Journal of Multivariate Analysis 171 (2019) 250–269

Contents lists available at ScienceDirect

Journal of Multivariate Analysis

journal homepage: www.elsevier.com/locate/jmva

Graph-based sparse linear discriminant analysis for high-dimensional classification

Jianyu Liu [a], Guan Yu [b], Yufeng Liu [a,c,*]

[a] Department of Statistics and Operations Research, University of North Carolina, Chapel Hill, NC 27599, USA
[b] Department of Biostatistics, University at Buffalo, Buffalo, NY 14214, USA
[c] Department of Genetics, Department of Biostatistics, and Carolina Center for Genome Sciences, University of North Carolina, Chapel Hill, NC 27599, USA

ARTICLE INFO

Article history:
Received 21 September 2017
Available online 17 December 2018

Keywords:
Feature structure
Gaussian graphical models
Regularization
Undirected graph

AMS subject classifications:
62H30

ABSTRACT

Linear discriminant analysis (LDA) is a well-known classification technique that enjoyed great success in practical applications. Despite its effectiveness for traditional low-dimensional problems, extensions of LDA are necessary in order to classify high-dimensional data. Many variants of LDA have been proposed in the literature. However, most of these methods do not fully incorporate the structure information among predictors when such information is available. In this paper, we introduce a new high-dimensional LDA technique, namely graph-based sparse LDA (GSLDA), that utilizes the graph structure among the features. In particular, we use the regularized regression formulation for penalized LDA techniques, and propose to impose a structure-based sparse penalty on the discriminant vector $\boldsymbol{\beta}$. The graph structure can be either given or estimated from the training data. Moreover, we explore the relationship between the within-class feature structure and the overall feature structure. Based on this relationship, we further propose a variant of our proposed GSLDA to utilize effectively unlabeled data, which can be abundant in the semi-supervised learning setting. With the new regularization, we can obtain a sparse estimate of $\boldsymbol{\beta}$ and more accurate and interpretable classifiers than many existing methods. Both the selection consistency of $\boldsymbol{\beta}$ estimation and the convergence rate of the classifier are established, and the resulting classifier has an asymptotic Bayes error rate. Finally, we demonstrate the competitive performance of the proposed GSLDA on both simulated and real data studies.

1. Introduction

Classification problems are commonly seen in practice. There are many existing classification techniques in the literature; see [2,17] for a comprehensive review. Among various existing methods, linear discriminant analysis (LDA) has a long history and remains an important tool in the standard classification toolbox. LDA can be viewed as a rule for a classification problem of two Gaussian populations with a common covariance matrix. Despite its seemingly strong assumptions, LDA often works well in practice, especially for low-dimensional problems [15]. It mimics Bayes' rule and has a simple closed form which only involves the within-class sample covariance matrix and group averages. Given these estimates, the original formulation for the discriminant vector of LDA is computed as the product of the inverse within-class sample covariance matrix and the

* Correspondence to: Department of Statistics and Operations Research, Department of Genetics, Department of Biostatistics, and Carolina Center for Genome Sciences, University of North Carolina, Chapel Hill, NC 27599, USA.
E-mail address: yfliu@email.unc.edu (Y. Liu).

https://doi.org/10.1016/j.jmva.2018.12.007
0047-259X/

mean difference vector. Thus, standard LDA can be computed and implemented easily in the traditional low-dimensional setting. LDA also has interpretations beyond the Gaussian model. In particular, the same formulation can be obtained from Fisher's discriminant analysis problem [13], the optimal scoring problem [16], and linear regression [17].

Despite the usefulness of LDA, it needs to be adapted when the dimension of features is high. For example, the form of standard LDA is only valid when the sample covariance matrix is invertible. Moreover, as the dimension grows, the errors in the sample covariance and group means accumulate and consequently LDA can become increasingly unstable [11,35]. To address this problem, a number of LDA extensions have been proposed for high-dimensional scenarios.

The existing high-dimensional LDA methods in the literature can be roughly divided into two categories, plug-in approaches and direct approaches. A plug-in approach tackles high-dimensional problems by using regularized estimates for the within-class covariance matrix and group means. For example, the naive Bayes method, or the independence rule, treats the covariance matrix as diagonal. Bickel and Levina [1] showed that it outperforms LDA with Moore–Penrose pseudoinverse covariance matrix when the dimension grows faster than the sample size. To further reduce the instability of LDA, Tibshirani et al. [36] additionally used shrunken estimates of group means. Fan and Fan [11] showed that, even under the independence feature assumption, naive Bayes can be as bad as random guessing due to error accumulation in group means. They resolved this issue by reducing the dimension via feature screening. In contrast to these independence rules, Shao et al. [35] assumed sparsity of the covariance matrix and the mean difference vector, and used thresholded estimates to construct a sparse LDA classifier. It was shown to be asymptotically optimal under certain conditions. All of these methods adopt the original formulation of LDA by calculating some improved estimates of the covariance matrix and group means. Thus, some strong assumptions on the covariance matrix and the group means need to be imposed for the resulting LDA rule.

In contrast to the plug-in methods, direct approaches aim at estimating the discriminant vector $\boldsymbol{\beta}$ directly. Since LDA can also be obtained from some risk minimization problems, it can be extended to high-dimensional scenarios via these formulations with regularization on $\boldsymbol{\beta}$. For example, Wu et al. [40] considered Fisher's discriminant analysis and proposed an ℓ_1-penalized version for dimension reduction. The corresponding problem has a piece-wise linear solution path which can be computed efficiently. Witten and Tibshirani [39] also used Fisher's discriminant analysis formulation for a general K-class problem with a general regularization. Clemmensen et al. [10] proposed the optimal scoring formulation with the ℓ_1 penalty. Following the idea of minimizing the misclassification rates, Fan et al. [12] proposed a method closely related to the method by Wu et al. [40] and directly computed the misclassification rate of the classifier. Mai et al. [26] took advantage of the regression formulation and estimated the discriminant vector of LDA by solving a Lasso-type problem, which was shown to have the same solution path as the method of Wu et al. [40] and the method of Clemmensen et al. [10] when $K = 2$; see [25]. Using a different idea of direct estimation, Cai and Liu [6] formed a linear programming problem to estimate $\boldsymbol{\beta}$ and showed that the error rate of the estimated classifier is close to the Bayes rule under certain conditions. Compared to plug-in approaches, these methods estimate LDA directly and the assumptions can be less stringent since only the sparsity of the discriminant vector of LDA is assumed [6].

Both plug-in and direct methods can work well for certain practical problems. However, these methods do not utilize the feature structure information when available. In practice, features are often correlated with some structure. Such structure can usually be represented by an undirected graph $\mathcal{G}$. Connected features may work together and thus be effective or not effective simultaneously for classification. For instance, in the diagnosis of a disease using genetic information, genes are naturally grouped by their functions or gene pathways. Relevant genes tend to contribute or not contribute to the disease together. Moreover, when the population in consideration is Gaussian, the conditional independence graph, or Gaussian graphical model, often represents a natural structure. By considering such structure information, we are likely to be able to construct a better classifier. For regression problems, there are some methods that utilize the graph structure in the literature; see, e.g., [3,18,33,53]. For example, Li and Li [19] proposed a penalty on the coefficient difference of each pair of connected features. Yang et al. [42] used pairwise ℓ_∞ penalties on relevant features to encourage simultaneous inclusion and exclusion. Based on the decomposition of the regression coefficient vector, Yu and Liu [44] proposed a node-wise penalty. In particular, the regularization term is the summation of penalties over all nodes rather than all edges. Compared to pairwise penalties, the node-wise penalty is better motivated and computationally efficient. More recently, Zhao and Shojaie [50] proposed new inference methods for such graph-constrained estimation.

Despite great progress for regression problems, much less research has been done for classification problems. Structured penalties such as group Lasso and fused Lasso have been employed in classification methods [27,39], but they are not applicable to a general sparse graph structure among predictors. Zhang et al. [49] considered logistic regression with a combination of ℓ_1 penalty and pairwise ℓ_2 difference penalty. Min et al. [29] generalized the regularization and provided a unified algorithm. However, both methods may also suffer from too much computational burden in high dimensions. Very recently, Wu et al. [41] proposed an unsupervised graph-based variable screening method for general problems.

In this paper, we propose a new method, called graph-based sparse LDA (GSLDA), that exploits the graphical structure of features. GSLDA estimates LDA in high dimensions directly by solving a convex optimization problem. Similar to the sparse regression method in [44], we incorporate the graph structure through a node-wise penalty. In the presence of an underlying feature structure, the new method outperforms existing high-dimensional LDA methods by utilizing the structure directly. As a key component, the graphical structure can be either given or estimated from the training data. In addition, we investigate the relationship between the within-class inverse covariance matrix and overall inverse covariance matrix. Based on these findings, we propose a variant of GSLDA that can utilize unlabeled data, which are often much more accessible than labeled data. We name this variant as the semi-supervised GSLDA. Selection consistency is shown for the estimated discriminant

vector. Moreover, we show that the misclassification rate of our classifier converges to the Bayes error rate at a fast rate under certain conditions. Numerical studies are used to demonstrate the performance of this method. In particular, the semi-supervised GSLDA enjoys higher classification accuracy than the original GSLDA method in most cases. This reveals the potential advantages of using unlabeled data in classification problems.

The rest of the paper is organized as follows. In Section 2, we review some existing high-dimensional LDA methods, and introduce our motivations and formulations of our proposed methods. Section 3 focuses on graph estimation and the implementation of GSLDA. In particular, graph estimation methods are discussed for both GSLDA and its variant. In Section 4, theoretical justification is provided for our method. Sections 5 and 6 demonstrate the performance of GSLDA by simulated examples and real data studies respectively. We conclude this paper with some discussion in Section 7. Proofs of the theoretical results are provided in the Appendix.

2. Methodology

In this section, we first review LDA and construct a relationship between $\boldsymbol{\beta}$ and the graph structure of features in Section 2.1, based on which GSLDA is proposed. We also explain how to estimate the graph structure when it is not directly available and discuss the connections of our methods with several existing classification methods. In Section 2.2, we investigate the overall graph structure of the features and consider a variant of GSLDA which can efficiently utilize unlabeled data.

2.1. Motivation and formulation of GSLDA

We first discuss the problem setting and introduce some notations. Given the training dataset $\{(\mathbf{x}_1, g_1), \ldots, (\mathbf{x}_n, g_n)\}$ where for each $i \in \{1, \ldots, n\}$, $\mathbf{x}_i \in \mathbb{R}^p$ is the feature vector and $g_i \in \{1, 2\}$ is the class label. A linear classifier $g_{\beta_0,\boldsymbol{\beta}}$ is defined as follows. For any $\mathbf{x} \in \mathbb{R}^p$, $g_{\beta_0,\boldsymbol{\beta}}(\mathbf{x}) = 1$ if $\beta_0 + \mathbf{x}^\top\boldsymbol{\beta} > 0$ and 2 otherwise. In particular, we consider the standard setting of the two-class LDA. That is, the binary label G takes 1 with probability π_1 and 2 with probability $\pi_2 = 1 - \pi_1$ and the feature vector $\mathbf{X}$ has a conditional Gaussian distribution, i.e., $\mathbf{X}|(G = k) \sim \mathcal{N}(\boldsymbol{\mu}^{(k)}, \boldsymbol{\Sigma})$ for $k \in \{1, 2\}$. Under this setting, the Bayes classifier $g_{\beta_0^*,\boldsymbol{\beta}^*}$ is specified by

$$\boldsymbol{\beta}^* = \boldsymbol{\Sigma}^{-1}\boldsymbol{\delta} \quad \text{and} \quad \beta_0^* = -(\boldsymbol{\mu}^{(1)} + \boldsymbol{\mu}^{(2)})^\top\boldsymbol{\beta}^*/2 + \ln(\pi_1/\pi_2), \tag{1}$$

where $\boldsymbol{\delta} = \boldsymbol{\mu}^{(1)} - \boldsymbol{\mu}^{(2)}$. By replacing $\boldsymbol{\Sigma}$ and $\boldsymbol{\delta}$ in (1) with their sample estimates, we have the LDA classifier with $\hat{\boldsymbol{\beta}} = \hat{\boldsymbol{\Sigma}}^{-1}\hat{\boldsymbol{\delta}}$. Typically, we take $\hat{\boldsymbol{\Sigma}} = (n_1\mathbf{S}^{(1)} + n_2\mathbf{S}^{(2)})/(n-2)$ and $\hat{\boldsymbol{\delta}} = \bar{\mathbf{x}}^{(1)} - \bar{\mathbf{x}}^{(2)}$, where n_k, $\bar{\mathbf{x}}^{(k)}$ and $\mathbf{S}^{(k)}$ denote respectively the sample size, mean and covariance matrix for group k. Note that this formulation is valid only when $n > p$. In high-dimensional problems or when $n \le p$, there are various extensions of LDA that either use the formulation with shrunken estimates of $\boldsymbol{\Sigma}$ and $\boldsymbol{\delta}$ or find a direct estimation of $\boldsymbol{\beta}$; see [7,12,26,35,36]. Here we focus on the direct estimation approach.

Inspired by the regression formulation of LDA [17], Mai et al. [26] proposed the direct sparse discriminant analysis (DSDA) method to estimate $\boldsymbol{\beta}$ by solving the Lasso problem

$$\hat{\boldsymbol{\beta}} = \operatorname*{argmin}_{\boldsymbol{\beta}} \frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - \mathbf{x}_i^\top\boldsymbol{\beta})^2 + \lambda\|\boldsymbol{\beta}\|_1,$$

where $y_i = n/n_1$ if $g_i = 1$ and $-n/n_2$ if $g_i = 2$. It was shown that DSDA gives the same solution path as the method in [25,40]. Compared to plug-in approaches, the DSDA estimates $\boldsymbol{\beta}$ directly in high dimensions and the assumptions are less stringent. However, it is unclear how we can utilize any structure information among features with the method or other high-dimensional LDA methods.

Assume that there is some structure among the features. In particular, we consider the case where the structure can be represented by a graph, denoted as $\mathcal{G}$. There are methods that effectively use the graph structure in regression problems. For example, Li and Li [19] used the penalty

$$\sum_{(j,\ell)\in\mathcal{G}}\left(\beta_j/\sqrt{d_j} - \beta_\ell/\sqrt{d_\ell}\right)^2,$$

where d_j denotes the neighborhood size of feature j, to encourage close coefficients for connected features. Yang et al. [42] employed pairwise ℓ_∞ penalty for connected features, i.e., $\sum_{(j,\ell)\in\mathcal{G}}\max\{|\beta_j|, |\beta_\ell|\}$, so their coefficients can be estimated zero or nonzero simultaneously. Recently, Yu and Liu [44] proposed a node-wise penalty

$$P_{\mathcal{G},\boldsymbol{\tau}}(\boldsymbol{\beta}) = \min_{\sum_{j=1}^p \mathbf{v}^{(j)}=\boldsymbol{\beta},\ \mathrm{supp}(\mathbf{v}^{(j)})\subseteq\mathcal{N}(j)} \sum_{j=1}^{p}\tau_j\|\mathbf{v}^{(j)}\|_2$$

based on the decomposition of regression coefficient vector $\boldsymbol{\beta} = \mathrm{var}(X)^{-1}\mathrm{cov}(X, Y)$. In contrast to these developments for regression problems, little work has been done for classification problems.

We propose our method formulation based on a decomposition of $\boldsymbol{\beta}^*$, the discriminant vector of Bayes' rule. Denote $\boldsymbol{\Omega} = \boldsymbol{\Sigma}^{-1}$ the within-class precision matrix and $\boldsymbol{\delta} = \boldsymbol{\mu}^{(1)} - \boldsymbol{\mu}^{(2)}$ the group mean difference. We can decompose the discriminant vector $\boldsymbol{\beta}^*$ in (1) as

$$\boldsymbol{\beta}^* = \boldsymbol{\Omega}\boldsymbol{\delta} = \sum_{j=1}^{p} \delta_j \boldsymbol{\omega}_j, \tag{2}$$

where $\boldsymbol{\omega}_j$ is the jth column of $\boldsymbol{\Omega}$. Recall that the support of $\boldsymbol{\Omega}$ in fact forms a conditional correlation graph of features X. In this way, the optimal discriminant vector is linked to the Gaussian graph structure of the features. We use a toy example for demonstration. In a 3-dimensional LDA setting, assume $\omega_{23} = \omega_{32} = 0$, then $\boldsymbol{\beta}^* = \boldsymbol{\Omega}\boldsymbol{\delta} = (\delta_1\omega_{11} + \delta_2\omega_{21} + \delta_3\omega_{31}, \delta_1\omega_{12} + \delta_2\omega_{22}, \delta_1\omega_{13} + \delta_3\omega_{33})^\top$. See Fig. A.1 in the Appendix for a graphical demonstration of the decomposition.

