

Arizona State University

From the Selected Works of Joseph M Hilbe

1993

Log-Negative Binomial Regression as a Generalized Linear Model

Joseph Hilbe, *Arizona State University*



Available at: https://works.bepress.com/joseph_hilbe/11/

LOG NEGATIVE BINOMIAL REGRESSION

AS A

GENERALIZED LINEAR MODEL

Joseph M. Hilbe
Arizona State University

Department of Sociology
and
Graduate College Committee on Statistics

Technical Report COS 93/94-5-26

LOG NEGATIVE BINOMIAL REGRESSION AS A GENERALIZED LINEAR MODEL

Joseph M. Hilbe

(c) 1994 Joseph Hilbe
Contact: atjmh@asuvm.inre.asu.edu

The negative binomial (NB) is a member of the exponential family of discrete probability distributions. The nature of the distribution is itself well understood, but its contribution to regression modeling, in particular as a generalized linear model (GLM), has not been appreciated. The mathematical properties of the negative binomial are derived and GLM algorithms are developed for both the canonical and log form. Geometric regression is seen as an instance of the NB. The log forms of both may be effectively used to model types of Poisson-overdispersed count data. A GLM-type algorithm is created for a general log negative binomial regression (LNB) which iteratively estimates a heterogeneity factor in addition to parameters and standard errors. Applications in terms of data generated from a NB random number generator are presented and modeled using a LNB SAS macro and a like program written in Stata.

KEY WORDS: Generalized linear model, Count data, Poisson, Overdispersion, Negative binomial, Log negative binomial, Geometric, Random number generator.

1. Introduction

Poisson regression, a member of the class of models known as generalized linear models (GLM), is the standard method used to analyze count data. However, many real data situations violate the assumptions upon which the Poisson model is based. For instance, the Poisson model assumes that the mean and variance of the response are identical. This means that events occur within a period of observation at a constant rate; an event is equally likely at any point within the period. When there is heterogeneity in the data, it is likely that the Poisson model is overdispersed. Such overdispersion is indicated if the variance of the response is greater than its mean. We may also check for model overdispersion by submitting the data to a Poisson model and observing the Chi²-based or Deviance-based dispersion statistic. The model is Poisson-overdispersed if the dispersion value is greater than unity. Several software programs provide such information to the user.

Log negative binomial regression can rather effectively be used to model count data in which the response variance is greater than that of the mean. I shall first describe the mathematical properties of the standard negative binomial. In section 3 I shall develop regression algorithms for both the canonical and log-linked form including an algorithm to estimate a heterogeneity factor in addition to parameters and standard errors. Section 4 deals with applications of the algorithms to appropriate synthetic data. Proofs of indicated mathematical transformations are listed in Appendix A; a log negative binomial SAS macro, constructed in the manner described in this paper, is provided in Appendix B.

Differentiating again, that is, finding the second moment of $b(\theta)$, defines the variance V .

$$V = b''(\theta) = r(1-p)/p^2 \quad (2)$$

To cast this specification of the variance function in terms of μ :

$$\begin{aligned} V &= (r-pr)/p^2 \\ &= \mu + \mu^2/r \end{aligned} \quad (3)$$

The linear predictor, $X'\beta$, η , or θ , can be translated in terms of μ and r as:

$$\begin{aligned} \theta &= \ln(\mu/(\mu+r)), \text{ or} \\ &= -\ln((\mu+r)/\mu) \end{aligned} \quad (4)$$

The canonical inverse link function, which determines the value of μ in terms of θ or η , takes the form of:

$$\begin{aligned} \mu &= r/(e^{-\theta}-1), \text{ or} \\ &= r/(e^{-\eta}-1) \end{aligned} \quad (5)$$

Reparameterizing r so that it is directionally proportional to μ^2 in the second term of the variance function, we define $k = 1/r$. Parameterized in terms of k , results of the above appear as:

$$\begin{aligned} \theta \text{ or } \eta &= -\ln(r/\mu+1) &= -\ln(1/k\mu+1) &\text{ or } -\ln(1+1/k\mu) \\ b(\theta) &= -r\ln(p) &= -\ln(p)/k &\text{ or } -\ln(1/(\mu k+1)) \\ \mu &= r(1-p)/p &= (1-p)/kp &\text{ or } \frac{1/k(e^{-\theta}-1)}{1/k(e^{-\eta}-1)} \\ V &= r(1-p)/p^2 &= \mu + \mu^2/r &= \mu + k\mu^2 \end{aligned}$$

The negative binomial pdf, parameterized in terms of μ and k , can be formulated as:

$$f(y; \mu, k)_i = \frac{\Gamma(y+1/k)}{\Gamma(y+1)\Gamma(1/k)} \frac{(k\mu)^y}{(1+k\mu)^{y+1/k}}$$

In exponential form the above appears as:

$$f(y; \mu, k)_i = \exp\{ y\ln(1/(1+1/k\mu)) - 1/k \ln(1+k\mu) + \dots \} \quad (6)$$

Note that the results displayed in the summary table above can be easily reproduced using the reparameterized exponential form of the pdf.

