 여러 종류의 미분 방정식을 구하기 위한 package이다. Edison에 적용한 미분 방정식은 Isoda라는 function으로 편미분 기호가 들어간 ordinary differential equation을 사용할 수 있도록 도와주는 기능이다. 이는 실제 NONMEM에서 advan13을 적

용하는 것과 같다. 또 다른 하나는 'Pharmacokinetic and Pharmacodynamic Data Analysis - Concepts and Applications'의 다양한 식들에 대해서 NONMEM과 같이 minimization을 실행하기 위해 만든 'wnl'이라는 package이다. 우리가 NONMEM을 통해서 route finding으로 근을 찾는 것이 중요한 역할이다.

그 역할로 OFV minimization을 해주는 것이 바로 wnl에서 'nlr' function혹은 최소 제곱법 (least square method)을 활용하게 되는 'wnl' function이다. (최대 우도 측정법 (maximum likelihood estimation method)은 점 추정치를 찾는 데까지는 오래 걸리나, 표준 오차(standard error of estimates)를 구하는게 유리하며, 실제로도 최근에는 최소 제곱법(Least square method) 보다 널리 쓰이고 있다.)

Fig1.은 실제 package ('deSolve','wnl')를 우리가 원하는 모델에 사용하기 위해 script를 입력한 것이다. 여기서 THETA[1], THETA[2], THETA[3]는 각각 파라미터 중 K(제거 속도 상수), K_a(흡수 속도 상수), V(부피)를 의미한다. 또한 실제 우리가 구하고자 하는 구조모형(예시로 든 모델은 One-compartment oral dosing을 계산하기 위한 equation으로 이미 prediction 안에 식이 내장되어 있다.)을 기본적으로 내장되어 있지 않을 경우 입력해 주게 된다. 입력된 equation을 대입해, 다시 농도값을 계산해 주며, 이를 통해 각각의 prediction이 진행되는 것이다.

```

fPK = function(THETA) # Prediction function
{
  DOSE = 320000 # in microgram
  TIME = e$DATA[, "TIME"] # use data in e$DATA

  K = THETA[1]
  Ka = THETA[2]
  V = THETA[3]

  P = DOSE/V*Ka/(Ka - K) * (exp(-K*TIME) - exp(-Ka*TIME))
  return(P)
}

IDs = unique(tData[, "ID"])
nID = length(IDs)
for (i in 1:nID) {
  Data = tData[tData$ID == IDs[i],]
  Res = nlr(fPK, Data, pNames=c("k", "ka", "v"), IE=c(0.1, 3, 500),
    SecNames=c("CL", "Thalf", "MRT"), SecForms=c(~V*k, ~log(2)/k, ~1/k))
  print(paste0("## ID = ", i, "##"))
  print(Res)
}

```

Fig1. 약동학 분석을 위해 입력된 스크립트의 예시.

Subject군이 여러명일 경우 각각의 대상자의 값을 모두 구하기 위해서 for loop라는 function을 다음과 같이 사용하게 된다.

각각의 nonlinear regression을 위해서 theta에 대한 initial value를 넣어주고. minimization이 된 후 위에서 언급된 파라미터 이외에 다른 파라미터에 대해서도 값을 구하기 위해서 다음과 같이 (CL(제거 속도), T_{half} (반감기), MRT(평균 체류 시간)) 추가적으로 넣어준 후, 입력한 파라미터값들을 구하기 위한 수식을 뒤쪽 부분 'SecForms'란에 입력하게 된다. 이런 방법을 통하여 우리가 원하는 파라미터들의 점추정치와(point estimate), 그리고 표준 오차(standard error of estimates)를 모두 구해 줄 수 있다.

각 모델에 맞는 적합한 모델에 따라 다음과 같은 script 형태로 입력해 주게 되며, 그에 대한 표준 오차(standard error of estimates), 공분산 행렬(covariance matrix of estimates), run test results on residuals, AIC, AICc, SBC를 내부 계산을 통하여 도출하게 된다.

