Biostat/Stat 571 Methods Paper

Entropy Regularized Multinomial Logistic Regression for High-Dimensional Count Data

Nicholas Irons, Leah Andrews, Ning Duan, Kaelan Yu

Contents

Abstract

The microbiome has been shown to be associated with complex diseases, such as cancer and diabetes. Thus, it is important to characterize it. However, since microbiome data consist of high dimensional zero-inflated taxon counts, many standard methods, like multinomial logistic regression (MLR), cannot converge due to linear separability. Since regularization methods have attempted to address this issue, in our study, we propose a novel statistical model involving a penalized form of MLR for prediction. We then evaluate our method in a simulation study and apply it to a Graft vs Host Disease microbiome dataset. We compare our method to MLR and MLR with lasso, elastic net, and ridge regression penalties and evaluate performance based on mean absolute error, mean squared error, mean absolute prediction error, and mean squared prediction error. Conventional statistical machine learning practices, such as cross validation for hyperparameter optimization, are employed appropriately. Our regularized model often outperforms the other benchmark methods on data that has a multinomial logistic distribution, independent or positively correlated covariates, and response variable with some linearly separable categories. Future work includes extension of the model to the Bayesian setting, optimization of method code, conducting bootstrapping for inference, and developing the theoretical statistical guarantees for the entropic MLR estimator. All data, code, and results described in this paper are publicly available at our GitHub repository[1].

1 Introduction

Trillions of microbial cells are living on and inside the human body, primarily in the gut [1]. The collection of genes that make up these cells is known as the microbiome. Recent studies have found associations between the human microbiome and complex diseases, including asthma, cancer, diabetes, and treatment efficacy. Relatively new evidence suggests the gut microbiome may be closely related to the development of Graft vs Host Disease (GvHD), which occurs when a donor's transplant cells attack the recipient [2] [3]. An ultimate goal is to fit a joint model for longitudinal microbiome data to detect its association with the onset of GvHD [4]. Hence, there is a growing need to accurately characterize the human microbiome before incorporating its longitudinal nature with GvHD and patient characteristics.

Gut microbiome data has several features that make it difficult to analyze appropriately. First, the microbiome is made up of taxa, which are species of microbes, and are the units of analysis. Thus, microbiome data consists of counts, where the total number of taxa and amount of each taxon varies greatly per individual [5]. Additionally, microbiome data are sparse, meaning they are zero-inflated, or have a large proportion of zeroes. Moreover, microbiome data are high-dimensional and multivariate. For example, we often observe a sample size of about 200 individuals with more than 850 taxa. Standard or naive approaches exist to address each characteristic, but fail to address all three simultaneously.

[1] https://github.com/njirons/stat571project

Some recent literature attempts to address at least two of the characteristics collectively. Xu, et al. propose a zero-inflated Poisson factor model that aims to reduce dimension [6]. However, it cannot deal with over-dispersion. Zhang, et al. propose a zero-inflated negative binomial model to account for over-dispersion [7]. Luna, et al. present a negative binomial mixed-effects model [4] . However, it only allows for up to three taxa and suggests implementing high dimensional Bayesian approaches in the future. We suggest that multinomial logistic regression (MLR) models with certain regularization would be better to address the three characteristics of gut microbiome data described above, including high-dimensionality. Also, MLR is appropriate to reflect that microbiome data are subject to a sum constraint resulting in a simplex sample space [8].

In a high-dimensional, multivariate regression on counts, for each level of the outcome variable, a different set of regression parameters is needed to link the response to the covariates. Fitting so many parameters often leads to failing maximum likelihood convergence [9]. The most common approach to solve this problem is regularization using a LASSO penalty term, or an L^1 penalty. Other approaches include the regularization using a ridge regression penalty term, or an L^2 penalty, and the regularization using the elastic net, which is a combination of the LASSO and the ridge regression.

Another problem that may occur with microbiome data is linear separability, which occurs when at least one covariate can perfectly classify an outcome for each observation [10]. With our outcome variable being high-dimensional and zero-inflated, it is likely that a microbiome data set is linearly separable. Several approaches from different theoretical frameworks can be used, including regularization methods [11], Bayesian priors in Bayesian literature [12], and support vector machines (SVM) [13] [14].

Although standard regularization methods can address the problem of linearly separable data, none of them exploit the geometry of the simplex on which the multinomial category probabilities are constrained to lie (see Figure 1). Therefore, we propose a novel MLR model with an entropy regularization term. Our model can address all three previously mentioned data characteristics, converge despite linear separability, and respect the geometry of the simplex sample space. We hope to fill the gap that no existing method provides a satisfactory solution to predict taxa in the microbiome.

In this study, we used machine learning methods for prediction. In terms of the microbiome data set, we predicted taxa counts based on GvHD and subject covariates. In our simulation studies, we generated data based on the multinomial model and sparse multinomial model and compare our method to multinomial logistic regression (MLR) and three other MLR with regularization methods (MLR + LASSO, MLR + ridge regression, and MLR + elastic net). We used four performance metrics, mean absolute error (MAE), mean squared error (MSE), mean absolute prediction error (MAPE), and mean squared prediction error (MSPE), to evaluate and compare our method. Finally, we applied our method to a real gut microbiome and GvHD data set. From our simulation study and data application, we found that our entropy regularization MLR outperforms MLR and standard regularization MLR methods under many circumstances, including when the data has a multinomial distribution, strong or no correlation in the covariates, and response variables with some linearly separable categories. Thus, our method can better handle high-dimensional multivariate categorical data that are sparse in the response.

The remainder of the article is structured as follows. In Section 2, we describe the stan-

dard multinomial logistic regression (MLR) model. Then we provide maximum likelihood estimation (MLE) for MLR and its algorithms for implementation. After motivating the linear separability issue, we introduce our entropy regularized model. We also introduce the regularization methods to compare to our model and the performance metrics to evaluate all methods. Next, we describe our synthetic data sets and a real-world microbiome-GvHD data set. In Section 3, we present the results of analyses of the synthetic data sets and the real microbiome-GvHD data set. In Section 4, we conclude our article with a discussion.

2 Methods

In this section, we discuss background motivating theory, propose our innovative statistical model, elaborate on the benchmark models we are using for comparison purposes in predictive power, identify performance metrics, and describe the data sets.

2.1 Standard multinomial logistic regression (MLR) model

The standard multinomial logistic regression (MLR) model works as follows. Let Y_{ij} denote the count of the jth taxon for the ith individual with $i = 1, \ldots, m$ and $j = 1, \ldots, J$. Let $Y_i = (Y_{i1}, \ldots, Y_{iJ})$ denote the response vector of counts for the ith individual. Also define the total count for the ith individual, $n_i = \sum_{j=1}^{J} Y_{ij}$. The model is

$$
\begin{aligned}
Y_i &\overset{ind.}{\sim} \text{Multinomial}(n_i, p_i), \quad i = 1, \ldots, m \\
p_{ij} &= \frac{\exp(X_i^T \beta_j)}{1 + \sum_{j=1}^{J-1} \exp(X_i^T \beta_j)}, \quad j = 1, \ldots, J-1 \\
p_{iJ} &= \frac{1}{1 + \sum_{j=1}^{J-1} \exp(X_i^T \beta_j)}.
\end{aligned}
$$

Note that $p_i = (p_{i1}, \ldots, p_{iJ})$ is the vector of probabilities for the ith subject. We define p_{iJ} as such because the probabilities must sum to 1.

