ZERO-INFLATED NEGATIVE BINOMIAL (ZINB) REGRESSION MODEL FOR OVER-DISPERSED COUNT DATA WITH EXCESS ZEROS AND REPEATED MEASURES, AN APPLICATION TO HUMAN MICROBIOTA SEQUENCE DATA

by

RUI FANG

Bachelor of Medicine, North China Coal Mining Medical College, 2008

A thesis submitted to the
Faculty of the Graduate School of the
University of Colorado in partial fulfillment
of the requirements for the degree of
Master of Science
Biostatistics Program
2013
This thesis for the Master of Science degree by

Rui Fang

has been approved for the

Biostatistics Program

by

Gary K. Grunwald, Chair

Brandie D. Wagner

Gary O. Zerbe

Date 4/19/2013
Fang, Rui (M. S., Biostatistics)

Zero-Inflated Negative Binomial (ZINB) Regression Model for Over-dispersed Count Data with Excess Zeros and Repeated Measures, an Application to Human Microbiota Sequence Data

Thesis directed by Assistant Professor Brandie D. Wagner.

**ABSTRACT**

In many biomedical applications, count outcomes are fairly common and often these count data have a large number of zeros. Zero-inflated regression models are useful for analyzing such data. Moreover, the non-zero observations may be over-dispersed in relation to the Poisson distribution, biasing parameter estimates and underestimating standard errors. In such a circumstance, a Zero-Inflated Negative Binomial (ZINB) regression model better accounts for these characteristics compared to a Zero-Inflated Poisson (ZIP). In addition, repeated measures are often collected on the same individual subjects, random effects are introduced to account for the within subject variation. The objective of this thesis is to present a ZINB regression model for over-dispersed count data with excess zeros and repeated measures. This mixture model contains components to model the probability of excess zero values and the negative binomial parameters, allowing for repeated measures using independent random effects between these two components. Parameter estimation is achieved by maximizing an appropriate likelihood function using a stable numerical procedure such as the Newton-Raphson algorithm. A small simulation study was performed for model verification and application of the proposed model is applied to data from a human microbiota study.
The form and content of this abstract are approved.

Approved: Brandie D. Wagner
ACKNOWLEDGEMENTS

This thesis research was partially supported by the Gastrointestinal Eosinophil Diseases Program at University of Colorado Denver and Children’s Hospital Colorado.

I would like to acknowledge Dr. Brandie Wagner, Dr. Gary Grunwald and Dr. Gary Zerbe who supervised and guided this thesis.
# TABLE OF CONTENTS

## CHAPTER

I. INTRODUCTION...........................................................................................................1  
1.1 Introduction of Zero-inflated Regression and Literature Review................1  
1.2 Introduction of Microbiota ..................................................................................4  
1.3 The Characteristics of Microbiota Sequence Data..........................................5  

II. THE ZERO-INFLATED NEGATIVE BINOMIAL (ZINB) REGRESSION MODEL .................................................................8  
2.1 ZINB Regression Model ...................................................................................8  
2.2 Single-level ZINB Regression Model with Random Effects .........................9  

III. SIMULATION STUDY AND RESULTS .................................................................11  

IV. APPLICATION TO HUMAN MICROBIOTA SEQUENCE DATA ..........13  
4.1 Study Description ...........................................................................................13  
4.2 The ZINB Regression Model for Human Microbiota Sequence Data ..........16  
4.3 Results ............................................................................................................18  

V. DISCUSSION .........................................................................................................25  

REFERENCES ...........................................................................................................27  

APPENDIX  
A. SAS CODE .........................................................................................................30
LIST OF TABLES

1. Simulation results for the single-level random-effects ZINB regression model using 1000 simulated datasets from the model given in section 3 with each of two sets of true parameter values ..........................................................12

2. Comparison between model with independent random effects and model with correlated random effects fitted for a subset of the organisms selected from the motivating dataset .................................................................19

3. Results of ZINB regression model with independent random effects for three organisms selected from the motivating dataset ..................................................20

4. ZINB model estimates and p-values for three organisms selected from the motivation dataset .................................................................20
LIST OF FIGURES

FIGURE

1. Variable sequencing effort ..................................................................................7

2. Esophageal String Test (EST) ...........................................................................15

3. 16S amplification and detection of nasal, oral and esophageal (EST and biopsy) microenvironments .................................................................16

4. Distribution of human microbiota sequence counts for each of four organisms classified at genus level .................................................................18

5. Empirical and fitted ZINB distributions of the human microbiota sequence data for each of four organisms classified at genus level ..........................23

6. Manhattan plots for the comparisons across all taxa ......................................24
CHAPTER I

INTRODUCTION

1.1 Introduction of Zero-inflated Regression and Literature Review

In biomedical or health research, outcomes of interest often consist of count variables. For such count data, the standard framework for explaining the relationship between the outcome variable and a set of explanatory variables includes the Poisson and negative binomial regression models. However, the basic Poisson regression model forces the conditional variance of the outcome to equal the conditional mean, which is of limited use in real life. The negative binomial regression can be written as an extension of Poisson regression and it enables the model to have greater flexibility in modeling the relationship between the conditional variance and the conditional mean compared to the Poisson model. Also, an often encountered characteristic of count data, is the number of zeros in the sample can exceed the number of zeros predicted by either Poisson or negative binomial model, and this is of interest because zero counts frequently have special status.

In general, Hurdle models and zero-inflated models are used for modeling count data with a preponderance of zeros. The hurdle model [1] is a two component model in which one component models the probability of zero counts and the other component uses a truncated Poisson/negative binomial distribution that modifies an ordinary distribution by conditioning on a positive outcome. The zero-inflated model has a distribution that is a mixture of a binary distribution that is degenerate at zero and an
ordinary count distribution such as Poisson or negative binomial. The Hurdle model considers the zeros to be completely separate from the non-zeros. The zero-inflated model is similar to the Hurdle model; however, it permits some of the zeros can analyzed along with the non-zeros. The choice of the zero-inflated model in this thesis is guided by the researcher’s beliefs about the source of the zeros. There are two distinct processes driving zeros, one is sampling zeros which occur by chance and can be assumed a result of a dichotomous process, and the other one is structural zeros (true zeros) which are inevitable and are part of the counting process. Beyond this substantive concern, the choice should be based on the model providing the closest fit between the observed and predicted values.

The Zero-Inflated Poisson (ZIP) regression model was first introduced by Lambert [2], and she applied this model to the data collected from a quality control study, in which the response typically is the number of defective products in a sample unit. Further applications for the ZIP regression model can be found in dental epidemiology [3], occupational health [4], and children’s growth and development [5]. In practice, even after accounting for zero-inflation, the non-zero part of the count distribution is often over-dispersed. In this case, Greene [6] described an extended version of the negative binomial model for excess zero count data, the Zero-Inflated Negative Binomial (ZINB), which may be more appropriate than the ZIP. It has been established that the ZIP parameter estimates can be severely biased if the non-zero counts are over-dispersed in relation to the Poisson distribution.

Often because of a hierarchical study design or data collection where the observations are either clustered or repeated outcomes from individual subjects, zero-
inflated regression models are extended to include random effects. The random-effects model that accounts for the subject to subject variation is a way to directly model the correlation among the repeated measures within a subject. Recently, extension of the ZIP model to incorporate random effects into the linear predictors can be found in Hall [7], Wang et al. [8], Hur et al. [9] and Yau and Lee [10]. In Hall’s model, a random intercept was added to only the Poisson component (i.e., not for the binary component) and the parameter estimation was obtained by maximizing an appropriate log-likelihood function using an EM algorithm. Whereas, Yau and Lee included two distinct, independent random effects for Poisson and binary components, but they used the Newton-Raphson algorithm to separately estimate the parameters. Use of Newton-Raphson algorithm, as opposed to the EM, is advantageous due to its quicker convergence and also because it readily provides standard errors for all parameter estimates. In both cases, the random effects were assumed to be normally distributed. In a similar manner, Yau et al. [11] developed the single-level ZINB mixed regression for modeling over-dispersed count data with extra zeros. In this paper, two random effects were included for the negative binomial and binary components to account for the dependency between observations within the same cluster. For simplicity, the two random effects are assumed to be independent and normally distributed. Parameter estimation was achieved by maximizing an appropriate log-likelihood function using an EM algorithm. Additionally, Lee et al. [12] extended the ZIP with random-effects to multi-level ZIP regression to model multi-level clustered count data, and Moghimbeigi et al. [13] developed multi-level ZINB regression for modeling over-dispersed count data with extra zeros.
Additional extensions that will not be the focus of this work include a special case of zero-inflated models, the zero-inflated (τ) model. This model is applicable when the covariates for both the binary components and ordinary counts are the same and their effects are functionally related. Under these circumstances, the zero-inflated (τ) model provides a more parsimonious model compared to the general zero-inflated model. However, this assumption of common covariate effects is rarely satisfied in practice. Depending on the relationship between the probability of zero counts and the Poisson mean, Lambert [2] described two variants: a ZIP and a ZIP (τ) model. In Hur et al. [9], they extended these models for the case of clustered data (e.g. patients observed within hospitals) and described random-effects ZIP and ZIP (τ) models. The random-effects ZINB and ZINB (τ) models can be easily obtained by replacing the Poisson distribution function with the negative binomial one [6]. We are not aware of applications with random-effects ZINB (τ) models.