Denote the graph corresponding to $\boldsymbol{\Omega}$ as $\mathcal{G}$, and the neighborhood of feature $j \in \{1, \ldots, p\}$ as $\mathcal{N}^{(j)}$. Replacing $\delta_j\boldsymbol{\omega}_j$ by $\mathbf{v}^{(j)}$, then $\boldsymbol{\beta}^* = \mathbf{v}^{(1)} + \cdots + \mathbf{v}^{(p)}$, where $\mathbf{v}^{(j)}$ is either $\mathbf{0}$ (when $\delta_j = 0$) or with a support $\mathrm{supp}(\mathbf{v}^{(j)}) = \mathcal{N}^{(j)}$ when $\delta_j \neq 0$. Instead of estimating $\boldsymbol{\beta}^*$ itself, we can estimate $\mathbf{v}^{(j)}$'s. Moreover, the decomposition (2) motivates a natural regularization on $\{\mathbf{v}^{(1)}, \ldots, \mathbf{v}^{(p)}\}$, viz.

$$\sum_{j=1}^{p} \tau_j \|\mathbf{v}^{(j)}\|_2,$$

in which $\mathrm{supp}(\mathbf{v}^{(j)}) \subseteq \mathcal{N}^{(j)} = \mathrm{supp}(\boldsymbol{\omega}_j)$ and the τ_js are positive weights. Note that the group ℓ_2 penalty on $\mathbf{v}^{(j)}$ encourages a group sparsity effect, i.e., $\mathbf{v}^{(j)}$ is estimated as $\mathbf{0}$ or a sparse vector with support $\mathcal{N}^{(j)}$, which matches the decomposition (2). In the formulations, the τ_js are weights for the group regularization. In particular, the larger τ_j is, the more likely $\mathbf{v}^{(j)}$ is estimated as $\mathbf{0}$. Similar to the group Lasso [45], we can take

$$\tau_j = \sqrt{|\mathcal{N}^{(j)}|}/|\hat{\delta}_j|,$$

where $\hat{\delta}_j = \bar{x}_j^{(1)} - \bar{x}_j^{(2)}$.

We need to apply this regularization to a risk minimization framework of LDA to formulate our method. The regression formulation is an appropriate one due to its simplicity and convenience for theoretical analysis. By combining the formulation with the group regularization, we can estimate $\{\mathbf{v}_1, \ldots, \mathbf{v}_p\}$ by

$$(\hat{\beta}_0, \hat{\mathbf{v}}_1, \ldots, \hat{\mathbf{v}}_p) = \underset{\beta_0,\, \mathbf{v}_1, \ldots, \mathbf{v}_p}{\mathrm{argmin}} \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \beta_0 - \mathbf{x}_i^\top \sum_{j=1}^{p} \mathbf{v}^{(j)}\right)^2 + \lambda \sum_{j=1}^{p} \tau_j \|\mathbf{v}^{(j)}\|_2, \tag{3}$$

where $\mathrm{supp}(\mathbf{v}^{(j)}) \subseteq \mathcal{N}^{(j)}$ for all $j \in \{1, \ldots, p\}$. Then $\boldsymbol{\beta}$ is estimated as $\hat{\mathbf{v}}_1 + \cdots + \hat{\mathbf{v}}_p$. Furthermore, from the perspective of $\boldsymbol{\beta}$ estimation, the formulation is equivalent to

$$(\hat{\beta}_0, \hat{\boldsymbol{\beta}}) = \underset{\beta_0, \boldsymbol{\beta}}{\mathrm{argmin}} \frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - \mathbf{x}_i^\top \boldsymbol{\beta})^2 + \lambda \|\boldsymbol{\beta}\|_{\mathcal{G},\boldsymbol{\tau}}, \tag{4}$$

where

$$\|\boldsymbol{\beta}\|_{\mathcal{G},\boldsymbol{\tau}} = \min_{\sum_{j=1}^{p} \mathbf{v}^{(j)} = \boldsymbol{\beta},\ \mathrm{supp}(\mathbf{v}^{(j)}) \subseteq \mathcal{N}^{(j)}} \sum_{j=1}^{p} \tau_j \|\mathbf{v}^{(j)}\|_2 \tag{5}$$

can be viewed as a structured regularization on $\boldsymbol{\beta}$; see [31]. Since the regularization is specified by the graph $\mathcal{G}$, we call the method graph-based sparse LDA (GSLDA). Although we use the same squared loss function as in [26], our method focuses on utilizing the graph structure of features in $\boldsymbol{\beta}^*$ estimation. We use the estimator $\hat{\boldsymbol{\beta}}$ from (4) for the discriminant vector $\boldsymbol{\beta}$. With respect to β_0, the estimator from (4) may not be a good choice for the classification problem due to the regression formulation. To solve this problem, we adopt a similar approach by [26] and estimate it by

$$\hat{\beta}_0 = -(\bar{\mathbf{x}}^{(1)} + \bar{\mathbf{x}}^{(2)})^\top \hat{\boldsymbol{\beta}}/2 + \ln(n_1/n_2)\hat{\boldsymbol{\beta}}^\top \hat{\boldsymbol{\Sigma}}\hat{\boldsymbol{\beta}}/(\hat{\boldsymbol{\delta}}^\top \hat{\boldsymbol{\beta}}).$$

While the GSLDA method is motivated from the discriminant vector decomposition (2), the decomposition of $\boldsymbol{\beta}^*$ is not restricted to this form only. Therefore, the graph structure $\mathcal{G}$ used in our method is not restricted to the conditional independence graph. We will present another decomposition of $\boldsymbol{\beta}^*$ in Section 2.2. In fact, any graph structure of features satisfying our assumptions in Section 4.1 can be possibly used. When the structure information is available, e.g., the gene pathways in genetic studies, we can construct a graph $\mathcal{G}$ using the gene pathway information. If the graph is not available, we can estimate it based on the training data. There are many methods for estimation of Gaussian graphical models, including the neighborhood selection [28], the graphical Lasso [14,46], and the CLIME [7]. We will discuss them further in Section 3. In summary, GSLDA can be implemented in two steps: (i) graph construction and (ii) direct estimation of $\boldsymbol{\beta}$ via solving formulation (4).

The formulation (4) is closely related to the regression method proposed in [44]. However, both the problem setting and the motivation of our paper are different. In our problem, the response y is a binary variable and the features are from a mixed population. Although our formulation also uses the squared loss as in regression, the "error" $y_i - \mathbf{x}_i^\top \boldsymbol{\beta}$ has a very different interpretation and distribution. In particular, the distribution of $y_i - \mathbf{x}_i^\top \boldsymbol{\beta}$ depends on $\mathbf{x}_i$. These issues bring unique challenges for the theoretical analysis of GSLDA. Although there are some classification methods that also utilize predictor structure, such as logistic regression with group Lasso penalty [27] and LDA with fused Lasso penalty [39], these methods do not utilize a general graph structure.

Depending on the feature structure, there are special cases in which GSLDA is closely connected with existing sparse LDA methods. For example, if we use an empty graph $\mathcal{G}$ with no edge at all, the regularization (5) simplifies to $\tau_1|\beta_1| + \cdots + \tau_p|\beta_p|$. Then, formulation (4) becomes an adaptive Lasso type problem, viz.

$$\underset{\beta_0,\,\boldsymbol{\beta}}{\operatorname{argmin}} \frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - \mathbf{x}_i^\top \boldsymbol{\beta})^2 + \lambda \sum_{j=1}^{p} \tau_j |\beta_j|.$$

When all penalty weights τ_j take value 1, the GSLDA is equivalent to the DSDA method in [26]. When the graph $\mathcal{G}$ consists of K disjoint complete subgraphs, denoted as $\mathcal{G}^{(1)}, \ldots, \mathcal{G}^{(K)}$, then the regularization (5) simplifies to $\tau^{(1)}\|\boldsymbol{\beta}_{G^{(1)}}\|_2 + \cdots + \tau^{(K)}\|\boldsymbol{\beta}_{G^{(K)}}\|_2$ where $\tau^{(k)} = \min_{j \in G^{(k)}} \tau_j$ and $G^{(k)}$ is the index set of predictors involved in the subgraph $\mathcal{G}^{(k)}$. In this case, GSLDA becomes a variant of DSDA with the group Lasso penalty, i.e.,

$$\underset{\beta_0,\,\boldsymbol{\beta}}{\operatorname{argmin}} \frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - \mathbf{x}_i^\top \boldsymbol{\beta})^2 + \lambda \sum_{k=1}^{K} \tau^{(k)} \|\boldsymbol{\beta}_{G^{(k)}}\|_2.$$

For a general graph $\mathcal{G}$, our method is different from the existing ones.

Remark 1. While we are mainly concerned with binary classification in this paper, there are many scenarios with more than two classes [21,47,48]. Our GSLDA method can also be extended to the multi-class case. For example, consider a formulation of K-class sparse LDA proposed in [24], viz.

$$(\hat{\boldsymbol{\theta}}_2, \ldots, \hat{\boldsymbol{\theta}}_K) = \underset{\boldsymbol{\theta}_2,\ldots,\boldsymbol{\theta}_K}{\operatorname{argmin}} \sum_{k=2}^{K} \{\boldsymbol{\theta}_k^\top \hat{\boldsymbol{\Sigma}} \boldsymbol{\theta}_k/2 - (\bar{\mathbf{x}}^{(k)} - \bar{\mathbf{x}}^{(1)})^\top \boldsymbol{\theta}_k\} + \lambda \sum_{j=1}^{p} \|\boldsymbol{\theta}_{\cdot j}\|_2,$$

where $\boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_K$ are discriminant vectors and $\boldsymbol{\theta}_{\cdot j} = (\theta_{2j}, \ldots, \theta_{Kj})^\top$ for $j \in \{1, \ldots, p\}$. The resulting discriminant rule is $\hat{g} = \operatorname{argmax}_k \{\hat{\boldsymbol{\theta}}_k^\top (\mathbf{x} - \bar{\mathbf{x}}^{(k)}/2) + \ln \hat{\pi}_k\}$ where $\hat{\boldsymbol{\theta}}_1 = \mathbf{0}$ and $\hat{\pi}_k$ is the proportion of class k in the sample. We can take advantage of a similar formulation with the graph-based regularization $\lambda \|\boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_K\|_{\mathcal{G},\boldsymbol{\tau},\text{grouped}}$, where

$$\|\boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_K\|_{\mathcal{G},\boldsymbol{\tau},\text{grouped}} = \underset{\sum_{j=1}^{p} \mathbf{v}_k^{(j)} = \boldsymbol{\theta}_k, \operatorname{supp}(\mathbf{v}_k^{(j)}) \subseteq \mathcal{N}^{(j)}}{\operatorname{argmin}} \sum_{j=1}^{p} \tau_j \|(\mathbf{v}_2^{(j)\top}, \ldots, \mathbf{v}_K^{(j)\top})^\top\|_2.$$

This formulation can be solved in a way similar to the binary GSLDA. Nevertheless, we do not pursue this direction in the paper so we can focus on core ideas of the GSLDA.

2.2. Semi-supervised GSLDA

With recent advances in graphical estimation [7,28,46], we can estimate $\mathcal{G}$ for the GSLDA based on the training data when the graph structure is unknown. However, as the dimension p increases, we expect the selection error to accumulate. When the dimension is much larger than the sample size, the graph estimate of GSLDA can be almost random. We use a toy example in Fig. 1 to illustrate this phenomenon. In the setting of standard LDA, we set weights $\pi_1 = \pi_2 = 0.5$, and group means $\boldsymbol{\mu}^{(1)} = (0.5, \ldots, 0.5, 0, \ldots, 0)^\top$ and $\boldsymbol{\mu}^{(2)} = (-0.5, \ldots, -0.5, 0, \ldots, 0)^\top$, which only differ in the first 10 features. To specify the graph structure, $\boldsymbol{\Omega}$ is generated from an AR(5) model, i.e., $\Omega_{jj} = c$, $\Omega_{j\ell} = -0.5$ if $1 \le |j - \ell| \le 5$ and 0 otherwise, where $c > 0$ is a scalar such that the eigenvalues of $\boldsymbol{\Omega}$ are between 0 and 1. We standardize $\boldsymbol{\Omega}$ so that $\operatorname{diag}(\boldsymbol{\Omega}) = \mathbf{1}$ and define in-class covariance matrix $\boldsymbol{\Sigma} = \boldsymbol{\Omega}^{-1}$. Let the sample size n be 50 and p vary from 10 to 200. We estimate the graph by SR-SLasso [22] with extended BIC for tuning. For each setting, we repeat the procedure 100 times and evaluate the accuracy of graph estimation by false positive rate (FPR) and false negative rate (FNR). Fig. 1 summarizes the performance of graph estimation for varying dimensions.

As shown in Fig. 1, the graph estimation using only labeled data deteriorates quickly as the dimension increases. Note that the structured penalty in (5) encourages the coefficients of all features in a neighborhood to be nonzero together as long as some of them is useful for classification. Inaccurate graph estimation can reduce the accuracy and the interpretability of GSLDA.

Compared to labeled data, unlabeled data can be more accessible in many applications. For example, in the handwritten digit recognition problem discussed in Section 6.2, we can easily obtain a large number of images of different digits. However, it can be expensive to label these images by corresponding digits. As a result, many semi-supervised methods try to utilize

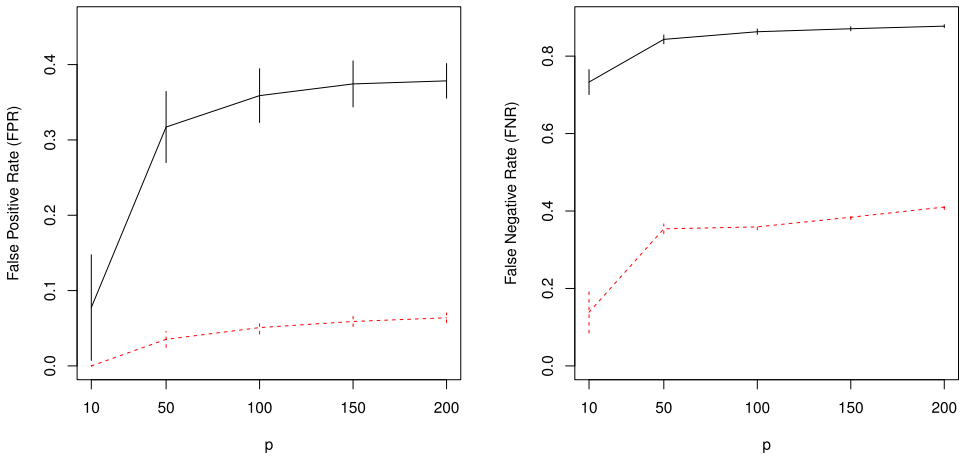

Fig. 1. Performance evaluation of graph estimation for varying dimensions. The black solid lines are for graph estimation based on a labeled dataset of size 50; the red dashed lines are for graph estimation based on an unlabeled dataset of size 1000; vertical segments indicate the standard deviations of FPR or FNR of 100 repetitions.

the unlabeled data to improve the classification accuracy [5,32]. In this paper, we focus on using unlabeled data for the graph construction when available. The following proposition studies the relationship between the within-class inverse covariance matrix and the overall one.

Proposition 1. *Assume X comes from a mixture of two populations with a common covariance matrix $\mathbf{\Sigma}$. The weight and the expectation of population $k \in \{1, 2\}$ is π_k and $\boldsymbol{\mu}^{(k)}$. Denote the mean difference of the two populations $\boldsymbol{\mu}^{(1)} - \boldsymbol{\mu}^{(2)}$ as $\boldsymbol{\delta}$. We denote $\tilde{\mathbf{\Sigma}} = \mathrm{var}(X)$ the overall covariance matrix of the population mixture and $\tilde{\mathbf{\Omega}} = \tilde{\mathbf{\Sigma}}^{-1}$ the overall precision matrix. Then $\tilde{\mathbf{\Sigma}} = \mathbf{\Sigma} + \pi_1\pi_2\boldsymbol{\delta}\boldsymbol{\delta}^\top$ and $\tilde{\mathbf{\Omega}} = \mathbf{\Omega} - c\boldsymbol{\beta}^*\boldsymbol{\beta}^{*\top}$, where $\boldsymbol{\beta}^* = \mathbf{\Omega}\boldsymbol{\delta}$ and $c = 1/\{(\pi_1\pi_2)^{-1} + \boldsymbol{\delta}^\top\mathbf{\Omega}\boldsymbol{\delta}\}$.*

As a remark, we do not require any specific distribution for the populations in Proposition 1, while $\boldsymbol{\beta}^*$ is the optimal discriminant vector if both classes are Gaussian populations. The overall precision matrix $\tilde{\mathbf{\Omega}}$ is sparse if both $\mathbf{\Omega}$ and $\boldsymbol{\beta}^*$ are sparse, and its support forms the conditional correlation graph of the mixed population. Moreover, we have $\tilde{\mathbf{\Omega}}\boldsymbol{\delta} = (1 - c\boldsymbol{\beta}^{*\top}\boldsymbol{\delta})\boldsymbol{\beta}^* \propto \boldsymbol{\beta}^*$. In our problem, a decomposition of the optimal discriminant vector analogous to (2) using $\tilde{\mathbf{\Omega}}$ can be written as

$$\boldsymbol{\beta}^* = \xi \sum_{j=1}^{p} \delta_j \tilde{\mathbf{w}}_j,$$

where ξ is a positive scalar and $\tilde{\mathbf{w}}_j$ is the jth column of $\tilde{\mathbf{\Omega}}$. Therefore, the Bayes classifier can be connected to the graph structure of the mixed population through the new decomposition. Define the graph corresponding to the support of $\tilde{\mathbf{\Omega}}$ as $\tilde{\mathcal{G}}$. Following the same rationale of GSLDA, we can formulate another estimator of $\boldsymbol{\beta}$ based on the overall graph structure, viz.

$$(\hat{\beta}_0, \hat{\boldsymbol{\beta}}) = \underset{\beta_0, \boldsymbol{\beta}}{\mathrm{argmin}} \frac{1}{n} \sum_{i=1}^{n} (y_i - \beta_0 - \mathbf{x}_i^\top \boldsymbol{\beta})^2 + \lambda \|\boldsymbol{\beta}\|_{\tilde{\mathcal{G}}, \tilde{\boldsymbol{\tau}}}, \tag{6}$$

where $\|\boldsymbol{\beta}\|_{\tilde{\mathcal{G}}, \tilde{\boldsymbol{\tau}}}$ is defined in (5) and $\tilde{\boldsymbol{\tau}}$ adapts to $\tilde{\mathcal{G}}$ as in (4). The only difference between (6) and (4) is which graph structure we use. When unlabeled data are abundant, the estimated graph $\tilde{\mathcal{G}}$ can be more accurate and thus the new formulation may provide better classification. We name the formulation (6) as semi-supervised GSLDA. Similar to the original GSLDA, the semi-supervised variant also has two step: (i) graph estimation based on all available data and (ii) direct estimation of $\boldsymbol{\beta}$ by solving formulation (6).

