



Generalising combinatorial discriminant analysis through conditioning truncated Rayleigh flow

Sijia Yang¹ · Haoyi Xiong² · Di Hu³ · Kaibo Xu⁴ · Licheng Wang¹ · Peizhen Zhu⁵ · Zeyi Sun⁴ 

Received: 28 October 2020 / Revised: 18 June 2021 / Accepted: 19 June 2021

/ Published online: 9 July 2021

© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2021

Abstract

Fisher's Linear Discriminant Analysis (LDA) has been widely used for linear classification, feature selection, and metrics learning in multivariate data analytics. To ensure high classification accuracy while optimally discovering predictive features from the data, this paper studied **CDA**, namely **C**ombinatorial **D**iscriminant Analysis that intends to combinatorially select a subset of features and assign weights to them optimally. **CDA** extends the *Truncated Rayleigh Flow* algorithm (Tan et al. in J R Stat Soc: Ser B (Stat Methodol) 80(5):1057–1086, 2018) and improves LDA estimation under k -sparsity constraint. The experimental results based on the synthesized and real-world datasets demonstrate that our algorithm outperforms other LDA baselines and downstream classifiers. The empirical analysis shows that our algorithm can recover the combinatorial structure of optimal LDA with empirical consistency.

1 Introduction

The Fisher's Linear Discriminant Analysis (LDA) [2,3] is a well-known technique for dimension reduction and metric learning [4,5]. It has been widely used in many applications [6] such as face recognition, image retrieval, and etc. Typically, LDA finds the projection directions such that for the projected data, the between-class variance is maximized relative to the within-class variance, thus achieving the optimal discrimination. An intrinsic limitation of LDA is that its decision function in Eq. (1) relies on a *well-estimated* projection vector (a.k.a β) for (optimal) linear classification.

For example, let \mathbf{X} and \mathbf{Y} be two p -dimensional random vectors following two Gaussian distributions, $\mathbf{X} \sim \mathcal{N}(\mu_+^*, \Sigma^*)$ and $\mathbf{Y} \sim \mathcal{N}(\mu_-^*, \Sigma^*)$, which is with the same covariance matrix Σ^* but with distinct mean vectors μ_+^* and μ_-^* . For a new observation Z that is drawn from the two Gaussian distributions with equal prior probabilities, the Fisher's linear discriminant analysis classifies z using the rule

✉ Licheng Wang
wanglc@bupt.edu.cn

✉ Zeyi Sun
sunzeyi@mininglamp.com

Extended author information available on the last page of the article

$$\tilde{f}(z) = \text{sign} \left((z - \mu)^\top \beta \right), \quad (1)$$

where $\text{sign}(\cdot) \in \{-1, +1\}$, the optimal estimation of global mean μ is $\mu^* = (\mu_+^* + \mu_-^*)/2$, the optimal estimation of projection vector β should be $\beta^* = \Theta^*(\mu_+^* - \mu_-^*)$ and $\Theta^* = \Sigma^{*-1}$. Note that above classification rule is optimal, if the data of two classes are drawn from two Gaussian distributions with the same covariance matrix but different means.

Though the optimal classification rule exists under the Gaussian distribution assumptions, the estimation of β using the given training data sets is somehow difficult. A straightforward solution is to use sample estimators to first calculate sample covariance matrix and sample mean vectors separately, then estimate β composately via matrix inverse. However, For many applications, such as the micro-array data analysis and biomedical data analysis, the number of dimensions (denoted as p) is significantly larger than the number of samples (denoted as n), i.e., $p \gg n$. In this case, the sample estimation of the covariance matrix is usually singular and non-invertible. Hence, the estimation of linear projection vector β is ill-posed under high dimension but low sample size settings (HDLSS) [7].

Our contributions. Inspired by the recent progress in combinatorial regression [8], this work studies a novel **combinatorial** Fisher's discriminant analysis problem incorporating new **sparsity** assumptions. The existing work [9–13] makes specific sparsity assumptions on β or Σ^{-1} , then estimates sparse LDA using either ℓ_1 -norm sparse inverse covariance estimator or direct ℓ_1 -norm sparse LDA estimators. Our work however assumes the true LDA projection vector β^* is k -sparse [14], and proposes a new algorithm **CDA** for ℓ_0 -norm sparse LDA estimation with improved *covariance matrix estimation* under mild covariance structure conditions.

More specific, we derive **CDA** algorithm from the Truncated Rayleigh Flow (TRifle) Optimization Framework [1], which was originally designed to solve generalized eigenvalue problems under k -sparse constraints. To enhance TRifle, we (1) precondition TRifle with a better initialization to reduce the optimization error, (2) replace sample covariance matrix estimators with the shrunken one to marginalize the statistical error, and (3) correct the direction of gradient ascent with de-biased estimator [15,16] for faster statistical convergence. The enhanced optimization process is named as *Conditioned Rayleigh Flow*. We evaluate **CDA** using both a synthesized dataset and real-world datasets. The experimental results using the synthesized dataset show that **CDA** outperforms common combinatorial linear models, including Truncated Rayleigh Flow (TRifle) [1] and Orthogonal Matching Pursuit (OMP) [17], for combinatorial structure recovery (retrieving nonzero elements in β), with higher precision, recall and F1-score.

2 Backgrounds and related work

Recently, many efforts have been devoted to bear on such HDLSS problem using shrunken estimators [6,9–13,18,19] Generally, these studies assume that the data for training and classification is randomly drawn from two (unknown) Gaussian distributions $\mathcal{N}(\mu_+, \Sigma)$ and $\mathcal{N}(\mu_-, \Sigma)$ with certain priors. Given n training samples, LDA needs to estimate the linear projection vector β , which is structured as $\beta = \Sigma^{-1}(\mu_+ - \mu_-)$ [3]. However, when the number of dimensions (denoted as p) is significantly larger than the number of samples (denoted as n), i.e., $p \gg n$, the sample estimation of the covariance matrix Σ is usually singular and non-invertible. In this case, it is impossible to use the *inverse of sample covariance matrix*

to estimate β . Related work which intends to solve this problem can be categorized into two folders as follow.

Covariance-regularized Estimation To estimate β , these studies [6,9,18] proposed to first estimate the covariance matrix Σ (or the inverse of covariance matrix Σ^{-1}) using shrunk covariance/precision matrix estimators, such as the empirical shrinkage estimator or Graphical Lasso [20], to obtain an estimation of inverse covariance matrix (frequently denoted as Θ), and then multiply it with the estimated mean vectors to compute β in an indirect manner. This type of estimators used to be not favored by the community, as the estimation of inverse covariance matrices is more computational expensive [10] while relying on additional assumptions to the sparsity of covariance matrices.

Direct Sparse LDA Estimation Instead of estimating Σ and μ_+, μ_- separately, this line of research [10–13,19] proposed to estimate β directly from the data (sometimes, with sample estimators [10,11]). Specifically, [11] studied the algorithm to estimate β through maximizing fisher's information; [12] used thresholding technique to pursue a sparse representation of β ; [10] leveraged Dantzig Selector to estimate a sparse approximation of β under ℓ_1 -norm sparsity; most recently, [13] studied using Lasso-like estimator to compute the sparse approximation of β with ℓ_1 -norm sparsity and least-square error [19].

3 Preliminaries and problem formulation

In this section, we first review the state of the art of LDA, then formulate the problem of the proposed research.

3.1 Sample LDA for binary classification

With the labeled data pairs $(x_1, y_1) \cdots (x_n, y_n)$ (where $x_i \in \mathbb{R}^p$ is a p -dimensional vector and $y_i \in \{-1, +1\}$ with $1 \leq i \leq n$) as inputs, LDA first estimates the sample covariance matrix $\bar{\Sigma}$ using the pooled sample covariance matrix estimator with respect to the two classes [3], as well as the mean vectors $\bar{\mu}_+$ and $\bar{\mu}_-$ for the two classes. Given all estimated parameters $\bar{\Sigma}$ (and $\bar{\Theta} = \bar{\Sigma}^{-1}$ as the inverse of sample covariance), $\bar{\mu}_+$ and $\bar{\mu}_-$, the LDA model classifies a new data vector x as the result of

$$\bar{f}(x) = \text{sign} \left(\left(x - \frac{\bar{\mu}_+ + \bar{\mu}_-}{2} \right)^\top \bar{\beta} + \log \frac{\pi_+}{\pi_-} \right), \quad (2)$$

where $\bar{\beta} = \bar{\Theta}(\bar{\mu}_+ - \bar{\mu}_-)$ relies on the accurate estimation of inverse covariance matrix $\bar{\Theta} = \bar{\Sigma}^{-1}$. The function $\text{sign}(\cdot) \in \{-1, +1\}$, π_+ and π_- refer to the (foreknown) frequencies of positive and negative samples in the whole population respectively.

