

한성필

R로 하는 약동학 모델링



Contents

List of Tables	5
List of Figures	7
Introduction	9
1 One compartment model - oral dosing	1
1.1 Compartmental analysis without Tlag	2
1.2 Compartmental analysis with Tlag	3
1.3 Modeling Result	3
2 One compartment model - IV dosing	5
2.1 Compartmental analysis	7
2.2 Modeling Result	8
3 Two compartment model – oral dosing	9
3.1 Compartmental analysis without Tlag	9
3.2 Compartmental analysis with Tlag	10



List of Tables



List of Figures



Introduction

이 자료는 Johan Gabrielsson과 Dan Weiner의 책, “Pharmacokinetic and Pharmacodynamic Data Analysis - Concepts and Applications” 5th ed. (Gabrielsson 2016)를 풀기 위한 것입니다. 서울아산병원 임상약리학과 배균섭 교수님께서 개발한 `wnl` 패키지 (Bae 2018)를 활용하였습니다.

오타자 신고 등은 [깃허브 저장소](#)에 남겨주십시오.

감사합니다.

2019년 11월
가톨릭대학교 약리학교실
연구강사 한성필

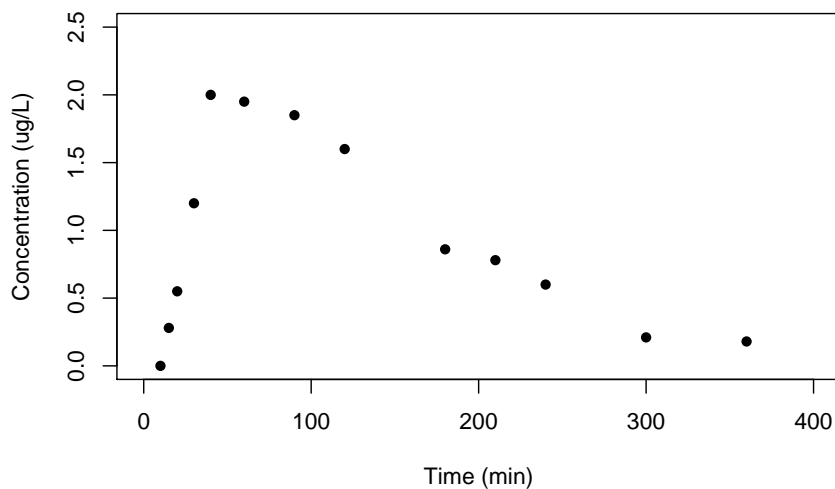


1

One compartment model - oral dosing

wn1 라이브러리를 불러오고 자료를 읽어옵니다.

```
library(wn1)
dPK02 = read.csv("data/hands-on1.csv", skip=1)
colnames(dPK02) = c("TIME", "DV")
plot(dPK02[, "TIME"], dPK02[, "DV"], xlim=c(0, 400), ylim=c(0, 2.5),
      xlab="Time (min)", ylab="Concentration (ug/L)", pch=16)
```



먼저 NCA 분석을 해 봅니다. Tmax는 40분, Cmax는 2.00 ug/L인 것을 알 수 있습니다.

```
NonCompart::sNCA(dPK02[, "TIME"], dPK02[, "DV"], dose=100, doseUnit="ug", timeUnit="min")
```

