On Mean-Optimal Robust Linear Discriminant Analysis

Xiangyu Li
Department of Computer Science
Colorado School of Mines
Golden, Colorado, USA
lixiangyu@mines.edu

Hua Wang
Department of Computer Science
Colorado School of Mines
Golden, Colorado, USA
huawangcs@gmail.com

2022 IEEE International Conference on Data Mining (ICDM) | 978-1-6654-5099-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICDM54844.2022.00129

***Abstract*—Linear discriminant analysis (LDA) is widely used for dimensionality reduction under supervised learning settings. Traditional LDA objective aims to minimize the ratio of squared Euclidean distances that may not perform optimally on noisy data sets. Multiple robust LDA objectives have been proposed to address this problem, but their implementations have two major limitations. One is that their mean calculations use the squared ℓ_2-norm distance to center the data, which is not valid when the objective does not use the Euclidean distance. The second problem is that there is no generalized optimization algorithm to solve different robust LDA objectives. In addition, most existing algorithms can only guarantee the solution to be locally optimal, rather than globally optimal. In this paper, we review multiple robust loss functions and propose a new and generalized robust objective for LDA. Besides, to better remove the mean value within data, our objective uses an optimal way to center the data through learning. As one important algorithmic contribution, we derive an efficient iterative algorithm to optimize the resulting non-smooth and non-convex objective function. We theoretically prove that our solution algorithm guarantees that both the objective and the solution sequences converge to globally optimal solutions at a sub-linear convergence rate. The experimental results demonstrate the effectiveness of our new method, achieving significant improvements compared to the other competing methods.**

***Index Terms*—Linear Discriminant Analysis, Generalized Loss Function, Robustness, Non-convex Optimization, Global Convergence, Mean-Optimal.**

I. Introduction

Dimensionality reduction plays an important role in pattern classification and has been successfully applied to many real-world applications, such as social media, business, computer vision, bioinformatics, *etc*. [1]. Among the most widely used supervised dimensionality reduction methods is the Linear Discriminant Analysis (LDA) that aims to find the optimal projection by simultaneously maximizing the between-class variance and minimizing the within-class variance in the projected subspace. The algorithm of solving the traditional LDA objective is to optimize a trace ratio problem that can be transformed into a tractable ratio trace form [2]. This transformation allows us to easily find a closed-form solution by using the generalized eigenvalue decomposition method [3]. Unfortunately, this solution often deviates from the original objectives when outlier samples are present, which can result in uncertainties within subsequent classification procedures [2]. Another problem with this method is that inverting a singular within-class variance matrix can cause the LDA formulation to be ill-conditioned, and in fact this is common when the training data are sampled from high-dimensional space or there exist strong-correlated features. The efforts to overcome these issues by optimizing the ratio trace transformation cannot thoroughly solve this singularity problem. Motivated to tackle the original trace ratio problem, many existing works [4]–[6] studied how to directly optimize the trace-ratio objective function.

One major limitation of the traditional LDA is its dependency upon the squared ℓ_2-norm (ℓ_2^2) distance that is highly vulnerable to the presence of outliers or noisy data. This is because the least square criterion can remarkably enlarge the effect of noise on the total loss. Unfortunately, noise appears more or less everywhere in real-world applications. Therefore, it is critical to improve the robustness of the model when random noise or even adversarial attack are present in the data. In contrast to regular noises that are usually small in magnitude and sometimes even invisible to human eyes, adversarial attacks are intentionally constructed by injecting small perturbations into the original data such that the model can be misled to output discouraged results [7]. A more recently used adversarial attacks for modeling real-world data are the adversarial patch attack that couples the original data with a human-manipulated patch [8]. The effectiveness of such attacks can be explained in terms of the black-box principle where the attacker knows nothing about the configurations of the studied model.

To enhance the robustness against noise, many LDA objectives changed to use other distance functions rather than the ℓ_2^2-norm distance. Wang *et al*. [9] proposed to measure the two variance matrices using a rotational invariant ℓ_1-norm. However, the runtime of this method is as high as the intractable memory cost of storing the representations because the greedy learning strategy proceeds one projection vector at a time. Tao *et al*. [10] proposed to select the discriminative patterns by using a $\ell_{2,1}$-norm regularization that is a row-sparsity constraint on the transformation matrix. Since this method fails to uncover the most important features in the given tasks and the learned representations are largely governed by the selected number of dimensions. Wen *et al*.

978-1-6654-5099-7/22/$31.00 ©2022 IEEE
DOI 10.1109/ICDM54844.2022.00129

[11] introduced an orthogonal matrix to connect the original features and transformed features, in which the main information of the original data can persevere in the discriminant subspace. However, their optimization methods easily fall into locally optimal solutions. Zheng *et al*. [12] tried to build a ℓ_1-norm discriminant analysis (L1-LDA) using Bayes error bound optimization, but the implementation of the Bayes error bound is not easy as it requires each class to contain the same number of samples.

