5-2018

On the Small Count Inflated Poisson Distribution

Michael Floren
University of Northern Colorado

UNIVERSITY OF NORTHERN COLORADO

Greeley, Colorado

The Graduate School

ON THE SMALL COUNT INFLATED POISSON DISTRIBUTION

A Dissertation Submitted in Partial Fulfillment
of the Requirement for the Degree of
Doctoral of Philosophy

Michael Floren

College of Education and Behavioral Sciences
Department of Applied Statistics and Research Methods

May 2018

This Dissertation by: Michael Floren

Entitled: *On the Small Count Inflated Poisson Distribution*

has been approved as meeting the requirement for the Degree of Doctor of Philosophy in College of Education and Behavioral Sciences in Department of Applied Statistics and Research Methods.

Accepted by the Doctoral Committee

Trent Lalonde, Ph.D., Research Advisor

Khalil Shafie, Ph.D., Committee Member

Jay Schaffer, Ph.D., Committee Member

Nancy Sileo, Ed.D., Faculty Representative

Date of Dissertation Defense ______________________________

Accepted by the Graduate School

Linda L. Black, Ed.D.
Associate Provost and Dean
Graduate School and International Admissions

ABSTRACT

Floren, Michael. *On the Small Count Inflated Poisson Distribution*. Published Doctor of Philosophy dissertation, University of Northern Colorado, 2018.

Inflated count distributions are used in situations where counts of an underlying distribution of a population are larger than expected by traditional count distributions. One of the most commonly used inflated count distributions is the zero-inflated Poisson distribution. This work demonstrates the construction of a small count inflated Poisson distribution, of which the zero-inflated Poisson distribution is a special case. Model construction and parameter estimation are shown. Simulations analyzing asymptotic properties, group prediction, and count prediction are presented. Conclusions and recommendations for future research are discussed.

ACKNOWLEDGMENTS

Unfortunately, there is not room in this brief section to thank all of the people who have invaluably contributed to my life and education for the last five years and beyond. Suffice it to say that I am who I am because of an immense collective effort of individuals who have wanted to see me succeed. To all you I owe first recognition: thank you for your time and and your continued faith in me, even (especially) when I didn't deserve it.

I would like to thank my advisor, Trent Lalonde, for the support, encouragement, and insight you have provided through the dissertation process. Thanks for being understanding whenever I was freaking out, and for politely smiling each time I promised it would be the last time.

I am also grateful to members of my committee, Jay Schaffer, Khalil Shafie Holighi, and Nancy Sileo, for the time that you have invested into review and consideration of this dissertation. Your guidance has extended far beyond this project, and I am so grateful that I have been able to work with each of you through my tenure here.

I would be remiss if I did not include a sincere thanks to all faculty and instructors I have worked with throughout my 25 years of schooling. Your patience and persistence with a recalcitrant student opened the door to this path, and deserves my highest recognition and thanks.

I would also like to thank my classmates in the Applied Statistics and Research Methods program at the University of Northern Colorado. There have been many long lunches and late nights between us, full of technical discussions and beer. I won't forget them.

I would like to acknowledge my former coworkers in high school education. I constantly reflect on your commitment, and insights from our discussions are what have driven me to this field and to the topic of this dissertation. To G, Harkness (go Team Awesome!), McKay, Tucker, Claire, and the rest: thank you for your friendship and your service.

To my family, I am immensely grateful to the unwaivering and unending support of my family. Daniel and Samuel: thanks for being encouraging, and pretending to be excited even when I'm talking about this project for the umpteenth time. Andrew and Lydia: words can't adequately express how much you have impacted my life. I'll say this: I love you, and I wouldn't change anything.

Last, but certainly not least, to my wife, Kiley: thank you for your truly limitless patience, for putting up with all of the late nights coding and the last minute cancellations while a ten minute fix takes two hours. I love you, and I'm sorry I couldn't come up with a clever acronym to make it the KILEY distribution.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

Count regression models are used to explain an outcome variable whose values fall within the whole numbers. This data type is quite common, and can be seen: in the number of pills an individual takes on a daily basis, in the number of cars in a parking lot at a certain time, or in the number of trees in a forest. A common source of count data is education: the number of students who engage in an activity, the number of detentions received by students in a given time, or the number of new teachers hired in the last year. Another common source is in health care: the number of cigarettes smoked by an individual in the last week, the number of high-risk behaviors engaged in by individuals, or the number of visits to an emergency room (ER) by individuals. All of these situations clearly have whole number values: that is, they have response values that are non-negative integers. The modeling of these values have a wide and growing applicability in education, health care, and politics. Additionally, as count modeling becomes more accessible, the use of count regression models has been increasing in recent years (Atkins, Baldwin, Zheng, Gallop, & Neighbors, 2013).

Count Data and Modeling

Count data are data that have observations falling within the whole numbers. These observations are traditionally assumed as being from the Poisson distribution, or in the case of overdispersion (a break in the mean variance relationship assumed by the Poisson), from a generalized Poisson or one of a series of negative binomial distributions (Agresti, 2002; Hilbe, 2011). As with many statistical assumptions, assumptions regarding the distribution of count data are usually made a-priori, and thus may or may not be correct for a given data situation. Additionally, the requirement of the assumption of normality for general linear models is inappropriate with count data, as count data are inherently non-negative (Agresti, 2002). To account for this, a generalized linear model (GLM) framework is used in the modeling of count data (McCullagh & Nelder, 1989). This framework is built via three pieces: a random component, a systematic component, and a link component. The random component is the assumed distribution of the outcome. For count variables, this is usually the Poisson or a negative binomial distribution. The systematic component allows for covariates to be related to the outcome. The link component allows for a relationship between the expectation of the outcome and the covariates to change, further specifying the model. A common link function for count regression is the natural log link, which eliminates the possibility of predicting values outside the response space (e.g., negative numbers).

Excess-Zero Models

One common issue with count models is that the amount of zeros exceeds that which would be expected by the count distribution. This is known as "excess-zero" data, and has provoked a separate class of models to deal with its occurrence. Some common situations where this may arise could be: the number of cups of coffee had by an individual in a week, the number of alcoholic beverages consumed in the previous week, or the number of emergency room visits by an individual in the last year. All of these situations are cases where the population of counts can be split into users and non-users (for instance, smokers and non-smokers), with both groups exhibiting different characteristics. Excess-zero models attempt to determine likelihood of an observation coming from a population. After the population has been determined, they attempt to predict the count for the "users" using GLM.

There are two main classes of models used for excess-zero cases: the zero-inflated model, and the hurdle model. Both of these models split the observations into classifications based on group. They differ, however, in the distribution allowed to the “users” group. The zero-inflated model allows participants in this group to take on a value of zero, while the hurdle model forces all participants classified as users to be non-zero. To account for this, the hurdle uses truncated versions of the Poisson or negative binomial distributions. In essence, these models each use two components: a logistic component, to determine group membership, and a count component, to predict the count. The zero-inflated and hurdle differ in the distribution allowed to the count component: a hurdle model uses a zero-truncated distribution, a zero-inflated model uses an un-truncated distribution.

The choice between the two models is often made based off of the situation. The hurdle model assumes that an observed zero defines a classification. On the other hand, the zero-inflated model allows observed zeros to come from two sources: the “non-users” group, and the “users” group. In an applied situation, the researcher must consider which model is best supported based on theoretical considerations of the groups. For example, consider the goal of modeling the number of cigarettes smoked in the past week. If the researcher believes that smokers may have smoked zero cigarettes, a zero-inflated model would be selected. If the researcher believes that all smokers smoked more than zero cigarettes in the past week, a hurdle model should be selected.

Just as general linear models are inappropriate for modeling count data, so too are count models inappropriate for modeling excess-zero count data. The heavy weight in the 0 observations creates an overdispersion issue which cannot be resolved by use of the negative binomial distribution (Perumean-Chaney, Morgan, McDowall, & Aban, 2013) . The cost is bais in the model’s parameter estimates and standard errors (Perumean-Chaney et al., 2013).

Finite Mixture Models

Finite mixture models (FMM) are a class of models that combines a finite number of probability distribution functions to better model the data (Everitt & Hand, 1981; McLachlan & Peel, 2000; Shalizi, 2012). These are best used when multiple populations contribute to the observed outcome. A classic example of this is in the production of stamps, where multiple production sites may each have a different distribution of stamp widths in their production. This example also demonstrates the unknown nature of the outcome variable in that, on observation of a single stamp, we do not know the source site or distribution. Finite mixture models are a class of models designed to estimate the outcome for such a variable.

In general, all excess-zero models can be considered a subset of finite mixture models. In the FMM framework, a zero-inflated model would consist of the combination of two distributions: a degenerate distribution at zero, and a count distribution over the whole numbers. Similarly, a hurdle model would consist of a combination of two distributions: a degenerate distribution at zero, and a count distribution over the natural numbers (whole numbers, not including zero).

Beyond Zero

In recent years, there has been a growing interest in modeling excess counts beyond just zero. The zero-and-one inflated model allows counts of zero and one to be inflated (Alshkaki, 2016, 2017; Lin & Tsai, 2013; Melkersson & Olsson, 1999; Zhang, Tian, & Ng, 2016). This may better model situations such as the casual drinker, who only has one drink per week. This model closely follows how zero-inflated models are constructed, save that it determines membership among three groups: the zero count group, the one count group, and the Poisson group. The zero-and-k inflated model allows counts of zero and an arbitrary *k* to be inflated (Lin & Tsai, 2013). This better models situations such as the number of dental visits in a previous year, where there is a clear inflation at zero (those who don't go) and two (those who go the recommended number of times). This model is very similar to the zero-and-one inflated model, save that it determines membership among the zero count group, *k* count group, and the Poisson group. Finally, zero-to-k models allow multiple counts to be inflated from zero to an arbitrary *k* (Giles, 2007). These models are also called multinomially-inflated models, indicating that multiple counts are allowed to be inflated. This model uses a multinomial distribution to assign membership among $k+1$ groups: the zero group, the one group, ..., the k group, and the Poisson count group.

Though this recent surge in broader models has been growing, research in the area has focused on individual points of inflation. That is, with multiple points of inflation in the small counts, each count value (zero, one, etc) is given its own prediction via probability. From this framework, relationships between the small count inflated values are not specified, even in cases where a clear theoretical relationship may exist. For instance, consider the number of emergency room visits by college students in a given year. Though the counts of zero and one may be inflated, it is theoretically supported that the inflation of zero is greater than the inflation of one. In such a case, a model that allows a relationship between the inflated counts would be appropriate.

Study Objectives

This study proposed a small count inflated model that allows the inflated small counts to follow a pre-specified distribution. In the appropriate situation, this model allows researchers to use a theoretical backing to better explain data situations where there is a relationship between small count inflated values. Though this model has a more complicated structure than data with single point inflation (e.g., zero-inflated), for data with multiple point inflation (e.g., zero-one inflated, zero-k inflated) the model specifies equivalent or fewer components than existing models.

The purpose of this study was to construct a small count inflated Poisson (SCIP) model, including maximum likelihood estimation of parameters. Additionally, this study compared the performance of the proposed model to that of other models that may be used in similar situations. Analysis of both simulated and observed data was used.

Research Questions

The following questions were addressed:

Q1 How can a SCIP distribution be specified?

Q2 How can parameters be estimated for the SCIP distribution?

Q3 How can a SCIP distribution be implemented in `R`?

Q4 In terms of percentage correct and area under the ROC curve, how well will the SCIP distribution predict group membership compared to the zero-inflated and multinomially-inflated Poisson models?

Q5 In terms of mean squared error (MSE), how well will the SCIP distribution predict counts compared to the zero-inflated and multinomially-inflated Poisson models?

Study Limitations

This study only addressed a small count inflated Poisson model while, for cases where a cutoff is misspecified, a negative binomial model may be more forgiving. Additionally, this study only considered situations analogous to the zero-inflated model, and not those analogous to the hurdle model. That is, this study allowed the distributions for small and large counts to overlap in the small count section.

Overview

This dissertation consists of five chapters. Chapter I introduces a brief history of the field, including where the current model builds on previous research. It ends with the questions this dissertation addresses, including a brief discussion of the limitations.

Chapter II offers a deeper discussion of the history of the field, including specification of previously used models. The literature review provided shows the gap in current research, and illustrates the applicability for the current research.

Chapter III illustrates the SCIP distribution, including the process of estimating parameters via maximum likelihood. The chapter closes with a discussion of how the research questions will be addressed.

Chapter IV describes the SCIP model and estimation via simulation. Comparisons between previous models and the SCIP model using both simulated and observed data are shown.

Chapter V discusses the results and presents conclusions. Limitations and calls for future work are also discussed.

CHAPTER II

LITERATURE REVIEW

Counts

Count Data

Many natural phenomena take a form known as “count data”. In general, instances of these variables indicate the number of times an event occurs within a fixed period of time. Examples in applied research include the number of manufacturing defects (Lambert, 1992), the number of cups of coffee consumed by an individual in a day (Mullahy, 1986), or the number of bacteria in leucocytes (Giles, 2007). Count variables such as these are natively nonnegative, and can only take on integer values.

Modeling of data for prediction or explanation can be more powerful if the underlying distribution of the outcome is known. Because of this, it is advantageous for researchers to identify distributions of potential outcome variables. The nature of each variable necessitates the use of a distribution which can align to the nature of the data (in this case, nonnegative integers). Several distributions used in the modeling of count data are now be presented.

Poisson distribution. One distribution, the Poisson distribution, can be used to model certain instances of count data. This distribution is a discrete distribution that expresses the probability of a given number of independent events occurring within a given amount of time. The probability mass function (PMF) for this distribution is given in Equation 1,

$$f(y|\lambda) = \frac{e^{-\lambda}\lambda^{y}}{y!}, \tag{1}$$

where $k = 0,1,2,\ldots,$ λ is the rate of occurrence, and Y a random count variable. The mean and variance of the Poisson distribution are equivalent, and are shown in Equation 2 to be

$$\begin{aligned} E[Y] &= \lambda, \\ Var(Y) &= \lambda. \end{aligned} \tag{2}$$

Negative binomial distribution. Another distribution, the Negative Binomial distribution, can also be used in the modeling of count data. This distribution is also a discrete distribution that expresses the probability of a certain number of independent events occurring before a specified number of successes. This distribution differs from the Poisson distribution in that the Negative Binomial allows the variance to exceed the mean (overdispersion). The negative binomial distribution is a commonly used solution for modeling overdispersed data.

Binomial distribution. The Binomial distribution may be used in specific cases of modeling count data. This distribution expresses the probability of k successes in a sequence of n independent trials. The Bernoulli distribution is a special case of this, and is modeled by the Binomial when $n = 1$. The PMF for this distribution is given in Equation 3,

$$P(Y = k) = \binom{n}{k} \pi^k (1-\pi)^{n-k}, \tag{3}$$

where $k = 0,1,\ldots,n$ is the number of successes, n is the number of trials, π is the probability of success, and Y is the random count variable. The mean and variance of the Binomial distribution are shown in Equation 4 to be

$$\begin{aligned} E[Y] &= \pi, \\ Var(Y) &= n\pi(1-\pi). \end{aligned} \tag{4}$$

Truncated distributions. Certain count models use zero-truncated versions of the Poisson distribution in addition to its traditional form. In general, zero-truncated distributions are calculated as

$$f_{\mathcal{ZT}}(y) = \frac{f(y)}{1-f(0)}, y \in \mathbb{N}^{+}, \tag{5}$$

where $f(y)$ is the PMF and $f_{\mathcal{ZT}}(y)$ is the zero-truncated PMF. The zero-truncated Poisson PMF thus takes on the form

$$f(y|\lambda) = \frac{\frac{e^{-\lambda}\lambda^{y}}{y!}}{1-e^{-\lambda}}, \tag{6}$$

where $y = 1, 2, \ldots$ is the observed count and λ is the rate of occurrence.

Count Models

The modeling of count data falls under the generalized linear model (GLM) framework of models. GLM allows a large degree of flexibility when modeling count data. Broadly, the GLM allows the mean of a nonnormal response to be functionally related to the predictors. This provides a powerful tool for the analysis of nonnormal data (Nelder & Wedderburn, 1972). Examples of such models are the number of pulmonary embolisms prevented by different brands of catheter, the number of emergency room patients by location of a facility, and the number of death notices given by season (Hasselblad, 1969).

The GLM is made up of three components: a random component, a systematic component, and a link function. The random component specifies the assumed distribution from which the outcome variable is expected. It can take on any distribution from the exponential family, including the Poisson distribution. The systematic component specifies the predictors and their associated parameters, which is related to the mean of the distribution through a link component. The model can be seen in Equation 7 where $\mathbf{Y}$ is the $n \times 1$ response vector, $\boldsymbol{\mu}$ is the $n \times 1$ mean vector, $\mathbf{X}$ is the $n \times p$ design matrix, $\boldsymbol{\beta}$ is the $p \times 1$ vector of parameters, and g is the link function. Each element of $\mathbf{Y}$

is assumed to be realized from a distribution in the exponential family (the random component), and $\mathbf{X}\boldsymbol{\beta}$ (the systematic component) is related to $\boldsymbol{\mu}$ via the differentiable, monotone link function g,

$$E\left[\mathbf{Y}\right] = \boldsymbol{\mu} = g^{-1}(\mathbf{X}\boldsymbol{\beta}). \tag{7}$$

The flexibility of the GLM allow its use in many situations where the outcome is assumed to be nonnormal.

Poisson generalized linear model. An application with a Poisson distributed count outcome is straightforward, with the random component being defined as a Poisson distribution and the link component being defined appropriately (traditionally the natural log). The GLM model is given in Equation 8,

$$E[\mathbf{Y}] = \boldsymbol{\lambda} = e^{(\mathbf{X}\boldsymbol{\beta})}, \tag{8}$$

where $\mathbf{Y}$ is the $n \times 1$ response vector, $\boldsymbol{\lambda}$ is the $n \times 1$ mean vector, $\mathbf{X}$ is the $n \times p$ design matrix, and $\boldsymbol{\beta}$ is the $p \times 1$ vector of parameters.

Poisson GLMs are utilized when the outcome variable is assumed to be a count variable where the mean and variance are the same. In this case, the each of the β's can be interpreted as the multiplicative change in the outcome for a unit change in the respective independent variable.

Bernoulli generalized linear model. A separate application with a Bernoulli outcome is equally convenient, with the random component being defined as a Bernoulli distribution and the link component being defined appropriately (traditionally the logit function). This model is also known as logistic regression. The GLM model is given in Equation 9,

$$E[\mathbf{Y}] = \boldsymbol{\pi} = \text{logit}^{-1}(\mathbf{X}\boldsymbol{\beta}), \tag{9}$$

where $\mathbf{Y}$ is the $n \times 1$ response vector, $\boldsymbol{\pi}$ is the $n \times 1$ mean vector, $\mathbf{X}$ is the $n \times p$ design matrix, $\boldsymbol{\beta}$ is the $p \times 1$ vector of parameters, and the logit function is defined as the log of the odds. That is,

$$\operatorname{logit}(\pi) = \log\left(\frac{\pi}{1-\pi}\right). \tag{10}$$

Bernoulli GLMs are utilized when the outcome variable is binary. In this case, each of the β's can be interpreted as the log of the odds ratio between the two outcome classes and two populations separated by a unit change in the respective independent variable.

Estimation and hypothesis testing. Estimation and hypothesis testing for GLM are performed via maximum likelihood (ML). First, the likelihood function is found from the PDF. In general, the PDF is taken to be the likelihood function, though the likelihood function is viewed as a function of the parameters given the data, while the PDF is a function of the data given the parameters. After the likelihood function is obtained, the derivative of the likelihood function is taken, set equal to zero, and solved for $\boldsymbol{\beta}$ to obtain parameter estimates via the estimating equations. In equation form, the estimating equations are given as

$$\frac{\partial L}{\partial \boldsymbol{\beta}} = \mathbf{0}, \tag{11}$$

where L is the log-likelihood, $\boldsymbol{\beta}$ are the parameters shown in Equation 7, and ∂ is the partial derivative.

Standard errors of the parameter estimates are found by first taking the second derivative of the log-likelihood function to find the information matrix. That is, asymptotically,

$$\hat{Cov}\left(\hat{\boldsymbol{\beta}}\right) = \left(-E\left[\left.\frac{\partial^2 L}{\partial \boldsymbol{\beta}^2}\right|_{\hat{\boldsymbol{\beta}}}\right]\right)^{-1}, \tag{12}$$

where L is the log-likelihood, $\boldsymbol{\beta}$ are the parameters shown in Equation 7, and ∂^2 is the partial second derivative represents the information matrix. The expected value of the information matrix is used, and in practice parameter estimates are used in place of unknown parameters. The standard errors are the square roots of the diagonal elements of the inverse information matrix (Agresti, 2002).

So long as regularity conditions are met, ML parameter estimates and standard errors can be used to perform hypothesis testing, with the ratio of the parameter to its standard error being compared to a given hypothesized value (usually zero). That is, with a nonnull standard error (denoted SE) of $\hat{\beta}$, the test statistic

$$z = \frac{(\hat{\beta} - \beta_0)}{SE} \tag{13}$$

has an approximate normal distribution when $\beta = \beta_0$, which may be compared to a standard normal table. For a two-sided alternative to the standard normal, z^2 has a chi-squared distribution with one degree of freedom. This type of statistic is called a Wald statistic (Agresti, 2002; Wald, 1943).

Excess-Zeros

Excess-Zero Data

Though the GLM is an excellent tool for modeling count data, often situations arise where counts do not strictly follow a Poisson distribution. One of these cases is known as an "excess-zero" case: where observed counts of zero exceed that which is expected by a Poisson distribution (Mullahy, 1986). Examples of these cases include the number of rare animals in a specified region, the number of cups of coffee had per day, or the number of detentions received by students in a school. The commonality of such data, including the importance of using appropriate methods to account for the overdispersion introduced by excess-zeros, has been recently discussed (He, Tang, Wang, & Crits-Christoph, 2014). These cases often can be seen as two populations who are mixed together: a population of individuals who have zeros, and a population of individuals who have a count either with or without a zero.

There are two classes of models which can describe this situation. The hurdle model, designed by Mullahy (1986), sets a "hurdle" which must be exceeded to move from the zero group to the count group. In this situation, individuals whose outcome is zero are always classified as the zero group. For example, consider the number of days a patient spends in a hospital. As a patient's first day in the hospital counts as day one, it would not be appropriate to assume that a patient in the hospital can record zero days. In this case, all records of zero days in the hospital represent participants who have not been to the hospital. One consequence of this is the implication that the count group can never obtain a count of zero, which may be unreasonable in certain situations.

The zero-inflated (ZI) model, introduced by Lambert (1992), allows individuals who are classified into the count group to have a count of zero. For example, consider participant use of marijuana in the previous day. Participants can be classified into "non-users", which don't use at all (zero), and "users", which use a certain amount of times (count). However, users may not have used the previous day, allowing the count to be zero *or* another number.

The analysis strategy in both cases is to consider a joint model. The first component uses a logistic regression to determine which group the individual is from. The second component uses a count GLM which predicts the count for an individual, conditional on count group membership. For the hurdle model, the count component is defined using a zero-truncated distribution. This reflects the assumption of the hurdle model: that any observed zeros are assumed to not be from the count distribution. For the ZI regression, an untruncated distribution is used. Both of these models, including their Poisson and Negative Binomial counterparts, will now be shown in further detail.

Zero-Inflated Models

Zero-inflated models are applied when observed zero values are assumed to come from multiple sub-populations: the zero population and the count population. In this case, count distributions with zeros are used to allow the zeros to come from either sub-population. These models, originally presented by Lambert (1992), add to the work of authors such as Mullahy (1986) by using a full Poisson distribution for the counts (in contrast to the zero-truncated Poisson). Several versions of ZI models are now presented.