3.2 Combinatorial discriminant analysis problem

Given labeled data pairs $(x_1, y_1) \cdots (x_n, y_n)$ for training, we assume each tuple (x_i, y_i) ($1 \leq i \leq n$) is i.i.d drawn from a joint distribution denoted as $(\mathcal{X}, \mathcal{Y})$. Specifically the p -dimension vector x_i is drawn from two (unknown) Gaussian distributions $\mathcal{N}(\mu_+^*, \Sigma^*)$ and $\mathcal{N}(\mu_-^*, \Sigma^*)$ with *equal priors*. While label $y_i = +1$ when x_i is drawn from $\mathcal{N}(\mu_+^*, \Sigma^*)$, corresponding label $y_i = -1$ when x_i is drawn from $\mathcal{N}(\mu_-^*, \Sigma^*)$.

Problem. Given an integer $k \leq p$, our research intends to find the sparse LDA estimator $\hat{\beta}$ with k nonzero coefficient that can recover the combinatorial structure of the optimal LDA estimate β^* . More specific, let us denote $\text{supp}(\beta)$ as a function that maps a vector β to the set of the nonzero coefficients in the vector (i.e., support set of the vector). Thus, we formulate the problem of combinatorial discriminant analysis as:

$$\hat{\beta} \leftarrow \arg \max_{\forall \beta \in \mathbb{R}^p} |\text{supp}(\beta^*) \cap \text{supp}(\beta)| \text{ s.t. } R(\beta) \leq \varepsilon \text{ and } |\text{supp}(\beta)| = k, \quad (3)$$

where $R(\beta) \leq \varepsilon$ refers to the bound of the expected error rate of β for sparse LDA classification [21] and $R(\beta)$ has been defined in Sect. 3 of [10]. This problem intends to search a subset of k features from the overall p features and assign each feature a weight, for optimal linear classification with generalizability ensured. Considering the combinatorics in the subproblem [22], the overall problem should be NP-hard. Note that we study binary LDA problem in this work, while multiclass LDA could be adopted through one-vs.-rest transformation.

4 The proposed algorithm

In this section, we introduce **CDA** with two steps: *Covariance Matrices Estimation* and *Conditioned Rayleigh Flow*. **CDA** outputs an near-optimal estimation of the projection vector denoted as $\hat{\beta}_t$, where t refers to the number of iterations for Conditioned Rayleigh Flow. **CDA** replaces $\hat{\beta}$ with $\hat{\beta}_t$ in Eq. (2) as the classification rule.

4.1 Covariance matrices estimation

Given the training data and the tuning parameter $\lambda > 0$, this step estimates three covariance matrices as the input of the Conditioned Rayleigh Flow.

4.1.1 Between-class covariance matrix estimation

Given the labeled data pairs $(x_1, \ell_1) \cdots (x_n, \ell_n)$ for training, the algorithm first sorts the the training data, using the label, into the two sets \mathbb{X}_+ and \mathbb{X}_- for the data with positive and negative label respectively. It estimates the mean vectors $\bar{\mu}_+ = \frac{1}{|\mathbb{X}_+|} \sum_{x_i \in \mathbb{X}_+} x_i$, and $\bar{\mu}_- = \frac{1}{|\mathbb{X}_-|} \sum_{x_i \in \mathbb{X}_-} x_i$ for the two classes separately. Based on the overall mean $\bar{\mu}$ and the mean vectors of the two classes $\bar{\mu}_+$, $\bar{\mu}_-$, the algorithm uses the sample estimator as the Between-Class Covariance Matrix [23] such that:

$$\hat{\Sigma}_b = \sum_{\ell \in \{\pm 1\}} \frac{|\mathbb{X}_\ell|}{n} (\bar{\mu}_\ell - \bar{\mu})(\bar{\mu}_\ell - \bar{\mu})^\top. \quad (4)$$

4.1.2 Shrunk within-class covariance matrix estimation

Given the training data and the mean estimation, the algorithm first estimates the sample within-class covariance matrix:

$$\bar{\Sigma}_w = \frac{1}{n} \left\{ \sum_{\ell \in \{\pm 1\}} \sum_{x_i \in \mathbb{X}_\ell} (x_i - \bar{\mu}_\ell)(x_i - \bar{\mu}_\ell)^\top \right\}. \quad (5)$$

It is obvious that, when $p \gg n$ (under HDLSS settings), the sample covariance matrix $\bar{\Sigma}_w$ is usually singular [24]. To obtain on a robust estimation of within-class covariance matrix, **CDA** proposes to use the inversion of Graphical Lasso estimator [9]. Such that $\hat{S}_w = \hat{\Theta}_w^{-1}$ and

$$\hat{\Theta}_w = \underset{\Theta \geq 0}{\operatorname{argmin}} \left(\operatorname{tr}(\bar{\Sigma}_w \Theta) - \log \det(\Theta) + \lambda \sum_{j \neq k} |\Theta_{jk}| \right), \quad (6)$$

where $\hat{\Theta}_w$ is the Graphical Lasso estimator, $\Theta \geq 0$ refers to the positive-semidefinite constraint, $|\Theta_{i,j}|$ refers to the absolute value of the matrix element $\Theta_{i,j}$, and the tuning parameter λ is used to control the sparsity. Note that $\hat{\Theta}_w$ is sparse and \hat{S}_w is biased (i.e., $\hat{S}_w^{-1} \bar{\Sigma}_w \neq I$) due to the induced sparsity.

4.2 De-biasing covariance estimation

As was mentioned, we set covariance structure on mild conditions and thus need to de-bias/de-sparse the within-class covariance estimator [16]. Given the sample estimator $\bar{\Sigma}_w$, we measure the bias of \hat{S}_w (or $\hat{\Theta}_w$) as \hat{B}_w and correct the bias with \hat{C}_w as follow.

$$\hat{B}_w = (\hat{\Theta}_w \bar{\Sigma}_w - I), \quad \text{and} \quad \hat{C}_w = ((I - \hat{B}_w) \hat{\Theta}_w)^{-1} \hat{B}_w. \quad (7)$$

CDA lowers the bias of \hat{S}_w via $\hat{S}_w + \hat{C}_w$. In fact, $\hat{S}_w + \hat{C}_w = (2\hat{\Theta}_w - \hat{\Theta}_w \bar{\Sigma}_w \hat{\Theta}_w)^{-1}$ is equivalent to the inverse matrix of the de-biased graphical Lasso estimator [16] with a faster statistical convergence rate.

4.2.1 Statistical convergence of covariance

In above sections, we present three within-class covariance matrix estimators $\bar{\Sigma}_w$, \hat{S}_w and $\hat{S}_w + \hat{C}_w$ that improve the estimation of Σ^* step-by-step. The sample estimator $\bar{\Sigma}_w$ is an unbiased estimator of Σ^* with statistical errors due to the finite sample estimation, however $\bar{\Sigma}$ is often singular under high-dimensional and low sample size ($p \gg n$) settings.

Remark 1 Let's denote $\Theta^* = \Sigma^{*-1}$ and d as the maximal node degree of the graph of Θ^* . The estimator \hat{S}_w is the inverse matrix of graphical Lasso estimator $\hat{\Theta}_w$ (i.e., $\hat{S}_w = \hat{\Theta}_w^{-1}$) which provides an improved non-singular estimator of Σ^* with a statistical convergence rate $\|\hat{\Theta}_w - \Theta^*\|_F^2 = O((p+d) \log p/n)$ under mild conditions on the eigenvalues of Θ^* and sparsity assumptions [25,26]. The de-biased estimator $(\hat{S}_w + \hat{C}_w)$ further improves \hat{S}_w while its inverse matrix $(\hat{S}_w + \hat{C}_w)^{-1} = (2\hat{\Theta}_w - \hat{\Theta}_w \bar{\Sigma}_w \hat{\Theta}_w)$ converging to Θ^* with a sharper ℓ_∞ -norm statistical convergence rate $O(\sqrt{\log p/n})$ under milder conditions [16].

With improved estimation of covariance matrices, **CDA** is expected to outperform the existing solution to the problem.

