Dose-Time Response Modeling of Rabbit RBC AChE Activity: Propoxur Dermal Dosing

June 23, 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
```

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

```
> printDataDoc(repeated_dermal)
```

```
----------------------------------------
Data set: repeated_dermal

Dataset creation date: Wed Jun 22 12:07:28 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 rabbit RBC. via dermal dosing.

Set up a dataset for plotting and a dataset for analysis by repeated measures.

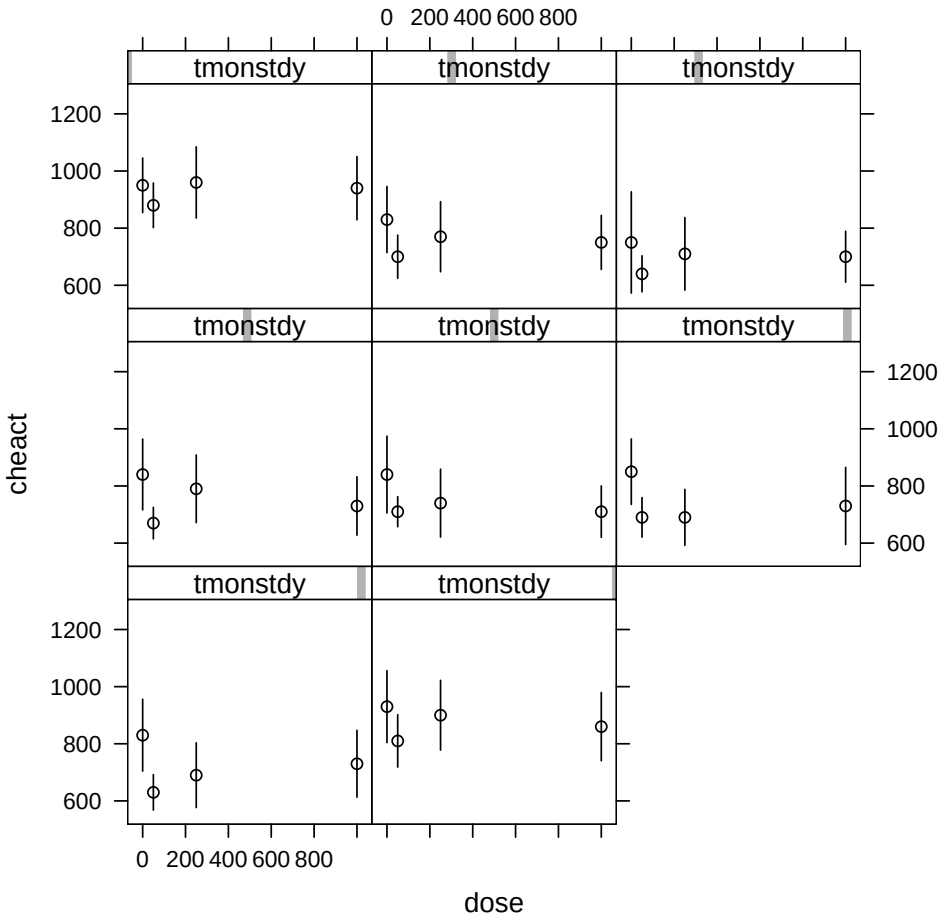
```
> dta <- CleanUp(subset(AggData, chemical %in% "PROPOXUR" & dsmtd %in%
+     "DERMAL" & sctn %in% "RBC" & !is.na(cheact), select = c("dose",
+     "cheact", "n", "sd", "tmonstdy", "sex", "tmpstds", "species",
+     "mrid")))
> idta <- CleanUp(subset(repeated_dermal, chemical %in% "propoxur"))
```

