Dose Response Modeling of Rat RBC AChE Activity: Oxamyl Inhalation Exposure

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
```

All the rat gavage 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`.

Now, set up the analysis dataset.

```
> dta <- CleanUp(subset(AggData, chemical %in% "OXAMYL" & dsmtd %in%
+     "INHALATION" & sctn %in% "RBC", select = c("cheact", "sd",
+     "n", "dose", "tmonstdy", "sctn", "species", "tmpstds", "sex",
+     "mrid")))
```

Summary of the relevant variables in this dataset:

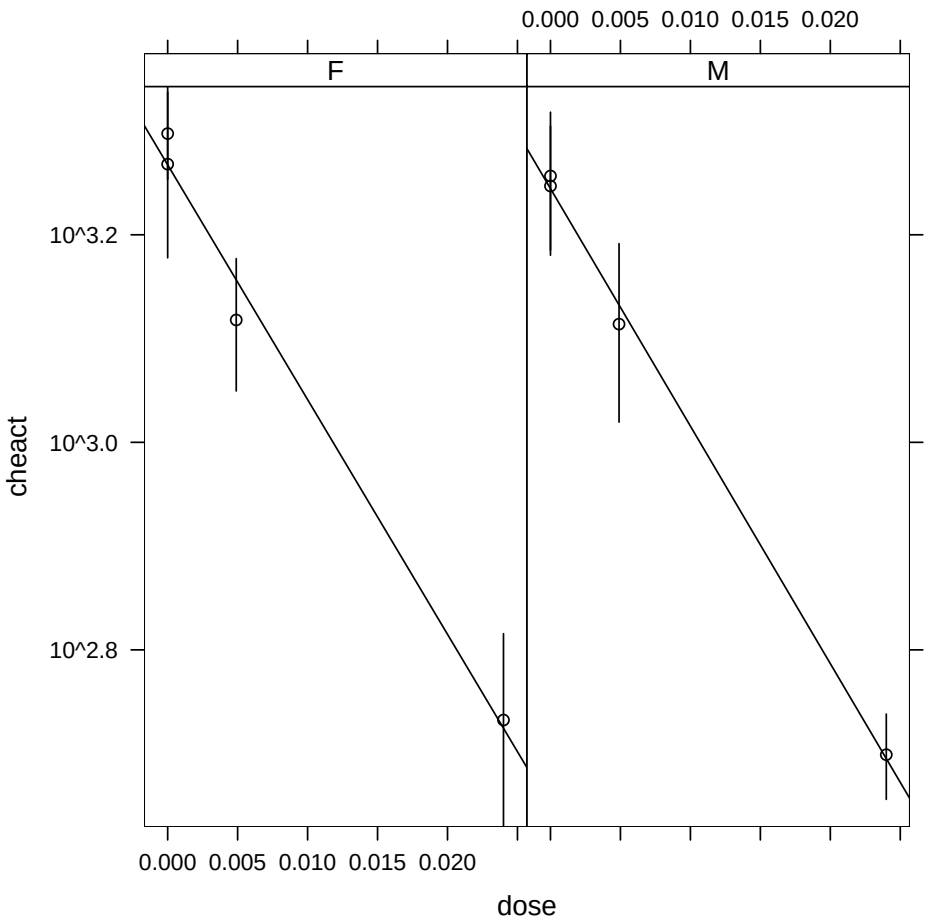
```
> summary(dta)

     cheact           sd              n               dose             tmonstdy
 Min.   : 500   Min.   : 66.0   Min.   : 9.000   Min.   :0.000000   Min.   :1
 1st Qu.:1110   1st Qu.:237.7   1st Qu.:10.000   1st Qu.:0.000000   1st Qu.:1
 Median :1539   Median :309.8   Median :10.000   Median :0.002450   Median :1
 Mean   :1383   Mean   :290.9   Mean   : 9.875   Mean   :0.007225   Mean   :1
 3rd Qu.:1818   3rd Qu.:362.4   3rd Qu.:10.000   3rd Qu.:0.009675   3rd Qu.:1
 Max.   :1984   Max.   :486.5   Max.   :10.000   Max.   :0.024000   Max.   :1
  sctn   species    tmpstds    sex         mrid
 RBC:8   RAT:8   Min.   :0.1   F:4   45155801:8
                 1st Qu.:0.1   M:4
```

