Practical Assessment, Research, and Evaluation

Volume 21 *Volume 21, 2016* Article 2

2016

Tutorial on Using Regression Models with Count Outcomes using R

A. Alexander Beaujean

Morgan B. Grant

Recommended Citation
Beaujean, A. Alexander and Grant, Morgan B. (2016) "Tutorial on Using Regression Models with Count Outcomes using R," *Practical Assessment, Research, and Evaluation*: Vol. 21 , Article 2.
DOI: https://doi.org/10.7275/pj8c-h254
Available at: https://scholarworks.umass.edu/pare/vol21/iss1/2

Practical Assessment, Research & Evaluation

A peer-reviewed electronic journal.

Volume 21, Number 2, February 2016 ISSN 1531-7714

Tutorial on Using Regression Models with Count Outcomes using R

A. Alexander Beaujean, Grant B. Morgan, *Baylor University*

Education researchers often study count variables, such as times a student reached a goal, discipline referrals, and absences. Most researchers that study these variables use typical regression methods (i.e., ordinary least-squares) either with or without transforming the count variables. In either case, using typical regression for count data can produce parameter estimates that are biased, thus diminishing any inferences made from such data. As count-variable regression models are seldom taught in training programs, we present a tutorial to help educational researchers use such methods in their own research. We demonstrate analyzing and interpreting count data using Poisson, negative binomial, zero-inflated Poisson, and zero-inflated negative binomial regression models. The count regression methods are introduced through an example using the number of times students skipped class. The data for this example are freely available and the **R** syntax used run the example analyses are included in the Appendix.

Count variables such as number of times a student reached a goal, discipline referrals, and absences are ubiquitous in school settings. After a review of published single-case design studies Shadish and Sullivan (2011) recently concluded that nearly all outcome variables were some form of a count. Yet, most analyses they reviewed used traditional data analysis methods designed for normally-distributed continuous data.

It is not surprising that most educational research uses errant analysis techniques for count data. It is seldom taught in education coursework, and methods surveys (Aiken, West, & Millsap, 2008; Little, Akin-Little, & Lee, 2003; Perham, 2010) do not even ask about the use of count data. Nonetheless, using inappropriate regression methods can produce biased coefficients and standard errors, which can lead to errant conclusions. Consequently, for educational researchers to make the appropriate data-based decisions about positive and problem behaviors, as well as the effectiveness of interventions that target these areas, they must recognize and acknowledge the nature of the collected variables and use the appropriate data analytic tools.

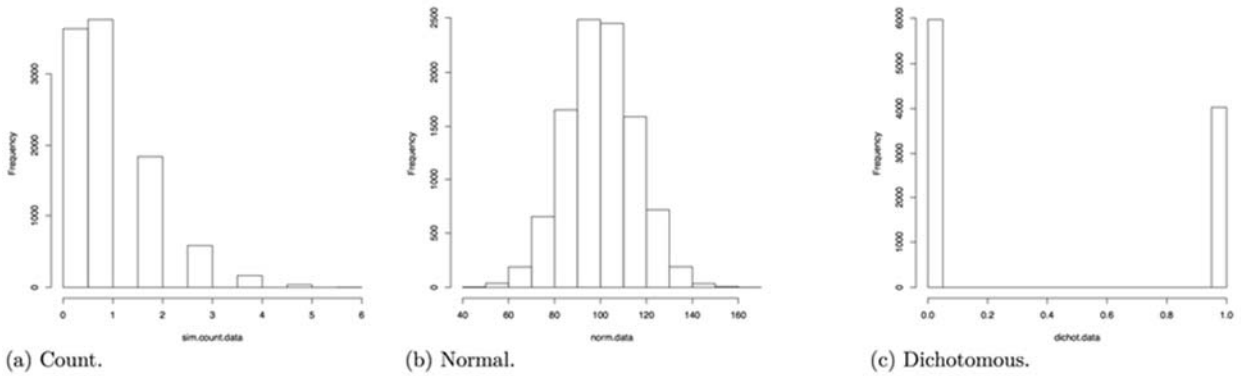
The purpose of this article is to assist educational researchers in understanding appropriate methods for count data, as well as being able to conduct independent analyses of such data. To that end, we discuss the nature of count data and present an example using freely available data. We provide the R (R Development Core Team, 2015) syntax to replicate and extend our analyses in the supplemental material. For those interested in using other software such (e.g., Stata, SAS) or extensions of the count models we discuss, Hilbe (2014) provides a book-length treatment on the topic as well as some worked examples.

Count Variables

Count variables share certain properties: (a) their values are always integers/whole numbers; (b) their lowest possible value is zero, so they can never be negative; and (c) they frequently appear to be positively skewed, with most values being low and relatively few values are high (Cameron & Trivedi, 1998). Figure 1a

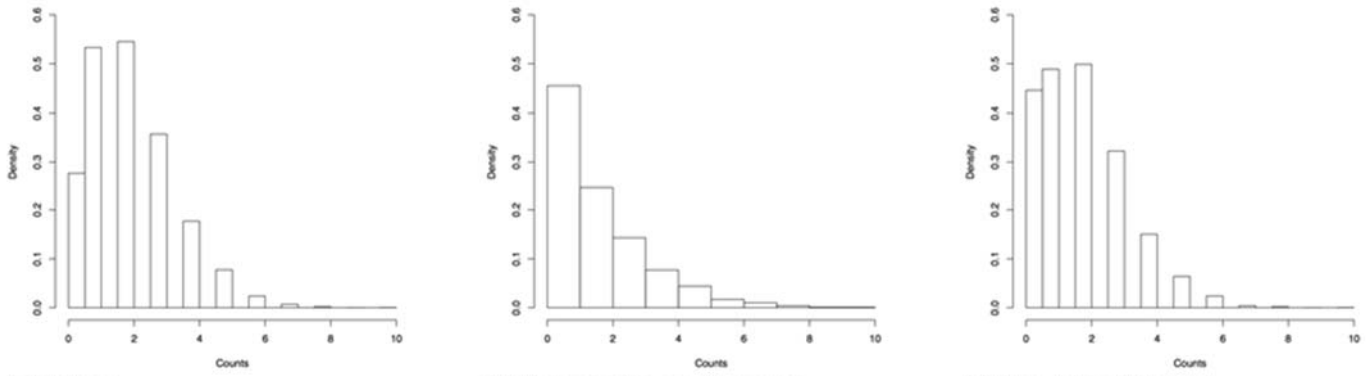
shows a histogram of a typical count variable. Histograms of normally distributed and dichotomous (binomial) variables are shown in Figures 1b and 1c, respectively.

To add to their complexity, some count variables have a considerable number of zero values (see Figures 2b and 2c). This typically occurs when the variable is thought to be particularly deleterious (e.g., drug use, school expulsions), and very few observations exhibit the behavior. No matter how many zeros a count variable has, the plots in Figure 2 make it obvious that such data are not even symmetrical, much less normally distributed.



(a) Count. (b) Normal. (c) Dichotomous.

Figure 1. Histograms of variable distributions.



(a) Poisson. (b) Poisson with overdispersion. (c) Zero-inflated Poisson.

Figure 2. Plots of count variable distributions.

Regression Models for Count Data

A major assumption of typical multiple regression models is that the residuals follow a normal distribution (Cohen, Cohen, West, & Aiken, 2003). Typically, the residuals follow the distribution of the outcome variable, which for count variables is often neither normally distributed nor even not even symmetrical (see Figure 2). Moreover, the residual variance often increases as the predictor variables increases, which produces heteroscedasticity. Thus, using typical regression methods with count outcome variables can lead to parameter bias as well as standard error and confidence interval estimates of the wrong size. This can ultimately lead an educational researcher to make invalid inferences and poor decisions. Instead, regression models that account for the count nature of the outcome variable (and the subsequent nature of the model's residuals) are more appropriate.

Prior to the development of regression models for count data and their availability in common statistical programs, count variables were typically dealt with in two ways. First, people used typical linear models,

surmising that the models were robust enough to handle any assumption violations caused by the count variables. Second, they transformed the count variables to make them fit more traditional models. Both approaches are problematic.

There is a lengthy literature devoted to the robustness of traditional linear regression (e.g., multiple regression, ANOVA) to departures from normality. Although previous research has shown that the traditional regression model can produce unbiased regression coefficients with some non-normal distributions (e.g., Box, 1953; Cochran, 1947; Lix, Keselman, & Keselman, 1996), it tends to produce inflated standard errors. Moreover, the traditional regression model makes continuous predictions even though count outcomes are discrete. Therefore, the residuals tend to be heteroscedastic, which violates a major assumption of the traditional regression model.