Recently, there are also several efforts that focus on how to develop robust lost functions [13]–[15]. Although these methods are able to improve the robustness against noisy data, their implementations are usually limited to two notorious problems. One is that some optimization algorithms just simply guarantee the decrease of the objective value, but not the algorithm convergence by nature. Another problem is that the derived solutions using their methods can easily get stuck in local minima rather than landing exactly on the global minima. Moreover, these loss functions were developed with their particular properties, each of which has its own independent variables to adjust the robustness. Hence, in this paper, we aim to provide a novel and robust objective for LDA that has better generalization with a shared tractable and efficient optimization algorithm.

We also noticed that most existing robust LDA objectives do not tackle the mean calculation in a correct way. In many cases, the mean of the data is not zero, and using ℓ_2^2-based distance to calculate the mean value is incorrect when the objective depends on other norm based distance [16]. In this paper, we thus do not only propose a novel generative LDA model that integrates a very broad family of robust loss functions, but also embed the mean calculation into the objective function that can remove the optimal mean automatically during learning time. Since existing optimization algorithms are not able to well handle our generative objective with the non-smooth and non-convex property, as one important algorithmic contribution, we further derive an efficient solution algorithm that can guarantee both the objective and sequence convergences. Through our mathematical analysis, our proposed algorithm is proved to find a globally optimal solution at a sub-linear rate. In our extensive experiments, our proposed method outperforms several competitive dimensionality reduction approaches and has a great advantage of handling various noisy date or adversarial attacks.

II. Formulation and Algorithm

Given the training dataset of $X \in \mathbb{R}^{d\times n}$, we can represent it using a few sub-design matrices $X = [X_1, ..., X_c]$, where $X_i = [x_1^i, ..., x_{n_i}^i] \in \mathbb{R}^{d\times n_i} (1 \leq i \leq c)$ corresponds to a collection of data points belonging to the i-th class. The traditional binary-class LDA method assumes that, given data points $X_i = [x_1^i, ..., x_{n_i}^i]$ in the i-th class and data points $X_j = [x_1^j, ..., x_{n_j}^j]$ in the j-th class, both multivariate probability densities are multivariate Gaussian distributions with arbitrary mean vectors μ and a covariance matrix Σ, that is: $p(x|\mu_i, \Sigma_i) = \frac{e^{\left(-\frac{1}{2}[x-\mu_i]^T\Sigma_i^{-1}[x-\mu_i]\right)}}{(2\pi)^{n/2}|\Sigma_i|^{1/2}}$, and $p(x|\mu_j, \Sigma_j) = \frac{e^{\left(-\frac{1}{2}[x-\mu_j]^T\Sigma_j^{-1}[x-\mu_j]\right)}}{(2\pi)^{n/2}|\Sigma_j|^{1/2}}$, where μ_i is the mean value of data points in class i, and μ is the mean value of the whole data points. Due to the homogeneity assumption that $\Sigma_i = \Sigma_j = \Sigma_{xx}$, the ratio of the two densities can be written as:

$$\frac{p(x|\mu_i, \Sigma_{xx})}{p(x|\mu_j, \Sigma_{xx})} = \frac{e^{\left(-\frac{1}{2}[x-\mu_i]^T\Sigma_{xx}^{-1}[x-\mu_j]\right)}}{e^{\left(-\frac{1}{2}[x-\mu_j]^T\Sigma_{xx}^{-1}[x-\mu_j]\right)}}. \tag{1}$$

Since the logarithms is a monotonically increasing function, Eq. (1) follows that:

$$\begin{aligned}\log_e \frac{p(x|\mu_i, \Sigma_{xx})}{p(x|\mu_j, \Sigma_{xx})} &= [\mu_i - \mu_j]^T\Sigma_{xx}^{-1}x \\ -\frac{1}{2}[\mu_i - \mu_j]^T\Sigma_{xx}^{-1}[\mu_i + \mu_j] &= [\mu_i - \mu_j]^T\Sigma_{xx}^{-1}(x - \mu),\end{aligned} \tag{2}$$

where $\mu = \frac{\mu_i+\mu_j}{2}$. Also, assume that given a randomly selected data point x belonging to the i-th class, we have the prior probability:$P(x \in X_i) = \pi_i$. Using the Baye's theorem, we have the logarithms of the ratio of the posterior probabilities:

$$L(x) = \log_e \left(\frac{p(x|\mu_i, \Sigma_{xx})\pi_i}{p(x|\mu_j, \Sigma_{xx})\pi_j}\right) = w_0 + w^T x, \tag{3}$$

where $w = (\Sigma_{xx}^{-1})(\mu_i - \mu_j)$ and $w_0 = -\frac{1}{2}(\mu_i^T\Sigma_{xx}^{-1}\mu_i - \mu_j^T\Sigma_{xx}^{-1}\mu_j) + \log_e(\frac{\pi_j}{\pi_i})$. Since the function $L(x)$ is a linear function of x, the classes i and j can be separated from the linear combination, $w^T X$, of the data matrix X. In other words, the objective of binary-class Linear discriminant analysis (LDA) is to find a projection w that can maximize the squared distance of the two class means, $w \propto \Sigma_{xx}^{-1}(\mu_i - \mu_j)$, comparing to the within-class variation of that distance. From this point of view, in multiclass LDA, we define the between-class scatter matrix as $S_b = \sum_{i=1}^c n_i(\mu_i - \mu)(\mu_i - \mu)^T$ and within-class scatter matrix as $S_w = \sum_{i=1}^c \sum_{j=1}^{n_i}(x_j^i - \mu_i)(x_j^i - \mu_i)^T$, where n_i is the total number of data points in class i, μ_i is the mean value of data points in class i, and μ is the mean value of the whole data points. The objective of multiclass Linear Discriminant Analysis is to find a transformation matrix $W \in \mathbb{R}^{d\times m}$ that can minimize the squared distance of data points within the same class and maximize the squared distance of data points between different classes. We can thus represent the objective function of multiclass LDA as:

$$\max_{W^TW=I} \frac{Tr(W^TS_bW)}{Tr(W^TS_wW)} = \frac{\sum_{i=1}^c \|W^T(\mu_i - \mu)\|_F^2}{\sum_{i=1}^c \sum_{j=1}^{n_i} \|W^T(x_j^i - \mu_i)\|_F^2}. \tag{4}$$

Rather than measuring the between-class scatter matrix S_w, we also need to measure the total scatter matrix $S_t = \sum_{i=1}^n (x_i - \mu)(x_i - \mu)^T$, given the fact that $S_t = S_w + S_b$. The objective function in Eq. (4) can be rewritten as:

$$\min_{W^TW=I} \frac{Tr(W^TS_wW)}{Tr(W^TS_tW)} = \frac{\sum_{i=1}^c \sum_{j=1}^{n_i} \|W^T(x_j^i - \mu_i)\|_F^2}{\sum_{i=1}^n \|W^T(x_i - \mu)\|_F^2}. \tag{5}$$

However, this objective is challenging to optimize due to the fact that the numerator is a minimization problem while the denominator is a problem to maximize variance in the projected space, $\max_{W^TW=I} \sum_{i=1}^{n} \|W^T(x_i - \mu)\|_F^2$. A smart way of solving this challenge is to convert the maximization problem into a minimization problem by using the reconstruction errors [6], [17]. We can therefore change it into:

$$\min_{W^TW=I} \left\|X - \mu 1^T - WW^T(X - \mu 1^T)\right\|_F^2. \tag{6}$$

The ℓ_2^2-norm used in this objective is known as the Euclidean norm. In many contexts, the ℓ_2^2 is undesirable because of the great sensitivity to noisy data or outliers [18]. In addition, as we discussed in the linear model $L(x)$ in Eq. (3), this squared error is motivated by the data with a Gaussian prior. However, [19] proposed that if the linear model changes to build upon a Laplace prior, this method can provide a more robust algorithm and produce lower error rates than previous techniques with a Gaussian prior. With this recognition, it is easy to see that the problem in Eq. (6) with a Laplace prior distribution reduces to the ℓ_1-norm optimization problem. We also note that the fragile ℓ_2^2-norm distance can be replaced with multiple robust loss functions (for example, Nie adaptive loss function [13], Log-cosh loss function, Huber loss function [20], Cauchy loss function or Lorentzian function [21], or Barron adaptive loss function [14], *etc*., as shown in Table I, we propose to combine the LDA objective with such robust loss functions. In the most practical scenarios, optimization algorithms used for solving such robust loss functions confront several intractable challenges as discussed in the introduction section. We thus aim to present a generalized LDA formulation that consists of different loss functions, and meanwhile, this objective can be solved using a shared and tractable optimization algorithm. From this point of view, we define a generative loss function $L(\cdot)$ that can be replaced by a family of loss functions like $\ell_2, \ell_1,$ Huber, etc, as listed in out Table 1, all of which is required to fulfill an important property - Lipschitz continuous gradient. Following intuitions above, Eq. (6) can be changed into:

$$\min_{W^TW=I} \left\|X - \mu 1^T - WW^T(X - \mu 1^T)\right\|_{L(\cdot)}. \tag{7}$$

Likewise, the numerator in Eq. (5) can be changed into:

$$\min_{W^TW=I} \sum_{i=1}^{c} \sum_{j=1}^{n_i} \|W^T(x_j^i - \mu_i)\|_{L(\cdot)}. \tag{8}$$

We can now view Eq. (7) as a constrained condition to Eq. (8). Specifically, if $L = I$, this constraint changes back to the Euclidean norm. If $L = 0$, all data points in this constraint collapse into a single point, which is useless in real-world scenarios. To avoid this collapse problem, we change the formulation to an inequality constraint:

$$\left\|X - \mu 1^T - WW^T(X - \mu 1^T)\right\|_{L(\cdot)} \geq c, \tag{9}$$

where c is a positive constant. To obtain a more convenient but equivalent problem, we observe that taking any positive values of c just results in the L being replaced by c^2L, meaning that the inequality constraint in Eq. (9) can be simplified into an equality constraint [22]. We can now write our optimization problem as:

$$\begin{aligned} &\min_{W^TW=I} \sum_{i=1}^{c} \sum_{j=1}^{n_i} \|W^T(x_j^i - \mu_i)\|_{L(\cdot)}, \\ &s.t.\ \|(I - WW^T)(X - \mu 1^T)\|_{L(\cdot)} = 1. \end{aligned} \tag{10}$$