```
                 Median :0.1
                 Mean   :0.1
                 3rd Qu.:0.1
                 Max.   :0.1
```

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

```
        tmpstds
dose     0.1
  0      4
  0.0049 2
  0.024  2
```

Plots of RBC AChE activity versus dose, by sex. There is only one time point in each of these two studies. The fitted line is the best fitting line to the log of the mean values within each panel.



2 Dose-Response Modeling

The fitted lines in the above figures seem to describe the data pretty well, and, given the log-scale of the y-axis, indicate that the simplest version of the exponential model describes these data adequately. That the confidence intervals do not seem to vary much in length across the doses suggests we can model log(cheact) as a linear function of dose.

Fitting the model will follow these steps:

1. Create a pseudo-individual dataset, and log the `cheact` variable.
2. Use lm to fit a linear model, where the intercept and slope differs across `sex`.
3. Test whether slopes differ between sexes; if not, refit without the sex difference.
4. Construct logBMDs, BMDs, standard errors, and confidence intervals using these estimates and the delta method.

The individual data-set:

```
> idta <- with(dta, PhonyDF(dose, n, cheact, sd, "dose", "cheact",
+     Avals = dta[, c("mrid", "sex")]))
> idta$lncheact <- log(idta$cheact)
```

Fit the model:

```
> 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
-0.511844 -0.136770  0.005484  0.144330  0.455072

Coefficients:
           Estimate Std. Error t value Pr(>|t|)
sexF        7.50518    0.04267 175.872   <2e-16 ***
sexM        7.44679    0.04265 174.594   <2e-16 ***
dose      -52.77895    3.61854 -14.586   <2e-16 ***
sexM:dose   0.64490    5.02211   0.128    0.898
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2178 on 75 degrees of freedom
Multiple R-Squared: 0.9991,        Adjusted R-squared: 0.9991
F-statistic: 2.114e+04 on 4 and 75 DF,  p-value: < 2.2e-16