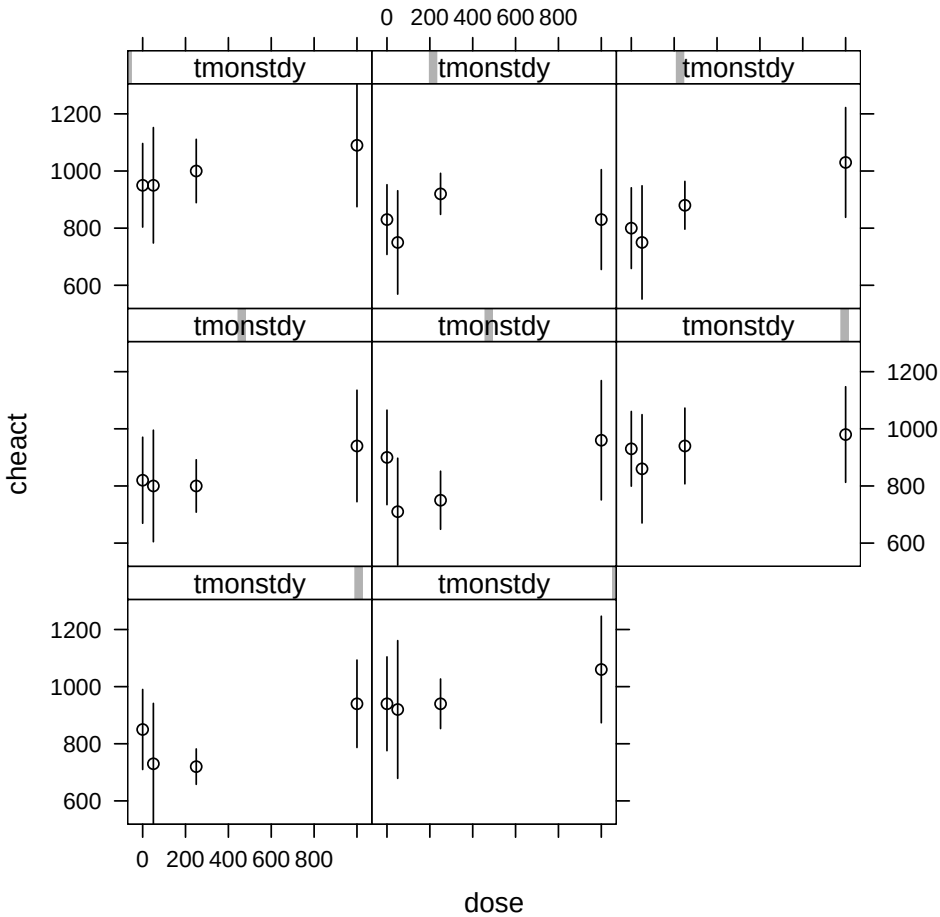
Summary of this dataset:

```
> summary(dta)

      dose            cheact             n            sd           tmonstdy
 Min.   :   0.0   Min.   : 630.0   Min.   :10   Min.   : 85.0   Min.   :-1.00
 1st Qu.:  37.5   1st Qu.: 730.0   1st Qu.:10   1st Qu.:153.8   1st Qu.:27.50
 Median : 150.0   Median : 830.0   Median :10   Median :199.0   Median :43.50
 Mean   : 325.0   Mean   : 831.6   Mean   :10   Mean   :210.9   Mean   :50.12
 3rd Qu.: 437.5   3rd Qu.: 932.5   3rd Qu.:10   3rd Qu.:267.8   3rd Qu.:86.00
 Max.   :1000.0   Max.   :1090.0   Max.   :10   Max.   :389.0   Max.   :91.00

 sex       tmpstds      species         mrid
 F:32   Min.   : NA   RABBIT:64   41066001:64
 M:32   1st Qu.: NA
```

```
        Median : NA
        Mean   :NaN
        3rd Qu.: NA
        Max.   : NA
        NA's   : 64

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

, , sex = F

      tmonstdy
dose   -1 22 23 29 30 42 43 44 45 85 86 87 91
  0     1  0  0  1  1  0  0  1  1  0  1  1  1
  50    1  0  0  1  1  0  0  1  1  0  1  1  1
  250   1  0  0  1  1  0  0  1  1  0  1  1  1
  1000  1  0  0  1  1  0  0  1  1  0  1  1  1

, , sex = M

      tmonstdy
dose   -1 22 23 29 30 42 43 44 45 85 86 87 91
  0     1  1  1  0  0  1  1  0  0  1  1  0  1
  50    1  1  1  0  0  1  1  0  0  1  1  0  1
  250   1  1  1  0  0  1  1  0  0  1  1  0  1
  1000  1  1  1  0  0  1  1  0  0  1  1  0  1
```