4.3 Conditioned Rayleigh flow

Given the desired number of nonzero coefficients k , **CDA** adopts a *Conditioned Rayleigh Flow* listed in Algorithm 1 to estimate a k -sparse discriminant projection vector $\hat{\beta}_1$. *Conditioned Rayleigh Flow* is a gradient ascent derived from [1], which maximizes the Fisher's information for optimal discrimination [2] under ℓ_0 -norm constraint. The step-size is $\eta = 1.0 \times 10^{-4}$, the

Algorithm 1 Conditioned “Rifle” Gradient Ascent

```

1: procedure CRIFLE( $\widehat{\Sigma}_b, \widehat{\Sigma}_w, \widehat{S}_w, \widehat{\Theta}_w, \widehat{C}_w, \bar{\mu}_+, \bar{\mu}_-, \eta, T, tol$ )
2: /*Preconditioning with Initialized  $\widehat{\beta}_0$ */
3:  $\widehat{\beta}^s \leftarrow \widehat{\Theta}_w(\bar{\mu}_+ - \bar{\mu}_-);$  /* Initial Estimation */
4:  $\widehat{\beta}^d \leftarrow (I - \widehat{B}_w)\widehat{\beta}^s;$  /* Bias Reduction in Initialization */
5:  $\widehat{\beta}^H \leftarrow \text{HardThreshold}(\widehat{\beta}^d, k);$ 
6:  $\widehat{\beta}_0 \leftarrow \widehat{\beta}^H / |\widehat{\beta}^H|_2;$ 
7:  $t \leftarrow 0;$ 
8: do
9:    $t \leftarrow t + 1;$ 
10: /*Conditioned Gradient Ascent with Projection*/
11:    $\rho_{t-1} \leftarrow \widehat{\beta}_{t-1}^\top \widehat{S}_w \widehat{\beta}_{t-1} / \widehat{\beta}_{t-1}^\top \widehat{\Sigma}_b \widehat{\beta}_{t-1};$ 
12:    $\rho'_{t-1} \leftarrow \rho_{t-1} + \widehat{\beta}_{t-1}^\top \widehat{C}_w \widehat{\beta}_{t-1} / \widehat{\beta}_{t-1}^\top \widehat{\Sigma}_b \widehat{\beta}_{t-1};$ 
13:    $\nabla_{t-1} = \widehat{\Sigma}_b \widehat{\beta}_{t-1} - \widehat{S}_w \widehat{\beta}_{t-1} / \rho'_{t-1};$ 
14:    $\nabla'_{t-1} = \nabla_{t-1} - \widehat{C}_w \widehat{\beta}_{t-1} / \rho'_{t-1};$  /* Bias Reduction in Gradient Estimation */
15:    $\widehat{\beta}_t^g \leftarrow \widehat{\beta}_{t-1} - \eta \cdot \nabla'_{t-1};$ 
16:    $\widehat{\beta}_t^h \leftarrow \text{HardThreshold}(\widehat{\beta}_t^g, k);$ 
17:    $\widehat{\beta}_t \leftarrow \widehat{\beta}_t^h / |\widehat{\beta}_t^h|_2;$ 
18: while  $t < T$  and  $|\widehat{\beta}_t - \widehat{\beta}_{t-1}|_\infty \geq tol$ 
19: return  $\widehat{\beta}_t;$ 
20: end procedure

```

Algorithm 2 Hard Thresholding

```

1: procedure HARDTHRESHOLD( $\widehat{\beta}, k$ )
2: /* List the absolute value of each element in  $\widehat{\beta}$  */
3:  $\mathbb{S} \leftarrow \{|\widehat{\beta}[1]|, |\widehat{\beta}[2]|, \dots, |\widehat{\beta}[p]|\};$ 
4: /*Descending-Sort the List */
5: Sort( $\mathbb{S}$ );
6: /*Set the threshold  $T_{hr}$  */
7:  $T_{hr} \leftarrow \mathbb{S}[k];$ 
8: /*Truncate with  $T_{hr}$  */
9: for  $i = 1, 2, 3, \dots, p$  do
10:   if  $|\widehat{\beta}[i]| \leq T_{hr}$  then  $\widehat{\beta}[i] \leftarrow 0;$ 
11:   end if
12: end for
13: Return  $\widehat{\beta};$ 
14: end procedure

```

maximal tolerated perturbation is $tol = 1.0 \times 10^{-3}$, and the maximal number of iterations is set to $T = 200$ in our research.

Specifically, the proposed *Conditioned Rayleigh Flow* algorithm consists of two major steps:

- *Step 1. Preconditioning with Initialized $\widehat{\beta}_0$* —In this step, given the shrunken within-class covariance matrix ($\widehat{S}_w + \widehat{C}_w$) and the mean vectors $\bar{\mu}_+, \bar{\mu}_-$, the algorithm first leverages a Scout [9] estimator (in Line 3 of Algorithm 1) to compute $\widehat{\beta}^d$, where the de-biased linear model [15,16] has been used. Further, given the desired number of nonzero elements k , hard thresholding and normalization (in Lines 4-5 of Algorithm 1) are used to project $\widehat{\beta}^d$ into the desired region to initialize $\widehat{\beta}_0$ (i.e., $|\widehat{\beta}_0|_0 \leq k$ and $|\widehat{\beta}_0|_2 = 1$), for gradient ascent. Note that the `HardThreshold` function used in Algorithm 1 is introduced in Algorithm 2. The further theoretical analysis on the preconditioned $\widehat{\beta}_0$ is presented in **Theorem 2**.

- *Step 2. Conditioned Gradient ascent with Projection*—Given the de-biased within-class covariance matrix $(\widehat{S}_w + \widehat{C}_w)$ and between-class covariance matrix $\widehat{\Sigma}_b$, the proposed algorithm **CDA** indeed intends to find the $\widehat{\beta}$ that maximize the Fisher's information:

$$\underset{\widehat{\beta} \in \mathbb{R}^p}{\operatorname{argmax}} \frac{\widehat{\beta}^\top \widehat{\Sigma}_b \widehat{\beta}}{\widehat{\beta}^\top (\widehat{S}_w + \widehat{C}_w) \widehat{\beta}} \quad \text{s.t. } |\widehat{\beta}|_0 = k \text{ and } |\widehat{\beta}|_2 = 1 \quad (8)$$

where $\frac{\widehat{\beta}^\top \widehat{\Sigma}_b \widehat{\beta}}{\widehat{\beta}^\top (\widehat{S}_w + \widehat{C}_w) \widehat{\beta}}$ refers to the generalized Rayleigh quotient based on the within/between-classes covariance matrices that have been used for Fisher's information maximization. In order to achieve the goal, a gradient ascent algorithm is proposed to minimize $\widehat{\beta}^\top (\widehat{S}_w + \widehat{C}_w) \widehat{\beta} / \widehat{\beta}^\top \widehat{\Sigma}_b \widehat{\beta}$ under the same constraint. Instead of minimizing the quotient of $\widehat{\beta}^\top (\widehat{S}_w + \widehat{C}_w) \widehat{\beta}$ and $\widehat{\beta}^\top \widehat{\Sigma}_b \widehat{\beta}$, the algorithm indeed minimizes the gap between them (Rayleigh flow). Starting with $\widehat{\beta}_0$, the algorithm iteratively update $\widehat{\beta}_t$ ($t = 1, 2, \dots$) using the gradient ∇_{t-1} (estimated in Lines 10–12 of Algorithm 1). Specifically, the gradient is estimated (in Line 12 of Algorithm 1) as $\nabla_{t-1} = \widehat{\Sigma}_b \widehat{\beta}_{t-1} - (\widehat{S}_w + \widehat{C}_w) \widehat{\beta}_{t-1} / \rho_{t-1}$, where ρ_{t-1} (estimated in Line 10 of Algorithm 1) is an adjusting parameter that adjusts gap to approximate the quotient minimization. Through projection, the algorithm diffuses $\widehat{\beta}_t$ in the desired region after each iteration, so as to meet sparsity constraints.

Note that the major contributions made in our research on top of [1] is that we propose to leverage *preconditioning initialization* and *conditioned gradient estimation* to improve the performance of optimization and estimation, so as to advance the state-of-the-art (which was based on the sample-based estimators) with significant performance enhancement.

4.4 The CDA Classifier

After obtaining the output of Algorithm 1, i.e., $\widehat{\beta}_t$, given any p -dimensional vector $\forall \mathbf{x} \in \mathbb{R}^p$ for classification, the proposed algorithm classify \mathbf{x} using the following linear classification rule:

$$\operatorname{sign} \left(\mathbf{x}^\top \widehat{\beta}_t - \frac{(\bar{\mu}_+ + \bar{\mu}_-)^T}{2} \widehat{\beta}_t + \log \frac{\pi_+}{\pi_-} \right), \quad (9)$$

where $\bar{\mu}_+$ and $\bar{\mu}_-$ refer to the sample mean vector of the two classes.

