Overcoming Overdispersion in Poisson Regression: quasi-Poisson and Negative Binomial Regression

Mark Krysan

August 3, 2023

Contents

Introduction

The anlaysis of count data has become very important in sceintific reserch, but simple linear and multilinear regression are unable to accuraly and effectivly model this type of data. Reserchers must use the Generalized Linear model, a family of regression models based on the Expoential family of distributions. A popular choice for modeling count data is the Poisson regression due to its simplicity and ease of implementation in R. However,becauce of the strong assumption that the data has equal mean and variance, Poisson regreesion is not always the best choice, since appliations rarly have equal mean and variance. In this paper, we will look at the effects that unequal mean and variances have on Poisson regression, more specifically, overdispersion, when the variacne is greater than the mean. We will also showcase Negative Binomial and Quasi-Poisson regression, two models that can account for overdispersion, and through a simulation study, how they are related to Poisson regression.

Generalized Linear Models

Before discussing Poisson regression models, we must first discribe the Generalized Linear Model, which, as its name suggests, genralizes linear regression. General linear models have two parts: the random component and the systematic componenet. The random component is the hypotheized distribution of the response variable, assumed to be from the exponential familiy of distributions. For example, if Y_i is binary, we assume binomial distribution. The systematic component, denoted $\eta = \beta_0 + \sum_{j=1}^{p} \beta_j x_j$, is the linear combination of the explanitory varibales, combined with the link function. The link function is responsible for relating the explanitory variables to the mean of the response variable. It is deoted $g(\cdot)$, is a monotonic, differentiable function such that $g(\mu) = \eta$, where $E[Y] = \mu$ [**Dunn2018**]. There are many link functions, such as the logit link, used in logisitic regression, where $g(\mu) = \ln(\frac{\mu}{1-\mu})$, or the identity link, where $\eta = g(\mu) = \mu$. For simple linear regression, we have that the random component is $Y_i \sim \mathbb{N}(\mu, \sigma^2)$, and the systemic component using the identity link to get, $g(\mu) = \mu = \mathbf{X}\beta$.

Poisson Regresssion

Poisson regression is a type of generalized liner model used for modeling count data. It is a special case of the more general negative binomial regression. Examples of problems that would requrie poisson regression include the number of chips in a chocholate chip cookie or the number of emails recived in an hour. Poisson regression uses Poisson distribution as the radom component with probability mass function, $p(y|\mu) = \frac{exp(-\mu)\mu^y}{y!}$ for $y = 0, 1, 2, \ldots$ and with expected counts $\mu > 0$. The link function used is the log link, or $g(\mu) = \ln(\mu)$. We can write Poisson regression as

$$\begin{cases} y \sim \text{Pois}(\mu) \\ log(\mu) = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p. \end{cases}$$

The poisson regression model estimates the maximum liklihood regression coeffiecents using the iteratively reweighted least squares algorthim, instead of ordianry least squares [**Hilbe2011**]. A key characteristic of poisson regression is that, following from the PMF, the mean is equal the variance, called equidispersion. Equidispersion is one of the main assumptions required for poisson regression, including non-negative, interger counts and independent observations. As noted by Hilbe, it is rare for a 'real life' poisson dataset to be equidisperised. More often than not, count data will have a variance greater than the mean, called overdispersion[**Hilbe2011**].

Overdispersion

As stated above, overdispersion is when the variance of a model is greater than the mean. In general, there are two types of overdispersion: apparent and real. Apparent overdispersion occurs when a model does not properly correspond to the obvserved data. This includes when when a model omits important explanitory predictors, the data includes outliers, the model lacks sufficent interaction terms, the predictors need to be transformed, or when the link function is mispecficed. Apprent overdispersion is an attribute that can be fixed or treated in the model. However, real overdispersion is when the data is inherenly overdispered.