Some dose-response plots for each value of `tmonstdy`, females first, then males:


tmonstdy
cheact
dose
0 200 400 600 800
600
800
1000
1200



2 Dose-Response Modeling

2.1 strategy

Given the above figures, it looks as if there may be no dose-response at all. However, the raw means as plotted do not adjust for individual-specific differences, variation of which among dose groups might help mask a dose-response. So, first, re-do the above plots, this time plotting the effects of dose as fraction of control activity, versus dose, and do the estimation in a way that accounts for interindividual variability. This can be easily accomplished using linear mixed effects models of log cholinesterase activity, which corresponds to a simple exponential model of cholinesterase activity. Since some of these curves look as if they have significant *positive* slopes, do not try to constrain the slopes in this analysis: if the slopes are significantly positive, it is probably telling us that something has gone wrong with the study, and we need to know it. So, we need to:

- add log(cheact) = `lncheact` to the data set
- add `dose2`, a variable that is 0 until dosing begins
- add a factor that picks out all the unique combinations of sex, time, and dose

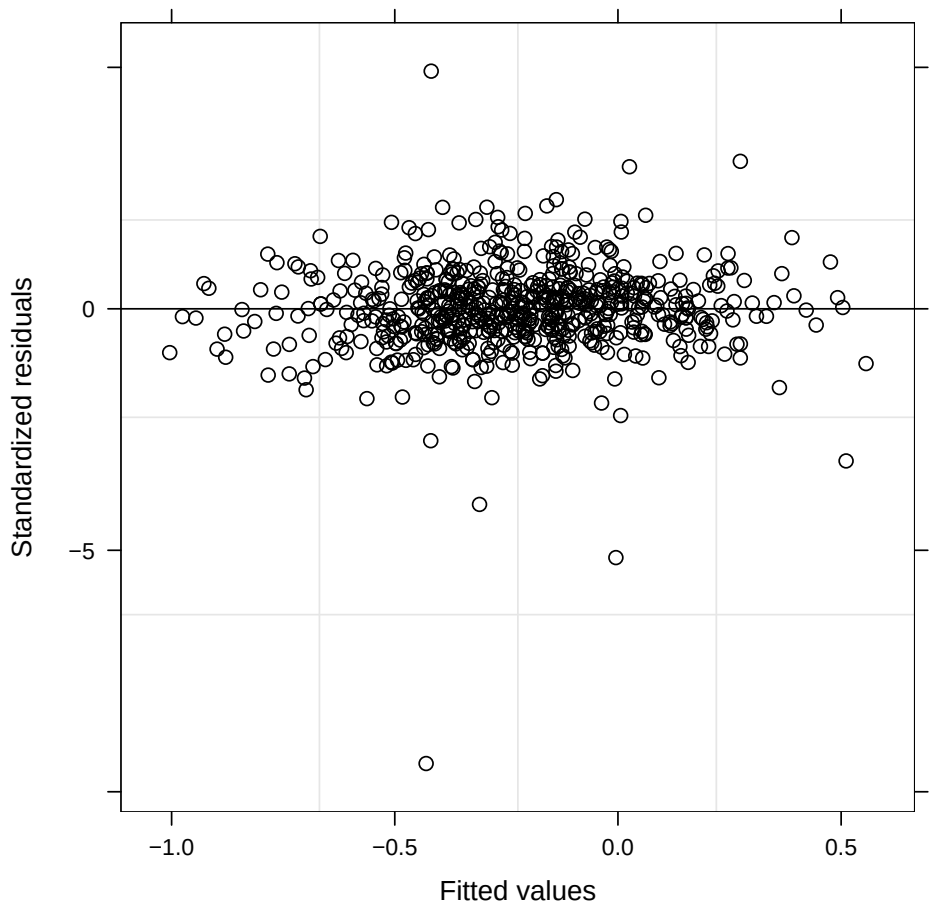
```
> idta$lncheact <- log(idta$cheact)
> idta$dose2 <- with(idta, ifelse(time > 0, dose, 0))
> idta$sexXtime <- with(idta, interaction(sex, time, drop = TRUE,
```

```
+     sep = ":"))
> idta$sexXtimeXdose <- with(idta, interaction(sex, time, dose2,
+     drop = TRUE, sep = ":"))
> lvls <- levels(idta$sexXtimeXdose)
> lvls[grep(":0$", lvls)] <- "Control"
> levels(idta$sexXtimeXdose) <- lvls
```

