

Offsets in GLM and MCMCpack*

Gebrenegus Ghilagaber
Department of Statistics, Stockholm University

November 27, 2008

1 Introduction

- An **offset** is a component of a linear predictor that is known in advance (typically from theory, or from a mechanistic model of the process).
- Because it is known, it requires no parameter to be estimated from the data
- For linear models with normal errors an **offset** is redundant, since one can simply subtract the **offset** from the values of the response variable and work with the residuals instead of the y values.
- For Generalized Linear Models (GLM), however, it is necessary to specify part of the variation in the response using an **offset**.
- In such a case, the **offset** is held constant while the other explanatory variables are evaluated.

2 Example 1 (timber data from the R-Book)

- Consider the relationship between the *volume*, v , the *girth*, g , and the *height*, h , of a (roughly) cylindrical object like firewood:

$$v = \frac{g^2}{4\pi} h$$

*Contribution (partial) to Session 5 in the R Study Group (24 November 2008)

- Taking logarithms gives

$$\log(v) = \log\left(\frac{1}{4\pi}\right) + 2\log(g) + \log(h)$$

- Thus, if we fit a multiple linear regression of $\log(v)$ on $\log(g)$ and $\log(h)$, we expect estimated slopes of 2.0 for $\log(g)$ and 1.0 for $\log(h)$.

```

- > data<-read.delim("E:\\timber.txt")
  > attach(data)
  > names(data)
  > girth<-girth/100      # converting centimeters to meters so
                           that girth and height have the same units
  > model1<-glm(log(volume)~log(girth)+log(height))  # this is
the usual glm
  > summary(model1)

```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-2.89938	0.63767	-4.547	9.56e-05 ***
log(girth)	1.98267	0.07503	26.426	< 2e-16 ***
log(height)	1.11714	0.20448	5.463	7.83e-06 ***

Residual deviance: 0.18555 on 28 degrees of freedom

AIC: -62.697

The estimates are reasonably close to expectation (1.98267 rather 2.0 for $\log(g)$ and 1.11714 rather than 1 for $\log(h)$)

Let's now use **offset** to specify the theoretical response of $\log(v)$ to $\log(h)$:

```

> model2<-glm(log(volume)~log(girth)+offset(log(height))
> summary(model2)

```

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-2.53419	0.01457	-174.0	<2e-16 ***
log(girth)	2.00545	0.06287	31.9	<2e-16 ***

Residual deviance: 0.18772 on 29 degrees of freedom

AIC: -64.336

Naturally, the residual deviance is greater, but only by a very small amount

The AIC has gone down from -62.697 to -64.336, so the model simplification was justified

Let's now try including the theoretical slope (2.0) for $\log(g)$ in the offset as well:

```
> model3<-glm(log(volume)~1+offset(log(height)+2*log(girth)))
> summary(model3)
```

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-2.53403	0.01421	-178.3	<2e-16 ***

Residual deviance: 0.18777 on 30 degrees of freedom
AIC: -66.328

Again, the residual deviance is only marginally greater, and AIC is smaller, so the simplification is justified

How about the intercept?

If our theoretical model of cylindrical logs (firewoods) is correct, then the intercept should be

$$\log\left(\frac{1}{4\pi}\right) = -2.531024$$

This is almost exactly the same as the intercept estimated by GLM in model3.

Thus, we are justified in putting the entire model in the offset and informing GLM not to estimate an intercept from the data ($y \sim 1$) :

```
> model4<-glm(log(volume)~offset(log(1/(4*pi))+log(height)+2*log(girth))-1)
> summary(model4)
```

No Coefficients

Residual deviance: 0.18805 on 31 degrees of freedom
AIC: -68.282

This is a rather curious model with no estimated parameters. But it has a residual deviance of just 0.18805 (compared with model1, where all

three parameters were estimated from the data, which had a deviance of 0.18555). Because we were saving one degree of freedom with each step in the procedure, AIC became smaller with each step, justifying all of the model simplifications.