1.2 Introduction of Microbiota

The community of microbial organisms and their functions constitute the microbiome. A microbiome is the totality of microbes, their genetic elements (genomes), and environmental interactions in a particular environment. The total microbial cells found on human associated communities are about 100 trillion, which may exceed the total number of cells making up the human body by a factor of ten-to-one [14]. Thus it can significantly affect many aspects of human physiology. For example, in healthy individuals, the microbiome provides a wide range of metabolic functions that humans lack, making their presence advantageous [15]. In diseased individuals, altered
microbiome is associated with diseases such as inflammatory bowel disease (IBD) [16]. Thus studying the human microbiome is an important task that has been undertaken by initiatives such as the Human Microbiome Project [17]. Microbiome studies typically are categorized as either microbiota or metagenomics; the former focuses on a single/specific gene while the latter focuses on all genes of all members of the community.

Until recently, bacteria were quantified using culturing methods which required prior knowledge of the organisms present and their growth rates. With the advent of DNA-based sequencing technology, the quantification of all organisms present in the community can now be performed. The process starts with the collection of human-associated samples and successful extraction of a single bacterial gene (16S rRNA). The DNA sequences from sections of the gene are obtained using next generation sequencing technologies and are aligned to a reference database for organism identification (taxonomy category). The number of sequences for a single organism is then counted.

1.3 The Characteristics of Microbiota Sequence Data

The microbiota sequence data are high-dimensional with added complexity. They consist of non-negative, highly skewed sequence counts with a large number of zeros. The number of zeros in the dataset is a result of combining counts from different samples. Samples collected from different groups provide unique organisms, and if an organism is detected in one but not another sample, insertion of zeros is performed. The absence of a count for an organism can be due to the fact that the organism simply isn’t present in the sample (structure zeros) or that the organism is present but happens not to appear in the
sample (sampling zeros). Sample sizes are often relatively small in many studies, where some asymptotic assumptions may not be reasonable and the statistical power is decreased. Limited samples are due to the expense of patient recruitment, sample collection and sequencing cost. In addition, the number of total sequences varies from sample to sample (Figure 1) and is a result of variable sequencing effort which cannot be controlled nor is it associated with the total bacterial load of a sample. A sample with a large number of total sequence counts will have a larger number of sequence counts for an organism compared to another sample with fewer total sequence counts. This constraint results in the community structure information gained from this technology to be in relative quantities rather than absolute quantities. The relative abundance/proportion, the sequence count divided by the total number of sequences obtained for each organism within a sample, is the common unit of measurement for microbiota analyses. Modeling binomial proportions can be cumbersome. Luckily count distributions can be used to approximate binomial proportions with the use of an offset, where the log of total sequences is added to the predictor function and forced to have a coefficient of 1 (more details are shown in section 4.2).

The focus of this thesis is on modeling over-dispersed count data with excess zeros and repeated measures. After briefly reviewing the standard ZINB regression model, a single-level ZINB regression model incorporating random effects to account for data dependency and over-dispersion is described in detail in Section 2. Section 3 presents the results from a simulation study. Section 4 applies the ZINB regression model to the human esophageal microbiota sequence data, and Section 5 provides a summary and discusses areas for further research.
Figure 1 Variable sequencing effort. The range of total sequence counts can be large. This figure displays the variable sequencing effort (total as variable name) from a microbiota dataset and is between 32 and 2056.
CHAPTER II

THE ZERO-INFLATED NEGATIVE BINOMIAL (ZINB) REGRESSION MODEL

2.1 ZINB Regression Model

The Zero-Inflated Negative Binomial (ZINB) regression model assumes there are two distinct data generation processes. The result of a Bernoulli trial is used to determine which of the two processes is used. For observation $i$, with probability $\pi_i$ the only possible response of the first process is zero counts, and with probability of $(1-\pi_i)$ the response of the second process is governed by a negative binomial with mean $\lambda_i$. The zero counts are generated from both the first and second processes, where a probability is estimated for whether zero counts are from the first or the second process. The overall probability of zero counts is the combined probability of zeros from the two processes.

Following Greene [6] and Yau et al. [11], a ZINB model for the response $Y_i$ can be written as:

$$Y_i = 0 \quad \text{with probability } \pi_i$$

$$Y_i \sim \text{negative binomial } (\lambda_i, k) \quad \text{with probability } (1-\pi_i)$$

So that,

$$\Pr(Y_i = 0) = \pi_i + (1-\pi_i)(1+k\lambda_i)^{-1/k}$$

$$\Pr(Y_i = y_i) = (1-\pi_i) \frac{\Gamma(y_i + 1/k)}{\Gamma(y_i + 1)\Gamma(1/k)(1+k\lambda)^{y_i}} \left(\frac{\lambda}{1+k\lambda}\right)^{y_i}, \quad y_i = 1, 2, ...$$

In this case, the mean and variance of the $Y_i$ are
\begin{align*}
E(Y_i) &= (1 - \pi_i)\lambda_i \\
V(Y_i) &= (1 - \pi_i)\lambda_i (1 + \lambda_i (\pi_i + k))
\end{align*}

where $\lambda_i$ is the mean of the underlying negative binomial distribution, and $k$ is the over-dispersion parameter. The ZINB distribution reduces to the ZIP distribution as $k\to 0$.

The parameter $\lambda_i$ is modeled as a function of a linear predictor, that is, $\hat{\lambda}_i = \exp(X_i'\beta)$. $\beta$ is the $(p+1)\times1$ vector of unknown parameters associated with the known covariate vector $X_i'=(1, X_{i1}, \ldots, X_{ip})$, where $p$ is the number of covariates not including the intercept. The parameter $\pi_i$, which is often referred as the zero-inflation factor, is the probability of zero counts from the binary process. For common choice and simplicity, $\pi_i$ is characterized in terms of a logistic regression model by writing as $\text{logit}(\pi_i) = Z_i'\gamma$. $\gamma$ is the $(q+1)\times1$ vector of zero-inflated coefficients to be estimated, associated with the known zero-inflation covariate vector $Z_i'=(1, Z_{i1}, \ldots, Z_{iq})$, where $q$ is the number of the covariates $Z$'s not including the intercept. In the terminology of generalized linear models (GLMs), $\log(\lambda)$ and $\text{logit}(\pi)$ are the natural links for the negative binomial mean and Bernoulli probability of success.

### 2.2 Single-level ZINB Regression Model with Random Effects

The ZINB model can be extended to include random effects to directly model the correlation among the repeated measures within a subject [11]. Let $Y_{ij}$ ($i=1,2,\ldots,m; j=1,2,\ldots,n_i$ and $\sum_{i=1}^m n_i = n$ gives the total number of observations) now be the response variable for the $i^{th}$ individual subject with $j^{th}$ repeated measurement. The random-effects ZINB model is defined as follows:
\[
\log(\lambda_{ij}) = X_{ij}'\beta + u_i
\]
\[
\log(\pi_{ij}) = Z_{ij}'\gamma + v_i
\]

where \(X_{ij}\) and \(Z_{ij}\) are respectively vectors of covariates for the negative binomial and the logistic components, and \(\beta\) and \(\gamma\) are the corresponding vectors of regression coefficients.

Here, for mathematical simplicity, the random effects \(u_i\) and \(v_i\) are assumed to be independent and distributed as \(N(0, \sigma_u^2)\) and \(N(0, \sigma_v^2)\) respectively. Although this is not a necessary assumption, it is commonly used in the previous literature regarding ZIP/ZINB with random effects [10, 11]. Besides, we believed that the process that generates the sampling zeros (zeros from binary component) is independent of the process that generated the ordinary counts. The models with independent and correlated random effects will be fit separately for a subset of the organisms from the motivating dataset in section 4.3 to test and verify this assumption.

