tetra1.R:

```
tetra <- read.table(file = "n:/courses/STAT8230/Fall09/tetracycline.dat", header = T)

plot(tetra$time, tetra$conc, xlab = "Time (hrs)", ylab =
   "Tetracycline Concentration (mug/ml)")
title(main = "Tetracycline Concentration vs. Time")
tetramod <- deriv( ~ (exp(t3) * exp(t1) * (exp( - x * exp(t1)) - exp( - x *
   exp(t2))))/
   (exp(t2) - exp(t1)),c("t1","t2","t3"),function(x,t1,t2,t3){})

t10<-log(.2)
t20<-log(.4)
t30<-log(6)
analyticeval <- tetramod(tetra$time,t10,t20,t30)

source("n:/courses/STAT8230/Fall09/compmodel.R")
J <- matrix(c(1,1,2,2,2,0,3,1,-1),nrow=3,byrow=T)
J
th <- c(log(.2),log(.4),log(6))
gamma0 <- c(0,0)
#formcompmodel(J,th,gamma0)

matrixexptialeval <- compmodel(J,th,gamma0,tetra$time,2)

analyticeval[1:4]
matrixexptialeval[1:4]

attr(analyticeval,"gradient")[1:4,]
attr(matrixexptialeval,"gradient")[1:4,]

#now fit with analytic solution
m1tetra.nls <- nls(conc ~ tetramod(time,t1,t2,t3),data=tetra,
 start = c(t1 = t10, t2 = t20, t3 = t30))
t0 <- seq(from = 1, to = 16, by = 0.1)
that <- coef(m1tetra.nls)
y0 <- tetramod(t0,that[1],that[2],that[3])
lines(t0, y0, lty = 2)

#now fit the matrix exponential method

m2tetra.nls <- nls(conc ~ compmodel(J,c(t1,t2,t3),c(0,0),time,2),data=tetra,
 start = c(t1 = t10, t2 = t20, t3 = t30))
# the previous two lines produces an error, but if we set up the data set
# as a list, it works:

datalist <- as.list(tetra)
datalist$J <- J
datalist$gamma0 <- c(0,0)
datalist

m2tetra.nls <- nls(conc ~ compmodel(J,c(t1,t2,t3),gamma0,time,2),data=datalist,
 start = c(t1 = t10, t2 = t20, t3 = t30))
coef(m1tetra.nls)
coef(m2tetra.nls)
summary(m1tetra.nls)
summary(m2tetra.nls)
logLik(m1tetra.nls)


J<-matrix(c(1,1,2,2,2,0,3,1,-1,4,0,0),nrow=4,ncol=3,byrow=T)
J
datalist$J <- J
```

```
m3tetra.nls <- nls(conc ~ compmodel(J,c(t1,t2,t3,t4),gamma0,time,2),data=datalist,
