Dose-Time Response Modeling of Rat Brain AChE Activity: Carbaryl Dermal Exposure

June 28, 2005

1 Preamble

Here is some code to set up the analysis: loading required libraries and datasets, and defining some functions.

First, `CarbamateData` loads the full dataset for this risk assessment, and causes the library `DRUtils` to be loaded.

```
> library(CarbamateData)

Loading required package: DRUtils
Loading required package: nlme
Loading required package: tcltk
```

Set up lattice to use B&W instead of color:

```
> library(lattice)
> ltheme <- canonical.theme(color = FALSE)
> ltheme$strip.background$col <- "transparent"
> lattice.options(default.theme = ltheme)
```

Use package `Hmisc` for some formatting support.

```
> library(Hmisc)

Hmisc library by Frank E Harrell Jr

Type library(help='Hmisc'), ?Overview, or ?Hmisc.Overview')
to see overall documentation.

NOTE:Hmisc no longer redefines [.factor to drop unused levels when
subsetting.  To get the old behavior of Hmisc type dropUnusedLevels().

Attaching package 'Hmisc':


        The following object(s) are masked from package:stats :

         ecdf
```

All the dermal data for this analysis are in `AggData`. The following code prints out documentation for the datasets in use:

```
> printDataDoc(AggData)
```

```
----------------------------------------
Data set: AggData

Dataset creation date: Wed Jun 22 12:07:24 2005
Script name: /home/setzer/tasks/CarbamateCRA/EmpiricalDoseResponsesFinal/Data/getdata.R
Script last modified: 2005-06-22 12:05:47
sysname: Linux
release: 2.4.24-openmosix2
version: #1 Wed Jul 28 14:33:52 CEST 2004
nodename: node24.cluster2.org
machine: i686
login: unknown
user: setzer
----------------------------------------
```

The following function turns out to be quite useful on subsetted dataframes. It just eliminates unused levels of all factors in the data frame:

```
> CleanUp <- function(x) {
+     for (nm in names(x)) {
+         if (is.factor(x[, nm]))
+             x[, nm] <- factor(x[, nm])
+     }
+     x
+ }
```

To get starting values, we often have to extract values from a previously fit model. The following function simplifies that. The argument `what` is a regular expression:

```
> getParms <- function(what, Par) {
+     Par[grep(what, names(Par))]
+ }
```

This script is for modeling the dose-time response for rat brain. via gavage dosing. It includes acute, subchronic, and chronic studies.

All the data used for this DR model are in `AggData`.

These are all the names used for the brain sections we want from `AggData`:

```
> BrainSections <- levels(AggData$sctn)[grep("(^BRAIN)|(^WHOLEBRAIN)|(LEFT HEMISPHERE)| (HALFBRAIN)",
+     levels(AggData$sctn))]
```

Now, set up the analysis dataset.

```
> dta <- CleanUp(subset(AggData, chemical %in% "CARBARYL" & dsmtd %in%
+     "DERMAL" & sctn %in% BrainSections & !is.na(cheact) & (n ==
+     1 | !is.na(sd)) & (is.na(ta) | ta %in% c("AI")), select = c("cheact",
+     "sd", "n", "dose", "tmonstdy", "tmpstds", "sex", "mrid",
+     "species")))
```

Summary of the relevant variables in this dataset:

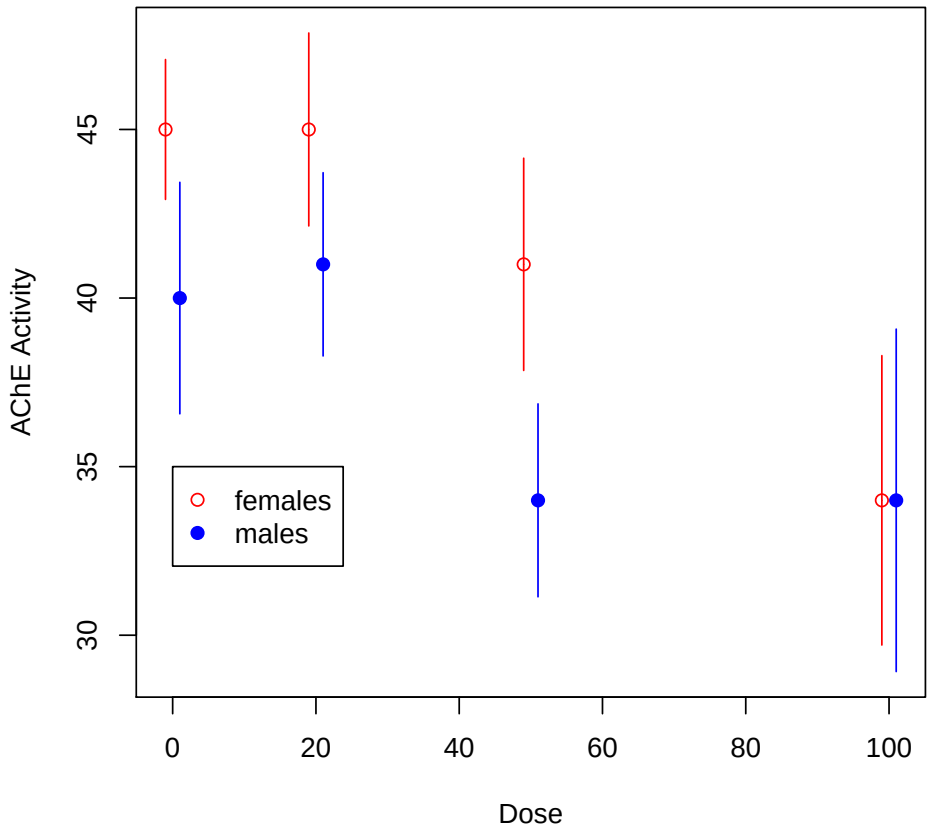
```
> summary(dta)

     cheact            sd              n           dose          tmonstdy
 Min.   :34.00   Min.   :2.900   Min.   :10   Min.   :  0.0   Min.   :26
 1st Qu.:34.00   1st Qu.:3.950   1st Qu.:10   1st Qu.: 15.0   1st Qu.:26
 Median :40.50   Median :4.200   Median :10   Median : 35.0   Median :26
```

```
 Mean   :39.25   Mean   :4.625   Mean   :10   Mean   : 42.5   Mean   :26
 3rd Qu.:42.00   3rd Qu.:5.100   3rd Qu.:10   3rd Qu.: 62.5   3rd Qu.:26
 Max.   :45.00   Max.   :7.100   Max.   :10   Max.   :100.0   Max.   :26
    tmpstds  sex         mrid   species
 Min.   :1   F:4   45630601:8   RAT:8
 1st Qu.:1   M:4
 Median :1
 Mean   :1
 3rd Qu.:1
 Max.   :1

> unique(subset(dta, select = c("tmonstdy", "tmpstds")))

    tmonstdy tmpstds
592       26       1

> with(dta, table(dose, sex))

     sex
dose  F M
  0   1 1
  20  1 1
  50  1 1
  100 1 1

> with(dta, table(dose, tmpstds))

     tmpstds
dose  1
  0   2
  20  2
  50  2
  100 2
```

2 Dose-Response Modeling

2.1 strategy

Compared to the gavage data sets, this one is simple: four doses at one time in each of males and females:



Error bars represent 95% confidence intervals for the means. The data from both sexes agree that there is a shoulder at the low-dose end of the dose-response curve, but disagree about whether the high dose has already hit a horizontal asymptote. Partly for convenience, I will assume that the female data are closer to the true curve shape (thus, the data for the middle dose in males are a bit low, and those for the high dose are a bit high), and fit `tcmfn4()` with no horizontal asymptote, but estimating `lg`. Fit the model with a homogeneous variance. Standard deviations look fairly constant except at the highest dose.

2.2 Initial Values

Save the initial values so that we do not need to go through all this to re-run the analysis. Also, set the argument `delta` to 0, `tz` to -10, `lTr` to log(1.5), and time to 0. This forces just a dose-response curve.

```
> formals(tcmfn4)$delta <- 1
> formals(tcmfn4)$tz <- -10
> formals(tcmfn4)$time <- 1.001
> formals(tcmfn4)$lTr <- log(1.5)
> initfile <- paste("initvals-brain-DR-1.RData", sep = "")
> if (!file.exists(initfile)) {
+     lA.start <- lm(I(log(cheact)) ~ sex - 1, data = CleanUp(subset(dta,
+         dose %in% 0)))
+     Start <- c(coef(lA.start), log(50), 0, log(3))
+     init1 <- GetInitialValues(cbind(resp = cheact, sd = sd, n = n) ~
```

```
+         tcmfn4(dose, lA = lA, lD = lD, lg = lg), data = dta,
+         params = list(lA ~ sex - 1, lD ~ sex, lg ~ 1), start = Start)
+     save(init1, file = initfile)
+ } else load(initfile)
```