2.2 The problem of linearly-separable data

To motivate our method, suppose we have a binary predictor $X_{i1} \in \{0, 1\}$ for each individual, e.g., the presence of disease. Suppose that the jth taxon is absent in the stool samples of every individual with the disease, i.e., $Y_{ij} = 0$ whenever $X_{i1} = 1$. This would suggest that the presence of the jth taxon is associated with absence of the disease, which is a scientifically interesting result. However, the fact that $Y_{ij} = 0$ for those individuals without the disease means that, in fitting the model, the parameter p_{ij} will be shrunk toward zero. Since the data are 'linearly separable', we can accomplish this by letting $\beta_{j1} \to -\infty$. Hence, our fitting method will diverge and we won't be able to estimate β_{j1}. This was a simplified example, but as our response $Y = (Y_{ij})$ becomes high-dimensional and more replete with zeros (which is the case for the GvHD data), the chance that the data become linearly separable in this way, or that the estimation method breaks down in some other way, will increase.

2.3 Maximum likelihood estimation for MLR: gradient ascent

The algorithm to perform maximum likelihood estimation for MLR via gradient ascent forms the foundation for our proposed model, so we describe it explicitly here. Since there is no closed-form solution for MLE of the regression coefficients $\beta = [\beta_1 \dots \beta_{J-1}]$, the log-likelihood is often maximized using gradient ascent. Alternatively, we could implement Newton-Raphson/Fisher scoring to fit the model if the Fisher information can be precomputed, which might speed up convergence. Note that we have written β as a matrix with columns formed by the vectors β_j. Therefore $\beta \in \mathbb{R}^{p\times(J-1)}$, where p is the number of covariates, i.e., $X_i \in \mathbb{R}^p$.

Up to a constant, the log-likelihood for MLR is

$$\ell(\beta) \stackrel{c}{=} \sum_{i=1}^{m}\sum_{j=1}^{J} Y_{ij} \log p_{ij}(\beta) := \sum_i \ell_i(\beta).$$

The gradient of ℓ_i with respect to the vector β_j is then

$$\begin{aligned}
\nabla_{\beta_j}\ell_i(\beta) &= X_i Y_{ij} - \sum_{k=1}^{J} Y_{ik}\nabla_{\beta_j}\left\{\log\left(1+\sum_{l=1}^{J-1}\exp(X_i^T\beta_l)\right)\right\} \\
&= X_i Y_{ij} - n_i\nabla_{\beta_j}\left\{\log\left(1+\sum_{l=1}^{J-1}\exp(X_i^T\beta_l)\right)\right\} \\
&= X_i Y_{ij} - n_i\frac{X_i\exp(X_i^T\beta_j)}{1+\sum_{l=1}^{J-1}\exp(X_i^T\beta_l)} \\
&= X_i\left\{Y_{ij} - \frac{n_i\exp(X_i^T\beta_j)}{1+\sum_{l=1}^{J-1}\exp(X_i^T\beta_l)}\right\} \\
&= X_i(Y_{ij} - n_i p_{ij}) \\
&= X_i(Y_{ij} - \hat{Y}_{ij}),
\end{aligned}$$

where we have defined the expected counts $\hat{Y}_{ij} = n_i p_{ij}$. Summing over i, we can write this in matrix notation as

$$\nabla_{\beta_j}\ell(\beta) = X^T(Y_{*j} - \hat{Y}_{*j}),$$

where $Y_{*j} = (Y_{1j}, \dots, Y_{mj})$, and similarly for $\hat{Y}_{*j}$. The matrix $X \in \mathbb{R}^{m\times p}$ is the design matrix with ith row X_i. The full gradient can then be written as the matrix

$$\nabla_\beta\ell(\beta) = [\nabla_{\beta_1}\ell \cdots \nabla_{\beta_{J-1}}\ell] = X^T(Y_{*1:(J-1)} - \hat{Y}_{*1:(J-1)}) \in \mathbb{R}^{p\times(J-1)}.$$

Note that $Y_{*1:(J-1)}$ is the matrix with jth column Y_{*j}, for $j = 1, \dots, J-1$.

Gradient ascent proceeds by initializing $\beta^{(0)} = 0$ and updating via the iterations

$$\beta^{(t+1)} = \beta^{(t)} + s_t\nabla_\beta\ell(\beta^{(t)}), \qquad t = 0, 1, 2, \dots$$

Here $s_t > 0$ is the step size at the tth iteration. In principle s_t could be a fixed constant, i.e., $s_t = s$ for all t, or it can vary at each step. In our data applications that follow, however, we opt to fit MLR via the widely used `nnet` R package [15], which trains neural networks to estimate the model parameters, rather than implementing our own gradient ascent algorithm by hand.

2.4 Entropy Regularized Multinomial Logistic Regression

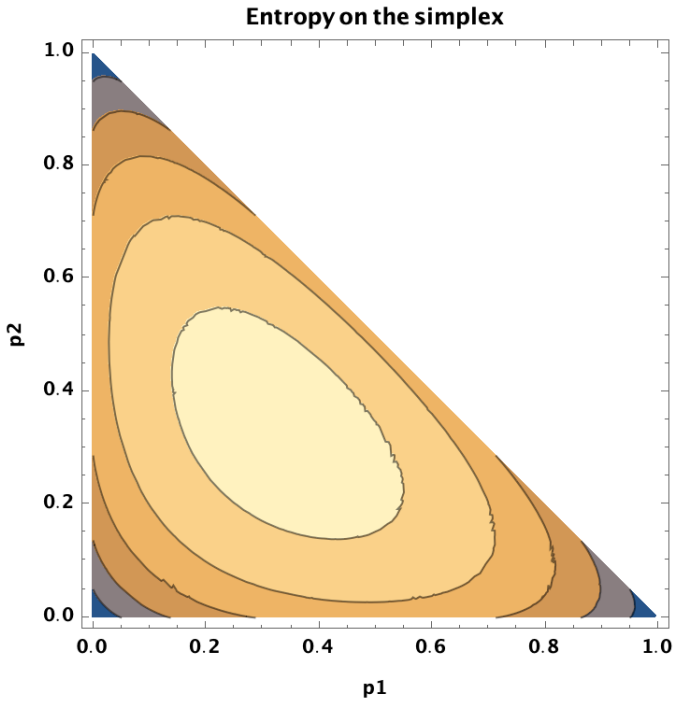

Figure 1: Level curves of entropy $H(p)$ on the 2-dimensional simplex $\Delta = \{(p_1, p_2, p_3) : p_i \geq 0, \sum_i p_i = 1\}$. Darker shades correspond to smaller values. Note that the entropy is maximized at the barycenter of the simplex $p = (1/3, 1/3, 1/3)$, which lies in the lightest shaded region. Entropy decreases as p becomes sparse, i.e., some of the probabilities p_i approach zero. This plot was generated using Wolfram Mathematica.