There are a variety of transformations available for count data. One transformation involves dichotomizing the responses (e.g., yes-no/present-absent), which is then used in a logistic regression. Problems with dichotomizing variables are well known, however, and are seldom appropriate (MacCallum, Zhang, Preacher, & Rucker, 2002). Another option is a nonlinear transformation (e.g., square root, logarithm) to make the variable more closely approximate a normal distribution. Unfortunately, such transformations often have little effect when the range of values is very narrow, do not handle having an excessive amount of small values well, and do not completely eliminate heteroscedasticity (Coxe, West & Aiken, 2009). The more general Box-Cox power transformations tend to work better, but they do not always fix the problems with normality and heteroscedasticity (Sakia, 1992). Moreover, any nonlinear transformation of the outcome comes at the cost of having a more difficult model to interpret (e.g., predicting the square root of times using a drug). This cost may acceptable when there are no known models to handle the outcome variable's native distribution. When models exist that can directly handle the variable's distribution, it is better to use them. With the development of the generalized linear model, models now exist that can directly handle the distribution of the count variables.

The generalized linear model (GLM; McCullagh & Nelder, 1989) is a framework designed to handle regression models for a variety of outcome variable types. All GLMs require two components: proper specification of residuals' distribution and a function to link the outcome and the linear combination of the predictor variables. In a typical regression, the residuals follow is a normal distribution and the link is the identity function (i.e., multiply the regression by one). For a logistic regression, the residuals follow a binomial distribution and the link is the logit function.

Count variables need to be modeled differently than either continuous or dichotomous variables (Cameron & Trivedi, 1998). Because of the different ways count variables can be distributed, there are multiple forms of the GLM for count data. In what follows, we discuss four common types of GLMs for count data, each of which is designed for a different type of count variable distribution.

Poisson Model

The most common type of distribution for count variables is a Poisson distribution, an example is shown in Figure 2a. The Poisson distribution is used because it is a probability distribution designed for non-negative

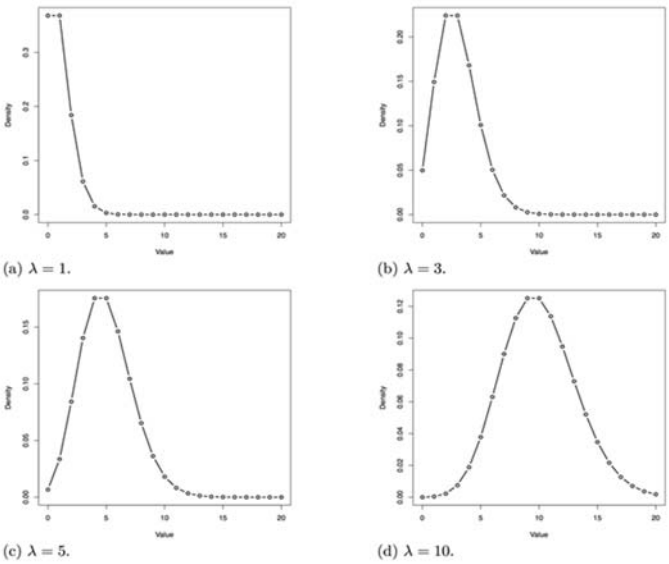


Figure 3. Plots of Poisson variable distributions with different values of λ.

integers. It is defined by a single parameter, λ, which estimates both the mean and variance of the distribution, thereby completely controlling the distribution's shape. When λ is close to zero the distribution is very positively skewed, but as λ increases the distribution becomes less skewed and appears closer to a normal distribution (see Figure 3).

The major differences between a Poisson regression and its typical regression counterpart are twofold. First, the Poisson regression model assumes the residuals follow a Poisson distribution rather than a normal distribution. Second, predictor variables are linked to the outcome via a natural log transformation (Cameron & Trivedi, 1998), similar to what is done with logistic regression (Hosmer & Lemeshow, 2000). The log transformation guarantees that the regression model's predicted values are never negative.

For a simple Poisson regression, the model is

$$\underbrace{\ln(\lambda_i) = \mu_i}_{\substack{\text{Predicted count} \\ \text{(Transformed by Link)}}} = \underbrace{a + bX_i}_{\substack{\text{Structural} \\ \text{(Original)}}} \quad (1)$$

where X is a predictor variable, *i* represents a group of observations with the same value on X, *a* and *b* are the intercept and slope, respectively, and μi is the expected value of the outcome variable for all respondents whose value on X is X_i. As the mean of a Poisson distribution is λ and the link function for a Poisson regression is the natural log, Equation (1) shows that the mean of the regression equation, μ_i, equals $\ln(\lambda_i)$.

To return the outcome variable to its original count scale requires transforming the structural part of Equation (1) by the inverse of the link function. The inverse of the natural log function is the exponent function, giving

$$\underbrace{\boldsymbol{\lambda_i = \exp(\mu_i)}}_{\substack{\textbf{Predicted count} \\ \textbf{(Original)}}} = \underbrace{\boldsymbol{\exp(a + bX_i)}}_{\substack{\textbf{Structural} \\ \textbf{(Transformed by Inverse Link)}}} \quad (2)$$

Negative Binomial Model

The Poisson distribution assumes that the mean and variance of the variable are equal. Sometimes count variables do not meet this assumption, especially when there are more zeros or more high values than expected. This is called overdispersion and results in a variable's variance (v) being much larger than its mean (λ). Overdispersion can be incorporated into the GLM regression by estimating the amount of extra variation. One way of doing this is by using a negative binomial (NB) distribution for the residuals. The NB distribution models variance as

$$v = \lambda + \frac{\lambda^2}{\theta} \quad (3)$$

where θ is an overdispersion parameter (Jay & Peter, 2007).

Zero-Inflated Models

When describing count variables, we stated that it is common for many of the respondents to have never have exhibited the behavior for outcomes that are particularly negative. The resulting variable's distribution has many zeros and just a few other values (Atkins, Baldwin, Zheng, Gallop, & Neighbors, 2013; Atkins & Gallop, 2007). For such cases, there is a class of regression models that can account for the excess zeros, called *zero-inflated models*.

Zero-inflated Poisson (ZIP) and zero-inflated negative binomial (ZINB) regression directly model the excessive number of zeros in the outcome variable. They do this by fitting a mixture model, which combines multiple distributions (Muthén & Shedden, 1999). For ZIP and ZINB models, the outcome variable's distribution is approximated by mixing two models and two distributions. This first model examines if the behavior ever occurred by using a logistic regression. Logistic regression is commonly used to predict a behavior's occurrence, but with ZIP/ZINB models the logistic regression part of the model predicts non-occurrence (i.e., it predicts the zeros). The second model examines how frequently the behavior occurred, using either a Poisson or NB regression. The resulting zero-inflated models produce two sets of coefficients, one predicting if the behavior never occurred (logistic) and the other predicting how frequently the behavior occurred (Poisson or NB). Because mixture models are flexible, the predictors for the two parts of the model can be different. Thus, ZIP and ZINB models are particularly well suited for variables thought to be determined by two different processes–one that influences occurrence and the other that influences the frequency of occurrence.

Parameter Estimation

Typical regression models often use ordinary least squares to estimate the parameters. For count data, the regression model uses maximum likelihood (ML) to estimate the parameters. ML seeks to find values for the regression coefficients that have the highest probability (i.e., maximum likelihood) to have produced the observed data. Enders (2005) provides a particularly readable introduction to ML estimation.

ML usually requires an iterative set of procedures to find the parameter estimates, which can be problematic with models that use many predictor variables or have small sample sizes. If the ML estimation procedure converges, it means it found a unique set of values for each parameter, the combination of which returned the highest likelihood value of all parameter values examined. Thus, it will return parameter estimates, standard errors, and the maximum likelihood value. This likelihood value, or transformations of it, is used to compare the fit of competing models. For more information about the parameter estimation process for count regression, see Cameron and Trivedi (1998).

Parameter Interpretation

In typical regression, the two major parameters to interpret are the intercept and slope/regression coefficients. The intercept is the expected value of the outcome variable when all the predictor variables have a value of zero. Each regression coefficient represents the expected change in the outcome variable for a one-unit change in the predictor variable, holding all the other predictor variables constant.