##	b0	Cmax	CmaxD	Tmax	TLag
##	1.557534e+00	2.000000e+00	2.000000e-02	4.000000e+01	1.000000e+01
##	CLST	CLSTP	TLST	LAMZHL	LAMZ
##	1.800000e-01	1.647689e-01	3.600000e+02	7.424929e+01	9.335405e-03

```
##          LAMZLL          LAMZUL          LAMZNPT          CORRXY          R2
## 9.000000e+01 3.600000e+02 7.000000e+00 -9.822141e-01 9.647446e-01
##          R2ADJ          AUCLST          AUCALL          AUCIFO          AUCIFOD
## 9.576935e-01 3.308750e+02 3.308750e+02 3.501564e+02 3.501564e+00
##          AUCIFP          AUCIFPD          AUCPEO          AUCPEP          AUMCLST
## 3.485249e+02 3.485249e+00 5.506520e+00 5.064170e+00 4.230750e+04
##          AUMCIFO          AUMCIFP          AUMCPEO          AUMCPEP          VZFO
## 5.131423e+04 5.055210e+04 1.755210e+01 1.630912e+01 3.059178e+01
##          VZFP          CLFO          CLFP          MRTEVLST          MRTEVIFO
## 3.073499e+01 2.855866e-01 2.869236e-01 1.278655e+02 1.465466e+02
##          MRTEVIFP
## 1.450459e+02
## attr("units")
## [1] ""          "ug/L"          "ug/L/ug"          "min"          "min"
## [6] "ug/L"          "ug/L"          "min"          "min"          "/min"
## [11] "min"          "min"          ""          ""          ""
## [16] ""          "min*ug/L"          "min*ug/L"          "min*ug/L"          "min*ug/L/ug"
## [21] "min*ug/L"          "min*ug/L/ug"          "%"          "%"          "min2*ug/L"
## [26] "min2*ug/L"          "min2*ug/L"          "%"          "%"          "L"
## [31] "L"          "L/min"          "L/min"          "min"          "min"
## [36] "min"
## attr("UsedPoints")
## [1] 7 8 9 10 11 12 13
```

1.1 Compartmental analysis without Tlag

경구 투여 일구획 분석, 지연시간이 없는 모형입니다. Ka, V, K로 농도를 나타낼 수 있습니다.

DOSE = 100

```
fPK02a = function(THETA) # Prediction function
{
  Ka = THETA[1]
  V = THETA[2]
  K = THETA[3]
  Cp = DOSE/V*Ka/(Ka - K)*(exp(-K*TIME) - exp(-Ka*TIME)) # eq 2:1
  return(Cp)
}
TIME = dPK02[, "TIME"]
r1 = nlr(fPK02a, dPK02, pNames=c("ka", "V", "k"), IE=c(0.1, 30, 0.05))
```

```
r1$Est
```

```
##          ka          V          k  AddErrVar  AddErrSD
## PE  0.013202142 21.017659 0.013202008 0.08701446 0.29498213
## SE  0.005442004 8.623411 0.005441954 0.03412981 0.05785063
## RSE 41.220611647 41.029360 41.220651532 39.22314350 19.61157175
```

1.2 Compartmental analysis with Tlag

경구 투여 일구획 분석, 지연시간이 있는 모형입니다. Ka, V, K에 Tlag가 추가되어 농도를 나타낼 수 있습니다.

```
fPK02b = function(THETA) # Prediction function
{
  Ka  = THETA[1]
  V   = THETA[2]
  K   = THETA[3]
  tlag = THETA[4]

  Cp = DOSE/V*Ka/(Ka - K)*(exp(-K*(TIME - tlag)) - exp(-Ka*(TIME - tlag))) # eq 2:2
  return(Cp)
}
TIME = dPK02[, "TIME"]
r2 = nlr(fPK02b, dPK02, pNames=c("ka", "V", "k", "tlag"), IE=c(0.1, 30, 0.05, 20))

r2$Est
```