 start = c(t1 = t10, t2 = t20, t3 = t30, t4 = 0))
coef(m3tetra.nls)

that <- coef(m3tetra.nls)
y1 <- compmodel(J,that,c(0,0),t0,2)
lines(t0, y1, lty = 3)
legend(x=6,y=1.4,lty=c(2,3),legend=c("2-Compartment model w/o Dead Time",
 "2-Compartment model w/ Dead Time"))

anova(m3tetra.nls,m2tetra.nls)
plot(resid(m3tetra.nls)~fitted(m3tetra.nls))
abline(h=0)
title(main="Residuals vs. Fitteds, 2-Comp. Model w/ Dead Time")
```

Output from tetra1.R:

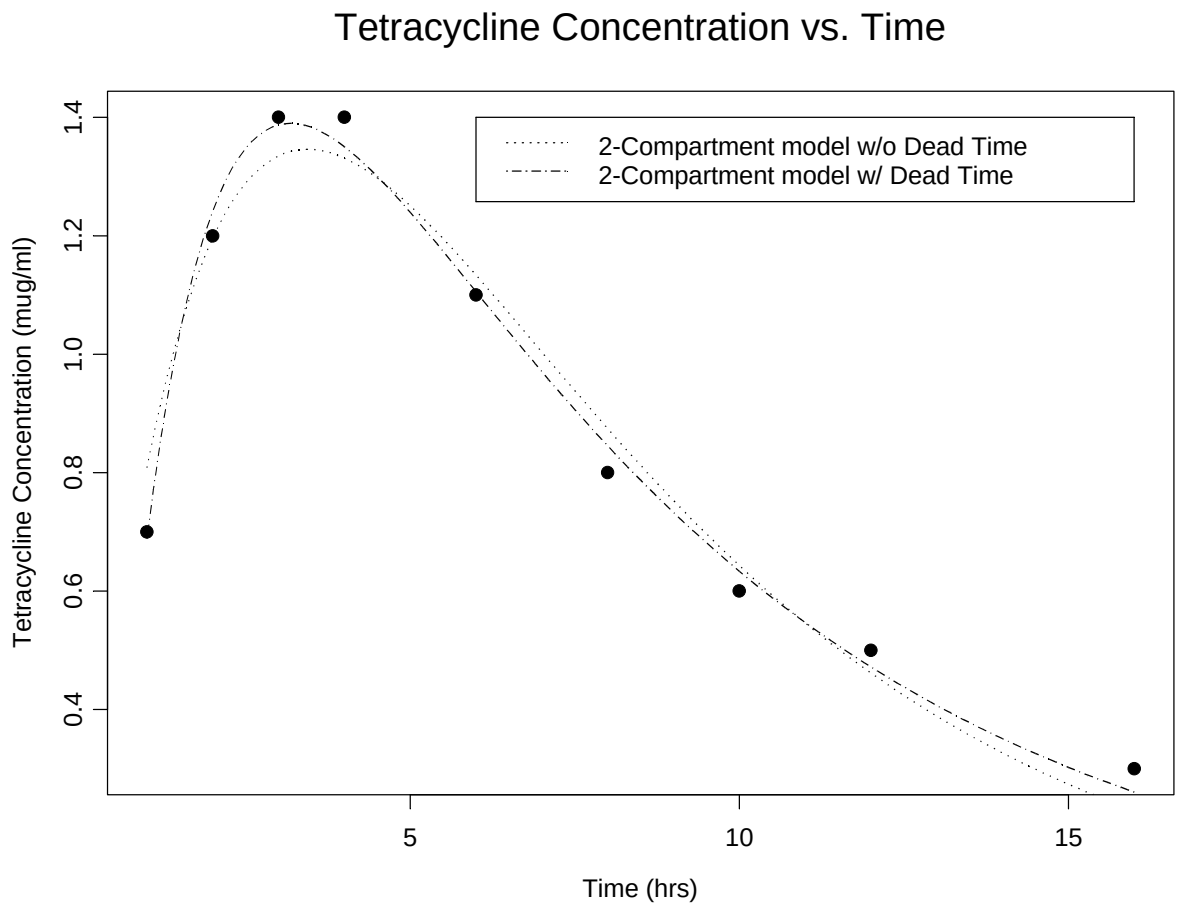
```
> tetra <- read.table(file = "n:/courses/STAT8230/Fall09/tetracycline.dat", header =
T)
>
> plot(tetra$time, tetra$conc, xlab = "Time (hrs)", ylab =
+ "Tetracycline Concentration (mug/ml)")
> title(main = "Tetracycline Concentration vs. Time")
> tetramod <- deriv( ~ (exp(t3) * exp(t1) * (exp( - x * exp(t1)) - exp( - x *
exp(t2))))/
+ (exp(t2) - exp(t1)),c("t1","t2","t3"),function(x,t1,t2,t3){})
>
> t10<-log(.2)
> t20<-log(.4)
> t30<-log(6)
> analyticeval <- tetramod(tetra$time,t10,t20,t30)
>
> source("n:/courses/STAT8230/Fall09/compmodel.R")
> J <- matrix(c(1,1,2,2,2,0,3,1,-1),nrow=3,byrow=T)
> J
     [,1] [,2] [,3]
[1,]    1    1    2
[2,]    2    2    0
[3,]    3    1   -1
> th <- c(log(.2),log(.4),log(6))
> gamma0 <- c(0,0)
> #formcompmodel(J,th,gamma0)
>
> matrixexptialeval <- compmodel(J,th,gamma0,tetra$time,2)
>
> analyticeval[1:4]
[1] 0.8904642 1.3259465 1.4857045 1.4845947
> matrixexptialeval[1:4]
[1] 0.8904642 1.3259465 1.4857045 1.4845947
>
> attr(analyticeval,"gradient")[1:4,]
           t1         t2        t3
[1,] 0.7984516 -0.1721604 0.8904642
[2,] 1.0431249 -0.4951140 1.3259465
```

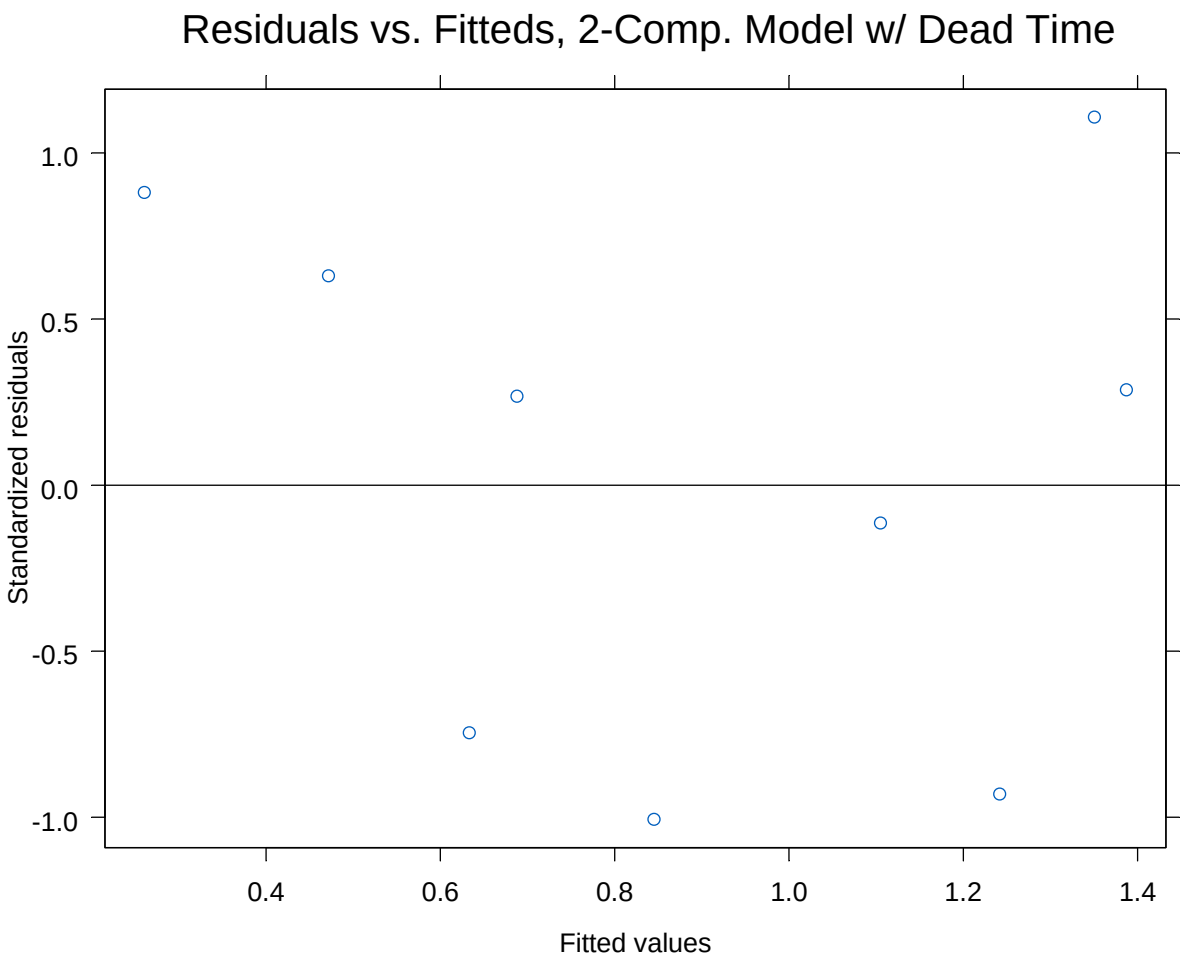
```
[3,] 0.9956872 -0.8028108 1.4857045
[4,] 0.8124103 -1.0309828 1.4845947
> attr(matrixexptialeval,"gradient")[1:4,]