The log-likelihood function can be derived from the pdf as:

$$LL(\mu; y, k)_i = y \ln(k\mu) - (y+1/k) \ln(1+k\mu) + \ln \Gamma(y+1/k) - \ln \Gamma(y+1) - \ln \Gamma(1/k)$$

where $\ln \Gamma$ is the log-gamma function. For estimation purposes we use the GLM approach of employing the deviance function as the basis of convergence. The deviance is defined as:

$$D/2 (y; \mu, k)_i = l(y; y) - l(\mu; y)$$

The first term of the function represents the maximal fit. It is easily determined by substituting the value of y for each μ in the log-likelihood function. Hence,

$$D/2 (y; \mu, k)_i = y \ln(y/\mu) - (1+ky)/k \ln((1+ky)/(1+k\mu)) \quad (7)$$

Notice that the log-gamma functions as well as the first term k cancel, thus simplifying the algorithm.

The quasi-likelihood approach of Wedderburn (1974) can also be used to calculate the deviance. Since the negative binomial is a member of the exponential family of distributions, the derived quasi-loglikelihood is identical to the loglikelihood. Using the McCullagh & Nelder (1989) specification:

$$\int_{\mu}^y (y-\mu) V(\mu)^{-1}$$

and substituting the NB variance, $\mu+k\mu^2$, for $V(\mu)$,

$$\int_{\mu}^y \frac{(y-\mu)}{(\mu+k\mu^2)}$$

Integration on μ yields

$$y \ln(\mu) - \frac{1+ky}{k} \ln(1+k\mu) \Big|_{\mu}^y$$

The deviance calculates to the same value as above.

3. The GLM negative binomial and geometric algorithms

The methodology of generalized linear models (GLMs) requires that estimation be in terms of μ ; i.e. in the case of the negative binomial k is placed into the variance and deviance function as a known constant as μ varies. A method to circumvent problems related to the value of k will be examined later in this section.

GLMs share a common estimation scheme. Although the algorithm may be formatted in various ways, the canonical negative binomial GLM paradigm can be structured as follows:

```
 $\mu$  = (y+mean(y))/2          /* Nelder initialization */
 $\eta$  = -ln(1/k $\mu$ +1)          /* link (k parameterization) */
i = 1
WHILE (abs( $\Delta$ dev)>tolerance) {
    w =  $\mu$ +k $\mu^2$               /* variance function */
    z =  $\eta$  + (y- $\mu$ )/w - offset /* derivative */
     $\beta$  = (X'wX)-1X'wz        /* regression */
     $\eta$  = X' $\beta$  + offset        /* linear predictor */
     $\mu$  = 1/k(e- $\eta$ -1)          /* inverse link */
    oldDev = dev
    dev = 2  $\sum$  {yln(y/ $\mu$ ) - (1+ky)/k ln((1+ky)/(1+k $\mu$ ))}
     $\Delta$ dev = dev - oldDev
    i = i + 1
}
```

It should be noted that for a value of $k=1$, the negative binomial model becomes a geometric regression with the following properties:

$$\text{Link} : \ln(\mu/(1+\mu)) \qquad \text{Inverse Link: } 1/(e^{-\eta}-1)$$

$$\text{Variance: } \mu(1+\mu)$$

$$LL(\mu; y)_i = y \ln(\mu) - (1+y) \ln(1+\mu)$$

$$D/2(y; \mu)_i = y \ln(y/\mu) - (1+y) \ln((1+y)/(1+\mu))$$

In this respect a comparison can be made between the relationship of the exponential to the gamma and the geometric to the negative binomial distributions.

To construct a log-linked negative binomial or geometric model one substitutes the log and inverse log links, which are the same as the canonical Poisson form, and adjusts the weighting. The initial η link then takes the form of

$$\eta = \log(\mu)$$

the inverse link becomes

$$\mu = \exp(\eta)$$

the initial weight is the Poisson variance

$$w = \mu$$

and a weight adjustment made just prior to the regression is

$$\begin{aligned} w &= w^2 / (\mu + k\mu^2) \\ &= \mu / (1 + k\mu) \end{aligned}$$

The above algorithm may be adjusted so that the actual or observed information matrix (IM) is used rather than the expected at each iteration. For data sets consisting of larger than 50 observations, the difference in standard errors between the two models are negligible. The algorithms using the expected IM, called Fisher's scoring, entail fewer calculations and are hence faster, but at the expense of accuracy when dealing with very small data situations. The expected IM slightly underestimates standard errors; greater underestimation occurring with increasingly smaller numbers of responses. This occurs only with non-canonical links, the log negative binomial being no exception. The algebra for canonical links result in the adjustments made for the observed-based algorithm reducing to the expected (Aitkin et al. 1989). On the other hand, observed-based IM have difficulty passing the first iteration when there are very large numbers of cases. Perhaps the optimal solution is to start the algorithm with a single iteration based on the expected IM, then change to the observed-based IM. This results in a very slight loss of calculational speed, but is more robust than the straight-forward approach. It is the approach I have employed in the LNB programs used later in this monograph.