The log-likelihood function is given by

\[
ll = \sum_{y_i=0} \log[\pi_{ij} + (1-\pi_{ij})(1+k\lambda_{ij})^{-1/k}]
+ \sum_{y_i>0} \left[ \log(1-\pi_{ij}) + \log(\Gamma(y_{ij} + 1/k)) \Gamma(y_{ij} + 1) \Gamma(1/k) + y_{ij}\log(k\lambda_{ij}) - (y_{ij} + 1/k)\log(1+k\lambda_{ij}) \right],
\]

and this model can be fit by a Newton-Raphson algorithm.
CHAPTER III

SIMULATION STUDY AND RESULTS

To determine pattern or results from hypothetical situations which may be applied to a real situation, a simulation study was performed for model verification and to study the performance of the standard errors estimates from Section 2. The data were simulated from a single-level random-effects ZINB regression model with

\[
\log(\lambda_{ij}) = \beta_0 + \beta_1 X_{ij1} + \beta_2 X_{ij2} + \beta_3 X_{ij3} + u_i \\
\text{Logit}(\pi_{ij}) = \gamma_0 + \gamma_1 Z_{ij1} + \gamma_2 Z_{ij2} + \gamma_3 Z_{ij3} + v_i
\]

and two independent random effects \( u_i \) and \( v_i \) as described in Section 2.2.

One thousand datasets with 1000 subjects each with 4 repeated measurements were generated using each of the two sets of the parameters values shown in Table 1. Here, \( X_1 (=Z_1) \), \( X_2 (=Z_2) \), and \( X_3 (=Z_3) \) are the dummy variables for the second, third and fourth repeated measures. The mean, variance, and the squared standard error (SE) of the 1000 estimates for the random-effects ZINB regression models from NLMIXED are also presented in the Table 1. The mean of 1000 estimates for each parameter is close the true value and appear to be unbiased. To determine whether the standard error (SE) from NLMIXED correctly captures the variability of the estimates, comparisons between variance of 1000 estimates and squared SE over 1000 datasets from NLXIMED were performed and the results show they are close. There are more variations (larger variance) when \( \beta \)'s and \( \gamma \)'s are negative and large.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>True value</th>
<th>Mean (Variance) of 1000 estimates</th>
<th>(SE)$^2$ of the 1000 estimates</th>
<th>True value</th>
<th>Mean (Variance) of 1000 estimates</th>
<th>(SE)$^2$ of the 1000 estimates</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta_0$</td>
<td>1.00</td>
<td>1.00 (0.03)</td>
<td>0.03</td>
<td>1.00</td>
<td>1.00 (0.03)</td>
<td>0.03</td>
</tr>
<tr>
<td>$\beta_1$</td>
<td>-1.50</td>
<td>-1.51 (0.18)</td>
<td>0.17</td>
<td>-1.50</td>
<td>-1.50 (0.16)</td>
<td>0.14</td>
</tr>
<tr>
<td>$\beta_2$</td>
<td>-2.00</td>
<td>-2.00 (0.40)</td>
<td>0.35</td>
<td>-2.00</td>
<td>-2.00 (0.29)</td>
<td>0.26</td>
</tr>
<tr>
<td>$\beta_3$</td>
<td>3.00</td>
<td>3.00 (0.03)</td>
<td>0.03</td>
<td>3.00</td>
<td>3.00 (0.04)</td>
<td>0.04</td>
</tr>
<tr>
<td>$\sigma_u$</td>
<td>0.50</td>
<td>0.50 (0.02)</td>
<td>0.02</td>
<td></td>
<td>1.00 (0.02)</td>
<td>0.01</td>
</tr>
<tr>
<td>$\gamma_0$</td>
<td>-0.50</td>
<td>-0.50 (0.16)</td>
<td>0.16</td>
<td>-0.50</td>
<td>-0.50 (0.12)</td>
<td>0.12</td>
</tr>
<tr>
<td>$\gamma_1$</td>
<td>0.75</td>
<td>0.75 (0.05)</td>
<td>0.05</td>
<td>0.75</td>
<td>0.75 (0.03)</td>
<td>0.03</td>
</tr>
<tr>
<td>$\gamma_2$</td>
<td>1.00</td>
<td>1.00 (0.11)</td>
<td>0.11</td>
<td>1.00</td>
<td>1.00 (0.06)</td>
<td>0.05</td>
</tr>
<tr>
<td>$\gamma_3$</td>
<td>-1.50</td>
<td>-1.50 (0.27)</td>
<td>0.26</td>
<td>-1.50</td>
<td>-1.50 (0.19)</td>
<td>0.21</td>
</tr>
<tr>
<td>$\sigma_v$</td>
<td>1.00</td>
<td>1.00 (0.01)</td>
<td>0.02</td>
<td>0.50</td>
<td>0.50 (0.04)</td>
<td>0.04</td>
</tr>
<tr>
<td>$k$</td>
<td>0.50</td>
<td>0.50 (0.01)</td>
<td>0.01</td>
<td>0.50</td>
<td>0.50 (0.02)</td>
<td>0.02</td>
</tr>
</tbody>
</table>
CHAPTER IV

APPLICATION TO HUMAN MICROBIOTA SEQUENCE DATA

4.1 Study Description

Esophagitis is a general term for any inflammation, irritation, or swelling of the esophagus, the tube that leads from the back of the mouth to the stomach. The two most common forms of esophagitis are related to acid injury, a condition called Gastroesophageal Reflux Disease (GERD) and allergy, an autoimmune disorder called Eosinophilic Esophagitis (EoE) [18]. To date, the underlying etiology of these diseases is not certain but several studies support the role of gastrointestinal microbiota in disease processes [19], so we are interested in further identifying the esophageal microbiota. Several studies have examined microbiological aspects of this microenvironment using mucosal biopsies obtained at the time of endoscopic procedures [19, 20, 21, 22, 23, 24]. While technically feasible, this methodology is invasive, costly, involves risk and is limited to capture a very small part of the esophageal mucosa. To address this issue, we utilized the proximal section of a minimally invasive device, the Enterotest TM, to sample the esophageal mucosal microbial microenvironment. As described, we harvested adherent secretions from the proximal portion that was situated in the esophagus and analyzed ribosomal RNA gene sequences.

The motivating example is from a study in which pediatric individuals with normal esophageal mucosa provided four samples to capture esophageal microbiota. The samples included the gold standard mucosal biopsy and the minimally invasive capsule-
based string, the Enterotest™. Additionally, oral and nasal cavity swabs were collected for comparison. The aim of the study was to show that the esophageal microbiota from biopsies and ESTs would produce near identical profiles and was different from samples collected from the nasal and oral cavity.

All human species were collected under approval of the Colorado Multiple Institutional Review Board (COMIRB). Written informed consent and HIPAA authorization were obtained from all participants or from parents or legal guardians of participants younger than 18 years. Assent was obtained from all participants under 18 years. The night prior to the endoscopy, subjects swallowed an Enterotest™ capsule (Figure 2). The pediatric Enterotest™ consists of a weighted gelatin capsule filled with 90 cm of nylon string. The proximal end of the string (~10 cm) extrudes from one end of the capsule, which is held and taped to the cheek after the capsule is swallowed. The EST was removed before or just after induction of general anesthesia. Locations of esophageal and gastric segments of the EST were determined using a combination of PH indicator sticks supplied with the tests, and by measuring the distance to the lower esophageal sphincter at the time of endoscopy.

Mucosal biopsies were then obtained from the middle of distal part of the esophagus, nasal swabs, a 2 cm segment of oral string and a 2 cm segment of the middle part of esophageal string were collected from each subject. Biopsies, swabs and string samples were snap frozen in liquid nitrogen and kept at -80 degrees centigrade until DNA extraction. DNA from all samples was extracted using Qiagen DNAeasy Extraction Kits for blood and tissue according to manufacturer’s specifications. DNA was amplified in triplicate with barcoded PCR primers that include adaptors for the Roche 454 sequencing...
Negative PCR controls were platformed for each barcode, and PCR was repeated for any sample where the negative control was positive. Amplicons were pooled after normalization of DNA concentration [26], and sequenced using the Roche 454 FLX platform according to manufacturer’s specification. Sequence data were assigned to samples of origin using bar code sequences added during PCR, and screened for basic quality defects by the program BARTAB [27]. Non-bacterial sequences and sequences identified as potential chimeras were removed from datasets. The Ribosomal Database Project Classifier was used to make taxonomic assignments [28]. Taxonomic information was used to construct sequence groups with identical taxonomic rank, which were used for bacterial community analyses, and to identify specific bacteria that were differentially present between locations.