For Poisson and NB regressions, the two major parameters to interpret are still the intercept and slope/regression coefficients. The interpretation is trickier than with typical regression models because of the log link function, which places the regression coefficients on the natural log scale. Exponentiating the regression coefficients places the predicted values for the outcome on its original scale (cf. Equation 2), but this does not completely solve the interpretation problem. A result of using the log link function is that it forces the continuous predictor variables to have a non-linear relationship with the outcome. Specifically, the Poisson and NB models really specify multiplicative regression models instead of additive ones.

To aid in interpreting Poisson and NB models' coefficients, Atkins and Gallop (2007) recommend different strategies. One strategy is to use the regression equation to generate predictions over a specified range of values of the predictor variables. Specifically, they recommend setting the predictor variables at multiple values of interest and examine how the expected values of the outcome variable change. For the predictor variables of major interest, they suggest using values across the middle 95% of their values. For predictor variables that are not of major interest (i.e., control variables, covariates) either set their values to the mean (if continuous) or the reference value (if categorical).

A second interpretive strategy is to use the inherent multiplicative nature of the variable's relationships to examine the percentage change in the expected counts, defined as

$$\text{Percentage Change in the Expected Counts} = 100 \times [\exp(b \times \Delta)-1] \quad (4)$$

where b is the regression coefficient from the Poisson or NB regression, and Δ is the amount of change in the predictor (e.g., for one unit change, $\Delta = 1$).

Because ZIP and ZINB regressions model two separate processes, they produce two sets of coefficients: one for the count part of the model and the other for the logistic part of the model. The coefficients for the count part of the model can be interpreted the same as for a typical Poisson or NB model. The coefficients for the logistic part of zero-inflated models are on the logit scale

$$\text{logit} = \ln\left(\frac{\pi}{1-\pi}\right) \quad (5)$$

where π is the proportion of zeros. A common way of interpreting logistic regression models is to exponentiate the coefficients, which places the coefficients in an odds-ratio scale. An alternative approach is to use the inverse logit function to transform the resulting regression model, which places the outcome on the probability scale:

$$\text{inverse logit} = \frac{\exp\left(\frac{\pi}{1-\pi}\right)}{\exp\left(\frac{\pi}{1-\pi}\right)+1} \quad (6)$$

Model Comparison

An important aspect of all regression models is to determine how well the model fits the data, either by comparing the actual values with the model- predicted values or by comparing a model to competing models. We demonstrate both approaches in the following example.

In typical regression, R^2 is usually used as the measure of how close the actual values are to the predicted values. While pseudo-R^2 values for count regression models exist, they have the same issues as pseudo-R^2 for logistic regression, such as not really measuring variance and different formulae producing disparate values. Consequently, since there are a finite number of possible outcome values for count regression models, we examine model-data fit by examining the raw difference between the predicted counts and actual counts at each value of the outcome.

Table 1. Variables Used in Count Regression Models (n = 889)

Variable (Name in NELS Dataset)	Description	Values
Skips (*F1S10B*)	Number of times student cut/skipped class (Outcome variable)	0 = 0 times; 1 = 1-2 times; 2= 3-6 times; 3 = 7-9 times; 4= ≥ 10 times
College (*F1S51*)	Plan on going to college	0 = No; 1 = Yes
Male (*BYS12*)	Sex	0 = Female; 1 = Male
Race (*BYS31A*)	Self-described race	0 = White; 1 = Asian; 2 = Hispanic; 3 = Black; 4 = Native American
Achievement (*BYTEXCOMP*)	Standardized reading and math achievement test composite	Continuous
Self Concept (*BYCNCPT1*)	Positive self concept, which is a composite of four items	Continuous
SES (*BYSES*)	Socioeconomic status composite	Continuous

Note. Continuous variables were mean centered for all analyses.

When comparing competing models, information-criterion based fit indices are useful (Burnham & Anderson, 2002). The basic principle of such fit measures is to select the simplest models that can describe the data well (Sherman & Funder, 2009). A commonly used measure from the information-theoretic tradition is Akaike's information criterion (AIC). AIC balances the model's goodness-of-fit to the data and a penalty for model complexity. The general method for using the AIC is to choose the model that has the smallest AIC value. Individual AIC values are not directly interpretable because they contain arbitrary constants and are greatly affected by sample size. These artificial increases in AIC values can sometimes make it appear that multiple models ostensibly appear to have very similar AIC values, even though some models fit the data substantially better than others. The AIC values for a set of models can be transformed so that they sum to the value one, so can be treated like probabilities. These values are called Akaike weights and are typically interpreted as the probability that model a given model is the best model for the data out of all the compared models.

Another model-fit measure that penalizes models for complexity is Schwartz's Bayesian information criterion (BIC). While it is not technically related to information theory, it can still be useful in model selection. The general method for using the BIC is to choose the model that has the smallest BIC value. With small sample sizes the BIC tends to be overly-conservative (i.e., prefer models with too few variables), but when the sample size is large it tends to select the correct model if a set of competing models includes the true model.

Model Diagnostics. An important part of all regression analyses is to examine residual diagnostics, influential data points, and nonlinearity in the predictors (Andersen, 2012; Belsley, Kuh, & Welsch, 2005). Many of the common tools to assess typical regression models have been extended to count regression, including standardized residuals, influence diagnostics (e.g., Cook's *D*), and predictor nonlinearity.

In our example, we graphically examined the residuals' distribution and their relation to the predicted values. We used deviance residuals for all the models except zero-inflated, where we used Pearson residuals.

To examine influential observations, we calculated Cook's (1977) *D* values for each case based on each model. Cook's *D* is an index that reflects the amount of influence each case has on the model parameter estimates. A common criterion used for identifying cases that could be influential is whether an observations's *D* value is greater than $4/n$ (Cohen et al., 2003). In addition, to assist with the identification of influential cases we plotted the *D* values for each observation and inserted a horizontal line at $y = 4/n$.

To examine linearity, we created scatterplots of each predictor variable and the residuals. We looked for a similar distribution of residuals at each level of the predictor variables. Since this requires multiple plots for each model, we only show the results (and **R** syntax) for the negative binomial model.

Count Regression Example

Data

Data for this example were taken from a subset of the National Education Longitudinal Study of 1988 (NELS), provided by Keith (2006)[1]. The variables used for this analysis are given in Table 1. We only used the observations with values for each of the variables.

The outcome is the number of times a student cut/skipped class (skips), placed into one of five categories. A histogram of the skips variable is shown in Figure 7a, and indicates that the variable is not symmetrical, so cannot follow a normal distribution. Thus, because it is a count variable that is distinctly not normally distributed, it is a prime candidate for a count regression model.

For this particular example, we were interested in predicting the number of skips by race, sex, positive self-concept, academic achievement, socioeconomic status (SES), and whether the student plans on going to college. We chose these variables as they represent a mixture of continuous and categorical predictor variables. To make interpretation easier, we mean centered the continuous variables (SES, self-concept, and academic achievement) and dummy-coded the categorical variables.

Typical Regression

As a baseline, we fit a typical regression model to the data, i.e., a model that assumes the residuals follow a normal distribution. Often, these regression parameters are estimated through ordinary least squares (OLS). With normally-distributed residuals, OLS and maximum likelihood (ML) parameter estimates are the same (Kutner, Neter, Nachtsheim, & Li, 2004). For consistency with the other models we fit, we used ML estimation for this model.

The results are shown in the second column of Table 2. The intercept is 0.90. Be-cause of our predictor variable coding mechanisms and centering, the intercept in the model is interpreted as the predicted number of skips for a white female who does not plan to go to college and who had average levels of self-concept, academic achievement, and SES.

The regression coefficients are interpreted as any other unstandardized coefficients from a typical regression. For example, the coefficient associated with going to college is −0.32, indicating that the average number of skips for those who plan on attending college is lower than the average for students not wanting to attend college, after controlling for all the other predictor variables. As another example, the regression coefficient associated with academic achievement is −0.01, meaning that for each one-unit increase in academic achievement, the average skip category decreases by 0.01 units.