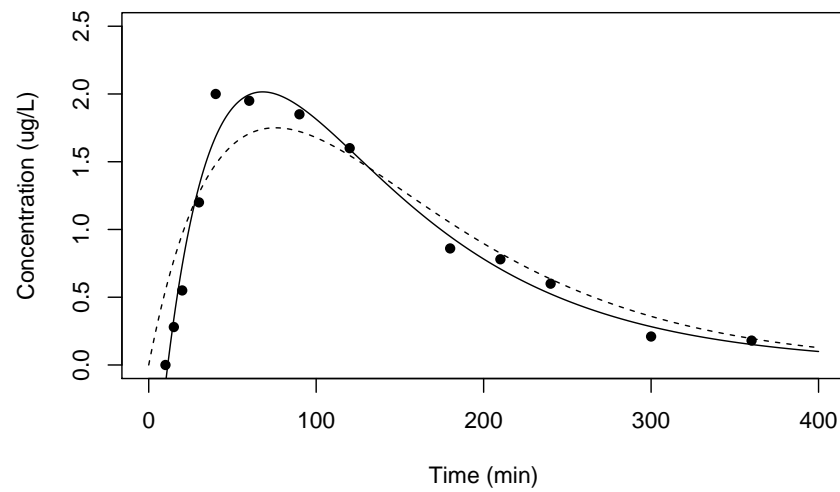
```
##          ka          V          k      tlag  AddErrVar  AddErrSD
## PE  0.027469911 27.353876 0.010511280 11.375450 0.015494408 0.12447654
## SE  0.006059932 4.148751 0.001937513 0.867595 0.006077431 0.02441195
## RSE 22.060253988 15.166957 18.432707310 7.626907 39.223383633 19.61169182
```

1.3 Modeling Result

지연 시간이 있는 모형의 적합이 더 좋은 것을 알 수 있습니다. Cmax 부분을 주의깊게 살펴보세요.

```
# Figure 2.3, p 480
plot(dPK02[, "TIME"], dPK02[, "DV"], xlim=c(0, 400), ylim=c(0, 2.5),
     xlab="Time (min)", ylab="Concentration (ug/L)", pch=16)
```

```
TIME = 0:400  
lines(TIME, fPK02a(r1$Est["PE", 1:3]), lty=2)  
lines(TIME, fPK02b(r2$Est["PE", 1:4]))
```



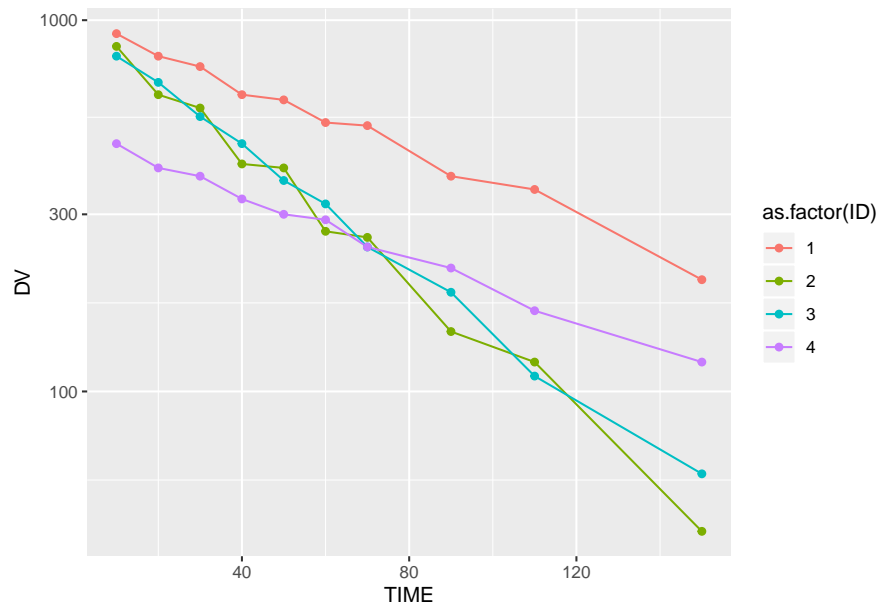
2

One compartment model - IV dosing

wnl 라이브러리를 불러오고 자료를 읽어옵니다. 4명의 IV 투약 후 농도 자료를 불러와 그림을 자료 탐색을 할 수 있습니다.

```
require(wnl)
dPK01 = read.csv("data/hands-on2.csv", skip=1)
colnames(dPK01) = c("TIME", "DV", "ID")

library(ggplot2)
ggplot(dPK01, aes(TIME, DV, group = ID, color = as.factor(ID))) +
  geom_line() + geom_point() + scale_y_log10()
```



- 4명의 피험자 모두 대략적으로 시간에 따른 농도 감소가 단항 지수함수적인 것을 관찰할 수 있습니다.
- 피험자 1과 2를 비교하면 2번 피험자가 AUC가 더 작으며, 따라서 청소율이 더 클 것이라 예상할 수 있으며, Y 절편이 거의 같은 것으로

보아 분포용적이 유사할 것으로 보입니다.