**Figure 2 displays Esophageal String Test (EST).** The top arrow shows the portion of the string that is taped to the cheek. The bottom arrow indicates the weighted capsule that is dislodged in the duodenum.
There were 15 subjects enrolled in this study and all of them had normal histological biopsy findings. Four types of samples were collected for most of the subjects, and the number of them is listed as follows: 13 nasal swabs and 15 oral strings, ESTs and biopsies. Bacterial ribosomal RNA gene amplification products from mucosal biopsies and from the nasal cavity, oral cavity and EST were visualized by gel electrophoresis (Figure 3).

Figure 3 displays 16S amplification and detection of nasal, oral and esophageal (EST and biopsy) microenvironments. This is a picture of a representative 2% agarose gel with a 200bp amplification product from the V2-V3 region of the 16S-rDNA gene. DNA samples were obtained from nasal swabs, oral strings, ESTs and esophageal biopsies.

4.2 The ZINB Model for Human Microbiota Sequence Data

Considering the variable sequencing effort of microbiota sequence data, we need to add an offset, the natural logarithm of the total sequence counts, into the linear predictor function for negative binomial component. That is, \( \log (E(\text{seq\_count})) = \beta_0 + \beta_1 \times \text{String} + \beta_2 \times \text{Nasal} + \beta_3 \times \text{Oral} + \log(\text{Total}) \), where the covariates “String”, “Nasal”, “Oral” are the indicators for sample types (EST, Nasal and Oral respectively). After a little algebra, it can be shown that \( \log (E(\text{seq\_count})/\text{Total}) = \beta_0 + \beta_1 \times \text{String} + \beta_2 \times \text{Nasal} + \beta_3 \times \text{Oral} \). The left side of this equation is modeling the log of the relative abundance/proportion as the outcome, assuming the total sequence counts is considered a
fixed value rather than a random variable. Note that the parameter $\pi_i$ is not affected by the variable sequencing effort. We then fit a ZINB model with random effects as described in Section 2.2. Let $Y_{ij}$ and $\pi_{ij}$ be the sequence count and zero-inflation factor for the $i^{th}$ subject and $j^{th}$ sample type, respectively. The log link function for modeling the relative abundance/proportion is

$$
\log \left( \frac{E(Y_{ij})}{l_{total_{ij}}} \right) = \beta_0 + \beta_1 \times \text{String}_{ij} + \beta_2 \times \text{Nasal}_{ij} + \beta_3 \times \text{Oral}_{ij} + u_i ,
$$

where the additional variable, $l_{total_{ij}}$, the natural logarithm of the total sequence count for the $i^{th}$ subject and $j^{th}$ sample type, is added to the equation as an offset. The logit link function for modeling the zero-inflation factor is

$$
\logit(\pi_{ij}) = \gamma_0 + \gamma_1 \times \text{String}_{ij} + \gamma_2 \times \text{Nasal}_{ij} + \gamma_3 \times \text{Oral}_{ij} + v_i .
$$

Here, $u_i$ and $v_i$ are the random intercepts and they are assumed to be independent and follow the bivariate normal distribution as

$$
\begin{bmatrix}
u_i \\
v_i
\end{bmatrix} \sim \text{BVN} \left( \begin{bmatrix} 0 \\
0 \end{bmatrix}, \begin{bmatrix} \sigma_u^2 & 0 \\
0 & \sigma_v^2 \end{bmatrix} \right).
$$

For simplicity, we assume the independence of two random effects. Although this is not a necessary assumption, it is commonly used in the previous literature regarding ZIP/ZINB with random effects [10, 11]. Besides, we believe that the process that generates the sampling zeros (dependent on sequencing depth) is independent of the process that generates the sequence counts. The models with independent and correlated random effects will be fit separately for a subset of the organisms from the motivating dataset in the next section to test and verify this assumption.
The models are fit via the NLMIXED procedure (SAS Version 9.3) using maximum likelihood estimation via Newton-Raphson algorithm.

4.3 Results

Within the entire dataset, 427 different taxa were identified. To demonstrate the performance of the ZINB regression model, we selected 4 organisms classified at genus level to represent the range of over-all described percentage of zeros resulting the combined zeros from the two processes in the distributions (Figure 4), in which *Gemella* had a small percentage of zero counts (12%) across 4 sample types, *Leptotrichia* and *Aggregatibacter* (36% and 55%) were intermediated and *Streptobacillus* had a high percentage at 79%.

**Figure 4** Distribution of human microbiota sequence counts for each of four organisms classified at genus level. This figure displays histogram for the following four organisms to represent the range of proportion of zeros present in the dataset: *Gemella, Leptotrichia, Aggregatibacter and Streptobacillus*.
Both models with independent and correlated random effects were fit using NLMIXED for each of four organisms classified at genus level. In all cases the model with independent random effects was found to be better than the model with correlated random effects based on the likelihood ratio test (reduced vs. full models) and Akaike’s Information Criterion (AIC) in Table 2.

<table>
<thead>
<tr>
<th>Organism</th>
<th>Model with independent random effects (reduced)</th>
<th>Model with correlated random effects (full)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-2 Log Likelihood</td>
<td>AIC</td>
</tr>
<tr>
<td>Gemella</td>
<td>344.6</td>
<td>366.6</td>
</tr>
<tr>
<td>Leptotrichia</td>
<td>259.6</td>
<td>281.6</td>
</tr>
<tr>
<td>Aggregatibacter*</td>
<td>203.9</td>
<td>225.9</td>
</tr>
<tr>
<td>Streptobacillus</td>
<td>107.7</td>
<td>129.7</td>
</tr>
</tbody>
</table>

* The final Hessian matrix is not positive definite

A ZINB regression model with independent random-effects described in section 4.2 was fit to each of the four organisms selected. The model fit for one organism resulted in a non-positive definite Hessian matrix. The parameter estimates for the remaining three organisms are given in Table 3.

Here, the value of \( \exp(\beta_0) \) represents the expected relative abundance/proportion for biopsy samples. The exponential values of \( \beta_1, \beta_2 \) and \( \beta_3 \) represent the expected ratio in relative abundance/proportion for string, nasal and oral samples, respectively, compared to biopsy samples. The zero-inflated factor \( \pi \), predicts the probability of observing a zero count from the point mass component. This parameter modeled by a logistic regression, is an increasing function of \( Z'\gamma \) for fixed random effect \( v \) (for simplicity we suppress the subscripts \( i \) and \( j \)). When \( Z'\gamma \) is positive and goes to \( +\infty \), the zero state becomes more
likely; when $Z'\gamma$ is negative and goes to $-\infty$, there are fewer zeros. The odds that a sample type does not contain a taxa would increase by a factor of $\exp(\gamma)$ compared to the biopsy sample. The sigmas are the estimated standard deviations for the normally distributed random subject effects.

<table>
<thead>
<tr>
<th>Table 3</th>
<th>Results of ZINB regression model with independent random effects for three organisms selected from the motivating dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Gemella</td>
</tr>
<tr>
<td>$\beta_0$</td>
<td>-4.68</td>
</tr>
<tr>
<td>$\beta_1$</td>
<td>0.15</td>
</tr>
<tr>
<td>$\beta_2$</td>
<td>-0.03</td>
</tr>
<tr>
<td>$\beta_3$</td>
<td>0.50</td>
</tr>
<tr>
<td>$\sigma_u$</td>
<td>-0.39</td>
</tr>
<tr>
<td>$\gamma_0$</td>
<td>-17.17</td>
</tr>
<tr>
<td>$\gamma_1$</td>
<td>-4.75</td>
</tr>
<tr>
<td>$\gamma_2$</td>
<td>16.03</td>
</tr>
<tr>
<td>$\gamma_3$</td>
<td>-4.29</td>
</tr>
<tr>
<td>$\sigma_v$</td>
<td>0.39</td>
</tr>
<tr>
<td>$k$</td>
<td>0.58</td>
</tr>
</tbody>
</table>

The group means for biopsy, string, nasal and oral samples are estimated by calculating the overall mean $E(Y) = (1 - \pi)\lambda = \frac{1}{\exp(Z'\gamma) + 1} \times \exp(X'\beta)$. Then contrasts for the difference between two groups are constructed. To show how the esophageal microbiota from biopsies and ESTs are identical, and are different from samples collected from the nasal and oral cavity, point estimate and p-values for the difference between two groups are shown in Table 4.

<table>
<thead>
<tr>
<th>Table 4</th>
<th>ZINB model estimates and p-values for three organisms selected from the motivating dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>EST vs. Biopsy</td>
</tr>
<tr>
<td>Gemella</td>
<td>0.001 (0.66)</td>
</tr>
<tr>
<td>Leptotrichia</td>
<td>-0.006 (&lt;0.05)</td>
</tr>
<tr>
<td>Streptobacillus</td>
<td>0.002 (0.47)</td>
</tr>
</tbody>
</table>
Compared to the EST samples, microbiota captured from mucosal biopsies and oral cavity samples had significant higher amounts in *Leptotrichia* (P<0.05). In addition, comparisons of the microbiota captured in the nasal and oral cavities revealed one significant difference in *Leptotrichia*. Although there were some statistically significant differences, the differences are relative small. Finally, the results with the EST samples and nasal microbiota did not reveal any significant differences in the three organisms selected.