Both versions of GSLDA need to estimate a graph when no prior graph structure is given. But there is a major difference: unlike $\mathcal{G}$ in (4), the graph $\tilde{\mathcal{G}}$ in (6) is not for a Gaussian population but a Gaussian mixture. As we will see in Section 3, likelihood-based estimation such as graphical Lasso would be too complicated to implement. Instead, we can still use neighborhood selection. In fact, in regressing the feature X_j on the other features X_{-j}, the coefficient vector corresponds to the conditional correlations between X_j and other features regardless of the distribution of the features, as stated by the following lemma.

Lemma 1. *For any random vector $X = (X_1, \ldots, X_p)^\top \sim F$, assume we have finite second-order moments and denote $\tilde{\boldsymbol{\mu}} = \mathrm{E}_F(X)$, $\tilde{\mathbf{\Sigma}} = \mathrm{E}_F\{(X - \tilde{\boldsymbol{\mu}})(X - \tilde{\boldsymbol{\mu}})^\top\}$ and $\tilde{\mathbf{\Omega}} = \tilde{\mathbf{\Sigma}}^{-1}$. Then for any $j, \ell \in \{1, \ldots, p\}$,*

(i) $\tilde{\omega}_{j\ell}$*, the* (j,ℓ)*th element of* $\tilde{\mathbf{\Omega}}$*, is 0 if and only if* X_j *and* X_ℓ *are conditionally uncorrelated, i.e.,* $\mathrm{cov}(X_j, X_\ell|X_{-\{j,\ell\}}) = 0$*, where* $X_{-\{j,\ell\}}$ *denotes all features other than* X_j *and* X_ℓ*;*

(ii) $\tilde{\omega}_{j\ell}$ *is* 0 *if and only if* $\gamma_\ell^{(j)} = 0$*, where* $\gamma_\ell^{(j)}$ *is the coefficient of* X_ℓ *in the regression of* X_j *on* X_{-j}*.*

This lemma is closely related to the results in [28]. According to Lemma 1, the graph based on the inverse covariance matrix always corresponds to the conditional correlation structure. As long as variable selection consistency of the regression is guaranteed, neighborhood selection methods are valid for graph estimation. Fig. 1 also shows the performance of graph estimation based on a large unlabeled dataset under the same settings. We can observe that the estimation still performs well when the dimension increases.

Remark 2. In practice, we generally use all available data, including both unlabeled and labeled data, in the first step of semi-supervised GSLDA. Note that even without unlabeled data, the method is still applicable. If we use neighborhood selection for graph estimation, then the error variance of the jth node-wise regression is $\mathrm{var}(X_j|X_{-j}) = 1/\tilde{\omega}_{jj} = 1/(\omega_{jj} - c\beta_j^{*2})$ by Proposition 1. In contrast, when using the labels as in the original GSLDA, the error variance is $\mathrm{var}(X_j|X_{-j}, G) = 1/\omega_{jj} < 1/(\omega_{jj} - c\beta_j^{*2})$. Therefore, the semi-supervised GSLDA has better graph estimation only when unlabeled data are abundant. When there are relatively little unlabeled data, the original GSLDA is more advantageous.

3. Graph estimation and method implementation

If the feature structure is given from prior knowledge, the graph can be directly constructed by assigning edges between related features. Otherwise, we need to estimate the graph based on training data. In particular, when unlabeled data are available, we can also use that to estimate the graph and implement semi-supervised GSLDA. In this section, we first discuss specific graph estimation methods for GSLDA. Then we introduce algorithms to solve formulation (4) as well as some strategies for efficient implementation.

3.1. Graph estimation

There have been extensive studies on graphical model estimation [7,9,14,28,38,46]. As we discussed in Section 2.2, the graph estimation based on labeled and unlabeled data are different to some extent. Next we discuss them separately. Given labeled data, the likelihood conditional on the labels becomes

$$(2\pi)^{-pn/2}|\mathbf{\Omega}|^{n/2}\exp\left\{-\frac{1}{2}\sum_{g_i=1}(\mathbf{x}_i-\boldsymbol{\mu}^{(1)})^\top\mathbf{\Omega}(\mathbf{x}_i-\boldsymbol{\mu}^{(1)})-\frac{1}{2}\sum_{g_i=2}(\mathbf{x}_i-\boldsymbol{\mu}^{(2)})^\top\mathbf{\Omega}(\mathbf{x}_i-\boldsymbol{\mu}^{(2)})\right\}.$$

Similar to the graphical Lasso, we can estimate $\mathbf{\Omega}$ by minimizing ℓ_1 penalized log-likelihood, i.e.,

$$\underset{\boldsymbol{\mu}^{(1)},\boldsymbol{\mu}^{(2)},\,\mathbf{\Omega}\in\mathbb{S}_{++}}{\operatorname{argmin}}\ \frac{n}{2}\ln|\mathbf{\Omega}|-\frac{1}{2}\sum_{g_i=1}(\mathbf{x}_i-\boldsymbol{\mu}^{(1)})^\top\mathbf{\Omega}(\mathbf{x}_i-\boldsymbol{\mu}^{(1)})-\frac{1}{2}\sum_{g_i=2}(\mathbf{x}_i-\boldsymbol{\mu}^{(2)})^\top\mathbf{\Omega}(\mathbf{x}_i-\boldsymbol{\mu}^{(2)})+\lambda\|\mathbf{\Omega}\|_1,$$

where $\mathbb{S}_{++}$ denotes the set of p-dimensional positive definite matrices and $\|\mathbf{\Omega}\|_1 = \sum_{j\neq\ell}|\omega_{j\ell}|$. It results in $\hat{\boldsymbol{\mu}}^{(1)} = \bar{\mathbf{x}}^{(1)}$, $\hat{\boldsymbol{\mu}}^{(2)} = \bar{\mathbf{x}}^{(2)}$, and

$$\hat{\mathbf{\Omega}} = \underset{\mathbf{\Omega}\in\mathbb{S}_{++}}{\operatorname{argmin}}\ \frac{n}{2}\ln|\mathbf{\Omega}|-\frac{1}{2}\sum_{i=1}^{n}(\mathbf{x}_i-\bar{\mathbf{x}}^{(g_i)})^\top\mathbf{\Omega}(\mathbf{x}_i-\bar{\mathbf{x}}^{(g_i)})+\lambda\|\mathbf{\Omega}\|_1. \tag{7}$$

This is equivalent to the graphical Lasso for the centered data $\mathbf{x}_1-\bar{\mathbf{x}}^{(g_1)},\ldots,\mathbf{x}_n-\bar{\mathbf{x}}^{(g_n)}$.

Instead of solving (7), we can also estimate the graph by neighborhood selection as proposed by [28]. This method solves p node-wise regularized regressions, viz.

$$\underset{\gamma_0^{(1j)},\gamma_0^{(2j)},\boldsymbol{\gamma}^{(j)}}{\operatorname{argmin}}\ \frac{1}{2n}\|\mathbf{X}_j^{(1)}-\gamma_0^{(1j)}-\mathbf{X}_{-j}^{(1)}\boldsymbol{\gamma}^{(j)}\|_2^2+\frac{1}{2n}\|\mathbf{X}_j^{(2)}-\gamma_0^{(2j)}-\mathbf{X}_{-j}^{(2)}\boldsymbol{\gamma}^{(j)}\|_2^2+\lambda\|\boldsymbol{\gamma}^{(j)}\|_1,$$

where $\mathbf{X}_j^{(k)}$ denotes the jth feature of sample from group k and $\mathbf{X}_{-j}^{(k)}$ represents the other features. One can verify that

$$\hat{\boldsymbol{\gamma}}^{(j)} = \underset{\boldsymbol{\gamma}^{(j)}}{\operatorname{argmin}}\ \frac{1}{2n}\|\dot{\mathbf{X}}_j-\gamma_0-\dot{\mathbf{X}}_{-j}\boldsymbol{\gamma}^{(j)}\|_2^2+\lambda\|\boldsymbol{\gamma}^{(j)}\|_1, \tag{8}$$

where $\dot{\mathbf{X}}$ denotes the data centered by subtracting corresponding group means. We can also use sequential Lasso [23] for computational efficiency. The graph $\mathcal{G}$ is constructed by connecting nodes j and ℓ if $\hat{\gamma}_\ell^{(j)} \neq 0$ and/or $\hat{\gamma}_j^{(\ell)} \neq 0$.

Both approaches for estimating $\mathcal{G}$ have been justified theoretically [28,46]. In this paper, we recommend to use neighborhood selection approaches for GSLDA. The main reason is that the former approaches, such as graphical Lasso, usually run through many iterations and can be slow for high-dimensional data ($p > 1000$). In contrast, neighborhood

selection approaches only require p penalized regressions. Moreover, our direct interest is not $\boldsymbol{\Omega}$ but the graph $\mathcal{G}$ on which neighborhood selection focuses. We use the extended BIC (EBIC) [8] to select λ in (8). As suggested in [8], we choose $1 - 1/(2\log_n p)$ as the EBIC tuning parameter.

When we have an extra unlabeled dataset, denoted as $\mathbf{x}_{n+1}, \ldots, \mathbf{x}_{n+m}$, the likelihood becomes complicated because of the Gaussian mixture distribution of the unlabeled data. Thus it is difficult to estimate the parameters via likelihood. Moreover, the graph we need is directly related to $\tilde{\boldsymbol{\Omega}} = \mathrm{var}(X)^{-1}$ rather than $\boldsymbol{\Omega}$. Thus, a penalized likelihood approach is not suitable. Nevertheless, the neighborhood selection approaches are still valid by Lemma 1, because we are concerned with conditional correlation. In particular, we estimate the neighborhoods by

$$\hat{\tilde{\boldsymbol{\gamma}}}^{(j)} = \underset{\tilde{\boldsymbol{\gamma}}^{(j)}}{\mathrm{argmin}} \frac{1}{2(n+m)} \|\tilde{\mathbf{X}}_j - \tilde{\gamma}_0 - \tilde{\mathbf{X}}_{-j}\tilde{\boldsymbol{\gamma}}^{(j)}\|_2^2 + \lambda\|\tilde{\boldsymbol{\gamma}}^{(j)}\|_1,$$

where $\tilde{\mathbf{X}} = (\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{x}_{n+1}, \ldots, \mathbf{x}_{n+m})^\top$ denotes the combined feature matrix. Similarly, we use EBIC to select the tuning parameter λ.

3.2. Parameter estimation and tuning parameter selection

Given the graph $\mathcal{G}$, formulation (4) is a latent group Lasso problem [31]. It can be transformed to an ordinary group Lasso problem as stated in Problem (3). There are many efficient algorithms to solve group Lasso problems, for example, groupwise majorization descent [43]. For very high-dimensional data, we use an iterative proximal algorithm as in [44]. For implementation, we use cross validation for tuning parameter selection.

3.3. Pre-screening

Suppose that there are some entries of $\boldsymbol{\delta}$ being zero. Then $\boldsymbol{\beta}^*$ can be a linear combination of only a few column vectors,

$$\boldsymbol{\beta}^* = \sum_{j\in J} \delta_j \mathbf{w}_j,$$

where $J = \{j : \delta_j \neq 0\}$. Using two-sample t tests for screening, we can specify $J' \subset \{1, \ldots, p\}$, which is a superset of J with a large probability. In particular, we have the following lemma.

Lemma 2. *Define the t-statistic* $T_j = \hat{\delta}_j/\{s_j^{(1)2}/n_1 + s_j^{(2)2}/n_2\}^{1/2}$, *where* $s_j^{(k)2}$ *is the sample variance of feature* $j \in \{1, \ldots, p\}$ *in group* $k \in \{1, 2\}$. *Assume* $\ln p = o(n^\gamma)$, $\ln|J| = o(n^{1/2-\gamma}B_n)$, *and* $\min_{j\in J}|\delta_j|/\sqrt{2\Sigma_{jj}} = B_n/n^\gamma$ *for some* $\gamma \in (0, 1/3)$ *and* $B_n \to \infty$. *Then there exists* $C > 0$ *such that*

$$\lim_{n\to\infty} \Pr\left\{\min_{j\in J}|T_j| \geq Cn^{\gamma/2},\ \max_{j\notin J}|T_j| < Cn^{\gamma/2}\right\} = 1.$$

The result in Lemma 2 was previously obtained by Fan and Fan [11] and the corresponding proof is omitted. Lemma 2 guarantees the accuracy of our pre-screening procedure.

After feature screening, the proposed regularization can be simplified as follows:

$$\|\boldsymbol{\beta}\|_{\mathcal{G}_{J'},\boldsymbol{\tau}} = \min_{\sum_{j\in J'}\mathbf{v}^{(j)}=\boldsymbol{\beta},\ \mathrm{supp}(\mathbf{v}^{(j)})\subseteq\mathcal{N}(j)} \sum_{j\in J'} \tau_j\|\mathbf{v}^{(j)}\|_2. \tag{9}$$

Compared with the original regularization (5), the new one in (9) is often simpler and enjoys computational advantages. Moreover, the new regularization (9) only requires part of the graph, i.e., the part corresponding to the support of $\{\boldsymbol{\omega}_j : j \in J'\}$. Graph estimation methods based on neighborhood selection fit into this idea naturally. When $\boldsymbol{\delta}$ is approximately sparse and $|J'| \ll p$, the computational cost can be reduced substantially. Unlike the feature screening in [11], features outside J' are not necessarily excluded. Instead, they can be introduced into the model via connection with other features in J'.

4. Theoretical properties

In this section, we study the theoretical properties of GSLDA. In particular, the original GSLDA in (4) with a known graph $\mathcal{G}$ is considered. Since the semi-supervised GSLDA only differs from GSLDA in the graph used, we do not consider it separately. In Section 4.1, we show the selection consistency of GSLDA. In Section 4.2, we study the misclassification rate of the GSLDA and compare it with the Bayes error.

Before diving into the theoretical analysis, we first introduce some notations for our setting. We define, for an n-dimensional vector $\mathbf{a}$, $\|\mathbf{a}\|_\infty = \max(|a_1|, \ldots, |a_n|)$; for an $n \times m$ matrix $\mathbf{A}$, $\|\mathbf{A}\|_\infty = \max_i\{|A_{i1}| + \cdots + |A_{im}|\}$ and $|||\mathbf{A}|||_\infty = \max_{i,j}|A_{ij}|$. We consider the problem setting of standard LDA, in which both within-class populations are Gaussian, i.e., $\mathcal{N}(\boldsymbol{\mu}^{(1)}, \boldsymbol{\Sigma})$ and $\mathcal{N}(\boldsymbol{\mu}^{(2)}, \boldsymbol{\Sigma})$. The discriminant vector of the Bayes rule, denoted as $\boldsymbol{\beta}^*$, is given in (1). Denote $A = \{j : \beta_j^* \neq 0\}$ the active set, and $s = |A|$. Define $\boldsymbol{\beta}^\dagger = \tilde{\boldsymbol{\Omega}}\boldsymbol{\delta}$, then $\boldsymbol{\beta}^\dagger$ is proportional to $\boldsymbol{\beta}^*$ (Proposition 1) and thus defines an equivalent classifier.

4.1. Selection consistency

Assume that the feature vectors are centralized, thus $\hat{\boldsymbol{\beta}} = \operatorname{argmin}_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2/n + \lambda\|\boldsymbol{\beta}\|_{\mathcal{G},\boldsymbol{\tau}}$. Denote $\tilde{\mathbf{S}} = \mathbf{X}^\top\mathbf{X}/n$, and $\kappa = \|\tilde{\boldsymbol{\Sigma}}_{A^\complement A}\tilde{\boldsymbol{\Sigma}}_{AA}^{-1}\|_\infty$. Define

$$\tilde{\tau}_j = \min_{\ell}\{\tau_\ell : j \in \mathcal{N}^{(\ell)}\}, \quad \tau^* = \max_{j\in A}\tilde{\tau}_j, \quad \tau_* = \min_{j\in A^\complement}\tau_j|\mathcal{N}^{(j)}|^{-1/2}.$$

We present several assumptions to be used as follows.