Now fit it:

```
> Start <- init1$start$beta
> idta <- init1$data
> drmod1 <- gnls(cheact ~ tcmfn4(dose, lA = lA, lD = lD, lg = lg),
+     data = idta, params = list(lA ~ sex - 1, lD ~ sex, lg ~ 1),
+     start = Start)
```

Does the BMD differ between sexes?

```
> summary(drmod1)$tTable
```

```
                   Value  Std.Error     t-value      p-value
lA.sexF        3.8250453 0.03025523 126.4259408 3.671080e-89
lA.sexM        3.6897331 0.03055084 120.7735303 1.115414e-87
lD.(Intercept) 3.8186747 0.31617900  12.0775723 2.884227e-19
lD.sexM        0.3076327 0.28636441   1.0742699 2.861458e-01
lg             0.2552564 0.34762215   0.7342927 4.650599e-01
```

No (the P-value for the term `lD.sexM` is 0.29), so refit with a common estimate for `lD`:

```
> Start <- coef(drmod1)
> Start <- Start[-grep("lD.sexM", names(Start))]
> drmod2 <- gnls(cheact ~ tcmfn4(dose, lA = lA, lD = lD, lg = lg),
+     data = idta, params = list(lA ~ sex - 1, lD ~ 1, lg ~ 1),
+     start = Start)
```

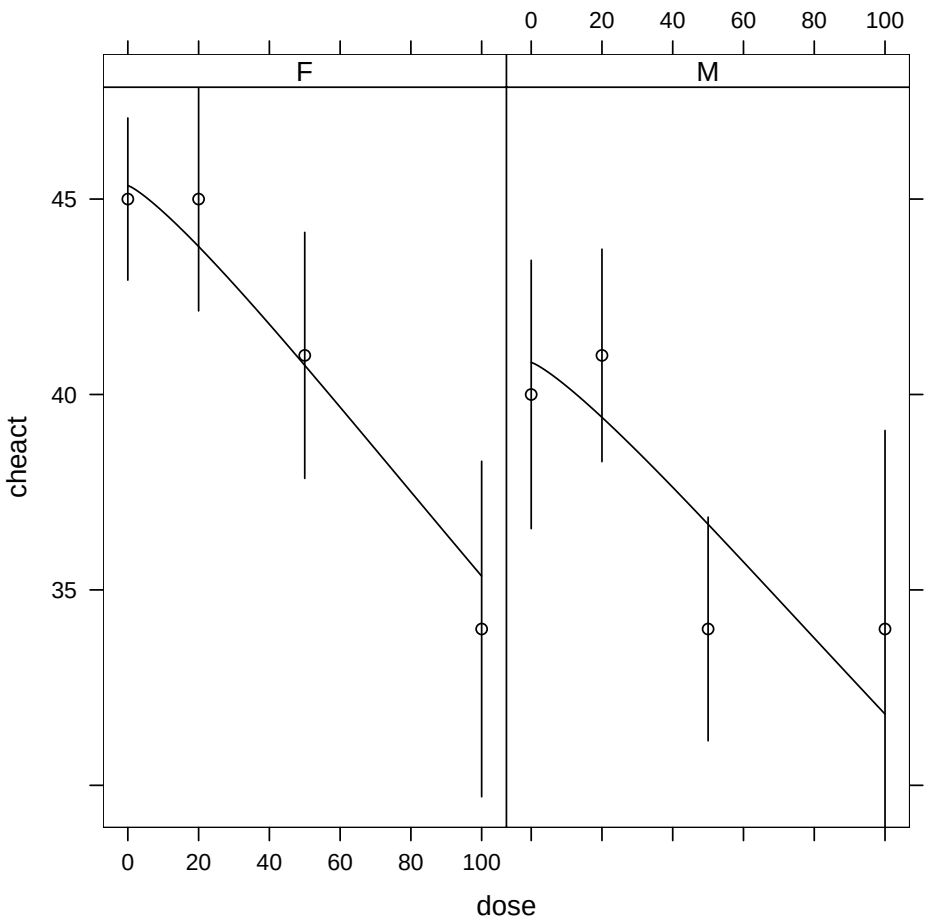
Summaries of this fit:

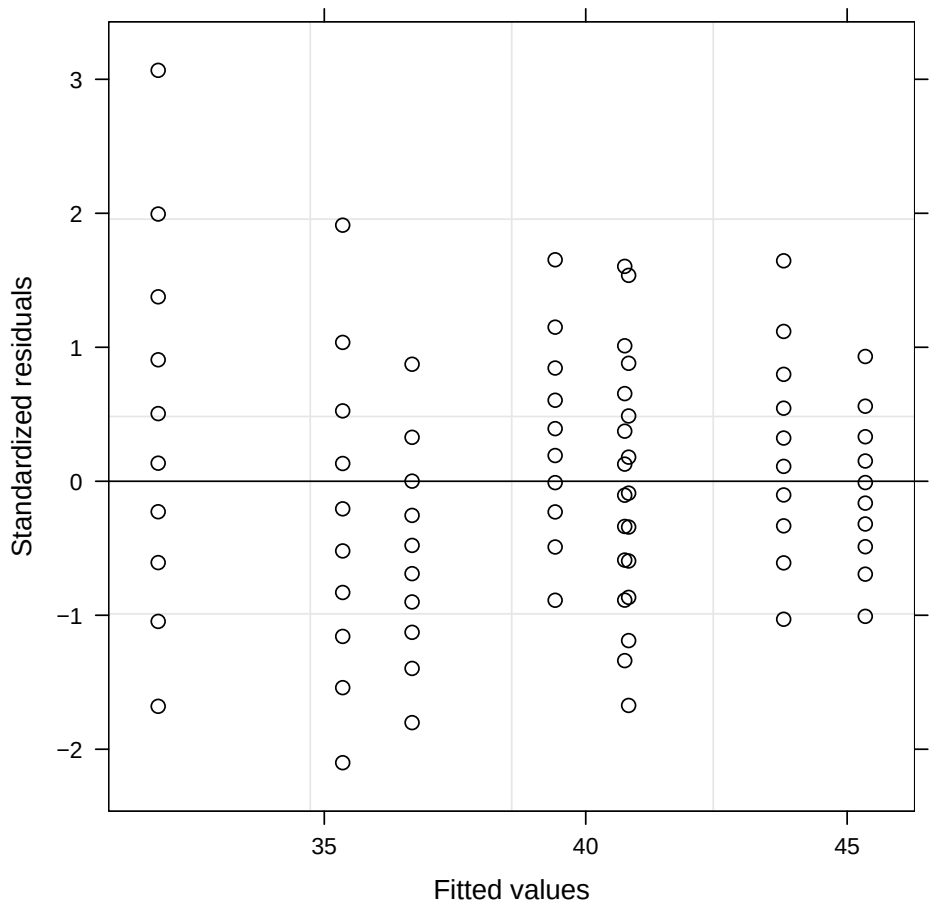
```
> tTab <- summary(drmod2)$tTable
> Ints <- intervals(drmod2, which = "coef")$coef
> Ints90 <- intervals(drmod2, which = "coef", level = 0.9)$coef
> summary(drmod2)
```

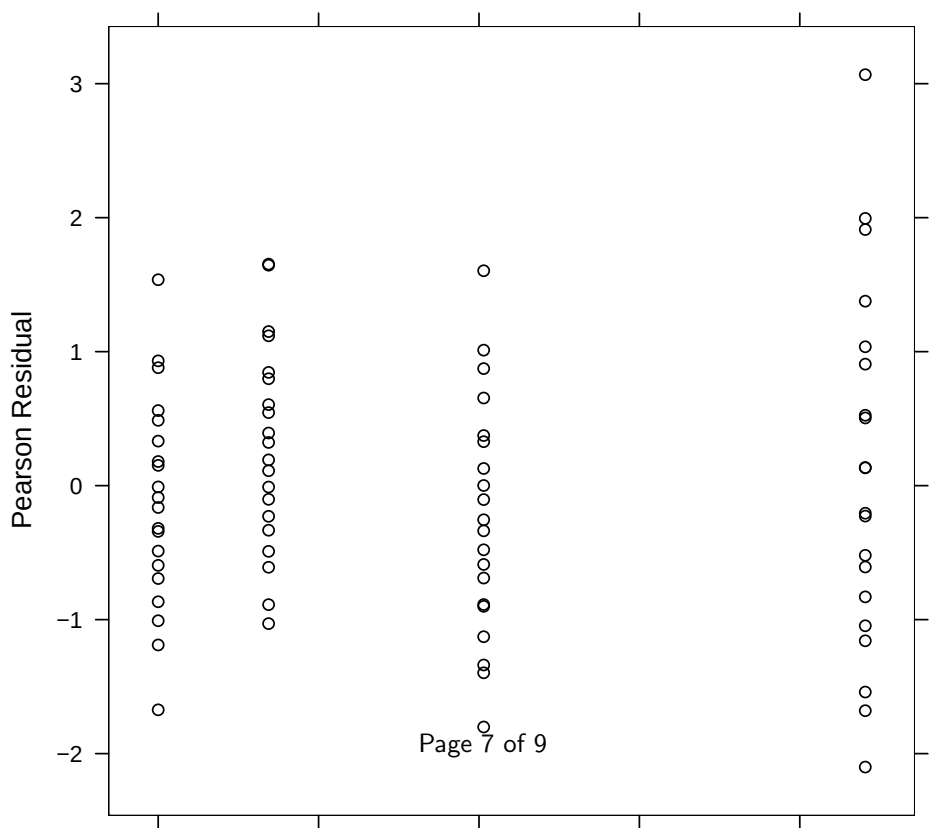
```
Generalized nonlinear least squares fit
  Model: cheact ~ tcmfn4(dose, lA = lA, lD = lD, lg = lg)
  Data: idta
       AIC      BIC    logLik
  487.8166 499.7267 -238.9083

Coefficients:
           Value Std.Error   t-value p-value
lA.sexF 3.814343 0.0273698 139.36340  0.0000
lA.sexM 3.709185 0.0288336 128.64114  0.0000
lD      3.899003 0.2878855  13.54359  0.0000
lg      0.197764 0.3486359   0.56725  0.5722

 Correlation:
        lA.sxF lA.sxM lD
lA.sexM  0.504
lD      -0.710 -0.674
lg      -0.561 -0.533  0.894
```