The random-effects ZINB model appears to describe the sequence data variations reasonably well, whose fitted percent of total frequency distribution plot is presented in **Figure 5**.

Next we fit the ZINB model to the 427 taxa identified in the entire dataset. Of them, 240 taxa did not have non-zero counts for the subset of samples used and were not included in this analysis. In the remaining 187 taxa, 86 of them were successfully fit and the convergence criterion satisfied. However, the final Hessian matrix is not positive definite for 64 of the models converged. For those models that could not be fit, the majority of the taxa had no zeros or had large percentage of zeros with extremely small or large non-zero counts. All the comparisons of the 187 taxa used the similar approach in the table 4 and then 427 p-values were generated for each comparison. Specifically, for the models that didn't converge or didn't have any data, the p-values were set to 1.00. Manhattan plot, commonly used in genetic studies, was used to display the magnitude of the p-values for each comparison with the taxa ordered by taxonomy line, and color-coded by phylum. As shown in Manhattan plots (**Figure 6**), there were 4 statistically
significant differences identified in comparing the microbiota from ESTs and biopsies (P<0.05). These results support the use of the EST to sample the microbiome as compared to the “gold standard”, the mucosal biopsy. Comparisons of the microbiota captured in ESTs and nasal cavity samples revealed 8 differences. In addition, comparisons of the microbiota captured in the EST and oral samples revealed 2 differences. Finally, results with the nasal and oral microbiota revealed several (7) differences in these two locations. These results suggest that each microenvironment harbors specific taxa that distinguish the sites from EST and biopsy.
Figure 5 Empirical and fitted ZINB distributions of the human microbiota sequence data for each of four organisms classified at genus level.
Figure 6 Manhattan plots for the comparisons across all taxa. The y-axis displays the negative log of the p-value; hence higher values indicate increased statistical significance. The reference lines in gray are included to designate the usual critical values. The Manhattan plot is ordered by taxonomy line and the colors correspond to different phyla.
CHAPTER V

DISCUSSION

The distributions of the microbial sequence counts are highly skewed, non-negative and have a large proportion of zeros, for which commonly used statistical approaches may not be appropriate. The large proportion of zeros is intrinsic to the creation of the dataset rather than the data generating process itself. The dataset contains sequence counts for organisms that were observed in at least one sample, if a particular organism was not observed in a sample it is given a zero value. Therefore, when comparing sequence counts across groups in the presence/absence of organisms, a large numbers of zero counts are expected. It is believed that the absence of a count for an organism can be explained by two underlying processes (structural and sampling zeros).

In this thesis, the ZINB model with random effects was described. This model is useful for analysis of over-dispersed count data with an excess of zeros and repeated measures. Also, it showed how this model maybe estimated using standard software for non-linear and generalized liner mixed models such as SAS PROC NLMIXED. Simulation study indicated that this method of estimation gave unbiased results for both fixed effects and random effects. The application to the three selected organism from the microbiota data demonstrated the usefulness of this approach. However, given the complexity of the model, we are not able to apply it to all organisms and it requires adaption and guidelines for high-dimensional applications. The majority of models that did not fit were bugs have all zeros or large proportion of zeros with extreme small/large
counts. It is more likely that this model will address more focused questions related to a small subset of organisms of clinical interest.

Due to the small sample size in our motivation study (15 subjects and 58 samples in total), we have lower power. We cannot assume that lack of significance in comparison means there is no difference. In such circumstance, we would look at the magnitude of the differences, rather than focusing on the p-values.

To assess the effects of misspecification of random effect distributions in the two parts of ZINB regression model, other distributional assumptions apart from normality should be considered in future research.

In our study, we separately fit the models to the organisms identified thus ignoring the correlation among organisms. We are interested in modeling a couple of the organisms multivariately or a multi-level (twofold random effects) zero-inflated model.
## REFERENCES


APPENDIX A

SAS CODE

/* Simulation study: a single-level random-effects ZINB regression model */
/* SubjectID = the number of subjects */
/* ui, random effect at ordinary component, normal distributed with mean = 0 and sd1 */
/* vi, random effect at logistic component, normal distributed with mean = 0 and sd2 */
/* N = number of repeated measures */
/* covariates = x1, x2 and x3, are the dummy variables for the 2nd, 3rd and 4th repeated measures */
/* b0, b1, b2, b3, c0, c1, c2 and c4 are the parameters */
/* k = over-dispersed parameter */
/* outcome = y */
%macro simdata(DatasetNum, NumSubjects, N, sd1, sd2, b0, b1, b2, b3, c0, c1, c2, c3, k, out);
data simdata(keep = DatasetNum SubjectID ui vi i x1 x2 x3 lambda mean ynb eta_p p0 Zero_inflate y);
DatasetNum = &DatasetNum;
DO SubjectID = 1 to &NumSubjects;
/* Two random effects follow normal distribution */
ui = RAND('NORMAL', 0, &sd1);
vi = RAND('NORMAL', 0, &sd2);
DO i = 1 TO &N;
/* covariates */
x1 = (i = 2);
x2 = (i = 3);
x3 = (i = 4);
/* negative binomial: Poisson-Gamma mixture distribution*/
lambda = exp(&b0 + &b1* x1 + &b2 * x2 + &b3 * x3 + ui);
shape=1/&k;
scale=lambda*&k;
mean=scale*RAND('GAMMA', shape);
ynb=RAND('POISSON', mean);
/* zero-inflated parameter, follows a logistic function */
eta_p = &c0 + &c1 * x1 + &c2 * x2 + &c3* x3 + vi;
p0 = 1/(1+exp(-eta_p));
Zero_inflate = RAND('BINOMIAL', p0, 1); /* 0 or 1 */
ELSE DO; ynegzi = ynb; END;
ELSE DO; ynegzi = 0; END;
y=ynegzi;
output;
END;
END;
run;

proc nlmixed data=simdata tech=newrap;
parms b0=&b0 b1=&b1 b2=&b2 b3=&b3 c0=&c0 c1=&c1 c2=&c2 c3=&c3 k=&k su=&sd1 sv=&sd2;
eta = b0 + b1*x1 + b2*x2 + b3*x3 + u;
lambda = exp(eta);
eta_p = c0 + c1*x1 + c2*x2 + c3*x3 + v;
p0 = 1/(1+exp(-eta_p));
/* define ZINB log likelihood */
if y = 0 then ll = log( p0 + (1-p0)/(1+k*lambda)**((1/k) ));
else ll = log((1-p0) + y*log(k*lambda) - (y+1)/k)) - log((1+k*lambda) + lgamma(y+1/k)) - lgamma(1/k) - lgamma(y+1);
model y ~ general(ll);
random u v  ~ normal ([0,0], [su*su, 0, sv*sv]) subject=SubjectID; /* the covariance is set to 0 */
ods output ParameterEstimates=parms;
run;
data parms;
set parms;
DatasetNum = &DatasetNum;
run;

data &out.;
set parms &out.;
run;
%mend;

%let seed=12345;
data _NULL_;/* create a reproducible stream of random numbers with the RAND function*/
CALL STREAMINIT(&seed);
run;

/* sd1=0.5 and sd2=1.0 */
%macro byss();
%do j=1 %to 1000; /*generate 1000 datasets */
%simdata(DatasetNum=&j, NumSubjects=100, N=4, sd1=0.5, sd2=1, b0=1, b1=-1.5, b2=-2, b3=3, c0=-0.5, c1=0.75, c2=1, c3=-1.5, k=0.5, out=out1);
%end;
%mend;
data out1; run;
%byss();

/*sd1=1.0 and sd2=0.5 */
%macro byss();
%do j=1 %to 1000; /*generate 1000 datasets */
%simdata(DatasetNum=&j, NumSubjects=100, N=4, sd1=1, sd2=0.5, b0=1, b1=-1.5, b2=-2, b3=3, c0=-0.5, c1=0.75, c2=1, c3=-1.5, k=0.5, out=out2);
%end;
%mend;
data out2; run;
%byss();

/* compare var(beta_hat) vs. [SE(beta_hat)]**2 */
data out1;
set out1;
se2=StandardError*StandardError;
run;

proc means data=out1 mean var;
class parameter;
var estimate se2;
run;
data out2;
set out2;
se2=StandardError*StandardError;
run;

proc means data=out2 mean var;
class parameter;
var estimate se2;
run;