> pv <- summary(drmod1)$coefficients["sexM:dose", 4]
```

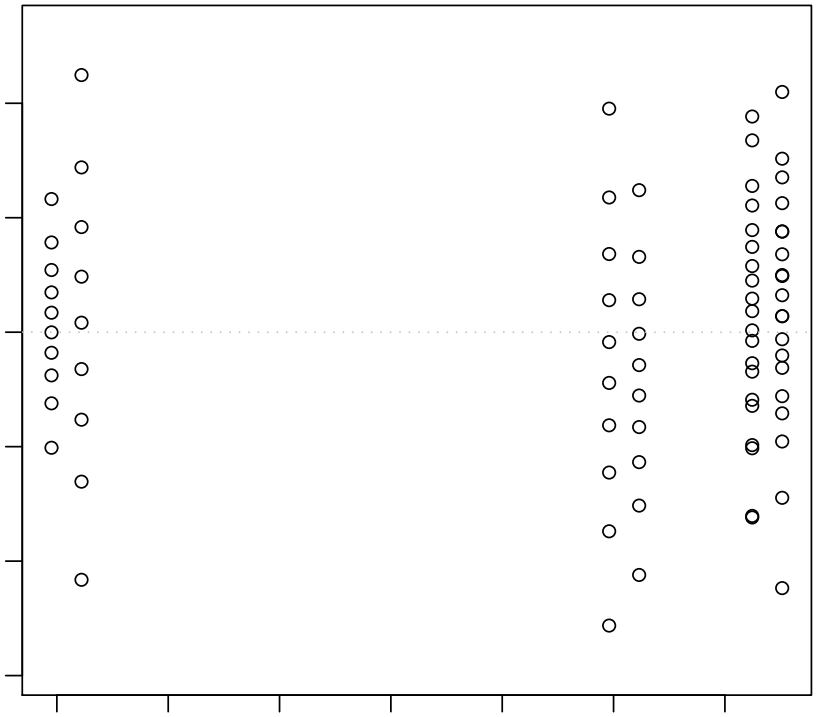
Note the sex effect on dose is non-significant: P-value = 0.9, so refit with a single slope for dose:

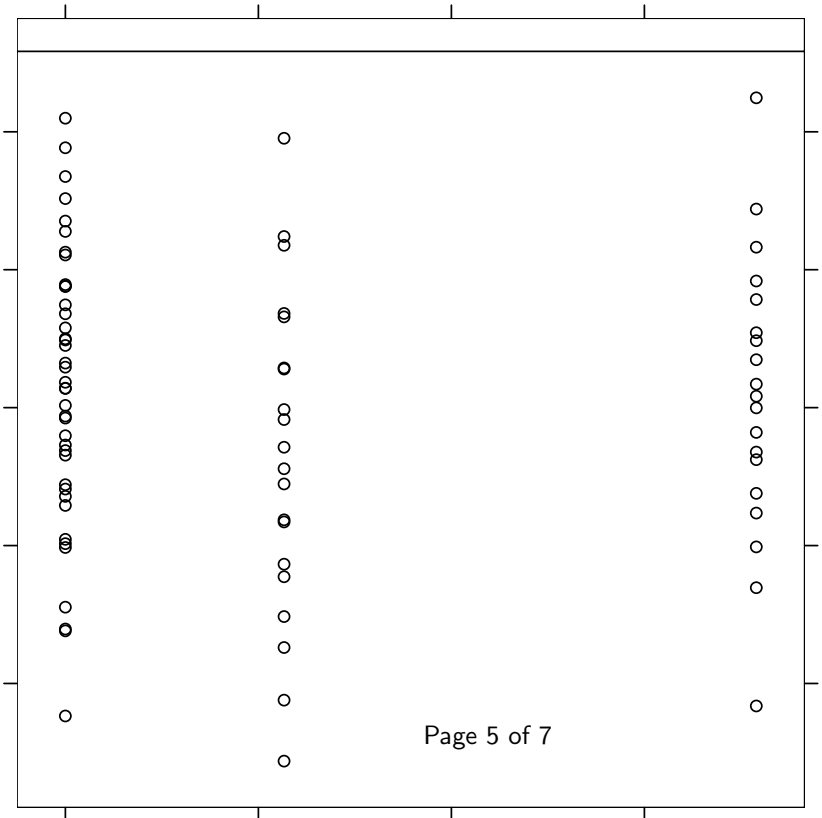
```
> drmod2 <- lm(lncheact ~ sex - 1 + dose, data = idta)
```

We will use the results in `drmod2` to estimate BMDs.

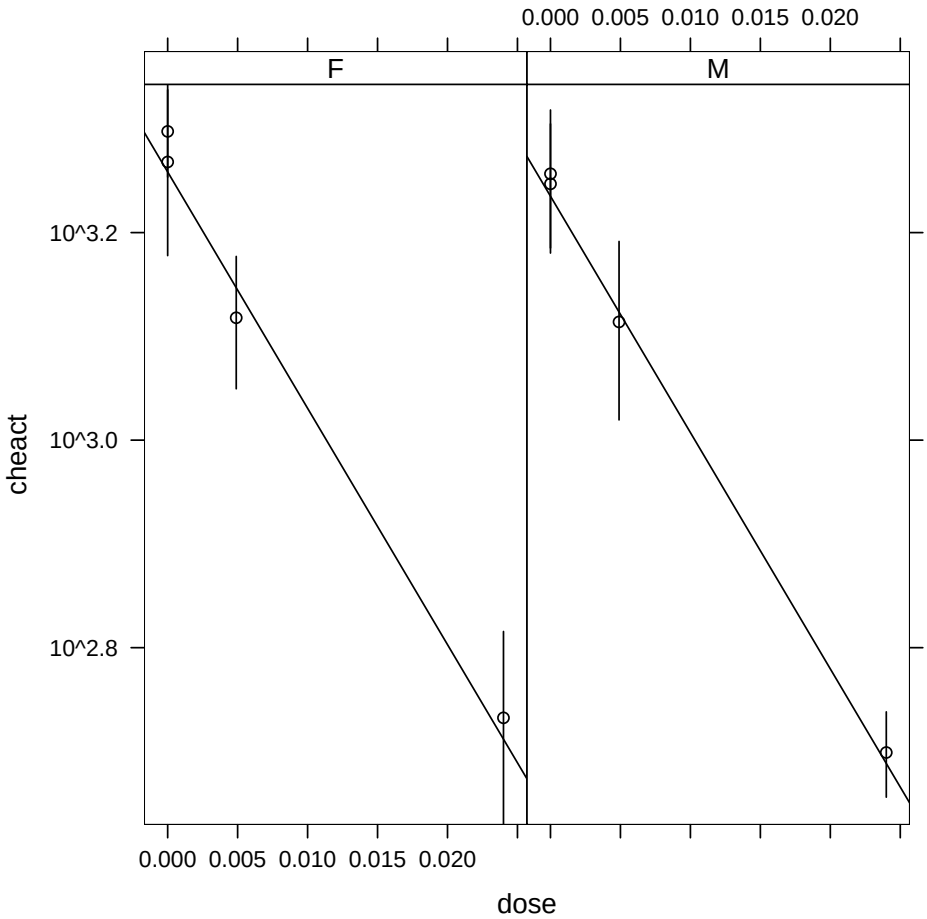
3 Model Summary

Diagnostic plots for this model:





shows the mean AChE values and their 95% confidence intervals, and the fitted line from `drmod2`.



4 Calculating BMDs

We have essentially fit the model

$$Y = Ae^{m \times dose}$$

with a constant coefficient of variation model for the error variance. In this model, for a BMD that results in BMR$\times$100% inhibition, the BMD is calculated as:

$$\frac{log(1 - BMR)}{m}$$

For consistency with the other endpoints and models, and to be able to calculate RPFs with confidence intervals, we need log BMD and its standard error. log BMD is just the log of the above quantity, and its standard error is, based on the delta method:

$$\hat{\sigma}_{lD} = \frac{\hat{\sigma}_m}{m}$$

The following code calculates the BMDs, etc.