(A1) $p = O\{\exp(n^\gamma)\}$, $s = o(n^a)$, for some $\gamma \in (0, 1)$, $a \in (0, (1-\gamma)/2)$.
(A2) For every $j \in \{1, \ldots, p\}$, either $\mathcal{N}^{(j)} \subseteq A$ or $\mathcal{N}^{(j)} \subseteq A^\complement$.
(A3) $\|\tilde{\boldsymbol{\Sigma}}_{AA}^{-1}\|_\infty$ is bounded by $\varphi < \infty$.
(A4) $\|\tilde{\boldsymbol{\Sigma}}_{A^\complement A}\tilde{\boldsymbol{\Sigma}}_{AA}^{-1}\|_\infty < \tau_*/\tau^*$.
(A5) $b = \min_{j\in A}|\beta_j^\dagger| \gg \sqrt{\ln p/n}$.

Here (A1) specifies the order of feature dimension as well as the number of discriminating features. By Assumption (A2), a discriminative feature can only be connected with other discriminative features. This is a reasonable condition in reality since a feature is often relevant for classification if it is related to another useful feature. Condition (A3) ensures that there is no extreme collinearity among discriminative features. Assumption (A4) is an irrepresentability condition that is often employed in showing the selection consistency of regularized estimators [28,51].

It may not be immediately clear why we impose the irrepresentability condition (A4) on $\tilde{\boldsymbol{\Omega}}$ rather than $\boldsymbol{\Omega}$. Note that the more similarity between predictive and non-predictive features, the more difficult it is to achieve selection consistency. While $\boldsymbol{\Omega}$ encodes the within-class feature dependence, the relationship among features in the whole dataset is determined by the overall covariance. Thus we impose the condition on $\tilde{\boldsymbol{\Omega}}$. The main theoretical result on the selection consistency of the GSLDA is given in the following theorem.

Theorem 1 (*Selection Consistency*). *Under conditions (A1)–(A5), let $\sqrt{\ln p/n} \le \lambda\tau^* \le O(b)$ and n be sufficiently large, then the GSLDA recovers the active set A and $\|\hat{\boldsymbol{\beta}}_A - \boldsymbol{\beta}_A^\dagger\|_\infty = O(\sqrt{\ln p/n})$ with probability at least $1 - O(p^{-C_1})$ for some $C_1 > 0$.*

When we use an empty graph $\mathcal{G}$ and set $\tau_j = 1$ for all j, our GSLDA is equivalent to the DSDA method. In this special case, $\tau^* = \tau_* = 1$, and the selection consistency conditions are similar to those for DSDA [26].

4.2. Convergence rate

With respect to a classifier, the error rate is one of the most important performance measures. In this section, we investigate the misclassification rate of GSLDA. We first present some basic results on the classification problem. For a linear classifier $g_{\beta_0,\boldsymbol{\beta}}$, denote its classification error under our settings as $Q_{\beta_0,\boldsymbol{\beta}} = \Pr\{g_{\beta_0,\boldsymbol{\beta}}(X) \ne G\}$. Then we have the following results from [6].

Lemma 3 (*Classification Error Rate in LDA Setting*). *Under our setting,*

$$Q_{\beta_0,\boldsymbol{\beta}} = \frac{1}{2}\,\Phi\left(\frac{-\beta_0 - \boldsymbol{\beta}^\top\boldsymbol{\mu}^{(1)}}{\sqrt{\boldsymbol{\beta}^\top\boldsymbol{\Sigma}\boldsymbol{\beta}}}\right) + \frac{1}{2}\,\Phi\left(\frac{\beta_0 + \boldsymbol{\beta}^\top\boldsymbol{\mu}^{(2)}}{\sqrt{\boldsymbol{\beta}^\top\boldsymbol{\Sigma}\boldsymbol{\beta}}}\right),$$

where Φ denotes the cumulative distribution function of $\mathcal{N}(0, 1)$. The misclassification rate of the Bayes classifier $g_{\beta_0^*,\boldsymbol{\beta}^*}$ is $Q_{\beta_0^*,\boldsymbol{\beta}^*} = \Phi(-\Delta^{1/2}/2)$, where $\Delta = \boldsymbol{\delta}^\top\boldsymbol{\Omega}\boldsymbol{\delta}$.

Since Q is a continuous function of β_0 and $\boldsymbol{\beta}$, the misclassification rate of the GSLDA classifier is asymptotically the same as the Bayes error rate, i.e., $Q_{\hat{\beta}_0,\hat{\boldsymbol{\beta}}} \xrightarrow{p} Q_{\beta_0^*,\boldsymbol{\beta}^*}$, as long as $\hat{\boldsymbol{\beta}} \xrightarrow{p} \boldsymbol{\beta}^*$. A more interesting problem is the order of the misclassification rate of the GSLDA when $Q_{\beta_0^*,\boldsymbol{\beta}^*} \to 0$. To investigate this, we first introduce a new condition, under which we can construct an ℓ_2 error bound for the GSLDA estimator.

(A6) Denote $\mathcal{C}(A) = \{\boldsymbol{\Delta} \in \mathbb{R}^p : \|\boldsymbol{\Delta}_{A^\complement}\|_{\mathcal{G},\boldsymbol{\tau}} \le 3\|\boldsymbol{\Delta}_A\|_{\mathcal{G},\boldsymbol{\tau}}\}$, where $\boldsymbol{\Delta}_A = (\Delta_j\mathbf{1}(j \in A))_{p\times 1}$ and $\boldsymbol{\Delta}_{A^\complement} = (\Delta_j\mathbf{1}(j \notin A))_{p\times 1}$. For all $\boldsymbol{\Delta} \in \mathcal{C}(A)$, $\boldsymbol{\Delta}^\top\tilde{\boldsymbol{\Sigma}}\boldsymbol{\Delta}/\boldsymbol{\Delta}^\top\boldsymbol{\Delta} \ge \sigma > 0$.

This is actually a restricted eigenvalue condition, which is often used in showing the error bound for regularized estimators [30]. Compared to the irrepresentability condition (A4), this is much less stringent. With the new condition, we have the following ℓ_2 error bound for the GSLDA estimator.

Theorem 2 (ℓ_2-*error Bound*). *Under conditions (A1)–(A2) and (A6), let $\lambda \ge 4C_2(1 + \|\boldsymbol{\beta}^\dagger\|_1)\sqrt{\ln p/n}$ for some $C_2 > 0$ and n be sufficiently large, then $\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^\dagger\|_2^2 \le 9\lambda^2 s\tau^{*2}/\sigma^2$ with probability at least $1 - sp^{-C_3}$ for some $C_3 > 0$.*

Based on Theorem 2, we can establish the asymptotic error rate of the GSLDA classifier as follows.

Theorem 3 (*Convergence Rate*). *Under conditions (A1)–(A2) and (A6), as* $n, p \to \infty$, *if* $\Delta \to \infty$, *we have*

$$Q_{\beta_0^*,\boldsymbol{\beta}^*} \to 0 \quad \text{and} \quad Q_{\hat{\beta}_0,\hat{\boldsymbol{\beta}}}/Q_{\beta_0^*,\boldsymbol{\beta}^*} \xrightarrow{p} 1,$$

given $\lambda\tau^* = o[\min\{\lambda_{\max}(\boldsymbol{\Sigma})^{-1}\Delta^{-2}s^{-1/2}\|\boldsymbol{\beta}^\dagger\|_2^{-1}, \Delta^{-1}s^{-1/2}\|\boldsymbol{\delta}\|_2^{-1}\}]$ *and* $\|\boldsymbol{\beta}^\dagger\|_1 = o(n^{1-\gamma}\Delta^{-1})$, *where* Δ *is defined as in Lemma 3 and* $\lambda_{\max}(\boldsymbol{\Sigma})$ *denotes the largest eigenvalue of* $\boldsymbol{\Sigma}$.

That is, under mild conditions, the misclassification rate of the GSLDA classifier is of the same order as the Bayes error rate in this case.

5. Simulation study

To demonstrate the performance of the GSLDA methods, we compare them with several existing high-dimensional LDA extensions and other classification methods. The methods in comparison include the naive Bayes rule (NB), nearest shrunken centroids (NSC), sparse LDA (SLDA) [35], ℓ_1 penalized Logistic regression (PLR), penalized Fisher's discriminant analysis (PLDA) [39], direct sparse discriminant analysis (DSDA) [26], linear programming discriminant (LPD) [6], and the ROAD [12]. In particular, the methods NSC, PLR, PLDA and DSDA are implemented with R packages `pamr`, `glmnet`, `penalizedLDA` and `dsda`, respectively. We implement the LPD method via the parametric simplex algorithm [37] as suggested in [34].

Besides the above supervised methods, there are many semi-supervised clustering (or classification) methods; see, e.g., [20,32,52]. We have implemented the semi-supervised spectral clustering (SSSC) method proposed in [20]. Both the original and the semi-supervised GSLDA are implemented, and the latter is denoted as GSLDA-S. We also include the GSLDA methods with the true graph, denoted as GSLDA-O (with $\mathcal{G}$) and GSLDA-SO (with $\tilde{\mathcal{G}}$), in the comparison. To make a fair comparison, pre-screening is not employed in the numerical studies. The Bayes rule, denoted as Oracle, is used as a benchmark.

In the simulation, we fix the dimension $p = 200$ and the sample size $n = 200$. The labels $g_1, \ldots, g_n$ are generated with $\pi_1 = \pi_2 = 1/2$ and the features are sampled from $\mathcal{N}(\boldsymbol{\mu}^{(g_i)}, \boldsymbol{\Omega}^{-1})$ based on the labels. Moreover, we generate an independent dataset of sample size 2000 and remove the labels, for the semi-supervised methods. All tuning parameters are selected by 10-fold cross validation. We consider four different feature structures as follows.

Example 1 (*Blockwise Sparse Model*). In this example, $\boldsymbol{\Sigma}^B$ is a 5×5 matrix with 1 for the diagonal and 0.7 for off-diagonal elements. We use 20 such blocks for the diagonal of the covariance matrix $\boldsymbol{\Sigma}$ and 0 for the rest, and let $\boldsymbol{\Omega} = \boldsymbol{\Sigma}$. The group means are generated such that $\mu_j^{(1)} = 0.5$ for $j \in \{5, 10, \ldots, 25\}$ and $\mu_j^{(1)} = 0$ otherwise; and $\boldsymbol{\mu}^{(2)} = -\boldsymbol{\mu}^{(1)}$.

Example 2 (*AR(3) Model*). The precision matrix $\boldsymbol{\Omega}$ is generated such that $\omega_{jj} = 1$, and $\omega_{j\ell} = -2/3$ if $1 \le |j - \ell| \le 3$ and 0 otherwise. The group means are generated such that $\mu_j^{(1)} = 0.75$ for $j \in \{5, 10, \ldots, 25\}$ and $\mu_j^{(1)} = 0$ otherwise; and $\boldsymbol{\mu}^{(2)} = -\boldsymbol{\mu}^{(1)}$.

Example 3 (*Random Sparse Model*). The graph $\mathcal{G}$ is generated in such a way that any two nodes are connected with probability 0.05. Based on $\mathcal{G}$, we generate the precision matrix $\boldsymbol{\Omega}$ by setting $\omega_{j\ell} = -0.5$ for all connected j and ℓ in the graph and 0 otherwise. We add $c\,\mathbf{I}_p$, where $c > 0$ and $\mathbf{I}_p$ is an identity matrix, to $\boldsymbol{\Omega}$ such that the eigenvalues are between 0 and 1. We standardize $\boldsymbol{\Omega}$ so that its diagonal elements are all 1. The group means are generated in such a way that $\mu_j^{(1)} = 0.75$ for all $j \in S$ and 0 otherwise; and $\boldsymbol{\mu}^{(2)} = -\boldsymbol{\mu}^{(1)}$.

Example 4 (*Scale-free Random Graph*). The graph is generated in a way similar to the Barabasi–Albert (BA) model. Starting from an identity matrix $\mathbf{L} \in \mathbb{R}^{p\times p}$, at step i we randomly assign -0.5 to $\min\{\lfloor 0.05p \rfloor, i-1\}$ entries in row i with probability $\Pr(i,j) \propto \#\{L_{\ell j} \neq 0 : 1 \le \ell \le p\}, j < i$. Repeat the procedure until $i = p$. Then we get a lower triangular matrix. We construct $\boldsymbol{\Omega} = \mathbf{L}^\top\mathbf{L}$ and standardize it such that the eigenvalues are between 0 and 1. Denote the 6th to 10th most connected nodes as J. The group means are generated such that $\mu_j^{(1)} = 0.75$ for all $j \in J$ and 0 otherwise; and $\boldsymbol{\mu}^{(2)} = -\boldsymbol{\mu}^{(1)}$.

All four graph structures are displayed in Fig. 2. The first two examples are fixed while the last two produce random graphs. Compared with the random sparse model, the scale-free random graphs are featured with hubs. For each graph structure, we repeat the simulation for 100 times and evaluate the performance, both prediction and selection accuracy, of all classification methods. Table B.1 in the Appendix displays the graph estimation accuracy for all examples.

Tables 1–4 give a summary of the performance comparison of all methods in Examples 1 and 4. In particular, misclassification rates in percentage (Error), false positives (FP) and false negatives (FN) of $\boldsymbol{\beta}$ estimation are computed. The misclassification rate is evaluated based on an independent test dataset of size 20,000. All metrics are averaged over 100 simulations and the numbers within parentheses are the standard errors. Both the NB and the SSSC are not considered in the comparison of variable selection, since these methods do not perform variable selection.

From Tables 1–4, we observe that the two plug-in extensions of LDA, namely the naive Bayes and the NSC, perform worse than ℓ_1 penalized logistic regression and other direct LDA methods under these settings. This is expected because there is substantial correlation among the features while both the plug-in extensions of LDA use diagonal estimates of $\boldsymbol{\Sigma}$. In contrast, the performance of the direct LDA methods varies across the settings. For example, the DSDA has lower misclassification rates

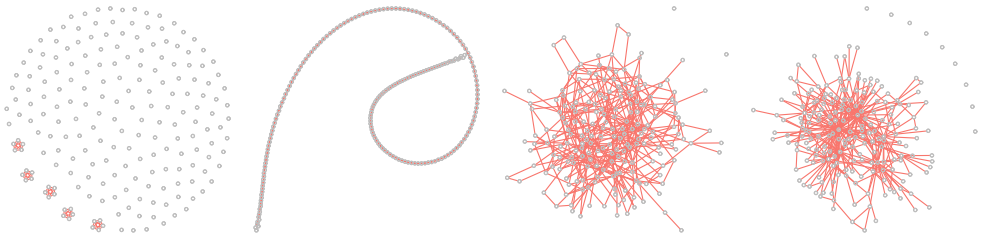

Fig. 2. The graph structures used in the simulation study. From left to right: the blockwise sparse model, the AR(3) model, the random sparse model, the scale-free model. The last two plots use one realization for demonstration, and the graphs may vary among different realizations.

Table 1
Performance comparisons of different classification methods for Example 1.

	Error	FP	FN	Size
NB	27.01 (0.18)	–	–	–
NSC	14.17 (0.11)	0.71 (0.54)	20.27 (0.13)	5.44 (0.62)
SLDA	10.28 (0.16)	5.71 (1.29)	12.53 (0.31)	18.18 (1.61)
PLR	7.17 (0.13)	14.73 (0.56)	8.1 (0.24)	31.63 (0.69)
DSDA	6.76 (0.13)	23.26 (1.53)	6.79 (0.27)	41.47 (1.71)
LPD	7.80 (0.38)	37.20 (1.97)	5.73 (0.29)	56.47 (2.17)
ROAD	6.54 (0.12)	23.45 (1.24)	6.01 (0.24)	42.44 (1.37)
PLDA	14.16 (0.10)	3.62 (1.16)	19.53 (0.16)	9.09 (1.29)
SSSC	8.11 (0.10)	–	–	–
GSLDA	5.57 (0.07)	20.48 (2.17)	7.31 (0.25)	38.17 (2.33)
GSLDA-S	4.53 (0.06)	18.79 (2.26)	0.74 (0.11)	43.05 (2.29)
GSLDA-O	4.86 (0.08)	18.55 (2.63)	0 (0)	43.55 (2.63)
GSLDA-SO	4.52 (0.07)	16.43 (1.93)	0 (0)	41.43 (1.93)
Oracle	3.27 (0.01)	0 (0)	0 (0)	25 (0)

Table 2
Performance comparisons of different classification methods for Example 2.