이때 run test란 residual들이 random하게 분포하는지를 보기 위해서 각각의 residual에서 최대 양수와 음수가 연속구간의 길이가 얼마나 되는지를 보는 방법이다. 이 때 계산된 p value

를 통해서 실제로 residual이 random하게 분포하는지, 편향되게 분포하는지를 고려해 볼 수 있다.

AIC, AICc, SBC는 Akaike information criterion, Akaike information criterion corrected, Schwarz criterion을 의미하며 각각의 통계적 모델에서 구조가 적합한지를 확인할수 있도록 도와주는 지표이다..

이번 프로그램을 통해, 실제 minimization을 실시할 뿐만 아니라, 실제로 모델이 적합한지에 대해서 모두 확인해 볼 수 있게 구현하였다. 다음부터는 각각의 3개의 모델에 대해서 실제로 Edison system을 보고자 한다.

Example1. One compartment oral dosing

Variable Name	Value	Description
number:	One-compartment oral dosing	
One-compartment oral dosing		
Two-compartment distribution models		
Modeling of fraction absorbed and nonlinear bioavailability across the liver		
		Confirm Cancel

Fig2. Edison을 통한 'One-compartment oral dosing' Simulation생성.

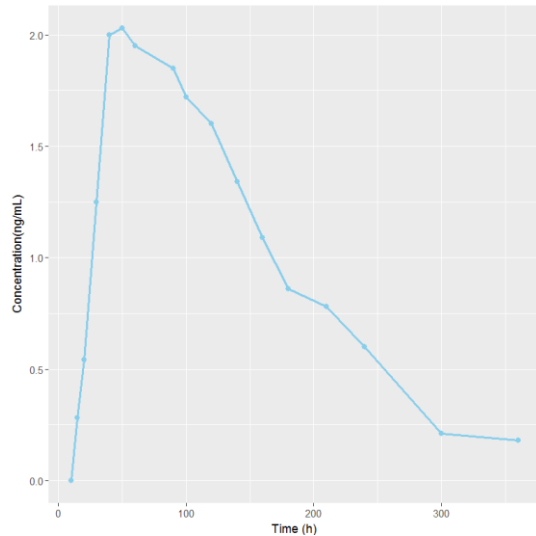


Fig3. 'One-compartment oral dosing analysis'에 사용된 데이터의 time-concentration profile.

경구 복용시 1차 지수로 감소하는 약물에서의 약동학 모형이다. 이 경우에는 'wnl'패키지 안에 기본적 equation이 정의되어 있으므로 추가적인 equation을 입력하지 않고 각각의 파라미터의 값에 해당하는 theta에 대해서만 입력하였다. (값을 구하기 위해 필요한 theta는 K (제거 속도 상수), K_a (흡수 속도 상수), V (부피), t_{lag} (흡수가 지연되는 시간)이다.) 또한 추가적으로 F (생체 이용률) 값을 구하기 위해 다음과 같이 식으로 표시하였으며, 이중에서 지연 시간(Lag time)이 있는지 없는지를 각각 가정하여 Fig5, Fig6와 같이 function을 정의 하였다.

```
PK02 = read.csv("data-raw/PK02.csv", skip=1)
colnames(PK02) = c("TIME", "DV")
DV2 = c(4.75, 4.43, 4.21, 3.81, 3.57, 3.28, 2.93, 2.21, 1.95, 1.72, 1.47, 1.24, 1.02, 0.77, 0.61, 0.36, 0.18, 0.20)
PK02 = cbind(PK02, DV2)
PK02
```

##	TIME	DV	DV2
## 1	10	0.00	4.75
## 2	15	0.28	4.43
## 3	20	0.54	4.21
## 4	30	1.25	3.81
## 5	40	2.00	3.57
## 6	50	2.03	3.28
## 7	60	1.95	2.93
## 8	90	1.85	2.21
## 9	100	1.72	1.95
## 10	120	1.60	1.72
## 11	140	1.34	1.47
## 12	160	1.09	1.24
## 13	180	0.86	1.02
## 14	210	0.78	0.77
## 15	240	0.60	0.61
## 16	300	0.21	0.36
## 17	360	0.18	0.20