Zero-inflated Poisson model. The PMF for the ZI model of a Poisson distributed outcome is readily described. Let π_0 represent the probability of being assigned to the zero count group, and $\pi_{\mathcal{P}}$ be the probability of being assigned to the Poisson group. Note that membership in these groups is mutually exclusive. That is, $\pi_0 + \pi_{\mathcal{P}} = 1$. The variable Y is said to follow a zero-inflated Poisson (ZIP) distribution, denoted $Y \sim ZIP(\pi_0, \lambda)$, if its PMF is given by

$$f(y|\pi,\lambda) = \begin{cases} \pi_0 + \pi_{\mathcal{P}} e^{-\lambda}, & y = 0 \\ \pi_{\mathcal{P}} \dfrac{e^{-\lambda}\lambda^y}{y!}, & y > 0, \end{cases} \tag{14}$$

where λ is the rate of occurrence of the Poisson.

The likelihood function is defined as follows. Let Z_0 and $Z_{\mathcal{P}}$ be used as indicators, with $Z_0 = 1$ when the observation is from the zero count group (and 0 otherwise), and $Z_{\mathcal{P}} = 1$ when the observation is from the Poisson group. It can be seen that Z_0 and $Z_{\mathcal{P}}$ are mutually exclusive. That is, for any given participant, $Z_0 + Z_{\mathcal{P}} = 1$. Given these, the PMF shown in Equation 14 can be expressed as shown in Equation 15, with

$$f(y|Z_0,\pi_0,\lambda) = \left(\pi_0 + \pi_{\mathcal{P}} e^{-\lambda}\right)^{Z_0} \left(\pi_{\mathcal{P}} \frac{e^{-\lambda}\lambda^y}{y!}\right)^{Z_{\mathcal{P}}}. \tag{15}$$

The ZIP model can be written in terms of a joint GLM with systematic, random, and link components. Lambert (1992) describes how $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_n)'$ and $\boldsymbol{\pi}_0 = (\pi_{0_1}, \ldots, \pi_{0_n})'$ satisfy

$$\log(\boldsymbol{\lambda}) = \mathbf{B}\boldsymbol{\beta}, \quad \text{and}$$

$$\operatorname{logit}(\boldsymbol{\pi}_0) = \mathbf{G}\boldsymbol{\gamma},$$

where $\mathbf{B}$ is an $n \times p$ covariate matrix, $\mathbf{G}$ is an $n \times q$ covariate matrix, $\boldsymbol{\beta}$ is a $p \times 1$ vector of parameters, $\boldsymbol{\gamma}$ is a $q \times 1$ vector of parameters, $\boldsymbol{\lambda}$ is an $n \times 1$ rate of occurrence vector, and $\boldsymbol{\pi}_0$ is an $n \times 1$ vector of probabilities of the i^{th} observation being in the zero count group. These identify the structural components as $\mathbf{B}\boldsymbol{\beta}$ and $\mathbf{G}\boldsymbol{\gamma}$ with the link functions as log and logit, respectively. The assumed distribution of the outcome is $Y \sim ZIP(\pi_0, \lambda)$.

In this case, the parameters $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ take on separate interpretations. The $\boldsymbol{\gamma}$ parameters relate to the probability that a case will fall in the zero or Poisson groups, while the $\boldsymbol{\beta}$ parameters are interpreted as parameters related to the count variables, conditional on the participant not being a part of the zero count group (though they may still have an observed value of zero). The $e^{\boldsymbol{\gamma}}$ vector can be interpreted as odds ratios while the $e^{\boldsymbol{\beta}}$ vector can be interpreted as the multiplicative change in expected counts.

Parameter estimates of the ZIP are obtained using the maximum likelihood estimation (MLE) via the expectation-maximization (EM) algorithm. MLE uses the likelihood function to find parameter estimates and standard errors. Parameters are found by solving the first derivative of the log-likelihood function for zero. Standard errors are found by evaluating the second derivative of the log-likelihood function. The EM algorithm is a numerical estimation method used in solving the above equations. An example of this process, using the ZIP model, is given below.

The first step in using maximum likelihood is to identify the likelihood equation. The log-likelihood may be maximized in place of the likelihood, as the log is a monotonically increasing function. For the ZIP model, the log-likelihood equation is shown as

$$\begin{aligned} L(\boldsymbol{\gamma},\boldsymbol{\beta}|\mathbf{y}) = &\sum_{y_i=0} \log\left(e^{\mathbf{G}_i\boldsymbol{\gamma}} + e^{-e^{\mathbf{B}_i\boldsymbol{\beta}}}\right) \\ &+ \sum_{y_i>0}\left(y_i\mathbf{B}_i\boldsymbol{\beta} - e^{\mathbf{B}_i\boldsymbol{\beta}}\right) \\ &- \sum_{i=1}^{n}\log\left(1+e^{\mathbf{G}_i\boldsymbol{\gamma}}\right) \\ &- \sum_{y_i>0}\log\left(y_i!\right), \end{aligned} \tag{16}$$

where $\mathbf{B}_i$ is the i^{th} row of an $n \times p$ covariate matrix $\mathbf{B}$, $\mathbf{G}_i$ is the i^{th} row of an $n \times q$ covariate matrix $\mathbf{G}$, $\boldsymbol{\beta}$ is a $p \times 1$ vector of parameters, $\boldsymbol{\gamma}$ is a $q \times 1$ vector of parameters, $\mathbf{y}$ is an $n \times 1$ vector of responses, and y_i is the i^{th} response.

Parameter estimates are found by taking the first derivative of the log-likelihood function and solving for zero for all parameters. In equation form, this can be shown as

$$\begin{aligned} \frac{\partial L}{\partial \boldsymbol{\beta}} &= \mathbf{0}, \\ \frac{\partial L}{\partial \boldsymbol{\gamma}} &= \mathbf{0}. \end{aligned} \tag{17}$$

Standard errors can be solved for by first finding the information matrix. That is, by determining the second derivative with regard to all sets of parameters. In equation form, if we let

$$\boldsymbol{\tau} = \begin{bmatrix} \boldsymbol{\beta} \\ \\ \boldsymbol{\gamma} \end{bmatrix},$$

the standard errors can be found by first determining

$$\hat{Cov}\left(\hat{\boldsymbol{\tau}}\right) = \left(-E\left[\left.\frac{\partial^2 L}{\partial \boldsymbol{\tau}^2}\right|_{\hat{\boldsymbol{\tau}}}\right]\right)^{-1} \tag{18}$$

for $\boldsymbol{\tau}$. The standard errors are the square roots of the diagonal elements for this inverse information matrix (Agresti, 2002).

The EM algorithm is often used to find solutions for maximum likelihood, and is well demonstrated by Lambert (1992). The method from Lambert (1992) assumes that one knows which group the zero is from, and then calculates the log-likelihood. That is, suppose it could be observed that $Z_i = 1$ when Y_i is from the zero group and $Z_i = 0$ when Y_i is from the count group. Lambert (1992) calculates the modified log-likelihood as

$$\begin{aligned}
\mathbf{L}_c(\boldsymbol{\gamma},\boldsymbol{\beta}|\mathbf{y},\mathbf{z}) =& \sum_{i=1}^{n} \log\left(f(z_i|\boldsymbol{\gamma})\right) + \sum_{i=1}^{n} \log\left(f(y_i|z_i,\boldsymbol{\beta})\right) \\
=& \sum_{i=1}^{n} \left(z_i\mathbf{G}_i\boldsymbol{\gamma} - \log\left(1 + e^{\mathbf{G}_i\boldsymbol{\gamma}}\right)\right) \\
&+ \sum_{i=1}^{n} (1 - z_i)\left(y_i\mathbf{B}_i\boldsymbol{\beta} - e^{\mathbf{B}_i\boldsymbol{\beta}}\right) \\
&- \sum_{i=1}^{n} (1 - z_i)\log(y_i!) \\
=& \mathbf{L}_c(\boldsymbol{\gamma}|\mathbf{y},\mathbf{z}) + \mathbf{L}_c(\boldsymbol{\beta}|\mathbf{y},\mathbf{z}) - \sum_{i=1}^{n} (1 - z_i)\log(y_i!),
\end{aligned} \tag{19}$$

where $\mathbf{B}_i$ is the i^{th} row of an $n \times p$ covariate matrix $\mathbf{B}$, $\mathbf{G}_i$ is the i^{th} row of an $n \times q$ covariate matrix $\mathbf{G}$, $\boldsymbol{\beta}$ is a $p \times 1$ vector of parameters, $\boldsymbol{\gamma}$ is a $q \times 1$ vector of parameters, $\mathbf{y}$ is an $n \times 1$ vector of responses, and y_i is the i^{th} response. Because the terms additively separate, the log-likelihood of $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ can be maximized separately and the log-likelihood becomes easier to maximize.

The EM algorithm alternates between the expectation step and maximization step. The estimation step estimates Z_i under the current estimates of $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$. The maximization step fixes the Z_i's with their estimated value, then maximizes $\mathbf{L}_c(\boldsymbol{\gamma},\boldsymbol{\beta}|\mathbf{y},\mathbf{z})$. The algorithm ends after the Z_i's, $\boldsymbol{\gamma}$'s, and $\boldsymbol{\beta}$'s converge. Details on each of these steps is now provided.

For the expectation step, Lambert (1992) estimates z_i by its posterior mean $Z_i^{(k)}$ (for the k^{th} iteration) under the estimates of $\boldsymbol{\gamma}^{(k)}$ and $\boldsymbol{\beta}^{(k)}$. That is

$$\begin{aligned} Z_i^{(k)} &= P[0|y_i, \boldsymbol{\gamma}^{(k)}, \boldsymbol{\beta}^{(k)}] \\ &\vdots \\ &= \left(1 + e^{\mathbf{G}_i \boldsymbol{\gamma}^{(k)} - e^{\mathbf{B}_i \boldsymbol{\beta}^{(k)}}}\right)^{-1} && \text{if } y_1 = 0 \\ &= 0 && \text{if } y_i = 1, 2, \ldots \end{aligned} \tag{20}$$

For the maximization step to find $\boldsymbol{\gamma}^{(k+1)}$ and $\boldsymbol{\beta}^{(k+1)}$, Lambert (1992) maximizes $\mathbf{L}_c(\boldsymbol{\gamma}|\mathbf{y},\mathbf{Z}^{(k)})$ and $\mathbf{L}_c(\boldsymbol{\beta}|\mathbf{y},\mathbf{Z}^{(k)})$, respectively (see Equation 19). Lambert (1992) notes that, in the $\boldsymbol{\beta}$ case, this amounts to a weighted Poisson regression (with weights of $1 - \mathbf{Z}^{(k)}$) as presented by McCullagh and Nelder (1989).

Hurdle Models

Hurdle models are applied when observed zero values are all assumed to come from the same sub-population. In this case, zero-truncated count distributions are used to distinguish the source of zero counts. These models were originally presented by Mullahy (1986) and are the first to address data with excess zeros. Several versions of the hurdle model are now presented.

Poisson hurdle model. The PMF for the hurdle model of a Poisson distributed outcome is readily described. Let π_0 represent the probability of being assigned to the zero count group, and $\pi_{\mathcal{TP}}$ be the probability of being assigned to the zero-truncated Poisson group. Note that membership in these groups is mutually exclusive. That is, $\pi_0 + \pi_{\mathcal{TP}} = 1$. The variable Y is said to follow a Poisson hurdle distribution, denoted $Y \sim PH(\pi_0, \lambda)$, if its PMF is given by

$$f(y|\pi_0,\lambda) = \begin{cases} \pi_0, & y = 0 \\ \pi_{\mathcal{TP}} \dfrac{\lambda^y e^{-\lambda}}{y!(1-e^{-\lambda})}, & y > 0, \end{cases} \tag{21}$$

where λ is the rate of occurrence of the zero-truncated Poisson.

The likelihood function is defined as follows. Let Z_0 and $Z_{\mathcal{TP}}$ be used as indicators, with $Z_0 = 1$ when the observation is from the zero count group (and 0 otherwise), and $Z_{\mathcal{TP}} = 1$ when the observation is from the zero-truncated Poisson group. It can be seen that Z_0 and $Z_{\mathcal{TP}}$ are mutually exclusive. That is, for any given participant, $Z_0 + Z_{\mathcal{TP}} = 1$. Given these, the PMF shown in Equation 21 can be expressed as shown in Equation 22, with

$$f(y|Z_0,\pi_0,\lambda) = (\pi_0)^{Z_0} \left(\pi_{\mathcal{TP}} \frac{\lambda^y e^{-\lambda}}{y!(1-e^{-\lambda})} \right)^{Z_{\mathcal{TP}}}. \tag{22}$$

The Poisson hurdle model can be written in terms of a joint GLM with systematic, random, and link components. This is shown as

$$\begin{aligned} \log(\boldsymbol{\lambda}) &= \mathbf{B}\boldsymbol{\beta}, \quad \text{and} \\ \text{logit}(\boldsymbol{\pi}_0) &= \mathbf{G}\boldsymbol{\gamma}, \end{aligned}$$

where $\mathbf{B}$ is an $n \times p$ covariate matrix, $\mathbf{G}$ is an $n \times q$ covariate matrix, $\boldsymbol{\beta}$ is a $p \times 1$ vector of parameters, $\boldsymbol{\gamma}$ is a $q \times 1$ vector of parameters, $\boldsymbol{\lambda}$ is an $n \times 1$ rate of occurrence vector, and $\boldsymbol{\pi}_0$ is an $n \times 1$ vector of probabilities of the i^{th} observation being in the zero count group. These identify the structural components as $\mathbf{B}\boldsymbol{\beta}$ and $\mathbf{G}\boldsymbol{\gamma}$ with the link functions as log and logit, respectively.

In this case, the parameters $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ take on separate interpretations. The $\boldsymbol{\gamma}$ parameters relate to the probability that a case will fall in the zero or Poisson groups, while the $\boldsymbol{\beta}$ parameters are interpreted as parameters related to the count variables, conditional on the participant not having an observed count of zero. The e^{β_j}'s can be interpreted as the multiplicative change in expected counts for the population with non-zero values.

Due to the truncated nature of the count distribution, estimation and hypothesis testing for the zero group and Poisson group can happen independently. This allows the parameters to be estimated using the ML processes shown previously for GLMs. That is, parameter estimates are calculated by taking Equation 22 as the likelihood function with hypothesis tests following from Wald statistics.

Zero-and-One Inflated

Zero-and-One Inflated Data

Allowing a mixture of the modeling of zero counts allowed a great deal of flexibility, however recent work has been done allowing mixtures of ones in addition to zeros, called the zero-and-one inflated Poisson (ZOIP) distribution. This work is an attempt to reflect models that are traditionally viewed as excess zero, but where counts of both zeros and ones may be in excess. For example, Melkersson and Olsson (1999) and Zhang et al. (2016) present examples of the number of dental visits in the past twelve months, while Zhang et al. (2016) presents other examples of criminal acts, fetal lamb movement, death notice data, and ammunition factory accidents.

Zero-and-One Inflated Models

Models in this class focus on a multinomial group placement, and split groups into three categories: a zero, a one, or a larger count. As with the ZIP models, individuals classified into the larger count group have counts modeled using a Poisson regression.

Originally presented in an unpublished paper by Melkersson and Olsson (1999) regarding dental visits, research into the ZOIP distribution fell off for several years. Later picked up by Zhang et al. (2016), they note that “to our best knowledge, only two papers... involve the ZOIP to date” (p. 11). Though such models for the hurdle distribution have been previously considered (for instance, see Silva and Covas (2000)), full development of these models in the ZI case is relatively recent.

The PMF of the ZOIP model can be readily described, and is quite similar to that shown in Equation 14 (Alshkaki, 2016, 2017; Lin & Tsai, 2013; Melkersson & Olsson, 1999; Zhang et al., 2016). Let π_0 represent the probability of being assigned to the zero count group, π_1 represent the probability of being assigned to the one count group, and $\pi_{\mathcal{P}}$ represent the probability of being assigned to the Poisson group. Note that membership in these groups is mutually exclusive. That is, $\pi_0 + \pi_1 + \pi_{\mathcal{P}} = 1$. The variable Y is said to follow a ZOIP distribution, denoted $Y \sim ZOIP(\lambda, \pi_0, \pi_1)$, if its PMF is given by

$$f(y|\lambda,\pi_0,\pi_1) = \begin{cases} \pi_0 + \pi_{\mathcal{P}} e^{-\lambda}, & y = 0 \\ \pi_1 + \pi_{\mathcal{P}} \lambda e^{-\lambda}, & y = 1 \\ \pi_{\mathcal{P}} \dfrac{\lambda^y e^{-\lambda}}{y!}, & y = 2, 3, \ldots, \end{cases} \tag{23}$$

where λ is the rate of occurrence of the Poisson.

The likelihood function is defined as follows. Let Z_0, Z_1, and $Z_{\mathcal{P}}$ be used indicators, with $Z_0 = 1$ when the observation is from the zero count group (and 0 otherwise), $Z_1 = 1$ when the observation is from the one count group, and $Z_{\mathcal{P}} = 1$ when the observation is from the Poisson group. It can be seen that Z_0, Z_1, and $Z_{\mathcal{P}}$ are mutually exclusive. That is, for a given participant, $Z_0 + Z_1 + Z_{\mathcal{P}} = 1$. Given these, the PMF shown in Equation 23 can be expressed as shown in Equation 24, with

$$\begin{aligned} f(y|\lambda,\pi_0,\pi_1,Z_0,Z_1) = & \left(\pi_0 + \pi_{\mathcal{P}} e^{-\lambda}\right)^{Z_0} \left(\pi_1 + \pi_{\mathcal{P}} \lambda e^{-\lambda}\right)^{Z_1} \\ & \times \left(\pi_{\mathcal{P}} \frac{\lambda^y e^{-\lambda}}{y!}\right)^{Z_{\mathcal{P}}} . \end{aligned} \tag{24}$$

Though these authors have defined the ZOIP distribution, they have not presented a specified model. For presentation of a model, a generalized logit function will be assumed. The ZOIP can then be written in terms of a joint GLM with systematic, random, and link components. This is shown as

$$\begin{aligned} \log(\boldsymbol{\lambda}) &= \mathbf{B}\boldsymbol{\beta}, \\ \log\left(\frac{\boldsymbol{\pi}_1}{\boldsymbol{\pi}_0}\right) &= \mathbf{G}\boldsymbol{\gamma}, \quad \text{and} \\ \log\left(\frac{\boldsymbol{\pi}_{\mathcal{P}}}{\boldsymbol{\pi}_0}\right) &= \mathbf{A}\boldsymbol{\alpha}, \end{aligned}$$

where $\mathbf{B}$ is an $n \times p$ covariate matrix, $\mathbf{G}$ is an $n \times q$ covariate matrix, $\mathbf{A}$ is an $n \times r$ covariate matrix, $\boldsymbol{\beta}$ is a $p \times 1$ vector of parameters, $\boldsymbol{\gamma}$ is a $q \times 1$ vector of parameters, $\boldsymbol{\alpha}$ is an $r \times 1$ vector of parameters, $\boldsymbol{\lambda}$ is an $n \times 1$ rate of occurrence vector, $\frac{\boldsymbol{\pi}_1}{\boldsymbol{\pi}_0}$ is an $n \times 1$ vector of ratios of the probability of the i^{th} observation being in the one count group to being in the zero count group, and $\frac{\boldsymbol{\pi}_{\mathcal{P}}}{\boldsymbol{\pi}_0}$ is an $n \times 1$ vector of ratios of the probability of the i^{th} observation being in the Poisson group to being in the zero count group. Parameter estimates are constructed using a maximum likelihood approach (Alshkaki, 2016; Zhang et al., 2016).

Though this model describes zero-and-one inflated count situations, it does not extend to inflations beyond one.

Zero-and-K Inflated

Zero-and-K Inflated Data

In addition to allowing counts of 0 and 1 to be inflated, Lin and Tsai (2013) proposed a more generic series of models where counts of 0 and an arbitrary k can be inflated. They present an example dataset of the number of Pap tests in the last six years, which is shown to be inflated at zero and six.

Zero-and-K Inflated Models

Lin and Tsai (2013) define the PMF of the zero-and-k inflated Poisson (ZKIP) similarly to that of Equation 23. To avoid confusion, the point of inflation other than zero will be denoted $\mathcal{K}$. Let π_0 represent the probability of being assigned to the zero count group, $\pi_{\mathcal{K}}$ represent the probability of being assigned to the one count group, and $\pi_{\mathcal{P}}$ represent the probability of being assigned to the Poisson group. Note that membership in these groups is mutually exclusive. That is, $\pi_0 + \pi_{\mathcal{K}} + \pi_{\mathcal{P}} = 1$. The variable Y is said to follow a ZKIP distribution, denoted $Y \sim ZKIP(\lambda, \pi_0, \pi_{\mathcal{K}})$, if its PMF is given by

$$f(y|\lambda, \pi_0, \pi_{\mathcal{K}}) = \begin{cases} \pi_0 + \pi_{\mathcal{P}} e^{-\lambda}, & y = 0 \\ \pi_{\mathcal{K}} + \pi_{\mathcal{P}} \dfrac{\lambda^y e^{-\lambda}}{y!}, & y = \mathcal{K} \\ \pi_{\mathcal{P}} \dfrac{\lambda^y e^{-\lambda}}{y!}, & y \in \mathbb{N}^+ \backslash \{\mathcal{K}\}, \end{cases} \tag{25}$$

where λ is the rate of occurrence of the Poisson.

The likelihood function is defined as follows. Let $Z_0, Z_{\mathcal{K}}$, and $Z_{\mathcal{P}}$ be used as indicators, with $Z_0 = 1$ when the observation is from the zero count group (and 0 otherwise), $Z_{\mathcal{K}} = 1$ when the observation is from the $\mathcal{K}$ group, and $Z_{\mathcal{P}} = 1$ when the observation is from the Poisson group. It can be seen that $Z_0, Z_{\mathcal{K}}$, and $Z_{\mathcal{P}}$ are mutually exclusive. That is, for a given participant, $Z_0 + Z_{\mathcal{K}} + Z_{\mathcal{P}} = 1$. Given these, the PMF shown in Equation 25 can be expressed as shown in Equation 26, with

$$\begin{aligned} f(y|\lambda, \pi_0, \pi_1, Z_0, Z_1) = \left(\pi_0 + \pi_{\mathcal{P}} e^{-\lambda}\right)^{Z_0} \left(\pi_{\mathcal{K}} + \pi_{\mathcal{P}} \frac{\lambda^{\mathcal{K}} e^{-\lambda}}{\mathcal{K}!}\right)^{Z_{\mathcal{K}}} \\ \times \left(\pi_{\mathcal{P}} \frac{\lambda^y e^{-\lambda}}{y!}\right)^{Z_{\mathcal{P}}}. \end{aligned} \tag{26}$$

Lin and Tsai (2013) present a model with the form of a generalized logit. The ZKIP using this model can be written in terms of a joint GLM with systematic, random, and link components. This is shown as

$$\begin{aligned}\log(\boldsymbol{\lambda}) &= \mathbf{B}\boldsymbol{\beta},\\ \log\left(\frac{\boldsymbol{\pi}_{\mathcal{K}}}{\boldsymbol{\pi}_0}\right) &= \mathbf{G}\boldsymbol{\gamma}, \quad \text{and}\\ \log\left(\frac{\boldsymbol{\pi}_{\mathcal{P}}}{\boldsymbol{\pi}_0}\right) &= \mathbf{A}\boldsymbol{\alpha},\end{aligned}$$

where **B** is an $n \times p$ covariate matrix, **G** is an $n \times q$ covariate matrix, **A** is an $n \times r$ covariate matrix, $\boldsymbol{\beta}$ is a $p \times 1$ vector of parameters, $\boldsymbol{\gamma}$ is a $q \times 1$ vector of parameters, $\boldsymbol{\alpha}$ is an $r \times 1$ vector of parameters, $\boldsymbol{\lambda}$ is an $n \times 1$ rate of occurrence vector, $\frac{\boldsymbol{\pi}_{\mathcal{K}}}{\boldsymbol{\pi}_0}$ is an $n \times 1$ vector of ratios of the probability of the i^{th} observation being in the $\mathcal{K}$ count group to being in the zero count group, and $\frac{\boldsymbol{\pi}_{\mathcal{P}}}{\boldsymbol{\pi}_0}$ is an $n \times 1$ vector of ratios of the probability of the i^{th} observation being in the Poisson group to being in the zero count group. Parameter estimates are constructed using a maximum likelihood approach (Lin & Tsai, 2013).