3 Poisson Regression of Count Data (Analysis of grouped survival data)

3.1 General

Consider an Occ/Exp rate, $\lambda_{ij} = D_{ij}/T_{ij}$, and a two factor multiplicative intensity model for the rate:

$$\lambda_{ij} = \beta_i \alpha_j, \quad i = 1, \dots, I; \quad j = 1, \dots, J \quad (1)$$

Taking logarithms, we have

$$\log(\lambda_{ij}) = \log(D_{ij}) - \log(T_{ij}) = \log(\beta_i) + \log(\alpha_j)$$

so that

$$\begin{aligned} \log(D_{ij}) &= \log(\beta_i) + \log(\alpha_j) + \log(T_{ij}) \\ &= A_i + B_j + \log(T_{ij}) = \underbrace{(\bar{A} + \bar{B})}_{\Delta} + \underbrace{(A_i - \bar{A})}_{a_i} + \underbrace{(B_j - \bar{B})}_{b_j} + \log(T_{ij}) \\ &= \Delta + a_i + b_j + \log(T_{ij}). \end{aligned}$$

Standard programs like SAS/SPSS yield estimates of the intercept (Δ) and the factor effects (a_i and b_j) such that one of the a_i 's and b_j 's is set to zero (by default the one corresponding to the highest level, but this can be changed).

But, in order to get the correct estimates of Δ , a_i , and b_j , we need to "inform" the corresponding procedure (GENMOD in SAS and General Loglinear in SPSS) that part of the variation in the response (occurrences or counts D_{ij}), is due to differences in exposure times. This is achieved by specifying $\log(T_{ij})$ as **offset** in the equation

$$\log(D_{ij}) = \Delta + a_i + b_j + \log(T_{ij}).$$

Thus,

$$\begin{aligned}\log(D_{ij}) &= \Delta + a_i + b_j + \log(T_{ij}) \implies \log\left(\frac{D_{ij}}{T_{ij}}\right) = \Delta + a_i + b_j \\ \implies \log(\lambda_{ij}) &= \Delta + a_i + b_j\end{aligned}$$

so that

$$\lambda_{ij} = \exp(\Delta + a_i + b_j).$$

If the last levels ($i = I$, and $j = J$) are used as baselines (reference levels), their corresponding values are set to zero ($a_I = b_J = 0$). Thus, the intensity for the baseline group (I, J) is given by $\lambda_{IJ} = \exp(\Delta + a_I + b_J) = \exp(\Delta + 0 + 0) = \exp(\Delta)$. Similarly, the intensities at i^{th} level of the first covariate and the J^{th} (baseline) level of the second covariate are given by

$$\hat{\beta}_i = \lambda_{iJ} = \exp(\Delta + a_i + b_J) = \exp(\Delta + a_i + 0) = \exp(\Delta + a_i)$$

while, the relative intensity of the j^{th} level (relative to the baseline J^{th} level) is given by

$$\hat{\alpha}_j = \frac{\lambda_{iJ}}{\lambda_{iJ}} = \frac{\exp(\Delta + a_i + b_j)}{\exp(\Delta + a_i)} = \exp(b_j)$$

3.2 Example 2: Poisson regression model of count data

Consider the following table of deaths (D_{ij}) and exposure times (T_{ij}) - in months, by Age and Region:

Age (i)	Region (j)	
	Urban($j = 1$)	Rural ($j = 2$)
< 12 months (i = 1)	$D_{11} = 147$	$D_{12} = 412$
	$T_{11} = 21615$	$D_{12} = 38392$
< 12 - 23 months (i = 2)	$D_{21} = 43$	$D_{22} = 132$
	$T_{21} = 15649$	$T_{22} = 26916$
24+ months (i = 3)	$D_{31} = 26$	$D_{32} = 83$
	$T_{31} = 18480$	$T_{32} = 32216$

Under the multiplicative intensity structure, the model takes the form:

$$D_{ij} \sim \text{Poisson}(\lambda_{ij}T_{ij})$$

where the inverse link function is:

$$\lambda_{ij} = \beta_i \alpha_j = \exp(\Delta + a_i + b_j)$$

with $\log(T_{ij})$ specified as **Offset**.

Note here that specifying $\log(T_{ij})$ as **Offset** amounts to "informing" the procedure that part of the variation in the D_{ij} is due to variations in the T_{ij} . Ignoring the **offset** means that we compare the deaths (D_{ij}) alone without due account of the exposures. (T_{ij}) which is not appropriate.