Fig4. 'One-compartment oral dosing analysis'를 적용하기 위해 사용된 데이터.

```
## Model without tlag
Dose = 100

fPK01 = function(THETA) # Prediction function
{
  DOSE = Dose
  TIME = e$DATA[, "TIME"]

  BA = BA
  K = THETA[1]
  Ka = THETA[2]
  V = THETA[3]

  F = BA*DOSE/V*Ka/(Ka - K) * (exp(-K*TIME) - exp(-Ka*TIME))
  H1 = 1

  return(cbind(F, H1))
}

nls(fPK01, dPK02, pNames=c("k", "Ka", "V"), IE=c(0.05, 0.1, 30))
```

Fig5 'One-compartment oral dosing analysis'를 계산하기 위한 스크립트.(without lag time)

```
## Model with tlag
fPK02 = function(THETA) # Prediction function
{
  DOSE = Dose
  TIME = e$DATA[, "TIME"]

  BA = BA
  K = THETA[1]
  Ka = THETA[2]
  V = THETA[3]
  tlag = THETA[4]

  F = BA*DOSE/V*Ka/(Ka - K) * (exp(-K*(TIME - tlag)) - exp(-Ka*(TIME - tlag)))
  H1 = 1

  return(cbind(F, H1))
}

BA = 1
Dose = 100
colnames(PK02) = c("TIME", "DV", "DV2")
nls(fPK02, dPK02, pNames=c("k", "Ka", "V", "tlag"), IE=c(0.05, 0.1, 30, 20))
```

Fig6. 'One-compartment oral dosing analysis'를 계산하기 위한 스크립트. (with lag

time)

```
## $Est
##           k           Ka           V   AddErrVar   AddErrSD
## PE    0.013698916   0.013699384  12.512458   0.27325141   0.52273455
## SE    0.009900321   0.009900641   8.994196   0.06627321   0.06339088
## RSE   72.270834504  72.270699963  71.881932  24.25356542  12.12678271
##
## $Cov
##           k           Ka           V   AddErrVar
## k      9.801636e-05 -9.172333e-05 -8.727111e-02 -3.783325e-09
## Ka     -9.172333e-05  9.802268e-05  8.603379e-02  3.734622e-09
## V      -8.727111e-02  8.603379e-02  8.089557e+01  3.532174e-06
## AddErrVar -3.783325e-09  3.734622e-09  3.532174e-06  4.392138e-03
##
## $run
## $run$m
## [1] 16
##
## $run$n
## [1] 18
##
## $run$run
## [1] 7
##
## $run$p.value
## [1] 7.541375e-05
##
## $AIC
## [1] 29.13356
##
## $AICC
## [1] 32.4669
```

Fig7. Lag time이 없는 ‘One-compartment orad dosing model’을 ‘nlr’ function을 통해 계산했을시의 결과 값.

```
## $Est
##           k           Ka           V           tlag   AddErrVar   AddErrSD
## PE    0.010110127   0.03021029  28.45123  11.495191   0.24179045   0.49172192
## SE    0.006120683   0.02122374  13.06396   3.189944   0.05864321   0.05963046
## RSE   60.540118876  70.25332892  45.91703  27.750249  24.25373334  12.12686667
##
## $Cov
##           k           Ka           V           tlag
## k      3.746276e-05 -1.174219e-04 -7.746652e-02 -3.191263e-03
## Ka     -1.174219e-04  4.504470e-04  2.590532e-01  2.241856e-02
## V      -7.746652e-02  2.590532e-01  1.706671e+02  8.061794e+00
## tlag    -3.191263e-03  2.241856e-02  8.061794e+00  1.017574e+01
## AddErrVar -1.856686e-09  2.585447e-09  4.069133e-06 -4.016032e-06
## AddErrVar
## k      -1.856686e-09
## Ka      2.585447e-09
## V      4.069133e-06
## tlag    -4.016032e-06
## AddErrVar 3.439026e-03
##
## $run
## $run$m
## [1] 17
##
## $run$n
## [1] 17
##
## $run$run
## [1] 12
##
## $run$p.value
## [1] 0.02721811
##
## $AIC
## [1] 26.97442
```

```
## $AICC
## [1] 32.42897
```