We propose an entropy regularized form of multinomial logistic regression (EMLR) to address the problem of linearly-separable data and (possibly zero-inflated) high-dimensional count data more broadly. Let $p_i = (p_{i1}, \ldots, p_{iJ})$ be the vector of probabilities for the ith subject, which satisfies $p_j \geq 0$ for all j and $\sum_j p_j = 1$ (this means that p_i lies in the simplex). The entropy of p_i is defined as

$$H(p_i) = -\sum_{j=1}^{J} p_{ij} \log p_{ij}.$$

The entropy H satisfies the following properties:

1. $H(p) \geq 0$ for all p
2. $H(p) = 0$ if and only if p is a point mass. (We use the convention that $0 \log 0 = 0$.)
3. $H(p)$ is strictly concave in p
4. $H(p)$ is uniquely maximized by the uniform distribution $p = (1/J, \ldots, 1/J)$.

Our proposed statistical model is an implementation of entropy-regularized MLR by penalizing the log-likelihood by the entropy of the probability vectors p_i. That is, we aim to maximize the penalized log-likelihood

$$\begin{aligned}\tilde{\ell}(\beta) &= \ell(\beta) + \sum_{i=1}^{m} \epsilon_i H(p_i) \\ &= \sum_{i=1}^{m}\sum_{j=1}^{J} Y_{ij} \log p_{ij} - \sum_{i=1}^{m} \epsilon_i \sum_{j=1}^{J} p_{ij} \log p_{ij} \\ &= \sum_{i=1}^{m} \{\ell_i + \epsilon_i H_i\} \\ &:= \sum_{i=1}^{m} \tilde{\ell}_i(\beta).\end{aligned}$$

Here $\epsilon_i > 0, i = 1, \ldots, m$ are tuning parameters. To simplify calculations, we could reduce this to a single tuning parameter $\epsilon_i = \epsilon$ for all i or set $\epsilon_i = \epsilon n_i$ for some fixed $\epsilon > 0$, in which case $\epsilon n_i p_{ij}$ can be interpreted as a pseudo-count modifying the original MLR log-likelihood. In the data applications that follow we opt for this latter choice. The intuition for this "maximum entropy" method is that we are maximizing a trade-off between the log-likelihood ℓ, whose maximum is the MLE, which may be undefined or diverge if the data is linearly separable, and the entropy H, which is maximized at the uniform distribution. Consequently, solutions are encouraged to move the probabilities p_{ij} away from 0 (which occurs when $\beta_j \to \pm\infty$) and toward $1/J$ (which occurs when $\beta_j = 0$). In this sense Entropic MLR performs implicit shrinkage estimation of the β_j, which bodes well for the predictive performance of the estimator, as we show in the results section.

Algorithm 1 Entropic MLR Accelerated Gradient Ascent

input step-size $s = 1$, target accuracy ε
initialization $\beta_0 = 0$, $\theta_0 = 0$
repeat for $t = 1, 2, \ldots$
Find s_t via Armijo backtracking line search (see Algorithm 2).
$\beta^{(t+1)} = \theta^{(t)} + s_t \nabla \tilde{\ell}(\theta^{(t)})$
$\theta^{(t+1)} = \beta^{(t+1)} + \frac{t}{t+3}(\beta^{(t+1)} - \beta^{(t)})$
until the stopping criterion $\tilde{\ell}(\beta^{(t+1)}) - \tilde{\ell}(\beta^{(t)}) \leq \varepsilon$.

We can also maximize this function using gradient ascent to fit the parameters β. A similar (but tedious) calculation as the one given for regular MLR shows that the gradient is given by

$$\begin{aligned}\nabla_{\beta_j} \tilde{\ell}_i(\beta) &= X_i\{Y_{ij} - p_{ij}[n_i + \epsilon_i(H(p_i) + \log p_{ij})]\} \\ &:= X_i\{Y_{ij} - \tilde{Y}_{ij}\}.\end{aligned}$$

Here we have defined the pseudo-counts $\tilde{Y}_{ij} = p_{ij}[n_i + \epsilon_i(H(p_i) + \log p_{ij})]$. We can sum over i and write this in matrix notation as

$$\nabla_{\beta_j} \tilde{\ell}(\beta) = X^T\{Y_{*j} - \tilde{Y}_{*j}\}.$$

Algorithm 2 Armijo Backtracking Line Search

input parameters $\alpha \in (0, 0.5), \gamma \in (0, 1)$
initialization step size $s = 1$, current iterate $\theta^{(t)}$, step direction $\Delta^{(t)} = \nabla\tilde{\ell}(\theta^{(t)})$,
Update $s \leftarrow \gamma s$.
until the stopping criterion $\tilde{\ell}(\theta^{(t)} + s\Delta^{(t)}) \geq \tilde{\ell}(\theta^{(t)}) + \alpha s\|\Delta^{(t)}\|_2^2$.

The full gradient, written as a matrix, is then

$$\nabla_\beta\tilde{\ell}(\beta) = X^T\{Y_{*1:(J-1)} - \tilde{Y}_{*1:(J-1)}\}.$$

The gradient ascent update is then

$$\beta^{(t+1)} = \beta^{(t)} + s_t\nabla_\beta\tilde{\ell}(\beta^{(t)}).$$

Calculating things in this way using matrices speeds up computation when we implement things in R by allowing us to avoid using for loops. In our data applications that follow, we proceed to implement the Armijo backtracking line search rule to determine a near-optimal step size s_t at each iteration [16]. See Algorithm 2. Furthermore, we implement an accelerated gradient ascent method, introduced by Nesterov [17], that uses weighted combinations of the current and previous gradient directions to speed up convergence. See Algorithm 1, which outlines the method. Note that in the code implementation, we divide the log-likelihood ℓ by the total count $n = \sum_{i=1}^m n_i$ to avoid overflow. As a result, we also divide the gradient by n.

2.5 Benchmark Models

To evaluate the performance of our innovative statistical model, we will use the following benchmark models from the statistical literature for comparison purposes.

- MLR with no penalty
- MLR with ridge penalty
- MLR with lasso penalty
- MLR with elastic net penalty

We have already discussed standard MLR above. We will discuss the remaining penalized variants of multinomial logistic regression in the following subsections.

