A sparse negative binomial classifier with covariate adjustment for RNA-seq data

Tanbin Rahman[1], Hsin-En Huang[2], An-Shun Tai[2], Wen-Ping Hsieh[2,*], George Tseng[1,*]

1 University of Pittsburgh
2 National Tsing Hua University

*** ctseng@pitt.edu**

Abstract

Supervised machine learning methods have been increasingly used in biomedical research and in clinical practice. In transcriptomic applications, RNA-seq data have become dominating and have gradually replaced traditional microarray due to its reduced background noise and increased digital precision. Most existing machine learning methods are, however, designed for continuous intensities of microarray and are not suitable for RNA-seq count data. In this paper, we develop a negative binomial model via generalized linear model framework with double regularization for gene and covariate sparsity to accommodate three key elements: adequate modeling of count data with overdispersion, gene selection and adjustment for covariate effect. The proposed method is evaluated in simulations and two real applications using cervical tumor miRNA-seq data and schizophrenia post-mortem brain tissue RNA-seq data to demonstrate its superior performance in prediction accuracy and feature selection.

In the past two decades, microarray and RNA sequencing (RNA-seq) are routine procedures to study transcriptome of organisms in modern biomedical studies. In recent years, RNA-seq [5, 20] has become a popular experimental approach for generating a comprehensive catalog of protein-coding genes and non-coding RNAs [13], and it largely replaces the microarray technology due to its low background noise and increased precision. The most important difference between RNA-seq and microarray technology is that RNA-seq outputs millions of sequencing reads rather than the continuous fluorescent intensities in microarray data. Unlike microarray, RNA-seq can detect novel transcripts, gene fusions, single nucleotide variants, and indels (insertion/deletion). It can also detect a higher percentage of differentially expressed genes than microarray, especially for genes with low expression [24].

In machine learning, classification methods are used to construct a prediction model based on a training dataset with known class label so future independent samples can be classified with high accuracy. For example, labels in clinical research can be case/control, disease subtypes, drug response or prognostic outcome. Many popular machine learning methods have been widely applied to microarray studies, such as linear discriminate analysis [9], support vector machines [3] and random forest [7]. However, for discrete data nature in RNA-seq, many powerful tools for microarray assuming continuous data input or Gaussian assumption may be inappropriate. A common practice to solve this problem is to transform RNA-seq data into continuous values such as FPKM or TPM [6] and possibly taking additional log-transformation. However, such data transformation can lead to loss of information from the original data [14, 18], producing less accurate inference. Particularly, the transformation often

produces greater loss of information for genes with lower counts [15]. To accommodate discrete data in RNA-Seq, Poisson distribution and negative binomial distribution are two common distributions expected to better fit the data generation process and data characteristics. Witten [22] proposed a sparse Poisson linear discriminant analysis (sPLDA) based on Poisson assumption for the count data. However, Poisson distribution assumes equal mean and variance, which is often not true. In real RNA-seq data, the variance is often larger than the mean, leading to the need of an overdispersion parameter. Witten [22] reconciled this problem by proposing a power transformation to the data for eliminating overdispersion. However, as we will see later, the power transformation can perform well when the overdispersion is small but performs poorly when overdispersion becomes large. Hence, direct modeling by negative binomial assumption rather than a Poisson distribution is more appropriate. To this end, Dong et al. [8] proposed negative binomial linear discriminant analysis (denoted as NBLDA_{PE}) by adding a dispersion parameter. They, however, borrowed the point estimation from sPLDA in [22] and did not pursue a principled inference such as maximum likelihood, consequently producing worse performance than the method we will propose later.

Since the number of genes is often much larger than the number of samples in transcriptomic studies (a standard "small-n-large-p" problem), feature selection is critical to achieve better prediction accuracy and model interpretation. Witten [22] proposed a somewhat ad hoc soft-thresholding operator, similar to univariate Lasso estimator in regression, for gene selection in sPLDA but the method is not applicable to the NBLDA_{PE} model due to the addition of dispersion parameter. In the NBLDA_{PE} model proposed by [8], feature selection issue was not discussed, except that they used "edgeR" package to reduce the number of genes in the input data. Such a two-step filtering method is well-known to have inferior performance than methods with embedded feature selection. In fact, Zararsiz et al. [23] have compared sPLDA and NBLDA_{PE}, and showed that the power transformed sPLDA generally performed better than NBLDA_{PE} in their simulations and the worse performance in NBLDA_{PE} mainly came from the lack of feature selection. Finally, another critical factor to consider in transcriptomic modeling is the adjustment of covariates such as gender, race and age since it is well-known that many genes are systematically impacted by these factors. For example, Peters et al. [16] have identified 1,497 genes that are differentially expressed with age in a whole-blood gene expression meta-analysis of 14,983 individuals. A classification model allowing for covariate adjustment is expected to provide better accuracy and deeper biological insight.

To account for all aforementioned factors, we propose a sparse negative binomial model (snbClass) for classification analysis with covariate selection and adjustment. The method is based on generalized linear model (GLM) with a first regularization for feature sparsity. The GLM framework also allows straightforward covariate adjustment and a second regularization term on covariates, facilitating further covariate selection. Such covariate adjustment is not possible through existing sPLDA or NBLDA_{PE} methods. The paper is structured as following. In Section 1.1, we will briefly describe the two existing methods sPLDA [22] and NBLDA_{PE} [8], and then followed by our proposed methods $\text{sNBLDA}_{\text{GLM}}$ and $\text{sNBLDA}_{\text{GLM.sC}}$ in Section 1.2. Section 1.3 and 1.4 will discuss parameter estimation and model selection of the proposed method. Benchmarks for evaluation are described in Section 1.5. Section 2 presents simulation studies and Section 3 shows two real applications of cervical tumor miRNA data and schizophrenia RNA-seq data. Conclusion and discussion are included in Section 4.

1 Existing and proposed methods

In this section, we will first describe two existing methods for classification analysis of count data from RNA-seq and then propose our new method. To unify the notation, denote by $\mathbf{X}$ the count data matrix with elements X_{ij} referred to the sequence count for the j-th gene and the i-th sample ($i = 1, 2, \dots n$ and $j = 1, 2, \dots p$). In addition, $\mathbf{x_i} = (X_{i1} \dots X_{ip})^T$ denotes $i-$th row of $\mathbf{X}$, corresponding to feature measurements of observation i. Also, define $X_{.j} = \sum_{i=1}^{n} X_{ij}$, $X_{i.} = \sum_{j=1}^{p} X_{ij}$ and $X_{..} = \sum_{i,j} X_{ij}$. Moreover, in the classification setting where each observation belongs to one of the K classes, we let disjoint sets $C_k \subset \{1, \dots, n\}$ contain the indices of observations in class k. That is, class label $y_i = k$ if and only if $i \in C_k$. Furthermore, we denote $X_{C_k j} = \sum_{i \in C_k} X_{ij}$.

1.1 Two existing methods for classification of RNA-seq data

1.1.1 Sparse Poisson linear discrimination analysis (sPLDA)

Witten [22] introduced a log-linear Poisson model with feature selection, which resulted in a simple diagonal linear discriminant analysis suitable for count data (referred as "sPLDA" hereafter in this paper). Under the assumption of gene independence, the model is based on the following formulation,

$$X_{ij}|y_i = k \sim \text{Poisson}(N_{ij} \cdot d_{kj}), \quad N_{ij} = s_i \cdot g_j,$$

where s_i is the normalizing factor (a.k.a. size factor) for sample i and g_j is the ground mean for the j-th gene, allowing for variations both in samples and genes. For a given gene j, $d_{1j}, \dots, d_{kj}$ allows the j-th gene to be differentially expressed between the classes if any of $d_{kj} \neq 1 (1 \leq k \leq K)$.