Overdispersion in regression since it can drastically impact the standard errors of a model by making them much smaller than what they should be, leading to inflated p-values for the significnace of predictors. Reserchers must therefore be on the lookout for overdispersion in models due to their influcnce on infrence. The main way to determine if data is overdispered is by calculating the Pearson (χ^2) statistic for a model and dividing by the degrees of freedom [**Hilbe2011**]. The χ^2 statistic is of the form

$$T = \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i},$$

where E_i is the expected count for the ith obervation, and O_i is the observed count for the ith observation. The χ^2 statisitic measures the goodness of fit of a model. This statisitc divieded by the degrees of freedom is called the disperson paramter. For moderatly sized models, a model is oversidpersed if the dispersion statisitic is greater than 1.25, but for larger models, a statisitc greater than 1.05 is considered overdispered. If overdispersion is identified in a model, there are many different ways to handle it, both for real and apperent overdisperesion. A discussion of these methods can be found in Hilbe's book [**Hilbe2011**]. We will be disussing two common methods for handeling real overdispersion: negative binomial regression and quasi-poisson.

Negative Binomial Regression

A logical step to take when posed with the question of overdispersion is to relax the assumption of equidispersion. The negative binomial regression fufills this by letting μ be a random variable, creating extra variabilty in the model. This adds a lot of flexabilty when modeling count data [**Gardner1995**]. To add another layer of randomess, we let the random component of our model be

$$y_i|\lambda_i \sim \text{Pois}(\lambda_i) \text{ and } \lambda_i \sim \text{Gamma}(\mu_i, \psi), \tag{1}$$

where $\mu_i > 0$ and $\psi > 0$, and ψ is the coefficent of variablitiy [**Dunn2018**]. However, it is not clear that the distribution of y_i is negative binomial. We will now show that a negative binomial distribuition has the same pdf as a mixed poisson-gamma distribution.

Let $X \sim NB(r, p)$. We have that the pmf of X is

$$P(X = n) = \binom{r+n-1}{r-1} p^n (1-p)^n, \tag{2}$$

with mean of $\mu = \frac{r(1-p)}{p}$. First, notice that

$$\begin{aligned}\binom{r+n-1}{r-1} = \binom{r+n-1}{n} &= \frac{(r+n-1)(r+n-2)\dots(n)}{r!} \\ &= \frac{(r+n-1)!}{n!r!} \\ &= \frac{\Gamma(r+n)}{\Gamma(n+1)\Gamma(r)}.\end{aligned}$$

Now, since $\mu = \frac{r(1-p)}{p}$, we have that $p = \frac{r}{\mu+r}$ and $1 - p = \frac{\mu}{\mu+r}$. Therefore, we can rewrite (2) as

$$P(X = n) = \frac{\Gamma(r+n)}{\Gamma(n+1)\Gamma(r)} \left(\frac{r}{\mu+r}\right)^r \left(\frac{\mu}{\mu+r}\right)^n$$

Now, letting $y = n$ and $k = 1/\psi = r$, we get

$$P(y|\mu, k) = \frac{\Gamma(y+k)}{\Gamma(y+1)\Gamma(k)} \left(\frac{\mu}{\mu+k}\right)^y \left(\frac{1-\mu}{\mu+k}\right)^k. \tag{3}$$

Through messy reparameterization and using the Law of Total Probability to find the marginal distribution of y, as defined in (1), we find that it is equal to the equation in (3).

Therfore, we see that letting λ be random with the Gamma distribution, we get a negative binomial distribution. As noted by Dunn and Smyth, the variance of (1) is $\text{Var}(y_i) = \mu_i + \mu_i^2/k$. Notice that as $k \to \infty$, or $\psi \to 0$, $\text{Var}(y_i) \to \mu_i$ [**Dunn2018**]. Therefore, as the dispersion approaches ∞, the Negative Binomial regression approaches a Poisson regression. A consequce of the variance being defnied as such is that there is a conave relationship between the weighting of observaions and the mean. There is very little weight given to small means, and as the size of the mean increases, the weights level off to $\psi = 1/k$ [**VerHoef2007**].

Like Poisson regression, the estimation of the coefficents are done through the IRLS (a full description of this alogrithm can be found in Hilbe)[**Hilbe2011**]. Although there is another parameter being estimated in the Negative Binomial model, the esitmation of ψ is uncorrelated with the β_is. The coefficnent estimatiors will tend to be very similar to the Poisson model, since the dispersion of the data does not change the coefficent estimation, but the standard erros of the Negative Binomial will tend to be larger[**Hilbe2011**].

