Chapter 8 Exercises

Data Analysis & Graphics Using R – Solutions to Exercises (May 1, 2010)

Preliminaries

\[ \texttt{library(DAAG)} \]

Exercise 1

The following table shows numbers of occasions when inhibition (i.e., no flow of current across a membrane) occurred within 120 s, for different concentrations of the protein peptide-C (data are used with the permission of Claudia Haarmann, who obtained these data in the course of her PhD research). The outcome yes implies that inhibition has occurred.

<table>
<thead>
<tr>
<th>conc</th>
<th>0.1</th>
<th>0.5</th>
<th>1</th>
<th>10</th>
<th>20</th>
<th>30</th>
<th>50</th>
<th>70</th>
<th>80</th>
<th>100</th>
<th>150</th>
</tr>
</thead>
<tbody>
<tr>
<td>no</td>
<td>7</td>
<td>1</td>
<td>10</td>
<td>9</td>
<td>2</td>
<td>9</td>
<td>13</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>yes</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>4</td>
<td>0</td>
<td>6</td>
<td>7</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>7</td>
</tr>
</tbody>
</table>

Use logistic regression to model the probability of inhibition as a function of protein concentration.

It is useful to begin by plotting the logit of the observed proportions against \( \log(\text{conc}) \). Concentrations are nearer to equally spaced on a scale of relative dose, rather than on a scale of dose, suggesting that it might be appropriate to work with \( \log(\text{conc}) \). In order to allow plotting of cases where no = 0 or yes = 0, we add 0.5 to each count.

\[ \text{Figure 1: Plot of } \log((\text{yes+0.5})/(\text{no+0.5})) \text{, against } \log(\text{conc}). \]

\[ \texttt{> conc <- c(.1, .5, 1, 10, 20, 30, + 50, 70, 80, 100, 150)} \]
\[ \texttt{> no <- c(7, 1, 10, 9, 2, 9, 13, 1, 1, 4, 3)} \]
\[ \texttt{> yes <- c(0, 0, 3, 4, 0, 6, 7, 0, 0, 1, 7)} \]
\[ \texttt{> n <- no + yes} \]
\[ \texttt{> plot(log(conc), log((yes+0.5)/(no+0.5)))} \]

The plot seems consistent with the use of \( \log(\text{conc}) \) as the explanatory variable.

The code for the regression is:

\[ \texttt{> p <- yes/n} \]
\[ \texttt{> inhibit.glm <- glm(p ~ I(log(conc)), family=binomial, weights=n)} \]
\[ \texttt{> summary(inhibit.glm)} \]

Call:
\[ \text{glm(formula = p ~ I(log(conc)), family = binomial, weights = n)} \]

Deviance Residuals:

\[
\begin{array}{cccc}
\text{Min} & \text{1Q} & \text{Median} & \text{3Q} & \text{Max} \\
-1.251 & -1.060 & -0.503 & 0.315 & 1.351 \\
\end{array}
\]
Exercise 2
In the data set (an artificial one of 3121 patients, that is similar to a subset of the data analyzed in Stiell et al. (2001)) minor.head.injury, obtain a logistic regression model relating clinically.important.brain.injury to other variables. Patients whose risk is sufficiently high will be sent for CT (computed tomography). Using a risk threshold of 0.025 (2.5%), turn the result into a decision rule for use of CT.

```r
> sapply(head.injury, range)

age.65 amnesia.before basal.skull.fracture GCS.decrease GCS.13
[1,] 0 0 0 0 0
[2,] 1 1 1 1 1

GCS.15.2hours high.risk loss.of.consciousness
[1,] 0 0 0
[2,] 1 1 1

open.skull.fracture vomiting clinically.important.brain.injury
[1,] 0 0 0
[2,] 1 1 1

> injury.glm <- glm(clinically.important.brain.injury ~ .,
+ data=head.injury, family=binomial)
> summary(injury.glm)

Call:
glm(formula = clinically.important.brain.injury ~ ., data = head.injury, family = binomial)

Deviance Residuals:
     Min       1Q   Median       3Q      Max
-2.277   -0.351   -0.210   -0.149    3.003

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)       -4.49700   0.16358  -27.61 < 2e-16
age.65            1.37315   0.18252   7.520 5.6e-14
amnesia.before    0.68921   0.17161   4.001 6.4e-05
basal.skull.fracture  1.96241   0.20623   9.501  < 2e-16
GCS.decrease      -0.26939   0.36817  -0.731    0.46515

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 16.683 on 10 degrees of freedom
Residual deviance: 9.395 on 9 degrees of freedom
AIC: 29.99

Number of Fisher Scoring iterations: 4
GCS.13 1.061 0.282 3.76 0.00017
GCS.15.2hours 1.941 0.166 11.67 < 2e-16
high.risk 1.111 0.159 6.98 2.9e-12
loss.of.consciousness 0.955 0.196 4.88 1.1e-06
open.skull.fracture 0.630 0.315 2.00 0.04542
vomiting 1.233 0.196 6.29 3.2e-10

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 1741.6 on 3120 degrees of freedom
Residual deviance: 1201.3 on 3110 degrees of freedom
AIC: 1223

Number of Fisher Scoring iterations: 6

Observe that \(\log(0.025/(1-0.025)) = -3.66\), an increase of 0.84 above the intercept (= -4.50). This change in risk results from (1) GCS.decrease with any other individual factor except amnesia.before, GCS.decrease and open.skull.fracture; (2) GCS.decrease with any two of amnesia.before, open.skull.fracture and loss.of.consciousness; (3) any of the individual factors age.65, basal.skull.fracture, GCS.15.2hours, high.risk and vomiting, irrespective of the levels of other factors.

**Exercise 3**
Consider again the moths data set of Section 8.4.

(a) What happens to the standard error estimates when the poisson family is used in \texttt{glm()} instead of the quasipoisson family?

(b) Analyze the P moths, in the same way as the A moths were analyzed. Comment on the effect of transect length.

(a) The dispersion estimate was 2.69. Use of the quasipoisson family has the effect of increasing SEs by a factor of \(\sqrt{2.69}\), relative to the poisson family. See the first two lines on p.215. SEs on pp.214-215 will thus be reduced by this factor if the poisson family is (inappropriately) specified.

(b) \texttt{> sapply(split(moths$P, moths$habitat), sum)}

<table>
<thead>
<tr>
<th>Bank</th>
<th>Disturbed</th>
<th>Lowerside</th>
<th>NEsoak</th>
<th>NWsoak</th>
<th>SEsoak</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>33</td>
<td>17</td>
<td>14</td>
<td>19</td>
<td>6</td>
</tr>
</tbody>
</table>

\texttt{SWsoak Upperside}

48 8

\texttt{> moths$habitat <- relevel(moths$habitat, ref="Lowerside")}
\texttt{> P.glm <- glm(P ~ habitat + log(meters), family=quasipoisson, +
  data=moths)}

The highest numbers are now for SWsoak and for Disturbed The number of moths increases with transect length, by a factor of approximately 1.74 (= e.55) for each one meter increase in transect length.
Exercise 4*

The factor dead in the data set mifem (DAAG package) gives the mortality outcomes (live or dead), for 1295 female subjects who suffered a myocardial infarction. (See Section 11.5 for further details.) Determine ranges for age and yronset (year of onset), and determine tables of counts for each separate factor. Decide how to handle cases for which the outcome, for one or more factors, is not known. Fit a logistic regression model, beginning by comparing the model that includes all two-factor interactions with the model that has main effects only.

First, examine various summary information:

```r
> str(mifem)
'data.frame': 1295 obs. of 10 variables:
  $ outcome : Factor w/ 2 levels "live","dead": 1 1 1 1 2 1 1 2 2 2 ...  
  $ age : num 63 55 68 64 67 66 63 68 46 66 ...  
  $ yronset: num 85 85 85 85 85 85 85 85 85 85 ...  
  $ premi : Factor w/ 3 levels "y","n","nk": 2 2 1 2 2 2 1 2 1 ...  
  $ smstat : Factor w/ 4 levels "c","x","n","nk": 2 1 4 2 4 2 3 3 1 1 ...  
  $ diabetes: Factor w/ 3 levels "y","n","nk": 2 2 3 2 3 3 2 2 2 2 ...  
  $ highbp : Factor w/ 3 levels "y","n","nk": 1 1 1 1 3 1 1 1 1 1 ...  
  $ hichol : Factor w/ 3 levels "y","n","nk": 1 1 3 2 3 3 2 1 3 2 ...  
  $ angina : Factor w/ 3 levels "y","n","nk": 2 2 1 1 3 3 2 1 3 2 ...  
  $ stroke : Factor w/ 3 levels "y","n","nk": 2 2 2 2 3 3 2 1 2 1 ...  
> sapply(mifem[, c("age", "yronset")], range)
 age yronset
[1,] 35 85
[2,] 69 93
> lapply(mifem[, -(1:3)], table)

$premi
   y  n  nk
311 928  56

$smstat
   c  x  n  nk
390 280 522 103

$diabetes
   y  n  nk
248 978  69

$highbp
   y  n  nk
813 406  76
```
For all of the factors, there are a large number of nk’s, i.e., not known. A straightforward way to handle them is to treat nk as a factor level that, as for y and n, may give information that helps predict the outcome. For ease of interpretation we will make n, the reference level.

```
> for(j in 4:10)mifem[,j] <- relevel(mifem[,j], ref="n")
> mifem1.glm <- glm(outcome ~ ., family=binomial, data=mifem)
> mifem2.glm <- glm(outcome ~ .^2, family=binomial, data=mifem)
> anova(mifem1.glm, mifem2.glm)
```

Analysis of Deviance Table

Model 1: outcome ~ age + yronset + premi + smstat + diabetes + highbp + hichol + angina + stroke
Model 2: outcome ~ (age + yronset + premi + smstat + diabetes + highbp + hichol + angina + stroke)^2

<table>
<thead>
<tr>
<th>Resid. Df</th>
<th>Resid. Dev</th>
<th>Df</th>
<th>Deviance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1277</td>
<td></td>
<td>1173</td>
</tr>
<tr>
<td>2</td>
<td>1152</td>
<td>125</td>
<td>159</td>
</tr>
</tbody>
</table>

```
> CVbinary(mifem1.glm)
```

```
Fold:  6 10 9 5 2 7 1 4 8 3
Internal estimate of accuracy = 0.807
Cross-validation estimate of accuracy = 0.803
```

```
> CVbinary(mifem2.glm)
```

```
Fold:  2 1 8 7 9 10 6 5 4 3
Internal estimate of accuracy = 0.839
Cross-validation estimate of accuracy = 0.775
```

The difference in deviance seems statistically significant ($\text{pchisq}(125,159) = 0.021$), but it may be unwise to trust the chi-squared approximation to the change in deviance.

It is safer to compare the cross-validated accuracy estimates, which in individual cross-validation runs were marginally lower for mifem2.glm than for mifem2.glm; 0.78 as against 0.80. Note also that there were convergence problems for the model that included all first order interaction terms.