RNA-Seq data often contain over-dispersion such that variances are larger than means, whereas an important constraint in Poisson model is the equivalent mean and variance. To overcome this, Witten [22] proposed a transformation of count data $X'_{ij} \leftarrow X^u_{ij}$ with a proper choice of u such that, $\sum_{i=1}^{n} \sum_{j=1}^{p} \frac{(X'_{ij} - X'_{i.} X'_{.j} / X'_{..})^2}{X'_{i.} X'_{.j} / X'_{..}} \approx (n-1)(p-1)$. From simulations of the original paper, this correction performs well in the presence of weak to moderate overdispersion.

Suppose $\mathbf{x}^* = (X^*_1, \dots, X^*_p)^T$ be a future new sample for prediction. The discriminant score for assigning $\mathbf{x}^*$ to class k is,

$$\log p(y^* = k|\mathbf{x}^*) = \sum_{j=1}^{p} X^*{}_j \cdot \log \hat{d}_{kj} - s^* \cdot \left(\sum_{j=1}^{p} \hat{g}_j \cdot \hat{d}_{kj} \right) + \log \hat{\pi}_k + c'$$

where y^* is the predicted label, $\hat{g}_j = X_{.j}$, $\hat{\pi}_k$ is the estimate of prior probability of belonging to the kth class estimated by the fraction of samples belonging to class k and s^* is the estimated normalization factor for the new sample $\mathbf{x}^*$ for which we do not know the class label. The classifier assigns $\mathbf{x}^*$ to the class with the largest discriminant score. The paper also implemented a somewhat ad hoc soft-thresholding operator for feature selection in the classifier, which is motivated from univariate lasso regularization in regression for feature selection: $\hat{d}_{kj} = 1 + S(a/b - 1, v/\sqrt{b})$, where $a = X_{C_k j} + \beta$, $b = \sum_{i \in C_k} \hat{N}_{ij} + \beta$, β is the hyperparameter pre-determined in the estimation of d_{kj}, v is the tuning parameter chosen by cross validation and $S(x, a) = \text{sign}(x)(|x| - a)_+$ is the soft thresholding parameter. $\hat{d}_{1j} = \hat{d}_{2j} = \dots = \hat{d}_{kj} = 1$ means gene j is not differentially expressed across the classes and thus, is not selected in the classifier.

1.1.2 Negative binomial linear discrimination analysis ($\mathbf{NBLDA_{PE}}$)

Dong et al. [8] extended sPLDA into a negative binomial model to explicitly allow overdispersion property in RNA-seq data:

$$X_{ij}|y_i = k \sim \text{NB}(\mu_{ij} \cdot d_{kj}, \phi_j), \quad \mu_{ij} = s_i \cdot g_j$$

Under the formulation, $E(X_{ij}) = \mu_{ij}$ and $Var(X_{ij}) = \mu_{ij} + \mu_{ij}^2/\phi_j$. Similar to sPLDA, for a new observation $\mathbf{x}^*$, prediction is made by the maximized discriminant score:

$$\log P(y^* = k|\mathbf{x}^*) = \sum_{j=1}^{p} X_j^*[\log \hat{d}_{kj} + \log \hat{g}_j - \log(\phi_j + s^* \hat{g}_j \hat{d}_{kj})] - \sum_{j=1}^{p} \phi_j \log(\phi_j + s^* \hat{g}_j \hat{d}_{kj}) + \log \hat{\pi}_k + c',$$

where ϕ_j is the dispersion parameter for the jth gene, $\hat{d}_{kj} = (\sum_{i \in C_k} X_{ij} + 1)/(\sum_{i \in C_k} \hat{s}_i X_{\cdot j} + 1)$ and $\hat{g}_j$ is the same as defined previously. We note that the point estimate of $\hat{d}_{kj}$ and $\hat{g}_j$ are borrowed directly from Witten's sPLDA model without theoretical justification and the similar soft-thresholding in sPLDA cannot be easily incorporated into the procedure due to the increased complexity with ϕ_j.

In the literature, several popular procedures have been used for estimating the size factor, including simple sum of counts, median ratio [1] and quantile method [4]. Witten [22] and Dong et al. [8] showed that the performance is comparable among the three methods. Here, we will use the quantile method for all methods for a fair comparison. In quantile method, the normalization factor for sample i $(1 \leq i \leq n)$ is estimated as $s_i = q_i / \sum_{i=1}^{n} q_i$ (or equivalently some papares also use $s_i = n \cdot q_i / \sum_{i=1}^{n} q_i$, which is what we adopt in this paper), where q_i is the 75th quantile of sequence counts of all genes for the ith sample. For a new sample $\mathbf{x}^*$, the normalizing factor is estimated as $s^* = q^* / \sum_{i=1}^{n} q_i$, where q_i $(1 \leq i \leq n)$ come from training data and q^* is the 75th count quantile for sample $\mathbf{x}^*$. Note that the vector of normalization factors and dispersion denoted by $\mathbf{s}$ and $\boldsymbol{\phi}$ respectively will be pre-estimated in all negative binomial models in this paper before inference. $\boldsymbol{\phi}$ are estimated by weighted likelihood empirical Bayes method using the edgeR package [17] with class label considered. We denote the method proposed by [8] as "NBLDA_{PE}" to emphasize the ad hoc "point estimation" procedure inherited from sPLDA in [22].

1.2 Proposed method: sparse negative binomial classifier via generalized linear model

We first consider a model without covariate in section 1.2.1. Then we extend it to covariate in section 1.2.2.

1.2.1 Sparse negative binomial classifier without covariate adjustment ($\mathbf{sNBLDA_{GLM}}$)

Similar to NBLDA_{PE}, we specify the following negative binomial model in a generalized linear model (GLM) setting:

$$X_{ij}|y_i = k \sim NB(\mu_{ijk}, \phi_j); \;\; \log(\mu_{ijk}) = \log(s_i) + \beta_{jk},$$

where s_i is the normalization factor of the i-th sample, β_{jk} is the mean count in log-scale of the k-th class for the j-th gene and ϕ_j is the dispersion parameter of the j-th gene. Under the assumption of independence between genes, the corresponding log-likelihood can be written as,

$$\log L(\Theta, \boldsymbol{\phi}; \mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} \sum_{k=1}^{K} \left[I(y_i = k) \cdot \sum_{j=1}^{p} \log f(X_{ij}; \beta_{jk}, \phi_j) \right],$$

where, $\Theta = \{(\boldsymbol{\beta}_k, \boldsymbol{\phi}); k = 1, \ldots, K\}$, $\boldsymbol{\beta}_k = (\beta_{1k}, \ldots, \beta_{pk})$, $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_p)$, $I(y_i = k)$ is the indicator function taking value 1 if $y_i = k$ and 0 otherwise, and $f(X_{ij}; \beta_{jk}, \phi_j)$ is the density function of $\text{NB}(s_i \exp(\beta_{jk}), \phi_j)$. Now, suppose we have a new observation $\mathbf{x}^*$ for which we intend to predict the class label. By Bayes theorem, we can derive the discriminant score as

$$\log P(y^* = k|\mathbf{x}^*) \propto \log \hat{\pi}_k - \sum_{j=1}^{p} \phi_j \log[\phi_j + s^* \exp(\hat{\beta}_{jk})] + \sum_{j=1}^{p} X_j^* [\hat{\beta}_{jk} - \log(\phi_j + s^* \exp(\hat{\beta}_{jk}))] \tag{1}$$

Here, $\mathbf{x}^*$ is assigned to class k for which the discriminant score is maximized. Note that the form of the discriminant score in the current model is identical to that proposed in [8], except that we reparametrize $\mu_{ijk} = s_i g_j d_{kj}$ to $\log(\mu_{ijk}) = \log(s_i) + \beta_{jk}$. The major difference is in the parameter estimation. [8] directly borrows the point estimation of μ_{ijk} from the Poisson model in [22], while we will derive MLE of Equation (2) (see below) using iteratively reweighted least squares (IRLS) method to be shown in the next subsection.