Fig8. Lag time이 있는 ‘One-compartment orad dosing model’을 ‘nlr’ function을 통해 계산했을시의 결과 값

```
## $PE
##           k           Ka           V           tlag
##    0.01011006   0.03021076  28.45136106  11.49550270
##
## $WRSS
## [1] 8.220817
##
## $run
## $run$m
## [1] 17
##
## $run$n
## [1] 17
##
## $run$run
## [1] 12
##
## $run$p.value
## [1] 0.02721811
##
## $AIC
## [1] 43.81338
##
## $SBC
## [1] 47.14624
```

Fig9. Lag time이 있는 ‘One-compartment orad dosing model’을 ‘wnl’ function을 통해 계산했을시의 결과 값.

값을 계산할시 다음과 같이(Fig7, Fig8, Fig9) 각각 파라미터 값들이(K, K_a , V, t_{lag}) 출력되는 것을 확인해 볼 수 있었으며, 이때 run\$p.value와 AIC를 통해서 실제 적절한 모델인지를 확인해 볼 수 있었다. 이번 example에서는 run test를 통해 구한 p.value가 0.05이하(0.027)로 계산되어, residual들이 치우쳐 있으며 모델의 개선이 필요하다는 것을 확인할 수 있었다. 또한 ‘wnl’ function과 ‘nlr’ function의 차이에서 보이듯 최대 우도 측정법(maximum likelihood method)에서는 표준 오차(standard error of estimates)

가 출력되는 것을 확인해 볼 수 있었지만, 최소 제곱법(least square method)에서는 표준 오차 (standard error of estimates)가 표시되지 않는 것을 볼 수 있었다.

Example2. Two-compartment distribution models

Variable Name	Value	Description
number :	Two-compartment distribution	?
	One-compartment oral dosing	
	Two-compartment distribution models	
	Modeling of fraction absorbed and nonlinear bioavailability across	

Fig10. Edison을 통한 'Two-compartment distribution models' simulation을 생성.

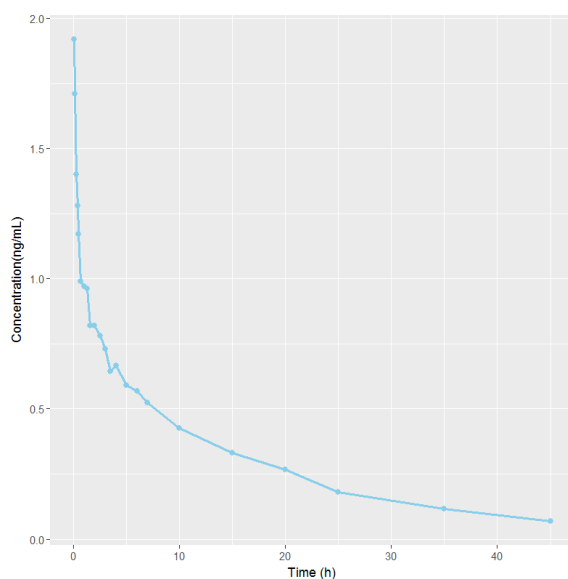


Fig11. 'Two-compartment distribution analysis'에 사용된 데이터의 time-concentration profile.

2차 지수로 감소하는 약물에서의 약동학 모형이며, 이때의 농도 값(C_p)을 구하기 위한 공식

은 다음과 같이 정해진다.

$$C_p = A \cdot e^{-\alpha \cdot t} + B \cdot e^{-\beta \cdot t}$$

따라서 다음과 같이 Script에 ' $C_p = A \cdot \exp(-\alpha \cdot \text{TIME}) + B \cdot \exp(-\beta \cdot \text{TIME})$ ' 식을 표시하였으며, 이 식을 구하기 위해 각각의 'A','alpha','B','beta'에 해당하는 값을 입력하였다. (Fig13)

```
dPK08 = read.csv("data-raw/PK08.csv", skip=1)
colnames(dPK08) = c("TIME", "DV") ; dPK08
```

```
## TIME DV
## 1 0.05 1.920
## 2 0.10 1.710
## 3 0.25 1.400
## 4 0.40 1.280
## 5 0.50 1.170
## 6 0.70 0.990
## 7 1.00 0.970
## 8 1.30 0.962
## 9 1.60 0.820
## 10 2.00 0.819
## 11 2.50 0.780
## 12 3.00 0.729
## 13 3.50 0.643
## 14 4.00 0.665
## 15 5.00 0.591
## 16 6.00 0.567
## 17 7.00 0.524
## 18 10.00 0.426
## 19 15.00 0.332
## 20 20.00 0.267
## 21 25.00 0.179
## 22 35.00 0.116
## 23 45.00 0.069
```