	Error	FP	FN	Size
NB	36.59 (0.43)	–	–	–
NSC	17.46 (0.14)	42.75 (2.16)	25.96 (0.45)	55.79 (2.48)
SLDA	14.39 (0.12)	19.28 (1.72)	17.59 (0.43)	40.69 (2.17)
PLR	7.86 (0.11)	15.83 (0.42)	20.58 (0.29)	34.25 (0.54)
DSDA	6.96 (0.09)	25.13 (1.22)	17.21 (0.38)	46.92 (1.46)
LPD	8.84 (0.69)	34.48 (1.56)	17.98 (0.48)	55.50 (1.97)
ROAD	7.42 (0.12)	25.16 (0.98)	17.36 (0.35)	46.80 (1.17)
PLDA	16.48 (0.12)	2.26 (0.48)	32.69 (0.14)	8.57 (0.57)
SSSC	9.27 (0.17)	–	–	–
GSLDA	6.60 (0.10)	25.48 (1.83)	15.41 (0.43)	49.07 (2.19)
GSLDA-S	5.56 (0.07)	34.43 (2.52)	3.33 (0.41)	70.1 (2.77)
GSLDA-O	6.19 (0.09)	27.26 (1.72)	7.37 (0.47)	58.89 (2.08)
GSLDA-SO	5.79 (0.07)	30.78 (1.94)	2.16 (0.39)	67.62 (2.31)
Oracle	3.32 (0.01)	0 (0)	0 (0)	39 (0)

than the ROAD in most cases, while ROAD has better classification accuracy in Example 1. Utilizing the graph structures, high-dimensional LDA is further improved in GSLDA. As we can see from the results, GSLDA methods have the best performance among all methods in these four settings. In particular, the GSLDA method has lower misclassification rates than all other methods except its semi-supervised variant. Since the DSDA is the special case of the GSLDA with an empty graph, it is a good benchmark to quantify the benefit of using graph structures. In most cases, the GSLDA provides better model selection than the DSDA. Therefore, utilizing the graph structure does help us to improve the LDA classifier in high dimensions.

With respect to the semi-supervised GSLDA, due to the large amount of unlabeled data, it often has better graph estimation and yields more accurate classifiers. In fact, the semi-supervised GSLDA has the lowest misclassification rates among all methods in all cases. Furthermore, the semi-supervised GSLDA has superior model selection over the original GSLDA in most cases. This demonstrates the advantages of using unlabeled data.

We notice that models estimated by the semi-supervised GSLDA often have larger sizes, sometimes more false positives in coefficient vectors, than the original GSLDA classifiers. This is probably because the graph used in the semi-supervised GSLDA often has more edges. There are two possible reasons: (i) the true graph $\tilde{\mathcal{G}}$ corresponding to $\tilde{\mathbf{\Omega}}$ has more edges than $\mathcal{G}$, and (ii) graph estimation based on unlabeled data uses a much larger training dataset which often leads to denser graphs estimate. While a denser graph estimate may recover more connections among the features, it can also result in more false edges. This

Table 3
Performance comparisons of different classification methods for Example 3.

	Error	FP	FN	Size
NB	36.86 (0.80)	–	–	–
NSC	24.16 (0.84)	29.15 (3.35)	44.78 (1.62)	50.37 (4.93)
SLDA	13.28 (0.72)	21.07 (2.29)	40.59 (1.57)	46.48 (3.87)
PLR	11.09 (0.12)	21.44 (0.56)	42.08 (0.39)	45.36 (0.75)
DSDA	10.94 (0.15)	30.32 (1.49)	38.12 (0.63)	58.20 (2.01)
LPD	13.19 (0.73)	41.67 (1.52)	39.84 (0.82)	67.83 (2.25)
ROAD	11.25 (0.15)	33.14 (1.46)	37.53 (0.55)	61.61 (1.92)
PLDA	26.31 (0.68)	22.34 (2.33)	50.89 (1.12)	37.45 (3.41)
SSSC	13.57 (0.91)	–	–	–
GSLDA	10.53 (0.10)	27.34 (1.91)	36.67 (0.85)	56.67 (2.67)
GSLDA-S	8.77 (0.08)	34.08 (2.77)	18.2 (0.72)	81.88 (3.37)
GSLDA-O	9.77 (0.08)	36.87 (2.54)	26.22 (0.78)	76.65 (3.27)
GSLDA-SO	8.91 (0.08)	35.17 (2.37)	16.31 (0.63)	84.86 (3.01)
Oracle	5.36 (0.02)	0 (0)	0 (0)	66 (0)

Table 4
Performance comparisons of different classification methods for Example 4.

	Error	FP	FN	Size
NB	32.84 (0.28)	–	–	–
NSC	22.78 (0.12)	8.62 (1.76)	48.87 (0.76)	18.75 (2.49)
SLDA	17.53 (0.27)	19.83 (1.29)	38.23 (0.67)	40.60 (1.98)
PLR	14.60 (0.21)	16.51 (0.66)	35.5 (0.5)	40.01 (1.06)
DSDA	13.48 (0.17)	33.71 (1.9)	28.18 (0.65)	64.53 (2.47)
LPD	16.87 (0.36)	46.86 (1.66)	29.17 (0.71)	76.69 (2.31)
ROAD	13.95 (0.19)	36.9 (2.01)	27.71 (0.77)	68.19 (2.69)
PLDA	22.64 (0.12)	6.6 (1.06)	51.58 (0.45)	14.02 (1.48)
SSSC	12.08 (0.21)	–	–	–
GSLDA	10.46 (0.11)	21.53 (1.7)	15.69 (0.55)	64.84 (2.14)
GSLDA-S	9.15 (0.12)	12.87 (1.29)	5.03 (0.54)	66.84 (1.57)
GSLDA-O	10.39 (0.18)	28.29 (1.73)	19.44 (0.8)	67.85 (2.43)
GSLDA-SO	9.36 (0.17)	19.87 (1.69)	5.47 (0.71)	73.05 (2.31)
Oracle	4.62 (0.02)	0 (0)	0 (0)	59 (0)

effect is enhanced by the difficulty of graph estimation with unlabeled data. As a consequence, the semi-supervised GSLDA may suffer from more false positives, as shown in Examples 2 and 3. To resolve this issue, we may consider to use more conservative graph estimation for the semi-supervised GSLDA.

6. Real data analysis

In this section, we implement our methods and several other existing classifiers on two real datasets. The first dataset is a genetic dataset with very high dimensions, and the second one consists of images of handwritten digits. We estimate the graphs from labeled training data and unlabeled data. We find that GSLDA methods have a good performance in both datasets and utilizing the feature structure is beneficial.

6.1. Arcene cancer data

Nowadays, genetic diagnosis is an important tool in the clinical study and medical practice. By using the genetic information, we can estimate the potential risk of cancer for healthy people or determine cancer subtypes for patients. The Arcene dataset is a gene dataset of 88 cancer patients and 112 healthy individuals. The dataset contains 10,000 features and was originally used in the NIPS 2003 feature selection challenge (https://archive.ics.uci.edu/ml/datasets/Arcene). Out of the 10,000 features, 7000 are real genes while the other 3000 are noise features that have no predictive power and make the prediction harder. Besides the labeled data, there is an unlabeled dataset of 700 individuals, which is used to construct a graph for GSLDA-S. As in the previous simulation studies, we apply the GSLDA and other methods on the dataset.

The labeled data are randomly split into a training set and a test set, of sizes 150 and 50, respectively. All methods except the naive Bayes are tuned by 10-fold cross validation. The experiment is repeated 100 times and the results are summarized in Table 5.

From Table 5, we can see that both GSLDA and semi-supervised GSLDA outperform other methods in prediction. Although semi-supervised GSLDA uses more data for graph estimation, its performance is inferior to GSLDA for this application, possibly due to the difficulty of graph estimation based on unlabeled data. In addition, the size of the unlabeled dataset is not substantially larger than that of the labeled dataset. Compared with PLR, DSDA and ROAD, our methods have significantly

Table 5
Comparison of GSLDA and other methods on the Arcene dataset.

	Error	Size
NB	35.50 (0.62)	–
NSC	36.05 (0.61)	9934.46 (9.06)
SLDA	34.64 (0.73)	297 (4.17)
PLR	28.36 (0.65)	16.57 (0.90)
DSDA	28.29 (0.72)	30.96 (2.69)
LPD	31.59 (1.33)	10.95 (3.58)
ROAD	29.29 (0.64)	31.86 (3.43)
PLDA	34.36 (0.61)	9.39 (1.63)
SSSC	27.93 (0.83)	–
GSLDA	**22.57 (0.70)**	229.36 (6.39)
GSLDA-S	24.50 (0.68)	319.57 (8.37)

Table 6
Comparison of GSLDA and other methods on the Semeion dataset.

	Error	Size
NB	13.81 (0.34)	–
NSC	15.21 (0.44)	84.74 (11.31)
SLDA	14.43 (0.67)	20.23 (2.80)
PLR	18.69 (0.88)	9.46 (0.40)
DSDA	13.76 (0.66)	16.76 (1.01)
LPD	17.15 (0.86)	15.32 (0.91)
ROAD	19.73 (0.98)	15.38 (1.25)
SSSC	13.97 (0.75)	–
GSLDA	12.65 (0.61)	28.46 (1.45)
GSLDA-S	**11.23 (0.56)**	33.28 (1.32)

larger model sizes. This may indicate that many genes are related to each other. It is likely that those genes contribute to cancer together, and including all of them in modeling can potentially make the classifier more robust. This characteristic may also contribute to the good performance of the proposed two GSLDA methods.

6.2. Semeion handwritten digits dataset

The Semeion dataset (https://archive.ics.uci.edu/ml/datasets/Semeion+Handwritten+Digit) consists of 1593 images of handwritten digits. Each digit is in the form of a 16×16 grayscale image and saved as a vector of 256 features. We take a subset of the dataset that only contains digits 1 and 7, which are generally difficult to distinguish. We randomly choose 40 images for training, and 80 for graph estimation of the semi-supervised GSLDA after removing labels. The remaining 200 images are used for testing. Other settings are the same as the cancer example. Table 6 gives a summary of the results.

As shown in Table 6, the semi-supervised GSLDA has excellent performance for this problem. It has the lowest misclassification rate among all methods in comparison. The original GSLDA method also has good classification accuracy for this problem. Moreover, we can see that both GSLDA methods have larger model sizes than other direct LDA methods, as in the previous analysis in Section 6.1.

7. Discussion

With many extensions in the literature, LDA can be readily applied to high-dimensional classification problems. In particular, the direct approaches of high-dimensional LDA are attractive due to their simplicity and good performance. Under the standard setting of LDA problems, we explore the relationship between the graph structure of features and the optimal discriminant vector $\boldsymbol{\beta}^*$. Our study shows that, by taking advantage of such structure, we can get better LDA classifiers in high dimensions. Based on this idea, we propose the GSLDA method. After investigating the overall graph structure of the Gaussian mixture population for unlabeled data, we further propose the semi-supervised GSLDA that can utilize unlabeled data. Both GSLDA methods have been evaluated on simulated and real data, which demonstrate the advantages of utilizing the graph structures. Moreover, we conclude that the performance of semi-supervised GSLDA depends on both the size of the unlabeled dataset and the graph complexity. When the graph structure is very complex, it is better to consider a conservative graph estimate for GSLDA. Finally, our focus in this paper is on binary problems. It will be useful to extend the methods for multicategory problems.

Acknowledgments

The authors would like to thank the Editor-in-Chief, Christian Genest, the Associate Editor, and reviewers for their valuable comments and suggestions which led to a much improved presentation. This research was supported in part by National Science Foundation, USA Grants IIS1632951, DMS1821231, and National Institute of Health, USA Grant R01GM126550.

$$\boldsymbol{\beta}^* = \begin{pmatrix} \omega_{11} & \omega_{12} & \omega_{13} \\ \omega_{21} & \omega_{22} & 0 \\ \omega_{31} & 0 & \omega_{33} \end{pmatrix} \begin{pmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \end{pmatrix} = \begin{pmatrix} \delta_1\omega_{11} + \delta_2\omega_{21} + \delta_3\omega_{31} \\ \delta_1\omega_{12} + \delta_2\omega_{22} \\ \delta_1\omega_{13} + \delta_3\omega_{33} \end{pmatrix}$$

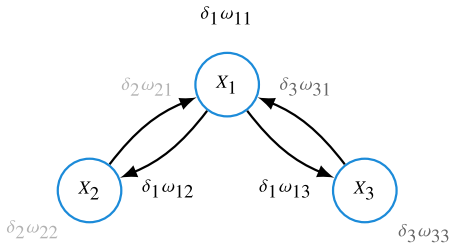

Fig. A.1. A 3-dimensional LDA example demonstrating how marginal differences of the three features $(\delta_1, \delta_2, \delta_3)$ contribute to the predictive power of all features. Here $\omega_{23} = \omega_{32} = 0$. The terms around each node represent a decomposition of the corresponding coefficient. The gray scale of each term and the edge direction together indicate the source of the marginal differences.

Table B.1
Graph estimation accuracy for all examples in the simulations. The graphs are estimated with labeled data (L) after centering, or with unlabeled data (U). The former estimation is compared with $\mathcal{G}$, and the latter is compared with both $\mathcal{G}$ and $\tilde{\mathcal{G}}$. The results are averaged over 100 repetitions and the standard errors are provided in the parentheses.

Graph type	Data	TP	FP	Size	True size
Block sparse	L	51.54 (0.25)	6.86 (0.48)	58.4 (0.55)	$\mathcal{G}$: 100 (0)
	U	100 (0)	82.96 (0.66)	182.96 (0.66)	$\mathcal{G}$: 100 (0)
	U	176.48 (0.46)	6.48 (0.38)	182.96 (0.66)	$\tilde{\mathcal{G}}$: 600 (0)
AR(3)	L	468.02 (1.31)	69.64 (1.24)	537.66 (1.34)	$\mathcal{G}$: 1188 (0)
	U	1178.04 (0.38)	76.72 (0.87)	1254.76 (1.01)	$\mathcal{G}$: 1188 (0)
	U	1235.76 (0.75)	19 (0.61)	1254.76 (1.01)	$\tilde{\mathcal{G}}$: 2508 (0)
Random sparse	L	353.14 (2.02)	69.34 (1.25)	422.48 (1.73)	$\mathcal{G}$: 818 (0)
	U	814.52 (0.18)	72.74 (1.04)	887.26 (1.12)	$\mathcal{G}$: 818 (0)
	U	866.14 (0.87)	21.12 (0.70)	887.26 (1.12)	$\tilde{\mathcal{G}}$: 2426 (0)
Scale-free	L	374.92 (1.37)	32.44 (0.92)	407.36 (1.48)	$\mathcal{G}$: 776 (0)
	U	709.88 (0.73)	103.5 (1.00)	813.38 (1.18)	$\mathcal{G}$: 776 (0)
	U	799.08 (1.05)	14.3 (0.53)	813.38 (1.18)	$\tilde{\mathcal{G}}$: 3564 (0)

Appendix A. Some comments on the GSLDA method

A.1. A graphical display of the discriminant vector decomposition

See Fig. A.1.

A.2. Connection between GSLDA and existing methods

We first consider the case when $\mathcal{G}$ is a complete graph. Without loss of generality, we assume that there is a unique minimum weight, i.e., there exists an ℓ such that $\tau_\ell < \tau_j$ for all $j \neq \ell$. In this case, for any $\boldsymbol{\beta} \in \mathbb{R}^p$ and $\mathbf{v}^{(1)} + \cdots + \mathbf{v}^{(p)} = \boldsymbol{\beta}$, we have

$$\sum_{j=1}^{p} \tau_j \|\mathbf{v}^{(j)}\|_2 \geq \tau_\ell \sum_{j=1}^{p} \|\mathbf{v}^{(j)}\|_2 \geq \tau_\ell \|\boldsymbol{\beta}\|_2.$$

By taking $\mathbf{v}^{(l)} = \boldsymbol{\beta}$ and $\mathbf{v}^{(j)} = \mathbf{0}$ for all $j \neq \ell$, the regularization (5) becomes $\|\boldsymbol{\beta}\|_{\mathcal{G},\boldsymbol{\tau}} = \tau_\ell \|\boldsymbol{\beta}\|_2$. Similarly, we can show the equivalence in the case where $\mathcal{G}$ consists of K disjoint complete subgraphs.

Appendix B. Numerical results

B.1. Graph estimation results

To better understand the performance of our proposed GSLDA methods, we also present the graph estimation results for the methods. In particular, we compare the graph estimation based on both labeled data (for supervised GSLDA) and unlabeled data (for semi-supervised GSLDA) with the true graphs, within-class graph $\mathcal{G}$ and overall graph $\tilde{\mathcal{G}}$. The accuracy metrics include false positives (TP) and false positives (FP).

B.2. Additional simulation results

The misclassification rates may not reflect the comprehensive performance of classification models, especially when the classes are unbalanced. Thus we present the receiver operating characteristic (ROC) curve for the classification models.