The typical regression model assumes that the residuals follow a normal distribution. A plot of the residuals for typical regression model is shown in Figure 4 and clearly shows they do not follow a normal distribution. Another plot that is useful to examine is to compare the residuals to the predicted values. There should be no relationship between these two values, so the LOWESS line should be horizontal and close to zero (for more about LOWESS lines, see Trexler & Travis, 1993). Figure 5 shows plots of the residuals vs. the predicted values. The typical regression shows a

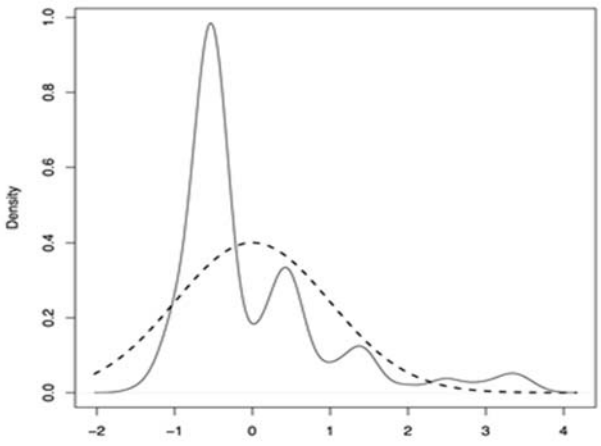


Figure 4. Residuals for typical regression model. A normal distribution is overlaid (dashed line) with the same mean and standard deviation of the residuals

[1] Data can be downloaded from https://baylor.box.com/countdata. The datafile's name is *count.dat*. It is a comma-delimited file with a period (.) and 999.98 used as missing value indicators.

Table 2. Comparison of the Regression Models for the School Skip Data (n = 889).

				ZIP		ZINB	
Variable	Typical	Poisson	NB	Count	Logistic	Count	Logistic
				Coefficients			
Intercept	0.90	-0.19	-0.20	0.54	0.08	0.39	-0.21
Male	-0.03	-0.04	-0.03	-0.30	-0.65	-0.32	-1.02
Asian[a]	0.05	0.08	0.05	0.42	0.74	0.43	1.00
Hispanic[a]	0.36	0.42	0.41	-0.07	-2.16	0.02	-15.36
Black[a]	-0.06	-0.08	-0.10	-0.28	-0.54	-0.24	-0.65
Native American[a]	0.03	0.06	0.04	0.08	0.04	0.08	0.08
College[b]	-0.32	-0.37	-0.37	-0.38	0.01	-0.40	-0.05
Self Concept	-0.03	-0.05	-0.05	0.06	0.29	0.05	0.38
SES	-0.07	-0.11	-0.13	-0.11	-0.01	-0.07	0.13
Achievement	-0.01	-0.01	-0.02	-0.01	0.01	-0.01	0.00
				Standard Errors			
Intercept	0.11	0.12	0.16	0.15	0.35	0.20	0.52
Male	0.07	0.09	0.11	0.12	0.30	0.14	0.52
Asian[a]	0.14	0.19	0.23	0.25	0.43	0.30	0.59
Hispanic[a]	0.11	0.12	0.16	0.16	1.35	0.18	939.35
Black[a]	0.12	0.15	0.19	0.23	0.64	0.25	0.98
Native American[a]	0.17	0.20	0.26	0.27	0.55	0.31	0.74
College[b]	0.11	0.11	0.15	0.14	0.36	0.16	0.49
Self Concept	0.05	0.06	0.07	0.07	0.18	0.08	0.25
SES	0.05	0.07	0.08	0.10	0.23	0.10	0.29
Achievement	0.00	0.01	0.01	0.01	0.02	0.01	0.03
				Likelihood			
Log Likelihood	-1258	-1000	-958	-951		-948	
Model *df*	11	10	11	20		21	
				Fit Measures			
AIC	2537	2021	1939	1942		1938	
AIC Weight	0.00	0.00	0.40	0.06		0.54	
BIC	2590	2069	1992	2038		2039	

Note. Typical: Model assuming normally-distributed residuals, fitted with maximum likelihood estimation. NB: Negative binomial; ZI: Zero-inflated. AIC: Akaike information criterion; BIC: Bayesian information criterion.

[a.] Reference category is White.

[b.] Reference category is not planning on going to college.

slight pattern in the results as the LOWESS line is slated downward. The other models all have horizontal LOWESS lines, with the negative-binomial mode lhaving the lowest range of residual values.

Plots of the Cook's *D* values for each observation are shown in Figure 6. Overall, the *NB* model resulted in fewest influential cases (i.e., cases with $D \leq 4/n$), indicating that each observation contributed equitably to the parameter estimates. As measures of model fit, AIC and BIC values are shown in the bottom of Table 2. Figure 7a graphically shows how well the model predicts the count values by overlaying the predicted probabilities for each skip category on the frequency histogram of the actual skip data. It appears that the typical regression model under-predicts the 0 and 4 skip categories, but over-predicts all the other categories. Figure 7b shows the actual and predicted values numerically, and echoes the over- and under-predictions shown in Figure 7a. Thus, it appears that both model fit and model diagnostics converge in indicating that the typical regression model does not account for the count nature of the skip data very well.

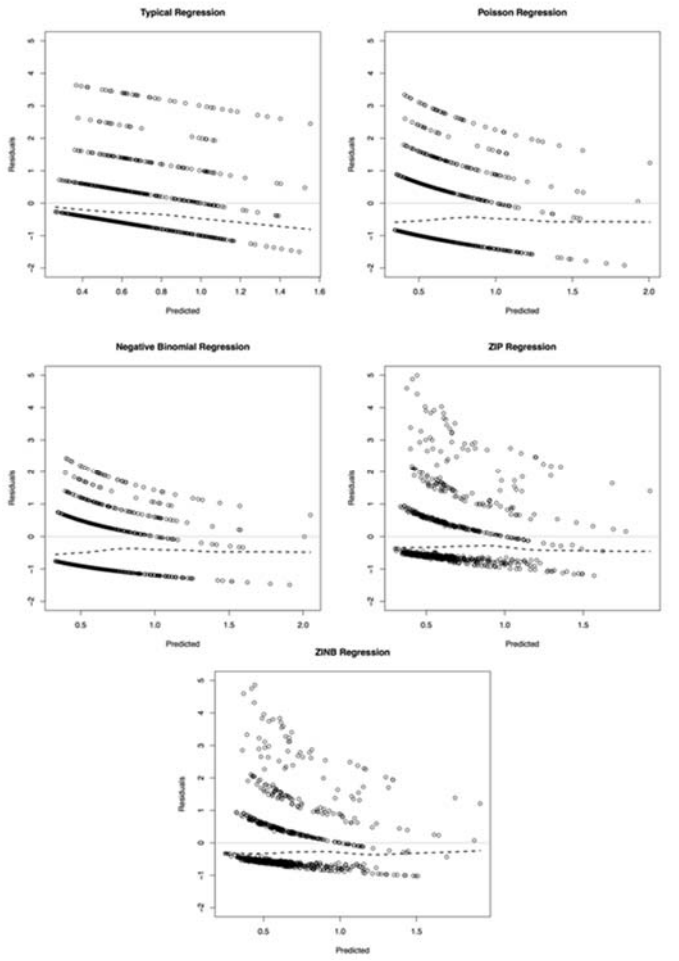


Figure 5. Predicted values vs. residual plots. LOWESS lines are dashed

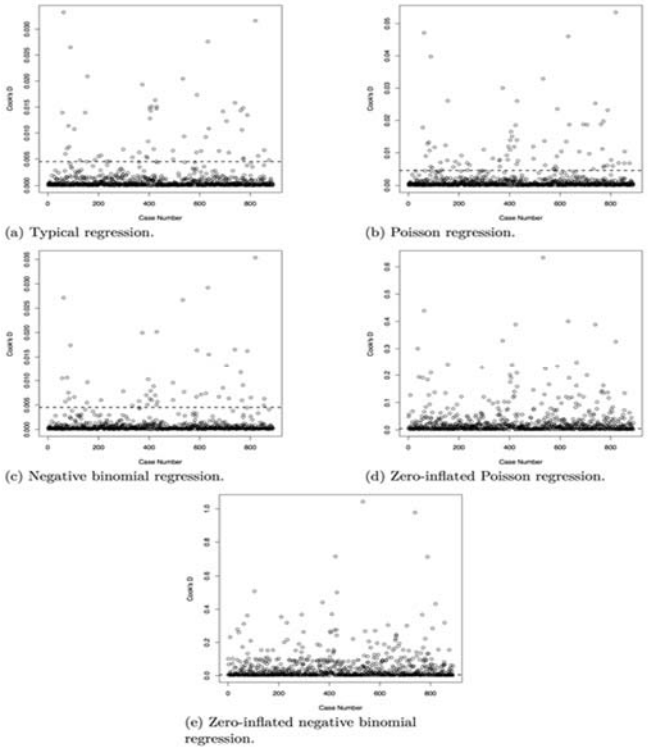
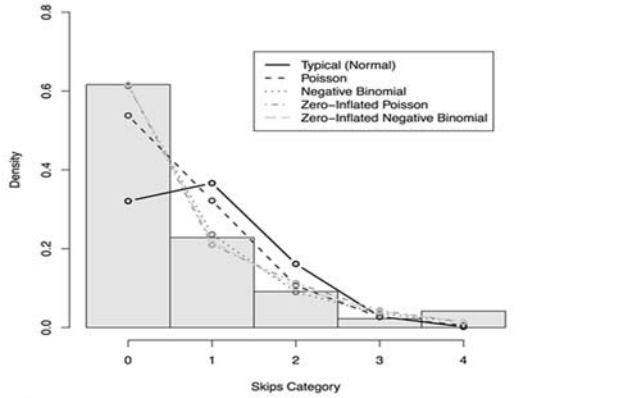


Figure 6. Plots of Cook's D values for each regression model. Threshold is a dashed line



(a) Histogram of school skip categories with overlaid predicted probabilities from each regression model.