/**********************************
/* fit the model to the real data */
/**********************************/
libname rui 'E:\BIOS MS\MS Thesis';

/* generate the sequencing data for 15 Normals */
PROC IMPORT out=seq
DATANAME= "E:\BIOS MS\MS Thesis\z93EoE_BAFGEN_RDP-Genus.xls"
DBMS=xls REPLACE;
SHEET="sheet1";
RUN;

data seq;
  set seq;
  genus=scan(taxon,-1,'/');
  phyla=scan(taxon,3,'/');
run;

proc sort data=seq; by phyla genus taxon; run;

proc transpose data=seq out=seq2(rename=(col1=seq_count _NAME_=sample) drop=_LABEL_);
by phyla genus taxon;
var _numeric_;
run;

data seq2;
  set seq2;
  if seq_count =0 then zero_inflate=1;
  else if seq_count >0 then zero_inflate=0;
run;

proc means data=seq2; class genus;
var zero_inflate;
output out=zero_inflate(drop=_freq_) mean=prob_zero;
run;

proc means data=seq2 min max; class genus;
var seq_count;
output out=seq3(drop=_freq_) min=seq_min max=seq_max;
run;

proc sort data=seq2; by genus; run;
proc sort data=zero_inflate where=(_type_=1) by genus; run;
proc sort data=seq3 where=(_type_=1) by genus; run;

data seqdata1;
merge seq2 zero_inflate(where=(_type_=1)) seq3(where=(_type_=1));
by genus;
run;

/* import the meta data */
PROC IMPORT out=sample
DATAFILE="E:\BIOS MS\MS Thesis\z93EoE_BAFGEN_RDP-Genus.xls"
DBMS=xls REPLACE;
SHEET="Sheet2";
RUN;

proc means data=seq2 n sum min max;
class sample;
var seq_count;
output out=total sum=total;
run;

proc sort data=seqdata1; by sample; run;
proc sort data=sample; by sample; run;
proc sort data=total; by sample; run;

data rui.seqdata;
merge sample seqdata1 total(where=(TYPE=1));
by sample;
  if type='S' then string=1; else string=0;
  if type='B' then biopsy=1; else biopsy=0;
  if type='N' then nasal=1; else nasal=0;
if type='O' then oral=1; else oral=0;
ltotal=log(total); /* natural logarithm */
/* generate the seq number for each genus */
p=lag1(sample);
do z=0 to 1;
  if p=' ' then SEQ=1;
  else if sample = p then SEQ=v+z;
  else if sample ne p then SEQ=1;
v=lag1(seq);
end;
drop _type_ _freq_ P z v;
run;

/* taxon, phyla, genus information */
data rui.taxon;
set rui.seqdata(obs=427);
keep phyla genus taxon SEQ prob_zero seq_min seq_max;
run;

proc print data=rui.taxon;
  where 0.1 < prob_zero < 0.3;
  var seq genus seq_max prob_zero;
run;

proc print data=rui.taxon;
  where 0.3 < prob_zero < 0.4;
  var seq genus seq_max prob_zero;
run;

proc print data=rui.taxon;
  where 0.4 < prob_zero < 0.6;
  var seq genus seq_max prob_zero;
run;

proc print data=rui.taxon;
  where 0.6 < prob_zero < 0.7;
  var seq genus seq_max prob_zero;
run;

proc print data=rui.taxon;
  where 0.7 < prob_zero < 0.8;
  var seq genus seq_max prob_zero;
run;

proc print data=rui.taxon;
  where 0.8 < prob_zero < 0.9;
  var seq genus seq_max prob_zero;
run;

proc print data=rui.taxon;
  where 0.9 < prob_zero <= 1.0;
  var seq genus seq_max prob_zero;
run;

/* select 3 bugs */
data rui.seqdata_sub;
set rui.seqdata;
where seq in (150 205 385);
if seq=150 then seq_num=1;
if seq=205 then seq_num=2;
if seq=385 then seq_num=3;
label seq_count='Sequence Count';
run;

proc format;
value seq_num 1='Gemella' 2='Leptotrichia' 3='Streptobacillus';
run;
ods graphics off;
title 'Distribution of sequence counts for 4 organisms';
proc capability data=rui.seqdata_sub normal;
var seq_count;
comphistogram seq_count/class=seq_num nrows=3 midpoints=0 to 58;
inset mean var min max / format=5.2 pos=ne header = "Summary Statistics";
format seq_num seq_num.;
run;

/*fit the model to 3 bugs selected, the random effects are independent */
%macro ZINB (fs=, ae=, cs=);
/*obtain the starting values, the random effects are not included */
proc countreg data=rui.seqdata_sub;
  where seq_num=&j;
  model seq_count=string nasal oral/dist=zinb offset=ltotal;
  zeromodel seq_count ~ string nasal oral/link=logistic;
  ods output ParameterEstimates=pe;
run;

proc sql;
select estimate as b0 into: b0 from pe where Parameter='Intercept';
select estimate as b1 into: b1 from pe where Parameter='string';
select estimate as b2 into: b2 from pe where Parameter='nasal';
select estimate as b3 into: b3 from pe where Parameter='oral';
select estimate as c0 into: c0 from pe where Parameter='Inf_Intercept';
select estimate as c1 into: c1 from pe where Parameter='Inf_string';
select estimate as c2 into: c2 from pe where Parameter='Inf_nasal';
select estimate as c3 into: c3 from pe where Parameter='Inf_oral';
select estimate as k into: k from pe where Parameter='Alpha';
quit;

proc nlmixed data=rui.seqdata_sub tech=newrap;
  where seq_num=&j;
  parms b0=&b0. b1=&b1. b2=&b2. b3=&b3. c0=&c0. c1=&c1. c2=&c2. c3=&c3. k=&k. su=1 sv=1;
  eta = b0 + b1*string + b2*nasal + b3*oral + ltotal + ui;
  lambda = exp(eta);
  eta_p = c0 + c1*string + c2*nasal + c3*oral + vi ;
  p0 = 1/(1+exp(-eta_p));
/* define ZINB log likelihood */
  if seq_count=0 then ll = log( p0 + (1-p0)/(1+k*lambda)**(1/k) );
  else ll = log(((1-p0)) + seq_count*log(k*lambda) - (seq_count+1/k)*log(1+k*lambda) +
          lgamma(seq_count+(1/k)) - lgamma(1/k) - lgamma(seq_count+1));
model seq_count ~ general(ll);
random ui vi ~ normal ([0,0], [su*su, 0, sv*sv]) subject=Subject;
/* the group mean is (1-p0)*lambda */
/*estimate 'Biopsy' exp(b0)/(exp(c0)+1)/;*
/*estimate 'String' exp(b0+b1)/(exp(c0+c1)+1)/;*
/*estimate 'Nasal' exp(b0+b2)/(exp(c0+c2)+1)/;*
/*estimate 'Oral' exp(b0+b3)/(exp(c0+c3)+1)/;*
/*estimate and test difference in mean counts*/
estimate 'S vs. B' exp(b0+b1)/(exp(c0+c1)+1) - exp(b0)/(exp(c0)+1);
estimate 'S vs. N' exp(b0+b1)/(exp(c0+c1)+1) - exp(b0+b2)/(exp(c0+c2)+1);
estimate 'S vs. O' exp(b0+b1)/(exp(c0+c1)+1) - exp(b0+b3)/(exp(c0+c3)+1);
estimate 'N vs. O' exp(b0+b2)/(exp(c0+c2)+1) - exp(b0+b3)/(exp(c0+c3)+1);
/* request predicted counts */
predict p0 out=predzidata_&j(keep=pred rename=pred=pred_zi);
predict lambda out=preddata_&j(keep=seq_count pred rename=pred=pred);
predict k out=pred_k_&j(keep=pred rename=pred=k);
ods output FitStatistics=fs_&j AdditionalEstimates=ae_&j ConvergenceStatus=cs_&j;
run;
proc transpose data=fs_&j out=fs_out_&j(rename=(COL1=ll1 COL2=aic1 COL3=aicc1 COL4=bic1)
drop=_NAME_);
run;
data fs_out_&j;
set fs_out_&j;
seq_num = &j.;
run;
data ae_&j;
set ae_&j;
seq_num = &j.;
run;
data cs_&j;
set cs_&j;
seq_num = &j.;
run;
data &fs.;
set fs_out_&j &fs.;
run;
data &cs.;
set cs_&j &cs.;
run;
data &ae.;
set ae_&j &ae.;
run;
proc freq data=rui.seqdata_sub noprint;
where seq_num=&j;
tables seq_count/missing out=pred2_&j;
run;
%macro estimate;
%do i = 0 %to 57;
v= 1/k;
if &i. = 0 then pred_&i. = pred_zi + (exp(lgamma(&i.+(v)))-lgamma(&i.+1)
- lgamma (v) + &i.*log((1/v)*pred)
 - (&i.+(v))*log(1+((1/v)*pred)))*(1-pred_zi));
else pred_&i. = (exp(lgamma(&i.+(v)))-lgamma(&i.+1)-lgamma (v) + &i.*log((1/v)*pred)
 - (&i.+(v))*log(1+((1/v)*pred)))*(1-pred_zi));
%end;
%mend;
data pred3_&j;
merge predzidata_&j preddata_&j pred_k_&j;
array preds (*) pred_0-pred_57;
%estimate;
pred_sum = sum(of preds(*));
run;
/* roll up - summarizing predicted probabilities of each count */
proc summary data=pred3_&j noprint;
var pred_:;
output out=pred4_&j sum=;
run;
data pred5_&j(keep=seq_count pred pct pctcum);
set pred4_&j(keep=pred_0-pred_57 pred_sum);
array preds[58] pred_0 pred_1 pred_2 pred_3 pred_4 pred_5 pred_6 pred_7 pred_8 pred_9 pred_10
pred_11 pred_12 pred_13
     pred_14 pred_15 pred_16 pred_17 pred_18 pred_19 pred_20 pred_21 pred_22 pred_23
pred_24 pred_25 pred_26(pred_27 pred_28 Pred_29 pred_30 pred_31 pred_32 pred_33 pred_34 pred_35 pred_36
pred_37 pred_38 pred_39
     pred_40 pred_41 pred_42 pred_43 pred_44 pred_45 pred_46 pred_47 pred_48 pred_49 pred_50 pred_51 pred_52
pred_53 pred_54 pred_55 pred_56 pred_57;
array ns (58) (0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
31 32 33 34 35 36 37 38
    39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57);
do i = 1 to dim(preds);
seq_count=ns(i);
pred=preds(i);
pct = (pred/pred_sum)*100;
pctcum+pct;
output;
end;
rung/* bring together actuals vs. predicted & test */
data pred6_&j;
merge pred2_&j pred5_&j(keep=seq_count pred pct);
by seq_count;
rung/* generate a count histogram from the sample overlaid with the predicted distribution */
goptions reset=all;
symbol value=plus i=join color=red w=2;
axis1 order= 0 to 100 by 10 label=(angle=90);
axis2 order= 0 to 100 by 10 label=none;
proc gbarline data=pred6_&j;
bar seq_count / sumvar=percent type=mean discrete axis=axis1;
plot / sumvar=pct raxis=axis2;
label seq_count = 'Sequence Count';
rung
title ' ';
%mend;