Fig12. 'Two-compartment distribution analysis'를 적용하기 위해 사용된 데이터.

```
##
fPK08a = function(THETA) # biexponential
{
  A = THETA[1]
  alpha = THETA[2]
  B = THETA[3]
  beta = THETA[4]
  TIME = e$DATA[, "TIME"]
  Cp = A*exp(-alpha*TIME) + B*exp(-beta*TIME) # eq 8:1
  return(Cp)
}

nlr(fPK08a, dPK08, pNames=c("A", "alpha", "B", "beta"), IE=c(2, 2, 1, 0.1), Error="POIS")
```

Fig13. 'Two-compartment distribution analysis'를 계산하기 위한 스크립트.

값을 계산할시 다음과 같이(Fig14, Fig15) 값이 출력되는 것을 확인해 볼 수 있었으며, run test의 p-value가 0.05이하로(0.017), residual들이 유의미하게 편향되어 있고, 모델의 개선이

좀 더 필요함을 확인해 볼 수 있었다. Example 1과 마찬가지로 보이듯 최대 우도 측정법(maximum likelihood method)에서는 표준 오차(standard error of estimates)가 출력되는 것을 확인해 볼 수 있었지만, 최소 제곱법(least square method)에서는 표준 오차(standard error of estimates)가 표시되지 않는 것을 볼 수 있었다.

```
## $est
##           A      alpha      B      beta  PoisErrVar  PoisErrSD
## PE  1.09702032  2.2241012  0.85517722  0.060580004  2.332974e-03  0.048300865
## SE  0.06213283  0.2868177  0.02330448  0.002526805  6.889081e-04  0.007131427
## RSE 5.66378097 12.8958944  2.72510509  4.171021296  2.952919e+01  14.764594032
##
## $cov
##           A      alpha      B      beta
## A      3.860488e-03  9.865563e-03 -3.096518e-08 -4.755957e-06
## alpha  9.865563e-03  8.226442e-02  4.401645e-03  3.029052e-04
## B      -3.096518e-08  4.401645e-03  5.430987e-04  4.115985e-05
## beta   -4.755957e-06  3.029052e-04  4.115985e-05  6.384743e-06
## PoisErrVar -4.690200e-08  8.737138e-07 -9.439557e-08  5.318549e-08
##
## A      -4.690200e-08
## alpha  8.737138e-07
## B      -9.439557e-08
## beta   5.318549e-08
## PoisErrVar 4.745944e-07
##
## $run
## $run$m
## [1] 10
##
## $run$n
## [1] 13
##
## $run$run
## [1] 7
##
## $run$.value
## [1] 0.01702786

## $AIC
## [1] -75.81262
##
## $AICC
## [1] -72.28321
```

Fig14. ‘Two-compartment distribution model’을 nlr function을 통해 계산했을시의 결과 값.

```
## $PE
##           A      alpha      B      beta
## 1.09725824 2.21980682 0.85563811 0.06031816
##
## $WRSS
## [1] 0.0535843
##
## $run
## $run$m
## [1] 10
##
## $run$n
## [1] 13
##
## $run$run
## [1] 7
##
## $run$.value
## [1] 0.01702786
##
##
## $AIC
## [1] -59.30948
##
## $SBC
## [1] -54.76751
```

Fig15. ‘Two-compartment distribution model’을 wnl function을 통해 계산했을시의 결과 값.

Example 3 ‘fraction absorbed and nonlinear bioavailability across the liver’ model

Variable Name	Value	Description
number :	Two-compartment distribution ▾	7
	One-compartment oral dosing	
	Two-compartment distribution models	
	Modeling of fraction absorbed and nonlinear bioavailability across	

Fig16. Edison을 통한 ‘fraction absorbed and nonlinear bioavailability across the liver’ simulation을 생성.