In order to incorporate variable (gene) selection, we add a penalty term $h(\beta) = \sum_{k=1}^{K} \sum_{j=1}^{G} |\beta_{jk} - \bar{\beta}_j|$. Here, $\bar{\beta}_j$ is the average of β_{jk}'s over the K classes for a given j-th gene. Hence, the following penalized likelihood is maximized to obtain estimation of β with pre-estimated $\boldsymbol{\phi}$:

$$\log L(\beta; \mathbf{x}, \mathbf{y}, \boldsymbol{\phi}) = \sum_{i=1}^{n} \sum_{k=1}^{K} \left[I(y_i = k) \cdot \sum_{j=1}^{p} \log f(X_{ij}; \beta_{jk}, \phi_j) \right] - \lambda h(\beta) \tag{2}$$

Here, λ is a tuning parameter controlling sparsity of the variable selection. The form of the discriminant scores for prediction is the same as in equation 1.

1.2.2 Sparse negative binomial classifier with covariate adjustment (sNBLDA$_{\text{GLM.C}}$ and sNBLDA$_{\text{GLM.sC}}$)

In real applications, information of multiple clinical variables is often available and some of them may be associated with subsets of genes. Commonly encountered clinical variables can include age, gender, race, etc. Failure of covariate adjustment can greatly reduce prediction accuracy and replicability. In our GLM framework, covariate adjustment can be straightforwardly incorporated in the linear regression term:

$$X_{ij}|y_i = k \sim NB(\mu_{ijk}, \phi_j); \;\; \log(\mu_{ijk}) = \log(s_i) + \beta_{jk} + \sum_{q=1}^{Q} \alpha_{qj} z_{iq}, \tag{3}$$

Here, $\mathbf{z_q} = (Z_{1q}, \ldots, Z_{nq})$ includes values of the q-th covariate over n samples and parameter α_{qj} corresponds to the coefficient of the q-th covariate in the j-th gene. Under the assumption of gene independence and adding penalty terms for both genes and covariates, the problem can be presented as maximization of the following penalized log-likelihood with double regularization:

$$\log L(\beta,\alpha;\mathbf{y},\mathbf{x},\mathbf{z_1},\ldots,\mathbf{z_Q},\boldsymbol{\phi}) = \sum_{i=1}^{n}\sum_{k=1}^{K} I(y_i=k)\sum_{j=1}^{p}\log f(X_{ij},Z_{i1},\ldots,Z_{iQ};\beta_{jk},\vec{\alpha}_j,\phi_j) \\ -\lambda_1 h(\beta) - \lambda_2\sum_{q=1}^{Q}\sum_{j=1}^{p}|\alpha_{qj}|, \tag{4}$$

where, β is the collection of all β_{jk} parameters and α is the collection of all α_{qj} parameters. λ_1 and λ_2 are tuning parameters controlling for levels of sparsity of variable selection in genes and covariates, respectively.

Similarly, for a new sample $\mathbf{x}^*$ with vector of clinical vector $\mathbf{z}^* = (z_1^*,\ldots z_Q^*)$ under this framework, we can derive the following discriminant score:

$$\log P(y^*=k|\mathbf{x}^*) \propto \log\hat{\pi}_k - \sum_{j=1}^{p}\phi_j\log[\phi_j + s^*\exp(\hat{\beta}_{jk}+\sum_{q}^{Q}z_q^*\hat{\alpha}_{qj})] \\ +\sum_{j=1}^{p}X_j^*[\hat{\beta}_{jk}+\sum_{q}^{Q}z_q^*\hat{\alpha}_{qj} - \log(\phi_j + s^*\exp(\hat{\beta}_{jk}+\sum_{q}^{Q}z_q^*\hat{\alpha}_{qj})] \tag{5}$$

As before, $\mathbf{x}^*$ is assigned to the class with the highest discriminant score. We note that when $\lambda_2 = 0$, Equation 4 performs covariate adjustment using all covariates for all genes without regularization in covariate parameters α_{qj}. We will denote this method as "sNBLDA$_{\text{GLM.C}}$". In this case, when the number of covariates Q becomes large, performance of parameter estimation and prediction accuracy are expected to decline. With proper choice of λ_2 in Equation (4), the method can adequately select a subset of covariates in each gene to improve the performance. For illustration purpose, we refer to this method as "sNBLDA$_{\text{GLM.sC}}$ " in this paper, where "sC" means sparsity on covariates. This is the method we recommend in general applications when clinical covariates are available and will be referred to as "snbClass" in the R package and future applications. When clinical covariates do not exist, the method naturally reduces to "sNBLDA$_{\text{GLM}}$".

1.3 Estimation in sNBLDA$_{\text{GLM}}$ and $sNBLDA_{GLM.sC}$

1.3.1 Estimation of sNBLDA$_{\text{GLM}}$

Maximizing the log-likelihood derived in Equation (2) is equivalent to minimizing the following penalized weighted least square function,

$$\sum_{i=1}^{n}\sum_{k=1}^{K}\left[I(y_i=k)\sum_{j=1}^{p}w_{ijk}(\tau_{ijk}-\log(s_i)-\beta_{jk})^2\right] + \lambda\sum_{j=1}^{p}\sum_{k=1}^{K}|\beta_{jk}-\bar{\beta}_j|, \tag{6}$$

where $w_{ijk} = \mu_{ijk}/(1+\phi_j^{-1}\mu_{ijk})$ and $\tau_{ijk} = \log(s_i)+\beta_{jk}+(x_{ij}-\mu_{ijk})/\mu_{ijk}$. Given the estimates at the t-th step, the updates of (t+1) step is:,

1. Calculate $w_{ijk}^{(t+1)} = \mu_{ijk}^{(t)}/(1+\phi_j^{-1}\mu_{ijk}^{(t)})$
2. Update $\tau_{ijk}^{(t+1)} = \log(s_i)+\beta_{jk}^{(t)}+(x_{ij}-\mu_{ijk}^{(t)})/\mu_{ijk}^{(t)}$
3. Solve $\beta_{jk}^{(t+1)} = \text{argmin}\ \frac{1}{2}\sum_i I(y_i=k)w_{ijk}^{(t+1)}(\tau_{ijk}^{(t+1)}-\log(s_i)-\beta_{jk})^2+\lambda|\beta_{jk}-\bar{\beta}_j|$

4. Update $\mu_{ijk}^{(t+1)} = \exp(\beta_{jk}^{(t+1)} + \log(s_i))$

This is repeated until convergence of $\hat{\beta}_{jk}$. The update of $\hat{\beta}_{jk}$ in Step (3) is given by,

$$\beta_{jk}^{(t+1)} = \bar{\beta}_j^{(t+1)} + \text{sign}(\tilde{\beta}_{jk}^{(t+1)} - \bar{\beta}_j^{(t+1)})$$
$$\left[\left|\frac{\sum_i w_{ijk}^{(t+1)}(\tau_{ijk}^{(t+1)} - \log(s_i)) - \lambda(1-1/K)\ \text{sign}(\tilde{\beta}_{jk}^{(t+1)} - \bar{\beta}_j^{(t+1)})}{\sum_i w_{ijk}^{(t+1)}}\right| - |\bar{\beta}_j^{(t+1)}|\right]_{(+)}$$

Here, $[\ ]_{(+)}$ is soft thresholding function such that $[u]_{(+)}$ takes the value u when u is positive and 0 otherwise, $\tilde{\beta}_{jk}^{(t+1)}$ is the estimate of β_{jk} under no penalization and $\bar{\beta}_j^{(t+1)} = \sum_{k=1}^{K} \tilde{\beta}_{jk}^{(t+1)}/K$.