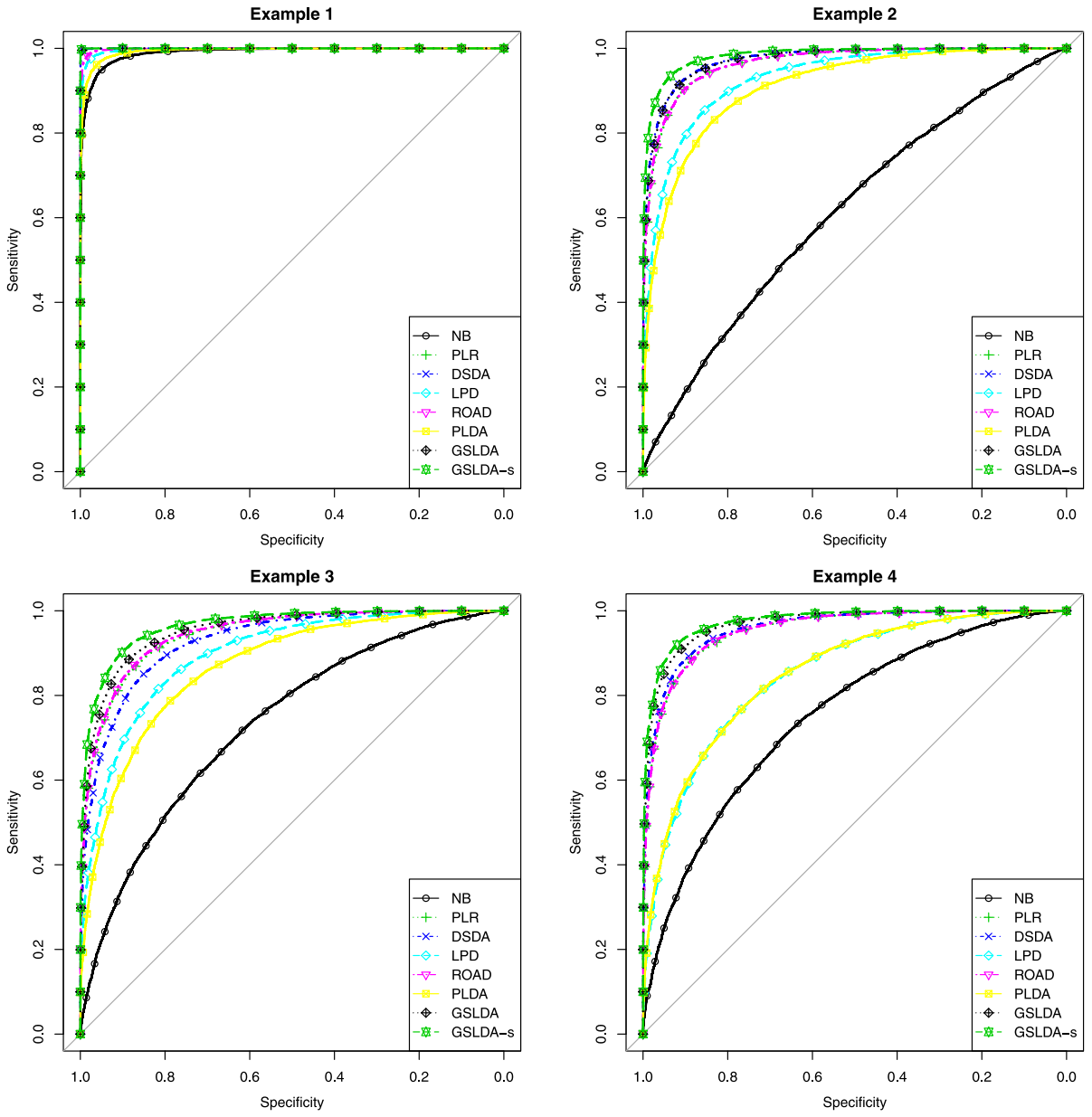

Fig. B.1. ROC Curve under the balanced setting for the four examples. The proportion of Class-0 sample is 50%. The ROC curve is computed based on 100 repetitions.

Besides the balanced class setting as in the main text, we also consider an unbalanced class setting in which Class-0 accounts for 80% of the whole dataset. As we can see from Figs. B.1 and B.2, our methods still outperform other methods in terms of higher sensitivities at each specificity level.

Appendix C. Proofs to the theoretical results

C.1. Proof of Proposition 1

The random variable $\mathbf{X}$ can be represented as $\mathbf{X} = \xi\mathbf{Z}_1 + (1-\xi)\mathbf{Z}_2$, where (i) $\xi \sim \text{Bin}(1, \pi_1)$ is a Bernoulli random variable and (ii) $\mathbf{Z}_1$ and $\mathbf{Z}_2$ are from the two population components, respectively. Moreover, ξ, $\mathbf{Z}_1$, and $\mathbf{Z}_2$ are mutually independent. We have $\text{var}(\mathbf{Z}_1) = \text{var}(\mathbf{Z}_2) = \boldsymbol{\Sigma}$, $\text{E}\mathbf{Z}_1 = \boldsymbol{\mu}^{(1)}$, and $\text{E}\mathbf{Z}_2 = \boldsymbol{\mu}^{(2)}$. Then $\text{E}(\mathbf{X}) = \pi_1\boldsymbol{\mu}^{(1)} + \pi_2\boldsymbol{\mu}^{(2)}$ and

$$\text{E}(\mathbf{X}\mathbf{X}^\top) = \text{E}\{\xi^2\mathbf{Z}_1\mathbf{Z}_1^\top + (1-\xi)^2\mathbf{Z}_2\mathbf{Z}_2^\top + \xi(1-\xi)\mathbf{Z}_1\mathbf{Z}_2^\top + \xi(1-\xi)\mathbf{Z}_2\mathbf{Z}_1^\top\} = \pi_1\text{E}(\mathbf{Z}_1\mathbf{Z}_1^\top) + \pi_2\text{E}(\mathbf{Z}_2\mathbf{Z}_2^\top).$$

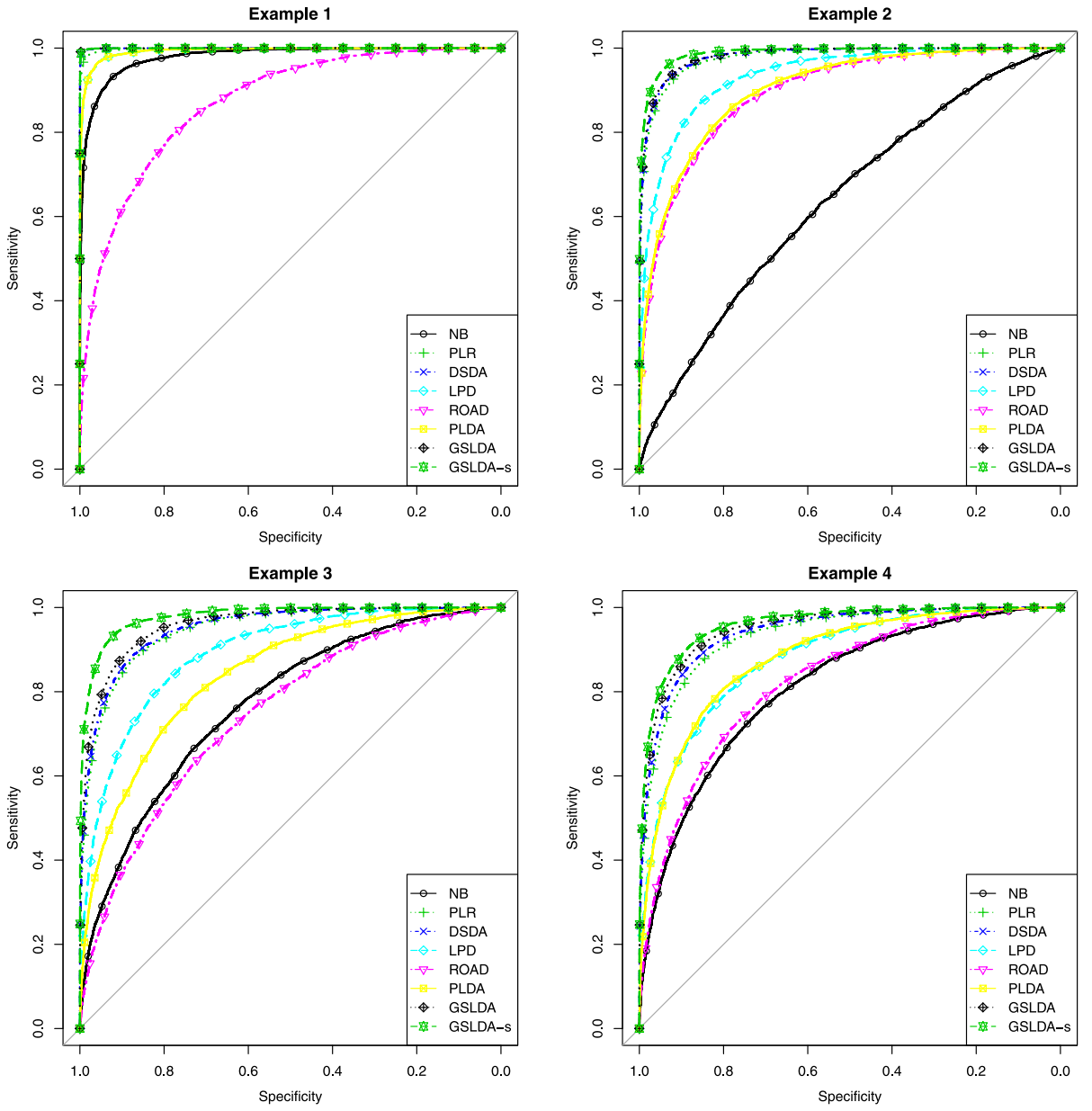

Fig. B.2. ROC Curve under the unbalanced setting for the four examples. In particular, the proportion of Class-0 sample is 80%. The ROC curve is computed based on 100 repetitions.

Thus the overall covariance matrix is $\text{var}(\mathbf{X}) = \boldsymbol{\Sigma} + \pi_1\pi_2\boldsymbol{\delta}\boldsymbol{\delta}^\top$, where $\boldsymbol{\delta} = \boldsymbol{\mu}^{(1)} - \boldsymbol{\mu}^{(2)}$.

Now we verify the inverse matrix of $\text{var}(\mathbf{X})$, i.e., the overall precision matrix of the mixture distribution. By setting $c = \pi_1\pi_2/(1 + \pi_1\pi_2\boldsymbol{\delta}^\top\boldsymbol{\Sigma}^{-1}\boldsymbol{\delta})$, we have

$$(\boldsymbol{\Sigma}^{-1} - c\boldsymbol{\Sigma}^{-1}\boldsymbol{\delta}\boldsymbol{\delta}^\top\boldsymbol{\Sigma}^{-1})(\boldsymbol{\Sigma} + \pi_1\pi_2\boldsymbol{\delta}\boldsymbol{\delta}^\top) = \mathbf{I} + \pi_1\pi_2\boldsymbol{\Sigma}^{-1}\boldsymbol{\delta}\boldsymbol{\delta}^\top - c\boldsymbol{\Sigma}^{-1}\boldsymbol{\delta}\boldsymbol{\delta}^\top - \pi_1\pi_2 c\boldsymbol{\Sigma}^{-1}\boldsymbol{\delta}\boldsymbol{\delta}^\top\boldsymbol{\Sigma}^{-1}\boldsymbol{\delta}\boldsymbol{\delta}^\top = \mathbf{I}.$$

Denote $\boldsymbol{\beta}^* = \boldsymbol{\Sigma}^{-1}\boldsymbol{\delta}$. Then we have $\text{var}(\mathbf{X})^{-1} = \boldsymbol{\Sigma}^{-1} - c\boldsymbol{\beta}^*\boldsymbol{\beta}^{*\top}$. □

C.2. Proof of Theorem 1

Before the proof, we introduce a lemma from [7]. The proof is omitted.

Lemma 4. *Let $\xi_1, \ldots, \xi_n$ be independent random variables with mean zero. Suppose that there exists some $t > 0$ and $\bar{B}_n$ such that $\sum_{k=1}^n \mathrm{E}(\xi_k^2 e^{t|\xi_k|}) \le \bar{B}_n^2$. Set $C_t = t + t^{-1}$. Then uniformly for $x \in (0, \bar{B}_n]$,*

$$\Pr\left(\sum_{k=1}^n \xi_k \ge C_t \bar{B}_n x\right) \le \exp(-x^2).$$

Denote $\xi_1 = \|\tilde{\mathbf{S}}_{AA} - \tilde{\mathbf{\Sigma}}_{AA}\|_\infty$, $\xi_2 = \|\tilde{\mathbf{S}}_{AA}^{-1} - \tilde{\mathbf{\Sigma}}_{AA}^{-1}\|_\infty$, and $\xi = \|\tilde{\mathbf{S}}_{A^\complement A}\tilde{\mathbf{S}}_{AA}^{-1} - \tilde{\mathbf{\Sigma}}_{A^\complement A}\tilde{\mathbf{\Sigma}}_{AA}^{-1}\|_\infty$. With simple calculations, one can show that for all $\boldsymbol{\beta} \in \mathbb{R}^p$,

$$\min_{\beta_0 \in \mathbb{R}} \sum_{i=1}^n (y_i - \beta_0 - \mathbf{x}_i^\top \boldsymbol{\beta})^2 = \sum_{i=1}^n (y_i - \dot{\mathbf{x}}_i^\top \boldsymbol{\beta})^2,$$

where $\dot{\mathbf{x}}_i = \mathbf{x}_i - (\mathbf{x}_1 + \cdots + \mathbf{x}_n)/n$ is the centralized feature vector. Thus the loss function of GSLDA in (4) is equivalent to $\sum_{i=1}^n (y_i - \dot{\mathbf{x}}_i^\top \boldsymbol{\beta})^2/n + \lambda \|\boldsymbol{\beta}\|_{\mathcal{G},\boldsymbol{\tau}}$. In the rest of our proof, we assume the sample $\mathbf{X}$ has been centralized. Then the GSLDA formulation (4) becomes

$$\hat{\boldsymbol{\beta}} = \operatorname*{argmin}_{\boldsymbol{\beta} \in \mathbb{R}^p} \|Y - \mathbf{X}\boldsymbol{\beta}\|_2^2/n + \lambda \|\boldsymbol{\beta}\|_{\mathcal{G},\boldsymbol{\tau}}. \tag{C.1}$$

Under the assumption (A1), we can define

$$\hat{\boldsymbol{\gamma}} = \operatorname*{argmin}_{\boldsymbol{\gamma} \in \mathbb{R}^s} \|Y - \mathbf{X}_A\boldsymbol{\gamma}\|_2^2/n + \lambda \|\boldsymbol{\gamma}\|_{\mathcal{G}_A,\boldsymbol{\tau}_A}, \tag{C.2}$$

where $\mathcal{G}_A$ denotes the subgraph of $\mathcal{G}$ corresponding to A. If we can show that (i) all elements of $\hat{\boldsymbol{\gamma}}$ are non-zero; and (ii) $\hat{\boldsymbol{\beta}}$ with $\hat{\boldsymbol{\beta}}_A = \hat{\boldsymbol{\gamma}}$ and $\hat{\boldsymbol{\beta}}_{A^\complement} = \mathbf{0}$ solves (C.1); then, GSLDA estimation recovers all significant features accurately.