			Predicted				
Outcome	Category	Actual	Typical	Poisson	NB	ZIP	ZINB
0 Skips	0	548	285	478	545	547	549
1-2 Skips	1	203	325	286	210	187	194
3-6 Skips	2	81	144	95	80	100	90
7-9 Skips	3	20	25	24	31	39	36
≥ 10 Skips	4	37	2	5	13	12	13

Note. NB: Negative binomial. ZIP: Zero-inflated Poisson. ZINB: Zero-inflated negative binomial.

(b) Actual and predicted category counts for each model, rounded to the nearest integer.

Figure 7. Comparison of actual and predicted category counts

Poisson Regression

We fit the Poisson regression model using the same predictors of skips we used with the typical regression model. The results from the Poisson regression are shown in the third column of Table 2.

As with the typical regression, the intercept represents the predicted number of skips for a white female who does not plan to go to college and who had average levels of self-concept, academic achievement, and SES. The intercept is −0.19, but the log link makes this value hard to interpret. This can be remedied by exponentiating the value. For this example, exp(−0.19) = 0.82, indicating that the average skip category for a white female who does not plan to go to college and who had average levels of self-concept, academic achievement, and SES is 0.82. This is between the 0 and 1-2 skips categories, closer to the latter than the former.

The coefficient associated with wanting to go to college is −0.37. As wanting to go to college is dummy-coded, the negative sign indicates that the average number of skips for those who want to go to college is smaller than for those who do not want to go to college. Since exp(−0.37) = 0.69, the difference between the two groups is over two-thirds of a skip category.

The method of interpreting dummy-coded categorical variables does not directly extend to continuous predictors. Previously, we discussed Atkins and Gallop's (2007) recommendations for interpreting these variables. As the first is a common way to help interpret any regression model (Cohen et al., 2003), we only focus on the second: percentage change in the expected counts.

The percentage change in the expected counts method of interpretation requires two values: the regression coefficient and the desired amount of change in the variable. For the academic achievement variable, the regression coefficient is −0.01. For the amount of change we use one SD, which is 9.98. Plugging those values into Equation (4) produces

$$100 \times [\exp(-0.01 \times 9.98) - 1] = -13.4$$

meaning there is a 13.4% decrease in the expected skip category value for a one SD increase in academic achievement.

To examine model fit, we first compared the AIC and BIC values for the Poisson model to those from the typical regression model (see Table 2). As the values are much smaller for the Poisson regression, this indicates the Poisson model provides an improvement over the typical regression model. Second, we compared the predicted skip category values against the actual values in Figure 7a and Figure 7b. The Poisson model appears to do a much better job capturing the skip data than the typical regression model across all skip value categories.

Negative Binomial Regression

The results from the negative binomial (NB) regression are shown in the fourth column of Table 2. The NB model is very similar to the Poisson model, thus the NB model's coefficients are interpreted in the same way as the Poisson regression. The main difference between the NB and Poisson models is that the NB model allows for more variability (dispersion) in the outcome by not assuming the mean and variance of the residuals are the same. Consequently, the NB model estimates one extra parameter than the Poisson model: a overdispersion parameter (see Equation 3). The value for overdispersion parameter for the skip data is 1.11. Since the NB and Poisson models are so similar, it is not surprising that the regression coefficients for the two models are very close. The standard errors, however, are larger for the NB model reflecting its larger residual variance.

The plots of the predictor variables against the standardized residuals are shown in Figure 8. Based on visual inspection, we determined that the residual distributions were approximately the same across levels of the predictor variables. We noticed that there were fewer observations in the lower tail of the self-concept distribution, which produces a slightly dissimilar pattern of residuals for that predictor variable. On the whole, the residual patterns across all predictor variables from the NB model were acceptable.

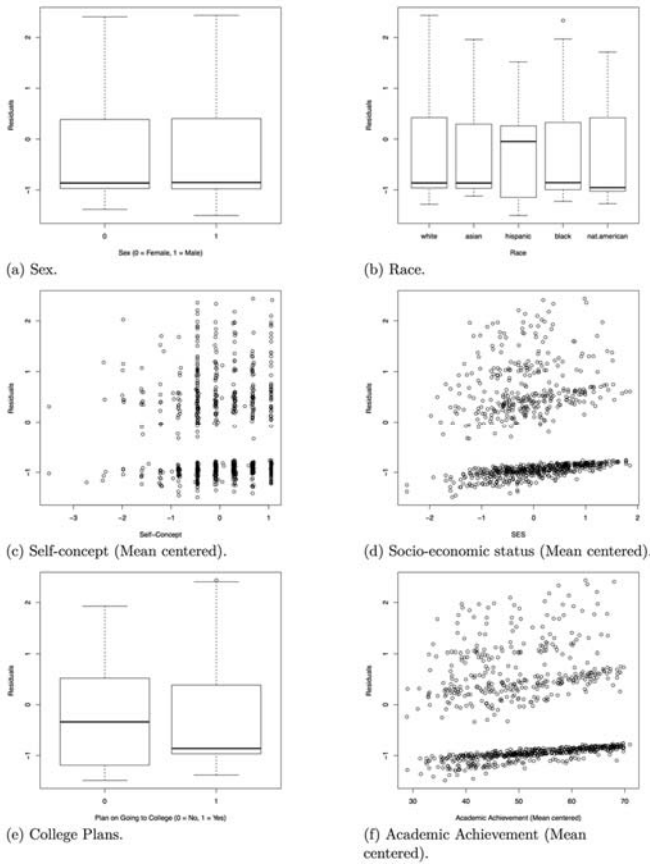


Figure 8. Plots of negative binomial model predictors by residuals.

The AIC and BIC values for NB model are smaller than those for the Poisson model, indicating that the NB model fits the data somewhat better than the Poisson model. Although the amount is not that large, there appears to be enough overdispersion in the skip variable that the Poisson model was not able to capture the variance as well as the NB model. This result is echoed in Figure 7a and Figure 7b, which shows that the NB model provides more accurate predictions than the Poisson regression model for all skip categories except category three (7-9 skips).

Table 3. Results for Final Count Regression Model (Negative Binomial)

				95% Confidence Interval for exp(B)	
Variable	*B*	SE	exp(*B*)	Lower Bound	Upper Bound
Intercept	−0.16	0.14	0.85	0.65	1.13
SES	−0.18	0.08	0.84	0.71	0.98
Achievement	−0.02	0.01	0.99	0.97	1.00
College[a]	−0.37	0.15	0.69	0.51	0.93

Note. B: Unstandardized coefficient; SE: Standard error; exp(B): Exponentiated regression coefficient. Log Likelihood: −963 (df = 5); AIC: 1935; BIC: 1959.
[a]. Reference category is not planning on going to college.

Zero-Inflated Regression

T The results from the zero-inflated models are shown in Table 2. Each zero-inflated model has two sets of regression coefficients. Thus, the zero-inflated Poisson (ZIP) regression values are shown in columns five and six, while the values from the zero-inflated negative binomial (ZINB) regression models are shown in columns seven and eight.

For each zero-inflated model's set of coefficients in Table 2, the first column shows the count regression part, while the second column is the logistic regression portion of the model. The coefficients for the count part of the model can be interpreted the same as was done previously for the Poisson model. The coefficients for the logistic regression are on the logit scale, so exponentiating them transforms the values to odds ratios. Remember, with zero-inflated models the logistic part of the model predicts non-occurrence of the outcome.

As an example interpretation, in the ZINB model the coefficients for planning on going to college are -0.40 and −0.05 for the count and logistic parts of the model, respectively. Exponentiating the count coefficient gives 0.67. Consequently, holding all the other variables constant, students planning on going to college typically skip fewer classes than those not planning on going to college, and the difference is approximately the size of two-thirds of a skip category.