1.3.2 Estimation of $sNBLDA_{GLM.sC}$

Similar to $\text{sNBLDA}_{\text{GLM}}$, the problem of maximizing the penalized log-likelihood in Equation (4) can be represented as minimizing the penalized weighted least square function given below in Equation (7),

$$\sum_{i=1}^{n}\sum_{k=1}^{K} I(y_i = k)\sum_{j=1}^{p} w_{ijk}(\tau_{ijk} - \log(s_i) - \beta_{jk} - \sum_{q=1}^{Q} z_{iq}\alpha_{qj})^2 + \lambda_1 \sum_{j=1}^{p}\sum_{k=1}^{K} |\beta_{jk} - \bar{\beta}_j| + \lambda_2 \sum_{q=1}^{Q}\sum_{j=1}^{p} |\alpha_{qj}| \quad (7)$$

where, $w_{ijk} = \mu_{ijk}/(1 + \phi_j^{-1}\mu_{ijk})$ and $\tau_{ijk} = \log(s_i) + \beta_{jk} + \sum_{q=1}^{Q} z_{iq}\alpha_{qj} + (x_{ij} - \mu_{ijk})/\mu_{ijk}$. The estimation of each of the $\hat{\beta}_{jk}$ and α_{qj} is given by the following algorithm. The steps involved in IRLS given the estimates obtained at the t-th step is given below,

1. Calculate $w_{ijk}^{(t+1)} = \mu_{ijk}^{(t)}/(1 + \phi_j^{-1}\mu_{ijk}^{(t)})$
2. Update $\tau_{ijk}^{(t+1)} = \log(s_i) + \beta_{jk}^{(t)} + \sum_{q=1}^{Q} z_{iq}\alpha_{qj} + (x_{ij} - \mu_{ijk}^{(t)})/\mu_{ijk}^{(t)}$
3. Solve $\beta_{jk}^{(t+1)} = \text{argmin}$ $\frac{1}{2}\sum_i w_{ijk}^{(t+1)}(\tau_{ijk}^{(t+1)} - \log(s_i) - \beta_{jk} - \sum_{q=1}^{Q} z_{iq}\alpha_{qj}^{(t)})^2 + \lambda_1|\beta_{jk} - \bar{\beta}_j| + \lambda_2 \sum_{q=1}^{Q}\sum_{j=1}^{p} |\alpha_{qj}^{(t)}|$
4. Solve $\alpha_{qj}^{(t+1)} = \text{argmin}\ \frac{1}{2}\sum_i \sum_{k=1}^{K} I(y_i = k) w_{ijk}^{(t+1)}(\tau_{ijk}^{(t+1)} - \log(s_i) - \beta_{jk}^{(t+1)} - \sum_{q=1}^{Q} z_{iq}\alpha_{qj})^2 + \lambda_1|\beta_{jk}^{(t+1)} - \bar{\beta}_j^{(t+1)}| + \lambda_2 \sum_{q=1}^{Q}\sum_{j=1}^{p} |\alpha_{qj}|$
5. Update $\mu_{ijk}^{(t+1)} = \exp(\beta_{jk}^{(t+1)} + \sum_{q=1}^{Q} z_{iq}\alpha_{qj}^{(t+1)} + \log(s_i))$

The steps are repeated until convergence of the parameters $\beta_{jk}, \alpha_{1j}, \ldots, \alpha_{qj}$. Then the penalized estimate of the parameters in step 3 and step 4 are respectively given by,

$$\beta_{jk}^{(t+1)} = \bar{\beta}_j^{(t+1)} + \text{sign}(\tilde{\beta}_{jk}^{(t+1)} - \bar{\beta}_j^{(t+1)}) \left[\left| \frac{\sum_i w_{ijk}^{(t+1)}(\tau_{ijk}^{(t+1)} - \log(s_i) - \sum_{q=1}^{Q} \alpha_{qj}^{(t)} z_{qj}) - \lambda_1(1-1/K)\ \text{sign}(\tilde{\beta}_{jk}^{(t+1)} - \bar{\beta}_j^{(t+1)})}{\sum_i w_{ijk}^{(t+1)}} \right| - |\bar{\beta}_j^{(t+1)}| \right]_{(+)}$$

$$\text{and, } \alpha_{qj}^{(t+1)} = \text{sign}(\tilde{\alpha}_{qj}) \left[|\tilde{\alpha}_{qj}| - \left| \frac{\lambda_2}{\sum_{i=1}^{n} \sum_{k=1}^{K} I(y_i=k) w_{ijk} z_{iq}^2} \right| \right]_{(+)} \text{ where,}$$

$$\tilde{\alpha}_{qj} = \sum_{i=1}^{n} \sum_{k=1}^{K} I(y_i = k) w_{ijk}^{(t+1)} (\tau_{ijk}^{(t+1)} - \log(s_i) - \beta_{jk}^{(t+1)} - \sum_{1 \le m \le Q, m \ne q} z_{im}\alpha_{mj}) / \sum_{i=1}^{n} \sum_{k=1}^{K} I(y_i = k) z_{iq}^2 w_{ijk}.$$

1.4 Selection of tuning parameters in regularization

Both $\text{sNBLDA}_{\text{GLM}}$ and $\text{sNBLDA}_{\text{GLM.sC}}$ methods involve selection of regularization parameters λ or (λ_1, λ_2). We apply V-fold cross validation as a tool to determin the tuning parameter [19]. For each given tuning parameter, we divide the dataset into V equal folds and samples in the K classes are split into V folds as even as possible. In each iteration, one fold is set aside as the test set and the remaining $(V-1)$ folds are used as the training set. The classifier is built from the training set and then validated in the test set for evaluating accuracy. This procedure is repeated until all V folds have been chosen as the test set and the averaged accuracy is calculated. The tuning parameter corresponding to the highest averaged accuracy is chosen for the final model construction. We apply 10-fold (V=10) cross validation for all simulations and real applications in this paper. We note that nested cross validation is used for real applications for a fair accuracy evaluation. In this case, the outer loop of 10-fold cross validation is conventionally used to estimate accuracy. In each cross validation, the 9 folds of training set undergo an inner loop of 10-fold cross validation to determine λ or (λ_1, λ_2).

1.5 Benchmarks for evaluation

Performance of different methods will be judged by two major criteria: accuracy of prediction and accuracy of feature selection. For prediction performance, simple averaged accuracy is used when true class labels are known:
Accuracy $= \frac{\text{Number of test samples correctly classified}}{\text{Number of test samples}}$. For feature selection performance, we derive the area under the curve (AUC) [2] values of the receiver operating characteristic (ROC) curves. We also evaluate the performance of $sNBLDA_{\text{GLM}}$, $sNBLDA_{\text{GLM.sC}}$ and sPLDA in terms of estimating the true parameters β_{jk} when the gene expression is affected by covariates. Here, we define RMSE$=\sqrt{(1/BpK) \sum_{b=1}^{B} \sum_{j=1}^{p} \sum_{k=1}^{K} (\hat{\beta}_{jk}^{(b)} - \beta_{jk})^2}$ where B is the number of datasets simulated.

2 Simulations

In this section, we will devise two simulation schemes to compare the performance of sPLDA and NBLDA_{PE} to our proposed model $\text{sNBLDA}_{\text{GLM}}$ and $\text{sNBLDA}_{\text{GLM.sC}}$ under different settings. In Simulation 1, there is no covariate effect over the expression levels of the genes. Here, we compare sPLDA, NBLDA_{PE} and $\text{sNBLDA}_{\text{GLM}}$ over different level of signal strength under three different levels of dispersion in the data. In Simulation 2, we develop a simulation scheme where two covariates are introduced which

can affect expression level of certain proportion of the genes. Here, we compare sPLDA, NBLDA_{PE}, $\text{sNBLDA}_{\text{GLM}}$ and $\text{sNBLDA}_{\text{GLM.sC}}$ in the presence of covariate effects.

In order to mimic real data, we use a real RNA-seq dataset downloaded from Gene Expression Omnibus (GEO, GSE47474) to retrieve key parameters and perform the simulation. The dataset includes 72 samples with 36 coming from HIV-1 transgenic and 36 from control rat strains [12]. We compute the mean counts of each gene over all samples to obtain an empirical distribution of mean counts, which will be used for obtaining baseline expression levels in all the simulations. Each simulation is repeated 100 times and the average result is reported.