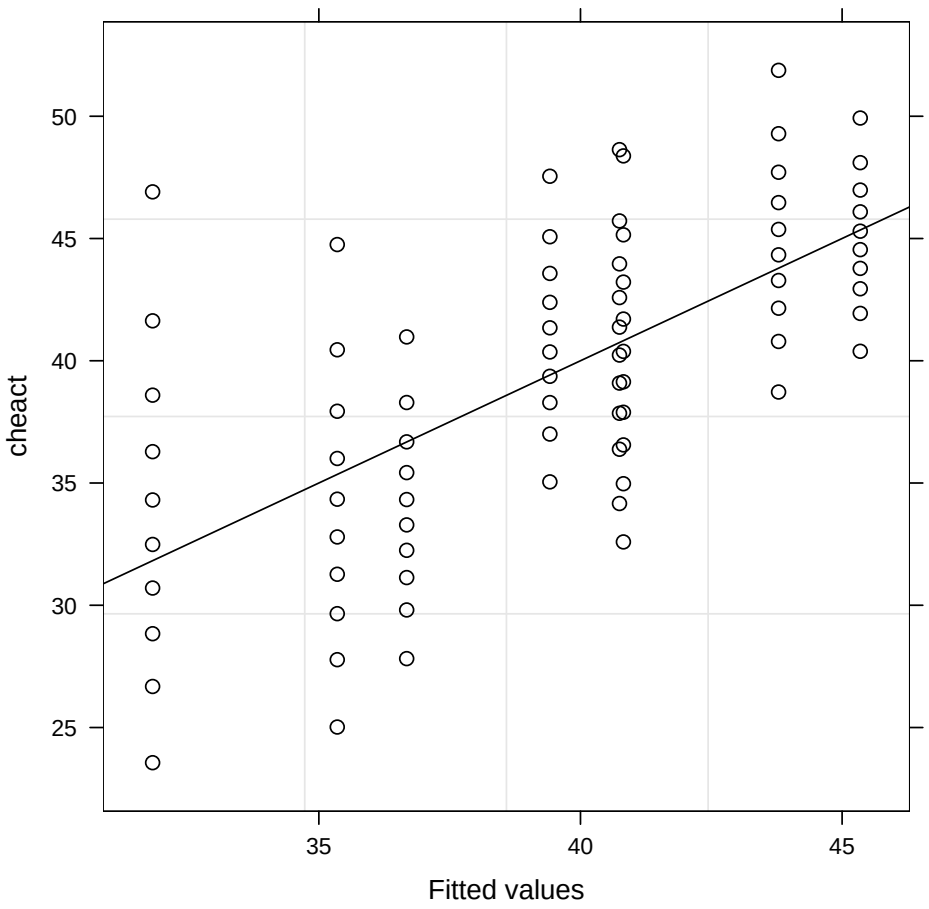
```
Standardized residuals:
        Min          Q1         Med          Q3         Max
-2.10028211 -0.62910352 -0.04937087  0.57141576  3.06708813

Residual standard error: 4.918787
Degrees of freedom: 80 total; 76 residual
```