Exponentiating the logistic coefficient gives 0.95, indicting that after holding all the other variables constant, students planning on going to college have slightly decreased odds of never having skipped a class than students not planning on going to college. Alternatively, by using Equation (6) the results indicate that the probability of not having skipped a class for students planning on going to college is 0.01 units lower than for students not planning on going to college. The interpretation of the logistic part of the model must be tempered, however, as an odds ratio of 0.95 represent a small effect and the 95% confidence interval (0.76 - 1.13) contains 1.0 (i.e., no difference). Thus, it is likely that planning on going to college only has an influence on the number of skips, not whether a students has ever skipped a class.

Final M When examining fit across all five of the models, the AIC values and AIC weights favor the ZINB and NB over the other three models, while the BIC favors the NB model. As the ZINB model requires twice as many parameters as the NB model, the NB model is the more parsimonious model. Thus, it appears that accounting for the overdispersion is sufficient to capture the excess number of zero values.

Support for the NB model over the ZINB model is bolstered by Figure 7a and Figure 7b. The NB model does as good of a job as the ZINB model in capturing the first two and last two skip categories, and does a slightly better for the third category (3-6 skips).

All the models fit thus far assume that all the predictor variables are needed. An examination of the values in Table 2 indicates this is likely not the case. Specifically, it appears that only the academic achievement and planning on going to college variables might be needed. Consequently, we pruned the model removing each variable singly and in sets.

The BIC indicted that the model with only the achievement and college plans variables fit the best, while the AIC indicated that achievement, college plans, and SES should be kept. We opted to keep achievement, college plans, and SES in the model. The results for the final, pruned NB binomial model are shown in Table 3.

Discussion

Count outcome variables are very common in many areas of education research. Older traditions of dichotomizing or transforming the outcome variable can produce estimation and interpretive problems. This tutorial introduced methods to analyze count data using the general linear model framework, which is a robust way to handle count outcomes in regression. We discussed four ways to model count data in regression, and then demonstrated the analysis of the data. **R** syntax for all the analyses is given in the Appendix.

Extensions of Count Regression Models

We ignored two related areas that are growing in this field. The first is including count data in a multilevel framework. Often, data that interests education researchers is multilevel in nature, as when using data from students coming from the same classroom or school (Graves & Frohwerk, 2009). Analyzing such nested data can be tricky when the variables are continuous, much less when they are counts. Nonetheless, recent work in this are has shown how to include count outcomes when the data is nested (Atkins et al., 2013; Gelman & Hill, 2006).

The second is including count data in a repeated measures framework. While this situation could be considered a type of nested data, it is more frequently used with single case designs. We stated in the beginning of this article that Shadish and Sullivan's (2011) single-case design (SCD) review showed that nearly all outcome variables used with SCDs are count variables. As with multilevel data, extending count regression models to SCDs is a difficult endeavor. Nonetheless, there are some promising signs that this, too, can be included in the educational researcher's data analysis tools (e.g. Shadish, Zuur, & Sullivan, 2014).

References

Aiken, L. S., West, S. G., & Millsap, R. E. (2008). Doctoral training in statistics, measurement, and methodology in psychology: Replication and extension of Aiken, West, Sechrest, and Reno's (1990) survey of PhD programs in North America. *American Psychologist*, 63, 32-50. doi: 10.1037/0003-066X.63.1.32

Andersen, R. (2012). Methods for detecting badly behaved data: Distributions, linear models, and beyond. In H. Cooper, P. M. Camic, D. L. Long, A. T. Panter, D. Rindskopf, & K. J. Sher (Eds.), APA handbook of research methods in psychology, Vol 3: Data analysis and research publication (pp. 5-26). Washington, DC: American Psychological Association.

Atkins, D. C., Baldwin, S., Zheng, C., Gallop, R. J., & Neighbors, C. (2013). A tutorial on count regression and zero-altered count models for longitudinal addictions data. *Psychology of Addictive Behaviors*, 27, 166-177. doi: 10.1037/a0029508

Atkins, D. C., & Gallop, R. J. (2007). Rethinking how family researchers model infrequent outcomes: A tutorial on count regression and zero-inflated models. *Journal of Family Psychology*, 21, 726-735. doi: 10.1037/0893-3200.21.4.726

Belsley, D. A., Kuh, E., & Welsch, R. E. (2005). Regression diagnostics: Identifying influential data and sources of collinearity. New York, NY: John Wiley & Sons.

Box, G. E. P. (1953). Non-normality and tests on variances. Biometrika, 40, 318-335. doi: 10.2307/2333350

Burnham, K. P., & Anderson, D. (2002). Model selection and multimodel inference: A practical information-theoretic approach (2nd ed.). New York: Springer-Verlag.

Cameron, A. C., & Trivedi, P. K. (1998). Regression analysis of count data. New York, NY: Cambridge University Press.

Cochran W. G. (1947). Some consequences when the assumptions for the analysis of variance are not satisfied. *Biometrics*, 3, 22-38. doi: 10.2307/3001535

Cohen, J., Cohen, P., West, S. G., & Aiken, L. S. (2003). Applied multiple regression/correlation analysis for the behavioral sciences (3rd ed.). Mahwah, NJ: Lawrence Erlbaum.

Cook, R. D. (1977). Detection of influential observation in linear regression. *Technometrics*, 19, 15-18.

Coxe, S., West, S. G., & Aiken, L. S. (2009). The analysis of count data: A gentle introduction to poisson regression and its alternatives. *Journal of Personality Assessment*, 91, 121-136. doi: 10.1080/00223890802634175

Enders, C. K. (2005). Maximum likelihood estimation. In B. Everitt & D. C. Howell (Eds.), Encyclopedia of behavioral statistics (pp. 1164-1170). West Sussex, England: Wiley.

Gelman, A., & Hill, J. (2006). Data analysis using regression and multilevel/hierarchical models. New York, NY: Cambridge.

Graves, J., Scott L., & Frohwerk, A. (2009). Multilevel modeling and school psychology: A review and practical example. *School Psychology Quarterly*, 24, 84-94. doi: 10.1037/a0016160

Hilbe, J. M. (2014). Modeling count data. New York, NY: Cambridge University Press.

Hosmer, D. W., & Lemeshow, S. (2000). Applied logistic regression (2nd ed.). Hoboke, NJ: Wiley.

Jay, M. V. H., & Peter, L. B. (2007). Quasi-Poisson vs. neagtive binomial regression: How should we model overdispersed count data? *Ecology*, 88, 2766-2772.

Keith, T. Z. (2006). Multiple regression and beyond. Boston: Pearson.

Kutner, M. H., Neter, J., Nachtsheim, C. J., & Li, W. (2004). Applied linear regression models (4th ed.). New York, NY: McGraw-Hill.

Little, S. G., Akin-Little, A., & Lee, H. B. (2003). Education in statistics and research design in school psychology. *School Psychology International*, 24, 437-448. doi: 10.1177/01430343030244006

Lix, L. M., Keselman, J. C., & Keselman, H. J. (1996). Consequences of assumption violation revisited: A quantitative review of alternatives to the one-way analysis of variance F test. *Review of Educational Research*, 66, 579-619.

MacCallum, R. C., Zhang, S., Preacher, K. J., & Rucker, D. D. (2002). On the practice of dichotomization of quantitative variables. *Psychological Methods*, 7, 19-40. doi: 10.1037/1082-989x.7.1.19

McCullagh, P., & Nelder, J. A. (1989). Generalized linear models (2nd ed.). London, England: Chapman and Hall.

Muthén, B. O., & Shedden, K. (1999). Finite mixture modeling with mixture outcomes using the EM algorithm. *Biometrics*, 55, 463-469. doi: 10.1111/j.0006-341X.1999.00463.x

Perham, H. (2010). Quantitative training of doctoral school psychologists: Statistics, measurement, and methodology curriculum. Unpublished master's thesis, Arizona State University, Tempe, AZ.

R Development Core Team. (2015). R: A language and environment for statistical computing [Computer software]. Vienna, Austria: R Foundation for Statistical Computing.