3.2.1 Parameter-estimation in SAS

Input file

```
data test;
input age region deaths exposures;
ltime = log(exposures);
if age = 2 then age2 = 1; else age2 =0;
if age = 3 then age3 = 1; else age3 =0;
if region = 2 then region2 = 1; else region2 =0;
cards;
1 1 147 21615
1 2 412 38392
2 1 43 15649
2 2 132 26916
3 1 26 18480
3 2 83 32216
;
run;
proc genmod data = test;
model deaths = age2 age3 region2/dist = poisson link = log
offset = ltime;
run;
```

Output file

Param	DF	Estimate	Std	95% CI		ChiSq	Pr > ChiSq
Intercept	1	-5.0235	0.0724	-5.1655	-4.8815	4808.10	<.0001
age2	1	-0.8145	0.0866	-0.9843	-0.6447	88.42	<.0001
age3	1	-1.4642	0.1047	-1.6694	-1.2590	195.55	<.0001
region2	1	0.5003	0.0789	0.3457	0.6550	40.22	<.0001

Thus, $\Delta = -5.0235$, $a_1 = 0$ (baseline), $a_2 = -0.8145$, $a_3 = -1.4642$, $b_1 = 0$ (baseline), and $b_2 = 0.5003$

3.2.2 Parameter-estimation in R

```
> Counts<-c(147,412,43,132,26,83)
> Exposures<-c(21615,38392,15649,26916,18480,32216)
> Age<-gl(3,2)      # gl stands for generate levels
> Region<-gl(2,1,6)
> lnexp<-log(Exposures)
> AgeRegionClaOff<-glm(Counts ~ Age + Region + offset(lnexp), pois-
son)
> summary(AgeRegionClaOff)
Call:
glm(formula = Counts ~ Age + Region + offset(lnexp), family = poisson)
Deviance Residuals:
 1  2  3  4  5  6
0.3952 -0.2326 -0.3905  0.2288 -0.4064  0.2356
Coefficients:
Estimate Std. Error z value Pr(>|z|)
(Intercept) -5.02348 0.07245 -69.340 < 2e-16 ***
Age2        -0.81452 0.08662  -9.403 < 2e-16 ***
Age3        -1.46420 0.10471 -13.984 < 2e-16 ***
Region2     0.50034 0.07890  6.342 2.27e-10 ***
—
Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
(Dispersion parameter for poisson family taken to be 1)

Null deviance: 326.4110 on 5 degrees of freedom
Residual deviance: 0.6358 on 2 degrees of freedom
AIC: 47.011
Number of Fisher Scoring iterations: 3
```

As we note the estimates (and associated quantities) are identical as those obtained in SAS above.

3.3 Doing it the Bayesian way in R (but without Offset!)

In the Task-View of the CRAN site I came across the package called **MCMCpack** which includes a program for Poisson Regression of count data in a Bayesian setting. Unfortunately, this does not yet support the **Offset** option. As a result the estimates from the Bayesian analysis below are not correct. But, the example may serve as a demonstration of the program, anyway. I also think that it is possible (though I have not yet tried it) to do the equivalent of Poisson regression of count data in WinBUGS:

Input to R

```
> Counts<-c(147,412,43,132,26,83)
> Age<-gl(3,2)
> Region<-gl(2,1,6)
> ModelBayes<-MCMCpoisson(Counts ~ Age + Region)
> Summary(ModelBayes)
```

Output from R

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
The Metropolis acceptance rate for beta was 0.34109
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
> summary(ModelBayes)
Iterations = 1001:11000
Thinning interval = 1
Number of chains = 1
Sample size per chain = 10000
1. Empirical mean and standard deviation for each variable,
plus standard error of the mean:
Mean      SD Naive SE Time-series SE
(Intercept) 4.964 0.07237 0.0007237 0.002785
Age2       -1.162 0.08629 0.0008629 0.003235
Age3       -1.638 0.10595 0.0010595 0.004011
Region2    1.066 0.07758 0.0007758 0.002770
2. Quantiles for each variable:
2.5% 25% 50% 75% 97.5%
(Intercept) 4.821 4.917 4.964 5.013 5.1077
Age2 -1.328 -1.221 -1.161 -1.102 -0.9934
Age3 -1.865 -1.706 -1.637 -1.566 -1.4445
```

Region2 0.911 1.014 1.065 1.119 1.2171

Obtaining plots of posterior distributions

```
> plot(ModelBayes)
```

gives plots of the trace-history and posterior densities as shown below.