Though this model describes zero-and-k inflated count situations, it does not allow inflated counts between zero and k.

Multinomially-Inflated and Models

Multinomially-Inflated Data

In addition to allowing counts of 0 and $\mathcal{K}$ to be inflated, Giles (2007) proposed a series of models allowing counts from 0 *to* $\mathcal{K}$ to be inflated. Additionally, applied practitioners have pursued the modeling of this data type through non-traditional forms of hurdle models (e.g., the double-hurdle model introduced by Miranda (2010)). Giles (2007) presents several example datasets: leucocyte data, which counts the number of bacteria in leucocytes; Botswana fertility data, which counts the number of living children for a sample of women from Botswana; and hits on the Hot 100 chart, which counts the number of weeks that a given recording is at the top spot in the Hot 100 chart.

Multinomially-Inflated Models

Giles (2007) defines the PMF of the zero-to-$\mathcal{K}$ multinomially-inflated Poisson (MIP) similarly to that of Equation 25. Let π_j represent the probability of being assigned to the j^{th} count group, where $j = 0, 1, \ldots, c$. In this instance, c represented the "cutoff" of inflation, as no values above c are treated as inflated. Additionally, let $\pi_{\mathcal{P}}$ represent the probability of being assigned to the Poisson count group. Note that membership in these groups is mutually exclusive. That is, $\sum \pi_j + \pi_{\mathcal{P}} = 1$. The variable Y is said to follow an MIP distribution, denoted $Y \sim MIP(\lambda, \pi_0, \ldots, \pi_c)$, if its PMF is given by

$$f(y|\lambda, \pi_0, \ldots, \pi_c) = \begin{cases} \pi_0 + \pi_{\mathcal{P}} e^{-\lambda}, & y = 0 \\ \pi_1 + \pi_{\mathcal{P}} \lambda e^{-\lambda}, & y = 1 \\ \vdots & \\ \pi_c + \pi_{\mathcal{P}} \dfrac{\lambda^c e^{-\lambda}}{c!}, & y = c \\ \pi_{\mathcal{P}} \dfrac{\lambda^y e^{-\lambda}}{y!}, & y > c, \end{cases} \tag{27}$$

where λ is the rate of occurrence of the Poisson.

The likelihood function is defined as follows. Let Z_j for $j = 0, \ldots, c$ be used as indicators, with $z_j = 1$ when the observation is from the j^{th} count group (and 0 otherwise). Additionally, let $Z_{\mathcal{P}} = 1$ when the observation is from the Poisson group. It can be seen that these groups are mutually exclusive. That is, for any given participant, $\sum_{\forall j} Z_j + Z_{\mathcal{P}} = 1$. Given these, the PMF shown in Equation 27 can be expressed as shown in Equation 28, with

$$\begin{aligned} f(y|\lambda, \pi_0, \pi_1, Z_0, Z_1) = &\left(\pi_0 + \pi_{\mathcal{P}} e^{-\lambda}\right)^{Z_0} \left(\pi_1 + \pi_{\mathcal{P}} \lambda e^{-\lambda}\right)^{Z_1} \times \cdots \\ &\times \left(\pi_c + \pi_{\mathcal{P}} \frac{\lambda^c e^{-\lambda}}{c!}\right)^{Z_c} \left(\pi_{\mathcal{P}} \frac{\lambda^y e^{-\lambda}}{y!}\right)^{Z_{\mathcal{P}}}. \end{aligned} \tag{28}$$

Giles (2007) presents a model with a multinomial logistic form. The MIP can then be written in terms of a joint GLM with systematic, random, and link components. This is shown as

$$\begin{aligned} \log(\boldsymbol{\lambda}) &= \mathbf{B}\boldsymbol{\beta}, \\ \log\left(\frac{\boldsymbol{\pi}_0}{\boldsymbol{\pi}_{\mathcal{P}}}\right) &= \mathbf{G}_0\boldsymbol{\gamma}_0, \\ &\vdots \\ \log\left(\frac{\boldsymbol{\pi}_c}{\boldsymbol{\pi}_{\mathcal{P}}}\right) &= \mathbf{G}_c\boldsymbol{\gamma}_c, \end{aligned}$$

where $\mathbf{B}$ is an $n \times p$ covariate matrix, $\mathbf{G}_j$ are $n \times q_j$ covariate matrices, $\boldsymbol{\beta}$ is a $p \times 1$ vector of parameters, $\boldsymbol{\gamma}_j$ are $q_j \times 1$ vectors of parameters, $\boldsymbol{\lambda}$ is an $n \times 1$ rate of occurrence vector, and $\frac{\boldsymbol{\pi}_j}{\boldsymbol{\pi}_{\mathcal{P}}}$ are $n \times 1$ vectors of ratios of the probabilities of the i^{th} observation being in the j^{th} count group to being in the Poisson group (for $j = 0, \ldots, c$). Parameter estimates are constructed using a maximum likelihood approach (Giles, 2007).

Giles (2007) defines ω as

$$\omega_{ij} = \frac{e^{G_i'\gamma_j}}{1 + \sum\limits_{l=0}^{J-1} e^{G_i'\gamma_l}}, \tag{29}$$

for $j = 0, 1, \ldots, J$, with $\gamma_J = 0$ imposed. Using this, Giles (2007) defines the log-likelihood function based on a sample of n independent observations as

$$\begin{aligned} L(\beta,\gamma_1,\gamma_2,\ldots,\gamma_J) = & \sum_{y_i \in R_0} \log\left(\omega_{i0} + \left(1 - \sum_{l=0}^{J-1} \omega_{il}\right) P_i\right) \\ & + \sum_{y_i \in R_1} \log\left(\omega_{i1} + \left(1 - \sum_{l=0}^{J-1} \omega_{il}\right) P_i\right) \\ & + \ldots \\ & + \sum_{y_i \in R_{J-1}} \log\left(\omega_{iJ-1} + \left(1 - \sum_{l=0}^{J-1} \omega_{il}\right) P_i\right) \\ & + \sum_{y_i \in R_J} \log\left(\left(1 - \sum_{l=0}^{J-1} \omega_{il}\right) P_i\right), \end{aligned} \tag{30}$$

where

$$P_i = \frac{e^{-\lambda_i}\lambda_i^{y_i}}{y_i!}, \tag{31}$$

is the Poisson probability. Estimation for the MIP follows the previously discussed maximum likelihood process.

Though this model describes zero-to-$\mathcal{K}$ inflated count situations, it does not allow for relationships among related counts. This model is also computationally intensive at higher levels of $\mathcal{K}$ due to the number of parameters being estimated.

Finite Mixture Models

The excess zero models discussed above can be classified as versions of a finite mixture model (FMM). Finite mixture models can be used in a wide variety of situations and circumstances, and are gaining increasing traction in the statistical literature (Everitt & Hand, 1981; McLachlan & Peel, 2000). From a broad perspective, FMM are a weighted combination of multiple density functions or classes.

Density Function

In general, finite mixture model densities may be expressed as

$$f(\mathbf{y}) = \sum_{i=1}^{g} \pi_i f_i(\mathbf{y}|\boldsymbol{\theta}_i), \tag{32}$$

where $\mathbf{y}$ are the $n \times 1$ vector of observed responses, f_i is the conditional probability density function for the observed response in the i^{th} class, π_i is the probability for the i^{th} class, and $\boldsymbol{\theta}_i$ are the vector of parameters for each class. The π_i's are subject to the further constraint that they are between zero and one, and sum to one; that is,

$$0 \leq \pi_i \leq 1 \qquad (i = 1, \ldots, g) \tag{33}$$

and

$$\sum_{i=1}^{g} \pi_i = 1. \tag{34}$$

The quantities $\pi_1, \ldots, \pi_g$ are called the mixing proportions or weights (McLachlan & Peel, 2000).

Estimation

Since the discovery of the EM algorithm, almost all papers that utilize ML estimation of FMM do so via the EM algorithm (McLachlan & Peel, 2000). Details of the utilization of the EM algorithm for FMM follow closely with its example provided for the ZIP.

Identifiability

An important concept in FMM is that of identifiability. In many ways, the estimation and interpretation of the parameters based on the observations are only meaningful if the parameters are identifiable. In general, identifiability indicates if a parameter can be estimated from experimental data (Raue & Timmer, 2013). As FMM uses multiple component densities, identifiability can be impacted simply by changing component labels (the "label-switching problem"). The lack of identifiability is generally handled by imposing a constraint on the parameters (say, that the mixing proportions are in a specified order). Another approach is to avoid restriction and simply report the result of one possible arrangement of the parameters. Using this approach, McLachlan and Peel (2000) state, "this lack of identifiability is not of concern in the normal course of events in the fitting of mixture models by maximum likelihood, say, via the EM algorithm" (p. 27). Though these are potential solutions to the label-switching identifiability issue in FMM, other issues with identifiability may exist even when these solutions are implemented.

Discussions of identifiability for previously used inflation models is scarce in the current literature. Though Li (2012) conducts an initial thorough exploration of identifiability for the ZIP, explorations of the identifiability of the ZOIP, ZKIP, and the MIP models have yet to be conducted.

Zero to K Inflated Data

As presented by Giles (2007), there are many situations involving inflation from zero to $\mathcal{K}$. Additional situations include be the number of detentions received by students, the number of ER visits made by patients, or the number of cigarettes smoked in the last week. Situations such as these clearly have inflation in the lower counts. While the model presented by Giles (2007) accounts for this inflation, it doesn't take into account the relationship that occurs among the inflated values. While Giles uses the multinomial model, the multinomial distribution isn't applied (the only limit on the probabilities is that they sum to one). In this case, models that account for relationships among small count inflation may offer better prediction of counts and group membership. The purpose of this dissertation is to address this gap in the literature by demonstrating the construction of a general zero-to-$\mathcal{K}$ inflated model using FMM. This model is named the small count inflated (SCI) model.

SCI models extend models such as ZI models, ZOI models, ZKI models, and MI models, which address count inflation of inflated counts individually. Building on MI models, the SCI model accounts for inflation by using two (instead of $k+2$) classes, where the first class represents the inflation and the second class represents the count distribution. As an MI model may be viewed as an FMM with $\mathcal{K}+2$ groups ($\mathcal{K}+1$ of which are degenerate), a SCI model may be viewed as an FMM with 2 groups, the first of which is a truncated count distribution designed to capture the small count inflation, the second which is the un-truncated count distribution. The purpose of this dissertation was to address this.

CHAPTER III

METHODS

The multinomially-inflated Poisson (MIP) model is an effective approach for the modeling of count data with multiple points of inflation above that expected by the Poisson distribution. Giles (2007) also demonstrates the necessity and usefulness of such models in applied situations, above and beyond previous models (e.g., Lambert, 1992).

However, due to the independent estimations of counts from zero to $\mathcal{K}$, Giles (2007) method does not place any order on the estimation of counts below $\mathcal{K}$. That is, there is assumed to be no relationship between the levels of inflation at values below $\mathcal{K}$. Additionally, as $\mathcal{K}$ increases, the sample size needed to form multinomial estimates of group membership also increase. This study proposes to add a distributional assumption on counts from zero to $\mathcal{K}$, extending research by Giles (2007) to include parametric estimation and by Lambert (1992) and Lin and Tsai (2013) to include a range of inflation from zero to $\mathcal{K}$.

Small count inflated (SCI) models extend models such as zero-inflated (ZI) models, zero and one inflated (ZOI) models, zero and k inflated (ZKI) models, and multinomially-inflated (MI) models, which address count inflation under the finite mixture model (FMM) framework. Building on MI models, the SCI model accounts for inflation by using two (instead of $\mathcal{K}$) classes, where the first class represents the inflation and the second class represents the count distribution. A MI model may be viewed as an FMM with $\mathcal{K}+2$ groups, $\mathcal{K}+1$ of which are degenerate. In contrast, a SCI model may be viewed as an FMM with 2 groups, the first of which is a truncated count distribution designed to capture the small count inflation, the second which is the un-truncated count distribution.

Research Questions

This study addressed the following questions:

Q1 How can a SCIP distribution be specified?

Q2 How can parameters be estimated for the SCIP distribution?

Q3 How can a SCIP distribution be implemented in `R`?

Q4 In terms of percentage correct and area under the ROC curve, how well will the SCIP distribution predict group membership compared to the zero-inflated and multinomially-inflated Poisson models?

Q5 In terms of mean squared error (MSE), how well will the SCIP distribution predict counts compared to the zero-inflated and multinomially-inflated Poisson models?

This chapter will first present the construction of the general SCI model using a finite mixture model (FMM) framework. The likelihood function will then be discussed, followed by score functions to obtain estimates of parameters and standard errors. Finally, evaluation methods of the proposed model will be discussed.

Small Count Inflation Poisson Model

This section is written to address the first research question. The SCI model is an appropriate model for data involving inflation of “small” counts. Examples may be the number of cigarettes smoked in a week, the number of detentions received by students in the previous year, and the number of times people visit the emergency room in the previous year. Previous studies have considered inflation of such counts on a case by case basis, using multinomial logistic regression to model the inflated counts individually (Giles, 2007). This study extends such methods to allow relationship between the inflated counts.

The SCIP model is a member of the finite mixture model family, and is formed by mixing a truncated and un-truncated Poisson distribution. The truncated Poisson distribution is a right truncated distribution, with the cutoff being chosen based on theory.

This distribution models the inflated nature of the lower counts. The un-truncated Poisson distribution models the larger counts, and overlaps with the truncated distribution at and below the cutoff. The expectation of both distributions is related to the parameters via a log link. Additionally, the mixing proportions represent the probability of an observation coming from a given population, and are described by a binomial distribution. The expectation of this distribution is linked to the parameters via a logit link function.

Following the form of a finite mixture distribution shown in Equation 32, the probability mass function (PMF) for the SCIP is readily described in answer to research question one. Let $\pi_{\mathcal{S}}$ represent the probability of an observation originating from the truncated Poisson group, and let $\pi_{\mathcal{L}}$ be the probability of an observation originating from the un-truncated Poisson group. Note that membership in these groups is mutually exclusive. That is, $\pi_{\mathcal{S}} + \pi_{\mathcal{L}} = 1$. The random variable Y is said to follow a SCIP distribution, denoted $Y \sim SCIP(\pi_{\mathcal{S}}, \lambda_{\mathcal{P}}, \lambda_{\mathcal{TP}})$, if its PMF is given by

$$f(y|\pi_0, \lambda) = \begin{cases} \pi_{\mathcal{S}} \left(\dfrac{e^{-\lambda_{\mathcal{S}}} \lambda_{\mathcal{S}}^{y}}{y!} \left(\displaystyle\sum_{k=0}^{c} \dfrac{e^{-\lambda_{\mathcal{S}}} \lambda_{\mathcal{S}}^{k}}{k!} \right)^{-1} \right) & \\ + (1 - \pi_{\mathcal{S}}) \left(\dfrac{e^{-\lambda_{\mathcal{L}}} \lambda_{\mathcal{L}}^{y}}{y!} \right), & y \leq c \\ (1 - \pi_{\mathcal{S}}) \left(\dfrac{e^{-\lambda_{\mathcal{L}}} \lambda_{\mathcal{L}}^{y}}{y!} \right), & y > c, \end{cases} \tag{35}$$

or

$$\begin{aligned} f(y|\pi_{\mathcal{S}}, \lambda_{\mathcal{S}}, \lambda_{\mathcal{L}}) = I_{y \leq c}\ \pi_{\mathcal{S}} & \left(\frac{e^{-\lambda_{\mathcal{S}}} \lambda_{\mathcal{S}}^{y}}{y!} \left(\sum_{k=0}^{c} \frac{e^{-\lambda_{\mathcal{S}}} \lambda_{\mathcal{S}}^{k}}{k!} \right)^{-1} \right) \\ & + (1 - \pi_{\mathcal{S}}) \left(\frac{e^{-\lambda_{\mathcal{L}}} \lambda_{\mathcal{L}}^{y}}{y!} \right), \end{aligned} \tag{36}$$

where y is the count, $\lambda_{\mathcal{S}}$ is the rate of occurance for the truncated Poisson, $\lambda_{\mathcal{L}}$ is the rate of occurance for the un-truncated Poisson, c is the truncation value (with the truncated distribution being defined for integers from 0 to c), and $I_{y \leq c}$ is an indicator function (one when $y \leq c$, and zero otherwise).

Using Equation 35, the PMF for n independent samples is

$$\prod_{i=1}^{n} f(y_i | \pi_{\mathcal{S}}, \lambda_{\mathcal{S}}, \lambda_{\mathcal{L}}) \tag{37}$$

for observations $y_1, y_2, \ldots, y_n$. The SCIP model can be written in terms of a joint generalized linear model (GLM), and can be expressed as

$$\begin{aligned} \log(\boldsymbol{\lambda}_{\mathcal{S}}) &= \mathbf{S}\text{ℨ}, \\ \log(\boldsymbol{\lambda}_{\mathcal{L}}) &= \mathbf{K}\text{ϟ}, \\ \text{logit}(\boldsymbol{\pi}_{\mathcal{S}}) &= \mathbf{G}\boldsymbol{\gamma}, \end{aligned} \tag{38}$$

or

$$\begin{aligned} \boldsymbol{\lambda}_{\mathcal{S}} &= e^{\mathbf{S}\text{ℨ}}, \\ \boldsymbol{\lambda}_{\mathcal{L}} &= e^{\mathbf{K}\text{ϟ}}, \\ \boldsymbol{\pi}_{\mathcal{S}} &= \frac{e^{\mathbf{G}\boldsymbol{\gamma}}}{1 + e^{\mathbf{G}\boldsymbol{\gamma}}}, \end{aligned} \tag{39}$$

where $\mathbf{S}$ is an $n \times p$ covariate matrix, $\mathbf{K}$ is an $n \times q$ covariate matrix, $\mathbf{G}$ is an $n \times r$ covariate matrix, ℨ is a $p \times 1$ vector of parameters, ϟ is a $q \times 1$ vector of parameters, $\boldsymbol{\gamma}$ is an $r \times 1$ vector of parameters, $\boldsymbol{\lambda}_{\mathcal{S}}$ is an $n \times 1$ rate of occurrence vector for the truncated Poisson, $\boldsymbol{\lambda}_{\mathcal{L}}$ is an $n \times 1$ rate of occurrence vector for the un-truncated Poisson, and $\boldsymbol{\pi}_{\mathcal{S}}$ is an $n \times 1$ vector of probabilities that the observation is in the truncated Poisson group.

In this case, the parameter vectors ℨ, ϟ, and $\boldsymbol{\gamma}$ take on separate interpretations. The $r \times 1$ $\boldsymbol{\gamma}$ parameter vector relates to the probability that a case will fall into the truncated Poisson group, the $p \times 1$ ℨ parameter vector relates to the truncated Poisson count group conditional on membership in the truncated Poisson group, and the $q \times 1$ ϟ parameter vector relates to the un-truncated Poisson count group conditional on membership in the un-truncated Poisson group. The $e^{\boldsymbol{\gamma}}$ vector can be interpreted as odds ratios, while the $e^{\boldsymbol{\beta}}$ vector can be interpreted as the multiplicative change in expected counts for the truncated Poisson group and the $e^{\boldsymbol{\zeta}}$ vector can be interpreted as the multiplicative change in the expected counts for the un-truncated Poisson group.

Parameter Estimation

This section is written to address the second research question. Parameter estimates of the SCIP are obtained using the maximum likelihood estimation (MLE) procedure via the expectation-maximization (EM) algorithm in answer to research question two. As before, MLE uses the likelihood function to find parameter estimates and standard errors. Parameters are found by solving the first derivative of the likelihood function for zero, while standard errors are found by evaluating the second derivative of the likelihood function.

Likelihood Function

The likelihood function for the SCIP is shown in Equation 37. Using this equation as a likelihood, however, we reverse the conditions to condition the parameters on the data. That is, we set

$$f_i(y_i|\pi_{\mathcal{S}},\lambda_{\mathcal{S}},\lambda_{\mathcal{L}}) = \ell_i(\pi_{\mathcal{S}},\lambda_{\mathcal{S}},\lambda_{\mathcal{L}}|y_i) \tag{40}$$

and take the likelihood to be

$$\ell = \prod_{i=1}^{n} \ell_i(\pi_{\mathcal{S}},\lambda_{\mathcal{S}},\lambda_{\mathcal{L}}|y_i). \tag{41}$$

First Derivatives

For simplification of the process of taking the first derivatives, we will first consider the derivative of a generic FMM as shown by Shalizi (2017), matching notation by replacing ℓ_i with f. For a generic FMM, the likelihood can be expressed as

$$\ell = \prod_{i=1}^{n} f(\boldsymbol{\theta}|y_i), \tag{42}$$

where $\boldsymbol{\theta}$ is a vector of the mixing weights and distributional parameters and y_i is the observed outcome for the i^{th} participant. Taking the logarithm of the likelihood, we find

$$\begin{aligned} L &= \sum_{i=1}^{n} \log\left(f(\boldsymbol{\theta}|y_i)\right) \\ &= \sum_{i=1}^{n} \log\left(\sum_{k=1}^{K} \pi_k f_k(\boldsymbol{\theta}_k|y_i)\right), \end{aligned} \tag{43}$$

where π_k is the mixing weight of the k^{th} distribution, f_k is the k^{th} PDF, $\boldsymbol{\theta}_k$ is the vector of parameters for the k^{th} distribution, and K is the total number of distribution in the FMM. Taking the derivative of Equation 43 with respect to a generic parameter vector $\boldsymbol{\theta}_j$, we find

$$\begin{aligned} \frac{\partial L}{\partial \boldsymbol{\theta}_j} &= \sum_{i=1}^{n} \frac{1}{\sum_{k=1}^{K} \pi_k f_k(\boldsymbol{\theta}_k|y_i)} \pi_j \frac{\partial f_j(\boldsymbol{\theta}_j|y_i)}{\partial \boldsymbol{\theta}_j} \\ &= \sum_{i=1}^{n} \frac{\pi_j f_j(\boldsymbol{\theta}_j|y_i)}{\sum_{k=1}^{K} \pi_k f_k(\boldsymbol{\theta}_k|y_i)} \frac{1}{f_j(\boldsymbol{\theta}_j|y_i)} \frac{\partial f_j(\boldsymbol{\theta}_j|y_i)}{\partial \boldsymbol{\theta}_j} \\ &= \sum_{i=1}^{n} \left(\left(\frac{\pi_j f_j(\boldsymbol{\theta}_j|y_i)}{\sum_{k=1}^{K} \pi_k f_k(\boldsymbol{\theta}_k|y_i)} \right) \left(\frac{\partial \log\left(f_j(\boldsymbol{\theta}_j|y_i)\right)}{\partial \boldsymbol{\theta}_j} \right) \right). \end{aligned} \tag{44}$$

Using this form of the derivative of the FMM, we will now show the derivative of the SCIP. First, pieces of the derivatives will be shown. Then, the derivatives will be given with respect to the three sets of parameters (ϡ, ϟ, and $\boldsymbol{\gamma}$).