2.5.1 MLR with a ridge penalty

In ridge regression we aim to maximize the penalized log-likelihood

$$\ell_R(\beta) = \ell(\beta) - \frac{\lambda}{2}\|\beta\|_2^2, \qquad \lambda \geq 0,$$

which also encourages the β_j not to diverge to $\pm\infty$. We can fit this model using gradient ascent and the gradient is given by

$$\nabla_\beta \ell_R(\beta) = \nabla_\beta \ell(\beta) - \lambda\beta.$$

In practice, however, we fit ridge-penalized MLR via the `glmnet` R package [18], which uses cyclical coordinate ascent.

2.5.2 MLR with a lasso penalty

In lasso regression we aim to maximize the penalized log-likelihood

$$\ell_L(\beta) = \ell(\beta) - \lambda\|\beta\|_1, \qquad \lambda \geq 0,$$

which also encourages the β_j not to diverge to $\pm\infty$ (and encourages them to be sparse). This objective function is not differentiable, but it can be maximized using subgradient or proximal gradient methods. However, in practice we fit the model using the `glmnet` R package [18], which implements cyclical coordinate ascent.

2.5.3 Elastic Net MLR

Elastic net MLR combines the ridge and lasso penalties and can also be fit using coordinate ascent in `glmnet` [18]:

$$\ell_{EN}(\beta) = \ell(\beta) - \lambda_1\|\beta\|_1 - \frac{\lambda_2}{2}\|\beta\|_2^2, \qquad \lambda_1, \lambda_2 \geq 0.$$

2.6 Performance Metrics

We evaluate our method using standard machine learning techniques and four performance metrics. Let $(\boldsymbol{X}, \boldsymbol{Y})$ be a data set with p covariates and a response variable with J categories generated from true coefficients $\beta_j \in \mathbb{R}^p$, j=$\{1, \ldots, J-1\}$. Then, we randomly split 80% of the data set into the training set and 20% of the data into testing set. Let $\hat{\beta}_j \in \mathbb{R}^p$, j=$\{1, \ldots, J-1\}$ denote the vector of estimated regression coefficients for the jth response category from the training set. Let $\hat{p}_{\ell j} = \frac{\exp(X_\ell^T \hat{\beta}_j)}{1+\sum_{j=1}^{J-1} \exp(X_i^T \hat{\beta}_j)}$ denote our predicted probabilities for the test set, where $\ell = \{1, \cdots, m\}$ are the indices of individuals in the test set. Let $\hat{Y}_{\ell j} = n_\ell \hat{p}_{\ell j}$ denote the estimated counts of the test set, where $n_\ell = \sum_{j=1}^{J} Y_{\ell j}$. Then

we compute the following performance metrics.

$$\text{Mean Squared Error (MSE)} = \frac{1}{(J-1)p}\sum_{j=1}^{J-1}\left(\hat{\beta}_j - \beta_j\right)'\left(\hat{\beta}_j - \beta_j\right)$$

$$\text{Mean Absolute Error (MAE)} = \frac{1}{(J-1)p}\sum_{j=1}^{J-1}\left(\left(\hat{\beta}_j - \beta_j\right)'\left(\hat{\beta}_j - \beta_j\right)\right)^{\frac{1}{2}}$$

$$\text{Mean Squared Prediction Error (MSPE)} = \frac{1}{J\ell}\sum_{\ell,j}\left(\hat{Y}_{\ell j} - Y_{\ell j}\right)^2$$

$$\text{Mean Absolute Prediction Error(MAPE)} = \frac{1}{J\ell}\sum_{\ell,j}\left|\hat{Y}_{\ell j} - Y_{\ell j}\right|$$

The MSE and MAE characterize the difference between the parameters and estimated parameters while the MSPE and MAPE describe the difference between the true response values and the predicted response values. Note that the MSE and MAE only rely on the parameters that generated the data and the parameters estimated from the training set. Thus, they measure the accuracy of the parameter. Since we are in a prediction setting, we are more focused on getting accurate predictions, so we are more interested in MSPE and MAPE. Additionally, in situations where we do not know the true parameters, such as in data applications, we can only calculate MSPE and MAPE.

Comparing MAE and MAPE to MSE and MSPE, the absolute errors penalizes small and large differences equally so they are less affected by outliers than the squared errors. The MAE and MAPE describe the median difference between the parameters or predictions, respectively. On the other hand, MSE and MSPE describe the mean difference between the parameters or predictors, respectively, with larger differences weighted higher than smaller differences. Using all four metrics may help us understand the behavior of our method under different situations.

2.7 Data sets

2.7.1 Simulated Data sets

We generated synthetic data sets under three different data generating mechanisms in R to conduct simulation studies.

Simulation 1 The data are generated from multinomial logistic model using 11 covariates (intercept, five binary covariates, and five independent normal covariates), varied sample size $m = 50, 100, 200$, and varied number of response categories $J = 25, 50, 100$.

Simulation 2 The data are generated from multinomial logistic model using 11 covariates (intercept, five binary covariates, and five correlated (exchangeable) normal covariates), sample size of $m = 100$, $J = 50$ response categories, and varied covariate correlation $\rho = 0, 0.1, 0.5, 0.9$.

Simulation 3 The data are first generated from multinomial logistic model using 11 covariates (intercept, five binary covariates, and five independent normal covariates), varied sample size $m = 50, 100, 200$, and varied number of response categories $J = 25, 50, 100$. Then a fixed fraction (10%) of the response categories are altered to be linearly separable with respect to the binary covariates.

For each scenario described above, we randomly generated 50 data sets. After the data generation, we randomly split each data set into 80% training set and 20% testing set. Then we applied MLR and the four penalized MLR methods to the training sets and used the estimated coefficients to predict on the test data set. We fit MLR with `nnet`, MLR with ridge and lasso penalty with `glmnet`, EN MLR with mixing parameter of $\frac{1}{2}$ with `glmnet`, and EMLR method using our own code. For the regularization methods, we used 5-fold cross validation on the training set to select the optimal tuning parameter from a grid of 50 values linearly spaced on the log scale. We used MSPE of the counts as the cross validation metric and a stopping criterion of 10^{-5} for all optimization algorithms. We computed MAE, MSE, MAPE, and MSPE for each data set and reported the mean and standard deviation.

2.7.2 Microbiome Data Set Related to GvHD

For patients with blood cancer, they often have to receive a bone marrow transplant, which transfers healthy blood-forming stem cells from a donor to the patient. Graft-vs-host disease (GvHD) may develop in some patients if the transferred cells attack the body of the patient or the host, resulting in considerable mortality. In recent years, multiple studies have provided evidence of the relationship between the gut microbiome and GvHD [2] [3]. Thus, it is relevant to identify differences in gut microbiome between patients who suffer from GvHD and those who don't.