Quasi-Poisson Regression

Another model that accounts for overdispersion is the quasi-poisson regression. In practice, the quasi-poisson model is a poisson model, but rescaled so the dispersion estimate is 1. The quasi-poisson model is an instance of the quasi-liklihood family of models, which only requires the specification of the first and second moment of the distribution of data[**Gardner1995**]. However, quasi-poisson regression can be framed in the form of a GLM. To do this, we let Y be a random variable with $E[Y] = \mu$ and $\mathrm{Var}(Y) = \theta\mu$ where $\mu, \theta > 0$, and we use the log link. In the GLM framework, the only difference bewteen the Poisson and quasi-Poisson is the diffference in the variance of the random componenet. This definition of variance will lead observation wights being directly proportional to the mean [**Hilbe2011**]. Since the quasi-Poisson model is not deinfed by a complete probabilty density function, just by the first two moments, there are many statistical tools, such as AIC and BIC which are unable to be calcualted [**Hilbe2011**]. There are some statitiscs which have been created for quasi-Lilkihood models, but they cannot be compared to standard staitics, such as R-squared, AIC, or BIC [**VerHoef2007**].

The addition of a new parameter for the variacnce of Y allows for an increased flexiblilty in the modeling of the data, specificlly for overdispersed data. In the quasi-poisson model, the parameter θ will be estiamted as the inverse square root of the dispersion estimate[**Hilbe2011**]. This choice of θ will lead to a dispersion estimate of 1, transforming the data in a way to be equidispersed. However, this first requires a poisson model to be created, then a dispersion estimate to be calculated, then a new quasi-poisson model to be run with the choen value of θ. These steps are all taken care of in the MASS function **glm(formula, family = quasipoisson)**.

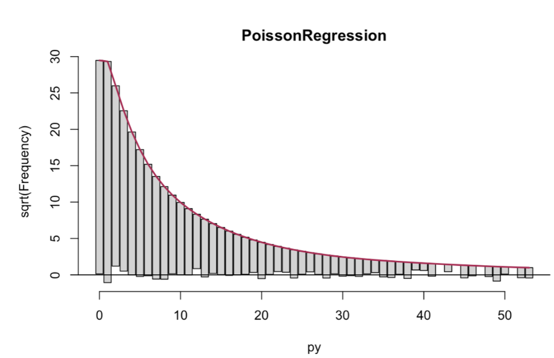
Like Poisson and Negative Binomial regression, the coefficents for quasi-Poisson regression are calculated using IRLS. However, due to the rescaling of the variance, there is one more itteration of the algorithm, except with the weight matrix multipled by the inverse square root of the dispersion parameter (more details of estimation of quasi-Poisson coefficents are outlined in [**Hilbe2011**]).

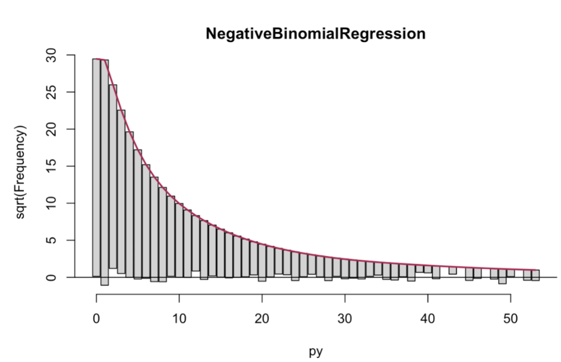
As mentioned above, overdispersion does not effect the estimated coefficents in the regression model, but the standard errors and the resulting statitics of those coefficents, such as confidence intervals and p-values. Therefore, for the quasi-poisson model, the only difference from the poisson model will be the standard errors and p-value of the coefficents of the model. These values will be larger than the Poisson model, accuarly accounting for the amount of variance in the model. An example of this will be demonstarted in the simulation study.