5 Algorithm analysis

Here we introduce the main result of analysis, where we first introduce the key assumptions and main results of the algorithm analysis, then present theoretical results of [1] and two key lemmas that help obtain our main result by reusing the results in [1], finally the sketched proofs of the two lemmas are given.

5.1 Main results: assumptions and theorems

Suppose n samples are randomly drawn from (unknown) Gaussian distributions $\mathcal{N}(\mu_+^*, \Sigma^*)$ and $\mathcal{N}(\mu_-^*, \Sigma^*)$ with equal priors, then the optimal projection vector β^* should be $\beta^* = \Sigma^{*-1}(\mu_+^* - \mu_-^*)$. Let $\bar{\beta}^*$ denote the normalized optimal projection vector $\bar{\beta}^* = \beta^* / |\beta^*|_2$. We focus on the Euclid distance between the estimated $\widehat{\beta}_t$ and $\bar{\beta}^*$ i.e., $|\widehat{\beta}_t - \bar{\beta}^*|_2$ and how

it converges with the number of dimensions p and the number of training samples n , under the structural assumption as follow.

Assumption 1 Let's denote $\Theta^* = \Sigma^{*-1}$. The optimal LDA [2] estimator $\beta^* = \Theta^*(\mu_+^* - \mu_-^*)$ should be s -sparse i.e., $|\beta^*|_0 = s$ and $1 \leq s < p$.

Assumption 2 We assume all training/testing samples are realized from a random vector X , where $|X|_2 \leq \mathcal{L}$. Thus, there has $|\mu_+^* - \mu_-^*|_2 \leq 2\mathcal{L}$, $|\bar{\mu}_+ - \bar{\mu}_-|_2 \leq 2\mathcal{L}$, $|\mu_-^* - \bar{\mu}_-|_2 \leq 2\mathcal{L}$, and $|\mu_+^* - \bar{\mu}_+|_2 \leq 2\mathcal{L}$.

Assumption 3 [25] There exists a positive constant \mathcal{B} that bounds the eigenvalues of Σ^* such as $1/\mathcal{B} \leq \lambda_{\min}(\Sigma^*) \leq \lambda_{\max}(\Sigma^*) \leq \mathcal{B}$.

Assumption 4 With appropriate setting of k and $k \geq s$, we assume that $\text{supp}(\beta^*) \subseteq \text{supp}(\hat{\beta}^h)$, where $\hat{\beta}^h$ is defined in Algorithm 1. Note that $\hat{\beta}^h$ is the hard-thresholding result of $\hat{\beta}^d$ with top- k elements preserved in Algorithm 1, there thus has $\text{supp}(\hat{\beta}^h) \subseteq \text{supp}(\hat{\beta}^d)$.

Assumption 5 To simplify our research, we suppose the data samples are well normalized. In this way, for $\forall 1 \leq i, j \leq p$ and $i \neq j$, there has $|\Sigma_{i,j}^*| \leq |\Sigma_{i,i}^*|$.

Based on above assumptions, we make the main theorem as follow.

Theorem 1 (Main Result) *The ℓ_2 -norm convergence rate of $\hat{\beta}_t$ (CDA after t iterations) in high probability should be:*

$$|\hat{\beta}_t - \bar{\beta}^*|_2 \leq \mathcal{O} \left(\left(\frac{v^t}{|\beta^*|_2} + 1 \right) \sqrt{\frac{(2k+s) \log p}{n}} \right). \quad (10)$$

where $\bar{\beta}^* = \beta^*/|\beta^*|_2$ normalizes the optimal LDA, $k = |\hat{\beta}_t|_0$ refers to the desired number of nonzero elements, $s = |\beta^*|_0$ refers to the number of true nonzero elements. Moreover, $v \in (0, 1)$ referring to the rate of error reduction per iteration.

5.2 The analytical framework based on truncated Rayleigh flow

The convergence rate of Truncated Rayleigh Flow [1] is sensitive to (1) the initial setting $\hat{\beta}_0$ in our algorithm, (2) the estimation error of Within-Class and Between-Class covariance matrices, and (3) how does s and k match. Thus, the earlier result can be summarized as:

$$|\hat{\beta}_t - \bar{\beta}^*|_2 \leq v^t \cdot |\hat{\beta}_0 - \bar{\beta}^*|_2 + \xi^* \sqrt{\rho(\Sigma^* - (\hat{S}_w + \hat{C}_w), 2k+s)^2 + \rho(\Sigma_b^* - \hat{S}_b, 2k+s)^2}, \quad (11)$$

where ξ^* denotes as a scalar that depends on Σ^* , μ_+^* and μ_-^* , $\rho(\mathbf{E}, 2k+s) = \sup_{v \in \mathbb{R}^p, |v|_0 \leq 2k+s, |v|_2=1} |v^T \mathbf{E} v|$ and \mathbf{E} is a $p \times p$ matrix here. There has $\rho(\Sigma_b^* - \hat{S}_b, 2k+s) \leq \mathcal{O}(\sqrt{(2k+s) \cdot \log p/n})$, according to [27].

To obtain our results in Theorem 1, our work introduce and prove two lemmas as follow.

Lemma 1 *Let denote $\bar{\beta}^* = \beta^*/|\beta^*|_2$ and $\hat{\beta}_0$ refers to the normalized vector in Line 6 of Algorithm 1 (and $|\bar{\beta}^*|_2 = |\hat{\beta}_0| = 1$). There has*

$$|\hat{\beta}_0 - \bar{\beta}^*|_2 \leq \mathcal{O} \left(|\beta^*|_2^{-1} \sqrt{\frac{k \log p}{n}} \right), \quad (12)$$

in high probability.

Lemma 2 Suppose s refers to the number of the nonzero elements in β^* , i.e., $|\beta^*|_0 = s$. Given any integer k and $2k + s \leq p$, there has:

$$\sup_{v \in \mathbb{R}^p, |v|_0 \leq 2k+s, |v|_2=1} |v^\top (\Sigma^* - (\hat{S}_w + \hat{C}_w)) v| \leq \mathcal{O} \left(\sqrt{\frac{(2k+s) \log p}{n}} \right), \quad (13)$$

in high probability.

Considering the two constants ν and ξ^* already defined in the earlier work [1], we can easily combine the results of **Lemmas 1** and **2** with the results listed in Eq. (11). The proofs of above two lemmas are in the following section.

6 Experiments

In this section, we report our experiments using both synthesized and real-world datasets.

6.1 Evaluation using the synthesized dataset

In this experiment, we evaluate **CDA** on the multivariate Gaussian data under HDLSS settings. Specifically, we intend to (1) study the performance of **CDA** to recover the combinatorial structure of the sparse projection vector β , and (2) understand the performance of **CDA** for classification task.

6.1.1 Data synthesis

The synthetic data are generated by two predefined Gaussian distributions $\mathcal{N}(\mu_+^*, \Sigma^*)$ and $\mathcal{N}(\mu_-^*, \Sigma^*)$ with equal priors. The settings of μ_+^* , μ_-^* and Σ^* are as follows: Σ^* is a $p \times p$ symmetric and positive-definite matrix, where each element $\Sigma_{i,j}^* = 0.8^{|i-j|}$, $1 \leq i \leq p$ and $1 \leq j \leq p$. μ_+^* and μ_-^* are both p -dimensional vectors, where $\mu_+^* = \langle 1, 1, \dots, 1, 0, 0, \dots, 0 \rangle^T$ (the first 10 elements are all 1's, while the rest $p - 10$ elements are 0's) and $\mu_-^* = \mathbf{0}$. (Settings of the two Gaussian distributions first appear in [10].) In our experiment, we set $p = 200$. Note that, under this setting, the optimal projection vector should be $\beta^* = \Sigma^{*-1}(\mu_+^* - \mu_-^*)$ and the normalized vector $\bar{\beta}^* = \beta^*/|\beta^*|_2$, which is with totally 11 nonzero elements—i.e., 11 variables/features are selected into the optimal discriminant analysis.

6.1.2 Baseline algorithms and settings

To understand the performance of the proposed algorithm, we compare **CDA** with following baselines:

TRifle and **Rifle**—**TRifle** is directly brought from [1] using the sample estimation of covariance matrices. Pseudo-inverse of the sample within-class covariance matrix is used for the algorithm initialization, when the inverse of the sample covariance matrix doesn't exist. Note that the parameter k is also used here to control the number of nonzero elements. **Rifle** leverages the common Rayleigh Flow optimization process to compute the projection vector using the sample covariance matrix estimation and without any sparsity constraint.