In this study, we applied our method to a microbiome-GvHD data set collected from a cohort study of 229 individuals. The GvHD data set contains a collection of hematologic markers and demographic information. 45.4% of participants have clinical chronic GvHD after transplant, 44.5% of participants were female, and the average age was 54 years. Stool samples were collected and sequenced every week before and after the transplant. Data on 870 taxa were collected as counts for all individuals, and 93.6% of the taxa counts were zeros. The high-dimensionality and zero-inflated nature of the gut microbiome data set motivated us to develop our novel regularized method.

To evaluate our method on the microbiome-GvHD data set, we extracted participant data from and carried out a complete case analysis, with 112 participants remaining. We then randomly selected 50 of those subjects' taxa counts to be included in the analysis and split our data randomly into 80% training set, 20% test sets. Note that we aggregate a subjects taxa counts across all measured time points. We repeated this process to generate 50 data sets from the complete case microbiome-GvHD data set. Taking the average across the 50 training data sets, 43.9% had clinical chronic GvHD, 48.9% were female, the mean age was 60 years , and 78.7% of the taxa counts for all individuals were 0.

Next, we applied MLR and the four MLR regularization methods using patient demographics (age, race, sex), donor demographics (age, sex, relationship to patient), transplant characteristics (cancer status at transplant, types of cells infused), and GvHD characteristics

(chronic GVH grade, acute GVH in the gut, liver, and skin) to predict taxa counts. For the regularization methods, we used 3-fold cross validation to select from a grid of 30 tuning parameters. We used MSPE on counts as the cross validation metric and a stopping criterion of 10^{-5}. We report average MSPE and MAPE for our method across the 50 datasets, as well as their standard errors. We were not able to get the other methods to converge due to the highly zero-inflated data. See Table 4.

3 Results

In this section, we share our results for the simulation study and the microbiome data application. We identify when the EMLR method works well and how it compares to MLR and the other standard regularization methods based on the four performance metrics, MAE, MSE, MAPE, and MSPE.

We note one important distinction between applying our model to the simulation studies and to the GvHD data set. For simulated data we can directly compare the estimated coefficients to the true values. However, for the real data set, we will employ standard statistical machine learning practice, i.e., splitting data into training/testing sets, performing cross validation, and evaluating predictive performance via MSPE and MAPE on the test set.

3.1 Simulation Study 1

In the first scenario, we fitted MLR and regularized MLR methods on data generated from a multinomial distribution based on five binary and five independent normally distributed continuous covariates with varied sample size and response categories (Table 1). For a given sample size, EMLR performs equally well with respect to MAE and MSE for response variables with any number of categories. Additionally, prediction performance for MAPE and MSPE improves with the number of response categories. EMLR performance for all measures improves for larger sample sizes. Compared to MLR and the other regularization methods, EMLR has the lowest MAPE and MSPE mean and variance for nearly every sample size and number of response categories. For MAE and MSE, EMLR performs better than MLR + Ridge Regression but slightly worse than the other methods.

3.2 Simulation Study 2

In the second scenario, we fitted the MLR and regularized MLR methods on data generated from a multinomial distribution based on 5 binary and 5 correlated (exchangeable) continuous covariates with a fixed sample size of $m = 100$ and $J = 50$ response categories (Table 2). For independent covariates and covariates with a small positive correlation, the EMLR method performs well across all metrics. However, when covariates have a stronger positive correlation, the EMLR method performs worse for MAE and MSE and better for MAPE and MSPE. Compared to the other methods, EMLR outperforms MLR and the regularized MLR methods in terms of MAPE and MSPE for all four correlations tested. When the covariates are correlated, EMLR has a much lower MAE and MSE than MLR and MLR

m		**MLR**	**Lasso**	**Ridge**	**Elastic Net**	**Entropy**
				$J = 25$		
50	MSE	0.09 (0.07)	60.33 (15.31)	2.59 (0.89)	2.88 (1.01)	5.62 (2.22)
	MAE	2.65 (0.90)	76.26 (10.40)	13.62 (1.79)	14.39 (2.06)	21.79 (4.67)
	MSPE	96.28 (29.67)	7.09 (3.97)	0.20 (0.20)	0.19 (0.19)	0.09 (0.09)
	MAPE	98.31 (9.17)	25.94 (5.04)	3.63 (1.03)	3.58 (1.00)	2.52 (0.75)
100	MSE	0.43 (0.39)	81.34 (19.31)	2.82 (0.89)	3.21 (0.97)	5.56 (1.70)
	MAE	5.53 (2.40)	89.50 (11.33)	13.91 (1.46)	14.87 (1.72)	21.64 (3.87)
	MSPE	33.76 (14.60)	3.14 (1.63)	0.07 (0.05)	0.07 (0.05)	0.03 (0.02)
	MAPE	51.34 (6.52)	15.71 (2.94)	1.90 (0.41)	1.88 (0.41)	1.44 (0.28)
200	MSE	2.12 (2.10)	100.00 (18.68)	2.71 (0.83)	3.02 (0.97)	5.61 (1.34)
	MAE	12.15 (5.52)	100.00 (10.05)	13.70 (1.54)	14.46 (1.72)	21.04 (2.94)
	MSPE	9.54 (2.38)	1.15 (0.44)	0.02 (0.02)	0.02 (0.02)	0.02 (0.02)
	MAPE	26.01 (2.42)	9.24 (1.54)	1.03 (0.21)	1.02 (0.19)	0.98 (0.17)
				$J = 50$		
50	MSE	0.04 (0.04)	49.28 (13.17)	1.66 (0.58)	1.95 (0.75)	5.75 (2.47)
	MAE	1.77 (0.57)	68.72 (10.27)	11.32 (1.41)	12.39 (1.82)	22.32 (4.73)
	MSPE	100.00 (19.29)	4.77 (1.79)	0.09 (0.05)	0.08 (0.04)	0.04 (0.02)
	MAPE	98.88 (7.56)	21.60 (3.50)	2.60 (0.35)	2.54 (0.31)	1.82 (0.24)
100	MSE	0.29 (0.23)	67.30 (18.80)	1.64 (0.46)	1.93 (0.57)	4.45 (1.17)
	MAE	4.55 (1.72)	80.94 (11.62)	11.24 (1.25)	12.28 (1.51)	20.21 (3.17)
	MSPE	34.27 (6.23)	2.29 (0.88)	0.03 (0.02)	0.03 (0.02)	0.02 (0.01)
	MAPE	51.91 (3.30)	13.29 (1.84)	1.37 (0.16)	1.36 (0.16)	1.13 (0.14)
200	MSE	1.72 (2.05)	87.67 (20.06)	1.53 (0.30)	1.77 (0.37)	3.95 (0.84)
	MAE	10.56 (5.17)	93.02 (11.46)	10.81 (0.99)	11.75 (1.19)	18.42 (2.45)
	MSPE	10.78 (4.40)	0.94 (0.30)	0.01 (0.01)	0.01 (0.01)	0.01 (0.00)
	MAPE	26.26 (1.76)	8.00 (1.06)	0.72 (0.09)	0.71 (0.09)	0.69 (0.08)
				$J = 100$		
50	MSE	0.04 (0.05)	40.90 (13.93)	1.20 (0.37)	1.46 (0.50)	3.92 (1.69)
	MAE	1.55 (0.77)	62.47 (11.25)	10.08 (1.32)	11.19 (1.63)	19.02 (4.30)
	MSPE	97.30 (17.34)	3.74 (1.21)	0.06 (0.03)	0.06 (0.03)	0.02 (0.01)
	MAPE	100.00 (5.71)	19.25 (2.96)	2.21 (0.22)	2.18 (0.22)	1.50 (0.21)
100	MSE	0.25 (0.39)	54.48 (14.18)	1.17 (0.22)	1.37 (0.27)	4.52 (1.68)
	MAE	3.79 (2.29)	72.47 (10.20)	9.94 (0.92)	10.82 (1.06)	20.51 (4.00)
	MSPE	33.19 (5.02)	1.63 (0.51)	0.02 (0.01)	0.02 (0.01)	0.01 (0.00)
	MAPE	51.82 (2.41)	11.53 (1.63)	1.16 (0.10)	1.13 (0.09)	0.99 (0.08)
200	MSE	1.01 (1.23)	71.64 (12.91)	1.13 (0.24)	1.37 (0.32)	3.26 (0.90)
	MAE	8.21 (4.14)	83.56 (8.20)	9.52 (1.02)	10.60 (1.28)	16.60 (2.59)
	MSPE	9.96 (1.48)	0.65 (0.20)	0.01 (0.00)	0.01 (0.00)	0.00 (0.00)
	MAPE	26.06 (1.17)	6.83 (0.77)	0.60 (0.04)	0.59 (0.04)	0.60 (0.04)