Simulation Study

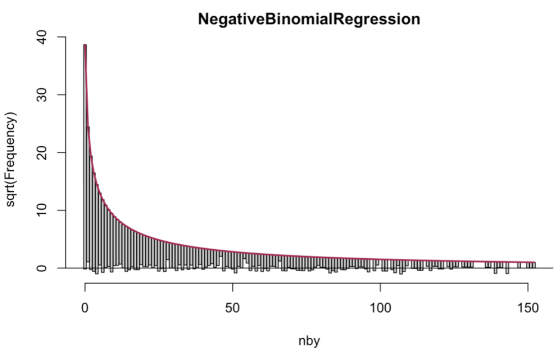
To demonstraite the relationships between the Poisson, quasi-Poisson, and Negative Binomial regression models and how they react to overdispersion, we have chosen to simulate count data. We have created two data sets, both based on the Poisson distribution, but one data set is equidisperesed and the other is overdisperesed. For the overdispersed data set, a Gamma distribution is used to simulate dispersion parameter. We used a function from [**Hilbe2011**] which takes in the number of observations and regression coefficnents desired for our simulation, and creates a dataframe with the explanitory and response variables (The code can be found in the Appendix: Figure 3 and 4). The explanitory values are normally distributed with mean of the desired coefficient and variance of 1. To get the response values function then takes the exponentiation of the explanitory varibales and uses this value as the mean of either a Poisson or Negative Binomial distribution. For the Negative Binomial distribution, the function also incorperates a varaince parameter, which is incorperated into a Gamma distribution. The values of the Gamma distribution are then multipled by the exponentiaion of the expalnitory values, and that is then use as the mean of a Poisson distribution.

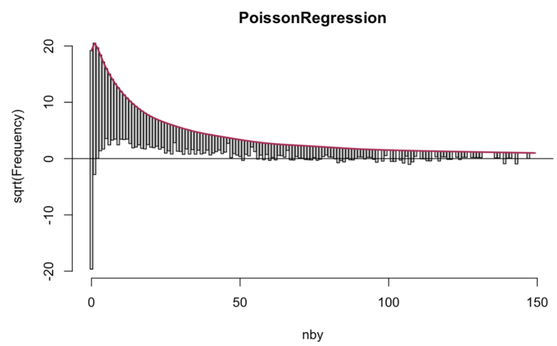
For both data sets, we ran Poisson, quasi-Poisson, and Negative Binomial regression, and compared thier results. For the equidisperesd dataset, we simulated that data with simulation coefficents of $\beta_0 = 1, \beta_1 = -.5$, and $\beta_2 = 1$ and 5000 observations. We then ran the corresponding regressions using the **glm** and **glm.nb** functions from the **MASS** package in R. As expected the Poisson and quassi-Poisson had the same coefficent estimates, with $\hat{\beta_0} = .9937, \hat{\beta_1} = -.5061$, and $\hat{\beta_2} = 1.0089$. The Negative Binomial model had slightly, but negligably different estimates of $\hat{\beta_0} = .9936, \hat{\beta_1} = -.5061$, and $\hat{\beta_2} = 1.0090$. The standard errors of all three models were all different from each other, but agian, in a negligable amount (The full models can be found in the Apendix: Figures 1 and 2). The two plots below comapre the fit of the Poisson and Negative Binomial models. The red line is the expected distribution of the data and it is overlayed on the observed counts. The boxes are placed to show the expected count, and the measeure of the distance is displayed in how far the bottom of the box is from $y = 0$. We see that both models fit the data very well. However, since the quasi-Poisson model is not defined by a probability distriution, we are unable to create this type of plot.





Rearding the overdispersed dataset, we start to see differecnes between the models. We created the dataset with simulation coefficents of $\beta_0 = 2, \beta_1 = .75$, and $\beta_2 = -1.25$ and a dispersion paramter of .5, this will lead to a variance of $\mu + \mu^2 \cdot 2$ as a fucntion of μ. We ran the same three models as above. The model results were vary similar to the equidispersed model, with a few notable diffrences (Full Models in Appendix: Figures 1 and 2). First off, the standard errors for the Negative Binomial and quasi-Poisson were much larger than those for the Poisson model. For the Negative Binomial, the standard errors were: .02, .02, and .02, for the respective coefficents, .03, .02, and .02 for the quasi-Poisson, and .006, .003, and .003 for the Poisson model. We see here that the Poisson model does not account for the true variablity in the model, resulting in very small error terms. These terms would then create very small p-vlaues, resulting in incorrect infrence. The following plots show how the Negative Binomial model fit the data very will, but the Poisson model underfit values very close to 0 and overfit values around 25.