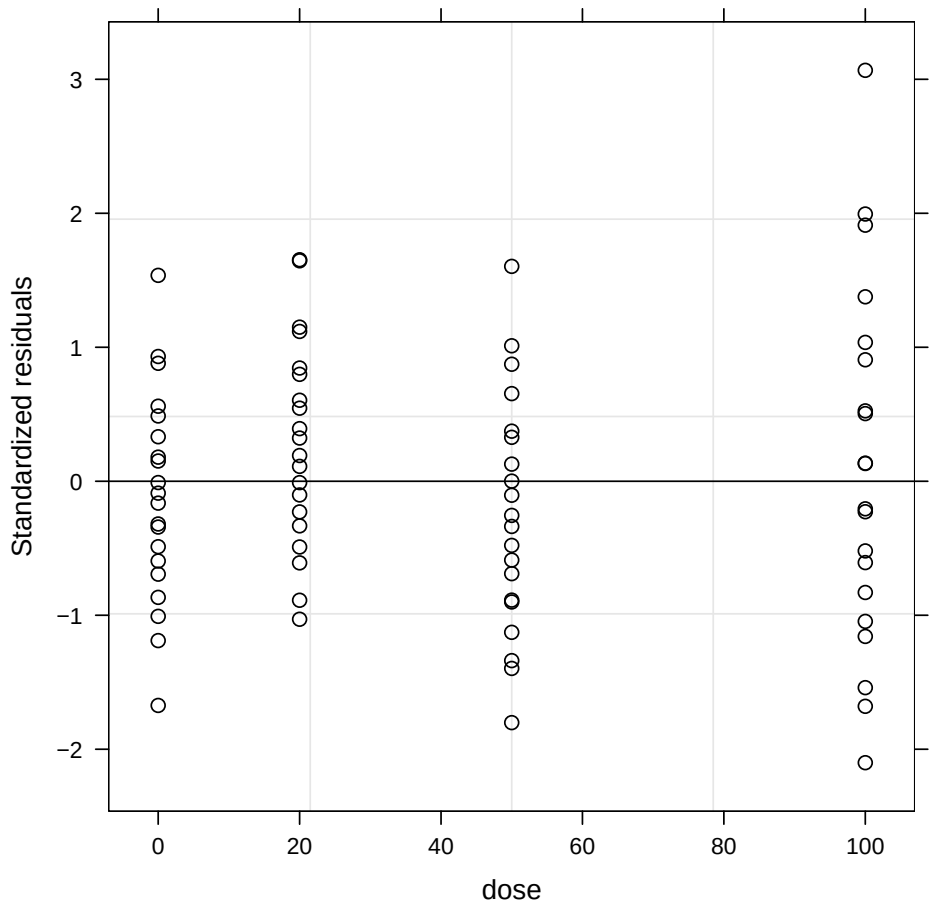
Diagnostic graphics for the model contained in `drmod2`. First, data and model fit, together:




Standardized residuals
Fitted values
3
2
1
0
−1
−2
35
40
45


Pearson Residual
3
2
1
0
−1
−2


cheact
50
45
40
35
30
25
35
40
45
Fitted values



sectionSummary The critical estimates from this analysis are listed below. They are printed with greater than usual precision, in case they are to be used in further computation. For reporting, round to two or three significant digits.

species RAT

mrid `[1] "45630601"`

`lD` **(standard error)** 3.89900293569791 (0.287885481956026)

`BMD` **(95% confidence interval)** 49.3532162353385,(27.8164982409317, 87.5645788220713)

BMDL, the one-sided lower 95% CL 30.5581075234853

Save everything:

```
> save.image(file = "DermalBrainDR.RData")
```

Save the results for incorporating into a database:

```
> carbaryl.dermal.brain <- list(mrid = levels(idta$mrid), species = levels(dta$species),
+     BMD = list(combined = list(lD = tTab["lD", 1], lD.se = tTab["lD",
+         2], BMD = exp(Ints["lD", 2]), BMD.CI = exp(Ints["lD",
+         c(1, 3)]), BMDL = exp(Ints90["lD", 1]))))
> save(carbaryl.dermal.brain, file = file.path("..", "..", "01Summaries",
+     "carbaryl.dermal.brain.RData"))
```