The observed IM adjustment as applied to the LNB algorithm may be constructed as follows (individual subscripts are assumed):

Common to both constructions:

$$\eta = g(\mu) = \log(\mu); \text{ hence } g'(\mu) = 1/\mu \text{ and } g''(\mu) = -1/\mu^2$$

$$V(\mu) = \mu + k\mu^2; \text{ hence } V'(\mu) = 1 + 2k\mu \text{ and } V^2 = (\mu + k\mu^2)^2$$

$$u = (y - \mu)g'(\mu) = (y - \mu)/\mu$$

$$w^{-1} = V\{g'(\mu)\}^2 = (\mu + k\mu^2)/\mu^2 = (1 + k\mu)/\mu$$

$$w = \mu / (1 + k\mu)$$

The observed IM adjusts the weights w such that

$$\begin{aligned} w_o &= w + (y - \mu)\{V(\mu)g''(\mu) + V'(\mu)g'(\mu)\} / (V^2g'(\mu)^3) \\ &= \mu / (1 + k\mu) + (y - \mu)\{[-(\mu + k\mu^2)/\mu^2] + [(1 + 2k\mu)/\mu]\} / [(\mu + k\mu^2)^2/\mu^3] \\ &= \mu / (1 + k\mu) + (y - \mu)\{k\mu / (1 + 2k\mu + k^2\mu^2)\} \end{aligned}$$

and a revised working variate z_o is defined as

$$z_o = \eta + W_o^{-1}Wu$$

$$= \eta + (y - \mu) / \{w_o(1 + k\mu)\}$$

with W and W_o representing diagonal weight matrices. Substituting w_o and z_o into the LNB algorithm provides the proper adjustment.

The goal of the GLM-type LNB algorithm is to adjust the value of k so that the chi2-based or deviance based dispersion approximates 1, which is the optimal value for a Poisson model. k represents the amount of positive heterogeneity in the otherwise Poisson count data. Simulation studies have given support to the notion that the optimal LNB model results when k is adjusted in such a manner that the Chi2-based dispersion, defined as $\{\sum (y - \mu)^2 / V(\mu)\} / (df)$, approximates 1.0. An example will be shown in the next section.

A LNB algorithm, which I have written in SAS and in Stata and which is to be part of the XploRe statistical package, iteratively converges to a dispersion value of 1.0. The value of k need not be known prior to modeling, but is re-estimated with each iteration. The initial value of k is taken as the inverse of the Poisson dispersion².

```
Poisson y <predictors>
chi2 =  $\sum (y - \mu)^2 / \mu$ 
disp = chi2/df
 $\alpha = 1/\text{disp}$ 
j=1
while (abs( $\Delta$ disp)>tolerance) {
  olddisp = disp
  LNB y <predictors>, k= $\alpha$ 
  chi2 =  $\sum (y - \mu)^2 / (\mu + k\mu^2)$ 
  disp = chi2/df
   $\alpha = \text{disp} * \alpha$ 
   $\Delta$ disp = disp - oldisp
  j = j+1
}
```

4. Application of the log negative binomial

Log negative binomial (LNB) regression is most effectively used to model count data that violates the Poisson assumption of the equality of mean and variance. In effect, the LNB model is based upon the premise that events enter a period of observation with a gamma distribution. Noting that the Poisson and gamma variances are μ and μ^2/r respectively, the negative binomial is considered as a Poisson-gamma mixture distribution with a variance of $\mu + \mu^2/r$. I have previously reparameterized this as $\mu + k\mu^2$ to allow a linear relationship in the second term. The parameter k can be regarded as a heterogeneity factor and is entered into the function as a known constant. In fact, the standard negative binomial formulation requires that k be fixed and independent of μ . If it is parameterized in any other manner, then the distribution is not a member of a GLM type exponential family (Nelder 1993). As a constant, k quantifies the overall

amount of Poisson-overdispersion in the data.

I shall first provide a small sample NB constant-only model consisting of 20 observations. The mean and standard deviation of the array is:

Variable	Obs	Mean	Std. Dev.	Min	Max
-----+-----					
xnb	20	10.4	6.26099	1	24

Solving the NB variance function for k, we have $k = (V - \mu) / \mu^2$ or $k = (SD^2 - \text{mean}) / \text{mean}^2$

$$(6.26099^2 - 10.4) / 10.4^2 = .26627215$$

K has a value of 0.26627215. Using a LNB program I wrote in Stata, we can estimate the coefficient as well as alpha (k). Given this data, we should find that k approximates .26627 and that the exponentiated value of the coefficient approximates 10.4.

```
. glmlnb xnb
1:  Deviance: 23.57674  Alpha (k): .2653062  Disp: 1.002672
2:  Deviance: 23.53502  Alpha (k): .2660151  Disp: 1.00071
3:  Deviance: 23.52394  Alpha (k): .266204  Disp: 1.000188
4:  Deviance: 23.5210  Alpha (k): .2662541  Disp: 1.00005
5:  Deviance: 23.52021  Alpha (k): .2662674  Disp: 1.000013

No obs.      = 20
Poisson Dev = 77.13131
Neg Bin Dev = 23.52021
Prob > Chi2 = 2.44e-13

Poisson Dp = 3.76923
Alpha (k) = .2662709

Chi2 = 19.00024
Prob>chi2 = .4568205
Dispersion = 1.000013

Deviance = 23.52021
Prob>chi2 = .2152044
Dispersion = 1.237906

GLM: Log negative binomial regression (integrated IM)
-----+-----
xnb |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-----+-----
_cons | 2.341806   .1346145    17.396  0.000    2.072032    2.611579
-----+-----
Loglikelihood = -63.928711
exp(2.341806) = 10.400002
```

The modeling appears remarkably accurate.

