Data Analysis & Graphics Using R – Solutions to Exercises (April 24, 2004)

Chapter 8 Exercises

Preliminaries

> 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(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(conc). In order to allow plotting of cases where no = 0 or yes = 0, we add 0.5 to each count.

> conc <- c(0.1, 0.5, 1, 10, 20, 30, 50, 70, 80, 100, 150)
> no <- c(7, 1, 10, 9, 2, 9, 13, 1, 1, 4, 3)
> yes <- c(0, 0, 3, 4, 0, 6, 7, 0, 0, 1, 7)
> n <- no + yes
> plot(log(conc), log((yes + 0.5)/(no + 0.5)))

The plot seems reasonably consistent with the use of log(conc) as the explanatory variable.

The code for the regression is:

> p <- yes/n
> inhibit.glm <- glm(p ~ I(log(conc)), family = binomial, weights = n)
> summary(inhibit.glm)
Exercise 2
This question has been reworded
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)) 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
> data(head.injury)
> sapply(head.injury, range)

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

 GCS.15.2hours high.risk.loss.of.consciousness open.skull.fracture vomiting
[1,] 0 0 0 0 0 0 0 0
[2,] 1 1 1 1 1

 clinically.important.brain.injury
[1,] 0
[2,] 1

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

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

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

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 16.6834 on 10 degrees of freedom
Residual deviance:  9.3947 on  9 degrees of freedom
AIC: 29.99

Number of Fisher Scoring iterations: 4
```
Coefficients:

|                     | Estimate | Std. Error | z value | Pr(>|z|) |
|---------------------|----------|------------|---------|----------|
| (Intercept)         | -4.497   | 0.163      | -27.61  | < 2e-16  |
| age.65              | 1.373    | 0.183      | 7.52    | 5.6e-14  |
| amnesia.before      | 0.689    | 0.172      | 4.00    | 6.4e-05  |
| basal.skull.fracture| 1.962    | 0.206      | 9.50    | < 2e-16  |
| GCS.decrease        | -0.269   | 0.368      | -0.73   | 0.46515  |
| 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 \( \log(.025/(1-.025)) = -3.66 \), an increase of 0.84 above the intercept (= -4.50). This change in risk results from (1) \textit{GCS.decrease} with any other individual factor except \textit{amnesia.before}, \textit{GCS.decrease} and \textit{open.skull.fracture}; (2) \textit{GCS.decrease} with any two of \textit{amnesia.before}, \textit{open.skull.fracture} and \textit{loss.of.consciousness}; (3) any of the individual factors \textit{age.65}, \textit{basal.skull.fracture}, \textit{GCS.15.2hours}, \textit{high.risk} and \textit{vomiting}, irrespective of the levels of other factors.

**Exercise 3**

Consider again the \textit{moths} data set of Section 8.4.

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

(b) To visualize the data, type in the following:

```r
library(lattice)
data(moths)
avmoths <- aggregate(moths[,2], by=list(moths$habitat), FUN=mean)
names(avmoths) <- c("habitat", "av")
dotplot(habitat ~ av, data=avmoths, pch=16, cex=1.5)
```

Reconcile this with the Poisson regression output that used \textit{Lowerside} as the reference.

(c) Analyze the \textit{P} moths, in the same way as the \textit{A} moths were analyzed. Comment on the effect of transect length.

(a) The dispersion estimate was 2.69. Use of the \textit{quasipoisson} family has the effect of increasing SEs by a factor of \( \sqrt{2.69} \), relative to the \textit{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) Here is the graph:

```r
> library(lattice)
> data(moths)
> avmoths <- aggregate(moths[, 2], by = list(moths$habitat), FUN = mean)
> names(avmoths) <- c("habitat", "av")
> print(dotplot(habitat ~ av, data = avmoths, pch = 16, cex = 1.5))
```

![Graph showing moth population by habitat]

Figure 2: Total number of species A moths, by habitat

Easily the greatest number of moths are in the NWsoak habitat, in agreement with the estimate of 1.56 for NWsoak on p.214. (The next largest coefficient is 0.23).

(c) > sapply(split(moths$P, moths$habitat), sum)

```
Bank  Disturbed  Lowerside  NEsoak  NWsoak  SEsoak  SWsoak  Upperside
 4     33        17       14       19       6       48        8
```

> moths$habitat <- relevel(moths$habitat, ref = "Lowerside")
> 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 (= exp(0.55)) 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 Subsection 10.7.1 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
> data(mifem)
> 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 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

$hichol
   y   n  nk
 452 655 188

$angina
   y   n  nk
 472 724  99

$stroke
   y   n  nk
 153 1063  79
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.

```r
> 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**

<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>

```r
> cv.binary(mifem1.glm)
```

Fold: 9 7 4 6 8 3 2 1 10 5
Internal estimate of accuracy = 0.807
Cross-validation estimate of accuracy = 0.8

```r
> cv.binary(mifem2.glm)
```

Fold: 1 5 6 7 3 10 9 2 4 8
Internal estimate of accuracy = 0.839
Cross-validation estimate of accuracy = 0.769

The difference in deviance seems statistically significant (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.