- 피험자 3과 4의 경우 곡선이 교차하는 형태로 눈으로는 어느 쪽이 AUC가 클지 알기 어렵고, Y절편에 해당하는 농도가 높은 쪽이 분포용적이 더 작을 것이라 예상할 수 있습니다.

4명 자료의 NCA 분석을 `tblNCA()` 함수를 사용해 계산할 수 있습니다.

```
NonCompartment::tblNCA(dPK01, key="ID", colTime="TIME", colConc="DV", dose=10, adm="Bolus")
```

```
##      ID      b0 CMAX CMAXD TMAX TLAG CLST      CLSTP TLST      LAMZHL
## 1  1 6.918913  920  92.0   10   NA  200 211.19414  150 66.38766
## 2  2 6.934162  850  85.0   10   NA   42  45.17696  150 33.28625
## 3  3 6.877619  800  80.0   10   NA   60  56.80923  150 36.63672
## 4  4 6.207537  465  46.5   10   NA  120 117.19292  150 72.01703
##      LAMZ LAMZLL LAMZUL LAMZNPT      CORRXY      R2      R2ADJ
## 1 0.010440904    10    150      10 -0.9943381 0.9887083 0.9872969
## 2 0.020823832    10    150      10 -0.9950967 0.9902174 0.9889946
## 3 0.018919466    10    150      10 -0.9984467 0.9968957 0.9965077
## 4 0.009624768    10    150      10 -0.9971818 0.9943716 0.9936681
##      AUCCLST  AUCALL  AUCIFO  AUCIFOD  AUCIFP  AUCIFPD  AUCPEO  AUCPEP
## 1 77590.00 77590.00 96745.43 9674.543 97817.57 9781.757 19.799828 20.67887
## 2 48374.13 48374.13 50391.05 5039.105 50543.61 5054.361  4.002536  4.29230
## 3 48430.88 48430.88 51602.22 5160.222 51433.57 5143.357  6.145738  5.83799
## 4 39677.81 39677.81 52145.65 5214.565 51853.99 5185.399 23.909634 23.48167
##      AUMCLST AUMCIFO AUMCIFP  AUMCPEO  AUMCPEP      CO  AUCPBEO  AUCPBEP
## 1 4337000 9044967 9308475 52.05068 53.40805 1058.0000 10.222705 10.110658
## 2 1967000 2366394 2396605 16.87776 17.92557 1146.8254 19.813295 19.753490
## 3 2077250 2720574 2686362 23.64661 22.67423  941.1765 16.871139 16.926460
## 4 2245250 5410815 5336766 58.50441 57.92864  540.5625  9.641865  9.696095
##      VZO      VZP      CLO      CLP MRTIVLST  MRTIVIFO  MRTIVIFP
## 1  9.899914  9.791405 0.1033641 0.1022311 55.89638  93.49244  95.16158
## 2  9.529848  9.501083 0.1984480 0.1978489 40.66223  46.96061  47.41658
## 3 10.242896 10.276482 0.1937901 0.1944256 42.89102  52.72203  52.22974
## 4 19.924695 20.036761 0.1917706 0.1928492 56.58704 103.76352 102.91908
##      VSSO      VSSP
## 1  9.663758  9.728475
## 2  9.319237  9.381321
## 3 10.217007 10.154796
## 4 19.898788 19.847860
```

```
IDs = unique(dPK01[, "ID"])
nID = length(IDs)
DOSE = 10000 # ug
```