Derivative of the logarithm of the Poisson distribution. Let $f_{\mathcal{P}}$ be a Poisson distribution with λ as the rate parameter and y as the count. The logarithm $f_{\mathcal{P}}$ can be simplified as follows:

$$\begin{aligned} \log\left(f_{\mathcal{P}}\right) &= \log\left(\frac{e^{-\lambda}\lambda^{y}}{y!}\right) \\ &= \log\left(e^{-\lambda}\right) + \log\left(\lambda^{y}\right) - \log\left(y!\right) \\ &= -\lambda + y\log(\lambda) - \log(y!). \end{aligned} \tag{45}$$

Taking the derivative of this equation with respect to λ, it can be seen that

$$\begin{aligned}\frac{\partial \log\left(f_{\mathcal{P}}\right)}{\partial \lambda} &= -1 + \frac{y}{\lambda} - 0 \\ &= \frac{y-\lambda}{\lambda}.\end{aligned} \tag{46}$$

Derivative of the logarithm of the truncated Poisson distribution. Let $f_{\mathcal{TP}}$ be a right truncated Poisson distribution with λ as the rate parameter, y as the count, and include the truncation value denoted c. The logarithm of $f_{\mathcal{TP}}$ can be simplified as follows:

$$\begin{aligned}\log\left(f_{\mathcal{TP}}\right) &= \log\left(\frac{e^{-\lambda}\lambda^{y}}{y!}\left(\sum_{k=0}^{c}\frac{e^{-\lambda}\lambda^{k}}{k!}\right)^{-1}\right) \\ &= \log\left(e^{-\lambda}\right) + \log\left(\lambda^{y}\right) - \log\left(y!\right) - \log\left(\sum_{k=0}^{c}\frac{e^{-\lambda}\lambda^{k}}{k!}\right) \\ &= -\lambda + y\log(\lambda) - \log(y!) - \log\left(\sum_{k=0}^{c}\frac{e^{-\lambda}\lambda^{k}}{k!}\right).\end{aligned} \tag{47}$$

Taking the derivative of this equation with respect to λ, it can be seen that

$$\begin{aligned}\frac{\partial \log(f_{\mathcal{TP}})}{\partial \lambda} &= \frac{y-\lambda}{\lambda} - \frac{1}{\sum_{k=0}^{c}\frac{e^{-\lambda}\lambda^{k}}{k!}}\sum_{k=0}^{c}\frac{(-1)e^{-\lambda}\lambda^{k} + (k)e^{-\lambda}\lambda^{k-1}}{k!} \\ &= \frac{(y-\lambda)\sum_{k=0}^{c}\frac{e^{-\lambda}\lambda^{k}}{k!} - \lambda\left(\sum_{k=0}^{c}\frac{e^{-\lambda}\lambda^{k-1}}{k!}(k-\lambda)\right)}{\lambda\left(\sum_{k=0}^{c}\frac{e^{-\lambda}\lambda^{k}}{k!}\right)} \\ &= \frac{\left(\sum_{k=0}^{c}\left(\frac{e^{-\lambda}\lambda^{k}}{k!}(y-\lambda)\right)\right) - \left(\sum_{k=0}^{c}\left(\frac{e^{-\lambda}\lambda^{k}}{k!}(k-\lambda)\right)\right)}{\lambda\left(\sum_{k=0}^{c}\frac{e^{-\lambda}\lambda^{k}}{k!}\right)} \\ &= \frac{\sum_{k=0}^{c}\left(\frac{e^{-\lambda}\lambda^{k}}{k!}\left((y-\lambda)-(k-\lambda)\right)\right)}{\lambda\left(\sum_{k=0}^{c}\frac{e^{-\lambda}\lambda^{k}}{k!}\right)} \\ &= \frac{\sum_{k=0}^{c}\frac{e^{-\lambda}\lambda^{k}}{k!}(y-k)}{\lambda\left(\sum_{k=0}^{c}\frac{e^{-\lambda}\lambda^{k}}{k!}\right)}.\end{aligned} \tag{48}$$

Derivatives of the log-likelihood of the mixture model. Using the above derivations and the general form of derivatives for FMMs shown in Equation 44, the derivatives with respect to the parameters of the SCIP will now be shown. The derivative of the SCIP with respect to $\lambda_{\mathcal{S}}$ for the i^{th} observation is found for $y_i \leq c$, and takes the form

$$\begin{aligned}\frac{\partial L}{\partial \lambda_{\mathcal{S}_i}} = &\left(\frac{\pi_{\mathcal{S}_i}\left(\frac{e^{-\lambda_{\mathcal{S}_i}}\lambda_{\mathcal{S}_i}^{y_i}}{y_i!}\left(\sum_{k=0}^{c}\frac{e^{-\lambda_{\mathcal{S}_i}}\lambda_{\mathcal{S}_i}^{k}}{k!}\right)^{-1}\right)}{\pi_{\mathcal{S}_i}\left(\frac{e^{-\lambda_{\mathcal{S}_i}}\lambda_{\mathcal{S}_i}^{y_i}}{y_i!}\left(\sum_{k=0}^{c}\frac{e^{-\lambda_{\mathcal{S}_i}}\lambda_{\mathcal{S}_i}^{k}}{k!}\right)^{-1}\right)+(1-\pi_{\mathcal{S}_i})\left(\frac{e^{-\lambda_{\mathcal{L}_i}}\lambda_{\mathcal{L}_i}^{y_i}}{y_i!}\right)} \right) \\ &\times \left(\frac{\sum_{k=0}^{c}\frac{e^{-\lambda_{\mathcal{S}_i}}\lambda_{\mathcal{S}_i}^{k}}{k!}(y_i-k)}{\lambda_{\mathcal{S}_i}\left(\sum_{k=0}^{c}\frac{e^{-\lambda_{\mathcal{S}_i}}\lambda_{\mathcal{S}_i}^{k}}{k!}\right)} \right).\end{aligned} \tag{49}$$

The derivative of the SCIP with respect to $\lambda_{\mathcal{L}}$ for the i^{th} observation takes the form

$$\begin{aligned}\frac{\partial L}{\partial \lambda_{\mathcal{L}_i}} = &\left(\frac{(1-\pi_{\mathcal{S}_i})\left(\frac{e^{-\lambda_{\mathcal{L}_i}}\lambda_{\mathcal{L}_i}^{y_i}}{y_i!}\right)}{\pi_{\mathcal{S}_i}\left(\frac{e^{-\lambda_{\mathcal{S}_i}}\lambda_{\mathcal{S}_i}^{y_i}}{y_i!}\left(\sum_{k=0}^{c}\frac{e^{-\lambda_{\mathcal{S}_i}}\lambda_{\mathcal{S}_i}^{k}}{k!}\right)^{-1}\right)+(1-\pi_{\mathcal{S}_i})\left(\frac{e^{-\lambda_{\mathcal{L}_i}}\lambda_{\mathcal{L}_i}^{y_i}}{y_i!}\right)} \right) \\ &\times \left(\frac{y_i-\lambda_{\mathcal{L}_i}}{\lambda_{\mathcal{L}_i}}\right).\end{aligned} \tag{50}$$

The derivative of the SCIP with respect to $\pi_{\mathcal{S}}$ for the i^{th} observation is based Equation 43 instead of Equation 44, and takes the form

$$\frac{\partial L}{\partial \pi_{\mathcal{S}_i}} = \frac{\left(\frac{e^{-\lambda_{\mathcal{S}_i}}\lambda_{\mathcal{S}_i}^{y_i}}{y_i!}\left(\sum_{k=0}^{c}\frac{e^{-\lambda_{\mathcal{S}_i}}\lambda_{\mathcal{S}_i}^{k}}{k!}\right)^{-1}\right) - \left(\frac{e^{-\lambda_{\mathcal{L}_i}}\lambda_{\mathcal{L}_i}^{y_i}}{y_i!}\right)}{\pi_{\mathcal{S}_i}\left(\frac{e^{-\lambda_{\mathcal{S}_i}}\lambda_{\mathcal{S}_i}^{y_i}}{y_i!}\left(\sum_{k=0}^{c}\frac{e^{-\lambda_{\mathcal{S}_i}}\lambda_{\mathcal{S}_i}^{k}}{k!}\right)^{-1}\right) + (1-\pi_{\mathcal{S}_i})\left(\frac{e^{-\lambda_{\mathcal{L}_i}}\lambda_{\mathcal{L}_i}^{y_i}}{y_i!}\right)}. \tag{51}$$

These derivatives are taken with respect to the shape parameter of the three distributions. However, Equation 39 shows these parameters to be functions of a set of covariates and a vector of parameters. Derivatives with respect to these vectors of parameters can be easily calculated via the chain rule, and will now be shown. The derivative of the SCIP with respect to [illegible] for the i^{th} observation is

$$
\begin{aligned}
\frac{\partial L}{\partial \boldsymbol{\sampi}} =& \frac{\partial L}{\partial \lambda_{\mathcal{S}_i}} \frac{\partial \lambda_{\mathcal{S}_i}}{\partial \boldsymbol{\sampi}} \\
=& \frac{\partial L}{\partial \lambda_{\mathcal{S}_i}} \mathbf{S}_i e^{\mathbf{S}_i \boldsymbol{\sampi}} \\
=& \frac{\partial L}{\partial \lambda_{\mathcal{S}_i}} \mathbf{S}_i \lambda_{\mathcal{S}_i} \\
=& \left(\frac{\pi_{\mathcal{S}_i} \left(\frac{e^{-\lambda_{\mathcal{S}_i}} \lambda_{\mathcal{S}_i}^{y_i}}{y_i!} \left(\sum_{k=0}^{c} \frac{e^{-\lambda_{\mathcal{S}_i}} \lambda_{\mathcal{S}_i}^{k}}{k!} \right)^{-1} \right)}{\pi_{\mathcal{S}_i} \left(\frac{e^{-\lambda_{\mathcal{S}_i}} \lambda_{\mathcal{S}_i}^{y_i}}{y_i!} \left(\sum_{k=0}^{c} \frac{e^{-\lambda_{\mathcal{S}_i}} \lambda_{\mathcal{S}_i}^{k}}{k!} \right)^{-1} \right) + (1 - \pi_{\mathcal{S}_i}) \left(\frac{e^{-\lambda_{\mathcal{L}_i}} \lambda_{\mathcal{L}_i}^{y_i}}{y_i!} \right)} \right) \\
& \times \left(\frac{\sum_{k=0}^{c} \frac{e^{-\lambda_{\mathcal{S}_i}} \lambda_{\mathcal{S}_i}^{k}}{k!} (y_i - k)}{\lambda_{\mathcal{S}_i} \left(\sum_{k=0}^{c} \frac{e^{-\lambda_{\mathcal{S}_i}} \lambda_{\mathcal{S}_i}^{k}}{k!} \right)} \right) \mathbf{S}_i \lambda_{\mathcal{S}_i},
\end{aligned}
\tag{52}
$$

where $\mathbf{S}_i$ is the i^{th} row of the design matrix $\mathbf{S}$.

Similarly, the derivative of the SCIP with respect to $\boldsymbol{\koppa}$ for the i^{th} observation is

$$
\begin{aligned}
\frac{\partial L}{\partial \boldsymbol{\koppa}} =& \frac{\partial L}{\partial \lambda_{\mathcal{L}_i}} \frac{\partial \lambda_{\mathcal{L}_i}}{\partial \boldsymbol{\koppa}} \\
=& \frac{\partial L}{\partial \lambda_{\mathcal{L}_i}} \mathbf{K}_i e^{\mathbf{K}_i \boldsymbol{\koppa}} \\
=& \frac{\partial L}{\partial \lambda_{\mathcal{L}_i}} \mathbf{K}_i \lambda_{\mathcal{L}_i} \\
=& \left(\frac{(1 - \pi_{\mathcal{S}_i}) \left(\frac{e^{-\lambda_{\mathcal{L}_i}} \lambda_{\mathcal{L}_i}^{y_i}}{y_i!} \right)}{\pi_{\mathcal{S}_i} \left(\frac{e^{-\lambda_{\mathcal{S}_i}} \lambda_{\mathcal{S}_i}^{y_i}}{y_i!} \left(\sum_{k=0}^{c} \frac{e^{-\lambda_{\mathcal{S}_i}} \lambda_{\mathcal{S}_i}^{k}}{k!} \right)^{-1} \right) + (1 - \pi_{\mathcal{S}_i}) \left(\frac{e^{-\lambda_{\mathcal{L}_i}} \lambda_{\mathcal{L}_i}^{y_i}}{y_i!} \right)} \right) \\
& \times \left(\frac{y_i - \lambda_{\mathcal{L}_i}}{\lambda_{\mathcal{L}_i}} \right) \mathbf{K}_i \lambda_{\mathcal{L}_i},
\end{aligned}
\tag{53}
$$

where $\mathbf{K}_i$ is the i^{th} row of the design matrix $\mathbf{K}$.

Finally, the derivative of the SCIP with respect to $\boldsymbol{\gamma}$ for the i^{th} observation is

$$
\begin{aligned}
\frac{\partial L}{\partial \boldsymbol{\gamma}} =& \frac{\partial L}{\partial \pi_{\mathcal{S}_i}} \frac{\partial \pi_{\mathcal{S}_i}}{\partial \boldsymbol{\gamma}} \\
=& \frac{\partial L}{\partial \pi_{\mathcal{S}_i}} \frac{\left(\mathbf{G}_i e^{\mathbf{G}_i \boldsymbol{\gamma}}\right)\left(1 + e^{\mathbf{G}_i \boldsymbol{\gamma}}\right) - \left(e^{\mathbf{G}_i \boldsymbol{\gamma}}\right)\left(\mathbf{G}_i e^{\mathbf{G}_i \boldsymbol{\gamma}}\right)}{\left(1 + e^{\mathbf{G}_i \boldsymbol{\gamma}}\right)^2} \\
=& \frac{\partial L}{\partial \pi_{\mathcal{S}_i}} \frac{\mathbf{G}_i e^{\mathbf{G}_i \boldsymbol{\gamma}}}{\left(1 + e^{\mathbf{G}_i \boldsymbol{\gamma}}\right)^2} \\
=& \frac{\partial L}{\partial \pi_{\mathcal{S}_i}} \mathbf{G}_i \pi_{\mathcal{S}_i}\left(1 - \pi_{\mathcal{S}_i}\right) \\
=& \left(\frac{\left(\frac{e^{-\lambda_{\mathcal{S}_i}} \lambda_{\mathcal{S}_i}^{y_i}}{y_i!} \left(\sum_{k=0}^{c} \frac{e^{-\lambda_{\mathcal{S}_i}} \lambda_{\mathcal{S}_i}^{k}}{k!} \right)^{-1} \right) - \left(\frac{e^{-\lambda_{\mathcal{L}_i}} \lambda_{\mathcal{L}_i}^{y_i}}{y_i!} \right)}{\pi_{\mathcal{S}_i} \left(\frac{e^{-\lambda_{\mathcal{S}_i}} \lambda_{\mathcal{S}_i}^{y_i}}{y_i!} \left(\sum_{k=0}^{c} \frac{e^{-\lambda_{\mathcal{S}_i}} \lambda_{\mathcal{S}_i}^{k}}{k!} \right)^{-1} \right) + \left(1 - \pi_{\mathcal{S}_i}\right) \left(\frac{e^{-\lambda_{\mathcal{L}_i}} \lambda_{\mathcal{L}_i}^{y_i}}{y_i!} \right)} \right) \\
& \times \mathbf{G}_i \pi_{\mathcal{S}_i}\left(1 - \pi_{\mathcal{S}_i}\right),
\end{aligned}
\tag{54}
$$

where $\mathbf{G}_i$ is the i^{th} row of the design matrix $\mathbf{G}$.

These equations are the estimating equations used to find parameter estimates. Parameter estimates are found by setting these equal to zero and solving. In equation form, this is shown as

$$
\begin{aligned}
\frac{\partial L}{\partial \boldsymbol{\beta}} =& \mathbf{0}, \\
\frac{\partial L}{\partial \boldsymbol{\delta}} =& \mathbf{0}, \\
\frac{\partial L}{\partial \boldsymbol{\gamma}} =& \mathbf{0}.
\end{aligned}
$$

Note that the derivatives with respect to each of the parameters are not independent. That is, each of the derivatives with respect to one parameter includes all of the other parameters.

Expectation Maximization Implementation

This section discusses the EM implementation of the SCIP. Note that a similar discussion is also provided in Chapter IV to address the third research question. Implementation of the EM algorithm for the SCIP follows the same principles as EM implementation for the ZIP. A Poisson model was used to determine initial parameter estimates for the small and large count components, while logistic regression (with the outcome being defined as 1 if $y_i < c$ and 0 otherwise) was used to determine initial parameter estimates for the logistic component. Given these parameters, initial probabilities of group membership were calculated. Using initial values, the process alternated between estimating group membership and estimating parameters, using the most updated estimates of each at each step. The process iterated until the change in parameters from one iteration to the next met convergence criterion (all changes were less than 10^{-9}) or met the maximum number of iterations (100). First, let

$$\boldsymbol{\Psi} = \begin{bmatrix} \text{[illegible]} \\ \\ \text{[illegible]} \\ \\ \boldsymbol{\gamma} \end{bmatrix}. \tag{55}$$

Following the algorithm described by McLachlan and Peel (2000), the k^{th} iteration of the algorithm uses

$$\hat{\tau}_{\mathcal{S}_i}^{(k+1)}(y_i, \hat{\boldsymbol{\Psi}}^{(k)}) = \frac{\hat{\pi}_{\mathcal{S}_i}^{(k)} f_{\mathcal{TP}}(y_i, \hat{\lambda}_{\mathcal{S}_i}^{(k)})}{\hat{\pi}_{\mathcal{S}_i}^{(k)} f_{\mathcal{TP}}(y_i, \hat{\lambda}_{\mathcal{S}_i}^{(k)}) + (1 - \hat{\pi}_{\mathcal{S}_i}^{(k)}) f_{\mathcal{P}}(y_i, \hat{\lambda}_{\mathcal{L}_i}^{(k)})} \tag{56}$$

as the next estimate of group membership in the small count inflated group, where $\hat{\tau}_{\mathcal{S}_i}^{(k+1)}(y_i, \hat{\boldsymbol{\Psi}}^{(k)})$ is the function of updated estimates of group membership given the i^{th} observed count (y_i) and the estimated parameters from the current iteration ($\hat{\boldsymbol{\Psi}}^{(k)}$), $\hat{\pi}_{\mathcal{S}_i}^{(k)}$ is

i^{th} element of the current estimated population proportion of the small count inflated group, $f_{\mathcal{TP}}(y_i, \hat{\lambda}_{\mathcal{S}_i}^{(k)})$ is the evaluation of the i^{th} element of the truncated Poisson given the observed count and the current estimate of its mean, and $f_{\mathcal{P}}(y_i, \hat{\lambda}_{\mathcal{L}_i}^{(k)})$ is the evaluation of the i^{th} element of the Poisson given the observed count and the current estimate of its mean. Similarly, McLachlan and Peel (2000) globally maximize

$$\begin{aligned} Q(\boldsymbol{\Psi}, \hat{\boldsymbol{\Psi}}^{(k)}) = \sum_{i=1}^{n} \Big(&\hat{\tau}_{\mathcal{S}_i}^{(k+1)}(y_i, \hat{\boldsymbol{\Psi}}^{(k)}) \log(\pi_{\mathcal{S}_i} f_{\mathcal{TP}}(y_i, \lambda_{\mathcal{S}_i})) + \\ &(1 - \hat{\tau}_{\mathcal{S}_i}^{(k+1)}(y_i, \hat{\boldsymbol{\Psi}}^{(k)})) \log((1 - \pi_{\mathcal{S}_i}) f_{\mathcal{P}}(y_i, \lambda_{\mathcal{L}_i})) \Big) \end{aligned} \tag{57}$$

with respect to $\boldsymbol{\Psi}$ to give the updated estimate $\boldsymbol{\Psi}^{(k+1)}$. Note that because $\boldsymbol{\Psi}$ is made up of $\boldsymbol{\beta}$, $\boldsymbol{\delta}$, and $\boldsymbol{\gamma}$ which define $\pi_{\mathcal{S}_i}$, $\lambda_{\mathcal{S}_i}$, and $\lambda_{\mathcal{L}_i}$, the means in Equation 70 are not the k^{th} iteration estimates because Q is being maximized with respect to them in order to determine their estimates.

Hypothesis Testing

Standard errors of the parameter estimates are found by first taking the second derivative of the likelihood function to find the information matrix. The expected value of the information matrix is used, and in practice parameter estimates are used in place of unknown parameters. The standard errors are the square roots of the diagonal elements of the inverse information matrix (Agresti, 2002). In equation form, if we let

$$\boldsymbol{\Psi} = \begin{bmatrix} \boldsymbol{\beta} \\ \boldsymbol{\delta} \\ \boldsymbol{\gamma} \end{bmatrix}, \tag{58}$$

the inverse information matrix is given as

$$Cov(\boldsymbol{\Psi}) = \left(-E\left[\frac{\partial^2 L}{\partial \boldsymbol{\Psi} \partial \boldsymbol{\Psi}'} \right] \right)^{-1}, \tag{59}$$

while the empirical inverse information matrix can be given as

$$\hat{Cov}\left(\hat{\mathbf{\Psi}}\right) = \left(-E\left[\frac{\partial^2 L}{\partial\mathbf{\Psi}\partial\mathbf{\Psi}'}\bigg|_{\hat{\mathbf{\Psi}}}\right]\right)^{-1}. \tag{60}$$

So long as regularity conditions are met, ML parameter estimates and standard errors are used to perform hypothesis testing, with the Wald ratio being compared to a standard normal distribution. That is, with a nonnull standard error (denoted SE) of $\hat{\beta}$, the test statistic

$$z = \frac{(\hat{\beta} - \beta_0)}{SE} \tag{61}$$

has an approximate normal distribution when $\beta = \beta_0$, which may be compared to a standard normal table. For a two-sided alternative to the standard normal, z^2 has a chi-squared distribution with one degree of freedom. This type of statistic is called a Wald statistic (Agresti, 2002; Wald, 1943), and is used as the primary means of hypothesis testing for the SCIP.

Model Comparisons

The third research question is addressed via `R` code alluded to in this section, and provided in Appendix B. A live version of this code is currently available as a package, and includes all code used for models and the simulation shown in this dissertation. It is available at https://github.com/flor3652/BigD. The package can be directly installed to `R` by following directions included in the read-me file.

This section is written to address the fourth and fifth research questions. In Chapter IV, simulated and observed data (available at http://tiny.cc/cdcdata1) are analyzed. The observed dataset contains health data from the Centers for Disease Control and Prevention (CDC) Behavioral Risk Factor Surveillance System (BRFSS). The data are collected via phone, with more than 400,000 adults being interviewed each year. The 2016 dataset will be used in this study. This dataset includes variables on behaviors such as drinking, smoking, and overall health. Performance in the analysis of simulated data based on this dataset will be compared between the SCIP, zero-inflated Poisson (ZIP), and multinomially-inflated Poisson (MIP) models to address research questions four and five.

The ZIP model constructed by Lambert (1992) is a common solution for modeling data that have inflation at zero. This model splits the observations into a degenerate population (at zero) and a count population, where the count population are allowed to have counts of zero. In this model, however, any count above zero is forced to belong to the count population. The MIP model constructed by Giles (2007) is a more recent solution to the inflated counts problem, and allows inflation in the same counts as the SCIP. The MIP splits observations into multiple degenerate groups (at the points of inflation) and a count population, where the count population can have counts at all of the points of inflation. The MIP model, however, does not specify a relationship between the points of inflation. That is, the order and pattern of the count values in the small population are not considered. In essence, each inflated count is treated as its own distribution, and is given a separate probability of occurrence.