This constrained minimization problem can be solved by constructing a Lagrange function, which is given by:

$$\begin{aligned} \mathcal{L}(W, \mu_i, \mu, \lambda) &= \sum_{i=1}^{c} \|W^T(X_i - \mu_i 1_i^T)\|_{L(\cdot)} \\ &+ \lambda\Big(\left\|(I - WW^T)(X - \mu 1^T)\right\|_{L(\cdot)} - 1\ \Big). \end{aligned} \tag{11}$$

This leads to the optimization problem as below:

$$\min_{W^TW=I} \mathcal{L}(W, \mu_i, \mu, \lambda), \tag{12}$$

where the new variable λ is a multiplier for the constraint. It is worth noting that the terms μ and μ_i are often obtained by $\mu_i = \sum_{j=1}^{n_i} x_j^i$ and $\mu = \sum_{i=1}^{n} x_i$. However, the intuition behind this calculation depends on the Euclidean distance, and, as a result, can be invalid when the loss function is built upon some robust norm based distance, such as ℓ_1-norm or $\ell_{2,1}$-norm, according to [16]. This is because that the mean of a set of input data depends on the definition of distance. Different distance metrics/loss functions lead to different mean. Also, using different loss functions is equivalent to use to a different distance. At this point, the mean in our method is a variable, but not a fixed value like the traditional one which is defined for the ℓ_2-norm distance, since our objective is designed for representing a family of different robust loss functions. We thus propose a novel generative LDA objective function that can automatically remove the optimal mean from the data over the iterations.

Learning Eq. (12) can be difficult in practice as the objective is non-convex and non-smooth, which cannot admit closed form solutions. Rather than minimizing $\mathcal{L}$ with respect to the entire parameters simultaneously at each time step, we can think of minimizing $\mathcal{L}$ alternatively with respect to a single parameter at a time. For example, if we minimize $\mathcal{L}$ with respect to the parameter W, then minimize it with respect to μ and so on, repeatedly cycling through all parameters, we are guaranteed to obtain an optimal minimum. This practice is called as Alternating Minimization method (AM). As a result, when applying the AM to the objective in Eq. (12), we obtain the solution update rule as:

$$\begin{aligned} W_{(k+1)} &= \mathcal{L}(W, \mu_{i(k)}, \mu_{(k)}, \lambda_{(k)}), \quad W \in \mathbf{W}, \\ \mu_{i(k+1)} &= \mathcal{L}(W_{(k)}, \mu_i, \mu_{(k)}, \lambda_{(k)}), \quad \mu_i \in \mathbf{M_i}, \\ \mu_{(k+1)} &= \mathcal{L}(W_{(k)}, \mu_{i(k)}, \mu, \lambda_{(k)}), \quad \mu \in \mathbf{M}, \\ \lambda_{(k+1)} &= \mathcal{L}(W_{(k)}, \mu_{i(k)}, \mu_{(k)}, \lambda), \quad \lambda \in \mathbf{\Lambda}, \end{aligned} \tag{13}$$

where k is the time step index, and $\mathbf{W}, \mathbf{M_i}, \mathbf{M}, \mathbf{\Lambda}$ are the sequences of associated variables.

TABLE I: Types of loss functions

Loss function	Formula	Gradient
Squared ℓ_2-norm	$L(y,\hat{y}) = (y-\hat{y})^2$	$\nabla L = 2(y-\hat{y})$
ℓ_p*-norm* $(1<p<2)$	$L(y,\hat{y}) = (y-\hat{y})^p$	$\nabla L = p(y-\hat{y})^{p-1}$
Huber loss	$L(y,\hat{y}) = \begin{cases} \frac{1}{2}(y-\hat{y})^2 & \text{if } \lvert y-\hat{y}\rvert \le \delta, \\ \delta\lvert y-\hat{y}\rvert - \frac{1}{2}\delta^2 & \text{otherwise.} \end{cases}$	$\nabla L = \begin{cases} y-\hat{y} & \text{if } \lvert y-\hat{y}\rvert \le \delta, \\ \delta sgn(y-\hat{y}) & \text{otherwise.} \end{cases}$
Cauchy loss	$L(y,\hat{y}) = \log\big((\frac{y-\hat{y}}{c_0})^2+1\big)$	$\nabla L = \frac{2(y-\hat{y})}{c_0^2+(y-\hat{y})^2}$
Log-cosh loss	$L(y,\hat{y}) = \log\big(cosh(y-\hat{y})\big)$	$\nabla L = tanh(y-\hat{y}) = \frac{e^{2(y-\hat{y})}-1}{e^{2(y-\hat{y})}+1}$
Nie adaptive loss	$L(y,\hat{y}) = \frac{(1+\sigma)(y-\hat{y})^2}{\lvert y-\hat{y}\rvert+\sigma}$	$\nabla L = 2(1+\sigma)\frac{\lvert y-\hat{y}\rvert+2\sigma}{2(\lvert y-\hat{y}\rvert+\sigma)^2}(y-\hat{y})$
Barron adaptive loss	$L(y,\hat{y}) = \begin{cases} \frac{1}{2}\big(\frac{y-\hat{y}}{c_0}\big)^2 & \text{if } \gamma = 2, \\ \log\big(\frac{1}{2}(\frac{y-\hat{y}}{c_0})^2+1\big) & \text{if } \gamma = 0, \\ 1-\exp\big(-\frac{1}{2}(\frac{y-\hat{y}}{c_0})^2\big) & \text{if } \gamma = -\infty, \\ \frac{\lvert\gamma-2\rvert}{\gamma}\Big(\big(\frac{(\frac{y-\hat{y}}{c_0})^2}{\lvert\gamma-2\rvert}+1\big)^{\frac{\gamma}{2}}-1\Big) & \text{otherwise.} \end{cases}$	$\nabla L = \begin{cases} \frac{y-\hat{y}}{c_0^2} & \text{if } \gamma = 2, \\ \frac{2(y-\hat{y})}{(y-\hat{y})^2+2c_0^2} & \text{if } \gamma = 0, \\ \frac{y-\hat{y}}{c_0^2}\exp\big(-\frac{1}{2}(\frac{y-\hat{y}}{c_0})^2\big) & \text{if } \gamma = -\infty, \\ \frac{y-\hat{y}}{c_0^2}\big(\frac{(\frac{y-\hat{y}}{c_0})^2}{\lvert\gamma-2\rvert}+1\big)^{\frac{\gamma}{2}-1} & \text{otherwise.} \end{cases}$