2.1 Compartmental analysis

V, K만 있으면 단항 지수함수적 농도 감소를 보이는 IV dosing의 농도를 나타낼 수 있으므로 아래와 같이 간단한 함수를 만들 수 있습니다.

```
fPK01 = function(THETA) # Prediction function
{
  V = THETA[1]
  K = THETA[2]
  Cp = DOSE/V*exp(-K*TIME) # External DOSE, TIME, eq 1:2
  return(Cp)
}
```

여러명의 자료를 분석하기 위해 for 함수를 사용하였습니다. 복잡해보이지만 nlr 함수를 사용하는 것이 핵심입니다.

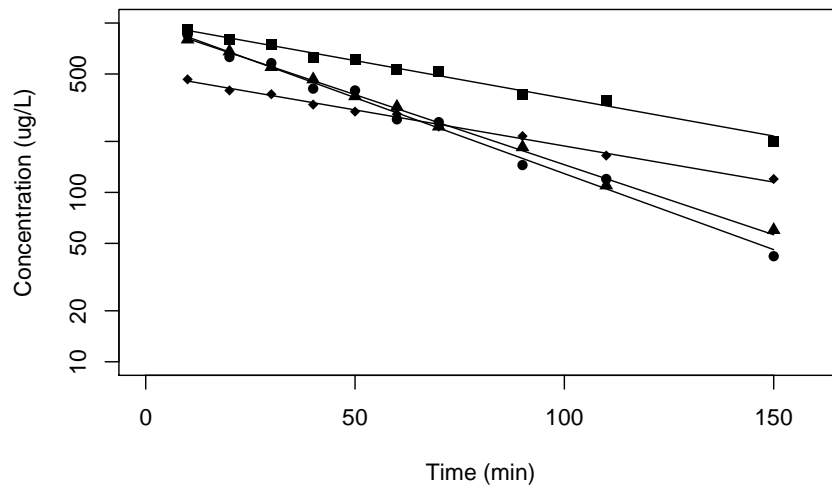
```
Result = vector()
for (i in 1:nID) {
  cID = IDs[i]
  Data = dPK01[dPK01$ID == cID,]
  TIME = dPK01[dPK01$ID == cID,"TIME"]
  Res = nlr(fPK01, Data, pNames=c("V", "k"), IE=c(20, 0.2),
            SecNames=c("CL", "AUC", "AUMC", "Thalf", "MRT"),
            SecForms=c(~V*k, ~DOSE/V/k, ~DOSE/V/k/k, ~log(2)/k, ~1/k))
  Result = rbind(Result, cbind(ID=cID, Res$Est))
} ; Result
```

	##	ID	V	k	AddErrVar	AddErrSD	CL
##	PE	1	9.9784487	0.0102560820	432.73767	20.802348	0.102339788
##	SE	1	0.1834206	0.0003873927	193.52612	4.651545	0.002597160
##	RSE	1	1.8381673	3.7771994487	44.72135	22.360674	2.537781676
##	PE	2	9.8162458	0.0206612797	753.97041	27.458522	0.202816199
##	SE	2	0.3308035	0.0010187679	337.18516	6.139900	0.005970426
##	RSE	2	3.3699589	4.9308074311	44.72127	22.360637	2.943762128
##	PE	3	10.2230093	0.0190412124	77.05108	8.777874	0.194658492
##	SE	3	0.1086744	0.0003052182	34.45832	1.962794	0.001891101
##	RSE	3	1.0630369	1.6029348653	44.72140	22.360699	0.971496569
##	PE	4	19.9471606	0.0098139570	72.19448	8.496733	0.195760575
##	SE	4	0.2954327	0.0003070959	32.28660	1.899942	0.004148218
##	RSE	4	1.4810766	3.1291754937	44.72170	22.360849	2.119026160
##			AUC	AUMC	Thalf	MRT	
##	PE		9.771371e+04	9.527391e+06	67.5840131	97.503121	
##	SE		2.479761e+03	5.875882e+05	2.5527830	3.682887	
##	RSE		2.537782e+00	6.167356e+00	3.7771994	3.777199	
##	PE		4.930573e+04	2.386383e+06	33.5481244	48.399713	

```
## SE 1.451443e+03 1.763351e+05 1.6541934 2.386497
## RSE 2.943762e+00 7.389220e+00 4.9308074 4.930807
## PE 5.137202e+04 2.697939e+06 36.4024709 52.517664
## SE 4.990774e+02 6.551250e+04 0.5835079 0.841824
## RSE 9.714966e-01 2.428243e+00 1.6029349 1.602935
## PE 5.108281e+04 5.205118e+06 70.6287162 101.895699
## SE 1.082458e+03 2.672940e+05 2.2100965 3.188495
## RSE 2.119026e+00 5.135215e+00 3.1291755 3.129175
```