As a more extensive constant-only model, I shall create a 10000 observation array that is distributed as standard negative binomial. The mean is set as 10 and k as 0.1. The random number generators (RNG) as well as the LNB and GLM Stata programs are available from the author.

```
. rndnbl 10000 10 .1
Variable |      Obs      Mean   Std. Dev.   Min      Max
-----+-----
xnb | 10000    10.0029   4.485678      0      37
```

Calculation of the appropriate value of k:

$$(4.485678^2 - 10.0029) / 10.0029^2 = .10112541$$

Submitted to the above described LNB program we find:

```
. glm1nb xnb
1:   Deviance: 3660.276 Alpha (k): .4971298 Disp: .336788
2:   Deviance: 7853.827 Alpha (k): .1674273 Disp: .752048
...
12:  Deviance: 10276.85 Alpha (k): .1011623 Disp: .9998164
13:  Deviance: 10277.75 Alpha (k): .1011437 Disp: .9999087

Poisson Dp = 2.011547
Alpha (k) = .1011245

Chi2 = 9998.087
Prob>chi2 = .5006952
Dispersion = .9999087

No obs. = 10000
Poisson Dev = 20125.70
Neg Bin Dev = 10277.75
Prob > Chi2 = 0.0000

Deviance = 10277.75
Prob>chi2 = .0251263
Dispersion = 1.027877

GLM: Log negative binomial regression (integrated IM)
-----
      xnb |      Coef.   Std. Err.      t    P>|t|    [95% Conf. Interval]
-----+-----
      _cons | 2.302875   .0044846   513.509   0.000    2.294085    2.311665
-----+-----
Loglikelihood = -28785.963

. di exp(2.302875)
10.002899
```

Note that the resultant value of alpha (k) is nearly identical to the value of k we calculated based on the summary information concerning the array. Exponentiating the estimated constant gives a value equal to that of the summary mean. Performing the same procedure on random variates created with various values of k produce similar results.

A more complex example can be created involving a 10000 observation log negative binomial data set using a random number generator which allows for input of a mean variable, defined as the inverse of the link function, and a value for k equal to 0.1. The model has two positive-valued random normal predictors together with coefficients set as $b_0=2$, $b_1=.5$, and $b_2=3$.

```
. set obs 10000
. gen x1=abs(invnorm(uniform()))
. gen x2=abs(invnorm(uniform()))
. gen byte b0=2
. gen b1=.5
. gen byte b2=3
. gen lp = b0 + b1*x1 + b2*x2
. gen mu = exp(lp) /* inverse log link */
. rndnblx mu, k(.1) /* Neg. Binomial random number generator, k=.1 */
```

Submitting the data to the above described program results in the following:


```

. glmnlb xnb x1 x2
1: Deviance: 71004.16 Alpha (k): .0066141 Disp: 7.103975
2: Deviance: 18688.18 Alpha (k): .0469862 Disp: 1.843185
...
7: Deviance: 10214.30 Alpha (k): .1001323 Disp: 1.000078
8: Deviance: 10213.64 Alpha (k): .1001401 Disp: 1.000012

Poisson Dp = 151.1929
Alpha (k) = .1001414

No obs. = 10000
Poisson Dev = 1470443
Neg Bin Dev = 10213.64
Prob > Chi2 = 0.0000

Chi2 = 9997.123
Prob>chi2 = .497728
Dispersion = 1.000012
Deviance = 10213.64
Prob>chi2 = .063523
Dispersion = 1.02167

```

GLM: Log negative binomial regression (integrated IM)

xnb	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
x1	.4992926	.0056736	88.003	0.000	.4881724	.5104128
x2	2.991436	.0056733	527.279	0.000	2.980316	3.002556
_cons	2.008955	.0077412	259.515	0.000	1.993782	2.024127

Loglikelihood = -51184.148

The estimates are nearly the same as those used to create the data set. Iteration on a deviance-based dispersion rather than the Chi2-based dispersion yields nearly identical coefficients and standard errors. However, the values for alpha (k) and LL change to 0.1027483 and -51185.2 respectively. Both of these values are somewhat worse than those produced using Chi2 as the bases of convergence. This fact, together with similar comparisons on other data sets, gives plausible reason to use Chi2 rather than deviance-based convergence.

