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

June 24, 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 `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 rabbit RBC. via dermal dosing.

Set up a dataset for plotting and a dataset for analysis.

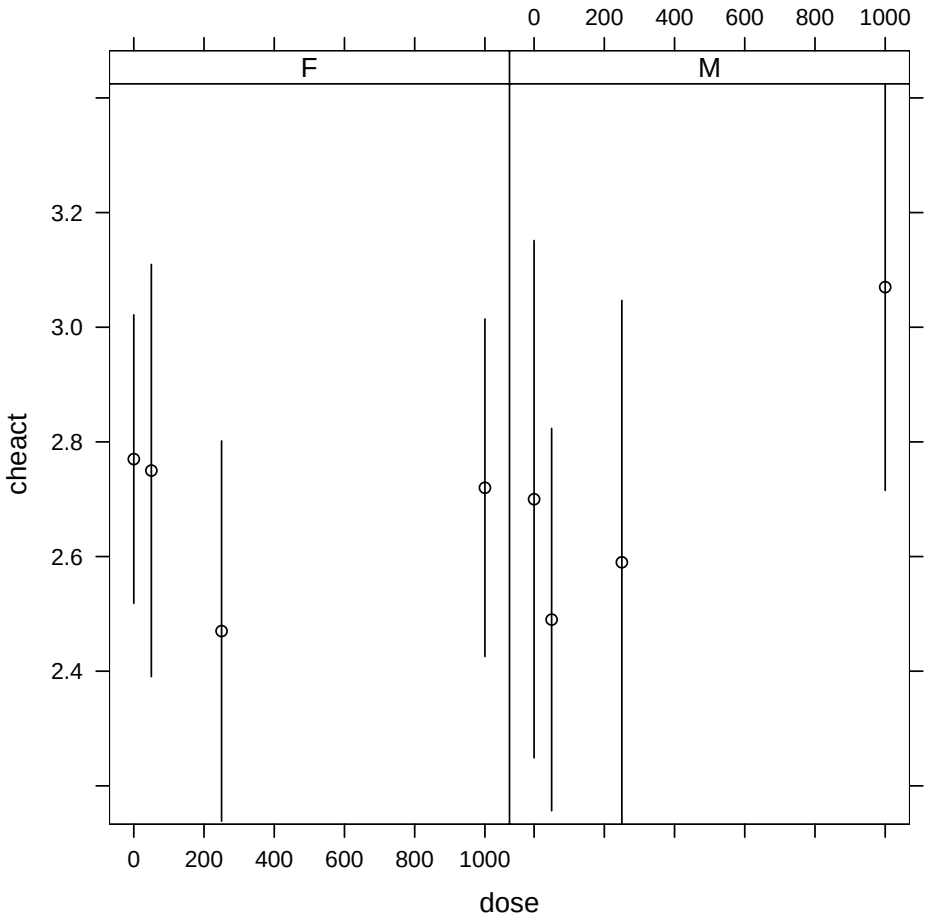
```
> dta <- CleanUp(subset(AggData, chemical %in% "PROPOXUR" & dsmtd %in%
+     "DERMAL" & sctn %in% "WHOLEBRAIN" & !is.na(cheact), select = c("dose",
+     "cheact", "n", "sd", "tmonstdy", "sex", "tmpstds", "species",
+     "mrid", "sctn")))
> idta <- with(dta, PhonyDF(dose, n, cheact, sd, "dose", "cheact",
+     dta[, c("sex"), drop = FALSE]))
```

Summary of this dataset:

```
> summary(dta)

      dose            cheact            n            sd            tmonstdy
 Min.   :   0.0   Min.   :2.470   Min.   :10   Min.   :0.4060   Min.   :92.0
 1st Qu.:  37.5   1st Qu.:2.565   1st Qu.:10   1st Qu.:0.5200   1st Qu.:92.0
 Median : 150.0   Median :2.710   Median :10   Median :0.5550   Median :92.5
 Mean   : 325.0   Mean   :2.695   Mean   :10   Mean   :0.5714   Mean   :92.5
 3rd Qu.: 437.5   3rd Qu.:2.755   3rd Qu.:10   3rd Qu.:0.6170   3rd Qu.:93.0
 Max.   :1000.0   Max.   :3.070   Max.   :10   Max.   :0.7370   Max.   :93.0

 sex      tmpstds      species        mrid           sctn
 F:4   Min.   : NA   RABBIT:8   41066001:8   WHOLEBRAIN:8
```

```
 M:4   1st Qu.: NA
       Median : NA
       Mean   :NaN
       3rd Qu.: NA
       Max.   : NA
       NA's   :  8

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

, , sex = F

      tmonstdy
dose   92 93
  0    0  1
  50   0  1
  250  0  1
  1000 0  1

, , sex = M

      tmonstdy
dose   92 93
  0    1  0
  50   1  0
  250  1  0
  1000 1  0
```

Some dose-response plots for females and males:



2 Dose-Response Modeling

Given the above figures, it looks as if there may be no dose-response at all. A quick analysis confirms this. Fit a linear model predicting log(cheact) as linear function of dose, and the interaction of sex and dose (which will be the difference of the slopes between the two sexes). Test the difference in slope between the two sexes and then the significance of the slope:

```
> idta$lncheact <- log(idta$cheact)
> drmod1 <- lm(lncheact ~ sex - 1 + dose + sex:dose, data = idta)
> summary(drmod1)

Call:
lm(formula = lncheact ~ sex - 1 + dose + sex:dose, data = idta)

Residuals:
      Min        1Q    Median        3Q       Max
-0.515431 -0.137970  0.002134  0.144158  0.493533

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
sexF       9.681e-01  4.414e-02  21.930   <2e-16 ***
sexM       9.072e-01  4.414e-02  20.552   <2e-16 ***
```

```
dose      -1.150e-06  8.555e-05  -0.013    0.989
sexM:dose  1.890e-04  1.210e-04   1.562    0.122
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2169 on 76 degrees of freedom
Multiple R-Squared: 0.9546,        Adjusted R-squared: 0.9522
F-statistic: 399.7 on 4 and 76 DF,  p-value: < 2.2e-16
```

So the slopes do not differ between sexes. Refit a model with a pooled slope estimate:

```
> drmod2 <- lm(lncheact ~ sex - 1 + dose, data = idta)
> summary(drmod2)

Call:
lm(formula = lncheact ~ sex - 1 + dose, data = idta)

Residuals:
      Min        1Q    Median        3Q       Max
-0.522517 -0.158829  0.005735  0.150811  0.462826

Coefficients:
      Estimate Std. Error t value Pr(>|t|)
sexF 9.374e-01  3.989e-02  23.498   <2e-16 ***
sexM 9.379e-01  3.989e-02  23.512   <2e-16 ***
dose 9.333e-05  6.106e-05   1.529    0.130
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2189 on 77 degrees of freedom
Multiple R-Squared: 0.9532,        Adjusted R-squared: 0.9513
F-statistic: 522.4 on 3 and 77 DF,  p-value: < 2.2e-16
```

The slope is *positive* and non-significant. No further modeling for these data.

3 Summary

There is no evidence of a dose-response, so no model was attempted.

species RABBIT

mrid [1] "41066001"

Save everything:

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

Save the results for incorporating into a database:

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