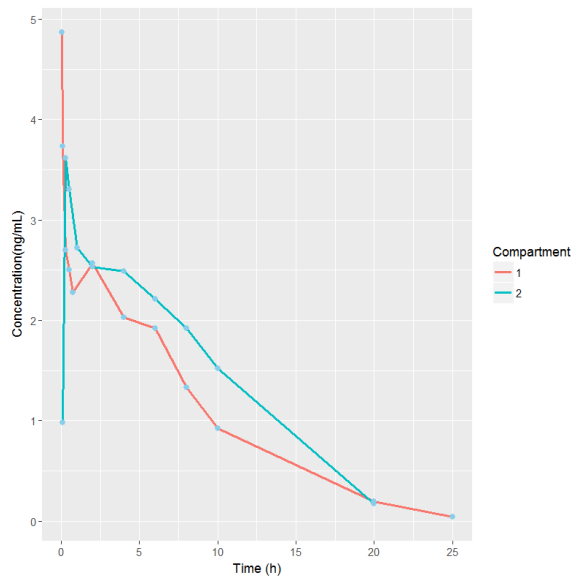


Fig17. 'fraction absorbed and nonlinear bioavailability across the liver analysis'에 사용된 데이터의 time-concentration profile.

```
dPK09[, "DV"] = as.numeric(dPK09[, "DV"]); dPK09
```

##	TIME	DV	CMT
## 1	0.030	4.8700	1
## 2	0.100	3.7300	1
## 3	0.250	2.7000	1
## 4	0.500	2.5000	1
## 5	0.750	2.2800	1
## 7	2.000	2.5700	1
## 9	4.000	2.0300	1
## 10	6.000	1.9200	1
## 11	8.000	1.3300	1
## 12	10.000	0.9200	1
## 14	20.000	0.1930	1
## 15	25.000	0.0438	1
## 16	0.081	0.9820	2
## 17	0.250	3.6200	2
## 18	0.500	3.3000	2
## 19	1.000	2.7200	2
## 20	2.000	2.5300	2
## 21	4.000	2.4900	2
## 22	6.000	2.2100	2
## 23	8.000	1.9200	2
## 24	10.000	1.5200	2
## 25	20.000	0.1710	2

Fig18. 'fraction absorbed and nonlinear bioavailability across the liver analysis'를 적용하기 위해 사용된 데이터

간을 통해 대사되는 비선형적 특성을 띄는 약물에서의 약동학 모형이며, 이 때에는 좀 더 복잡한 equation들을 다수 활용하기 위하여 function을 통해서 각각 compartment에서의 농도를 구하기 위한 equation을 정의해 주었다. 이를 통하여, 'absorption compartment', 'central compartment', 'gut compartment', 'hepatic compartment'에서의 각각 equation을 입력하였으며, Fig19. 와 같이 표현되었다.

```
ivPKde = function(t, y, p)
{
  if (t < 0.0056) {
    RateIn = 357.1429 # 2/0.0056
  } else {
    RateIn = 0
  }
  CL = p["vmax"]/(p["km"] + y[4])
  dy1dt = -p["ka"]*y[1] # absorption, gut comp
  dy2dt = (RateIn - QH*(y[2] - y[4]) - p["cl_d"]*(y[2] - y[3]))/p["vc"] # central, plasma
  dy3dt = (p["cl_d"]*(y[2] - y[3]))/p["vt"] # peripheral, tissue
  dy4dt = (p["ka"]*y[1] + QH*(y[2] - y[4]) - CL*y[4])/VH # hepatic comp
  return(list(c(dy1dt, dy2dt, dy3dt, dy4dt)))
}

popPKde = function(t, y, p)
{
  CL = p["vmax"]/(p["km"] + y[4])
  if (t < p["t_lag"]) {
    ka = 0
  } else {
    ka = p["ka"]
  }
  dy1dt = -ka*y[1] # absorption, gut comp
  dy2dt = (-QH*(y[2] - y[4]) - p["cl_d"]*(y[2] - y[3]))/p["vc"] # central, plasma
  dy3dt = (p["cl_d"]*(y[2] - y[3]))/p["vt"] # peripheral, tissue
  dy4dt = (ka*y[1] + QH*(y[2] - y[4]) - CL*y[4])/VH # hepatic comp
  return(list(c(dy1dt, dy2dt, dy3dt, dy4dt)))
}
```

Fig19. 'fraction absorbed and nonlinear bioavailability across the liver model'을 계산하기 위한 스크립트1.