Running the same model using Poisson regression results in estimates which are fairly close to those produced by the above LNB model. However, as expected, the standard errors are much too narrow. This provides evidence that the predictor significance values may be overly optimistic when modeling overdispersed data with Poisson regression; i.e. values may indicate that predictors significantly contribute to the model when in fact they do not. Moreover, model GOF statistics based on Chi2 and Deviance values prove satisfactory for the LNB but are clearly poor for the Poisson.

```

. glm xnb x1 x2, f(poi)
Iter 1 : Dev = 9572079.0000
Iter 2 : Dev = 2459654.2500
Iter 3 : Dev = 1496865.0000
...

```

Iter 7 : Dev = 1470443.1250

```

Chi2 = 1511475
Prob>chi2 = 0
Dispersion = 151.1929

```

```

No obs. = 10000
Deviance = 1470443
Prob>chi2 = 0
Dispersion = 147.0884

```

Poisson distribution
Canonical link

xnb	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
x1	.4375138	.0003826	1143.478	0.000	.4367639	.4382637
x2	2.96794	.0003005	9877.023	0.000	2.967351	2.968529
_cons	2.118263	.0009559	2215.872	0.000	2.116389	2.120136

Log-likelihood: -768064.9375

I shall next provide an example using a SAS log negative binomial macro on a random data set consisting of 200 observations with covariates and constant set at the same values as before. Data is stored in a file named "lnbxmp". The macro call is:

```
%hilbenb(dsin=lnbxmp, yvar=xnb, xvars=x1 x2);
```

Partial SAS output consists of the following:

```
Log Negative Binomial Regression
The GENMOD Procedure
Analysis Of Parameter Estimates
```

Parameter	DF	Estimate	Std Err	ChiSquare	Pr>Chi
INTERCEPT	1	2.0394	0.0804	643.3546	0.0000
X1	1	0.5041	0.0392	164.9752	0.0000
X2	1	2.9717	0.0530	3140.3094	0.0000
SCALE	0	1.0000	0.0000	.	.

NOTE: The scale parameter was held fixed.

```
Log Negative Binomial Regression
Number of iterations: 6
Alpha: 0.1114
Deviance: 197.9161 Deviance/DF: 1.0047
Pearson Chi2: 197.0015 Pearson Chi2/DF: 1.0000
LogLikelihood: -1324.6208
```

Stata program output on the same data results in:

```
. glmnlb xnb x1 x2
1: Deviance: 2654.544 Alpha (k): .0058463 Disp: 13.67701
2: Deviance: 272.3188 Alpha (k): .0799599 Disp: 1.378137
...
5: Deviance: 197.9161 Alpha (k): .1113618 Disp: 1.000008
6: Deviance: 197.9147 Alpha (k): .1113626 Disp: 1.0000
```

Poisson Dp = 171.0485	No obs. = 200
Alpha (k) = .1113627	Poisson Dev = 33058.57
	Neg Bin Dev = 197.9147
	Prob > Chi2 = 0.0000
Chi2 = 197	Deviance = 197.9147
Prob>chi2 = .4865995	Prob>chi2 = .4682813
Dispersion = 1	Dispersion = 1.004643

```
GLM: Log negative binomial regression (integrated IM)
-----+-----+-----+-----+-----+-----+-----+
xnb | Coef. Std. Err. t P>|t| [95% Conf. Interval]
-----+-----+-----+-----+-----+-----+
x1 | .5040734 .0392447 12.844 0.000 .427102 .5810449
x2 | 2.971669 .053029 56.039 0.000 2.867662 3.075676
_cons | 2.039392 .0804031 25.365 0.000 1.881695 2.197088
-----+-----+-----+-----+-----+-----+
Loglikelihood = -1324.6208
```

As can be observed, the output is nearly identical. The algorithms appear consistent and appropriate to the modeling of log negative binomial data. Moreover, it is clear that a foremost feature of LNB regression is that standard errors for otherwise overdispersed Poisson count data are properly adjusted.

Subsequent residual analysis must be performed prior to casting final judgments about a model. Standard negative binomial residuals can be defined as

Pearson :

$$(y-\mu)/\sqrt{\mu+k\mu^2}$$

Standardized Pearson:

$$(y-\mu)/\sqrt{(\mu+k\mu^2)*(1-h)}$$

Deviance:

$$\text{sgn}(y-\mu)*\sqrt{2*(y\ln(y/\mu)-(1+k\mu)\ln((1+ky)/(1+k\mu)))}$$

Standardized deviance:

$$\text{sgn}(y-\mu)*\sqrt{2*(y\ln(y/\mu)-(1+k\mu)\ln((1+ky)/(1+k\mu)))/(1-h)}$$

Summary:

The negative binomial model fits squarely within the GLM-type exponential family of distributions. The log-linked form of the distribution allows useful modeling of many types of overdispersed count data. However, if overdispersion results from autocorrelation, the event counts are not independent and hence the likelihood function is not equal to the product of the individual probability functions. Obviously the standard GLM-type model should not be used in these situations. But where there is independence and heterogeneity is not the result of longitudinal effects, the negative binomial may prove to be a powerful addition to an analysts statistical toolbox.