2.1 Simulation settings

Simulation 1: Without covariate effect

In this simulation, we sample the count data by $x_{ij}|C_i = k \sim NB(s_i b_j \exp(\delta_{jk}\Delta_j), \phi_j)$ for each gene $j(1 \leq j \leq 1000)$ and sample $i(1 \leq i \leq 120)$ in class $k(1 \leq k \leq 3)$, where the number of informative feature is 300. The notation of the parameters as well as the settings are given below:

- The library size factor s_i is sampled from Unif(0.75,1.25) for each sample i.
- b_j is the baseline which is sampled from the empirical distribution of the mean expression described previously.
- δ_{jk} represents the pattern of genes j in class k. For all $\delta_{jk} \in \{-1, 0, 1\}$, 1 indicating a up-regulated trend of genes in this class relative to other classes, -1 indicating it is down-regulated and 0 indicating no difference.
- There exists three gene patterns for the 300 informative genes: $(\delta_{j1}, \delta_{j2}, \delta_{j3}) =$ (1,0,-1), (0,1,1) and (-1,-1,0). For non-informative genes, the pattern is (0,0,0).
- Sample the main effect size parameter Δ_j for each gene j from a truncated normal distribution $TN(\zeta, 0.1^2, \zeta/2, \infty)$, where ζ is the mean and values smaller than $\zeta/2$ are truncated.
- $\phi_j \sim TN(\nu, 0.1, 0, \infty)$ and ν is chosen as 1, 5 and 10 .
- 100 of the samples are used as training set and the remaining 1,000 samples are used as testing set

Simulation 2: Incorporating covariate effect

We sample the count data by $x_{ij}|C_i = k \sim NB(s_i b_j \exp(\delta_{jk}\Delta_j + \sum_{q=1}^{2} \gamma_{qj}\epsilon_{qj}z_{qi})), \phi_j)$ for each gene $j(1 \leq j \leq 1000)$ and sample $i(1 \leq i \leq 120)$ in class $k(1 \leq k \leq 3)$ with two covariates (z_1 and z_2; Q=2), where the number of informative feature is 300. The notation of parameters are as follows:

- We generate a binary covariate (e.g. gender) for each sample i from $Ber(0.5)(i.e. z_{1i} \sim Ber(0.5))$ and generate a continuous covariate (e.g. age) for each sample i from $Gamma(5, 10)$
- $\phi_j \sim TN(\nu, 0.1, 0, \infty)$ where $\nu \in \{10, 1\}$
- γ_{qj} represents the pattern of gene j in covariate q for all $\gamma_{qj} \in \{0, 1\}$; there exist three patterns: $(\gamma_{1j}, \gamma_{2j}) = (1, 1), (1, 0), (0, 1),$ and $(0, 0)$ with probablity ($\rho/3$,$\rho/3$,$\rho/3$ and 1-ρ) respectively. When $\rho = 0$, all genes are not impacted by covariates. We choose the proportion of covariate-impacted genes ρ to be 0.125, 0.25 and 0.5.

- Sample the main effect size parameter Δ_j for each gene j in class k from a truncated normal distribution $TN(0.25, 0.1^2, 0.125, \infty)$
- The effect size parameter of covariates ϵ_{qj} for each gene j in covariate q is drawn from the product of random sign (i.e. half probability to be 1 and half to be -1) and a truncated normal distribution $TN(\eta, 0.1^2, \eta/2, \infty) \times \kappa$ where κ takes value 1 with probability 0.5 and -1 otherwise. We use the different value of $\eta \in \{0.1, 0.3, 0.5, 0.7\}$ for different level of signal strength.
- Other parameters are set the same as Simulation 1 except that ζ is set at 0.25.
- 100 of the samples are used as training set and the remaining 1,000 samples are used as testing set.

2.2 Simulation results

Results of Simulation 1 are summarized in Figure 1. In Figure 1(a), average prediction accuracy of the three models sPLDA, NBLDA_{PE} and $\text{sNBLDA}_{\text{GLM}}$ were compared over three different levels of dispersions $\nu \in \{1, 5, 10\}$. The larger the value of ν, the smaller the level of dispersion in the simulated datasets. In all different levels of ζ and ν, $\text{sNBLDA}_{\text{GLM}}$ outperformed the other two methods. As expected, NBLDA_{PE} was superior to sPLDA when ν was small (large overdispersion) but their performances became comparable when ν was large, confirming good performance of power transformation to correct dispersion in sPLDA only for small overdispersion. Figure 1(b) shows results of variable selection by AUC. $\text{sNBLDA}_{\text{GLM}}$ clearly outperformed sPLDA in all cases while NBLDA_{PE} could not perform variable selection and was not applicable in this plot. The new method was also compared to three popular classification methods such as support vector machines (SVM), random forest (RF) and classification and regression tree (CART) in supplement Figure S3. The result showed inferior performance in these methods due to ignorance of count data and transformation to continuous inputs.

Figure 2 demonstrates results of Simulation 2 using sPLDA, NBLDA_{PE}, $\text{sNBLDA}_{\text{GLM}}$ (no covariate adjustment) and $\text{sNBLDA}_{\text{GLM.sC}}$ (with covariate adjustment and regularization) when varying percent of genes impacted by covariates $\rho = 0.125, 0.25$ and 0.5. Figure 2(a) shows averaged prediction accuracy of varying η and $\nu = 1$ or 10. When $\nu = 1$ (high level of dispersion), $\text{sNBLDA}_{\text{GLM.sC}}$ outperformed all other methods as the impact of covariates on gene expression η increased. The prediction accuracy for $\text{sNBLDA}_{\text{GLM.sC}}$ remained high with increased η due to its capacity of adjusting covariate effect, while prediction accuracy of the other three methods dropped with increased η although $\text{sNBLDA}_{\text{GLM}}$ still outperformed sPLDA and NBLDA_{PE}. When $\nu = 10$, similar pattern was observed. The margin between $\text{sNBLDA}_{\text{GLM.sC}}$ and $\text{sNBLDA}_{\text{GLM}}$ became much smaller but $\text{sNBLDA}_{\text{GLM.sC}}$ was still the best performer. Supplementary Figure S4 includes comparison with SVM, RF and CART, all of which performed much worse than $\text{sNBLDA}_{\text{GLM.sC}}$.

Variable selection performance between sPLDA, $\text{sNBLDA}_{\text{GLM}}$ and $\text{sNBLDA}_{\text{GLM.sC}}$ is shown in 2(b). Similarly, we observed stable and high performance of $\text{sNBLDA}_{\text{GLM.sC}}$ with increasing η, while performance of $\text{sNBLDA}_{\text{GLM}}$ dropped for increased η due to the lack of covariate adjustment. sPLDA performed the worst in all cases. It is intriguing that the variable selection gap between $\text{NBLDA}_{\text{GLM.sC}}$ and $\text{NBLDA}_{\text{GLM}}$ was larger in $\nu = 10$ than in $\nu = 1$, which is contrary to the prediction accuracy in Figure 2(a). An evaluation of the parameter estimates between sPLDA, $\text{sNBLDA}_{\text{GLM}}$ and $\text{sNBLDA}_{\text{GLM.sC}}$ was carried out in terms of RMSE in supplement Figure S1, where $\text{sNBLDA}_{\text{GLM.sC}}$ performed the best. To examine the advantage of covariate regularization, we compared $\text{sNBLDA}_{\text{GLM.C}}$ (i.e. $\lambda_2 = 0$ in Equation 4; all covariates