```
> tTab <- summary(drmod2)$coefficients
> Ints <- confint(drmod2)
> Ints90 <- confint(drmod2, level = 0.9)
> BMD <- log(1 - 0.1)/tTab["dose", "Estimate"]
```

```
> BMDL <- log(1 - 0.1)/Ints90["dose", "5 %"]
> BMD.CI <- log(1 - 0.1)/Ints["dose", c("2.5 %", "97.5 %")]
> if (BMD.CI[2] < 0) BMD.CI[2] <- Inf
> lD <- log(BMD)
> lD.se <- tTab["dose", "Std. Error"]/tTab["dose", "Estimate"]
```

5 Summary

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. BMD has units mg/L.

species RAT

mrid `[1] "45155801"`

`lD` **(standard error)** -6.21011615993328 (-0.0475348304215046)

BMD (95% CI) 0.00200900408473521,(0.00183525361480332, 0.00221909420999035)

BMDL, the one-sided lower 95% CL 0.00186164952239408

Save everything:

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

Save the results for incorporating into a database:

```
> oxamyl.inhalation.rbc <- list(mrid = levels(dta$mrid), species = levels(dta$species),
+     combined = list(lD = lD, lD.se = lD.se, BMD = BMD, BMD.CI = BMD.CI,
+         BMDL = BMDL))
> save(oxamyl.inhalation.rbc, file = file.path("..", "..", "01Summaries",
+     "oxamyl.inhalation.rbc.RData"))
```