          [,1]       [,2]      [,3]
[1,] 0.7984516 -0.1721604 0.8904642
[2,] 1.0431249 -0.4951140 1.3259465
[3,] 0.9956872 -0.8028108 1.4857045
[4,] 0.8124103 -1.0309828 1.4845947
>
> #now fit with analytic solution
> m1tetra.nls <- nls(conc ~ tetramod(time,t1,t2,t3),data=tetra,
+  start = c(t1 = t10, t2 = t20, t3 = t30))
> t0 <- seq(from = 1, to = 16, by = 0.1)
> that <- coef(m1tetra.nls)
> y0 <- tetramod(t0,that[1],that[2],that[3])
> lines(t0, y0, lty = 2)
>
> #now fit the matrix exponential method
>
> m2tetra.nls <- nls(conc ~ compmodel(J,c(t1,t2,t3),c(0,0),time,2),data=tetra,
+  start = c(t1 = t10, t2 = t20, t3 = t30))
Error in model.frame.default(formula = ~conc + J + time, data = tetra) :
  variable lengths differ (found for 'J')
> # the previous two lines produces an error, but if we set up the data set
> # as a list, it works:
>
> datalist <- as.list(tetra)
> datalist$J <- J
> datalist$gamma0 <- c(0,0)
> datalist
$time
[1]  1  2  3  4  6  8 10 12 16

$conc
[1] 0.7 1.2 1.4 1.4 1.1 0.8 0.6 0.5 0.3

$J
     [,1] [,2] [,3]
[1,]    1    1    2
[2,]    2    2    0
[3,]    3    1   -1

$gamma0
[1] 0 0

>
> m2tetra.nls <- nls(conc ~ compmodel(J,c(t1,t2,t3),gamma0,time,2),data=datalist,
+  start = c(t1 = t10, t2 = t20, t3 = t30))
> coef(m1tetra.nls)
        t1         t2         t3
-1.6981391 -0.8335924  1.7909761
> coef(m2tetra.nls)
        t1         t2         t3
-1.6981391 -0.8335924  1.7909761
> summary(m1tetra.nls)
```

```
Formula: conc ~ tetramod(time, t1, t2, t3)

Parameters:
   Estimate Std. Error t value Pr(>|t|)
t1  -1.6981     0.2440  -6.961 0.000437 ***
t2  -0.8336     0.2724  -3.060 0.022217 *
t3   1.7910     0.3180   5.632 0.001340 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.07708 on 6 degrees of freedom

Number of iterations to convergence: 7