We first show statement (i). By Section 4.6 of [31], the formulation (C.2) is equivalent to $\hat{\boldsymbol{\gamma}} = \sum_{j \in A} \hat{\mathbf{u}}^{(j)}$ where

$$\{\hat{\mathbf{u}}^{(j)} : j \in A\} = \operatorname*{argmin}_{\mathbf{u}^{(j)} \in \mathbb{R}^s : \mathrm{supp}(\mathbf{u}^{(j)}) \subseteq \mathcal{N}^{(j)} \cap A, j \in A} \|Y - \sum_{j \in A} \mathbf{X}_A \mathbf{u}^{(j)}\|_2^2/n + \lambda \sum_{j \in A} \tau_j \|\mathbf{u}^{(j)}\|_2.$$

Since this is a convex optimization problem, any solution $\{\mathbf{u}^{(j)} : j \in A\}$ satisfies the KKT conditions [4], which are for all $j \in A$, either

$$\mathbf{u}^{(j)} \ne \mathbf{0} \quad \text{and} \quad 2\mathbf{X}_{\mathcal{N}^{(j)}}^\top (Y - \mathbf{X}_A\boldsymbol{\gamma})/n = \lambda \tau_j \mathbf{u}^{(j)}_{\mathcal{N}^{(j)}}/\|\mathbf{u}^{(j)}\|_2,$$

or

$$\mathbf{u}^{(j)} = \mathbf{0} \quad \text{and} \quad 2\|\mathbf{X}_{\mathcal{N}^{(j)}}^\top (Y - \mathbf{X}_A\boldsymbol{\gamma})\|_2/n \le \lambda \tau_j,$$

where $\boldsymbol{\gamma} = \sum_{j \in A} \mathbf{u}^{(j)}$. Thus we have $\|\mathbf{X}_A^\top (Y - \mathbf{X}_A\hat{\boldsymbol{\gamma}})\|_\infty/n \le \lambda\tau^*/2$, and we can write $\hat{\boldsymbol{\gamma}}$ as $\hat{\boldsymbol{\gamma}} = \tilde{\mathbf{S}}_{AA}^{-1}(\hat{\boldsymbol{\delta}}_A + \lambda\tau^* \mathbf{t}_A/2)$, where $\mathbf{t}_A \in \mathbb{R}^s$ satisfies $\|\mathbf{t}_A\|_\infty \le 1$. We have

$$\begin{aligned}
\|\hat{\boldsymbol{\gamma}} - \boldsymbol{\beta}_A^\dagger\|_\infty &= \|(\tilde{\mathbf{S}}_{AA}^{-1} - \tilde{\mathbf{\Sigma}}_{AA}^{-1})\boldsymbol{\delta}_A - \tilde{\mathbf{S}}_{AA}^{-1}(\boldsymbol{\delta}_A - \hat{\boldsymbol{\delta}}_A) + \tilde{\mathbf{S}}_{AA}^{-1}\lambda\tau^* \mathbf{t}_A/2\|_\infty \\
&\le \|\boldsymbol{\delta}_A\|_\infty \xi_2 + (\varphi + \xi_2)\|\hat{\boldsymbol{\delta}}_A - \boldsymbol{\delta}_A\|_\infty + \lambda\tau^*(\varphi + \xi_2)/2 \\
&\le \xi_2(\|\boldsymbol{\delta}_A\|_\infty + \|\hat{\boldsymbol{\delta}}_A - \boldsymbol{\delta}_A\|_\infty + \lambda\tau^*/2) + \varphi(\|\hat{\boldsymbol{\delta}}_A - \boldsymbol{\delta}_A\|_\infty + \lambda\tau^*/2) \\
&\le \frac{\varphi^2 \xi_1}{1 - \varphi\xi_1}(\|\boldsymbol{\delta}_A\|_\infty + \|\hat{\boldsymbol{\delta}}_A - \boldsymbol{\delta}_A\|_\infty + \lambda\tau^*/2) + \varphi(\|\hat{\boldsymbol{\delta}}_A - \boldsymbol{\delta}_A\|_\infty + \lambda\tau^*/2) \equiv L_1,
\end{aligned}$$

in which the second inequality holds for sufficiently large n because $\varphi\xi_1 \le 1$ and $\xi_2 \le (1 - \varphi\xi_1)^{-1}\varphi^2\xi_1$. If $\xi_1 \le \epsilon$ and $\|\hat{\boldsymbol{\delta}}_A - \boldsymbol{\delta}_A\|_\infty \le \epsilon$, then $L_1 = O(\epsilon) + \lambda\tau^*\varphi/2 > 0$, which proves (i). By Lemma 4, the statement (i) is true with probability at least $1 - 2s^2 \exp(-a_1 n\epsilon^2/s^2) - 2s\exp(-a_2 n\epsilon^2)$, for some positive a_1 and a_2.

Now we prove statement (ii). The formulation (C.1) is equivalent to $\hat{\boldsymbol{\beta}} = \sum_{j=1}^p \hat{\mathbf{v}}^{(j)}$, where

$$\{\hat{\mathbf{v}}^{(1)}, \ldots, \hat{\mathbf{v}}^{(p)}\} = \operatorname*{argmin}_{\mathbf{v}^{(j)} : \mathrm{supp}(\mathbf{v}^{(j)}) \subseteq \mathcal{N}^{(j)},\ 1 \le j \le p} \frac{1}{n} \left\| Y - \sum_{j=1}^p \mathbf{X}\mathbf{v}^{(j)} \right\|_2^2 + \lambda \sum_{j=1}^p \|\mathbf{v}^{(j)}\|_2. \tag{C.3}$$

This is also a convex optimization problem and the KKT conditions of formulation (C.3) are for all $j \in \{1, \ldots, p\}$, either

$$\mathbf{v}^{(j)} \ne \mathbf{0} \quad \text{and} \quad 2\mathbf{X}_{\mathcal{N}^{(j)}}^\top (Y - \mathbf{X}\boldsymbol{\beta})/n = \lambda \tau_j \mathbf{v}^{(j)}_{\mathcal{N}^{(j)}}/\|\mathbf{v}^{(j)}\|_2, \tag{C.4}$$

or

$$\mathbf{v}^{(j)} = \mathbf{0} \quad \text{and} \quad 2\|\mathbf{X}_{\mathcal{N}^{(j)}}^\top (Y - \mathbf{X}\boldsymbol{\beta})\|_2/n \le \lambda \tau_j, \tag{C.5}$$

where $\boldsymbol{\beta} = \mathbf{v}^{(1)} + \cdots + \mathbf{v}^{(p)}$. Let $\mathbf{v}^{(j)} = 0$ for all $j \in A^\complement$, and $\mathbf{v}_A^{(j)} = \mathbf{u}^{(j)}$, $\mathbf{v}_{A^\complement}^{(j)} = \mathbf{0}$ for all $j \in A$. Then $\boldsymbol{\beta}_A = \hat{\boldsymbol{\gamma}}$ and $\boldsymbol{\beta}_{A^\complement} = \mathbf{0}$.

For $j \in A$, (C.4) holds owing to the definition of $\hat{\boldsymbol{\gamma}}$. For $j \in A^{\complement}$, $2\|\mathbf{X}_{\mathcal{N}^{(j)}}^{\top}(Y-\mathbf{X}\boldsymbol{\beta})\|_2/n \le 2\sqrt{|\mathcal{N}^{(j)}|}\|\mathbf{X}_{\mathcal{N}^{(j)}}^{\top}(Y-\mathbf{X}\boldsymbol{\beta})\|_\infty/n$. Denote $\eta = \|\hat{\boldsymbol{\delta}}-\boldsymbol{\delta}\|_\infty$. If $\|\hat{\boldsymbol{\delta}}-\boldsymbol{\delta}\|_\infty \le \epsilon$ and $\|\tilde{\mathbf{S}}_{A^{\complement}A} - \tilde{\boldsymbol{\Sigma}}_{A^{\complement}A}\|_\infty \le \epsilon$, then

$$\|\mathbf{X}_{A^{\complement}}^{\top}(Y-\mathbf{X}_A\hat{\boldsymbol{\gamma}})\|_\infty/n = \|\tilde{\mathbf{S}}_{A^{\complement}A}\tilde{\mathbf{S}}_{AA}^{-1}(\hat{\boldsymbol{\delta}}_A + \lambda\tau^*\mathbf{t}_A/2) - \hat{\boldsymbol{\delta}}_{A^{\complement}}\|_\infty \le (\|\boldsymbol{\delta}_A\|_\infty + 1 + \epsilon + \kappa + \lambda\tau^*/2)\epsilon + \lambda\tau^*\kappa/2 \le O(\epsilon) + \lambda\tau_*/2.$$

By Lemma 4, the statement (ii) is true with probability at least $1 - 2ps\exp(-a_1n\epsilon^2/s^2) - 2p\exp(-a_2n\epsilon^2)$. By taking $\epsilon = \sqrt{\ln p/n}$, the active set is recovered and $\|\hat{\boldsymbol{\beta}}_A - \boldsymbol{\beta}_A^{\dagger}\|_\infty \le O(\sqrt{\ln p/n})$ with probability at least $1 - 2s^2\exp(-a_1n\epsilon^2/s^2) - 2ps\exp(-a_1n\epsilon^2/s^2) - 2p\exp(-a_2n\epsilon^2) = 1 - O(p^{-C_1})$ for some $C_1 > 0$. □

C.3. Proof of Theorem 2

The proof uses the following lemma from [30].

Lemma 5. *Denote $\mathcal{M}$ a subspace of $\mathbb{R}^p$ and $\mathcal{M}^{\perp}$ its orthogonal complement. For a regularized estimation problem*

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^p} L(\boldsymbol{\theta}) + \lambda R(\boldsymbol{\theta}),$$

where

(i) R is a norm and is decomposable with respect to $(\mathcal{M}, \mathcal{M}^{\perp})$, i.e., $R(\boldsymbol{\theta}+\boldsymbol{\eta}) = R(\boldsymbol{\theta}) + R(\boldsymbol{\eta})$ for all $\boldsymbol{\theta} \in \mathcal{M}$, $\boldsymbol{\eta} \in \mathcal{M}^{\perp}$;
(ii) L is convex and differentiable, and satisfies restricted strong convex condition with curvature κ_L, i.e., $\delta L(\boldsymbol{\theta}^*, \boldsymbol{\Delta}) = L(\boldsymbol{\theta}^* + \boldsymbol{\Delta}) - L(\boldsymbol{\theta}^*) - \nabla L(\boldsymbol{\theta}^*)^{\top}\boldsymbol{\Delta} \ge \kappa_L\|\boldsymbol{\Delta}\|_2^2$ for some $\boldsymbol{\theta}^*$, for all $\boldsymbol{\Delta}$ such that $R(\boldsymbol{\Delta}_{\mathcal{M}^{\perp}}) \le 3R(\boldsymbol{\Delta}_{\mathcal{M}}) + 4R(\boldsymbol{\theta}^*_{\mathcal{M}^{\perp}})$.

Let $\lambda \ge 2R^*\{\nabla L(\boldsymbol{\theta}^*)\}$, where R^* denotes the dual norm of R, then any solution $\hat{\boldsymbol{\theta}}$ to the problem satisfies

$$\|\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\|_2 \le 9\lambda^2\psi(\mathcal{M})/\kappa_L^2 + 4\lambda R^*(\boldsymbol{\theta}^*_{\mathcal{M}^{\perp}})/\kappa_L,$$

where $\psi(\mathcal{M}) = \sup_{\mathbf{u}\in\mathcal{M}/\{0\}} R(\mathbf{u})/\|\mathbf{u}\|_2$.

Proof. In the GSLDA formulation, the loss function is $L(\beta_0, \boldsymbol{\beta}) = \|\mathbf{y} - \beta_0\mathbf{1} - \mathbf{X}\boldsymbol{\beta}\|_2^2/n$, and the regularization is $R(\boldsymbol{\beta}) = \|\boldsymbol{\beta}\|_{\mathcal{G},\boldsymbol{\tau}}$. It has been shown in [31] that R is a norm and its dual norm is $R^*(\mathbf{u}) = \max_{1\le j\le p}\tau_j^{-1}\|\mathbf{u}_{\mathcal{N}^{(j)}}\|_2$. When we take $\tau_j = \sqrt{|\mathcal{N}^{(j)}|}$, $R^*(\mathbf{u}) \le \max_j\|\mathbf{u}_{\mathcal{N}^{(j)}}\|_\infty = \|\mathbf{u}\|_\infty$.

For some $\epsilon > 0$, denote the event $\chi = \{\|\hat{\boldsymbol{\delta}}-\boldsymbol{\delta}\|_\infty \le \epsilon, \|\tilde{\mathbf{S}}_{\cdot,A} - \tilde{\boldsymbol{\Sigma}}_{\cdot,A}\|_\infty \le \epsilon\}$. Then by Lemma 4, $\Pr(\chi) \ge 1 - 2p\exp(-a_2n\epsilon^2) - 2ps\exp(-a_1n\epsilon^2)$. Under the event χ,

$$\begin{aligned}\|\nabla L(\boldsymbol{\beta}^{\dagger})\|_\infty &= \|2n^{-1}\mathbf{X}^{\top}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}^{\dagger})\|_\infty \\ &= \|2(\hat{\boldsymbol{\delta}} - \tilde{\mathbf{S}}\boldsymbol{\beta}^{\dagger})\|_\infty = \|2(\hat{\boldsymbol{\delta}} - \boldsymbol{\delta}) - 2(\tilde{\mathbf{S}} - \tilde{\boldsymbol{\Sigma}})\boldsymbol{\beta}^{\dagger}\|_\infty \le 2\|\hat{\boldsymbol{\delta}} - \boldsymbol{\delta}\|_\infty + 2\|(\tilde{\mathbf{S}} - \tilde{\boldsymbol{\Sigma}})_{\cdot,A}\boldsymbol{\beta}_A^{\dagger}\|_\infty \\ &\le 2\|\hat{\boldsymbol{\delta}} - \boldsymbol{\delta}\|_\infty + 2\|(\tilde{\mathbf{S}} - \tilde{\boldsymbol{\Sigma}})_{\cdot,A}\|_\infty\|\boldsymbol{\beta}_A^{\dagger}\|_1 \le 2\epsilon + 2\|\boldsymbol{\beta}_A^{\dagger}\|_1\epsilon.\end{aligned}$$

We take $\epsilon = C_2\sqrt{\ln p/n}$ where $C_2 > (a_1 \wedge a_2)^{-1}$. Then $\lambda \ge 2R^*(\nabla L(\boldsymbol{\beta}^{\dagger}))$. Under the event χ with $\epsilon \le c\sigma$ for some $c > 0$, for sufficiently large n, we have $\boldsymbol{\Delta}^{\top}\tilde{\mathbf{S}}\boldsymbol{\Delta} \ge \boldsymbol{\Delta}^{\top}\tilde{\boldsymbol{\Sigma}}\boldsymbol{\Delta} - |\boldsymbol{\Delta}^{\top}(\tilde{\mathbf{S}} - \tilde{\boldsymbol{\Sigma}})\boldsymbol{\Delta}| \ge \sigma\|\boldsymbol{\Delta}\|_2^2/2$ for $\boldsymbol{\Delta} \in \mathcal{C}(A)$. Thus $\delta L(\boldsymbol{\beta}^*, \boldsymbol{\Delta}) \ge 2\boldsymbol{\Delta}^{\top}\tilde{\mathbf{S}}\boldsymbol{\Delta} \ge \sigma\|\boldsymbol{\Delta}\|_2^2$ for all $\boldsymbol{\Delta} \in \mathcal{C}(A)$.

We take $\mathcal{M} = \{\boldsymbol{\beta} \in \mathbb{R}^p : \boldsymbol{\beta}_{A^{\complement}} = \mathbf{0}\}$. Then $\boldsymbol{\beta}^{\dagger} \in \mathcal{M}$ and $\boldsymbol{\beta}^{\dagger}_{\mathcal{M}^{\perp}} = \mathbf{0}$. Moreover,

$$\psi(\mathcal{M}) = \sup_{\boldsymbol{\beta}\in\mathcal{M}}\frac{R(\boldsymbol{\beta})}{\|\boldsymbol{\beta}\|_2} = \sup_{\boldsymbol{\beta}_{A^{\complement}}=\mathbf{0}}\frac{\min_{\sum\mathbf{v}^{(j)}=\boldsymbol{\beta}}\sum\tau_j\|\mathbf{v}^{(j)}\|_2}{\|\boldsymbol{\beta}\|_2} \le \sup_{\boldsymbol{\beta}_{A^{\complement}}=\mathbf{0}}\frac{\sum_{j\in A}\tilde{\tau}_j|\beta_j|}{\|\boldsymbol{\beta}_A\|_2} \le \tau^*\sqrt{s}\,.$$

Therefore, by Lemma 5, we have

$$\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^{\dagger}\|_2^2 \le 9\lambda^2 s\,\tau^{*2}/\sigma^2,$$

with probability at least $1 - 2ps\exp(-a_1n\epsilon^2) - 2p\exp(-a_2n\epsilon^2) \ge 1 - sp^{-C_3}$ where $C_3 = C_2(a_1 \vee a_2) - 1 > 0$. □

C.4. Proof of Theorem 3

We use the same notations as in the proof above. Without loss of generality, we assume $\boldsymbol{\mu}^{(1)} + \boldsymbol{\mu}^{(2)} = \mathbf{0}$, then $\boldsymbol{\mu}^{(1)} = \boldsymbol{\delta}/2$, $\boldsymbol{\mu}^{(2)} = -\boldsymbol{\delta}/2$, and $\beta_0^{\dagger} = 0$. According to Lemma 3, we have

$$\begin{aligned}Q(\hat{\beta}_0, \hat{\boldsymbol{\beta}})/Q(\beta_0^{\dagger}, \boldsymbol{\beta}^{\dagger}) - 1 = \Bigg[&\left\{\Phi\left(\frac{-\hat{\beta}_0 - \hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\mu}^{(1)}}{\sqrt{\hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\Sigma}\hat{\boldsymbol{\beta}}}}\right) - \Phi\left(\frac{-\boldsymbol{\beta}^{\dagger T}\boldsymbol{\mu}^{(1)}}{\sqrt{\boldsymbol{\beta}^{\dagger T}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger}}}\right)\right\} \\ &+ \left\{\Phi\left(\frac{\hat{\beta}_0 + \hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\mu}^{(2)}}{\sqrt{\hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\Sigma}\hat{\boldsymbol{\beta}}}}\right) - \Phi\left(\frac{\boldsymbol{\beta}^{\dagger T}\boldsymbol{\mu}^{(2)}}{\sqrt{\boldsymbol{\beta}^{\dagger T}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger}}}\right)\right\}\Bigg]\end{aligned}$$

$$\times \left\{ \Phi\left(\frac{-\boldsymbol{\beta}^{\dagger T}\boldsymbol{\mu}^{(1)}}{\sqrt{\boldsymbol{\beta}^{\dagger T}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger}}}\right) + \Phi\left(\frac{\boldsymbol{\beta}^{\dagger T}\boldsymbol{\mu}^{(2)}}{\sqrt{\boldsymbol{\beta}^{\dagger T}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger}}}\right)\right\}^{-1}$$

$$\leq \max\left\{ \Phi\left(\frac{-\hat{\beta}_0 - \hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\delta}/2}{\sqrt{\hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\Sigma}\hat{\boldsymbol{\beta}}}}\right) \Big/ \Phi\left(\frac{-\boldsymbol{\beta}^{\dagger T}\boldsymbol{\delta}/2}{\sqrt{\boldsymbol{\beta}^{\dagger T}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger}}}\right) - 1,\ \Phi\left(\frac{\hat{\beta}_0 - \hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\delta}/2}{\sqrt{\hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\Sigma}\hat{\boldsymbol{\beta}}}}\right) \Big/ \Phi\left(\frac{-\boldsymbol{\beta}^{\dagger T}\boldsymbol{\delta}/2}{\sqrt{\boldsymbol{\beta}^{\dagger T}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger}}}\right) - 1 \right\}$$

$$\equiv \max\{R^{(1)}, R^{(2)}\}.$$

We will use the following property of standard Gaussian distribution function [6]:

$$|\Phi(x_0 + r)/\Phi(x_0) - 1| \leq c_1|r|(|x_0| + 1)\exp(c_2|x_0 r|). \tag{C.6}$$

For $k \in \{1, 2\}$, let

$$r^{(k)} = \left|(\hat{\beta}_0 + \hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\mu}^{(k)})/\sqrt{\hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\Sigma}\hat{\boldsymbol{\beta}}} - \boldsymbol{\beta}^{\dagger T}\boldsymbol{\mu}^{(k)}/\sqrt{\boldsymbol{\beta}^{\dagger T}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger}}\right|.$$

Since $\boldsymbol{\beta}^{\dagger T}\boldsymbol{\delta}/\sqrt{\boldsymbol{\beta}^{\dagger T}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger}} = \Delta^{1/2}$, it suffices to verify the orders of $r^{(k)}$ and Δ.