The matrix factorization in Eq. (13) is still a non-convex problem due to the constraint on each sequence set. An efficient way that such a constraint can be used in the service of regularizing the optimization function is by adding a proximal regularization term to the update. This method has been used primarily in the context of nonconvex and nonsmooth problems [23]–[25]. Using the proximal terms introduced in the alternating linearized minimisation, we denote our regularized optimization problems as:

$$
\begin{aligned}
W_{(k+1)} &= \arg\min < \nabla\mathcal{L}(W_{(k)}), W - W_{(k)} > + \frac{\alpha_1}{2}\|W - W_{(k)}\|_F^2, \\
\mu_{i(k+1)} &= \arg\min < \nabla\mathcal{L}(\mu_{i(k)}), \mu_i - \mu_{i(k)} > + \frac{\alpha_2}{2}(\mu_i - \mu_{i(k)})^2, \\
\mu_{(k+1)} &= \arg\min < \nabla\mathcal{L}(\mu_{(k)}, \mu - \mu_{(k)} > + \frac{\alpha_3}{2}(\mu - \mu_{(k)})^2, \\
\lambda_{(k+1)} &= \arg\min < \nabla\mathcal{L}(\lambda_{(k)}), \lambda - \lambda_{(k)} > + \frac{\alpha_4}{2}(\lambda - \lambda_{(k)})^2,
\end{aligned}
\tag{14}
$$

where $\nabla\mathcal{L}(W_k)$ means the gradient of Eq. (11) at the point of $W_{(k)}$, and $< \nabla\mathcal{L}(W_{(k)}), W - W_{(k)} >$ means the dot product between the gradient $\nabla\mathcal{L}(W_{(k)})$ and the offset $(W - W_{(k)})$. Another important advantage of this method of adding the proximal term to optimization problem is that the regularization terms can avoid the updated solution being changed to a value that differs significantly from the value of previous solutions.

A. Algorithm

We now discuss the solution algorithm in more details, with emphasis on its application to a variety of loss functions defined in the generative loss function. Table I shows a set of loss functions that can be incorporated into the generative loss function, including, but not limited to the ℓ_2^2-norm, Huber loss function [20], Nie adaptive loss function [13], Barron adaptive loss function [14], *etc*. Many extensions to this generative loss function are possible, as long as they remain the Lipschitz continuous gradient. To give a sense of how to update the solution given in Eq. (14), we need to define two notations by simplifying $A_i = W^T(X_i - \mu_i 1_i^T)$ and $B = (I - WW^T)(X - \mu 1^T)$. The gradient of A_i and B with respect to a variable is written as $\mathrm{grad}(A_i)$ and $\mathrm{grad}(B)$, each of which is derived using the element-wise product. For example, suppose we want to train our objective that incorporates Barron adaptive loss function [14] with the use of $\lambda = 0$, the general gradient listed in Table I (defined as $\mathrm{grad}(Y)$ in this case) can be written as $\mathrm{grad}(Y) = 2Y \oslash (Y \circ Y + 2c_0^2 1^T)$, where $\oslash$ is the Hadamard division, $\circ$ is the Hadamard product.

Step 1: Updating $W_{(k+1)}$

$$
\begin{aligned}
W_{(k+1)} &= \arg\min_{W^TW=I} < \nabla\mathcal{L}(W_{(k)}), W - W_{(k)} > \\
&\quad + \frac{\alpha_1}{2}\|W - W_{(k)}\|_F^2 = \arg\max_{W^TW=I} Tr[W^T M_{(k)}],
\end{aligned}
\tag{15}
$$

where

$$
\begin{aligned}
M_{(k)} =& \alpha_1 W_{(k)} + \lambda\ \mathrm{grad}(B)X^TW + X\,\mathrm{grad}(B^T)W \\
&+ \sum_{i=1}^{c}(X_i - \mu_i 1_i^T)\,\mathrm{grad}(A_i^T).
\end{aligned}
\tag{16}
$$

We can easily obtain this solution by finding the nearest orthogonal matrix to the given matrix $M_{(k)}$, known as the orthogonal Procrustes problem [26],

$$
W_{(k+1)} = \arg\max_{W^TW=I} Tr[W^T M_{(k)}] = P_1 P_2^T, \tag{17}
$$

where P_1 and P_2 are from the singular value decomposition $M = P_1 \Sigma P_2^T$.