2.2 Modeling Result

```
# Figure 1.1, p 470
plot(0, 1, type="n", xlim=c(0, 160), ylim=c(10, 1000), log="y", xlab="Time (min)", ylab="C")
for (i in 1:nID) {
  cID = IDs[i]
  TIME = dPK01[dPK01$ID == cID,"TIME"]
  points(TIME, dPK01[dPK01$ID == cID,"DV"], pch=14+i)
  cTHETA = Result[Result[, "ID"] == cID & rownames(Result) == "PE", c("V", "k")]
  lines(TIME, fPK01(cTHETA))
}
```



3

Two compartment model – oral dosing

```
require(wnl)
dPK14 = read.csv("data-old/PK14.csv", skip=1)
colnames(dPK14) = c("TIME", "DV") ; dPK14
```

```
##      TIME      DV
## 1  0.083  13.90
## 2  0.167 152.00
## 3  0.250 226.00
## 4  0.500 204.00
## 5  1.000 149.00
## 6  1.500 100.00
## 7  2.000  66.00
## 8  3.000  36.00
## 9  4.000  17.70
## 10 6.000   6.90
## 11 8.000   3.96
## 12 12.000  2.89
## 13 24.000  0.90
## 14 25.000  0.90
```

```
Dpo = 23158
```

3.1 Compartmental analysis without Tlag

경구 투여 2구획 분석, 지연시간이 없는 모형입니다. K_a , V , k_{21} , α , β 로 농도를 나타낼 수 있습니다.

```
## without lag
fPK14a = function(THETA)
{
  Vc = THETA[1]
  Ka = THETA[2]
  k21 = THETA[3]
```

```

a  = THETA[4] # alpha
b  = THETA[5] # beta

T1 = e$DATA[, "TIME"]
Co = Ka*Dpo/Vc*((k21-a)/(Ka-a)/(b-a)*exp(-a*T1) +
               (k21-b)/(Ka-b)/(a-b)*exp(-b*T1) +
               (k21-Ka)/(a-Ka)/(b-Ka)*exp(-Ka*T1)) # Erratum in eq 14:1

  return(Co)
}

r1 <- nlr(fPK14a, dPK14,
          pNames=c("Vc/F", "Ka", "k21", "alpha", "beta"),
          IE=c(350, 11, 1, 0.1, 0.01))

r1$Est

##          Vc/F          Ka          k21          alpha          beta AddErrVar
## PE  43.63080  2.349584    0.6979441    2.346109    0.3973269  537.19125
## SE  40.60547  2.222099    0.9644243    2.221849    0.4937935  203.03693
## RSE 93.06606 94.574121  138.1807351  94.703582  124.2789064   37.79602
##      AddErrSD
## PE   23.177387
## SE    4.380065
## RSE  18.898011