According to Theorem 2, $\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^{\dagger}\|_2 \leq 3\lambda\tau^*\sqrt{s}/\sigma$ with probability going to 1. Moreover, by the definition of $\boldsymbol{\beta}^{\dagger}$, we have $\boldsymbol{\beta}^{\dagger} = 4/(4 + \Delta)\boldsymbol{\beta}^*$, and $\boldsymbol{\beta}^{\dagger T}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger} = 16\Delta/(4 + \Delta)^2$. Since

$$\begin{aligned}
|\hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\Sigma}\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^{\dagger T}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger}| &\leq |(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^{\dagger})^{\top}\boldsymbol{\Sigma}(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^{\dagger})| + 2|(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^{\dagger})^{\top}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger}| \\
&\leq \lambda_{\max}(\boldsymbol{\Sigma})\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^{\dagger}\|_2^2 + 2\lambda_{\max}(\boldsymbol{\Sigma})\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^{\dagger}\|_2\|\boldsymbol{\beta}^{\dagger}\|_2 \leq 3\lambda_{\max}(\boldsymbol{\Sigma})\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^{\dagger}\|_2\|\boldsymbol{\beta}^{\dagger}\|_2, \\
&\leq 9\lambda_{\max}(\boldsymbol{\Sigma})\lambda\tau^*\sqrt{s}\,\|\boldsymbol{\beta}^{\dagger}\|_2/\sigma,
\end{aligned}$$

for sufficiently large n, we have

$$\begin{aligned}
|(\hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\Sigma}\hat{\boldsymbol{\beta}})^{-1/2} - (\boldsymbol{\beta}^{\dagger T}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger})^{-1/2}| &= \left|\frac{(\hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\Sigma}\hat{\boldsymbol{\beta}})^{1/2} - (\boldsymbol{\beta}^{\dagger T}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger})^{1/2}}{(\hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\Sigma}\hat{\boldsymbol{\beta}})^{1/2}(\boldsymbol{\beta}^{\dagger T}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger})^{1/2}}\right| \\
&= \left|\frac{\hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\Sigma}\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^{\dagger T}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger}}{(\hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\Sigma}\hat{\boldsymbol{\beta}})^{1/2}\cdot(\boldsymbol{\beta}^{\dagger T}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger})^{1/2}[(\hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\Sigma}\hat{\boldsymbol{\beta}})^{1/2} + (\boldsymbol{\beta}^{\dagger T}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger})^{1/2}]}\right| \\
&\leq \left|\frac{\hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\Sigma}\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^{\dagger T}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger}}{3/4(\boldsymbol{\beta}^{\dagger T}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger})^{3/2}}\right| \leq \frac{(4 + \Delta)^3}{4\Delta^{3/2}}\lambda_{\max}(\boldsymbol{\Sigma})\lambda\tau^*\sqrt{s}\,\|\boldsymbol{\beta}^{\dagger}\|_2/\sigma,
\end{aligned}$$

in which the first inequality holds because $\hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\Sigma}\hat{\boldsymbol{\beta}} \geq \boldsymbol{\beta}^{\dagger T}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger}/2$ for sufficiently large n. Moreover, under χ we have,

$$\begin{aligned}
|(\hat{\beta}_0 + \hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\delta}/2) - (\boldsymbol{\beta}^{\dagger T}\boldsymbol{\delta}/2)| &\leq |(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^{\dagger})^{\top}\boldsymbol{\delta}|/2 + |-\bar{\mathbf{x}}^{\top}\hat{\boldsymbol{\beta}}| \\
&\leq \|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^{\dagger}\|_2\|\boldsymbol{\delta}\|_2/2 + |\bar{\mathbf{x}}^{\top}\boldsymbol{\beta}^{\dagger}| + |\bar{\mathbf{x}}^{\top}(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^{\dagger})| \leq 3\lambda\tau^*\sqrt{s}\,\|\boldsymbol{\delta}\|_2/(2\sigma) + \|\boldsymbol{\beta}^{\dagger}\|_1\epsilon.
\end{aligned}$$

Therefore,

$$\begin{aligned}
r^{(1)} &\leq |(\hat{\beta}_0 + \hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\delta}/2)\{(\hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\Sigma}\hat{\boldsymbol{\beta}})^{-1/2} - (\boldsymbol{\beta}^{\dagger T}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger})^{-1/2}\}| + |(\hat{\beta}_0 + \hat{\boldsymbol{\beta}}^{\top}\boldsymbol{\delta}/2) - \boldsymbol{\beta}^{\dagger T}\boldsymbol{\delta}/2|/(\boldsymbol{\beta}^{\dagger T}\boldsymbol{\Sigma}\boldsymbol{\beta}^{\dagger})^{1/2} \\
&\leq \frac{(4 + \Delta)^2}{2\Delta^{1/2}}\lambda\tau^*\lambda_{\max}(\boldsymbol{\Sigma})\sqrt{s}\,\|\boldsymbol{\beta}^{\dagger}\|_2/\sigma + \frac{4 + \Delta}{4\Delta^{1/2}}\{2\lambda\tau^*\sqrt{s}\,\|\boldsymbol{\delta}\|_2/\sigma + \|\boldsymbol{\beta}^{\dagger}\|_1\epsilon\}.
\end{aligned}$$

Using the property (C.6), then we have

$$R^{(1)} = \left|\frac{\Phi(-\Delta^{1/2}/2 + r^{(1)})}{\Phi(-\Delta^{1/2}/2)} - 1\right| \leq c_1|r^{(1)}|(\Delta^{1/2}/2 + 1)\exp(c_2 r^{(1)}\Delta^{1/2}/2) = O\{r^{(1)}\Delta^{1/2}\exp(c_2 r^{(1)}\Delta^{1/2}/2)\}.$$

Since $\Delta^2\lambda\tau^*\lambda_{\max}(\boldsymbol{\Sigma})\sqrt{s}\,\|\boldsymbol{\beta}^{\dagger}\|_2/\sigma \to 0$, $\Delta\lambda\tau^*\sqrt{s}\,\|\boldsymbol{\delta}\|_2/\sigma \to 0$, and $\Delta n^{\gamma-1}\|\boldsymbol{\beta}^{\dagger}\|_1 \to 0$, we have $r^{(1)}\Delta^{1/2} \to 0$ and thus $R^{(1)} \xrightarrow{p} 0$. Similarly, we can show $R^{(2)} \xrightarrow{p} 0$, which proves the theorem.

References

[1] P.J. Bickel, E. Levina, Some theory for Fisher's linear discriminant function, 'naive Bayes', and some alternatives when there are many more variables than observations, Bernoulli 10 (6) (2004) 989–1010.

[2] C.M. Bishop, Pattern Recognition and Machine Learning (Information Science and Statistics), Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[3] H.D. Bondell, B.J. Reich, Simultaneous regression shrinkage, variable selection, and supervised clustering of predictors with OSCAR, Biometrics 64 (1) (2008) 115–123.
[4] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge university press, 2004.
[5] D. Cai, X. He, J. Han, Semi-supervised discriminant analysis, in: 2007 IEEE 11th International Conference on Computer Vision, IEEE, 2007, pp. 1–7.
[6] T. Cai, W. Liu, A direct estimation approach to sparse linear discriminant analysis, J. Amer. Statist. Assoc. 106 (496) (2011) 1566–1577.
[7] T. Cai, W. Liu, X. Luo, A constrained $\ell 1$ minimization approach to sparse precision matrix estimation, J. Amer. Statist. Assoc. 106 (494) (2011) 594–607.
[8] J. Chen, Z. Chen, Extended Bayesian information criteria for model selection with large model spaces, Biometrika 95 (3) (2008) 759–771.
[9] S. Chen, D.M. Witten, A. Shojaie, Selection and estimation for mixed graphical models, Biometrika 102 (1) (2014) 47–64.
[10] L. Clemmensen, T. Hastie, D. Witten, B. Ersbøll, Sparse discriminant analysis, Technometrics 53 (4) (2011) 406–413.
[11] J. Fan, Y. Fan, High dimensional classification using features annealed independence rules, Ann. Statist. 36 (6) (2008) 2605.
[12] J. Fan, Y. Feng, X. Tong, A road to classification in high dimensional space: the regularized optimal affine discriminant, J. R. Stat. Soc. Ser. B Stat. Methodol. 74 (4) (2012) 745–771.
[13] R.A. Fisher, The use of multiple measurements in taxonomic problems, Ann. Eugenics 7 (2) (1936) 179–188.
[14] J. Friedman, T. Hastie, R. Tibshirani, Sparse inverse covariance estimation with the graphical lasso, Biostatistics 9 (3) (2008) 432–441.
[15] D.J. Hand, Classifier technology and the illusion of progress, Stat. Sci. 21 (1) (2006) 1–14.
[16] T. Hastie, R. Tibshirani, A. Buja, Flexible discriminant analysis by optimal scoring, J. Amer. Statist. Assoc. 89 (428) (1994) 1255–1270.
[17] T. Hastie, R. Tibshirani, J. Friedman, Elements of Statistical Learning : Data Mining, Inference, and Prediction, second ed., Springer New York, New York, NY, 2009.
[18] S. Kim, W. Pan, X. Shen, Network-based penalized regression with application to genomic data, Biometrics 69 (3) (2013) 582–593.
[19] C. Li, H. Li, Network-constrained regularization and variable selection for analysis of genomic data, Bioinformatics 24 (9) (2008) 1175–1182.
[20] B. Liu, X. Shen, W. Pan, Semi-supervised spectral clustering with application to detect population stratification, Front. Genetics 4 (2013) 215.
[21] Y. Liu, M. Yuan, Reinforced multicategory support vector machines, J. Comput. Graph. Statist. 20 (4) (2011) 901–919.
[22] S. Luo, Z. Chen, Edge detection in sparse Gaussian graphical models, Comput. Statist. Data Anal. 70 (2014) 138–152.
[23] S. Luo, Z. Chen, Sequential Lasso cum EBIC for feature selection with ultra-high dimensional feature space, J. Amer. Statist. Assoc. 109 (507) (2014) 1229–1240.
[24] Q. Mai, Y. Yang, H. Zou, Multiclass sparse discriminant analysis, 2015, arXiv preprint arXiv:1504.05845.
[25] Q. Mai, H. Zou, A note on the connection and equivalence of three sparse linear discriminant analysis methods, Technometrics 55 (2) (2013) 243–246.
[26] Q. Mai, H. Zou, M. Yuan, A direct approach to sparse discriminant analysis in ultra-high dimensions, Biometrika 99 (1) (2012) 29–42.
[27] L. Meier, S. Van De Geer, P. Bühlmann, The group lasso for logistic regression, J. R. Stat. Soc. Ser. B Stat. Methodol. 70 (1) (2008) 53–71.
[28] N. Meinshausen, P. Bühlmann, High-Dimensional Graphs and Variable Selection with the Lasso, Ann. Statist. 34 (3) (2006) 1436–1462.
[29] W. Min, J. Liu, S. Zhang, Network-regularized sparse logistic regression models for clinical risk prediction and biomarker discovery, IEEE/ACM Trans. Comput. Biol. Bioinform. (TCBB) 15 (3) (2018) 944–953.
[30] S.N. Negahban, P. Ravikumar, M.J. Wainwright, B. Yu, A Unified Framework for High-Dimensional Analysis of M-Estimators with Decomposable Regularizers, Statist. Sci. 27 (4) (2012;2010;) 538–557.
[31] G. Obozinski, L. Jacob, J.-P. Vert, Group lasso with overlaps: the latent group lasso approach, 2011, arXiv preprint arXiv:1110.0413.
[32] W. Pan, X. Shen, Penalized model-based clustering with application to variable selection, J. Mach. Learn. Res. 8 (2007) 1145–1164.
[33] W. Pan, B. Xie, X. Shen, Incorporating predictor network in penalized regression with application to microarray data, Biometrics 66 (2) (2010) 474–484.
[34] H. Pang, H. Liu, R. Vanderbei, The fastclime package for linear programming and large-scale precision matrix estimation in r, J. Mach. Learn. Res. 15 (1) (2014) 489–493.
[35] J. Shao, Y. Wang, X. Deng, S. Wang, Sparse linear discriminant analysis by thresholding for high dimensional data, Ann. Statist. 39 (2) (2011) 1241–1265.
[36] R. Tibshirani, T. Hastie, B. Narasimhan, G. Chu, Diagnosis of multiple cancer types by shrunken centroids of gene expression, Proc. Natl. Acad. Sci. 99 (10) (2002) 6567–6572.
[37] R.J. Vanderbei, Linear Programming: Foundations and Extensions, fourth ed., Springer, 2015.
[38] A. Voorman, A. Shojaie, D. Witten, Graph estimation with joint additive models, Biometrika 101 (1) (2013) 85–101.
[39] D.M. Witten, R. Tibshirani, Penalized classification using Fisher's linear discriminant, J. R. Stat. Soc. Ser. B Stat. Methodol. 73 (5) (2011) 753–772.
[40] M.C. Wu, L. Zhang, Z. Wang, D.C. Christiani, X. Lin, Sparse linear discriminant analysis for simultaneous testing for the significance of a gene set/pathway and gene selection, Bioinformatics 25 (9) (2009) 1145–1151.
[41] M. Wu, L. Zhu, X. Feng, et al., Network-based feature screening with applications to genome data, Ann. Appl. Stat. 12 (2) (2018) 1250–1270.
[42] S. Yang, L. Yuan, Y.-C. Lai, X. Shen, P. Wonka, J. Ye, Feature Grouping and Selection Over an Undirected Graph, ACM, 2012, pp. 922–930.
[43] Y. Yang, H. Zou, A fast unified algorithm for solving group-lasso penalize learning problems, Stat. Comput. 25 (6) (2015) 1129–1141.
[44] G. Yu, Y. Liu, Sparse Regression Incorporating Graphical Structure Among Predictors, J. Amer. Statist. Assoc. 111 (514) (2016) 707–720.
[45] M. Yuan, Y. Lin, Model selection and estimation in regression with grouped variables, J. R. Stat. Soc. Ser. B Stat. Methodol. 68 (1) (2006) 49–67.
[46] M. Yuan, Y. Lin, Model selection and estimation in the Gaussian graphical model, Biometrika 94 (1) (2007) 19–35.
[47] C. Zhang, Y. Liu, Multicategory large-margin unified machines, J. Mach. Learn. Res. 14 (1) (2013) 1349–1386.
[48] C. Zhang, Y. Liu, J. Wang, H. Zhu, Reinforced angle-based multicategory support vector machines, J. Comput. Graph. Statist. 25 (3) (2016) 806–825.
[49] W. Zhang, Y.-W. Wan, G.I. Allen, K. Pang, M.L. Anderson, Z. Liu, Molecular pathway identification using biological network-regularized logistic models, BMC Genomics 14 (8) (2013) S7.
[50] S. Zhao, A. Shojaie, A significance test for graph-constrained estimation, Biometrics 72 (2) (2016) 484–493.
[51] P. Zhao, B. Yu, On model selection consistency of Lasso, J. Mach. Learn. Res. 7 (Nov) (2006) 2541–2563.
[52] H. Zhou, W. Pan, X. Shen, Penalized model-based clustering with unconstrained covariance matrices, Electron. J. Statist. 3 (2009) 1473.
[53] Y. Zhu, X. Shen, W. Pan, Simultaneous grouping pursuit and feature selection over an undirected graph, J. Amer. Statist. Assoc. 108 (502) (2013) 713–725.