Sakia, R. M. (1992). The Box-Cox transformation technique: A review. Journal of the Royal Statistical Society. *Series D (The Statistician)*, 41, 169-178. doi: 10.2307/2348250

Shadish, W. R., & Sullivan, K. J. (2011). Characteristics of single-case designs used to assess intervention effects in 2008. *Behavior Research Methods*, 43, 971-980. doi: 10.3758/s13428-011-0111-y

Shadish, W. R., Zuur, A. F., & Sullivan, K. J. (2014). Using generalized additive (mixed) models to analyze single case designs. *Journal of School Psychology*, 52, 149-178. doi: 10.1016/j.jsp.2013.11.004

Sherman, R. A., & Funder, D. C. (2009). Evaluating correlations in studies of personality and behavior: Beyond the number of significant findings to be expected by chance. *Journal of Research in Personality*, 43, 1053-1063. doi: 10.1016/j.jrp.2009.05.010

Trexler, J. C., & Travis, J. (1993). Nontraditional regression analyses. *Ecology*, 74, 1629-1637. doi: 10.2307/1939921

Appendix

R Syntax

Import Data

```
# import data
nels.data <- read.table("count.dat", sep=",", na.strings = c(".","999.98"), header=TRUE)
# subset variables of interest from NELS data
```

```
nels.var <- c("F1S10B","BYS12","BYS31A", "BYCNCPT1", "BYSES", "BYTXCOMP", "F1S51")
# create new dataset with only variables of interest
count.data <- nels.data[nels.var]
# change names of variables
names(count.data) <- c("skipped", "sex", "race", "self.con1", "ses", "achievement",
      "F1S51")
# change sex so that female = 0
count.data$male <- ifelse(count.data$sex==2, 0,count.data$sex)
# recode college plans variable to 0 = no, 1= yes,
count.data$college <- count.data$F1S51
count.data$college[count.data$F1S51==1] <- 0
count.data$college[count.data$F1S51==2 | count.data$F1S51==3 | count.data$F1S51==4 |
      count.data$F1S51==5] <- 1
# make race variable a factor and name the levels
count.data$race <- factor(count.data$race, levels=1:5, labels=c("asian", "hispanic",
      "black", "white", "nat.american"))
# make white the comparison race group
count.data$race <- relevel(count.data$race, ref = 4)
# create new dataset without missing data
count.data <- na.omit(count.data)
# mean center continuous variables
count.data$self.con1.m <- scale(count.data$self.con1, center = TRUE, scale = FALSE)
count.data$ses.m <- scale(count.data$ses, center = TRUE, scale = FALSE)
count.data$achievement.m <- scale(count.data$achievement, center = TRUE, scale = FALSE)
```

Fit Regression Models

```
# normal theory regression using maximum likelihood
skip.normal <- glm(skipped ~ male + race + college + self.con1.m + ses.m + achievement.m,
      data = count.data, family = gaussian)
# summary of results
summary(skip.normal)
# poisson regression
skip.pois <- glm(skipped ~ male + race  + college + self.con1.m + ses.m + achievement.m,
      data = count.data, family = poisson)
# summary of results
summary(skip.pois)

# load MASS package
library(MASS)
# negative binomial regression
skip.nb <- glm.nb(skipped ~ male + race + college + self.con1.m + ses.m + achievement.m,
      data = count.data)
# summary of results
summary(skip.nb)
# overdispersion
summary(skip.nb)$theta

# load pscl package
library(pscl)
# zero-inflated Poisson regression
# the | seperates the count model from the logistic model
skip.zip <- zeroinfl(skipped ~ male + race + college + self.con1.m + ses.m +
      achievement.m | male + race + college + self.con1.m + ses.m + achievement.m, data =
      count.data, link = "logit", dist = "poisson", trace = TRUE)
# summary of results
summary(skip.zip)

# load pscl package
library(pscl)
```

```
# zero-inflated negative binomial regression
# the | seperates the count model from the logistic model
skip.zinb <- zeroinfl(skipped ~ male + race + college + self.con1.m + ses.m +
      achievement.m | male + race + college + self.con1.m + ses.m + achievement.m, data =
      count.data, link = "logit", dist = "negbin", trace = TRUE, EM = FALSE)
# summary of results
summary(skip.zinb)
```

Model Fit

```
# AIC values
AIC(skip.normal)
AIC(skip.pois)
AIC(skip.nb)
AIC(skip.zip)
AIC(skip.zinb)


# AIC weights
compare.models <- list( )
compare.models[[1]] <- skip.normal
compare.models[[2]] <- skip.pois
compare.models[[3]] <- skip.nb
compare.models[[4]] <- skip.zip
compare.models[[5]] <- skip.zinb
compare.names <- c("Typical", "Poisson", "NB", "ZIP", "ZINB")
names(compare.models) <- compare.names
compare.results <- data.frame(models = compare.names)
compare.results$aic.val <- unlist(lapply(compare.models, AIC))
compare.results$aic.delta <- compare.results$aic.val-min(compare.results$aic.val)
compare.results$aic.likelihood <-  exp(-0.5* compare.results$aic.delta)
compare.results$aic.weight <-
      compare.results$aic.likelihood/sum(compare.results$aic.likelihood)
# BIC values
AIC(skip.normal, k = log(nrow(count.data)))
AIC(skip.pois, k = log(nrow(count.data)))
AIC(skip.nb, k = log(nrow(count.data)))
AIC(skip.zip, k = log(nrow(count.data)))
AIC(skip.zinb, k = log(nrow(count.data)))

# observed zero counts
# actual
sum(count.data$skip < 1)
# typical
sum(dnorm(0, fitted(skip.normal)))
# poisson
sum(dpois(0, fitted(skip.pois)))
# nb
sum(dnbinom(0, mu = fitted(skip.nb), size = skip.nb$theta))
# zip
sum(predict(skip.zip, type = "prob")[,1])
# zinb
sum(predict(skip.zinb, type = "prob")[,1])

Diagnostics
# normal residuals density plot
plot(density(residuals(skip.normal)))
# histogram plot with fitted probabilities
# predicted values for typical regression
```

```
normal.y.hat <- predict(skip.normal, type = "response")
normal.y <- skip.normal$y
normal.yUnique <- 0:max(normal.y)
normal.nUnique <- length(normal.yUnique)
phat.normal <- matrix(NA, length(normal.y.hat), normal.nUnique)
dimnames(phat.normal) <- list(NULL, normal.yUnique)
for (i in 1:normal.nUnique) {
      phat.normal[, i] <- dnorm(mean = normal.y.hat, sd = sd(residuals(skip.normal)),x =
      normal.yUnique[i])
}
# mean of the normal predicted probabilities for each value of the outcome
phat.normal.mn <- apply(phat.normal, 2, mean)
# probability of observing each value and mean predicted probabilities for
#count regression models
phat.pois <- predprob(skip.pois)
phat.pois.mn <- apply(phat.pois, 2, mean)
phat.nb <- predprob(skip.nb)
phat.nb.mn <- apply(phat.nb, 2, mean)
phat.zip <- predprob(skip.zip)
phat.zip.mn <- apply(phat.zip, 2, mean)
phat.zinb <- predprob(skip.zinb)
phat.zinb.mn <- apply(phat.zinb, 2, mean)
# histogram
hist(count.data$skip, prob = TRUE, col = "gray90", breaks=seq(min(count.data$skip)-0.5,
      max(count.data$skip)+.5, 1), xlab = "Skips Category", ylim=c(0,.8))
# overlay predicted values
lines(x = seq(0, 4, 1), y = phat.normal.mn, type = "b", lwd=2, lty=1, col="black")
lines(x = seq(0, 4, 1), y = phat.pois.mn, type = "b", lwd=2, lty=2, col="gray20")
lines(x = seq(0, 4, 1), y = phat.nb.mn, type = "b", lwd=2, lty=3, col="gray40")
lines(x = seq(0, 4, 1), y = phat.zip.mn, type = "b", lwd=2, lty=4, col="gray60")
lines(x = seq(0, 4, 1), y = phat.zinb.mn, type = "b", lwd=2, lty=5, col="gray80")
# legend
legend(1, 0.7, c("Typical (Normal)","Poisson", "Negative Binomial", "Zero-Inflated
      Poisson", "Zero-Inflated Negative Binomial"), lty=seq(1:5), col =
      c("black","gray20","gray40","gray60","gray80"), lwd=2)
# predicted vs. residual plots
# typical
plot(predict(skip.normal, type="response"), residuals(skip.normal), main="Typical
      Regression", ylab="Residuals", xlab="Predicted", ylim=c(-2,5))
abline(h=0,lty=1,col="gray")
lines(lowess(predict(skip.normal,type="response"),residuals(skip.normal)), lwd=2, lty=2)
# poisson
plot(predict(skip.pois,type="response"),residuals(skip.pois), main="Poisson Regression",
      ylab="Residuals", xlab="Predicted", ylim=c(-2,5))
abline(h=0,lty=1,col="gray")
lines(lowess(predict(skip.pois,type="response"),residuals(skip.pois)),lwd=2, lty=2)
# negative binomial
plot(predict(skip.nb,type="response"),residuals(skip.nb), main="Negative Binomial
      Regression", ylab="Residuals", xlab="Predicted", ylim=c(-2,5))
abline(h=0,lty=1,col="gray")
lines(lowess(predict(skip.nb,type="response"),residuals(skip.nb)), lwd=2, lty=2)
# ZIP
plot(predict(skip.zip,type="response"),residuals(skip.zip), main="ZIP Regression",
      ylab="Residuals", xlab="Predicted", ylim=c(-2,5))
abline(h=0,lty=1,col="gray")
lines(lowess(predict(skip.zip,type="response"),residuals(skip.zip)),lwd=2, lty=2)
# ZINB
plot(predict(skip.zinb,type="response"),residuals(skip.zinb), main="ZINB Regression",
      ylab="Residuals", xlab="Predicted", ylim=c(-2,5))
abline(h=0,lty=1,col="gray")
```

```
lines(lowess(predict(skip.zinb,type="response"),residuals(skip.zinb)),lwd=2, lty=2)
```