OMP and **SDA**—The **OMP** algorithm is derived from Orthogonal Matching Pursuit [17], which approximates a sparse projection vector for discrimination under ℓ_0 -norm sparsity

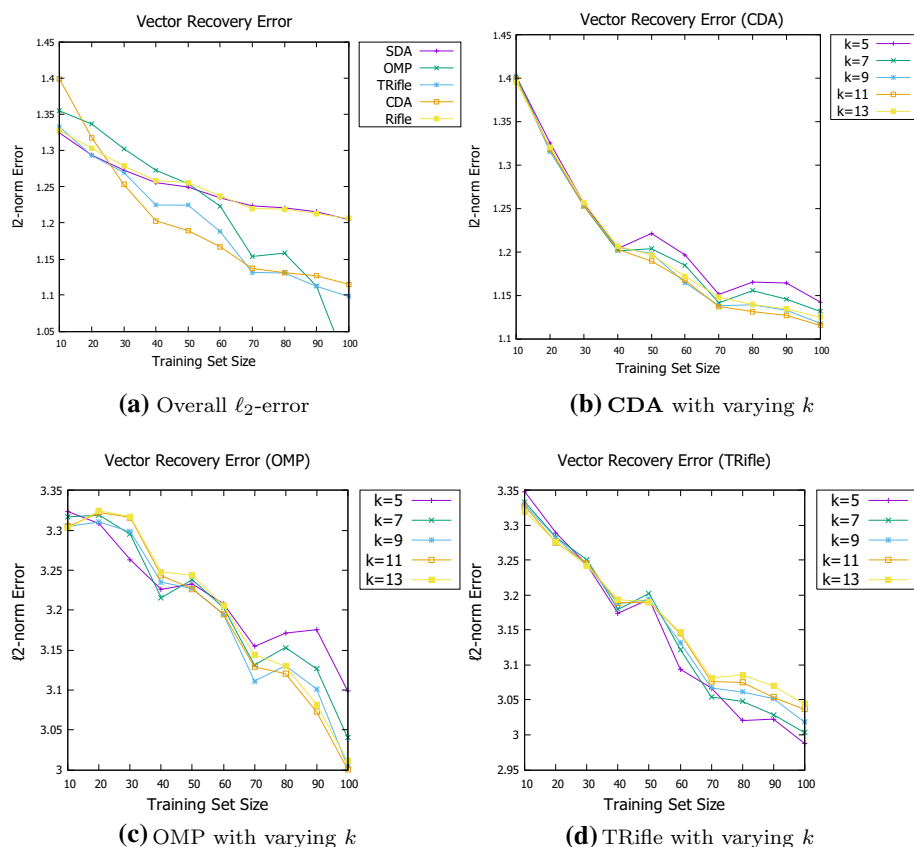


Fig. 1 Performance on ℓ_2 -norm estimation error

constraint with a least-square loss [13,19]. The desired number of nonzero elements k is used here. In addition to ℓ_0 -norm sparsity, **SDA** is derived from Scout [9] using Graphical Lasso estimator [20] (using ℓ_1 -norm sparsity). $\hat{\beta}_0$ in the Line 3 of Algorithm 1 should perform the same function as this algorithm. Note that the tuning parameter λ is used here for shrunk covariance matrix estimation,

All these algorithms first estimate the projection vector $\hat{\beta}$, then classify the new vector using the same rule in Eq. (2). To simulate the HDLSS settings, we train **CDA** and baseline algorithms, with 10 to 100 samples randomly drawn from the distributions with equal priors, and test the two algorithms with 500 samples. For each settings, we repeat the experiments for 100 times and take the averaged results. Further, a grid search algorithm is used to evaluate the algorithms under various parameter (i.e., k and λ) settings.

6.1.3 Estimation error and consistency

First of all, to verify the consistency of proposed estimators, we present the ℓ_2 -norm error of the estimated projection vector (i.e., $|\hat{\beta}_l - \beta^*|_2$ for **CDA**) and compare it to other algorithms including OMP, SDA, TRifle and Rifle. In Fig. 1a, the overall ℓ_2 -norm error of all algorithms under varying training dataset size is presented, where all algorithms are tuned with the

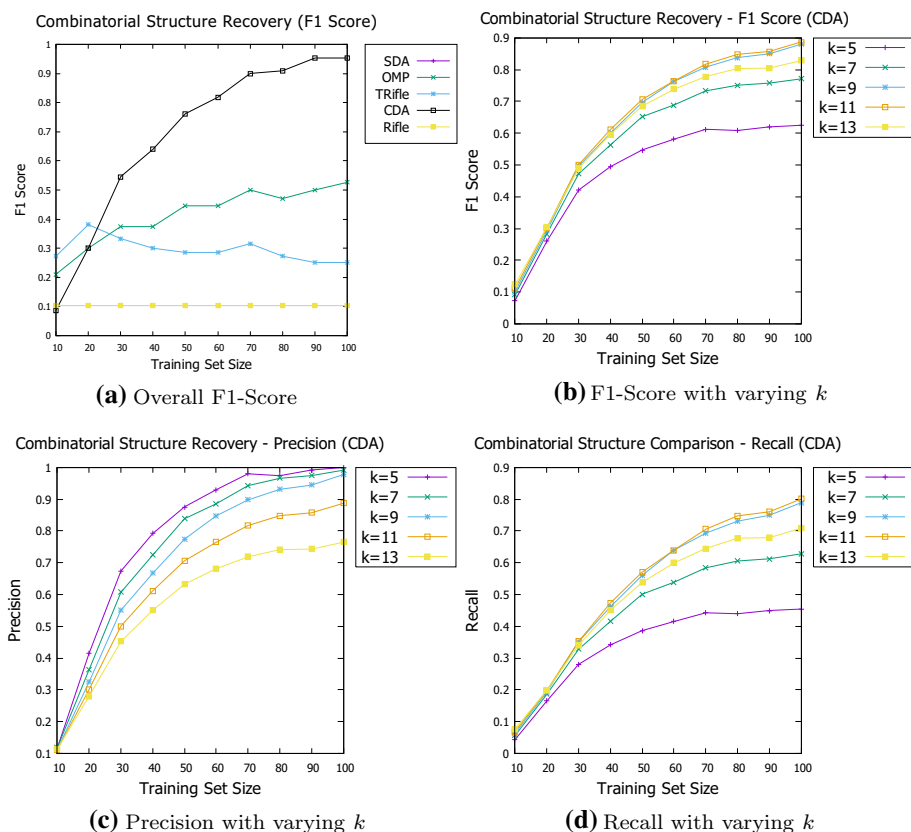


Fig. 2 Performance on combinatorial structure recovery

best parameters (through grid search and three-fold cross validation) in each setting. For fair comparison, vectors estimated by OMP, SDA, TRifle and Rifle are normalized, and compare to $\hat{\beta}^*$. It is obvious that **CDA** is with lower ℓ_2 -norm error than all baseline algorithms, when sample size is between 30 and 70. Figure 1b illustrates the ℓ_2 -norm estimation error using **CDA** with varying desired number of nonzero elements k (while λ is set to the best), where we can see **CDA** gets its lowest ℓ_2 -norm error when $k = 11$. The descending trend of $|\hat{\beta}_t - \hat{\beta}^*|_2$ shown in Fig. 1b demonstrates the potentials of empirical consistency.