```

3.2 Compartmental analysis with Tlag

경구 투여 일구획 분석, 지연시간이 있는 모형입니다. Ka, V, k21, alpha, beta 에 Tlag가 추가되어 농도를 나타낼 수 있습니다.

```

## with lag
fPK14b = function(THETA)
{
  Vc  = THETA[1]
  Ka  = THETA[2]
  k21 = THETA[3]
  a   = THETA[4] # alpha
  b   = THETA[5] # beta
  TL  = THETA[6] # Tlag

  T1 = e$DATA[, "TIME"]

```

```

Co = Ka*Dpo/Vc*((k21-a)/(Ka-a)/(b-a)*exp(-a*(T1-TL)) +
              (k21-b)/(Ka-b)/(a-b)*exp(-b*(T1-TL)) +
              (k21-Ka)/(a-Ka)/(b-Ka)*exp(-Ka*(T1-TL)))
Co[Co < 0] = 0 # remove negative concentrationss before tlag

return(Co)
}

r2 <- nlr(fPK14b, dPK14,
         pNames=c("Vc/F", "Ka", "k21", "alpha", "beta", "Tlag"),
         IE=c(150, 11, 0.12, 0.1, 0.01, 0.05))
r2$Est

##          Vc/F          Ka          k21          alpha          beta          Tlag
## PE  88.530347 19.777685    0.0079704 0.73178583 2.021746e-03 0.121649379
## SE   2.318778  3.774496    0.1245099 0.04591788 1.130516e-01 0.008133633
## RSE  2.619190 19.084619 1562.1537739 6.27477041 5.591779e+03 6.686127672
##      AddErrVar  AddErrSD
## PE  17.889619  4.2296122
## SE   6.848254  0.8095605
## RSE 38.280600 19.1403000

```

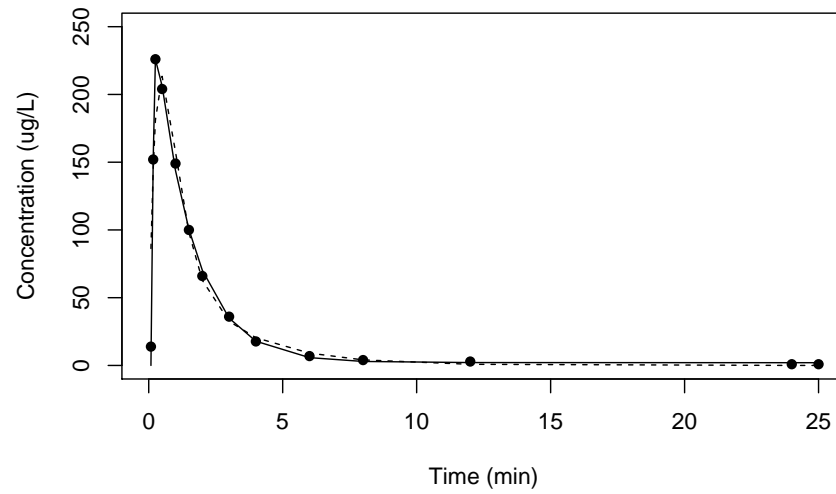
3.3 Modeling Result

지연 시간이 있는 모형의 적합이 더 좋은 것을 알 수 있습니다. Cmax 부분을 주의깊게 살펴보세요.

```

plot(dPK14[, "TIME"], dPK14[, "DV"], xlim=c(0, 25), ylim=c(0, 250), xlab="Time (min)", ylab=
TIME = dPK14[, "TIME"]
lines(TIME, fPK14a(r1$Est["PE", 1:5]), lty=2)
lines(TIME, fPK14b(r2$Est["PE", 1:6]))

```



Bae, Kyun-Seop. 2018. *Wnl: Minimization Tool for Pharmacokinetic-Pharmacodynamic Data Analysis*. <https://CRAN.R-project.org/package=wnl>.

Gabrielsson, Johan. 2016. *Pharmacokinetic and Pharmacodynamic Data Analysis : Concepts and Applications*. Stockholm: Apotekarsocieteten.