Candidate Models

This section presents the specific models that are compared in both empirical and simulated data situations. The empirical comparison uses the variables as listed, while the simulation generates data which mimics the values observed from the empirical fit. Generated data will mimic the descriptives for each variable (e.g., mean and variance) in addition to the parameter estimates attained from fitting the SCIP model. For these models, the number of days a participant has participated in binge drinking (5 or more drinks B) in the previous month is used as the outcome, and the participant's weight (W) is used as the predictor. The cutoff value of $c = 8$ is used for the models below. The construction of the models is shown below.

Zero-Inflated Model

For the ZIP, the model is shown as

$$\begin{aligned}\log(\lambda_{\mathcal{B}_i}) &= \beta_0 + \beta_1 x_{W_i},\\ \operatorname{logit}(\pi_{0_i}) &= \gamma_0 + \gamma_1 x_{W_i}.\end{aligned} \tag{62}$$

Note that the ZIP model has only two components: the zero inflation component, and the Poisson count component.

Multinomially-Inflated Model

For the MIP, the model is shown as

$$\begin{aligned}\log(\lambda_{\mathcal{B}_i}) &= \beta_0 + \beta_1 x_{W_i},\\ \log\left(\frac{\pi_{0_i}}{\pi_{\mathcal{P}_i}}\right) &= \gamma_{00} + \gamma_{10} x_{W_i},\\ \log\left(\frac{\pi_{1_i}}{\pi_{\mathcal{P}_i}}\right) &= \gamma_{01} + \gamma_{11} x_{W_i},\\ \log\left(\frac{\pi_{2_i}}{\pi_{\mathcal{P}_i}}\right) &= \gamma_{02} + \gamma_{12} x_{W_i},\\ \log\left(\frac{\pi_{3_i}}{\pi_{\mathcal{P}_i}}\right) &= \gamma_{03} + \gamma_{13} x_{W_i},\\ \log\left(\frac{\pi_{4_i}}{\pi_{\mathcal{P}_i}}\right) &= \gamma_{04} + \gamma_{14} x_{W_i},\\ \log\left(\frac{\pi_{5_i}}{\pi_{\mathcal{P}_i}}\right) &= \gamma_{05} + \gamma_{15} x_{W_i},\\ \log\left(\frac{\pi_{6_i}}{\pi_{\mathcal{P}_i}}\right) &= \gamma_{06} + \gamma_{16} x_{W_i},\\ \log\left(\frac{\pi_{7_i}}{\pi_{\mathcal{P}_i}}\right) &= \gamma_{07} + \gamma_{17} x_{W_i},\\ \log\left(\frac{\pi_{8_i}}{\pi_{\mathcal{P}_i}}\right) &= \gamma_{08} + \gamma_{18} x_{W_i}.\end{aligned} \tag{63}$$

Note that the MIP has $8+2=10$ components: a component for each of the inflated counts from one to eight, a component for the probability of a Poisson, and a component for the Poisson count.

Small Count Inflated Model

For the SCIP, the model is shown as

$$\begin{aligned} \log(\lambda_{\mathcal{SB}_i}) &= \text{ϡ}_0 + \text{ϡ}_1 x_{W_i}, \\ \log(\lambda_{\mathcal{LB}_i}) &= \text{ϟ}_0 + \text{ϟ}_1 x_{W_i}, \\ \operatorname{logit}(\pi_{\mathcal{SB}_i}) &= \gamma_0 + \gamma_1 x_{W_i}. \end{aligned} \tag{64}$$

Note that the SCIP model has three components: the small count Poisson component, the large count Poisson component, and the mixing weight component.

Comparisons of Group Membership

All three models being compared assign observations to respective groups. It is of interest to know how well each of these models classifies the observation into the correct group. Through the generation of data with known groups, correct classification of group membership can be compared for each of the three models. Curves illustrating the correct classifications, such as ROC curves, are presented. Additionally, area under the ROC curve will also be presented.

Predicted group membership for the ZIP and SCIP will be based on their logistic components. Predicted group membership for the MIP will use the sum of all of the inflated groups probabilities as the estimate for membership in the inflated count group. That is,

$$\boldsymbol{\pi}_{\mathcal{S}} = \sum_{g=0}^{c} \boldsymbol{\pi}_g, \tag{65}$$

where $\boldsymbol{\pi}_{\mathcal{S}}$ is the vector of probabilities of an observation being in the small count inflated group and c is the cutoff of inflation. This formulation of $\boldsymbol{\pi}_{\mathcal{S}}$ was used to best compare with predictions of the ZIP and the SCIP.

Comparison of Predicted Counts

In addition to group membership, accuracy of predictions for each of the three models is of interest. This method of comparison is based upon the theory that the MIP may accurately model an observed distribution (as each value may represent the observed probability in a training dataset), but the specification of a correct underlying distribution will prevent such models from overfitting. In this way, though error estimates of the SCIP and MIP may be similar for a training dataset, prediction errors may differ. Comparison of the MSE for predictions of known outcomes is conducted. The equation used to calculate MSE is

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(\hat{Y}_i - Y_i\right)^2, \tag{66}$$

where $\hat{Y}_i$ is the i^{th} predicted value, Y_i is the i^{th} value, and n is the number of observations in the test set. Additionally, plots of predicted values over weight ranges will be given for the models.

Simulation

The simulation for this study was conducted using the most recent version of `R` (currently 3.4.1). Separate values of c (the cutoff for inflation) and n (the sample size of the training dataset) were used.

Cutoffs of $c = 2, c = 6$, and $c = 8$ are used. Giles (2007) demonstrates the use of two-inflated count models, while Lin and Tsai (2013) shows an example of a six-inflated count model. The cutoff of eight was additionally selected as a theoretical relationship to the empirical situation: it is the number of nights that higher drinking behavior may be expected in a four week month (a.k.a. number of weekend nights).

Sample sizes of training datasets used in simulation are $n = 25, n = 50, n = 100, n = 200$, and $n = 500$, representing multiple ranges of samples that may be seen in practice. Zhang et al. (2016) demonstrates the use of sample sizes from $n = 50$ to $n = 500$ in their simulation study on zero-and-one inflated Poisson distributions. A sample size of 25 is included to evaluate the relative accuracy of these models in small sample data situations. The `optim` function in `R` is used to solve the derivatives to obtain maximum likelihood estimates (R Core Team, 2017)

Data Generation

Data generation will use randomly generated outcomes and independent variables, with parameters from the empirical fitting of the SCIP. First, the independent variable weight is generated according to the observed mean and variance of the weight variable from the BRFSS. Second, estimated parameters from fitting the SCIP model to the BRFSS dataset are used as population parameters in the data generation. That is, parameter estimates from the model

$$\begin{aligned}\log(\lambda_{\mathcal{SB}_i}) &= \hat{ϡ}_0 + \hat{ϡ}_1 x_{W_i}\\ \log(\lambda_{\mathcal{LB}_i}) &= \hat{ϟ}_0 + \hat{ϟ}_1 x_{W_i}\\ \operatorname{logit}(\pi_{\mathcal{SB}_i}) &= \hat{\gamma}_0 + \hat{\gamma}_1 x_{W_i}\end{aligned} \tag{67}$$

are used as population parameters for generating data. Third, given these parameter estimates, a Bernoulli distribution with probability $\pi_{\mathcal{SB}_i}$ randomly generates group membership. Finally, if the observation falls within the small count group, a right-truncated Poisson (truncated at c) is generated with shape parameter $\lambda_{\mathcal{SB}_i}$. If the observation does not fall into the small count group (it falls into the large count group), an un-truncated Poisson is generated with shape parameter $\lambda_{\mathcal{LB}_i}$.

Starting values for the Poisson components are calculated via fitting a GLM to the data, while starting values for the logistic component are calculated through a logistic regression with the dataset split at c. An expectation-maximization process is conducted, alternating between estimation of the parameters and estimation of the group membership. Following the process of Zhang et al. (2016), this parameter estimation procedure occur with $k = 1000$ datasets. For each dataset, models shown in Equation 62, Equation 63, and Equation 64 are fitted.

After all parameter estimates have been obtained, research question four is addressed by comparing average correct group classification percentages and average area under the ROC curve. Research question five is addressed by comparing average prediction MSEs. Models with higher classification percentages, higher area under the ROC curve, and lower average MSEs will be judged as more efficient.

CHAPTER IV

RESULTS

This chapter presents the empirical and simulation statistics used to evaluate the model that is presented in Chapter 3. First, review of previously addressed topics is provided. Then, details on implementation are provided, ending with results of the fitting of the empirical dataset. Finally, details and results of the simulation study are presented.

Specification

To address research question one, specification of the small count inflated Poisson (SCIP) distribution has been demonstrated in Chapter III. This includes piecewise and joint probability mass functions (PMFs) for the SCIP, as well as details on designating a SCIP model.

Parameter Estimation

To address research question two, parameter estimation for the SCIP has been demonstrated in Chapter III. This includes the definition of the likelihood function and determining derivatives of the log-likelihood with respect to parameter vectors, setting these derivatives to zero, and solving simultaneously for the parameter estimates. The expectation-maximization (EM) algorithm, further discussed below, is used to determine an iterative solution for the parameters (Dempster, Laird, & Rubin, 1977).

Implementation

To address research question three, `R` code demonstrating the implementation is included in the Appendix B. Details of SCIP implementation is discussed below. First, the EM process is briefly discussed. Then, asymptotic distributions of parameter estimates using the implemented process are presented. Finally, application of the implemented process to a large dataset is demonstrated.

Expectation Maximization

The EM algorithm is a popular method for parameter estimation, used for estimation by a majority of the finite mixture model (FMM) literature since its discovery (McLachlan & Peel, 2000). The process leverages the fact that were group membership known, estimation would be straightforward. Using this process, the EM algorithm alternates between estimating group membership (given the parameters) and estimating parameters (given group membership). This process iterates until convergence criterion are met. A detailed example of the process is provided for the ZIP in Chapter II.

Implementation of the EM algorithm for the SCIP follows the same principles. A Poisson model was used to determine initial parameter estimates for the small and large count components, while logistic regression (with the outcome being defined as 1 if $y_i < c$ and 0 otherwise) was used to determine initial parameter estimates for the logistic component. Given these parameters, initial probabilities of group membership were calculated. Using initial values, the process alternated between estimating group membership and estimating parameters, using the most updated estimates of each at each step. The process iterated until the change in parameters from one iteration to the next met convergence criterion (all changes were less than 10^{-9}) or met the maximum number of iterations (100). First, let

$$\boldsymbol{\Psi} = \begin{bmatrix} \boldsymbol{\beta} \\ \boldsymbol{\zeta} \\ \boldsymbol{\gamma} \end{bmatrix}. \tag{68}$$

Following the algorithm described by McLachlan and Peel (2000), the k^{th} iteration of the algorithm uses

$$\hat{\tau}_{\mathcal{S}}^{(k+1)}(y_i,\hat{\mathbf{\Psi}}^{(k)}) = \frac{\hat{\pi}_{\mathcal{S}_i}^{(k)} f_{\mathcal{TP}}(y_i,\hat{\lambda}_{\mathcal{S}_i}^{(k)})}{\hat{\pi}_{\mathcal{S}_i}^{(k)} f_{\mathcal{TP}}(y_i,\hat{\lambda}_{\mathcal{S}_i}^{(k)}) + (1-\hat{\pi}_{\mathcal{S}_i}^{(k)}) f_{\mathcal{P}}(y_i,\hat{\lambda}_{\mathcal{L}_i}^{(k)})} \tag{69}$$

as the next estimate of group membership in the small count inflated group, where $\hat{\tau}_{\mathcal{S}}^{(k+1)}(y_i,\hat{\mathbf{\Psi}}^{(k)})$ is the function of updated estimates of group membership given the i^{th} observed count (y_i) and the estimated parameters from the current iteration ($\hat{\mathbf{\Psi}}^{(k)}$), $\hat{\pi}_{\mathcal{S}_i}^{(k)}$ is i^{th} element of the current estimated population proportion of the small count inflated group, $f_{\mathcal{TP}}(y_i,\hat{\lambda}_{\mathcal{S}_i}^{(k)})$ is the evaluation of the i^{th} element of the truncated Poisson given the observed count and the current estimate of its mean, and $f_{\mathcal{P}}(y_i,\hat{\lambda}_{\mathcal{L}_i}^{(k)})$ is the evaluation of the i^{th} element of the Poisson given the observed count and the current estimate of its mean. Similarly, McLachlan and Peel (2000) globally maximize

$$\begin{aligned} Q(\mathbf{\Psi},\hat{\mathbf{\Psi}}^{(k)}) = \sum_{i=1}^{n} \Big(& \hat{\tau}_{\mathcal{S}_i}^{(k+1)}(y_i,\hat{\mathbf{\Psi}}^{(k)}) \log(\pi_{\mathcal{S}_i} f_{\mathcal{TP}}(y_i,\lambda_{\mathcal{S}_i})) + \\ & (1-\hat{\tau}_{\mathcal{S}_i}^{(k+1)}(y_i,\hat{\mathbf{\Psi}}^{(k)})) \log((1-\pi_{\mathcal{S}_i}) f_{\mathcal{P}}(y_i,\lambda_{\mathcal{L}_i})) \Big) \end{aligned} \tag{70}$$

with respect to $\mathbf{\Psi}$ to give the updated estimate $\mathbf{\Psi}^{(k+1)}$. Note that because $\mathbf{\Psi}$ is made up of ϡ, ϟ, and $\boldsymbol{\gamma}$ which define $\pi_{\mathcal{S}_i}$, $\lambda_{\mathcal{S}_i}$, and $\lambda_{\mathcal{L}_i}$, the means in Equation 70 are not the k^{th} iteration estimates because Q is being maximized with respect to them in order to determine their estimates.

Asymptotics

Chapter II discusses the importance of identifiability in FMM. As discussed, there is little research into identifiability of excess-zero models (and beyond), with the zero-inflated Poisson (ZIP) being the only model in this class where identifiability has been shown (Li, 2012). Additionally, McLachlan and Peel (2000) discuss that the label-switching problem for identifiability is not of concern when utilizing the EM algorithm (and presenting a possible arrangement of parameters). This said, it is important to determine asymptotic properties of parameter estimates, especially when lack of identifiability from other sources is of question. Though a formal proof of identifiability is not included as it is not the focus of this study, this section demonstrates that the parameter estimates of the SCIP used for the simulation show desirable asymptotic properties.

A simulation of 1000 replications was run with the SCIP, using a cutoff of $c = 8$ and sample sizes of $n = 25, 50, 100, 200$, and 500. Parameters were set at an intercept of -0.99 and slope of 0 for the small count inflated group, an intercept of 2.52 and a slope of 0 for the Poisson count group, and an intercept of 2.21 and a slope of 0 for the logistic regression (parameter estimates from the empirical example, shown in Table 2). For each sample size, histograms of parameter estimates were calculated. In these histograms, there are three main characteristics that were checked: the center of the distribution, the shape of the distribution, and the spread of the distribution. It is expected for parameter estimates to be asymptotically normal, centered on the actual value, and to have a reduced spread as the sample size increases.

Especially in the smaller sample sizes, there are large outliers that make meaningful visual summary of the data difficult. To most clearly illustrate the patterns of the estimates, histograms were constructed both with and without the outliers included, and are shown in Figures 1 and 2. A value was calculated to be an outlier for a parameter if it exceeded 1.5 times the interquartile range of the parameter estimates at that sample size. For convenience and clarity, the percentage of observations removed in creating the graph is noted in each histogram's subtitle (0% for all graphs in Figure 1).

To better illustrate the relative comparison among parameters and sample sizes, Table 1 shows the variance of each of the parameter estimates at each sample size. Note that outliers are not removed for the calculations shown in Table 1. It is clear that these variances decrease as the sample size increases, and that the pattern is consistent across all parameters.

Table 1.

Asymptotic Variation For Each Small Count Inflated Poisson Parameter by Sample Size

	n				
	25	50	100	200	500
Small Count Intercept	31.373817	1.899284	0.765476	0.331990	0.144733
Small Count Slope	0.000574	0.000048	0.000019	0.000008	0.000004
Large Count Intercept	222.668992	25.519419	0.935186	0.170806	0.046214
Large Count Slope	0.006363	0.000602	0.000021	0.000004	0.000001
Logistic Intercept	8718.888224	23.275833	3.430233	1.624473	0.575867
Logistic Slope	0.210430	0.000415	0.000083	0.000040	0.000014

In addition to other observations, is clear that the logistic and large count components require the largest sample sizes to produce estimates without outliers. A likely reason for this slow convergence is the large weight assigned to the small count group, making group prediction and estimation of the large count group (with a much lower relative sample size) difficult. Further discussion of this is presented in Chapter 5.

Figure 1. Empirical Asymptotic Distribution of SCIP Parameter Estimates Without Outliers Removed

Figure 2. Empirical Asymptotic Distribution of SCIP Parameter Estimates With Outliers Removed

Figures 1 and 2 illustrate that SCIP parameter estimates have the three desired traits of parameter estimates. Each shows that the distribution is roughly centered around the actual value (shown via the vertical dotted line). It is also clear that as the sample size increases, the variation decreases. This is best seen through the change in range of values shown on the x-intercept of the graphs (not held constant for relative viewing, especially in Figure 1, because of the drastic differences in range of parameter estimates between sample sizes). Finally, as the sample size increases, the distribution of each of the parameter estimates appears to approach normality.

Empirical Example

For an empirical example on the use of the SCIP, the model was fit to the 2016 Behavioral Risk Factor Surveillance System (BRFSS) data collected by the Centers for Disease Control and Prevention (CDC). Binge drinking behavior over the previous 30 days is used as the outcome and weight is used as the independent variable. Two histograms of binge drinking behavior are shown in Figure 3. The histogram on the left shows all data, including the scale of inflation at zero. The histogram on the right illustrates the small population of individuals who display more regular binge drinking behavior.

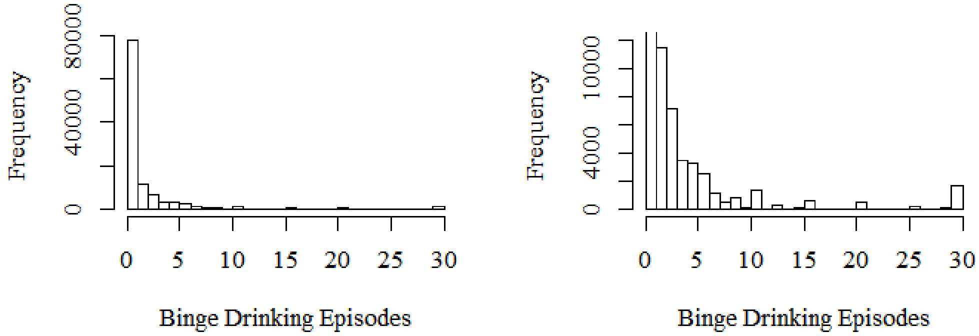

Figure 3. Histogram of Binge Drinking Behavior from the Behavioral Risk Factor Surveillance System

Figure 3 clearly illustrates the heavy weighting of the dataset towards smaller amounts of binge drinking episodes with a relatively small population of individuals with more binge drinking episodes. An estimated 90% of individuals fall into the "small count inflated" group, while roughly 10% of individuals fall into the "regular count" group.

The model is given as

$$\begin{aligned} \log(\lambda_{\mathcal{SB}_i}) &= \text{ϡ}_0 + \text{ϡ}_1 W_i, \\ \log(\lambda_{\mathcal{LB}_i}) &= \text{ϟ}_0 + \text{ϟ}_1 W_i, \\ \operatorname{logit}(\pi_{\mathcal{SB}_i}) &= \gamma_0 + \gamma_1 W_i, \end{aligned} \tag{71}$$

with W_i as the weight of the i^{th} observation, $\lambda_{\mathcal{SB}_i}$ as the small count inflated rate of occurrence of binge drinking for the i^{th} observation, $\lambda_{\mathcal{LB}_i}$ as the Poisson rate of occurrence of binge drinking for the i^{th} observation, and $\pi_{\mathcal{SB}_i}$ as the probability of the small count inflated population for the i^{th} observation.

Due to the heavy proportion of values in the small count group, the sample is limited to males (which are slightly more heterogeneous). Cleaning of the data involved translating survey codes into R, removing female respondents, and removing observations with missing values. For an example of the code cleaning, the codes 77 and 99 were used to identify missing values for binge drinking days, while 7777 and 9999 were used to identify missing values for weight. Additionally, values of 888 and 999 were removed as being outside of the realm of reasonable possibility (likely a coding error). Females and individuals with missing responses were completely removed from the cleaned sample. After cleaning, a sample size of $n = 113277$ respondents remained and were used for analysis. Table 2 shows the fit of the SCIP with the cutoff set to 8. SCIP models with other cutoffs (2 and 6) were also fit and gave similar results. These tables are shown in Appendix A.

Table 2.

Coefficients from Fitting the Small Count Inflated Poisson Model to the Behavioral Risk Factor Surveillance System, $c = 8$

	Inflated Count	Poisson Count	Logistic
(Intercept)	-0.989	2.520	2.208
Weight	0.001	0.000	0.000

Interpreting Table 2, the most obvious takeaway is the small relationship between weight on the binge drinking behavior of males in 2016. In the inflated count population, a 1 pound increase in weight corresponds with a multiplicative increase of $e^{0.001} = 1.001$ in the expected number of binge drinking episodes for males in 2016. In the Poisson count population, an increase in weight appears to have no relationship with the expected number of binge drinking episodes for males in 2016. Also, weight appears to have no relationship with the expected probability of a male being in the inflated count population vs in the Poisson population.

Due to the small relationship between weight and binge drinking behavior shown in Table 1, the intercepts become the primary areas of interest. The inflated count intercept at -0.989 implies that the expected number of binge drinking episodes per month for males in the inflated count population is $e^{-0.989} = 0.37$ episodes. The Poisson count intercept of 2.520 implies that the expected number of binge drinking episodes per month for males in the Poisson population is $e^{2.520} = 12.43$ episodes. The logistic regression intercept of 2.208 implies that the odds of being in the inflated count population are $e^{2.208} = 9.10$ times that of the odds of being in the Poisson count population.

Practically, this information defines two heavily unbalanced populations: a small count population (which has a large majority of the participants) that rarely binges, and a Poisson population that binges multiple times per week. Weight appears to have no relationship with the number of binge drinking episodes in either population, nor a relationship with the probability that an individual belongs to one population or the other.

Simulation Details

To address research questions four and five, a simulation study was conducted. This section will describe details of the general simulation procedure, while future sections will describe further details specific to each research question. First, details of the simulation conditions are discussed. Second, details of the data generation procedure are presented. Next, information on the implementation of comparison models are shown. Then, convergence information for all models is given. Finally, information on model predictions is included.

Simulation Conditions

Simulation conditions are presented in Chapter III. The simulation conducted 1000 trials over each of three models, five sample sizes, and three cutoffs of inflation. The multinomially-inflated Poisson (MIP) and the ZIP were used to compare with the SCIP

(Giles, 2007; Lambert, 1992). Cutoff values of $c = 2, 6$, and 8 were used (Giles, 2007; Lin & Tsai, 2013). Sample sizes of $n = 25, 50, 100, 200$, and 500 were used for model fitting, while sample sizes of $n = 5, 10, 20, 40$, and 100 (20% of the training sample sizes) were used for model prediction (Zhang et al., 2016).