Now, fit the linear mixed effects model:

```
> drmod0 <- lme(lncheact ~ sexXtime + sexXtimeXdose - 1, data = idta,
+     random = ~1 | id)
```

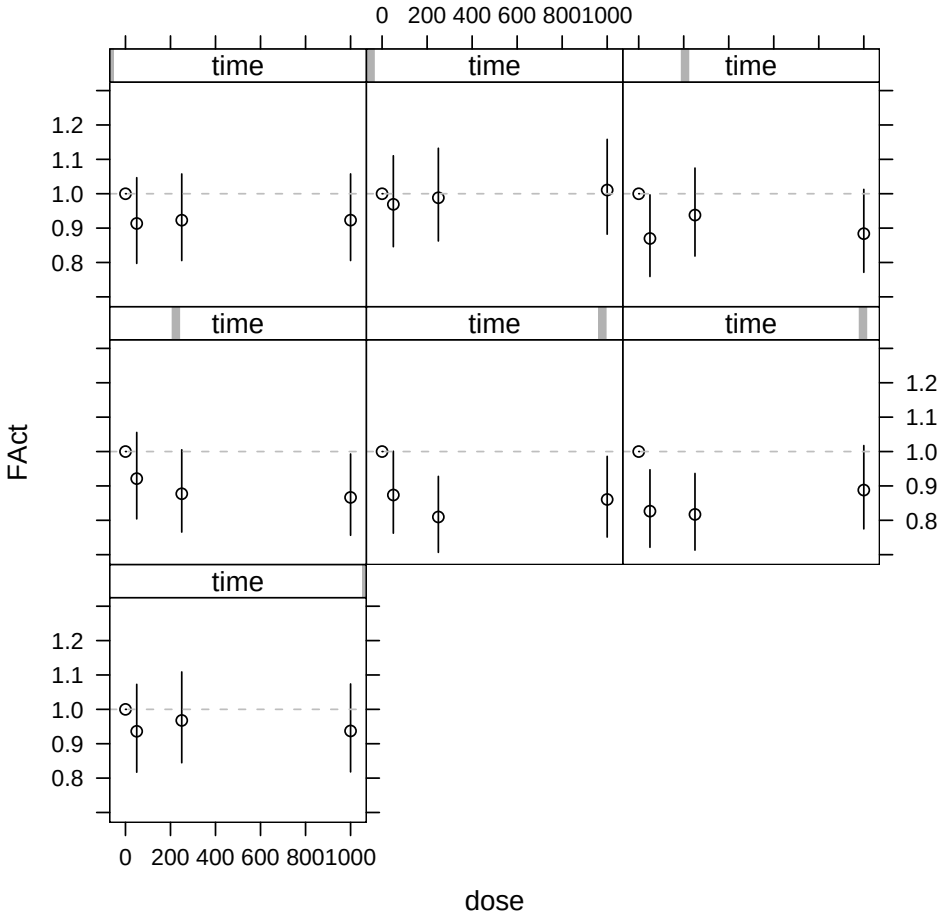
Here is a plot of residuals versus fitted value for that fit. Note that, on top of perhaps other problems for this dataset, there are some very extreme observations:



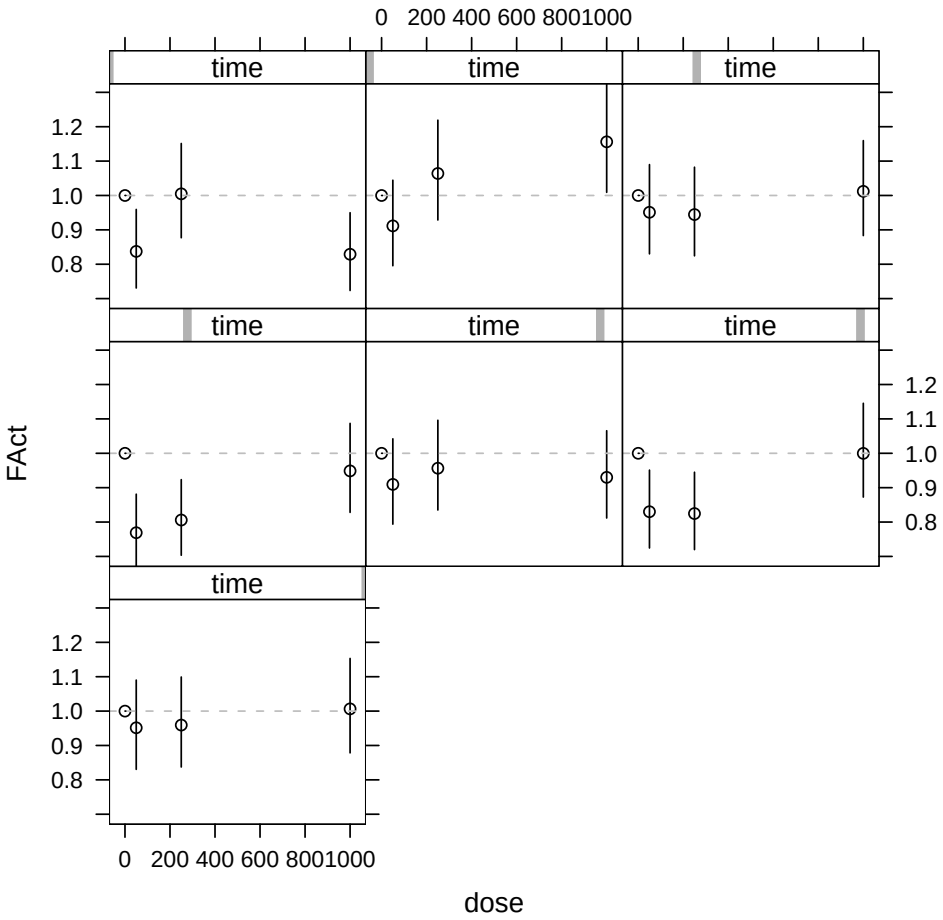
Now, set up a dataframe with fraction of inhibition at each dose and time point for both sexes, and plot (with confidence intervals):

```
> Ints <- intervals(drmod0, which = "fixed")$fixed
> nms <- rownames(Ints)
> indx <- grep("dose", nms)
> Ints <- Ints[indx, ]
> nms <- gsub("sexXtimeXdose", "", nms[indx])
> pdta <- data.frame(lower = exp(Ints[, "lower"]), upper = exp(Ints[,
```

```
+     "upper"]), FAct = exp(Ints[, "est."]), sex = factor(sapply(strsplit(nms,
+     ":"), function(x) x[1])), time = as.numeric(sapply(strsplit(nms,
+     ":"), function(x) x[2])), dose = as.numeric(sapply(strsplit(nms,
+     ":"), function(x) x[3])))
> ftms <- sort(unique(pdta$time[pdta$sex == "F"]))
> mtms <- sort(unique(pdta$time[pdta$sex == "M"]))
> nftms <- length(ftms)
> nmtms <- length(mtms)
> npdta <- data.frame(lower = rep(1, nftms + nmtms), upper = rep(1,
+     nftms + nmtms), FAct = rep(1, nftms + nmtms), sex = factor(rep(c("F",
+     "M"), c(nftms, nmtms))), dose = rep(0, nftms + nmtms), time = c(ftms,
+     mtms))
> pdta <- rbind(pdta, npdta)
```



Females: Males:



Based on the appearance of these graphs, these data do not give a coherent picture of any dose-response. No model will be fit to these data.

3 Summary

There was no dose-response, so no BMD, etc. was calculated.

species RABBIT

mrid [1] "41066001"

Save everything:

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

Save the results for incorporating into a database:

```
> propoxur.dermal.rbc <- list(mrid = levels(dta$mrid), species = levels(dta$species))
> save(propoxur.dermal.rbc, file = file.path("..", "..", "01Summaries",
+     "propoxur.dermal.rbc.RData"))
```