Acknowledgements:

I should like to thank Walter Linde-Zwirble of Health Outcomes Technologies in Pennsylvania for his assistance in the development of the related RNGs, and John Nelder, Imperial College, London, for his helpful suggestions. Gordon Johnston, SAS Institute, assisted in the development of the SAS macro.

Notes:

1. Lawless (1987) discussed the log-link specification of the negative binomial within the context of overdispersed Poisson data. However, it appears that relatively little attention has been given to his arguments. The negative binomial has heretofore been largely ignored in commercial software packages having GLM capabilities.

2. Breslow (1984), following William's (1982) method for accommodating extra-binomial variation, devised an algorithm that is somewhat similar to the one here proposed for estimating k , the heterogeneity or "index" factor. However, the initialization is different and the algorithm is based solely on Poisson errors. Modeling takes a quasi-likelihood approach. Lawless details the maximum likelihood technique of handling negative binomial regression, but appears to use Breslow's Poisson method to estimate LNB models. This paper takes a quite different tactic in that it starts from the probability distribution, derives moments etc., develops the appropriate GLM algorithms, and subsequently finds the LNB appropriate to model certain types of overdispersed count data. Moreover, the use of the observed information matrix is emphasized.

APPENDIX A: Proofs

Proof of (1): Determining the value of the mean, μ

$$\mu = b'(\theta) = dp/d(\theta) * d(b)/p \text{ using the chain rule}$$

Given θ as equal to $\ln(1-p)$
and $b(\theta)$ equal to $-r \ln(p)$

$$\begin{aligned} d(\theta)/dp &= -1/(1-p) \\ dp/d(\theta) &= (1-p)/-1 \\ d(-r \ln(p))/dp &= -r/p \end{aligned}$$

$$\begin{aligned} \text{Hence } b'(\theta) &= (1-p)/-1 * -r/p \\ &= r(1-p)/p \end{aligned}$$

Proof of (2): Determining the value of the variance, V

$$V = b''(\theta) = dp/d(\theta) * d(b')/p \text{ using the chain rule}$$

With $dp/d(\theta)$ equal to $(1-p)/-1$ it remains to calculate $d(b')/p$

$$\begin{aligned} d(r(1-p)/p)/dp &= (-r/p) - r(1-p)/p^2 \\ &= -r/p^2 \end{aligned}$$

$$\begin{aligned} \text{Hence } b''(\theta) &= (1-p)/-1 * -r/p^2 \\ &= r(1-p)/p^2 \end{aligned}$$

Proof of (3): Determining V in terms of μ

$$\begin{aligned} V &= r(1-p)/p^2 \\ &= r(1-p)/p * 1/p \\ &= r(1-p)/p * \{p/p + (1-p)/p\} \\ &= r(1-p)/p * \{1 + (1-p)/p\} \\ &= r(1-p)/p * \{1 + r(1-p)/rp\} \\ &= r(1-p)/p * \{1 + (r(1-p)/p)/r\} \end{aligned}$$

Having

$$\mu = r(1-p)/p$$

then

$$\begin{aligned} V &= \mu(1 + \mu/r) \\ &= \mu + \mu^2/r \end{aligned}$$

Proof of (4): determining the link function in terms of μ

$$\begin{aligned} \text{Since } \mu &= r(1-p)/p \\ &= r((1-p)/p), \end{aligned}$$

$$\begin{aligned} \mu/r &= (1-p)/p \\ 1/p &= \mu/r + 1 \\ p &= r/(\mu+r) \\ 1-p &= \mu/(\mu+r) \end{aligned}$$

$$\begin{aligned} \text{Recalling that } \theta &= \ln(1-p), \\ \text{then } \theta &= \ln(\mu/(\mu+r)) \text{ or } -\ln((\mu+r)/\mu) \end{aligned}$$

Proof of (5): Determining the inverse link function

$$\begin{aligned} \text{Given } \theta &= -\ln((\mu+r)/\mu) \\ &= -\ln(r/\mu + 1) \\ e^{-\theta} &= r/\mu + 1 \\ r/\mu &= e^{-\theta} - 1 \\ \mu &= r/(e^{-\theta} - 1) \text{ or } r/(e^{-\eta} - 1) \end{aligned}$$

Proof of (6): Transforming to exponential family form

$$\begin{aligned} f(y; \mu, k)_i &= \exp\{ y \ln(k\mu) - \ln(1+k\mu) \} - 1/k \ln(1+k\mu) + \dots \\ &= \exp\{ y \ln(k\mu/(1+k\mu)) - 1/k \ln(1+k\mu) + \dots \} \\ &= \exp\{ y \ln(1/(1+1/k\mu)) - 1/k \ln(1+k\mu) + \dots \} \end{aligned}$$

Proof of (7): Determining the Deviance from the LL function

$$\begin{aligned} D/2 (y; \mu, k)_i &= y \ln(ky) - (y+1/k) \ln(1+ky) - y \ln(k\mu) + (y+1/k) \ln(1+k\mu) \\ &= y \ln(y/\mu) - (y+1/k) \ln((1+ky)/(1+k\mu)) \\ \text{or } y \ln(y/\mu) &- (1+ky)/k \ln((1+ky)/(1+k\mu)) \end{aligned}$$