Table 1: Results of simulation study 1. Average MSE, MAE, MSPE, and MAPE are reported for the 5 estimators considered across the 50 simulated data sets for each parameter setting. Standard errors are reported in parentheses. Each metric is scaled to lie between 0 and 100.

	MLR	Lasso	Ridge	Elastic Net	Entropy
			$\rho = 0$		
MSE	0.01 (0.01)	2.99 (0.83)	0.07 (0.02)	0.09 (0.03)	0.20 (0.05)
MAE	1.34 (0.51)	23.85 (3.43)	3.31 (0.37)	3.62 (0.45)	5.96 (0.93)
MSPE	100.00 (18.17)	6.68 (2.55)	0.09 (0.05)	0.08 (0.05)	0.05 (0.02)
MAPE	100.00 (6.36)	25.61 (3.55)	2.64 (0.31)	2.62 (0.30)	2.17 (0.26)
			$\rho = 0.1$		
MSE	75.70 (67.53)	2.94 (0.71)	0.07 (0.02)	0.08 (0.02)	0.20 (0.06)
MAE	100.00 (46.58)	23.63 (2.99)	3.23 (0.32)	3.51 (0.39)	5.85 (0.93)
MSPE	83.06 (21.13)	6.64 (2.85)	0.09 (0.04)	0.09 (0.04)	0.05 (0.03)
MAPE	96.55 (6.76)	25.88 (3.40)	2.72 (0.31)	2.70 (0.29)	2.24 (0.27)
			$\rho = 0.5$		
MSE	47.47 (50.08)	3.88 (1.05)	0.08 (0.03)	0.09 (0.03)	0.20 (0.07)
MAE	80.53 (38.96)	27.60 (3.94)	3.32 (0.47)	3.60 (0.52)	5.96 (1.13)
MSPE	90.25 (29.10)	7.18 (2.68)	0.08 (0.06)	0.08 (0.05)	0.05 (0.04)
MAPE	95.46 (7.88)	28.29 (4.59)	2.57 (0.38)	2.53 (0.37)	2.18 (0.28)
			$\rho = 0.9$		
MSE	100.00 (128.77)	5.23 (1.09)	0.19 (0.09)	0.17 (0.07)	0.32 (0.10)
MAE	98.92 (74.69)	32.20 (3.99)	4.49 (1.11)	4.26 (0.92)	7.17 (1.17)
MSPE	82.58 (29.94)	5.58 (3.14)	0.10 (0.08)	0.09 (0.07)	0.03 (0.01)
MAPE	90.28 (11.54)	24.79 (4.51)	2.84 (0.74)	2.66 (0.64)	2.08 (0.23)

Table 2: Results of simulation study 2. Average MSE, MAE, MSPE, and MAPE are reported for the 5 estimators considered across the 50 simulated data sets for each parameter setting. Standard errors are reported in parentheses. Each metric is scaled to lie between 0 and 100.

+ Ridge Regression, but about twice as high of an MAE and MSE than MLR + Lasso and MLR + EN.

3.3 Simulation Study 3

In the last scenario, we fitted MLR and regularized MLR methods on data similar to the first scenario, except we alter a fixed fraction of the response categories so the data is linearly separable (Table 3). There is a non-monotonic relationship with sample size and MAE and MSE for EMLR, such that EMLR has the lowest prediction error at a sample size of 200 and the highest prediction error at a sample size of 100. For MSPE and MAPE, the prediction performance improves with larger sample size. Increasing the number of response categories causes MSE to increase and MAE, MSPE, and MAPE to decrease. EMLR outperforms the other methods in terms of MSE when sample size is 200 and number of response categories is 25. Overall, EMLR performs as well or better than other methods in terms of MSPE when the numer of response categories is 100 or more for all sample sizes.