Data Generation

The data generation procedure for SCIP data was also designed for this dissertation. The function takes a sample size, design matrix, cutoff of inflation, and a vector for each set of population parameters. First, the function calculates $\boldsymbol{\lambda}_S, \boldsymbol{\lambda}_L$, and $\boldsymbol{\pi}_S$ based on their respective parameter vectors and the design matrix (see Equation 39). Then, a random Bernoulli is generated with probability π_{S_i}. This represents the actual group that is the source of an observation's count. After this, if the observation is from the small count group, a value is generated from the truncated Poisson distribution using λ_{S_i} as the mean parameter (with a maximum of c). If the observation is not from the small count group, a value is generated from the Poisson distribution using λ_{L_i} as the mean parameter.

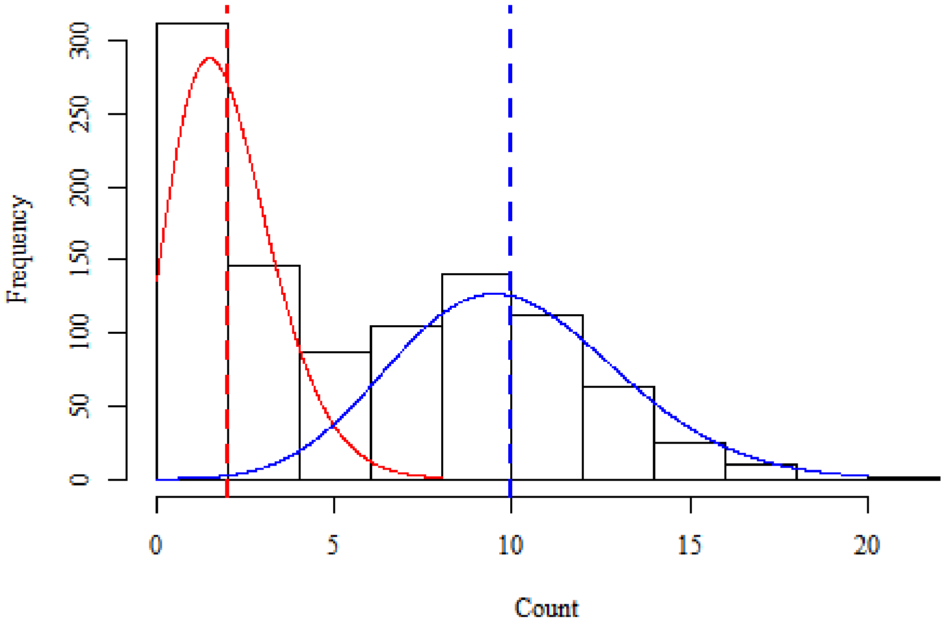

Figure 4. Example Histogram of Small Count Inflated Poisson Data Generation

Figure 4 shows an example histogram of sample generated data for demonstration purposes with the population curves overlayed. A series of these histograms generated with a variety of parameters were examined to determine if the resulting datasets seemed reasonable. The population parameters used to generate Figure 4 set all values of the independent variable to zero, set $\lambda_S = \log(2)$ for the intercept of the small count inflated group, $\lambda_L = \log(10)$ as the intercept for the Poisson group, and 0 for the intercept of the logistic regression. The resulting distribution is evenly balanced between inflated data and Poisson data, with the inflated counts centered at 2 and the Poisson counts centered at 10 (shown by the red and blue dotted lines, respectively). The example histogram shown in Figure 4 displays the expected mean, variation, and shape, indicating that data appear to be generated appropriately.

In addition to histograms, the examination into the asymptotics of the SCIP parameters offered a separate chance to test the correctness of the data generation function. As the parameters were consistently centered on population parameters given to the data generation function, there is evidence of alignment between SCIP estimation and data generation.

Comparison Model Implementation

This section presents details of implementation of the comparison models to supplement discussion on implementation of the SCIP. All models were implemented in `R` version 3.4.3 (R Core Team, 2017). The ZIP model was implemented via the `zeroinfl` function from the `pscl` package (Zeileis, Kleiber, & Jackman, 2008). No freely available program or package for the implementation of the MIP was found. Because of this, a custom function was created to implement the MIP based on Giles (2007) paper.

As Giles (2007) method gives a log-likelihood for the MIP, the implementation focused on recreation and direct maximization of the log-likelihood through the use of `optim` (as opposed to an EM approach). Giles (2007) defines the log-likelihood function based on a sample of *n* independent observations as

$$\begin{aligned} L(\beta,\gamma_1,\gamma_2,\ldots,\gamma_J) = &\sum_{y_i\in R_0}\log\left(\omega_{i0}+\left(1-\sum_{l=0}^{J-1}\omega_{il}\right)P_i\right) \\ &+\sum_{y_i\in R_1}\log\left(\omega_{i1}+\left(1-\sum_{l=0}^{J-1}\omega_{il}\right)P_i\right) \\ &+\ldots \\ &+\sum_{y_i\in R_{J-1}}\log\left(\omega_{iJ-1}+\left(1-\sum_{l=0}^{J-1}\omega_{il}\right)P_i\right) \\ &+\sum_{y_i\in R_J}\log\left(\left(1-\sum_{l=0}^{J-1}\omega_{il}\right)P_i\right), \end{aligned} \tag{72}$$

where

$$P_i = \frac{e^{-\lambda_i}\lambda_i^{y_i}}{y_i!}, \tag{73}$$

is the Poisson probability and ω is defined as

$$\omega_{ij} = \frac{e^{G_i'\gamma_j}}{1+\sum\limits_{l=0}^{J-1} e^{G_i'\gamma_l}}, \tag{74}$$

for $j = 0, 1, \ldots, J$, with $\gamma_J = 0$ imposed.

Giles method allows for distinct parameters for each of the inflated values, the implementation allowed distinct parameters for each inflated value. Implementation used the Poisson group as the reference group for each of the logistic regressions and used an adapted notation from (Giles, 2007) original method.

To test the implementation of the three models, all three were fit to the same zero-inflated data using the zero-inflated reductions of the MIP and SCIP. Based on theory, all of the models should have the same parameter estimates (to within rounding error). Trials showed all models to match exactly to the thousandths place, indicating that the SCIP and MIP implementations match the ZIP (which has already been ratified) with zero-inflated data. This also demonstrates how the traditional ZIP model is a subset of the MIP and SCIP.

It is important to note that while the MIP and ZIP are implemented via a single optimization of the likelihood through the use of `optim`, the SCIP is implemented via the EM algorithm (and uses `optim` during each iteration). There are many considerations for optimization of a model, each of which can have important impacts on parameter estimation and convergence percentages. For instance, changes in the maximum number of iterations may directly affect model convergence. Additionally, differing starting values and methods of optimization may effect parameter estimation and convergence for a given model. This dissertation does not propose to address the breadth of these conditions to implement ideal optimization for each model under the current data situation. Instead, standard system defaults are used for each implementation. For SCIP implementation, `optim` defaults are used for maximization within each iteration. For MIP implementation, `optim` defaults are used for maximization. For ZIP implementation, `zeroinfl` defaults are used.

Model Fitting

Under each of the simulation conditions, data were generated and all models were fit. Model information and fit statistics were stored for each iteration. In addition to parameter estimates, information was stored for the run-time, iterations, model convergence, and errors.

Convergence

Table 3 shows convergence percentages for all models and all trials. Convergence issues of the ZIP are reported directly from the `zeroinfl` function, while convergence issues from the SCIP are marked when the maximum number of iterations is reached. As parameter estimation for the MIP uses a direct maximization of the log-likelihood, convergence is reported directly from `optim` (which uses the default of 500 for the maximum number of iterations). As discussed previously, conditions under which the likelihood for each set of parameters is optimized may have a direct impact on the convergence rates shown in Table 3. As such, convergence rates should be considered not just in the context of each model, but also in context of each model's optimization procedure.

Table 3.

Convergence Percentages by Model, Cutoff of Inflation, and Sample Size

		n				
		25	50	100	200	500
$c = 2$	ZIP	99.6%	99.9%	100%	100%	100%
	MIP	0.2%	0%	0.1%	0%	0%
	SCIP	92.1%	98.4%	100%	100%	100%
$c = 6$	ZIP	99%	100%	100%	100%	100%
	MIP	0%	0%	0%	0%	0%
	SCIP	92.2%	98%	99.9%	100%	100%
$c = 8$	ZIP	99.5%	100%	100%	100%	100%
	MIP	0%	0%	0%	0%	0%
	SCIP	90.5%	98.2%	99.8%	100%	100%

Some degree of lack of convergence was present for all models in the smallest sample size. The ZIP and the SCIP show clear improvement in convergence as the sample size increases, while the MIP shows no apparent growth. It can be seen that the MIP rarely converges, even for the lowest cutoff. Also, convergence of the MIP doesn't seem to improve as the sample size increases. The lack of convergence for the MIP may be due to the large numbers of parameters being estimated or the optimization procedure used, and results in skepticism regarding MIP parameter estimates. Unfortunately, previous convergence rates of the MIP under similar data conditions could not be found for comparison.

Investigation into issues of non-convergence for the SCIP indicated that one reason for non-convergence of the SCIP is complete or quasi-complete separation of the data (Albert & Anderson, 1984). In such a case, the parameters from the logistic component cannot be estimated, and the SCIP cannot maximize the likelihood (causing the error and non-convergence). Similarly, one reason for lack of convergence for the ZIP was from a singularity error, likely due to the small size of the dataset relative to the heavy weighting towards the small count inflated population. In such cases, the Poisson count group membership is a “rare event,” causing instability in parameter estimates.

Overall, the ZIP seems to have the best convergence for very small sample sizes, though both the ZIP and the SCIP consistently converge. At more reasonable sample sizes (as low as $n = 100$), both the ZIP and the SCIP converge almost 100% of the time. Though not considered here, use of an optimization method that results in consistent convergence may improve results for models when convergence criterion were not met (the SCIP in small sample sizes, the MIP for all sample sizes).

Prediction

For each set of parameter estimates, new data of size $n = 5, 10, 20, 40$, and 100 were generated using actual parameters. All actual values were compared to predictions of the three models based on the simulated data. Predicted summary statistics for each run (MSE, AUC, percent correct) were saved and aggregated across 1000 runs. Due to outliers (especially at the low sample sizes), medians are presented in conjunction with empirical 95% confidence intervals. Sample size labels on tables and graphics presented represent the sample size the models were trained on (e.g., $n = 25, 50, 100, 200$, and 500) as opposed to the sample size the models were tested on (e.g., $n = 5, 10, 20, 40$, and 100, respectively).

Group Prediction

To address research question four, this section presents results of group prediction from the simulation study. Estimates of group membership for the ZIP and SCIP used the logistic component to indicate predicted group membership. For the MIP, all inflated logistic components (from 0 to c) were summed to give the most accurate prediction of inflated group membership. That is,

$$\boldsymbol{\pi}_{\mathcal{S}} = \sum_{g=0}^{c} \boldsymbol{\pi}_{g}, \tag{75}$$

where $\boldsymbol{\pi}_{\mathcal{S}}$ is the vector of probabilities of an observation being in the small count inflated group and c is the cutoff of inflation.

These were compared to a cutoff of 0.5: if above, the predicted group was the small count inflated group, and if below, the predicted group was the Poisson group. Each prediction was compared to actual group membership via % correct and AUC statistics, with percentage correct being directly calculated and AUC being calculated through use of the pROC package (Robin et al., 2011).

Percentage Correct Classification

Observed percentage correct classification values for all methods are shown in Figure 5 and Table 4.

Table 4.

Percent Correct Classification Medians with Empirical 95% Confidence Intervals by Model, Cutoff of Inflation, and Sample Size

		n				
		25	50	100	200	500
	ZIP	0.8 (0, 1)	0.8 (0.3, 1)	0.85 (0.6, 0.951)	0.85 (0.725, 0.95)	0.85 (0.77, 0.92)
$c = 2$	MIP	0.8 (0.4, 1)	0.9 (0.6, 1)	0.85 (0.7, 1)	0.85 (0.725, 0.95)	0.85 (0.78, 0.92)
	SCIP	0.8 (0.4, 1)	0.9 (0.6, 1)	0.85 (0.7, 1)	0.85 (0.725, 0.95)	0.85 (0.78, 0.92)
	ZIP	0.6 (0, 1)	0.7 (0, 1)	0.85 (0.15, 1)	0.875 (0.475, 0.975)	0.89 (0.75, 0.95)
$c = 6$	MIP	1 (0.6, 1)	0.9 (0.7, 1)	0.9 (0.75, 1)	0.9 (0.8, 0.975)	0.9 (0.83, 0.95)
	SCIP	0.8 (0.4, 1)	0.9 (0.7, 1)	0.9 (0.75, 1)	0.9 (0.8, 0.975)	0.9 (0.83, 0.95)
	ZIP	0.6 (0, 1)	0.7 (0, 1)	0.85 (0.2, 1)	0.875 (0.549, 0.975)	0.89 (0.76, 0.95)
$c = 8$	MIP	1 (0.6, 1)	0.9 (0.7, 1)	0.9 (0.75, 1)	0.9 (0.8, 0.975)	0.9 (0.83, 0.95)
	SCIP	1 (0.4, 1)	0.9 (0.6, 1)	0.9 (0.75, 1)	0.9 (0.8, 0.975)	0.9 (0.83, 0.95)

It can be seen that the ZIP model has the worst prediction accuracy of all of the models and the widest confidence interval. The ZIP confidence intervals are also wider for larger values of c. The SCIP and MIP models seem extremely similar in terms of prediction accuracy, with the confidence intervals of the SCIP also being wider for larger values of c at lower sample sizes. As the sample size increases, all three models appear to approach the same median prediction accuracy and confidence interval for all values of c.

Figure 5. Percentage Correct Classification Medians with Empirical 95% Confidence Intervals by Method, Cutoff of Inflation, and Sample Size

Area Under the Curve

During the prediction process, errors appeared in the calculation of the AUC statistic. Review indicated that generated datasets (especially at smaller sample sizes and larger cutoffs) were generating all observations from the same group (the small count inflated group), making fitting of the logistic model and calculation of the AUC impossible. In such cases, calculation of AUC statistics continued with remaining datasets (% correct statistics were not impacted). Error rates for the AUC of the SCIP are presented in Table 5, and are almost identical for all models (tables for other models shown in Appendix A).

Table 5.

Error Rate of the Area Under the Curve by Cutoff of Inflation and Sample Size for the Small Count Inflated Poisson Model

	n				
	25	50	100	200	500
$c = 2$	44.3%	20.5%	3.2%	0%	0%
$c = 6$	56.9%	35.1%	10.5%	1.1%	0%
$c = 8$	59%	34.8%	9.1%	1%	0%

Observed AUC values for all methods are shown in Figure 6 and Table 6.

Table 6.

Area Under the Curve Medians with Empirical 95% Confidence Intervals by Model, Cutoff of Inflation, and Sample Size

		n				
		25	50	100	200	500
$c = 2$	ZIP	0.75 (0.5, 1)	0.688 (0.438, 1)	0.639 (0.451, 0.947)	0.589 (0.451, 0.827)	0.555 (0.449, 0.689)
	MIP	0.75 (0.5, 1)	0.688 (0.438, 1)	0.639 (0.444, 0.947)	0.59 (0.451, 0.829)	0.555 (0.449, 0.687)
	SCIP	0.75 (0.5, 1)	0.688 (0.438, 1)	0.639 (0.446, 0.947)	0.589 (0.451, 0.827)	0.555 (0.449, 0.689)
$c = 6$	ZIP	0.75 (0.5, 1)	0.714 (0.5, 1)	0.667 (0.457, 1)	0.605 (0.45, 0.921)	0.558 (0.457, 0.731)
	MIP	0.75 (0.5, 1)	0.688 (0.5, 1)	0.667 (0.466, 1)	0.605 (0.447, 0.921)	0.558 (0.457, 0.731)
	SCIP	0.75 (0.5, 1)	0.714 (0.481, 1)	0.667 (0.451, 1)	0.605 (0.447, 0.921)	0.558 (0.457, 0.731)
$c = 8$	ZIP	0.75 (0.5, 1)	0.688 (0.5, 1)	0.667 (0.451, 1)	0.611 (0.451, 0.897)	0.561 (0.454, 0.736)
	MIP	0.75 (0.5, 1)	0.688 (0.5, 1)	0.667 (0.444, 1)	0.611 (0.45, 0.897)	0.561 (0.454, 0.736)
	SCIP	0.75 (0.5, 1)	0.688 (0.5, 1)	0.667 (0.451, 1)	0.611 (0.451, 0.897)	0.561 (0.454, 0.736)

Results for each method appear almost identical in terms of the AUC statistic, both in terms of median prediction and in terms of the confidence intervals. Confidence intervals for all methods are smallest for $c = 2$, though the median estimates of AUC increase slightly as c increases. Also, as n increases, the AUC decreases for all cutoffs of inflation and methods.

Figure 6. Area Under the Curve Medians with Empirical 95% Confidence Intervals by Model, Cutoff of Inflation, and Sample Size

For all methods, confidence intervals decrease in width as the sample size increases. Unlike the percentage correct statistic, estimates of the AUC decrease as sample size increases. A discussion of AUC and percentage correct behavior follows.

Group Statistic Behavior

At first glance, behavior of the AUC across sample size seems to be at odds with that indicated by the percentage correct statistic. The percentage correct seems to indicate that group prediction accuracy increases as the sample size increases, while the AUC seems to indicate that it decreases as sample size increases. The apparent discrepancy is due to the different meanings and properties of the different statistics.

Percentage correct is a raw percentage, comparing the predicted group membership with the actual group membership. Due to the heavy weighting of the sample towards the small count inflated group and the small relationship between the dependent variable (DV) and independent variable (IV), static predictions of group membership (a.k.a just guessing 1 every time) can seem to perform well. In this case, the success of static prediction is due to the heavy weighting of the variable towards the small count inflated group (and the small relationship between the DV and IV, giving the model minimal information to better predict). This would explain why group prediction shown in Figure 5 levels off at the inverse-logit of the intercept from the logistic model (.86 for $c = 2$, .90 for $c = 6$, and .90 for $c = 8$).

In contrast to percentage correct, ROC curves (and thus AUC statistics) are insensitive to changes in the distribution of classes (Fawcett, 2006). Fawcett (2006) explains how examination of a confusion matrix indicates the reason for this insensitivity: percentage correct statistics compare the number correct to the total number of attempts, while ROC curves compare true positives and false positives. Essentially, this places equal weight on correct classification *within each population*. Again, the low sample size in the Poisson count population may explain the behavior of the classification statistic. In the case of the smallest sample size, all AUC statistics included in Figure 6 have at least one value that comes from each distribution, and likely only one value that comes from the Poisson count distribution. Considering correct classification in terms of each group, each model's group prediction is likely to correctly classify all of the observations coming from the heavy weighted population (the small count inflated population). For the value(s) coming from the Poisson count population, each model may correctly or incorrectly classify these values, but is less likely to have a correct classification due to the large weight towards the small count population and the small relationship between the IV and DV.

Given only one value from the Poison count population, a 75% AUC (shown across values of c for $n = 25$ in Figure 6) indicates an average ranking of a false positive in the middle of rankings of true positive values (a.k.a the inability of the model to distinguish between the small count inflated and Poisson count groups). As the sample size increases, the probability of the inclusion of more instances of the Poisson group increases. Decreases in the AUC as the sample size increases indicate the increasing probability of false prediction relative to true prediction as the samples begin to better represent both populations. In other words, the AUC indicates that accurate distinction between the small count inflated group and the Poisson group is poor. The statistics are deceptive at small sample sizes due to the large probability of belonging to the small count inflated group, and thus, the small probability that small samples include values from the Poisson count group. This is evidenced by the decreasing AUC as the sample increases and includes more and more instances of the Poisson count group. The decrease appears to converge to 0.5 (random guessing), which is consistent with a small predictor effect.

Count Prediction

To address research question five, this section presents results of count prediction from the simulation study. Values of predicted mean squared error (MSE) for all methods are shown in Figure 7 and Table 7.

Table 7.

Mean Squared Error Medians with Empirical 95% Confidence Intervals by Model, Cutoff of Inflation, and Sample Size

				n		
		25	50	100	200	500
$c = 2$	ZIP	9.8 (0, 56.313)	12.4 (0.1, 43.802)	13.3 (0.599, 33.883)	13.55 (3.424, 27.702)	13.455 (6.778, 21.843)
	MIP	7.4 (0, 58.005)	12.275 (0.1, 44.504)	13.225 (0.499, 34.561)	13.562 (3.374, 27.702)	13.46 (6.788, 21.843)
	SCIP	7.155 (0.028, 56.898)	11.129 (0.118, 42.502)	12.393 (0.381, 32.657)	12.698 (3.153, 26.461)	12.69 (6.286, 20.79)
$c = 6$	ZIP	8.401 (0.039, 74.41)	12.252 (0.2, 53.101)	13.881 (0.55, 40.772)	14.525 (2.775, 33.422)	15.185 (5.967, 26.421)
	MIP	1 (0, 78.605)	10.55 (0.2, 52.807)	13.625 (0.4, 42.156)	14.488 (2.599, 33.081)	15.17 (6.018, 26.47)
	SCIP	1.089 (0.101, 81.825)	9.601 (0.182, 50.169)	12.667 (0.312, 39.49)	13.331 (2.239, 31.014)	14.062 (5.522, 24.63)
$c = 8$	ZIP	7.8 (0.036, 68.727)	12.752 (0.2, 57.005)	14.045 (0.55, 38.862)	14.65 (3.272, 34.3)	15.14 (6.91, 26.319)
	MIP	1 (0, 72.415)	11.2 (0.1, 57.825)	13.5 (0.4, 40.377)	14.488 (2.474, 34.504)	15.085 (6.818, 26.45)
	SCIP	1.118 (0.109, 72.28)	11.107 (0.18, 55.26)	12.604 (0.323, 37.369)	13.435 (2.15, 32.543)	13.966 (6.181, 24.547)

It can be seen that confidence intervals are smallest for a cutoff of $c = 2$, while the size of the confidence intervals decrease for all models and cutoffs as the sample size increases. For almost all conditions, the SCIP has a slightly lower median MSE and slightly smaller confidence interval than the ZIP and the MIP, which are nearly identical to one another. The exception of this is for the smallest sample size ($n = 25$), where the SCIP and MIP have more comparable MSEs and confidence intervals.

Figure 7. Mean Squared Error Medians with Empirical 95% Confidence Intervals by Model, Cutoff of Inflation, and Sample Size

Summary

To address research questions three through five, implementation of the SCIP (outlined in Chapter III) has been presented alongside details of a simulation study. Results were presented for a variety of cutoffs of inflation and sample sizes, chosen to align with previous research. Problems in sparse data introduced errors which made logistic fitting and AUC calculation impossible. Confidence intervals of statistics evaluating fit of the SCIP overlapped with those of the ZIP and MIP in metrics of group prediction and in the metric of count prediction for all cutoffs of inflation and sample sizes.

CHAPTER V

CONCLUSION

In this dissertation, a small count inflated Poisson (SCIP) model was developed to model count data with multiple points of inflation that are related through an underlying distribution. Maximum likelihood estimation via the expectation-maximization (EM) algorithm was used to determine parameter estimates. Implementation of the SCIP method was described, coded in `R`, and demonstrated through the fitting of the Behavioral Risk Factor Surveillance System (BRFSS) dataset. A simulation study was conducted using parameter estimates calculated from the fitting of the BRFSS as population estimates. The simulation fitted the SCIP as well as two comparison models: the zero-inflated Poisson (ZIP) and multinomially-inflated Poisson (MIP).

Research questions one and two were addressed in Chapter III, with the specification of the distribution and parameter estimation for the SCIP. Research questions three, four, and five were addressed in Chapter IV and included an illustration of implementation with fitting of empirical data, group prediction comparisons between the models, and count prediction comparisons between the models.