Step 2: Updating $\mu_{i(k+1)}$

$$
\begin{aligned}
\mu_{i(k+1)} &= \arg\min_{W^TW=I} < \nabla\mathcal{L}(\mu_{i(k)}), \mu_i - \mu_{i(k)} > + \frac{\alpha_2}{2}(\mu_i - \mu_{i(k)})^2 \\
&= \arg\min_{W^TW=I} \frac{\alpha_2}{2}(\mu_i - \mu_{i(k)})^2 + (\mu_i - \mu_{i(k)})\Psi_1[\mu_{i(k)}],
\end{aligned}
\tag{18}
$$

where $\Psi_1[\mu_{i(k)}] = Tr\left(\mathrm{grad}(A_i^T)(-W^T 1_i^T)\right)$.

Setting its derivative equal to 0 we have the solution:$\mu_{i(k+1)} = \mu_{i(k)} - \frac{1}{\alpha_2}\Psi_1[\mu_{i(k)}]$.

Step 3: Updating $\mu_{(k+1)}$

$$\begin{aligned}\mu_{(k+1)} &= \underset{W^TW=I}{\arg\min} < \nabla\mathcal{L}(\mu_{(k)}), \mu - \mu_{(k)} > + \frac{\alpha_3}{2}\|\mu - \mu_{(k)}\|_F^2 \\ &= \underset{W^TW=I}{\arg\min} \frac{\alpha_3}{2}(\mu - \mu_{(k)})^2 + (\mu - \mu_{(k)})\Psi_2[\mu_{(k)}],\end{aligned} \tag{19}$$

where $\Psi_2[\mu_{(k)}] = \lambda Tr\left(\text{grad}(B)(WW^T1^T - 1^T)^T\right)$.

Setting its derivative equal to 0 we have the solution: $\mu_{k+1} = \mu_{(k)} - \frac{1}{\alpha_3}\Psi_2[\mu_{(k)}]$.

Step 4: Updating $\lambda_{(k+1)}$

$$\begin{aligned}\lambda_{(k+1)} &= \underset{W^TW=I}{\arg\min} < \nabla\mathcal{L}(\lambda_{(k)}), \lambda - \lambda_{(k)} > + \frac{\alpha_4}{2}(\lambda - \lambda_{(k)})_F^2, \\ &= \underset{W^TW=I}{\arg\min} \frac{\alpha_4}{2}(\lambda - \lambda_{(k)})^2 + (\lambda - \lambda_{(k)})\Psi_3[\lambda_{(k)}],\end{aligned} \tag{20}$$

where $\Psi_3[\lambda_{(k)}] = \|B\|_{Barron} - 1$.

The solution can be obtained by setting its derivative equal to 0: $\lambda_{k+1} = \lambda_{(k)} - \frac{1}{\alpha_4}\Psi_3[\lambda_{(k)}]$.

Algorithm 1: Proximal Alternating Linearized Minimization

Input: data $X \in \mathbb{R}^{d\times n}$, number of clusters c, hyperparameters λ, α_1, α_2, α_3, α_4, number of iterations K.
Initialization: projection matrix $W \in \mathbb{R}^{d\times m}$, mean values μ_i, μ.
while $k \leq K$ **do**
  optimize $W_{(k+1)}$ as **Step 1**;
  optimize $\mu_{i_{(k+1)}}$ as **Step 2**;
  optimize $\mu_{(k+1)}$ as **Step 3**;
  optimize $\lambda_{(k+1)}$ as **Step 4**;
end

The solution algorithm to solve the generative LDA objective in Eq. (10) by using the proximal alternating linearized minimisation is summarized in Algorithm 1. It is worth noting that our new solution algorithm guarantees that both the objective and the solution sequences converge to the globally optimal solution at a sub-linear convergence rate. Due to space limitation, the proof of the convergence of Algorithm 1 will be provided in the extended journal version of this paper.

III. Experiment

We evaluated the performance of our model on several real data benchmarks including COIL20, FERET, USPS, MNIST, and Olivetti. [27]–[29]. As our model aims to improve the model robustness against noise, we were motivated to design a few experiments with noisy data, in which block disturbance or Gaussian noise was added into the input data. Several state-of-the-art methods were also implemented for comparison, involving RSLDA [11], $\ell_{2,1}$-LDA [30], GLDA [31], LDA, stacked Restricted Boltzmann machine (RBM) [32], and Stacked AutoEncoder (SAE) network [33].

Datasets descriptions. The COIL20 dataset contains 1440 images of 20 subjects, in which each subject has 72 images taken at pose intervals of 5 angle degree. The size of each image is 128×128. The FERET dataset contains more than $14,000$ facial images of 1199 individuals. Since the main purpose of our experiment is to test the object recognition against noise attacks, we randomly selected a subset of the FERET dataset for our experiment. This gives rise to 1400 images of 200 individuals, each of which has 7 face images. The size of the image was converted to 60×60 pixels. The USPS image dataset is commonly used for handwritten text recognition research. This dataset contains a total of $9,298$ samples and each sample has 16×16 greyscale pixels. The MNIST dataset is another large database of handwritten digits that contains $70,000$ samples. Each greyscale image is 28×28, representing the digits $0-9$. The Olivetti (or AT&T) dataset consists of 400 different images of 40 distinct individuals. Each individual has 10 different images taken at different conditions, and each image was resized to 64×64 pixels.