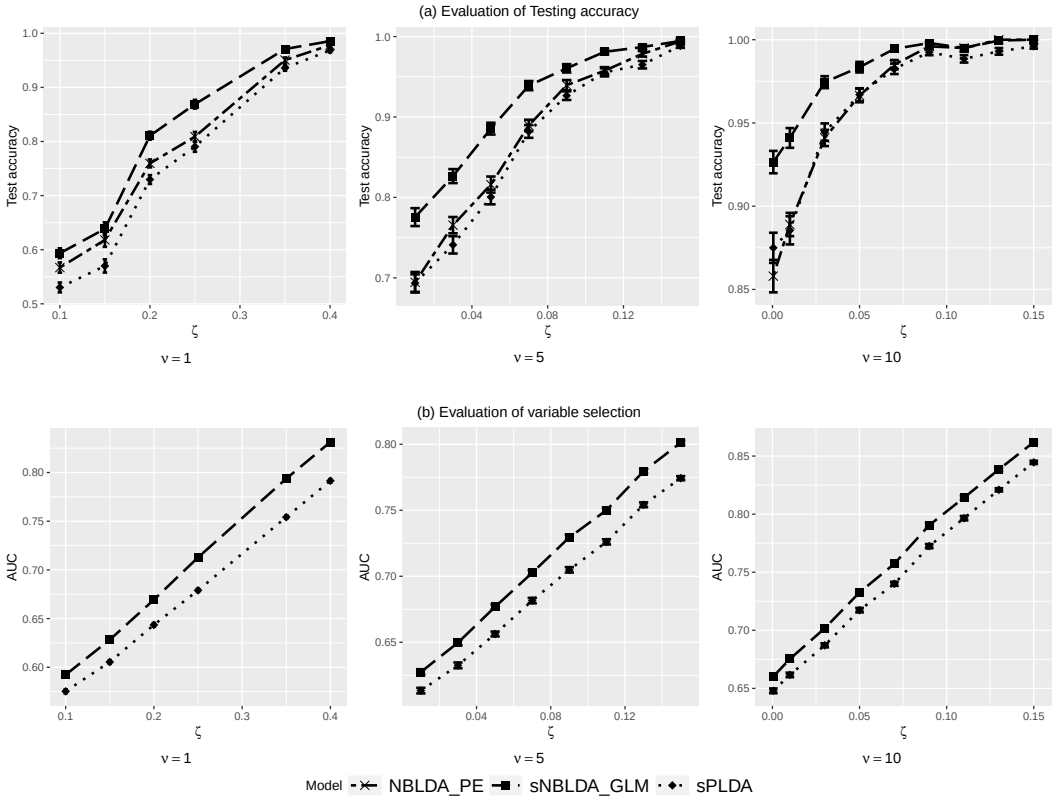

Figure 1. Results for Simulation 1 without covariate effect

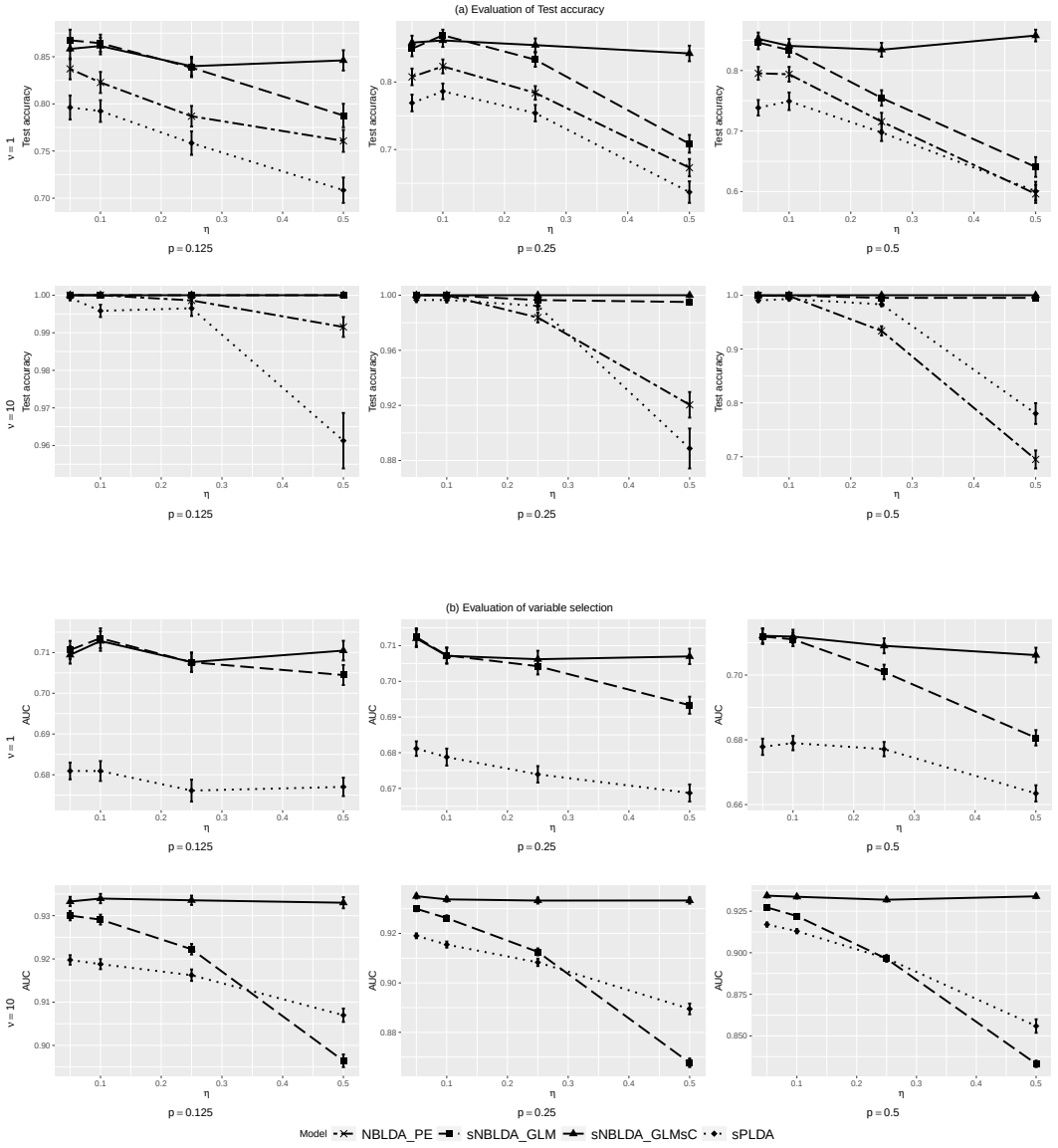

Figure 2. Results for Simulation 2 setting

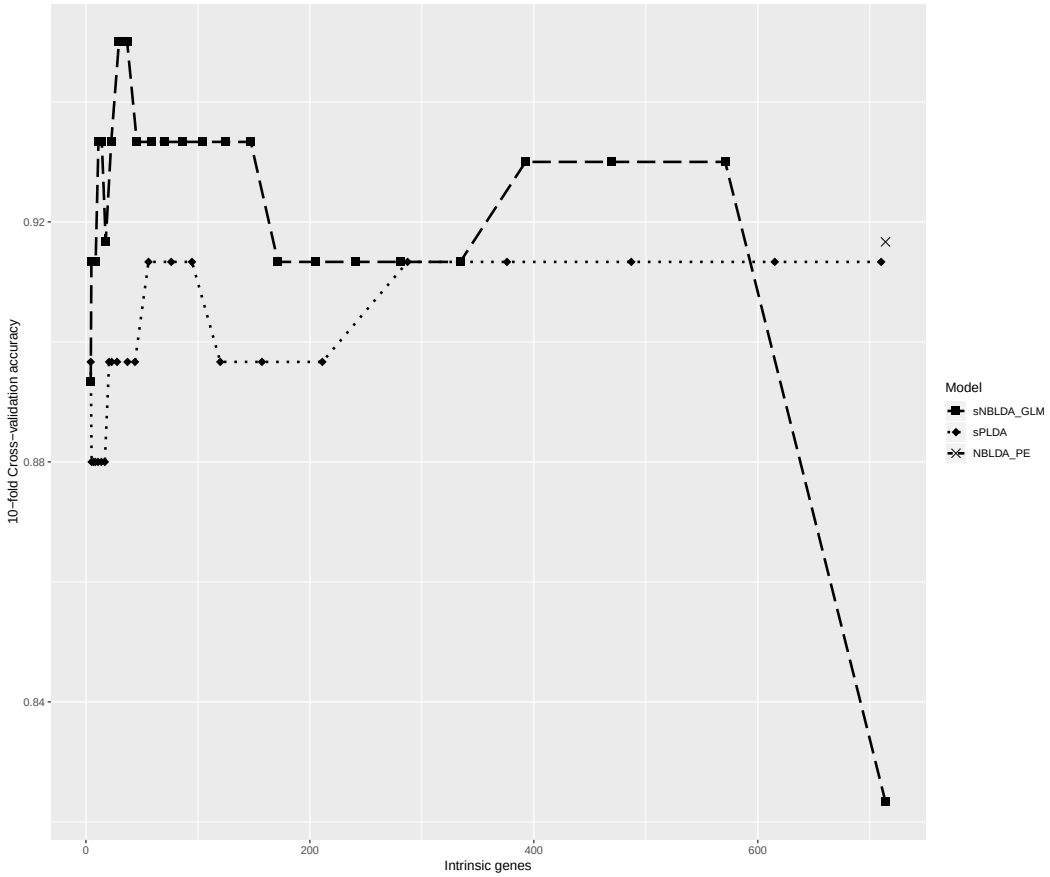

Figure 3. Prediction accuracy (y-axis) of $\text{sNBLDA}_{\text{GLM}}$ (dashed line) and sPLDA (dotted line) with varying number of selected miRNAs (x-axis) in the cervical tumor application. NBLDA_{PE} does not allow variable selection and is shown with "X" symbol.

are used) with $\text{sNBLDA}_{\text{GLM.sC}}$ (with covariate regularization) in Supplement Figure S2 . The result shows clear improvement of covariate regularization on prediction accuracy but less on feature selection.

3 Real applications

3.1 Cervical tumor miRNA data

This RNA-seq dataset measures expression level of miRNAs in tumor and nontumor cervical tissues in human samples [21]. The data contains information of over 714 microRNAs for 29 control samples (samples with no tumor) and 29 tumor samples. No clinical information (covariates) is available for adjustment. This dataset has been used in both sPLDA and NBLDA_{PE} papers and thus is a good dataset to evaluate our new method. [8] found that NBLDA_{PE} performed better than sPLDA in terms of prediction accuracy because of high dispersion estimate in this dataset. In Figure 3 , we compare prediction accuracy (y-axis) between sPLDA and $\text{sNBLDA}_{\text{GLM}}$ based on 10-fold cross-validation when different number of genes are selected (x-axis) as proposed for the corresponding models. Since there is no variable selection in NBLDA_{PE}, we only perform cross-validation considering all miRNAs (shown as "X" in the figure). $\text{sNBLDA}_{\text{GLM}}$ generally outperforms the other two methods in different number of

selected genes. Specifically, it achieves 95% prediction accuracy with a small number of 37 genes while NBLDA_{PE} and sPLDA generally achieves 91% accuracy. The result shows clear improvement of $\text{sNBLDA}_{\text{GLM}}$ in prediction accuracy and variable selection.

3.2 Schizophrenia RNA-seq dataset

The schizophrenia RNA-seq dataset (http://www.synapse.org/CMC) was obtained from the CommonMind Consortium [11] using post-mortem human dorsolateral prefrontal cortex tissues from 258 schizophrenia patients and 279 controls. Here we restrict our analysis to patients with age below 50 and post-mortem interval (PMI; time relapsed from the person has died to the tissues are frozen) less than 30 hours, producing 150 subjects where 100 are controls and 50 suffer from schizophrenia. Five clinical variables are available: age of death, gender, PMI, pH level and ethnicity (Caucasian or African American). At first, we ran a differential expression analysis on each covariate and found a higher percentage of DE genes affected by age of death, ethnicity, PMI and pH. However, since pH had some missing values, we only considered the other three clinical variables in the $\text{sNBLDA}_{\text{GLM.sC}}$ model. We performed routine data preprocessing and filtering to keep genes with at least 70% of the samples having gene expression level greater than 0 and mean count across the samples greater than 10, producing a count data matrix with 16989 genes for machine learning. Similar to simulation and previous application, 10-fold cross-validation was performed to evaluate sPLDA, NBLDA_{PE}, $\text{sNBLDA}_{\text{GLM}}$ and $\text{sNBLDA}_{\text{GLM.sC}}$. We further perform DE analysis to narrow down to 250, 500, 750, 1000, 1500, 2000 and 5000 genes in each training set before adopting the four machine learning methods.Even though three of the four methods have embedded feature selection capacity, the feature selection is usually difficult for ultra-high dimensionality (e.g. 16,989 gene features in our case). We performed a pre-screening by differential expression analysis to reduce dimensionality to 250-5000. This procedure is similar to the sure independence screening idea in [10] and can usually improve prediction performance. Figure 4 shows the 10-fold cross validation accuracy of the four methods for different gene size after DE analysis pre-selection. For the three methods with embedded feature selection (sPLDA, $\text{sNBLDA}_{\text{GLM}}$ and $\text{sNBLDA}_{\text{GLM.sC}}$), varied tuning parameter for feature selection was applied and the best prediction accuracy was reported in Figure 4. The result clearly demonstrates better prediction performance of $\text{sNBLDA}_{\text{GLM.sC}}$, especially when the pre-screening by DE analysis reduce the input gene size to 250-1000. However, when large number of genes are input to the $\text{sNBLDA}_{\text{GLM.sC}}$ algorithm (e.g. 2000 or 5000 genes after pre-screening), its performance dropped to close to $\text{sNBLDA}_{\text{GLM}}$ and the advantage of covariate adjustment is diminished. Nevertheless, our proposed GLM approach generally outperforms sPLDA and NBLDA_{PE}. As a result, we recommend pre-screening of ultra-high dimensional data, such as regular RNA-seq datasets, down to 250-5000 features before applying $\text{sNBLDA}_{\text{GLM.sC}}$. The result shows inferior performance of $\text{sNBLDA}_{\text{GLM.C}}$, showing necessity of covariate regularization. The accuracy performance of the methods discussed in this paper is compared with other methods appropriate for continuous data is summarized in Figure S5.

4 Conclusion and Discussion

In this paper, we proposed a sparse negative binomial classifier based on a GLM framework with and without covariate adjustment. The method incorporates three key elements in RNA-seq machine learning modeling: adequate modeling for count data, feature selection and adjustment of covariate effects. Existing methods such as sPLDA does not consider overdispersion properly, NBLDA_{PE} does not embed regularization for feature selection and both methods cannot adjust for covariate effect in gene expression.