m		**MLR**	**Lasso**	**Ridge**	**Elastic Net**	**Entropy**
				$J = 25$		
50	MSE	19.35 (34.91)	1.14 (0.25)	1.57 (0.57)	1.72 (0.57)	1.47 (0.68)
	MAE	34.71 (14.65)	77.76 (9.43)	18.73 (2.04)	21.40 (2.39)	31.21 (7.08)
	MSPE	97.51 (25.09)	7.14 (3.56)	0.14 (0.27)	0.16 (0.29)	0.15 (0.29)
	MAPE	100.00 (11.85)	26.78 (5.74)	2.37 (0.87)	2.73 (0.95)	2.40 (0.83)
100	MSE	100.00 (154.37)	1.37 (0.21)	1.92 (0.36)	2.05 (0.39)	1.66 (0.46)
	MAE	56.31 (32.64)	87.68 (7.35)	19.34 (1.63)	20.58 (1.59)	37.14 (4.46)
	MSPE	32.74 (11.00)	2.77 (1.21)	0.05 (0.08)	0.04 (0.07)	0.08 (0.14)
	MAPE	50.32 (4.44)	15.33 (2.41)	1.15 (0.48)	1.20 (0.45)	1.48 (0.65)
200	MSE	42.46 (72.44)	1.68 (0.27)	1.78 (0.26)	1.98 (0.29)	1.41 (0.31)
	MAE	39.09 (19.02)	100.00 (9.79)	20.61 (1.69)	20.89 (1.61)	35.28 (3.63)
	MSPE	10.49 (3.00)	1.28 (0.62)	0.02 (0.04)	0.02 (0.03)	0.03 (0.04)
	MAPE	26.35 (2.52)	9.36 (1.51)	0.70 (0.18)	0.66 (0.16)	0.90 (0.22)
				$J = 50$		
50	MSE	17.92 (37.17)	0.92 (0.21)	1.58 (0.54)	1.72 (0.53)	1.59 (0.63)
	MAE	27.95 (16.94)	67.79 (9.72)	15.99 (1.41)	18.46 (1.63)	30.07 (4.92)
	MSPE	95.88 (17.93)	4.75 (1.59)	0.05 (0.06)	0.06 (0.06)	0.03 (0.03)
	MAPE	98.14 (7.70)	21.31 (2.86)	1.52 (0.33)	1.78 (0.33)	1.44 (0.24)
100	MSE	8.99 (11.26)	1.18 (0.20)	1.89 (0.39)	1.97 (0.38)	1.67 (0.39)
	MAE	27.25 (6.40)	79.89 (8.63)	16.61 (1.08)	18.30 (1.15)	35.22 (4.50)
	MSPE	34.48 (7.55)	2.55 (1.03)	0.01 (0.01)	0.01 (0.01)	0.01 (0.01)
	MAPE	52.09 (3.97)	13.71 (2.09)	0.72 (0.13)	0.82 (0.13)	0.86 (0.10)
200	MSE	55.03 (103.10)	1.42 (0.21)	1.78 (0.22)	1.95 (0.25)	1.47 (0.26)
	MAE	44.02 (21.26)	90.77 (7.97)	16.68 (0.95)	17.28 (0.97)	33.80 (3.34)
	MSPE	10.61 (1.96)	0.96 (0.34)	0.01 (0.01)	0.00 (0.01)	0.01 (0.01)
	MAPE	26.31 (1.50)	8.06 (1.11)	0.44 (0.08)	0.42 (0.07)	0.58 (0.09)
				$J = 100$		
50	MSE	51.26 (67.01)	0.86 (0.18)	1.60 (0.45)	1.76 (0.46)	1.63 (0.52)
	MAE	44.17 (28.38)	64.77 (8.12)	15.26 (1.43)	17.75 (1.71)	30.17 (6.24)
	MSPE	100.00 (15.48)	3.97 (1.47)	0.02 (0.01)	0.03 (0.02)	0.01 (0.01)
	MAPE	99.17 (5.87)	19.48 (2.81)	1.25 (0.18)	1.46 (0.19)	1.16 (0.12)
100	MSE	29.50 (46.88)	1.08 (0.21)	1.97 (0.38)	2.06 (0.37)	1.78 (0.41)
	MAE	35.96 (16.61)	74.74 (9.69)	16.14 (0.87)	17.79 (0.87)	35.72 (4.59)
	MSPE	34.01 (6.42)	1.82 (0.61)	0.01 (0.01)	0.01 (0.01)	0.01 (0.00)
	MAPE	51.41 (2.44)	12.07 (1.80)	0.60 (0.07)	0.67 (0.08)	0.72 (0.05)
200	MSE	8.46 (15.70)	1.24 (0.19)	1.88 (0.22)	2.02 (0.25)	1.53 (0.25)
	MAE	27.67 (4.87)	83.31 (8.14)	15.45 (0.75)	16.09 (0.71)	33.24 (3.52)
	MSPE	10.99 (2.30)	0.74 (0.26)	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)
	MAPE	26.51 (1.22)	7.02 (0.88)	0.35 (0.02)	0.35 (0.02)	0.47 (0.03)

Table 3: Results of simulation study 3. Average MSE, MAE, MSPE, and MAPE are reported for the 5 estimators considered across the 50 simulated data sets for each parameter setting. Standard errors are reported in parentheses. Each metric is scaled to lie between 0 and 100.

	MLR	Lasso	Ridge	Elastic Net	Entropy
MSPE	NA	NA	NA	NA	6064933 (5923124)
MAPE	NA	NA	NA	NA	462 (208)

Table 4: Empirical performance of the 5 estimators considered on the GvHD data measured by average MSPE and MAPE, with standard errors in parentheses. Only entropic MLR was able to converge and produce estimates.

3.4 Microbiome-GvHD Application

For the microbiome-GvHD data set, only our EMLR method was able to converge and compute MAPE and MSPE (Table 4), while the other methods had issues with convergence due to the sparsity of the taxa in the training data sets.

Though the original data set does not include taxa that none of the subjects have, when we divide the complete case data set into smaller training and testing data sets, 70% of the training data sets generated had at least one taxon that none of the individuals in the training set had. Thus, the probability of an individual having that taxon was exactly 0, and MLR and the standard regularization methods failed to converge. Though at least one subject had every taxa in the remaining 30% of the training data sets, the probabilities of some taxa were less than 10^{-5}, and the MLR and penalized MLR methods failed to converge as well. One workaround we considered was to exclude all the taxa with probabilities below the threshold 10^{-5}. However, eliminating those taxa as categories in the response variable changes the relative probabilities of the other taxa and constrains all future predictions of those taxa to be 0, regardless of if new subjects have that taxa or not. Thus, altering the data set could skew the results and worsen predictions.

4 Discussion

In this final section, we summarize our study and discuss the advantages and limitations of our model relative to those already present in the statistical literature. We conclude with discussing possible future directions we can take with this project.

4.1 Model Performance and Evaluation

In this study, we developed a novel entropy regularized multinomial logistic regression method to handle multivariate, high-dimensional, zero-inflated and linearly separable data under a prediction framework. From our simulation studies and application to a gut microbiome data set, we found that EMLR outperforms multinomial logistic regression and standard regularized MLRs in several scenarios, when comparing mean absolute error, mean squared error, mean absolute prediction error, and mean squared prediction error. We found that EMLR can achieve the minimal prediction errors even when the covariates are positively correlated and a significant fraction of the response categories are linearly separable. We note, however, that since EMLR performed better with the prediction metrics, MAPE

and MSPE, compared to MAE and MSE, this suggests that our method gives more accurate predictions than estimates of the coefficients. Thus, we should not over-interpret our estimated coefficients in the prediction setting. We also found that EMLR is able to predict on data sets where the other methods fail to converge, such as on data where some of the response categories are linearly separable due to an abundance of zeros, which is common in sequencing data.

4.2 Future Work

Since the MLR + EN had similar or better predictive performance compared to EMLR in some cases, in the future, we should expand our simulation studies and explore fitting elastic nets with different mixing parameters. We should also compare EMLR to the method discussed in the data application of constraining the fitted parameters of taxa with too small of probabilities to zero and then fitting MLR and regularized MLR methods. Additionally, we should include simulation scenarios with response variables with more categories and response variables with categories with small probabilities, such as we saw in the data application.