%macro driver ();
data fs1; run;
data ae1; run;
data cs1; run;
%do j=1 %to 3;
%ZINB (fs=fs1, ae=ae1, cs=cs1);
%end;
%mend;
%driver();

/**********************************************************************
/* fit the model to 3 bugs selected, the random effects are correlated */
/**********************************************************************
%macro ZINB (fs=, ae=, cs= );
/*obtain the starting values */
proc countreg data=rui.seqdata_sub;
where seq_num=;&j;
model seq_count~string nasal oral/dist=zinb offset=ltotal;
zeromodel seq_count ~ string nasal oral/link=logistic;
ods output ParameterEstimates=pe;
rung;
proc sql;
select estimate as b0 into: b0 from pe where Parameter='Intercept';
select estimate as b1 into: b1 from pe where Parameter='string';
select estimate as b2 into: b2 from pe where Parameter='nasal';
select estimate as b3 into: b3 from pe where Parameter='oral';
select estimate as c0 into: c0 from pe where Parameter='Inf_Intercept';
select estimate as c1 into: c1 from pe where Parameter='Inf_string';
select estimate as c2 into: c2 from pe where Parameter='Inf_nasal';
select estimate as c3 into: c3 from pe where Parameter='Inf_oral';
select estimate as k into: k from pe where Parameter='_Alpha';
quit;

proc nlmixed data=rui.seqdata_sub tech=newrap;
where seq_num=&j;
parms b0=&b0.
b1=&b1.
b2=&b2.
b3=&b3.
c0=&c0.
c1=&c1.
c2=&c2.
c3=&c3.
k=&k.
su=1
sv=1
rho=0.5;
eta = b0 + b1*string + b2*nasal + b3*oral + ltotal + ui;
lambda = exp(eta);
eta_p = c0 + c1*string + c2*nasal + c3*oral + vi;
p0 = 1/(1+exp(-eta_p));/* define ZINB log likelihood */
if seq_count=0 then ll = log( p0 + (1-p0)/(1+k*lambda)**(1/k) );
else ll = log((1-p0)) + seq_count*log(k*lambda) - (seq_count+(1/k))*log(1+k*lambda) +
lgamma(seq_count+(1/k)) - lgamma(1/k) - lgamma(seq_count+1);
model seq_count ~ general(ll);
random ui vi ~ normal ([0,0], [su*su, rho*su*sv, sv*sv]) subject=Subject;
/* the group mean is (1-p0)*lambda */
/*estimate 'Biopsy' exp(b0)/(exp(c0)+1);*/
/*estimate 'String' exp(b0+b1)/(exp(c0+c1)+1);*/
/*estimate 'Nasal' exp(b0+b2)/(exp(c0+c2)+1);*/
/*estimate 'Oral' exp(b0+b3)/(exp(c0+c3)+1);*/
/*estimate and test difference in mean counts*/
estimate 'S vs. B' exp(b0+b1)/(exp(c0+c1)+1) - exp(b0)/(exp(c0)+1);
estimate 'S vs. N' exp(b0+b1)/(exp(c0+c1)+1) - exp(b0+b2)/(exp(c0+c2)+1);
estimate 'S vs. O' exp(b0+b1)/(exp(c0+c1)+1) - exp(b0+b3)/(exp(c0+c3)+1);
estimate 'N vs. O' exp(b0+b2)/(exp(c0+c2)+1) - exp(b0+b3)/(exp(c0+c3)+1);
/* request predicted counts */
predict p0 out=predzidata_&j(keep=pred rename=pred=pred_zi);
predict lambda out=preddata_&j(keep=seq_count pred rename=pred=pred);
predict k out=pred_k_&j(keep=pred rename=pred=k);
ods output FitStatistics=fs ConvergenceStatus=ZINB_cs AdditionalEstimates=ZINB_ae_&j;
run;
title ' ';
proc transpose data=fs out=fs_out(rename=(COL1=ll2 COL2=aic2 COL3=aicc2 COL4=bic2) drop_NAME_); run;
data fs_out;
set fs_out;
seq_num = &j.; run;
data ZINB_cs;
set ZINB_cs;
seq_num = &j.; run;
data ae_&j;
set ae_&j;
seq_num = &j.;
run;

data &fs.;
set fs_out &fs.;
run;

data &cs.;
set ZINB_cs &cs.;
run;

data &ae.;
set ae_&j &ae.;
run;
%mend;

%macro driver ();
%do j=1 %to 3;
%ZINB (fs=fs2, ae=ae2, cs=cs2);
%end;
%mend;
data fs2; run;
data ae2; run;
data cs2; run;
%driver();

proc sort data=fs1; by seq_num; run;
proc sort data=fs2; by seq_num; run;

data better_model;
merge fs1 fs2;
where seq_num ne .;
by seq_num;
lr=abs(ll1-ll2);
prob=1-probchi(lr, 1);
if aic1 < aic2 then better=1; else better=2;
run;

