

Stats 4590: Lab #8
Generalized Linear Models in R
March 29, 2010

In today's lab we'll work through two datasets where we'll use logistic and log-linear regression to analyze the data. This lab will leave you to fill in a lot of the blanks, as everyone has progressed superbly through the course, and you don't need many pointers from me at this point. However, we will discuss a couple of R commands in some detail.

Your first steps should be, as always, to do the following.

1. Move to your **ST4590** directory using **cd ST4590**.
2. Create a directory for this lab using **mkdir lab8**.
3. Move to this directory using **cd lab8**.

Finally, start a script file for today's session, if that's how you want to save your output:

```
$ script lab8  
Script started, file is lab8
```

Now everything will get saved in the file **lab8**. Start R as usual.

Example 1: Low Birth Weights

This is a dataset we examined in our previous lab, where we used a linear regression model to estimate systolic blood pressure. This week we'll use a different variable as the response. The variable is labelled **grmhem**, which equals 1 if an infant experienced a germinal matrix hemorrhage, and 0 if not.

To load the data, type:

```
> lowbwt.dat <- read.table("/users/math/faculty/sneddon/DATA/lowbwt.dat", header = T)
```

Is a linear regression model appropriate here? Explain.

Construct plots of the data against the following explanatory variables: (1) **apgar5** (2) **toxemia** (3) **gestage**. What do the plots suggest?

Now, fit a model that estimates $\text{logit}(\pi_i)$ from **apgar5**, where $\pi_i = P(\text{hemorrhage})$:

```
grm.apgar5 <- glm(grmhem ~ apgar5, family = binomial, data = lowbwt.dat)
```

Use the `summary` command to examine the results. Write the model for estimating $\text{logit}(\pi_i)$, and the model for estimating π_i below:

Is `apgar5` helpful in predicting the probability of experiencing a germinal matrix hemorrhage?

Next, fit a model that estimates $\text{logit}(\pi_i)$ from `apgar5`, `toxemia` and `gestage`, including an interaction term for `toxemia` and `gestage`:

```
> grm.aptoxges <- glm(grmhem ~ apgar5 + toxemia * gestage, family = binomial,
  data = lowbwt.dat)
```

Use the `summary` command and determine if this model is useful.

Now, find the drop-in-deviance statistic to determine if all the terms involving `toxemia` and `gestage` are needed in the model:

H_0 :

H_a :

Test statistic =

p-value =

What is your conclusion?

We can also use the `anova` command to get the drop in deviance values, just like we used it to get $(SSE_R - SSE_F)$ in linear regression:

```

> anova(grm.aptoxges)
Analysis of Deviance Table

Model: binomial, link: logit

Response: grmhem

Terms added sequentially (first to last)

              Df Deviance Resid. Df Resid. Dev
NULL                               99      84.542
apgar5             1      5.615           98      78.927 = Dev. with apgar5 in model
toxemia            1      1.960           97      76.967
gestage            1      0.307           96      76.660
toxemia:gestage   1      0.138           95      76.522 = Dev. with apgar5, tox,
                                                gest and tox*gest in model

```

Example 2: Epileptic Seizures

The following dataset was collected during a randomized study of the anti-epileptic drug progabide. The variables in the dataset are:

1. `treat`, where 0 = control and 1 = treated with drug.
2. `age`, age of patient (yrs).
3. `basecount`, the number of seizures in the 8 weeks prior to administration of treatment.
4. `fincount`, the number of seizures in the 8 weeks after start of treatment.

To load the data, type:

```

> seizure.dat <- read.table("/users/math/faculty/sneddon/DATA/CASE2202.ASC",
                           header = T)

```

We want a model that can estimate the number of seizures after treatment.

Begin by plotting the other variables vs. `fincount`. In particular, plot `basecount` vs. `fincount` and `log(basecount)` vs. `fincount`. Which would be more appropriate to use?

The factor command

Next, use the `glm` command to fit a log-linear model that estimates `fincount` from the other 3 variables, using `basecount` and no interaction terms.

```
> seizure.fit <- glm(fincount~treat+age+log(basecount), family = poisson,
                    data = seizure.dat)
```

Write your fitted log-linear model here:

Before doing any tests on this model, we should investigate if overdispersion is an issue. From class, what should we examine to see if overdispersion is a potential problem?

Is overdispersion a problem in this example?

To account for overdispersion, we make the following change in our **summary** command, as discussed in class:

```
> summary(seizure.fit,
          dispersion = seizure.fit$deviance/seizure.fit$df.residual)
```

Our drop-in-deviance test statistic becomes:

$$F_{obs} = \frac{(\text{Deviance}_R - \text{Deviance}_F)/(p - q)}{\hat{\sigma}^2}$$

where $\hat{\sigma}^2 = \text{Deviance}_F/(n - p - 1)$. This follows an F-distribution with $(p - q)$ and $(n - p - 1)$ df.

Now, for the epilepsy example, answer the following (you can use the back of the page):

1. Test if the model is useful;
2. If it is useful, test if **treat** can be dropped from the model.

When you're done, you can exit R and end your script file.

```
> q()

$ exit
Script done, file is lab8
$ $HOME/descript lab8 > lab8.sc
```