6.1.4 Combinatorial structure recovery

Further, Fig. 2 demonstrates the performance of **CDA** and baseline algorithms for combinatorial structure recovery. In Fig. 2a, the overall F1-score for combinatorial structure recovery of all algorithms under varying training dataset size is presented, where all algorithms are tuned with the best parameters (through grid search) in each setting. Compared to the optimal $\hat{\beta}^*$, F1-score of the estimated $\hat{\beta}$ takes both precision and recall of nonzero element retrieval into consideration, and it is calculated as

$$\text{Precision} = \frac{|\text{supp}(\hat{\beta}) \cap \text{supp}(\beta^*)|}{|\text{supp}(\hat{\beta})|}, \quad (14)$$

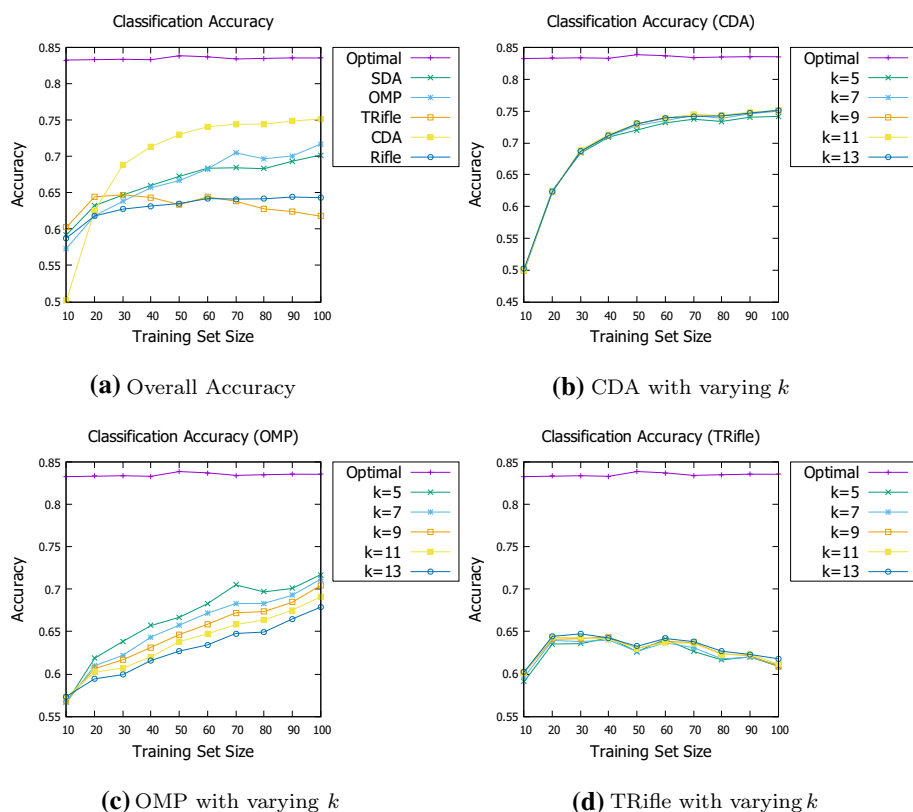


Fig. 3 Performance comparison on classification

$$\text{Recall} = \frac{|\text{supp}(\hat{\beta}) \cap \text{supp}(\beta^*)|}{|\text{supp}(\beta^*)|}, \quad (15)$$

$$\text{F1-Score}(\hat{\beta}, \beta^*) = 2 \cdot \frac{|\text{supp}(\beta^*)| \cdot |\text{supp}(\hat{\beta})|}{|\text{supp}(\beta^*)| + |\text{supp}(\hat{\beta})|}. \quad (16)$$

It demonstrates that **CDA** outperforms all baseline algorithms including OMP, TRifle, SDA and Rifle with higher F1-score, when sample size is larger than 30. Figure 2b, c illustrate the F1-score, Precision and Recall of nonzero element retrieval applying **CDA** with varying desired number of nonzero elements k (while λ is set to the best). Obviously, **CDA** is with its best F1-score when $k = 11$, it also gets its best precision and recall when $k = 5$ and $k = 11$ respectively. The comparison shows the potential of **CDA** to select features optimally under HDLSS settings.

Classification accuracy. We also compare the classification performance of **CDA** with baseline algorithms. Figure 3a presents the accuracy of each algorithm using the same parameters/settings that was used in Fig. 2a. Apparently, **CDA** outperforms all baseline algorithms. In Fig. 3b, we demonstrate the classification of **CDA** with varying k . We can also observe that, in our experiments, the performance of **CDA** is not significantly susceptible to the choice of k .

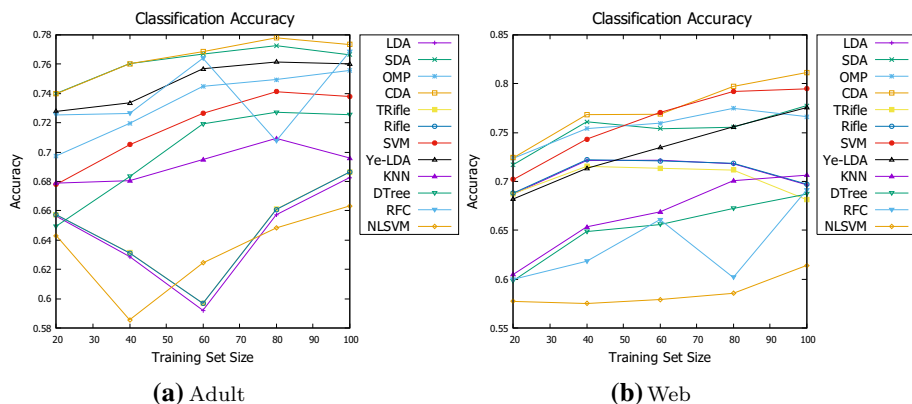


Fig. 4 Performance on adult and web datasets

In summary, compared **CDA** to **SDA**, the performance advantage of **CDA** is contributed by the Conditioned Rayleigh Flow gradient ascent to refine $\hat{\beta}_0$ (which is equivalent to **SDA** in our experiment); compared **CDA** to **TRifle**, the performance advantage of **CDA** is due to that we condition the truncated Rayleigh Flow gradient ascent with shrunk covariance matrix estimator and bias reduction.

6.2 Classification with benchmark datasets

In this experiment, we evaluate **CDA** on real-world datasets. We evaluate our algorithm using binary classification benchmarks—“Adult” and “Web” datasets imported from [28], where Adult dataset is with 123 dimensions ($p = 123$) and Web is with 300 dimensions ($p = 300$), respectively. To evaluate the algorithms under HDLSS settings, we train the algorithms using 20 to 100 samples randomly drawn from the datasets with equal priors (of the two classes).

Figure 4 presents the classification accuracy of **CDA**, baseline algorithms and downstream classifiers, including Support Vector Machine (**SVM**), the least-square discriminant analysis (**Ye-LDA**) proposed by [19], k-Nearest Neighbor (**KNN**), Decision Tree (**DTree**), Random Forest classifier (**RFC**) and Kernel SVM with Gaussian Kernel (**NLSVM**) for nonlinear classification. All algorithms are tuned with the best parameter/settings through grid search (e.g., searching the best from 1-NN, 3-NN, 5-NN...). It is obvious that **CDA** outperforms all other algorithms using both datasets under such HDLSS settings. The performance of **LDA** is not quite stable with the increasing number of training samples, since all these methods use Pseudo-inverse when the sample covariance matrix is singular.

6.3 HDLSS classification with biomedical data

Two ultra-high dimensional biomedical datasets—Leukemia cancer datasets ($p = 7,128$) [29] and Colon cancer datasets ($p = 2,000$) [28] are used to further evaluate our algorithms. We compare **CDA** to Decision Tree, Random Forest, and SVM. With respect to HDLSS settings, we train these classifiers using 20 samples randomly drawn from the datasets with equal priors ($n = 20$), and test the classifiers through cross-validation.

Table 1 demonstrates the averaged accuracy and F1-score with standard deviation of these algorithms on the two datasets after 100 rounds of cross-validation, where all algorithms

Table 1 Accuracy and F1-score comparison between **CDA** and other baselines

Algorithm	Colon		Leukemia	
	Accuracy	F1 Score	Accuracy	F1 Score
CDA	0.801±0.099	0.798±0.109	0.958±0.039	0.960±0.037
OMP	0.806±0.096	0.806±0.102	0.952±0.039	0.952±0.039
SDA	0.590±0.152	0.571±0.171	0.964±0.034	0.964±0.032
D-Tree	0.669±0.113	0.658±0.140	0.804±0.099	0.800±0.111
Rand. Forest	0.801±0.097	0.798±0.109	0.957±0.037	0.956±0.036
SVM	0.797±0.095	0.812±0.091	0.906±0.047	0.914±0.040

are tuned with the best parameters. compared to [29], our experiment takes fewer samples for training, the performance however is still comparable to the previous work. Result shows **CDA** delivers one of the best performance with decent accuracy and F1-score in all experiments for the both datasets. While OMP delivers slightly better performance in Colon datasets, **CDA** outperforms OMP in Leukemia datasets. Similar patterns could be also found in the comparison between **CDA** and SDA. Though **CDA** only selects a small subset of features for classification, it could deliver comparable performance against the baselines.

6.4 Combinatorial inference for tobacco-related behaviors

The frequent tobacco use is one of the leading causes of “*preventable illness and death*” in the world. This experiment proposed to use **CDA** to understand the tobacco-related behaviors in United States.