Comparison Between Quasi-Poisson and NB2

Although both quasi-Poisson and Negative Binomial regression can model overdispersed count data, there are differences in the two which help determine which to use in a given situiation. First off, the quasi-Poisson model is overall more simple than the Negative Binoimal. Since the quasi-Poisson model acts as a transformed Poisson, it makes both the model and interpretation easier to work with. This simplicity does come with a draw back though. Since quasi-Poisson regression is a psudeo-transformation of a Poisson model, a Poisson model must be created first in order to detiermine the dispersion parameter. Therefore, question of p-hacking could be raised, depending on the goal of the model. If that is of no worry, then the quasi-Poisson model is best used for data that is not know to be overdispersed *a priori*.

However, to use Negative Binomial regression, the resercher should already know that the data is overdispersied before modeling. Since the dispersion is estimated at the same time as the mean, there is no worry of p-hacking. Another notiable differnece regarding selection of model is the relationship between the mean and variance. If it is known that a data set is overdispered before modeling, there are some caeses where the approximate relationship between the mean and variance can be determiend. If there is a linear realtionship, then the quasi-Poisson method will result in a better fitting model. However, if there is a quadradic relationship, the Negative Binomial model will be better. This is due to the definition of the variance in the two models. In addition, VerHoef and Boveng state that this difference in defintion of variance, specifically the ways in which the observations are weighted, are very helpful for choosing which model to use (See [**VerHoef2007**] for greater explination with real world data).

Appendix

Figure 1: Models of Equidispersed Data

	Poisson	quasi-Poisson	Negative Binomial
(Intercept)	0.993734	0.993734	0.993677
	(0.009323)	(0.009419)	(0.009330)
x1	−0.506117	−0.506117	−0.506107
	(0.006196)	(0.006260)	(0.006204)
x2	1.008970	1.008970	1.009050
	(0.005835)	(0.005895)	(0.005854)
Num.Obs.	5000	5000	5000
AIC	18772.9		18774.9
BIC	18792.4		18801.0

Figure 2: Models of Overdispersed Data

	Poisson	quasi-Poisson	Negative Binomial
(Intercept)	2.067111	2.067111	1.987917
	(0.005437)	(0.033657)	(0.021418)
x1	0.666306	0.666306	0.740751
	(0.003148)	(0.019486)	(0.021714)
x2	−1.212119	−1.212119	−1.257935
	(0.003389)	(0.020982)	(0.022579)
Num.Obs.	5000	5000	5000
AIC	153329.1		29856.7
BIC	153348.6		29882.8

The values in parantheses are the standard erors of the coefficnent estimate.

Figure 3

Equidisperesed Simulation Function

```
poisson_syn  <- function(nobs = 5000, xv = c(1, -.5,  1))  {
  p <- length(xv) - 1
  X <- cbind(1, matrix(rnorm(nobs * p), ncol = p))
  xb <- X %*% xv
  exb <- exp(xb)
  py <- rpois(nobs, exb)
  out <- data.frame(cbind(py, X[,-1]))
  names(out) <- c("py", paste("x", 1:p, sep=""))
  return(out)
}
```

Figure 4

Overdispersed Simulation Function

```
nb2_syn  <- function(nobs = 5000,
                     alpha = 1,
                     xv = c(1, 0.75, -1.5))  {
  p <- length(xv) - 1
  X <- cbind(1, matrix(rnorm(nobs * p), ncol = p))
  xb <- X %*% xv
  a <- alpha
  ia <- 1/a
  exb <- exp(xb)
  xg <- rgamma(nobs, a, a)
  xbg <-exb*xg
  nby <- rpois(nobs, xbg)
  out <- data.frame(cbind(nby, X[,-1]))
  names(out) <- c("nby", paste("x", 1:p, sep=""))
  return(out)
  }
```