위의 식을 활용해 'wnl'package의 'nlr' function을 활용하여 equation에 필요한 theta값들 (K_a (흡수 속도 상수), V_c (중심 구획의 부피), V_t (조직에서의 부피), V_{max} (최대 대사 속도), Cl_d (구획 간 분포))을 정의해 주고 지정한 후, 관련된 초기 추정값을(IE) 입력하였다.

제7회 EDISON SW 활용 경진대회

```

fPK09 = function(THETA)
{
  Ka = THETA[1]
  Vc = THETA[2]
  Vt = THETA[3]
  Km = THETA[4]
  Vmax = THETA[5]
  Clid = THETA[6]
  Tlag = THETA[7]
  Fa = THETA[8]

  Fc1 = lsoda(y=c(0, 0, 0, 0), times=Times1, func=fvPKde, parms=c(Ka=Ka, Vc=Vc, Vt=Vt, Km=Km, Vmax=Vmax, Clid=Clid, Tlag=Tlag, Fa=Fa))
  Fc2 = lsoda(y=c(Fa*Dpo, 0, 0, 0), times=Times2, func=poPKde, parms=c(Ka=Ka, Vc=Vc, Vt=Vt, Km=Km, Vmax=Vmax, Clid=Clid, Tlag=Tlag, Fa=Fa))
  return(c(Fc1[1:time1,3], Fc2[1:time2,3]))
}

nls(fPK09, dPK09, pNames=c("Ka", "Vc", "Vt", "Km", "Vmax", "Clid", "Tlag", "Fa"),
    IE=c(10, 0.4, 0.6, 0.3, 0.1, 2, 0.08, 0.5),
    LB=c(0, 0.001, 0.001, 0, 0, 0, 0, 0),
    UB=c(30, 1, 2, 1, 4, 1, 1), Error="p")

```

Fig20. ‘fraction absorbed and nonlinear bioavailability across the liver model’을 계산하기 위한 스크립트2.

다음과 같이(Fig21, Fig22) 파라미터의 값과 표준 오차(standard error of estimates)가 출력되는 것을 확인하였다. 이 때 run test의 residual error가 한쪽으로 유의미하게 치우쳐 있지 않음을 확인해 볼 수 있었다.

```

## $Est
##      Ka      Vc      Vt      Km      Vmax      Clid
## PE  13.59331  0.3338348  0.3498523  0.7322811  0.16967703  2.112227
## SE  16.77230  0.1400570  0.1294923  0.2604637  0.02801978  2.637323
## RSE 123.38644  41.9539676  37.0134157  35.5688139  16.51359919  124.859804
##      Tlag      Fa  PropErrVar  PropErrSD
## PE  0.06196690  0.36328212  0.012238056  0.11062575
## SE  0.02286281  0.01496935  0.003782218  0.01709465
## RSE 36.89519340  4.12058676  30.905377448  15.45268872
##
## $Cov
##      Ka      Vc      Vt      Km
## Ka  281.309959832 -2.056469e+00  1.846295e+00  3.0953092539
## Vc  -2.056468767  1.961595e-02 -1.770738e-02 -0.0269569498
## Vt   1.846294914 -1.770738e-02  1.676826e-02  0.0224826026
## Km   3.095309254 -2.695695e-02  2.248260e-02  0.0678413313
## Vmax 0.328812365 -2.864023e-03  2.441778e-03  0.0072581913
## Clid 40.798969747 -3.572394e-01  3.214438e-01  0.5225521968
## Tlag 0.379319427 -2.963002e-03  2.670982e-03  0.0043961139
## Fa   0.096275864 -8.735160e-04  9.361984e-04  0.0023116731
## PropErrVar -0.007091359  6.354574e-05 -4.915759e-05 -0.0000819814
##      Vmax      Clid      Tlag      Fa
## Ka  3.288124e-01  40.798969747  3.793194e-01  9.627586e-02
## Vc  -2.864023e-03 -0.357239382 -2.963002e-03 -8.735160e-04
## Vt   2.441778e-03  0.321443844  2.670982e-03  9.361984e-04
## Km   7.258191e-03  0.522552197  4.396114e-03  2.311673e-03
## Vmax 7.851083e-04  0.055560437  4.680736e-04  2.697044e-04
## Clid 5.556044e-02  6.95472807  5.743056e-02  1.745153e-02
## Tlag 4.680736e-04  0.057430565  5.227080e-04  1.444884e-04
## Fa   2.697044e-04  0.017451533  1.444884e-04  2.240816e-04
## PropErrVar -7.630608e-06 -0.001136308 -9.831995e-06 -2.399614e-06