Discussion

Convergence

Simulation results empirically demonstrate convergence issues with the MIP that were an inciting reason for this study. Table 3 shows the MIP rarely converging, even at large sample sizes. Though convergence criterion are calculated by different programs for the three models, the use of a standardized function in `R` (`optim`) makes observed results comparable across other algorithms. In addition to lack of convergence for the MIP, Table

3 also shows convergence issues for the SCIP at small sample sizes. Though not exclusive, issues with separation (prohibiting logistic prediction) contributed to the convergence issues for the SCIP.

Though the MIP rarely converged, results do not indicate a drop in performance when comparing to the models that consistently converged. In fact, the MIP provided equivalent group prediction accuracy when compared to the other models, and slightly better count prediction accuracy than the ZIP. This may be for a variety of reasons. First, due to the small relationship between weight and binge drinking, lack of convergence in slopes may not appreciably affect overall performance of model prediction. Second, due to heavy weighting of the small count inflated population, lack of estimation specificity may not fully be exemplified. Third, though default settings in `optim` were used, alternate settings may provide better estimation specifically for the MIP. As the purpose of this dissertation was *not* a detailed investigation into the fitting of the MIP, alternate settings were not considered during parameter estimation.

The SCIP seemed to have decent convergence rates at the smallest sample size, $n = 25$, and converged almost all of the time at sample sizes of $n = 50$. Similar to the MIP, it is expected that convergence percentages will differ as the underlying population parameters differ (e.g., a higher convergence rate for more balanced populations and better predictors).

Group Prediction

In addition to issues of convergence, problems with the AUC lowered the effective simulation size for AUC calculation considerably, especially for lower sample sizes. Due to the lack of multiple groups, AUC was unable to be calculated for over 50% of the $n = 25$ simulation. Instead of treating these as errors, however, the lack of multiple groups can be treated as information in its own right. The presence of only one group (the small count inflated group) for more than 50% of the simulations for the sample size of $n = 25$ can give information on the prediction accuracy shown in the percentage correct tables.

Namely, were models to predict only one, we would expect high prediction accuracy given that the majority of the simulations were made up of only ones (for the $n = 25$ sample size). This also explains why there is such a discrepancy between AUC and percentage correct, especially at the lower sample sizes.

Percentage correct indicated that all models performed well. MIP and SCIP percentage correct group prediction were almost identical, with the SCIP having wider confidence interval bands at lower sample sizes. As c increases, the ZIP did increasingly poorer as a group predictor, showing both lower median prediction percentages and wider confidence intervals. Though this information seems promising at first glance, insight into the AUC errors and the AUC graphics indicate that the shown "accuracy" is likely due to a naive prediction: that is, predicting the same group for all observations. There are two major reasons for this. The very high proportion of small count inflated observations relative to Poisson count observations gives a lot of weight to just predicting one population. Also, the small coefficient of weight in the logistic model (shown in Table 2) provides little information for models to distinguish between populations.

AUC was virtually identical for each of the models and each of the cutoffs. As the sample size rose, the widths of the confidence intervals decreased while the median estimates also decreased, appearing to approach 0.5 (random guessing). This is indicative of the increased presence of observations from both groups. Again, the high relative frequency of one population and the low coefficient of weight in the logistic model play key roles in lack of group prediction accuracy. As explained in Chapter 4, though the AUC appears to be counter-intuitively decreasing as the sample size increases, this likely results from the naive prediction method employed coupled with the small sample sizes. As sample size increases, heterogeneity in the sample for each run increases, and the AUC shows a decrease (as it gets closer to its true value).

Overall, there was little to distinguish between the models in terms of group prediction in this simulation. Though the ZIP appears to have performed worse than the

MIP or the SCIP for smaller sample sizes according to the median percentage correct statistic, it is virtually identical according to the AUC. Also, as all of the confidence intervals of models cross for all cutoffs and sample sizes, there can be no claim of superiority in group membership prediction. Again, the size of the coefficient of the predictor and the relative frequency of the populations may influence the relative performance of these parameters.

It is somewhat unexpected that the ZIP is comparable to the MIP and SCIP, as the latter account for similar natures of inflation that aren't included in the ZIP model. This said, the MIP and SCIP model the information differently than one another, yielding different parameter estimates and different interpretations. Comparability of the two in terms of group prediction allows theory to guide model selection, and encourages practitioners to select the model they believe represents population.

Count Prediction

Mean squared error (MSE) for the three models, shown in Figure 7, also appeared to be equivalent. For all models, as the sample sizes increase, the confidence interval for the MSE shrinks. For most sample sizes and all cutoffs of inflation, the SCIP appears to have a slightly smaller median MSE and slightly narrower confidence interval than the other models. Though this may indicate some small superiority of the SCIP (as the confidence intervals are empirical), there is still wide overlap among the confidence intervals.

Reasons for increasing median error as the sample size increases align with those presented in group prediction. In short: if group prediction is more accurate (e.g., for the small sample sizes), the MSE will be smaller. As sample sizes increase, groups become more and more heterogeneous, and prediction error becomes larger and larger for models using a naive group prediction. In addition to group prediction components, the small relationship between weight and binge drinking episodes in the small count inflated and Poisson count populations may also play a role in the poor prediction by the models.

Overall

It appears that groups and count prediction suffered in all models. The two primary reasons are the heavy weighting of the small count inflated population over the Poisson count population, and the small coefficient of weight in all of the population models (small count, large count, and logistic). All models seemed to perform roughly equally, with mild exceptions (the ZIP was slightly inferior in percentage correct group prediction, and the SCIP was slightly superior in MSE). Ultimately, given a difficult data situation, the SCIP performed as well or better than current available methods, with a much higher convergence percentage than the MIP.

Recommendations

Reviewing these results, there seems little to distinguish between models in terms of prediction in situations where the underlying population is similar to that used in this study. Because of this, a theoretical guide is recommended when selecting an inflated count model. For instances where theory explains that an underlying population has an inflation at zero (and nowhere else), the ZIP would be recommended. In cases where theory explains that an underlying population has multiple, unrelated points of inflation, the MIP would be recommended. In cases where theory explains a single cause for underlying inflation at multiple related points, the SCIP would be recommended. Similar model prediction accuracy demonstrated in this study shows that, at least in the data situation presented here, model selection may occur purely for the theory one wishes to test (and not for prediction accuracy).

If prediction is the only goal of model selection, statistics not presented (e.g., runtime) may be a more influential factor than model accuracy for similar data situations. In this, the SCIP loses outright (much longer runtimes than the other two). If convergence is of concern, tis implementation of the MIP should not be selected (though it is notable that it performed comparable to other models, even with convergence issues). For count prediction accuracy, the SCIP seems to be slightly better than the other two; for group prediction, the ZIP seems to be slightly worse than the other two.

The SCIP offers a unique look at the potential underlying structure of data. When it is believed that the underlying distribution is made up of two populations, a count population and a truncated count population, the SCIP allows interpretation of parameters that relate to both underlying populations as well as parameters that influence the difference between the two, which is not offered by other methods currently available. Additionally, this study has demonstrated that the SCIP can handle prediction as well or better than other methods across a range of sample sizes and inflated counts.

Convergence issues and heavy outliers, as displayed in the asymptotic plots, indicate that the SCIP should be used with a minimum sample of 50 to 100 for consistent results with populations of similar characteristics to the one used in this study. If used with a smaller sample, extreme results may be an indication of an outlier and should not be used (or, at the least, treated with extreme caution). Additionally, convergence and iterations should be checked when examining results. Though not presented in this study, there appeared to be a correlation between extremely low and high iterations (e.g., 2 iterations, 100 iterations) and outliers.

This said, as Figure 1 indicates, the distributions for each of the components of the model (small count, large count, logistic) have different asymptotic properties for this population. In this data situation, the small count component seems to perform much better at lower sample sizes than the large count or logistic components. This should also be taken into account when considering model use.

In a larger context, it is likely that the SCIP (and the other models, for that matter) will improve in performance as the underlying populations are more balanced and as the predictor has a larger relationship with the outcome. A meaningful predictor in the SCIP logistic component will likely improve group prediction, while further adding a meaningful predictor to the count components will likely improved count prediction.

Further study is needed to determine relative group and count prediction accuracy among models in these data situations.

Future Research

This dissertation used a Poisson distribution to model both the small count inflation and the larger count population. Though Poisson models are widely used for count data, a negative binomial distribution may also be considered, especially for the large count population. Additionally, the SCIP model applied in this dissertation assumed data were collected at only a single time. For instance, analyses of the BRFSS used data from only the 2016 survey. Longitudinal applications through use of mixed models or the generalized estimating equations (GEE) would provide more information for model fitting when data over time is available, as is often the case in potential applications for the SCIP. As the SCIP model is based upon a finite mixture model (FMM) framework, incorporation of other distributions may also be used. For instance, in the BRFSS data, a spike at 30 days (drinkers who binge drink every day) may be modeled through a degenerate distribution built on top of the SCIP via FMM. Additionally, with enough data, spikes at popular intervals (e.g., 5, 10, 15) may be accounted for. Though such flexibility continues to blur the lines between traditional FMM and inflated count modeling, it can allow for more precise modeling when the nature of the outcome is suspected or known. Finally, formal investigation into the identifiability of the SCIP model (and other inflated models) is recommended

Though not the focus of this dissertation, work with the MIP has indicated several areas for growth. One such area is a proportional odds extension of the MIP, which may help with issues of MIP non-convergence, especially for higher levels of c.

In addition to theoretical changes in the models, practical changes in the simulation conditions are also recommended. It would be of interest to compare these three models in a more balanced data situation, with a higher relationship between the predictor and outcome. Implementation of such a study would be straightforward, utilizing a minor change in starting conditions of the simulation. Also of interest may be the relative importance of strong predictors versus balanced groups in group and count prediction accuracy for the three models.

REFERENCES

Agresti, A. (2002). *Categorical Data Analysis* (2nd ed.).

Albert, A. & Anderson, J. A. (1984). On the Existence of Maximum Likelihood Estimates in Logistic Regression Models. *Biometrika*, *71*(1), 1–10. doi:10.2307/2336390

Alshkaki, R. S. A. (2016). On the Zero-One Inflated Poisson Distribution. *International Journal of Statistical Distributions and Applications*, *2*(4), 42–48. doi:10.11648/j.ijsd.20160204.11

Alshkaki, R. S. A. (2017). Moment Estimators of the Parameters of Zero-One Inflated Negative Binomial Distribution. *International Journal of Mathematical, Computational, Physical, Electrical and Computer Engineering*, *11*(1), 38–41.

Atkins, D. C., Baldwin, S. A., Zheng, C., Gallop, R. J., & Neighbors, C. (2013). A tutorial on count regression and zero-altered count models for longitudinal substance use data. *Psychology of Addictive Behaviors*, *27*(1), 166–177. doi:10.1037/a0029508

Dempster, A., Laird, N., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society Series B Methodological*, *39*(1), 1–38. doi:http://dx.doi.org/10.2307/2984875. arXiv: 0710.5696v2

Everitt, B. & Hand, D. (1981). *Finite Mixture Distributions*. Chapman and Hall.

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, *27*(8), 861–874. doi:10.1016/j.patrec.2005.10.010. arXiv: /dx.doi.org/10.1016/j.patrec.200 [http:]

Giles, D. E. (2007). Modeling Inflated Count Data. *International Congress on Modelling*, (January 2007), 919–925.

Hasselblad, V. (1969). Estimation of Finite Mixtures of Distributions from the Exponential Family. *1Journal of the American Statistical Association*, *64*(328), 1459–1471.

He, H., Tang, W., Wang, W., & Crits-Christoph, P. (2014). Structural zeroes and zero-inflated models. *Shanghai archives of psychiatry*, *26*(4), 236–42. doi:10.3969/j.issn.1002-0829.2014.04.008

Hilbe, J. M. (2011). *Negative Binomial Regression* (2nd). Cambridge University Press.

Lambert, D. (1992). Zero-inflated poisson regression, with an application to defects in manufacturing. *Technometrics*, *34*(1), 1–14. doi:10.1080/00401706.1992.10485228

Li, C. S. (2012). Identifiability of zero-inflated Poisson models. *Brazilian Journal of Probability and Statistics*, *26*(3), 306–312. doi:10.1214/10-BJPS137

Lin, T. H. & Tsai, M. H. (2013). Modeling health survey data with excessive zero and K responses. *Statistics in Medicine*, *32*(9), 1572–1583. doi:10.1002/sim.5650

McCullagh, P. (& Nelder, J. A. [John A.]. (1989). *Generalized linear models*. New York : Chapman and Hall. Retrieved from http://encore.unco.edu/iii/encore/record/C%7B%5C_%7D%7B%5C_%7DRb1347936?lang=eng

McLachlan, G. & Peel, D. (2000). *Finite Mixture Models*. Wiley-Interscience.

Melkersson, M. & Olsson, C. (1999). Is visiting the dentist a good habit? Analyzing count data with excess zeros and excess ones. *Umea Economic Studies*, *32*.

Miranda, A. (2010). Department of quantitative social science a double-hurdle count model for completed fertility data from the developing world. *DoQSS*, (1001), 1–45.

Mullahy, J. (1986). Specification and testing of some modified count data models. *Journal of Econometrics*, *33*(3), 341–365. doi:10.1016/0304-4076(86)90002-3

Nelder, J. A. [J. A.] & Wedderburn, R. W. M. (1972). Generalized Linear Models. *Journal of the Royal Statistical Society. Series A (General)*, *135*(3), 370. doi:10.2307/2344614

Perumean-Chaney, S. E., Morgan, C., McDowall, D., & Aban, I. (2013). Zero-inflated and overdispersed: what's one to do? *Journal of Statistical Computation and Simulation*, *83*(9), 1671–1683. doi:10.1080/00949655.2012.668550

R Core Team. (2017). R: A Language and Environment for Statistical Computing. Vienna, Austria: R Foundation for Statistical Computing. Retrieved from https://www.r-project.org/

Raue, A. & Timmer, J. (2013). Identifiability. In W. Dubitzky, O. Wolkenhauer, K.-H. Cho, & H. Yokota (Eds.), *Encyclopedia of systems biology* (p. 937). New York, NY: Springer New York. doi:10.1007/978-1-4419-9863-7_1363

Robin, X., Turck, N., Hainard, A., Tiberti, N., Lisacek, F., Sanchez, J.-C., & Müller, M. (2011). pROC: an open-source package for R and S+ to analyze and compare ROC curves. *BMC Bioinformatics*, *12*(1), 77. doi:10.1186/1471-2105-12-77

Shalizi, C. R. (2012). Mixture Models. *Advanced Data Analysis (lecture notes)*, 390–419.

Shalizi, C. R. (2017). *Advanced data analysis from an elementary point of view*. Retrieved from http://www.stat.cmu.edu/%7B~%7Dcshalizi/ADAfaEPoV/ADAfaEPoV.pdf

Silva, J. S. & Covas, F. (2000). A Modified Hurdle Model for Completed Fertility. *Journal of Population Economics*, *13*, 173–188.

Wald, A. (1943). Tests of statistical hypotheses concerning several parameters when the number of observations is large. *Transactions of the American Mathematical Society*, *54*(3), 426–482.

Zeileis, A., Kleiber, C., & Jackman, S. (2008). Regression Models for Count Data in R. *Journal of Statistical Software*, *27*(8), 1–25. doi:10.18637/jss.v027.i08

Zhang, C., Tian, G.-l., & Ng, K.-w. (2016). Properties of the zero-and-one inflated Poisson distribution and likelihood-based inference methods. *Statistics and its interface*, *9*(1), 11–32.

APPENDIX A

ADDITIONAL TABLES

Coefficients from Fitting the Small Count Inflated Poisson Model to the Behavioral Risk Factor Surveillance System, $c = 2$

	Inflated Count	Poisson Count	Logistic
(Intercept)	-1.489	2.268	1.791
Weight	0.001	0.000	0.000

Coefficients from Fitting the Small Count Inflated Poisson Model to the Behavioral Risk Factor Surveillance System, $c = 6$

	Inflated Count	Poisson Count	Logistic
(Intercept)	-0.992	2.518	2.200
Weight	0.001	0.000	0.000

Error Rate of the Area Under the Curve by Cutoff of Inflation and Sample Size for the Zero-Inflated Poisson Model

			n		
	25	50	100	200	500
$c = 2$	44.5%	20.5%	3.2%	0%	0%
$c = 6$	57.1%	35.1%	10.5%	1.1%	0%
$c = 8$	58.5%	34.8%	9%	1%	0%

Error Rate of the Area Under the Curve by Cutoff of Inflation and Sample Size for the Multinomially-Inflated Poisson Model

			n		
	25	50	100	200	500
$c = 2$	44.3%	20.5%	3.2%	0%	0%
$c = 6$	56.9%	35.1%	10.5%	1.1%	0%
$c = 8$	58.5%	34.8%	9%	1%	0%

APPENDIX B

R CODE

Making a Right-Truncated Poisson Density Function

```
#' @title Right-Truncated Poisson Probability Mass
   Function
#'
#' @description PMF of a right truncated Poisson
   distribution.
#'
#' @param y The vector of counts.
#' @param lambda The vector of means
#' @param c The cutoff of inflation
#'
#' @author Michael Floren
dtpois <- function(y, lambda, c){
  n <- length(y)
  out <- numeric(length=n)
  b <- numeric()
  for(i in 1:n){
    for(j in 0:c){
      b[j+1] <- ((exp(-lambda[i])*lambda[i]^j)/(factorial(
         j)))
    }
    b <- sum(b)
    out[i] = dpois(x=y[i], lambda=lambda[i]) / b
  }
  out[y>c] <- 0 #this PMF is 0 by definition for y>c
  out
}
```

Making the Small Count Inflated Poisson Model

```
#' @title SCIP Function
#'
#' @description Fits a Small Count Inflated Poisson model.
#'
#' @author Michael Floren

scip <- function(y, desmat, c, conv=1e-9, maxit=100){
  ### Functions used later on ###
  tau_s <- function(y, desmat, param, c){ #take the design
      matrix with sampi, koppa, gamma, and the design
     matrix (written short as "gamma" is a system word)
    sam <- param[1:ncol(desmat)]
    kop <- param[(ncol(desmat)+1):(2*ncol(desmat))]
    gam <- param[(2*ncol(desmat)+1):(3*ncol(desmat))]
    lam_s <- exp(desmat %*% sam)
    lam_l <- exp(desmat %*% kop)
    pi <- exp(desmat%*%gam) / (1+exp(desmat%*%gam))
    scip_pmf <- pi*dtpois(y=y, lambda=lam_s, c=c) + (1-pi)
       *dpois(x=y, lambda=lam_l)

    out <- (pi * dtpois(y=y, lambda=lam_s, c=c)) / scip_
       pmf
    out[y>c] <- 0 #if the outcome is greater than c,
       manually set the tau to be zero (it should be
       anyways, but the auto-zero of the dtpois has been
       removed for problems in the logarithm)
    out
  }

  Q <- function(y, desmat, tau, param, c){
    sam <- param[1:ncol(desmat)]
    kop <- param[(ncol(desmat)+1):(2*ncol(desmat))]
    gam <- param[(2*ncol(desmat)+1):(3*ncol(desmat))]
    lam_s <- exp(desmat %*% sam)
    lam_l <- exp(desmat %*% kop)
    pi <- exp(desmat%*%gam) / (1+exp(desmat%*%gam))

    sum(log((pi*dtpois(y=y, lambda = lam_s, c = c))^tau) +
          log(((1-pi)*dpois(x=y, lambda = lam_l))^(1-tau))
             )
  }

  log_pdf_scip <- function(y, desmat, tau, param, c){
    sam <- param[1:ncol(desmat)]
    kop <- param[(ncol(desmat)+1):(2*ncol(desmat))]
    gam <- param[(2*ncol(desmat)+1):(3*ncol(desmat))]
    lam_s <- exp(desmat %*% sam)
    lam_l <- exp(desmat %*% kop)
    pi <- exp(desmat%*%gam) / (1+exp(desmat%*%gam))
    pois <- dpois(x=y, lambda=lam_l) #so that you only
       calculate it once
    sum(log(pi*(dtpois(y=y, lambda=lam_s, c=c) - pois) +
       pois))
  }



  ### Actual implementation ###
  # Creating estimates for sampi, koppa, and gamma.
     Tracking all iterations currently (don't have to do
     it this way: could just wipe the previous iteration
     out on each run...)
  est_param <- matrix(ncol=3*ncol(desmat))
  init_est_sampi <- coef(glm(y~desmat[,-1], family=poisson
     ))
  init_est_koppa <- coef(glm(y~desmat[,-1], family=poisson
     ))
  init_est_gamma <- coef(glm(as.numeric(y<c)~desmat[,-1],
     family=binomial))
  est_param[1,] <- c(init_est_sampi, init_est_koppa, init_
     est_gamma)

  # Creating initial weights: not tracking these for each
     iteration
  tau <- tau_s(y=y, desmat=desmat, param=est_param[1,], c=
     c)

  for(i in 2:maxit){
    #optimization function for parameters (only takes
       parameters as arguments)
    of_param <- function(param){
      Q(y=y, desmat=desmat, tau=tau, param=param, c=c)
    }
    #solving the derivative for the current iteration
    est_param <- rbind(est_param, optim(est_param[i-1,],
       of_param, control = list(fnscale=-1))$par)

    #updating tau
    tau <- tau_s(y=y, desmat=desmat, param = est_param[i
       ,], c=c)

    #checking if convergence has been reached
    check <- max(abs(est_param[i,] - est_param[i-1,]))
    if(check < conv){
      break
    }
  }

  convergence_issue = 0
  if(i >= maxit){
    warning("Convergence␣not␣met")
    convergence_issue = 1
  }

  # setting output
```

```
  est_sam <- est_param[i, 1:ncol(desmat)]
  est_kop <- est_param[i, (ncol(desmat)+1):(2*ncol(desmat)
     )]
  est_gam <- est_param[i, (2*ncol(desmat)+1):(3*ncol(
     desmat))]

  # out
  list(sc = est_sam, lc=est_kop, logistic=est_gam,
     iterations=i, convergence_issue=convergence_issue)
}
```

Generating Truncated Poisson Data

```
#' @title Generation of truncated Poisson data
#'
#' @description This function generates truncated Poisson
   data, with a truncation at c (counts can include c).
#'
#' @param n The number of data points to be generated.
#' @param lambda The mean parameter for the truncated
   Poisson.
#' @param c The cutoff of inflation.
#'
#' @details This generates data from a right truncated
   Poisson distribution, where generated y's are in the
   range \eqn{0 \le y \le c}.
#'
#' @examples
#' #show a histogram of right truncated Poisson data
#' hist(tp_dat(n=3000, lambda=3, c=5))
#'
#' @author Michael Floren
#' @md

tp_dat <- function(n, lambda, c){ #truncated Poisson
  if(n != length(lambda) & length(lambda)>1)
    stop("The␣length␣of␣lambda␣must␣be␣1␣or␣match␣n.") #
       only support if a different lambda is given for
       each n, or the same lambda is given for all of them
       ...
  if(c <= lambda)
    stop("You're␣being␣an␣idiot␣(lambda␣is␣greater␣than␣c)
       ") #doesn't make sense for the mean of the
       distribution to be larger than the cutoff (tell
       user their being an idiot)

  if(length(lambda)!=n){ #this would be the length(lambda)
     ==1 and n>1 case...
    lambda <- rep(lambda, n)
  }

  out <- numeric()
  for(j in 1:length(lambda)){
    probs <- numeric()
    for(i in 0:c){
      probs[i+1] <- ((exp(-lambda[j])*lambda[j]^i)/(
         factorial(i)))
    }
    probs = probs/(sum(probs))
    #sample(seq(0,c), n, replace=TRUE, prob=probs)
    out[j] <- sample(0:c, size=1, prob=probs)
  }

  out
}
```