Data. We collected Foursquare check-in data of 120,853 United States Residence for 2-years (2011–2013) [30]. These human subjects are homed in 32 states of United States, where 16 states are with more *frequent smokers than the national average* (positive) and the rest are with less *frequent smokers than the national average* (negative). In this experiment, each state was first labeled according to its frequent smokers (as was mentioned). Then, the behavior pattern of each state was characterized using an *averaged check-in frequency vector*, where each dimension of the vector refers to the averaged frequency of check-ins on a certain type of venue (e.g., Bar, Gym and Church) per human subject per year in the state. There are totally 428 types of venues are considered in this experiments (i.e., $p = 428$ and $n = 32$).

Methodologies. In this experiment, we intend to analyze the correlation between check-ins and the tobacco use through **CDA** path—an combinatorial inference tool based on proposed algorithm. To achieve the goal, we first estimated the projection vectors $\hat{\beta}$ using the increasing k (i.e., number of nonzero elements in the vector) from 1 to 428. We vary k from 1 to 428. For each k , to ensure the stability of **CDA** path, we repeat the $\hat{\beta}$ estimation with sub-sampling for 100 times, and use *averaging-and-truncation* strategy to estimate the robust projection vectors (denoted as $\hat{\beta}^k$ and $k = 1, 2, \dots, 428$).

We sort these vectors $\hat{\beta}^k$ with $k = 1, 2, \dots, 428$, then track of the trends of each coefficient. With a specific setting of $k = k'$, suppose the j th element equals to 0—i.e., $(\hat{\beta}^{k'})_j = 0$, it refers to that the corresponding behavior (such as visiting Gym) is not selected, when any k' behaviors were considered. Intuitively, with a small k' , the selected behaviors are more important than unselected behaviors, in terms of discriminating frequent smoking or infre-

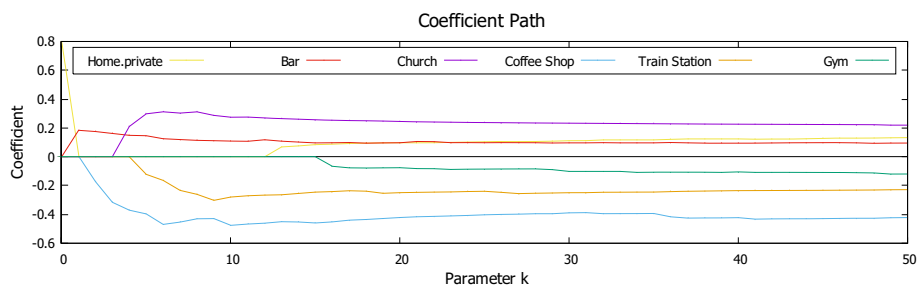


Fig. 5 CDA Path of 6 most significant behaviors/venue types (i.e. Home.private, Bar, Church, Coffee Shop, Train Station, Gym selected from 484 Behaviors/Venue Types) as parameter k increases

quent smoking populations. In this case, we defined that, when $(\hat{\beta}^{k'-1})_j = 0$ but $(\hat{\beta}^{k'})_j \neq 0$, we call the corresponding behavior of (the j th coefficient) was *picked-up* by **CDA** at k' . On the other hand, when $(\hat{\beta}^{k'-1})_j \neq 0$ but $(\hat{\beta}^{k'})_j = 0$, we call the corresponding behavior was *rejected* by **CDA** at k' . We, thus, record the last time that each behavior is *picked-up* by **CDA** and will be never *rejected* with increasing k .

Results. Figure 5 illustrates the **CDA** path of top 6 most significant behaviors with the largest absolute coefficients (when $k = 428$). It shows the coefficient path based on **CDA**, when $1 \leq k \leq 50$. When $k = 1$, a single behavior of “Home.private” is firstly *picked-up* by **CDA**, but then rejected when $k = 2$. In the same time, the behavior of “bar” is selected when $k = 2$ without further rejection with increasing k (i.e., “Bar” behaviors is always selected for $k \geq 3$). The “Home.private” is *picked-up* again when $k = 12$ with no further rejection. Other behaviors such as “Gym”, “Coffee Shop”, “Church” and “Train Station” are *picked-up* by **CDA** when k is relevantly small and are not rejected with increasing k . The result indicates that a state with more normalized check-ins (per human subject per year) in “Bar”, “Church”, and “Home.private” would have more frequent smokers. On the other hand, if a state is with more check-ins in “Gym”, “Train Station” and “Gym”, such state would have a smaller number of frequent smokers.

Note that this figure only includes the coefficient path of these six behaviors from $k = 1$ to 50. They are all *picked-up* at a small k and not rejected (again) with increasing k . Many other behaviors are also *picked-up* at small k , however, they are all latterly rejected when k increases. We believe these 6 behaviors are most relevant to the level of frequent smokers in a state. All in all, it is quite obvious that the behaviors featured as “Home.private”, “Bar” and “Church” are positively related to the level of frequent smokers, while “Coffee Shop”, “Train Station” and “Gym” are negatively related to the level of frequent smokers. Intuitively, it is quite easy to understand the causation of “Bar” and “Gym” to smoking. We are very glad to observe some new patterns that link “Home.private”, “Coffee Shop”, “Church” and “Train Station” to smoking. In our future work, we will work with psychologists to understand the relevance of observed patterns.

7 Conclusion

In this paper, we studied the novel combinatorial discriminant analysis problem, and proposed **CDA** algorithm that can approximate the sparse estimation of the projection vector with desired number (denoted as k) of nonzero elements. The proposed **CDA** operates on top of a

Conditioned Rayleigh Flow gradient ascent algorithm that leverages covariance-regularized initialization ($\hat{\beta}_0$) and de-biased shrunk covariance estimators [16] to further improve TRifle [1]. The experiment results show that **CDA** outperforms a bunch of the existing sparse LDA estimators with lower error for projection vector estimation under HDLSS settings while enjoying better accuracy for combinatorial structure recovery; **CDA** also performs better than the sparse LDA algorithms and downstream classifiers with higher accuracy for HDLSS data classification. Note that though binary LDA problem was studied in this work, the multiclass LDA could be easily adopted through one-vs.-rest transformation.

Supplementary Information The online version contains supplementary material available at <https://doi.org/10.1007/s10115-021-01587-z>.

References

1. Tan KM, Wang Z, Liu H, Zhang T (2018) Sparse generalized eigenvalue problem: optimal statistical rates via truncated Rayleigh flow. *J R Stat Soc: Ser B (Stat Methodol)* 80(5):1057–1086
2. RA Fisher (1936) The use of multiple measurements in taxonomic problems. *Ann Hum Genet* 7(2), 179–188
3. R.O. Duda, P.E. Hart, D.G. Stork (2001) Pattern classification, 2nd edn. Wiley, Hoboken
4. Alipanahi B, Biggs M, Ghodsi A et al (2008) Distance metric learning vs. fisher discriminant analysis. In: Proceedings of the 23rd national conference on artificial intelligence, vol 2, pp 598–603
5. B Kulis et al. (2013) Metric learning: a survey. *Found Trends Mach Learn* 5(4), 287–364
6. R Peck, J Van Ness (1982) The use of shrinkage estimators in linear discriminant analysis. *IEEE Trans Pattern Anal Mach Intell* 5:530–537
7. Buhlmann P, Van De Geer S (2011) Statistics for high-dimensional data: methods, theory and applications. Springer, Berlin
8. KM Amin (2012) Combinatorial regression and improved basis pursuit for sparse estimation. California Institute of Technology, Pasadena
9. Witten DM, Tibshirani R. (2009) Covariance-regularized regression and classification for high dimensional problems. *J R Stat Soc: Ser B (Stat Methodol)* 71(3):615–636
10. Cai T, Liu W (2011) A direct estimation approach to sparse linear discriminant analysis. *J Am Stat Assoc* 106(496), 1566–1577
11. Clemmensen L, Hastie T, Witten D, Ersboll B (2011) Sparse discriminant analysis. *Technometrics* 53(4)
12. Shao J, Wang Y, Deng X, Wang S et al. (2011) Sparse linear discriminant analysis by thresholding for high dimensional data. *Ann Stat* 39(2), 1241–1265
13. Li Y, Jia J et al. (2017) L1 least squares for sparse high-dimensional LDA. *Electron J Stat* 11(1), 2499–2518
14. Baraniuk RG. (2007) Compressive sensing. *IEEE Signal Process Mag* 24(4)
15. Javanmard A, Montanari A (2014) Confidence intervals and hypothesis testing for high-dimensional regression. *J Mach Learn Res* 15(1), 2869–2909
16. Jankova J, Geer S et al (2015) Confidence intervals for high-dimensional inverse covariance estimation. *Electron J Stat* 9(1):1205–1229
17. TT Cai, L Wang. (2011) Orthogonal matching pursuit for sparse signal recovery with noise. *IEEE Trans Inf Theory* 57(7), 4680–4688
18. Krzanowski WJ, Jonathan P, McCarthy WV, Thomas MR (1995) Discriminant analysis with singular covariance matrices: methods and applications to spectroscopic data. *Appl Stat* 44:101–115
19. Ye J (2007) Least squares linear discriminant analysis. In: Proceedings of the 24th international conference on machine learning, pp 1087–1093. ACM
20. Friedman J, Hastie T, Tibshirani R (2008) Sparse inverse covariance estimation with the graphical lasso. *Biostatistics* 9(3), 432–441
21. Anderson TW (1962) An introduction to multivariate statistical analysis. Technical report, Wiley, New York
22. Tropp JA, Gilbert AC (2007) Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Trans Inf Theory* 53(12), 4655–4666
23. Globerson A, Roweis ST (2006) Metric learning by collapsing classes. In: Advances in neural information processing systems, pp 451–458