A Cook's *D* computation function is built into **R** for the typical, Poisson, and negative binomial regression models, but not zero-inflated models. Consequently, we wrote an iterative function to compute the D values for each case in the zero-inflated models.

```
# plot Cook's D for the typical regression
plot(cooks.distance(skip.normal), main="Cook's D Estimates", ylab="Cook's D",
      xlab="Observation")
abline(h=(4/nrow(count.data)), col="red", lwd=2)

# plot Cook's D for the Poisson model
plot(cooks.distance(skip.pois), main="Cook's D Estimates", ylab="Cook's D",
      xlab="Observation")

# plot Cook's D for the negative binomial model
plot(cooks.distance(skip.nb), main="Cook's D Estimates", ylab="Cook's D",
      xlab="Observation")
abline(h=(4/nrow(count.data)), col="red", lwd=2)

# compute generalized Cook's D for zero-inflated models
g.cooks.zi<-function(model){
      n <- nrow(count.data)
      cooks <- as.matrix(rep(0,nrow(count.data)))
      for (i in 1:n){
            if(model=="ZIP"){
            skip.zip.red <- zeroinfl(skipped ~ male + race + self.con1.m + ses.m +
      achievement.m + college | male + race + self.con1.m + ses.m + achievement.m +
      college, data = count.data[-i,],
      link = "logit", dist = "poisson", trace = TRUE)
      cooks[i]<-t(rbind(as.matrix(skip.zip.red$coefficients$count),
      as.matrix(skip.zip.red$coefficients$zero))-
      rbind(as.matrix(skip.zip$coefficients$count),
      as.matrix(skip.zip$coefficients$zero)))%*%
      (-(skip.zip$optim$hessian))%*%(rbind(
      as.matrix(skip.zip.red$coefficients$count),
      as.matrix(skip.zip.red$coefficients$zero))-
      rbind(as.matrix(skip.zip$coefficients$count),
      as.matrix(skip.zip$coefficients$zero)))
      }
    if(model=="NB"){
      skip.zinb.red <- zeroinfl(skipped ~ male + race + self.con1.m + ses.m +
      achievement.m + college | male + race + self.con1.m + ses.m + achievement.m +
      college, data = count.data[-i,], link = "logit", dist = "negbin", trace = TRUE, EM
      = FALSE)
    cooks[i]<-t(rbind(as.matrix(skip.zinb.red$coefficients$count),
    as.matrix(skip.zinb.red$coefficients$zero),
    as.matrix(skip.zinb.red$theta))-
    rbind(as.matrix(skip.zinb$coefficients$count),
    as.matrix(skip.zinb$coefficients$zero),
    as.matrix(skip.zinb$theta)))%*%
      (-(skip.zinb$optim$hessian))%*%(rbind(
    as.matrix(skip.zinb.red$coefficients$count),
    as.matrix(skip.zinb.red$coefficients$zero),
    as.matrix(skip.zinb.red$theta))-
    rbind(as.matrix(skip.zinb$coefficients$count),
    as.matrix(skip.zinb$coefficients$zero),
    as.matrix(skip.zinb$theta)))
} }
```

```
   return(cooks)
}

# generate and plot Cook's D for the zero-inflated Poisson model
cooks.out<-g.cooks.zi(model="ZIP")
plot(cooks.out ,xlab="Case Number", ylab="Cook's D")
abline(h=(4/nrow(count.data)), col="red")

# generate and plot Cook's D for the zero-inflated negative binomial model
cooks.out <- g.cooks.zi(model="ZINB")
plot(cooks.out ,xlab="Case Number", ylab="Cook's D")
abline(h=(4/nrow(count.data)), col="red")

# linearity plots for negative binomial model
plot(as.factor(count.data$male),resid(skip.nb),xlab="Sex (0 = Female, 1 = Male)",
      ylab="Residuals")
plot(count.data$race,resid(skip.nb),xlab="Race",ylab="Residuals")
plot(count.data$self.con1.m,resid(skip.nb),xlab="Self-concept", ylab="Residuals")
plot(count.data$ses.m,resid(skip.nb),xlab="SES", ylab="Residuals")
plot(as.factor(count.data$college),resid(skip.nb),xlab="Plan on Going to College (0 = No,
      1 = Yes)", ylab="Residuals")
plot(count.data$achievement.m,resid(skip.nb),xlab="Academic Achievement",
      ylab="Residuals")
```

Select Final Model

```
# define the NB models to compare
cand.models <- list( )
cand.models[[1]] <- glm.nb(skipped ~ male + race + college + self.con1.m + ses.m +
      achievement.m, data = count.data)
cand.models[[2]] <- glm.nb(skipped ~ male + race + college + self.con1.m  +
      achievement.m, data = count.data)
cand.models[[3]] <- glm.nb(skipped ~ male + race + college  + ses.m + achievement.m, data
      = count.data)
cand.models[[4]] <- glm.nb(skipped ~ male + race + self.con1.m + ses.m + achievement.m,
      data = count.data)
cand.models[[5]] <- glm.nb(skipped ~ male  + college + self.con1.m + ses.m +
      achievement.m, data = count.data)
cand.models[[6]] <- glm.nb(skipped ~  race + college + self.con1.m + ses.m +
      achievement.m, data = count.data)
cand.models[[7]] <- glm.nb(skipped ~ male + race + college + self.con1.m + ses.m, data =
      count.data)
cand.models[[8]] <- glm.nb(skipped ~  race + college +  ses.m + achievement.m, data =
      count.data)
cand.models[[9]] <- glm.nb(skipped ~  college +  ses.m + achievement.m, data =
      count.data)
cand.models[[10]] <- glm.nb(skipped ~  college +   achievement.m, data = count.data)
cand.models[[11]] <- glm.nb(skipped ~  college , data = count.data)

# name the models
model.names <- c("Full", "SES", "SlfCon", "College", "Race", "Sex", "Ach",
      "Sex.SlfCon","Sex.SlfCon.Race", "Sex.SlfCon.Race.SES", "Sex.SlfCon.Race.SES.Ach")
names(cand.models) <- model.names

# calculate and combine AIC, AIC weights, and BIC
results <- data.frame(models = model.names)
results$bic.val <- unlist(lapply(cand.models, BIC))
results$bic.rank <- rank(results$bic.val)
results$aic.val <- unlist(lapply(cand.models, AIC))
results$aic.delta <- results$aic.val-min(results$aic.val)
```

```
results$aic.likelihood <-  exp(-0.5* results$aic.delta)
results$aic.weight <- results$aic.likelihood/sum(results$aic.likelihood)
# sort models by AIC weight
results <- results[rev(order(results[, "aic.weight"])),]
esults$cum.aic.weight <- cumsum(results[, "aic.weight"])

# final model
skip.final.nb <- glm.nb(skipped ~  college +  ses.m + achievement.m, data = count.data)
```

Citation:

Beaujean, A. Alexander, Grant, Morgan B. (2016). Tutorial on Using Regression Models with Count Outcomes using R. *Practical Assessment, Research & Evaluation*, 21(2). Available online: http://pareonline.net/getvn.asp?v=21&n=2

Corresponding Author:

A. Alexander Beaujean
Department of Educational Psychology
One Bear Place #97301
Waco TX 76798-7301

email: Alex_Beaujean [at] baylor.edu