/*****************************************************/
/* fit model to 427 bugs, independent random effects */
/*****************************************************/
%macro ZINB (fs=, ae=, cs=);
/* start values */
proc countreg data=rui.seqdata;
where seq=&j;
model seq_count=string nasal oral/dist=zinha offset=ltotal;
zeromodel seq_count ~ string nasal oral/link=logistic;
ods output ParameterEstimates=pe;
run;
proc sql;
select estimate as b0 into: b0
from pe where Parameter='Intercept';
select estimate as b1 into: b1
from pe where Parameter='string';
select estimate as b2 into: b2
from pe where Parameter='nasal';
select estimate as b3 into: b3
from pe where Parameter='oral';
select estimate as c0 into: c0
from pe where Parameter='Inf_Intercept';
select estimate as c1 into: c1
from pe where Parameter='Inf_string';
select estimate as c2 into: c2
from pe where Parameter='Inf_nasal'
select estimate as c3 into: c3
from pe where Parameter='Inf_oral'
select estimate as k into: k
from pe where Parameter='_Alpha'
quit;
/* independent random effects */
proc nlmixed data=rui.seqdata tech=newrap;
  where seq=&j;
 parms b0=&b0.
b1=&b1.
b2=&b2.
b3=&b3.
c0=&c0.
c1=&c1.
c2=&c2.
c3=&c3.
k=&k.
su=1 sv=1;
  eta = b0 + b1*string + b2*nasal + b3*oral + ltotal + ui;
  lambda = exp(eta);
  eta_p = c0 + c1*string + c2*nasal + c3*oral + vi;
  p0 = 1/(1+exp(-eta_p));
/* define ZINB log likelihood */
  if seq_count=0 then ll = log( p0 + (1-p0)/(1+k*lambda)**(1/k) );
  else ll = log((1-p0) + seq_count*log(k*lambda) - (seq_count+1/k)*log(1+k*lambda) +
                 lgamma(seq_count+(1/k)) - lgamma(1/k) - lgamma(seq_count+1));
  model seq_count ~ general(ll);
  random ui vi ~ normal ([0,0], [su*su, 0, sv*sv]) subject=Subject;
/* the group mean is (1-p0)*lambda */
  estimate 'Biopsy' exp(b0)/(exp(c0)+1);
  estimate 'String' exp(b0+b1)/(exp(c0+c1)+1);
  estimate 'Nasal' exp(b0+b2)/(exp(c0+c2)+1);
  estimate 'Oral' exp(b0+b3)/(exp(c0+c3)+1);
/*estimate and test difference in mean counts*/
  estimate 'S vs. B' exp(b0+b1)/(exp(c0+c1)+1) - exp(b0)/(exp(c0)+1);
  estimate 'S vs. N' exp(b0+b1)/(exp(c0+c1)+1) - exp(b0+b2)/(exp(c0+c2)+1);
  estimate 'S vs. O' exp(b0+b1)/(exp(c0+c1)+1) - exp(b0+b3)/(exp(c0+c3)+1);
  estimate 'N vs. B' exp(b0+b2)/(exp(c0+c2)+1) - exp(b0)/(exp(c0)+1);
  estimate 'N vs. O' exp(b0+b2)/(exp(c0+c2)+1) - exp(b0+b3)/(exp(c0+c3)+1);
  estimate 'O vs. B' exp(b0+b3)/(exp(c0+c3)+1) - exp(b0)/(exp(c0)+1);
ods output FitStatistics=fs_&j AdditionalEstimates=ae_&j ConvergenceStatus=cs_&j;
run;
proc transpose data=fs_&j out=fs_out_&j(rename=(COL1=ll1 COL2=aic1 COL3=aicc1 COL4=bic1)
drop=_NAME_);
  run;
  data fs_out_&j;
  set fs_out_&j;
  seq =&j.;
run;
  data ae_&j;
  set ae_&j;
  seq=&j.;
run;
  data cs_&j;
  set cs_&j;
  seq=&j.;
run;
  data &fs.;
  set fs_out_&j &fs.;
run;
  data &cs.;
  set cs_&j &cs.;
run;
data &ae.;
set ae_&j &ae.;
run;
%mend;

%macro driver ();
%do j=1 %to 427;
%ZNIB (fs=fs3, ae=ae3, cs=cs3);
%end;
%mend;
data fs3; run;
data ae3; run;
data cs3; run;
%driver ();

proc freq data=cs3;
tables status;
run;

proc sort data=rui.taxon; by seq; run;
proc sort data=fs3; by seq; run;
proc sort data=cs3; by seq; run;
data out1;
merge fs3(where=(seq ne .)) cs3(where=(seq ne .)) rui.taxon; by seq;
run;

data s_vs_b s_vs_n s_vs_o n_vs_o n_vs_b o_vs_b;
set ae3;
if label in (‘S vs. B’) then output s_vs_b;
if label in (‘S vs. N’) then output s_vs_n;
if label in (‘S vs. O’) then output s_vs_o;
if label in (‘N vs. O’) then output n_vs_o;
if label in (‘N vs. B’) then output n_vs_b;
if label in (‘O vs. B’) then output o_vs_b;
run;

proc sort data=s_vs_b; by seq; run;
proc sort data=s_vs_n; by seq; run;
proc sort data=s_vs_o; by seq; run;
proc sort data=n_vs_o; by seq; run;
proc sort data=n_vs_b; by seq; run;
proc sort data=o_vs_b; by seq; run;
proc sort data=out1; by seq; run;
data rui.s_vs_b2;
merge s_vs_b out1;
by seq;
run;

proc print data=rui.s_vs_b2;
where probt ne . and probt < 0.05;
var genus;
run;

data rui.s_vs_n2;
merge s_vs_n out1;
by seq;
run;

proc print data=rui.s_vs_n2;
where probt ne . and probt < 0.05;
var genus;
run;

data rui.s_vs_o2;
merge s_vs_o out1;
by seq;
run;

proc print data=rui.s_vs_o2;
where probt ne . and probt < 0.05;
var genus;
run;
data rui.n_vs_o2;
merge n_vs_o out1;
by seq;
run;
proc print data=rui.n_vs_o2;
where probt ne . and probt < 0.05;
var genus;
run;
data rui.n_vs_b2;
merge n_vs_b out1;
by seq;
run;
proc print data=rui.n_vs_b2;
where probt ne . and probt < 0.05;
run;
data rui.o_vs_b2;
merge o_vs_b out1;
by seq;
run;
proc print data=rui.o_vs_b2;
where probt ne . and probt < 0.05;
run;
/*generate Manhattan plots*/
%macro Mplots (data, label, top);
proc sort data=&data;
by taxon;
run;
data plot2;
set &data;
if probt= . then p = 0;
else p = -(log10(probt));
t = _n_; 
format pvalue 5.2;
pvalue=p;
run;
proc means data=plot2 max;
var pvalue;
run;

goptions reset=all ;
%annomac;
data label;
set plot2;
length text $ 25;
retain when 'a' style 'swissb' xsys ysys '2';
%label(10,1.25, 'p = 0.05', grey,0 ,0 , .8, swiss);
%label(10,1.95, 'p = 0.01', grey,0 ,0 , .8, swiss);
%label(10,2.95, 'p = 0.001', grey,0 ,0 , .8, swiss);
run;
goptions htext=1.3 ftext=swissb;
title &label;
axis1 label=(a=90 '-log pvalue') order=(0 to &top by .5);
axis2 label=none value=none order=(1 to 430 by 2) offset=(2,2);
legend1 value=(h=1);
proc gplot data=plot2;
plot pvalue*t=phyla/vref=(1.30 2 3) vaxis=axis1 haxis=axis2 annotate=label legend=legend1;
symbol1 i=j v=dot w=2 repeat=30;
run;
proc datasets nolist;
delete plot2;
run;
%mend;
ods listing; run;
ods select all;
%mplots (rui.s_vs_b2, 'EST vs. Biopsy', 3.0);
ods listing; run;
ods select all;
%mplots (rui.s_vs_n2, 'EST vs. Nasal', 3.0);
ods listing; run;
ods select all;
%mplots (rui.s_vs_o2, 'EST vs. Oral', 3.0);
ods listing; run;
ods select all;
%mplots (rui.n_vs_o2, 'Nasal vs. Oral', 3.0);
ods listing; run;
ods select all;
%mplots (rui.n_vs_b2, 'Nasal vs. Biopsy', 3.0);
ods listing; run;
ods select all;
%mplots (rui.o_vs_b2, 'Oral vs. Biopsy', 3.0);
/* examine the distribution for the bugs cannot fit */
ods graphics off;
proc capability data=rui.seqdata normal;
where seq=240;
var seq_count;
comphistogram seq_count/class=type nrows=4 midpoints=0 to 1049 by 50;
inset mean var min max pnormal/format=5.2 pos=ne header = "Summary Statistics";
label seq_count='Sequence Count';
run;
title '';
ods graphics off;
proc capability data=rui.seqdata normal;
where seq=311;
var seq_count;
comphistogram seq_count/class=type nrows=4 midpoints=0 to 950 by 50;
inset mean var min max pnormal/format=5.2 pos=ne header = "Summary Statistics";
label seq_count='Sequence Count';
run;
title '';
ods graphics off;
proc capability data=rui.seqdata normal;
where seq=386;
var seq_count;
comphistogram seq_count/class=type nrows=4 midpoints=0 to 800 by 50;
inset mean var min max pnormal/format=5.2 pos=ne header = "Summary Statistics";
label seq_count='Sequence Count';
run;
title ' ';

/* end of the code */