Experimental results and analysis. We compared our model against seven other methods involving RSLDA [11], $\ell_{2,1}$-LDA [30], GLDA [31], LDA, stacked Restricted Boltzmann machine (RBM) [32], and Stacked AutoEncoder (SAE) network [33]. In this experiment, we demonstrate the robustness of our model on different noisy data sets. The loss function we used in our model is Nie and Barron adaptive loss function, respectively. Our training procedures were implemented using the stratified 6-fold cross-validation that partitions data into k non-overlapping folds to fit the train/test set by preserving the percentages of samples for each class. We repeated the stratified 6-fold cross-validation six times and then reported the mean performance across all folds and all repeats. See Table II for a comparison of the performance of our method and seven other models on noisy datasets. Note that the noisy data used in the our tables were formed by adding Gaussian noise.In these cases, the Gaussian noise was randomly distributed with the variance of 0.05. The dimensionality of the above data was reduced to a fixed level that is 19, 25, 9, 9, and 25 for COIL20, FERET, USPS, MNIST, and Olivetti datasets, respectively. The table report the average classification accuracy, precision, and recall scores over 50 runs with their standard deviations. We can see that our model clearly outperforms other competitive methods.

IV. Conclusion

We propose a generative LDA learning method in which multiple robust loss functions can be interchangeable. Also, to solve the singularity problem in a natural way, our objective depends on the trace-ratio formulation that can avoid the inversion of the within-class variance matrix. In addition, our objective automatically center the data by quantifying the optimal mean value during the learning process. We further introduce an effective algorithm to solve the resulting non-convex and non-smooth problem. Through our mathematical analysis, we conclude that our optimization algorithm can find globally optimal solutions with guaranteed convergences of both the objective value and the solution sequence, and the converging rate is as fast as sub-linearity. The experimental results on several real-world datasets shows that our model consistently outperformed its state of the art counterparts.

TABLE II: Classification performance of different methods on the noiseless data and Gaussian noise (var=0.04) data. The polynomial SVM algorithm was used as our classifier. The evaluation metric consists of the average recognition accuracy.

Accuracy (noiseless)	SVM	LDA	GLDA	RSLDA	$\ell_{2,1}$-LDA	SAE	RBM	Ours (Nie)	Ours (Barron)
Olivetti	0.598 ± 0.054	0.498 ± 0.079	0.601 ± 0.056	0.662 ± 0.052	0.571 ± 0.024	0.567 ± 0.032	0.611 ± 0.044	0.785 ± 0.021	**0.788 ± 0.023**
COIL20	0.541 ± 0.054	0.658 ± 0.057	0.671 ± 0.048	0.674 ± 0.050	0.550 ± 0.021	0.761 ± 0.038	0.687 ± 0.041	**0.859 ± 0.030**	0.843 ± 0.022
FERET	0.679 ± 0.093	0.507 ± 0.071	0.548 ± 0.051	0.603 ± 0.048	0.623 ± 0.027	0.681 ± 0.030	0.674 ± 0.042	0.707 ± 0.022	**0.710 ± 0.027**
USPS	0.484 ± 0.035	0.488 ± 0.025	0.515 ± 0.034	0.432 ± 0.039	0.497 ± 0.030	0.520 ± 0.027	0.523 ± 0.036	0.630 ± 0.026	**0.630 ± 0.031**
MNIST	0.401 ± 0.002	0.404 ± 0.010	0.477 ± 0.061	0.521 ± 0.044	0.515 ± 0.031	0.481 ± 0.027	0.507 ± 0.004	0.631 ± 0.041	**0.638 ± 0.033**
Accuracy (noisy)	SVM	LDA	GLDA	RSLDA	$\ell_{2,1}$-LDA	SAE	RBM	Ours (Nie)	Ours (Barron)
Olivetti	0.334 ± 0.021	0.321 ± 0.044	0.338 ± 0.059	0.336 ± 0.078	0.433 ± 0.067	0.401 ± 0.085	0.398 ± 0.088	0.525 ± 0.044	**0.527 ± 0.059**
COIL20	0.341 ± 0.064	0.342 ± 0.040	0.348 ± 0.032	0.442 ± 0.051	0.511 ± 0.026	0.564 ± 0.019	0.508 ± 0.052	0.621 ± 0.037	**0.622 ± 0.071**
FERET	0.302 ± 0.019	0.278 ± 0.034	0.253 ± 0.011	0.315 ± 0.041	0.398 ± 0.033	0.371 ± 0.012	0.316 ± 0.025	0.542 ± 0.015	**0.564 ± 0.024**
USPS	0.254 ± 0.055	0.279 ± 0.065	0.262 ± 0.002	0.268 ± 0.051	0.329 ± 0.027	0.309 ± 0.039	0.378 ± 0.024	**0.438 ± 0.017**	0.434 ± 0.020
MNIST	0.298 ± 0.031	0.274 ± 0.026	0.300 ± 0.073	0.303 ± 0.009	0.300 ± 0.026	0.310 ± 0.034	0.309 ± 0.046	0.433 ± 0.256	**0.437 ± 0.044**

Both theoretical analysis and empirical results indicate a great advantage of our model on the discriminant projections.