24. Cai TT, Ren Z, Zhou HH. et al. (2016) Estimating structured high-dimensional covariance and precision matrices: optimal rates and adaptive estimation. *Electron J Stat* 10(1), 1–59
25. Rothman AJ, Bickel PJ, Levina E, Zhu J. et al. (2008) Sparse permutation invariant covariance estimation. *Electron J Stat* 2:494–515
26. Witten DM, Friedman JH, Simon N. (2011) New insights and faster computations for the graphical lasso. *J Comput Graph Stat* 20(4), 892–900
27. Yu Y, Wang T, Samworth RJ (2014) A useful variant of the Davis–Kahan theorem for statisticians. *Biometrika* 102(2):315–323
28. Lin C-J (2017) Libsvm data: classification (binary class)
29. Tibshirani R, Hastie T, Narasimhan B, Chu G. (2002) Diagnosis of multiple cancer types by shrunken centroids of gene expression. *Proc Natl Acad Sci* 99(10), 6567–6572
30. Yang D, Zhang D, Chen L, Qu B. (2015) Nantelescope: monitoring and visualizing large-scale collective behavior in lbsns. *J Netw Comput Appl* 55:170–180

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

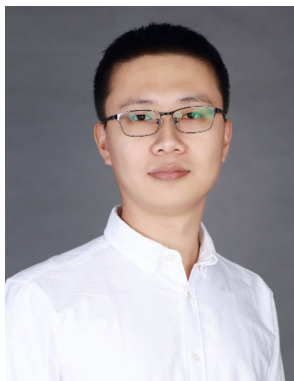


Sijia Yang received MSc in Information, Communications and Technology Business Management from Telecom Ecole de Management, Paris, France, 2015, and Bachelor of Engineering Degree from Zhejiang Gongshang University, Zhejiang, China, 2011. She is currently working towards her PhD degree in Cyberspace Security in Beijing University of Posts and Telecommunications, Beijing, China. Her research interests include cyber security, data analytics, and machine learning.



Haoyi Xiong received the Ph.D. degree in computer science from Telecom SudParis jointly with Universite Pierre et Marie Curie, Paris, France, in 2015. From 2016 to 2018, he was a Tenure-Track Assistant Professor with the Department of Computer Science, Missouri University of Science and Technology, Rolla, MO, USA (formerly known as University of Missouri at Rolla). From 2015 to 2016, he was a Research Associate with the Department of Systems and Information Engineering, University of Virginia, Charlottesville, VA, USA. He joined Big Data Laboratory, Baidu Research, Beijing, China, in 2018 as a Staff R&D Engineer and Research Scientist, where he is currently a Principal R&D Architect and Research Scientist. He also holds an honorary appointment as a Graduate Faculty Scholar affiliated to the ECE PhD Program at University of Central Florida, Orlando FL, USA. His current research interests include automated deep learning (AutoDL), ubiquitous computing, artificial intelligence, and cloud computing. He has published more than 70 papers in top computer science conferences

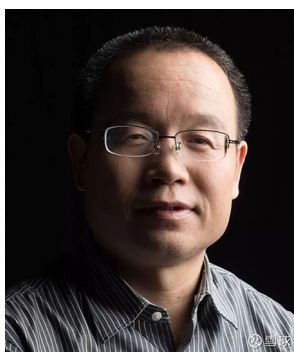
and journals.



Di Hu assistant professor at Gaoling School of Artificial Intelligence, Renmin University of China. His research interests include multimodal perception and learning. He has published more than 20 peer-reviewed top conference and journal papers, including NeurIPS, CVPR, ICCV, ECCV, etc. He served as a PC member of several top-tier conferences. Di is the recipient of the Outstanding Doctoral Dissertation Award by the Chinese Association for Artificial Intelligence, also the recipient of ACM XI'AN Doctoral Dissertation Award.



Kaibo Xu received his Bachelor degree (1998) in Computer Science from Beijing University of Chemical Technology and his Master (2005) and PhD (2010) in Computer Science from the University of the West of Scotland. He worked as a Teaching Assistant (1998–2004), Lecturer (2004–2009), Associate Professor (2009–2017) at Beijing Union University. He has supervised more than 20 master and doctoral students who are successful in their academic and industrial careers. As the principal investigator, he has received 7 governmental funds and 5 industrial funds with the total amount of 5M in the Chinese dollar. Dr. Kaibo Xu has also consulted extensively and been involved in many industrial projects. He worked as the Chief-Information-Officer (CIO) of Yunbai Clothing Retail Group, China (2016–2019). Currently, he is serving as the vice president and principal scientist of Mininglamp Tech. His research interests include graph mining, knowledge graph and knowledge reasoning.



Licheng Wang received the B.S. degree in engineering from Northwest Normal University, Lanzhou, China, in 1995, the M.S. degree in mathematics from Nanjing University, Nanjing, China, in 2001, and the Ph.D. degree in engineering from Shanghai Jiao Tong University, Shanghai, China, in 2007. He is currently the Full Professor with Beijing University of Posts and Telecommunications, Beijing, China. His current research interests include cryptography, blockchain, and future Internet architecture




Peizhen Zhu is an Assistant Teaching Professor in the Computer Science Department at Missouri University of Science and Technology, Rolla, MO, USA. She holds a Ph.D. in Computational Mathematics from the University of Colorado. Her research interests span a range of areas related to matrix computations, including numerical linear algebra, numerical analysis, optimization, graph algorithms, data mining, eigenvalue, and model predictive control.



Zeyi Sun received the B.Eng. degree in material science and engineering from Tongji University, Shanghai, China, in 2002, the M.Eng. degree in manufacturing from the University of Michigan Ann Arbor, Ann Arbor, MI, USA, in 2010, and the Ph.D. degree in industrial engineering and operations research from the University of Illinois at Chicago, Chicago, IL, USA, in 2015. He served as an Assistant Professor with the Department of Engineering Management and Systems Engineering, Missouri University of Science and Technology, Rolla, MO, USA, from 2015 to 2020. Currently, he is a senior research scientist with Mininglamp Academy of Sciences, Mininglamp Technology, Beijing, China. His research interest is mainly focused on using reinforcement learning algorithms to solve dynamic decision-making problem formulated by the Markov Decision Process.

Authors and Affiliations

Sijia Yang¹ · Haoyi Xiong² · Di Hu³ · Kaibo Xu⁴ · Licheng Wang¹ · Peizhen Zhu⁵ · Zeyi Sun⁴ 

✉ Licheng Wang
wanglc@bupt.edu.cn

✉ Zeyi Sun
sunzeyi@mininglamp.com

Sijia Yang
annieyang@bupt.edu.cn

Haoyi Xiong
haoyi.xiong.fr@ieee.org

Di Hu
dihu@ruc.edu.cn

Kaibo Xu
xukaibo@mininglamp.com

Peizhen Zhu
zhupe@mst.edu

- ¹ School of Cyberspace Security & State Key Laboratory of Networking and Switching, Beijing University of Posts and Telecommunications, Haidian, Beijing, China
- ² Big Data Lab, Baidu Research, Beijing, China
- ³ Gaoling School of Artificial Intelligence, Renmin University of China, Beijing, China
- ⁴ Mininglamp Academy of Sciences, Mininglamp Technology, Beijing 100084, China
- ⁵ Department of Computer Sciences, Missouri University of Science and Technology, Rolla, MO, USA