Another possible improvement we can make to our project is speeding up computation time. We note that the code used to perform data generation, train our statistical model, and employ standard machine learning techniques is all written in a serial fashion. In accordance with parallel computing, if we were to parallelize our code and make use of such parallel computing packages such as `snow`, we may be able to increase computational efficiency. This would enable us to potentially test our model's performance on much larger data sets (both synthetic and real). Furthermore, there are also cluster computing software we can make use of to interface our parallelized R code with the command line and the cluster. This message passing interface (MPI) would enable us to further speed up computational efficiency.

Additionally, we could broaden our prediction method to carry out inference by constructing bootstrapped confidence intervals. We could then compare our method to other inference methods for multivariate categorical analyses. In future work, we can also explore the theoretical properties of the entropic MLR estimator. Although we have exhibited simulation studies and experimental data analysis demonstrating the competitive, and often superior, predictive performance of entropic MLR, it would be reassuring to obtain statistical guarantees for the estimator, for example in the form of a prediction error bound, as exist for the other regularization methods considered in this paper. Given the form of the EMLR penalty, it may be fairly straightforward to obtain a high probability bound on the relative entropy (or KL divergence) between the true probabilities p_{ij} and the EMLR fitted probabilities $\tilde{p}_{ij}$ when the multinomial model is correctly specified. We believe that a detailed theoretical treatment of entropic MLR may shed some light on why the estimator exhibits competitive and often superior performance to the benchmark regularization methods.

Our entropic regularization method can also be extended to the Bayesian setting. Consider a Multinomial model with a single observation Y for simplicity

$$Y|\alpha \sim \text{Multinomial}(n, \alpha = \{\alpha_1, \ldots, \alpha_J\}).$$

We can specify the following prior density on the model parameters α, which lie in the

simplex:

$$\pi(\alpha|\epsilon) \propto \exp\left(-\epsilon \sum_j \alpha_j \log \alpha_j\right).$$

In this case, the corresponding posterior density of $\alpha|Y$ is precisely the entropy-regularized MLR objective $\tilde{\ell}(\alpha)$. Consequently, the MAP estimator for this model is given by our EMLR estimator. Here ϵ can be considered a constant, or we can go fully Bayesian by specifying a prior distribution for ϵ on $[0, \infty)$. Since we know the functional form of the posterior density up to a constant, we can also sample from this distribution using the Metropolis-Hastings MCMC algorithm, for example, to estimate the full posterior.

It might also be interesting to extend this framework to a Multinomial-Dirichlet, or just a Dirichlet model to handle compositional data rather than counts, by imposing entropy regularization via an appropriately chosen prior distribution. For example, we can parametrize any Dirichlet(α) model as $\alpha_j = \gamma\varphi_j$, where $\gamma, \varphi_j > 0$ and $\sum_j \varphi_j = 1$. Penalizing the entropy of the φ_j could help to stabilize estimation in this setting.

References

[1] Peter J Turnbaugh, Ruth E Ley, Micah Hamady, Claire M Fraser-Liggett, Rob Knight, and Jeffrey I Gordon. The human microbiome project. *Nature*, 449(7164):804–810, 2007.

[2] Ernst Holler, Peter Butzhammer, Karin Schmid, Christian Hundsrucker, Josef Koestler, Katrin Peter, Wentao Zhu, Daniela Sporrer, Thomas Hehlgans, Marina Kreutz, et al. Metagenomic analysis of the stool microbiome in patients receiving allogeneic stem cell transplantation: loss of diversity is associated with use of systemic antibiotics and more pronounced in gastrointestinal graft-versus-host disease. *Biology of Blood and Marrow Transplantation*, 20(5):640–645, 2014.

[3] Nathan D Mathewson, Robert Jenq, Anna V Mathew, Mark Koenigsknecht, Alan Hanash, Tomomi Toubai, Katherine Oravecz-Wilson, Shin-Rong Wu, Yaping Sun, Corinne Rossi, et al. Gut microbiome–derived metabolites modulate intestinal epithelial cell damage and mitigate graft-versus-host disease. *Nature immunology*, 17(5):505–513, 2016.

[4] Pamela N Luna, Jonathan M Mansbach, and Chad A Shaw. A joint modeling approach for longitudinal microbiome data improves ability to detect microbiome associations with disease. *PLoS computational biology*, 16(12):e1008473, 2020.

[5] Mei Dong, Longhai Li, Man Chen, Anthony Kusalik, and Wei Xu. Predictive analysis methods for human microbiome data with application to parkinson's disease. *Plos one*, 15(8):e0237779, 2020.

[6] Tianchen Xu, Ryan T Demmer, and Gen Li. Zero-inflated poisson factor model with application to microbiome read counts. *Biometrics*, 2020.

[7] Xinyan Zhang and Nengjun Yi. Fast zero-inflated negative binomial mixed modeling approach for analyzing longitudinal metagenomics data. *Bioinformatics*, 36(8):2345–2351, 2020.

[8] Matthew CB Tsilimigras and Anthony A Fodor. Compositional data analysis of the microbiome: fundamentals, tools, and challenges. *Annals of epidemiology*, 26(5):330–335, 2016.

[9] Gerhard Tutz and Jan Gertheiss. Regularized regression for categorical data. *Statistical Modelling*, 16(3):161–200, 2016.

[10] Scott J Cook, John Niehaus, and Samantha Zuhlke. A warning on separation in multinomial logistic models. *Research & Politics*, 5(2):2053168018769510, 2018.

[11] Taedong Kim and Stephen J Wright. Pmu placement for line outage identification via multinomial logistic regression. *IEEE Transactions on Smart Grid*, 9(1):122–131, 2016.

[12] Bob Carpenter. Lazy sparse stochastic gradient descent for regularized multinomial logistic regression. *Alias-i, Inc., Tech. Rep*, pages 1–20, 2008.

[13] Lianru Gao, Jun Li, Mahdi Khodadadzadeh, Antonio Plaza, Bing Zhang, Zhijian He, and Huiming Yan. Subspace-based support vector machines for hyperspectral image classification. *IEEE Geoscience and Remote Sensing Letters*, 12(2):349–353, 2014.

[14] Mahesh Pal. Multinomial logistic regression-based feature selection for hyperspectral data. *International Journal of Applied Earth Observation and Geoinformation*, 14(1):214–220, 2012.

[15] Brian D Ripley. *Modern applied statistics with S*. springer, 2002.

[16] Trevor Hastie, Robert Tibshirani, and Martin Wainwright. *Statistical Learning with Sparsity: the Lasso and Generalizations*. Chapman and Hall/CRC Press, 2015.

[17] Yurii Nesterov. Gradient methods for minimizing composite objective function, technical report 76. *Center for Operations Research and Econometrics (CORE), Catholic University of Louvain (UCL)*, 2007.

[18] Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1):1, 2010.