APPENDIX B: SAS log negative binomial macro

```

/*****
/*
/*          SAS SAMPLE LIBRARY
/*
/*
/* NAME      : GLM - LOG NEGATIVE BINOMIAL REGRESSION , Version 1.0
/* TITLE     : NEGATIVE BINOMIAL, LOG LINK WITH
/*           HETEROGENEITY PARAMETER ESTIMATION
/* PRODUCT   : SAS
/* SYSTEM    : ALL, VERSIONS 6.08 AND ABOVE
/* KEYS      :
/* PROCS     : SAS/STAT GENMOD
/* SUPPORT   : SASGJJ
/* AUTHOR    : Joseph Hilbe, Arizona State University, Tempe, AZ
/*           Internet: atjmh@asuvm.inre.asu.edu
/*
/* REF       : Hilbe, Joseph (1994), Log Negative Binomial Regression
/*           Using the GENMOD Procedure SAS/STAT Software,
/*           Proceedings of SUGI 19, SAS Institute.
/*           Hilbe, Joseph (1994), Log Negative Binomial Regression as
/*           a Generalized Linear Model, Technical Report 26,
/*           Graduate College Committee on Statistics, Arizona
/*           State University, Tempe, AZ 85287.
/* MISC:
/*
*****/

/*****
|  USAGE:
|  1. Load data to be modeled into memory
|  2. Load this macro into memory: hilbenb.sas
|  3. Call macro with desired options
|
|  %hilbenb(dsin=<data name>, yvar=<response>, xvars=<predictors>,
|  clsvars=<factors>, offvar=<offset>, ithist=<default=0, set 1 to
|  show iteration history>, expected=<default=0, use of observed
|  information matrix; 1 = expected Fisher information matrix.
|
|  EXAMPLE:
|  Data set named lnbex with response xnb, two predictors x1 and x2,
|  a factor f1, and offset of1.
|
|  %hilbenb(dsin=lnbex, yvar=xnb, xvars=x1 x2 f1, clsvars=f1,
|  offvar=of1);
|
*****/

%macro hilbenb(dsin=, yvar=, xvars=, clsvars=, offvar=, ithist = 0, expected = 0);
* (c) 1994 Joseph Hilbe
* Uncomment the following for a debug trace;
* options mprint;
title Log Negative Binomial Regression ;

* Turn off printing;
%global _print_;
%let _print_ = OFF;
%let maxiter = 50;
%let iter = 1;
%let conv = 0;

%if %upcase(&offvar) ne %then %let offstmt = OFFSET=&offvar;
%else %let offstmt= ;
%if %upcase(&clsvars) ne %then %let clsstmt = %str(CLASS &clsvars;);
%else %let clsstmt= ;
%if(&expected=1) %then %let expstmt = EXPECTED ;
%else %let expstmt= ;

* Data is first modeled using Poisson regression;
proc genmod data=&dsin;
  &clsstmt
  make 'modfit' out=A;
  model &yvar = &xvars / dist = poisson
    &offstmt;
run;

```



```

data NULL ;
%if( &ithist = 1 ) %then %str(file print;);
set A;
if _N_ = 3 then
do;
    call symput( 'disp', put( valuedf, best10.6 ) );
    %if( &ithist = 1 ) %then
        %do;
            temp2 = 1/valuedf;
            put 'Iteration number: ' "&iter";
            put 'Pearson Chi2/DF: ' valuedf;
            put 'Alpha: ' temp2;
        %end;
    end;
run;

%let alpha = 1 / &disp;

* Iterate on NB model;
* Iterate until dispersion stops changing;
%do %while( &conv = 0 ) ;
* Data is now modeled using a log-linked negative binomial;
proc genmod data=&dsin;
    &clsstmt
    make 'modfit' out = A;

    _K = &alpha;
    _A = MEAN;
    _Y = RESP;
    variance _VAR = _A+_K*_A*_A;
    if ( _Y>0) then
        _D = 2 * ( _Y*log(_Y/_A)-
            (1+_K*_Y)/_K * log((1+_K*_Y)/(1+_K*_A)));
    else if ( _Y=0) then _D = 2 * log(1+_K*_A)/_K;
    deviance _DEV = _D;
    model &yvar = &xvars / &offstmt link = log itprint;
run;

%let iter = %eval( &iter + 1 );

data NULL ;
%if( &ithist = 1 ) %then %str(file print;);
set A;
if _N_ = 1 then
do;
    call symput( 'deviance', put( value, 10.4 ) );
    call symput( 'devdf', put( valuedf, 10.4 ) );
end;

if _N_ = 3 then
do;
    call symput( 'PX2', put( value, 10.4 ) );
    call symput( 'PX2df', put( valuedf, 10.4 ) );

    temp3 = &alpha;
    if ( ABS( valuedf - &disp ) <= 1.e-3 OR &iter > &maxiter )
        then call symput( 'conv', '1' ) ;
    else
        do;
            temp3 = valuedf * temp3;
            call symput( 'disp', put( valuedf, best10.6 ) );
            call symput( 'alpha', put( temp3, best20.10 ) );
        end;
    %if( &ithist = 1 ) %then
        %do;
            put 'Iteration number: ' "&iter";
            put 'Pearson Chi2/DF: ' valuedf;
            put 'Alpha: ' temp3;
        %end;
    end;
run;
* Print final model;
%if( &conv = 1 ) %then
%do;
    proc genmod data=&dsin;
        &clsstmt
        make 'modfit' out = A;