```

Fig21. ‘fraction absorbed and nonlinear bioavailability across the liver model’을 nls function을 통해 계산했을시의 결과 값.

```

##      PropErrVar
## Ka      -7.091359e-03
## Vc      6.354574e-05
## Vt     -4.915759e-05
## Km     -8.198140e-05
## Vmax   -7.630608e-06
## Clid   -1.136308e-03
## Tlag   -9.831995e-06
## Fa     -2.399614e-06
## PropErrVar 1.430517e-05
##
## $run
## $run$m
## [1] 9
##
## $run$n
## [1] 13
##
## $run$run
## [1] 10
##
## $run$p.value
## [1] 0.3049536
##
##
## $AIC
## [1] 1.265977
##
## $AICC
## [1] 16.26598

```

Fig22. ‘fraction absorbed and nonlinear bioavailability across the liver model’을 wnl function을 통해 계산했을시의 결과 값

Conclusion.

비선형 회귀분석(Nonlinear regression analysis)은 대상자의 인구 집단에서의 측정되는 데이터를 설명하는데 사용하며, 실제 약물을 분석하는데 가장 중요한 방법 중 하나로 생각되고 있다. 이중 NONMEM®이라는 software는 다양한 산업과 학문에서 가장 많이 사용되는 표준으로 생각되고 있다. 하지만 실제 NONMEM® software는 다루는데 있어서 어려

운 부분들이 있을뿐만 아니라, 일반 사람들이 적용하기에는 어렵다는 단점을 가지고 있다. 위와같이 다양한 약동학 모델들을 통해, NONMEM® software의 기능을 Edison에 적용시킬 수 있다는 것을 확인하였다. 아직까지는 좀 더 복잡한 모델들을 적용시키지는 못했다는 한계가 있으나, 이러한 비선형 회귀분석(Nonlinear regression analysis)을 Edison에 적용할 수 있다는 점만으로도 상당히 의미 있는 결과라고 생각된다. 남은 Edison 기간 동안 현재 약물 농도를 분석하기 위한 다양한 약동/약력학 모델을 update할 수 있을 것으로 보이며, 이를 통해 좀 더 다양한 데이터들에 대해 Edison을 적용하고 이를 통해, 더 많은 사람들이 이러한 비선형 회귀분석(Nonlinear regression analysis)에 대해서 이해할 수 있을 것이라 기대한다.

감사의 글

본 논문은 2017년도 정부(미래창조과학부)의
재원으로 한국연구재단 첨단 사이언스·교육 허
브 개발 사업의 지원을 받아 수행된 연구임
(NRF-2011-0020576)

참고문헌

[1] R-based reproduction of the estimation process
hidden behind NONMEM® Part 1: first-order
approximation method, Min-Gul Kim¹, Dong-Seok
Yim² and Kyun-Seop Bae^{3,4*}, 2015;23(1):01-07

[2] Hadley Wickham, Advanced R (Chapman &
Hall/CRC The R Series) 1st Ed ,25 sep 2014

[3] Gabrielsson J and Weiner D. Pharmacokinetic
and Pharmacodynamic Data Analysis - Concepts
and Applications. 5th ed. 2016. (ISBN:9198299107).

[4] <http://www.edison.re.kr>. EDISON 웹사이트.