Achieved convergence tolerance: 6.495e-06

> summary(m2tetra.nls)

Formula: conc ~ compmodel(J, c(t1, t2, t3), gamma0, time, 2)

Parameters:
   Estimate Std. Error t value Pr(>|t|)
t1  -1.6981     0.2440  -6.961 0.000437 ***
t2  -0.8336     0.2724  -3.060 0.022217 *
t3   1.7910     0.3180   5.632 0.001340 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.07708 on 6 degrees of freedom

Number of iterations to convergence: 7
Achieved convergence tolerance: 6.495e-06

> logLik(m1tetra.nls)
'log Lik.' 12.12050 (df=4)
>
>
> J<-matrix(c(1,1,2,2,2,0,3,1,-1,4,0,0),nrow=4,ncol=3,byrow=T)
> J
     [,1] [,2] [,3]
[1,]    1    1    2
[2,]    2    2    0
[3,]    3    1   -1
[4,]    4    0    0
> datalist$J <- J
>
> m3tetra.nls <- nls(conc ~ compmodel(J,c(t1,t2,t3,t4),gamma0,time,2),data=datalist,
+  start = c(t1 = t10, t2 = t20, t3 = t30, t4 = 0))
> coef(m3tetra.nls)
        t1         t2         t3         t4
-1.9051465 -0.3344220  2.3120895  0.4122367
>
> that <- coef(m3tetra.nls)
> y1 <- compmodel(J,that,c(0,0),t0,2)
> lines(t0, y1, lty = 3)
```

```
> legend(x=6,y=1.4,lty=c(2,3),legend=c("2-Compartment model w/o Dead Time",
+  "2-Compartment model w/ Dead Time"))
>
> anova(m3tetra.nls,m2tetra.nls)
Analysis of Variance Table

Model 1: conc ~ compmodel(J, c(t1, t2, t3, t4), gamma0, time, 2)
Model 2: conc ~ compmodel(J, c(t1, t2, t3), gamma0, time, 2)
  Res.Df Res.Sum Sq Df    Sum Sq F value  Pr(>F)
1      5   0.010045
2      6   0.035647 -1 -0.025601  12.743 0.01605 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> plot(resid(m3tetra.nls)~fitted(m3tetra.nls))
> abline(h=0)
> title(main="Residuals vs. Fitteds, 2-Comp. Model w/ Dead Time")
>
>
```

Plots from tetra1.R:




Residuals vs. Fitteds, 2-Comp. Model w/ Dead Time
Standardized residuals
1.0
0.5
0.0
-0.5
-1.0
0.4
0.6
0.8
1.0
1.2
1.4
Fitted values