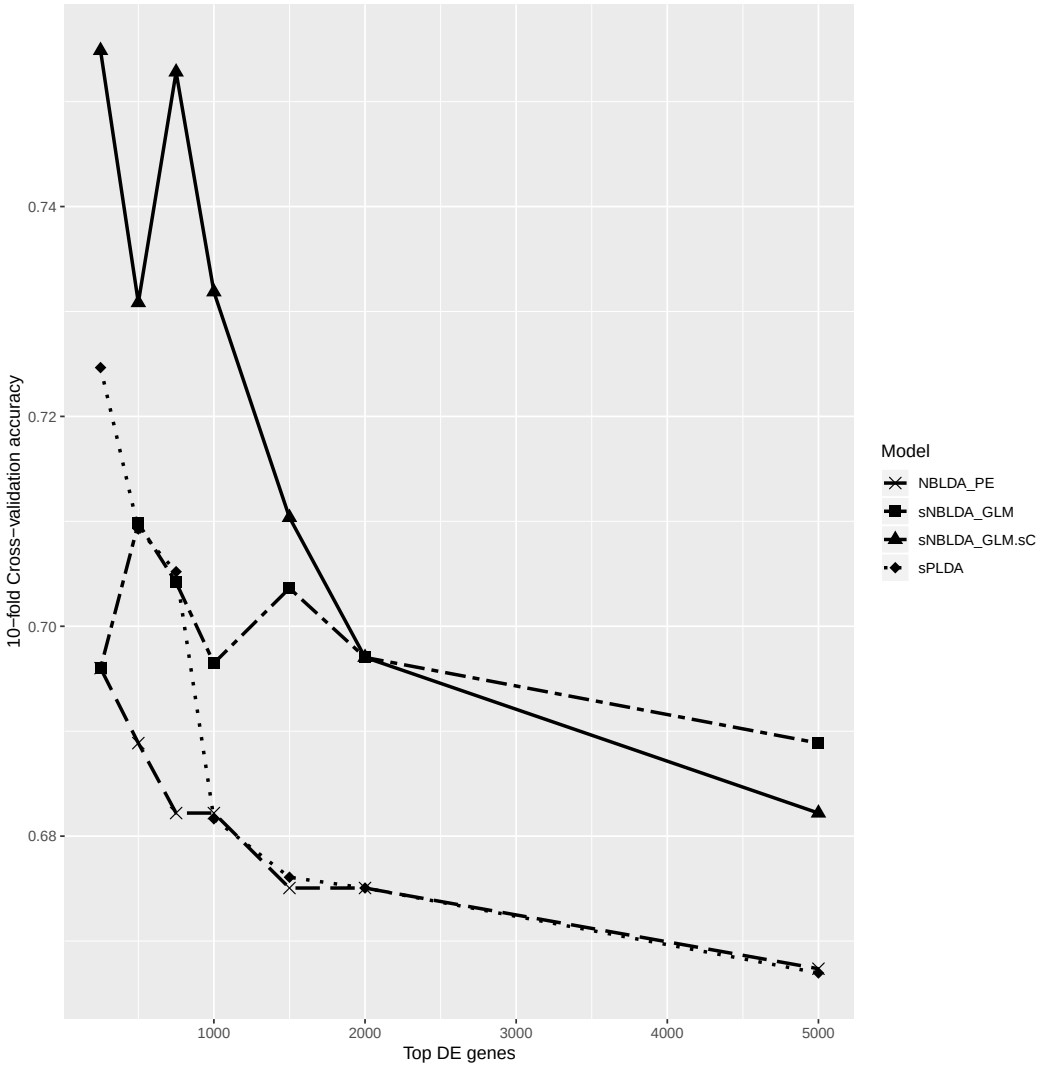

Figure 4. Prediction accuracy (y-axis) of $\text{sNBLDA}_{\text{GLM.sC}}$, $\text{sNBLDA}_{\text{GLM}}$, NBLDA_{PE} and sPLDA with varying input gene number after DE analysis pre-screening (x-axis) in the schizophrenia post-mortem brain RNA-seq data.

Our new approach assumes a negative binomial model to allow overdispersion, adopts GLM to allow covariate adjustment and facilitates double regularization for feature selection and covariate selection. Extensive simulations and two real applications showed superior performance of the proposed approach in terms of prediction accuracy and feature selection. Particularly, the new methods achieved higher prediction accuracy with smaller number of selected genes or miRNAs in the two real applications.

One major limitation of all methods in this paper is that the methods are based on independent assumption of gene expression. Due to the complex form of multivariate negative binomial model and the potentially heavy computational cost, this is not addressed in this paper but will be a future direction.

References

1. S. Anders and W. Huber. Differential expression analysis for sequence count data. *Genome biology*, 11(10):R106, 2010.

2. A. P. Bradley. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recogn.*, 30(7):1145–1159, July 1997.

3. M. P. Brown, W. N. Grundy, D. Lin, N. Cristianini, C. W. Sugnet, T. S. Furey, M. Ares, and D. Haussler. Knowledge-based analysis of microarray gene expression data by using support vector machines. *Proceedings of the National Academy of Sciences*, 97(1):262–267, 2000.

4. J. H. Bullard, E. Purdom, K. D. Hansen, and S. Dudoit. Evaluation of statistical methods for normalization and differential expression in mrna-seq experiments. *BMC bioinformatics*, 11(1):94, 2010.

5. Y. Chu and D. R. Corey. Rna sequencing: Platform selection, experimental design, and data interpretation. *Nucleic Acid Therapeutics*, 22(4):271–274, 2012. PMID: 22830413.

6. A. Conesa, P. Madrigal, S. Tarazona, D. Gomez-Cabrero, A. Cervera, A. McPherson, M. W. Szcześniak, D. J. Gaffney, L. L. Elo, X. Zhang, et al. A survey of best practices for rna-seq data analysis. *Genome biology*, 17(1):13, 2016.

7. R. Díaz-Uriarte and S. A. De Andres. Gene selection and classification of microarray data using random forest. *BMC bioinformatics*, 7(1):3, 2006.

8. K. Dong, H. Zhao, T. Tong, and X. Wan. Nblda: negative binomial linear discriminant analysis for rna-seq data. *BMC Bioinformatics*, 17(1):369, Sep 2016.

9. S. Dudoit, J. Fridlyand, and T. P. Speed. Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American statistical association*, 97(457):77–87, 2002.

10. J. Fan and J. Lv. Sure independence screening for ultrahigh dimensional feature space. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(5):849–911, 2008.

11. M. Fromer, P. Roussos, S. K. Sieberts, J. S. Johnson, D. H. Kavanagh, T. M. Perumal, D. M. Ruderfer, E. C. Oh, A. Topol, H. R. Shah, et al. Gene expression elucidates functional impact of polygenic risk for schizophrenia. *Nature neuroscience*, 19(11):1442, 2016.

12. M. D. Li, J. Cao, S. Wang, J. Wang, S. Sarkar, M. Vigorito, J. Z. Ma, and S. L. Chang. Transcriptome sequencing of gene expression in the brain of the hiv-1 transgenic rat. *PLoS One*, 8(3):e59582, 2013.

13. D. J. Lorenz, R. S. Gill, R. Mitra, and S. Datta. Using rna-seq data to detect differentially expressed genes. In *Statistical analysis of next generation sequencing data*, pages 25–49. Springer, 2014.

14. J. C. Marioni, C. E. Mason, S. M. Mane, M. Stephens, and Y. Gilad. Rna-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome research*, 18(9):1509–1517, 2008.

15. D. J. McCarthy, Y. Chen, and G. K. Smyth. Differential expression analysis of multifactor rna-seq experiments with respect to biological variation. *Nucleic acids research*, 40(10):4288–4297, 2012.

16. M. J. Peters, R. Joehanes, L. C. Pilling, C. Schurmann, K. N. Conneely, J. Powell, E. Reinmaa, G. L. Sutphin, A. Zhernakova, K. Schramm, et al. The transcriptional landscape of age in human peripheral blood. *Nature communications*, 6:8570, 2015.

17. M. D. Robinson, D. J. McCarthy, and G. K. Smyth. edger: a bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, 26(1):139–140, 2010.

18. M. D. Robinson and A. Oshlack. A scaling normalization method for differential expression analysis of rna-seq data. *Genome biology*, 11(3):R25, 2010.

19. M. Stone. Cross-validatory choice and assessment of statistical predictions. *Roy. Stat. Soc.*, 36:111–147, 1974.

20. Z. Wang, M. Gerstein, and M. Snyder. Rna-seq: a revolutionary tool for transcriptomics. *Nature reviews genetics*, 10(1):57, 2009.

21. D. Witten, R. Tibshirani, S. G. Gu, A. Fire, and W.-O. Lui. Ultra-high throughput sequencing-based small rna discovery and discrete statistical biomarker analysis in a collection of cervical tumours and matched controls. *BMC Biology*, 8(1):58, May 2010.

22. D. M. Witten. Classification and clustering of sequencing data using a poisson model. *Ann. Appl. Stat.*, 5(4):2493–2518, 12 2011.

23. G. Zararsız, D. Goksuluk, S. Korkmaz, V. Eldem, G. E. Zararsiz, I. P. Duru, and A. Ozturk. A comprehensive simulation study on classification of rna-seq data. *PloS one*, 12(8):e0182507, 2017.

24. S. Zhao, W.-P. Fung-Leung, A. Bittner, K. Ngo, and X. Liu. Comparison of rna-seq and microarray in transcriptome profiling of activated t cells. *PloS one*, 9(1):e78644, 2014.