Generating Small Count Inflated Poisson Data

```
#' @title Generation of Small Count Inflated Poisson Data
#'
#' @description This function generates small count
   inflated Poisson (SCIP) data, giving a list of values
   and group membership.
#'
#' @param n The number of data points to be generated.
#' @param desmat The design matrix to use for data
   generation.
#' @param sam A vector of parameters to use for the small
   count distribution.
#' @param kop A vector of parameters to use for the large
   count distribution.
#' @param gam A vector of parameters to use for the
   binomial distribution (1 is small count group).
```

```
#' @param c The cutoff of inflation
#'
#' @details This function takes a set of arguments and
   returns a vector of two elements. The first element is
   the generated count from the SCIP distribution, the
   second element is the group of membership (for
   dissertation purposes). The returning of the group
   membership element may be removed at a later date.
#'
#' @return This function currently returns a list of two
   vectors
#'   \item{y}{A vector of the generated count variables}
#'   \item{g}{A vector of the group each generated count
   belongs to}
#'
#' @examples
#' #show a histogram of small count inflated data
#' hist(scip_dat(n=3000, desmat=matrix(rep(1, 3000), ncol
   =1), sampi=log(.1), koppa=log(4), gamma=log(1), c=3)$y)
#'
#' @author Michael Floren

scip_dat <- function(n, desmat, sam, kop, gam, c){
  if(!all.equal(ncol(desmat), length(sam), length(kop),
     length(gam)))
    stop("Dimensional␣issues␣between␣design␣matrix,␣sampi,
       ␣koppa,␣and␣gamma.␣Please␣check␣dimensions␣(ncol␣of
       ␣design␣matrix␣should␣match␣length␣of␣sampi,␣koppa,
       ␣and␣gamma).")

  lam_s <- exp(desmat%*%sam) #lambda_s
  lam_l <- exp(desmat%*%kop) #lambda_l
  pi <- exp(desmat%*%gam) / (1+exp(desmat%*%gam)) #pi_s

  obs <- numeric()
  group <- numeric()
  for(i in 1:n){
    group[i] <- rbinom(1,1,pi)
    if(group[i] == 1){
      obs[i] <- tp_dat(n=1, lambda=lam_s[i], c=c)
    } else{
      obs[i] <- rpois(n=1, lambda=lam_l[i])
    }
  }
  list(y=obs, g=group)
}
```

Making the Multinomially Inflated Poisson Model

```
#' @title Multinomially Inflated Poisson Model
#'
#' @description This is the multinomially inflated Poisson
    distribution as demonstrated by Giles (2007).
#'
```

```
mip <- function(y, desmat, c){
  # log-liklihood given outcome, parameters, and design
     matrix...
  ll <- function(y, desmat, param){
    #reading in gammas and betas from the long list of
       parameters
    p <- ncol(desmat)
    c <- length(param)/ncol(desmat) - 2
    param_mat <- matrix(ncol=c+3, nrow=p)
    for(i in 1:(c+1)){ #so this is 0:c (doesn't do gamma_J
       )
      param_mat[,i] <- param[(i*p-(p-1)):(i*p)]
    }
    param_mat[,c+2] <- rep(0,p)#this is gamma_J
    param_mat[,c+3] <- param[((c+2)*p-(p-1)):((c+2)*p)]
    colnames(param_mat) <- c(paste0("gamma",0:(c+1)), "
       beta") #remember: gamma_(c+1) is gamma_J (0's)

    # prepping omega calculation
    denom_for_omega_mat <- matrix(ncol=c+1, nrow=length(y)
       )
    for(i in 0:c){ #for all the gammas except for gamma
       zero, cause this is what the denom sum is. gamma_(c
       +1) is gamma_J (so including J). Gamma_J is zero.
      denom_for_omega_mat[,i+1] <- exp(desmat %*% param_
         mat[,paste0("gamma",i)])
    }
```

```
denom_for_omega <- 1 + apply(denom_for_omega_mat, 1,
   sum)

# calculating omegas : something weird going on here (
   matches the long hand through the log-likelihood
   output) ********************************************
omega_mat <- matrix(ncol=c+2, nrow=length(y))
for(i in 0:(c+1)){ #creating an omega_(c+1) (aka,
   omega_J)
  omega_mat[,i+1] <- exp(desmat %*% param_mat[,paste0(
     "gamma",i)])/denom_for_omega #right now, desmat %
     *% param (not exponentiated) looks to be a good
     prediction of the weights (for whatever reason) (
     if you use the optimized values for your
     parameters and run through this line by line...).
      Almost just proportional: cbind(desmat[,2],
     desmat%*%param_mat[,1], desmat[,2]-desmat%*%param
     _mat[,1], (desmat[,2]-desmat%*%param_mat[,1])/
     desmat%*%param_mat[,1])
}
colnames(omega_mat) <- paste0("omega", 0:(c+1)) #
   remember: omega_(c+1) = omega_J

#calculating coefficient for poisson
coef_p <- 1-apply(omega_mat[,-ncol(omega_mat), drop=
   FALSE], 1, sum)# take off the last column (the c+2
   nd column (omega_(c+1) aka omega_J)) from omega mat
  #calculating pois
  lambda <- exp(desmat %*% param_mat[,"beta"])
  pois <- (exp(-lambda)*lambda^y)/(factorial(y))

  ll_out_vec <- numeric(length=c+2) #this is each of the
      sums in the log likelihood from Giles (0 to J (
     which is c+1)), listed as a vector (eventually will
      be summed for the log-likelihood).
  for(i in 0:c){
    ll_out_vec[i+1] <- sum((y==i)*log((omega_mat[,paste0
       ("omega",i)] + (coef_p)*pois))) #add everything
       up (add 0's unless y is i)
  }
  ll_out_vec[c+2] <- sum((y>c)*log((coef_p)*pois)) #
     doing the poisson alone by hand
  ll_out <- sum(ll_out_vec) #matches with longhand...
  ll_out
}

p <- ncol(desmat)

#starting guess for parameters (just use 0 vector I
   guess?)
initial_guess_param <- rep(0, p*(c+2)) #gammas from 0
   through c, then beta (each has p parameters)...
  of <- function(params){ #optimization function (just a
     function of parameters)
    ll(y=y, desmat=desmat, param=params)
  }

  full_out <- optim(initial_guess_param, of, control =
     list(fnscale=-1))
  est <- full_out$par
  convergence_issue <- full_out$convergence!=0



  out <- list()
  for(i in 1:(c+1)){ #so this is gamma_0:c
    out[[i]] <- est[(i*p-(p-1)):(i*p)]
    names(out)[i] <- paste0("logistic",i-1)
  }
  out[[c+2]] <- est[((c+2)*p-(p-1)):((c+2)*p)]
  names(out)[c+2] <- "beta"
  out$convergence_issue <- convergence_issue
  out
}
```

Designing a Function for Model Fitting for the Dissertation Simulation

```
#' @title Simulation Function for Dissertation
#'
#' @description This function will run the simulation for
   my dissertation. Some of the aspects of the function
   are specifically designed for this dissertation (such
   as population parameters), and won't be included as
   arguments.
#'
#'
#'

sim <- function(n, c, k=1000, meta_time_unit="s", progress
   =TRUE){
  p <- 2
  mean_of_IV <- 197.7828
  sd_of_IV <- 40.21599

  # setting different true values depending on c
  if(c==2){
    true_sampi <- c(-1.4890070090395,
       0.000896026713969103)
    true_koppa <- c(2.26780053964254,
       -0.000373799698134193)
    true_gamma <- c(1.79118900609313,
       -0.000338077141916478)
  } else if (c==6){
    true_sampi <- c(-0.991816836348344,
       0.000632613090524932)
    true_koppa <- c(2.51780279408921,
       -0.000375747468669122)
    true_gamma <- c(2.1998559804243, -0.00025944850892789)
```

```
} else if (c==8){
  true_sampi <- c(-0.989457553026938,
     0.000623795679864685)
  true_koppa <- c(2.51994084026393,
     -0.000385423153504593)
  true_gamma <- c(2.20836925037315,
     -0.000300453184670207)
} else{
  stop(paste0("Population␣parameters␣for␣c=",c,"␣not␣set
     .␣Simulation␣stopped."))
}



#creating the place to save all of the results. These
   will have elements named for each piece of the output
    (e.g., sc, lc, logistic), where each row is a
   different run
zip_results <- list()
scip_results <- list()
mip_results <- list()
meta <- list()

#creating list elements for each piece of output
for(i in 1:2){
  zip_results[[i]] <- matrix(ncol=p, nrow=k) #columns is
      the number of parameters (in this case, intercept
     and slope)
}
for(i in 1:3){
  scip_results[[i]] <- matrix(ncol=p, nrow=k)
}
for(i in 1:(c+2)){ #0-c and beta
  mip_results[[i]] <- matrix(ncol=p, nrow=k)
}
names(zip_results) <- c("count", "logistic") #the names
   and order from zeroinfl
names(scip_results) <- c("sc","lc","logistic")
names(mip_results) <- c(paste0("logistic",0:c),"beta")

#creating a place to store meta-information (iterations
   for SCIP, runtime for all) for each run and overall.
   NA's by default (don't want 0's by default, as they
   may not get noticed as errors (if they aren't
   overwritten, for some reason))
zip_results$runtime <- as.numeric(rep(NA,k))
zip_results$convergence_issue <- as.numeric(rep(NA,k))
mip_results$runtime <- as.numeric(rep(NA,k))
mip_results$convergence_issue <- as.numeric(rep(NA,k))
scip_results$runtime <- as.numeric(rep(NA,k))
scip_results$iterations <- as.numeric(rep(NA,k))
scip_results$convergence_issue <- as.numeric(rep(NA,k))
```

```
meta$runtime <- as.numeric(rep(NA,k))

#creating an error log for each of the methods
zip_results$error_log <- data.frame(error_bin=rep(0,k),
   error_msg=character(length=k), stringsAsFactors=FALSE
   )
mip_results$error_log <- data.frame(error_bin=rep(0,k),
   error_msg=character(length=k), stringsAsFactors=FALSE
   )
scip_results$error_log <- data.frame(error_bin=rep(0,k),
    error_msg=character(length=k), stringsAsFactors=
   FALSE)

#performing the trials
for(i in 1:k){
  #starting the run clock
  meta_starttime <- Sys.time()



  # generating the dsign matrix
  desmat <- cbind(rep(1,n), rnorm(n, mean=mean_of_IV, sd
     =sd_of_IV)) #these are the population parameters
     from weight in the BRFSS (agam, manually entered)

  # generating data
```

```
y <- scip_dat(n=n, desmat=desmat, sam=true_sampi, kop=
   true_koppa, gam=true_gamma, c=c)$y




### Fitting the ZIP ####
# Making a dataset for the ZIP
zip_dat <- cbind(y=y,x=desmat[,-1])

# fitting the zip model
#library(pscl) #eventually need to take this out
zip_start<-Sys.time()
zip_attempt <- tryCatch(zip_mod <- pscl::zeroinfl(y~.,
    data=as.data.frame(zip_dat)), error=function(e) c(
   error_bin=1,e))
zip_stop<-Sys.time()

if("error_bin" %in% names(zip_attempt)){ #if an error
  zip_results$error_log$error_bin[i] <- zip_attempt$
     error_bin
  zip_results$error_log$error_msg[i] <- zip_attempt$
     message
} else { #record the fit
  zip_results$count[i,] <- zip_mod$coefficients$count
  zip_results$logistic[i,] <- zip_mod$coefficients$
     zero
  zip_results$runtime[i] <- difftime(zip_stop, zip_
     start, units=meta_time_unit)
  zip_results$convergence_issue[i] <- 1-zip_mod$
     converged
}



### Fitting the SCIP ####
scip_start<-Sys.time()
scip_attempt <- tryCatch(scip_mod <- scip(y=y, desmat=
   desmat, c=c), error=function(e) c(error_bin=1,e))
scip_stop<-Sys.time()

#error checking and recording results (if no error)
if("error_bin" %in% names(scip_attempt)){ #if an error
  scip_results$error_log$error_bin[i] <- scip_attempt$
     error_bin
  scip_results$error_log$error_msg[i] <- scip_attempt$
     message
} else { #record the fit
  scip_results$sc[i,] <- scip_mod$sc
  scip_results$lc[i,] <- scip_mod$lc
  scip_results$logistic[i,] <- scip_mod$logistic
  scip_results$runtime[i] <- difftime(scip_stop, scip_
     start, units=meta_time_unit)
  scip_results$iterations[i] <- scip_mod$iterations
  scip_results$convergence_issue[i] <- scip_mod$
     convergence_issue
}



### Fitting the MIP ####
mip_start<-Sys.time()
mip_attempt <- tryCatch(mip_mod <- mip(y=y, desmat=
   desmat, c=c), error=function(e) c(error_bin=1,e))
mip_stop<-Sys.time()

#error checking and recording results (if no error)
if("error_bin" %in% names(mip_attempt)){ #if an error
  mip_results$error_log$error_bin[i] <- mip_attempt$
     error_bin
  mip_results$error_log$error_msg[i] <- mip_attempt$
     message
} else { #record the fit
  # as they should all be named the same, using the
     names from one to iterate over the other...
  for(j in c(paste0("logistic",0:c),"beta")){ #the
     names of the mip_results columns that don't have
     to do with meta stuff...
    mip_results[[j]][i,] <- mip_mod[[j]]
  }
  mip_results$runtime[i] <- difftime(mip_stop, mip_
     start, units=meta_time_unit)
    mip_results$convergence_issue[i] <- mip_mod$
       convergence_issue
  }



  #ending the run clock
  meta_endtime <- Sys.time()

  #Recording meta-time (time for the full run)
  meta$runtime[i] <- difftime(meta_endtime, meta_
     starttime, units = meta_time_unit)

  if(progress)
    cat(paste0("\r",round(i/k*100),"%␣complete␣(working␣
       on␣", toOrdinal::toOrdinal(i+1),"␣run).␣Last␣run␣
       took␣", round(meta$runtime[i],2), "␣", meta_time_
       unit,".␣Average␣run␣is␣", round(mean(meta$runtime
       , na.rm=TRUE),2), "␣", meta_time_unit, ".␣
       Predicted␣completion␣time␣is␣", Sys.time()+round(
       mean(meta$runtime, na.rm=TRUE))*(k-i),"."))
}



list(zip_results=zip_results, scip_results=scip_results,
    mip_results=mip_results, meta=meta, n=n, c=c)
```

```
}
```

Designing a Function for Prediction for the Dissertation Simulation

```
#' @title Prediction Function for Dissertation
#'
#'
#' @description This function will run the prediction for
   my dissertation. The goal is to establish prediction
   accuracy for each of the models. So, we'll need to
   generat a design matrix, generate some data, have a
   predicted value based on the design matrix, have an
   actual value, then also have an actual group vs a
   predicted group for each of the models...
#'
#' I want to matrix multiple all of the things, then use
   logistic to determine which one I should actually use
   ...
#'
#' @param n The sample size to use. This should be 20\% of
     the actual data used in the simulation.
#' @param c The cutoff to use. This should match that used
     in the simulation.
#' @param sim_dat Data from the simulation function.
#'
#' @author Michael Floren
pred <- function(sim_dat){
  ### initial parameters ###
```

```
percent_for_test <- .2 #test sample size is 20% of the
   actual sample. May choose to use 25% instead, but not
    a big deal...
n <- sim_dat$n*percent_for_test
c <- sim_dat$c

### functions ###
mse <- function(y,yhat){ #mean squared error
  sum((yhat-y)^2)/length(y)
}

#the next couple functions need a threshhold to compare
   to
thresh <- .5

pc <- function(g, ghat){ #percent correct, given
   decimals for ghat. Remember: 1 is small count, 0 is
   large count (this works either way, but g and ghat
   have to be consistent...)
  sum((ghat>thresh)==g)/length(g)
}

auc <- function(g, ghat){
  tryCatch(pROC::auc(pROC::roc(as.numeric(g),as.numeric(
     ghat))),error=function(x)NA)
}
```

```
# all overall numbers should be the same as for the
   simulation
k <- nrow(sim_dat$zip_results$count) #the number of rows
    for any of these should be the same...
p <- ncol(sim_dat$zip_results$count)
mean_of_IV <- 197.7828
sd_of_IV <- 40.21599
if(c==2){
  true_sampi <- c(-1.4890070090395,
     0.000896026713969103)
  true_koppa <- c(2.26780053964254,
     -0.000373799698134193)
  true_gamma <- c(1.79118900609313,
     -0.000338077141916478)
} else if (c==6){
  true_sampi <- c(-0.991816836348344,
     0.000632613090524932)
  true_koppa <- c(2.51780279408921,
     -0.000375747468669122)
  true_gamma <- c(2.1998559804243, -0.00025944850892789)
} else if (c==8){
  true_sampi <- c(-0.989457553026938,
     0.000623795679864685)
  true_koppa <- c(2.51994084026393,
     -0.000385423153504593)
  true_gamma <- c(2.20836925037315,
     -0.000300453184670207)
```

```
} else{
  stop(paste0("Population␣parameters␣for␣c=",c,"␣not␣set
     .␣Simulation␣stopped."))
}

# making generic lists to hold the information
verbose_results <- list()
results <- list()

#defining matrices for the results
for(i in 1:3){
  results[[i]] <- as.data.frame(matrix(ncol=3, nrow=k))
  colnames(results[[i]]) <- c("zip","mip","scip")
}
names(results) <- c("mse","percent_cor", "auc")
results$weighted_mse <- as.data.frame(matrix(ncol=3,
   nrow=k))
colnames(results$weighted_mse) <- c("zip", "scip", "mip"
   )

for(i in 1:k){
  # generating data for the run
  desmat <- cbind(rep(1,n), rnorm(n, mean=mean_of_IV, sd
     =sd_of_IV))
  actual <- scip_dat(n=n, desmat=desmat, sam=true_sampi,
      kop=true_koppa, gam=true_gamma, c=c)
```

```
verbose_results[[i]] <- as.data.frame(matrix(ncol=p+8,
    nrow=n)) #the verbose results for the iteration/
   run. Columns: p for desmat, 2 for out (group and
   count), 2 for predictions (count and group) for ZIP
   , SCIP, and MIP
colnames(verbose_results[[i]]) <- c(paste0("desmat",
   seq(0,ncol(desmat)-1)), "y", "g", "yhat_zip", "ghat
   _zip", "yhat_scip", "ghat_scip", "yhat_mip", "ghat_
   mip")

#setting info for the run
for(j in 1:ncol(desmat))
  verbose_results[[i]][,paste0("desmat",j-1)] <-
     desmat[,j]

verbose_results[[i]]$y <- actual$y
verbose_results[[i]]$g <- actual$g




#### Predictions: ZIP ####
zip_group_pred <- exp(desmat%*%sim_dat$zip_results$
   logistic[i,])/(1+exp(desmat%*%sim_dat$zip_results$
   logistic[i,])) #exp(x\beta)/(1+exp(x\beta))
zip_count_pred <- exp(desmat%*%sim_dat$zip_results$
   count[i,])
```

```
#setting the weighted mse
results$weighted_mse$zip[i] <- mse(y=actual$y,
                                   yhat=zip_group_pred
                                     *0 + (1-zip_
                                     group_pred)*zip_
                                     count_pred)

#for logistic results that point towards zero, set the
   predicted count to 0 (use same threshold as above)
zip_count_pred[zip_group_pred>thresh] <- 0

#set the verbose results information
verbose_results[[i]]$yhat_zip <- zip_count_pred
verbose_results[[i]]$ghat_zip <- zip_group_pred

#set the results information
results$mse$zip[i] <- mse(y=actual$y, yhat=zip_count_
  pred)
results$percent_cor$zip[i] <- pc(g=actual$g, ghat=zip_
  group_pred)
results$auc$zip[i] <- auc(g=actual$g, ghat=zip_group_
  pred)



#### Predictions: SCIP ####
```

```
# Just treating the logistic piece as the group
   prediction...
scip_group_pred <- exp(desmat%*%sim_dat$scip_results$
   logistic[i,])/(1+exp(desmat%*%sim_dat$scip_results$
   logistic[i,])) #the logistic piece
scip_sc_pred <- exp(desmat%*%sim_dat$scip_results$sc[i
   ,])
scip_lc_pred <- exp(desmat%*%sim_dat$scip_results$lc[i
   ,])
scip_count_pred <- ifelse(scip_group_pred>thresh, scip
   _sc_pred, scip_lc_pred)

#set the verbose results information
verbose_results[[i]]$yhat_scip <- scip_count_pred
verbose_results[[i]]$ghat_scip <- scip_group_pred

#set the results information
results$mse$scip[i] <- mse(y=actual$y, yhat=scip_count
   _pred)
results$percent_cor$scip[i] <- pc(g=actual$g, ghat=
   scip_group_pred)
results$auc$scip[i] <- auc(g=actual$g, ghat=scip_group
   _pred)
results$weighted_mse$scip[i] <- mse(y=actual$y,
                                    yhat= scip_group_
                                       pred*scip_sc_
                                       pred + (1-scip_
                                       group_pred)*
                                       scip_lc_pred)



#### Predictions: MIP ####
names_of_logistic_pieces <- paste0("logistic", 0:c)

#doing the denominator first
denom_for_logistic_mat <- matrix(ncol=c+1, nrow=n)
for(j in 1:(c+1)){ #for all the gammas except for
   gamma zero, cause this is what the denom sum is.
   gamma_(c+1) is gamma_J (so including J). Gamma_J is
    zero.
  denom_for_logistic_mat[,j] <- exp(desmat %*% sim_dat
     $mip_results[[names_of_logistic_pieces[j]]][i,])
}
denom_for_logistic <- 1 + apply(denom_for_logistic_mat
   , 1, sum)

#determining the logistic prediction pieces
mip_logistic <- as.data.frame(matrix(ncol=c+2, nrow=n)
   )
colnames(mip_logistic) <- c(names_of_logistic_pieces,
   "logisticJ")
for(j in 1:(c+1)){
  mip_logistic[,j] <- exp(desmat%*%sim_dat$mip_results
     [[names_of_logistic_pieces[j]]][i,])/denom_for_
     logistic
}
mip_logistic$logisticJ <- 1-apply(mip_logistic[,-ncol(
   mip_logistic)], 1, sum)
mip_group_pred <- 1-mip_logistic$logisticJ #the
   probability of being in the small count is 1-\pi_l
mip_count_pred <- exp(desmat%*%sim_dat$mip_results$
   beta[i,])

#setting the weighted MSE information
weighted_mip <- mip_logistic
for(j in 1:(c+1))
  weighted_mip[,j] <- mip_logistic[,j]*(j-1) #the
     column times its count
weighted_mip$logisticJ <- mip_logistic$logisticJ * mip
   _count_pred #the count probability times the count
   prediction
results$weighted_mse$mip[i] <- mse(y=actual$y,
                                   yhat=apply(weighted
                                      _mip, 1, sum))

#for each row, check which logistic piece is higher
group <- apply(mip_logistic, 1, function(x) which(x==
   max(x))[1])-1 #if multiple are tied, just take the
```

```
       first one... subtract 1 to match the column index
       with the meaning (first column is a count of zero)
    for(j in 0:c){ #don't do J, as if the max group is J
       we want to leave it alone. Only change for
      mip_count_pred[group==j] <- j
    }

    #set the verbose results information
    verbose_results[[i]]$yhat_mip <- mip_count_pred
    verbose_results[[i]]$ghat_mip <- mip_group_pred

    #set the results information
    results$mse$mip[i] <- mse(y=actual$y, yhat=mip_count_
       pred)
    results$percent_cor$mip[i] <- pc(g=actual$g, ghat=mip_
       group_pred)
    results$auc$mip[i] <- auc(g=actual$g, ghat=mip_group_
       pred)
  }

  list(verbose_results=verbose_results, results=results)
}
```