```



```

make 'parmes' out = B;
_K = &alpha;
_A = MEAN;
_Y = RESP;
variance _VAR = _A+_K*_A*_A;

_D = -2 * ( Y*log( _K*_A ) - ( Y+1/_K ) * log( 1+ _K*_A ) +
lgamma( _Y+1/_K ) - lgamma( _Y+1 ) - lgamma( 1/_K ) );

deviance _DEV = _D;
model &yvar = &xvars / &offstmt &expstmt link = log ;

run;
data NULL;
file print;
set A;
if _N_ = 5 then do;
  iter10 = &iter;
  alpha10 = &alpha;
  put 'Number of iterations: ' iter10 10.0 ;
  put 'Alpha: ' alpha10 10.4;
  put 'Deviance: ' "&deviance" "
  put 'Deviance/DF: ' "&devdf";
  put 'Pearson Chi2: ' "&PX2" "
  put 'Pearson Chi2/DF: ' "&PX2df";
  put 'LogLikelihood: ' value 10.4;
end;
run;
%end;
* End DO loop;
%end;
proc print data = B;
run;
%let _print_ = ON;
title;
options nomprint;
%mend hilbenb;

```

References:

- Aitken, Murray, et al. (1989) *Statistical Modeling in GLIM*, New York: Oxford University Press.
- Breslow, N. (1984) Extra-Poisson Variation in Log-linear Models. *Applied Statistics*. 33,1:38-44.
- Hilbe, J. (1993) Generalized Additive Models Software. *The American Statistician*. 47 1:59-64.
- Hilbe, J. (1993) GLM: A unified power-link based program including the negative binomial. *Stata Technical Bulletin* 14: 16-19.
- Hilbe, J. (1994) Log Negative Binomial Regression Using the GENMOD Procedure SAS/STAT Software, SUGI 19 Proceedings, SAS Institute.
- Hilbe, J. (1994) Negative Binomial Regression. *Stata Technical Bulletin* 18: 2-5.
- Hilbe, J. (1994) Generalized Linear Models. *The American Statistician*. 48 3:TBA.
- Lawless, J. (1987) Negative binomial and mixed Poisson regression. *Canadian Jour. Stat.* 15,3:209-225.
- McCullagh, P. and Nelder, J. A. (1989) *Generalized Linear Models*. 2nd ed. New York: Chapman & Hall.
- Nelder, J. and Lee, Y. (1992) Likelihood, Quasi-likelihood and Pseudolikelihood: Some Comparisons. *J.R. Statist. Soc B*, 54:1,273-284.
- Nelder, J. (1993) Generalized Linear Models with Negative Binomial or Beta-binomial Errors, unpublished manuscript.
- Wedderburn, R. (1974) Quasilielihood functions, generalized linear models and the Gauss-Newton method. *Biometrika* 61,439-47.
- Williams, D. (1982) Extra-binomial variation in logistic linear models. *Appl. Statist.*, 31:144-148

2. Mathematical properties of the negative binomial

The negative binomial is a member of the exponential family of discrete distributions. In this respect, the negative binomial is closely related to the Bernoulli, binomial, Poisson, hypergeometric, and geometric distributions. The first three have enjoyed substantial use as distributions underlying logistic and Poisson regression - the most noteworthy of the GLM family. The negative binomial distribution is itself well understood, but its contribution to regression modeling has not been appreciated.¹ This is unfortunate since its proper use can significantly enhance our perception of certain types of count data situations.

The negative binomial probability distribution can be thought of as the probability that there will be y failures before the r th success in a series of Bernoulli trials. Taken in this manner, the negative binomial pdf appears as

$$f_Y(y; r, p)_i = \binom{y+r-1}{r-1} p^r (1-p)^y$$

In the complete exponential family canonical form (McCullagh & Nelder, 1989)

$$f_Y(y; r, p)_i = \exp\{((y\theta - b(\theta))/\alpha(\phi)) + c(y, \phi)\}$$

the above appears as:

$$f_Y(y; r, p)_i = \exp\{y \ln(1-p) + r \ln(p) + \ln \binom{y+r-1}{r-1}\}$$

Hence,

$$\begin{aligned} \theta &= \ln(1-p) \\ e^\theta &= 1-p \\ p &= 1-e^\theta \\ b(\theta) &= -r \ln(p) \\ &= -r \ln(1-e^\theta) \end{aligned}$$

$\alpha(\phi)$, the scale parameter, is taken as 1, and $c(y, \phi)$ is the normalization factor.

The expected mean, μ , is determined by differentiating $b(\theta)$ with respect to the canonical parameter.

$$\mu = b'(\theta) = r(1-p)/p \quad (1)$$