ACKNOWLEDGMENT

Corresponding author: Hua Wang (huawangcs@gmail.com).

This work was supported in part by the National Science Foundation (NSF) under the grants of IIS 1652943, IIS 1849359, CNS 1932482 and CCF 2029543.

REFERENCES

[1] G. T. Reddy, M. P. K. Reddy, K. Lakshmanna, R. Kaluri, D. S. Rajput, G. Srivastava, and T. Baker, "Analysis of dimensionality reduction techniques on big data," *IEEE Access*, vol. 8, pp. 54 776–54 788, 2020.

[2] H. Wang, S. Yan, D. Xu, X. Tang, and T. Huang, "Trace ratio vs. ratio trace for dimensionality reduction," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2007, pp. 1–8.

[3] K. Fukunaga, *Introduction to statistical pattern recognition*. Elsevier, 2013.

[4] H. Wang, F. Nie, and H. Huang, "Robust distance metric learning via simultaneous l1-norm minimization and maximization," in *International conference on machine learning*. PMLR, 2014, pp. 1836–1844.

[5] K. Liu, H. Wang, F. Nie, and H. Zhang, "Learning multi-instance enriched image representations via non-greedy ratio maximization of the ℓ_1-norm distances," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7727–7735.

[6] K. Liu, L. Brand, H. Wang, and F. Nie, "Learning robust distance metric with side information via ratio minimization of orthogonally constrained ell_{21}-norm distances," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, 2019.

[7] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, "Practical black-box attacks against machine learning," in *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, 2017, pp. 506–519.

[8] T. B. Brown, D. Mané, A. Roy, M. Abadi, and J. Gilmer, "Adversarial patch," *arXiv preprint arXiv:1712.09665*, 2017.

[9] H. Wang, X. Lu, Z. Hu, and W. Zheng, "Fisher discriminant analysis with l1-norm," *IEEE transactions on cybernetics*, vol. 44, no. 6, pp. 828–842, 2013.

[10] H. Tao, C. Hou, F. Nie, Y. Jiao, and D. Yi, "Effective discriminative feature selection with nontrivial solution," *IEEE transactions on neural networks and learning systems*, vol. 27, no. 4, pp. 796–808, 2015.

[11] J. Wen, X. Fang, J. Cui, L. Fei, K. Yan, Y. Chen, and Y. Xu, "Robust sparse linear discriminant analysis," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 2, pp. 390–403, 2018.

[12] W. Zheng, Z. Lin, and H. Wang, "L1-norm kernel discriminant analysis via bayes error bound optimization for robust feature extraction," *IEEE transactions on neural networks and learning systems*, vol. 25, no. 4, pp. 793–805, 2013.

[13] F. Nie, H. Wang, H. Huang, and C. Ding, "Adaptive loss minimization for semi-supervised elastic embedding," in *Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.

[14] J. T. Barron, "A general and adaptive robust loss function," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4331–4339.

[15] X. Wang, L. Bo, and L. Fuxin, "Adaptive wing loss for robust face alignment via heatmap regression," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 6971–6981.

[16] F. Nie, J. Yuan, and H. Huang, "Optimal mean robust principal component analysis," in *International conference on machine learning*. PMLR, 2014, pp. 1062–1070.

[17] K. Liu, H. Wang, F. Han, and H. Zhang, "Visual place recognition via robust ℓ_2-norm distance based holism and landmark integration," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 8034–8041.

[18] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *Journal of the ACM (JACM)*, vol. 58, no. 3, pp. 1–37, 2011.

[19] J. Goodman, "Exponential priors for maximum entropy models," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, 2004, pp. 305–312.

[20] P. J. Huber, "Robust estimation of a location parameter," in *Breakthroughs in statistics*. Springer, 1992, pp. 492–518.

[21] X. Li, Q. Lu, Y. Dong, and D. Tao, "Robust subspace clustering by cauchy loss function," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 7, pp. 2067–2078, 2018.

[22] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell, "Distance metric learning with application to clustering with side-information," in *NIPS*, vol. 15, no. 505–512. Citeseer, 2002, p. 12.

[23] R. T. Rockafellar, "Augmented lagrangians and applications of the proximal point algorithm in convex programming," *Mathematics of operations research*, vol. 1, no. 2, pp. 97–116, 1976.

[24] A. Kaplan and R. Tichatschke, "Proximal point methods and nonconvex optimization," *Journal of global Optimization*, vol. 13, no. 4, pp. 389–406, 1998.

[25] J. Bolte, S. Sabach, and M. Teboulle, "Proximal alternating linearized minimization for nonconvex and nonsmooth problems," *Mathematical Programming*, vol. 146, no. 1, pp. 459–494, 2014.

[26] P. H. Schönemann, "A generalized solution of the orthogonal procrustes problem," *Psychometrika*, vol. 31, no. 1, pp. 1–10, 1966.

[27] P. J. Phillips, H. Wechsler, J. Huang, and P. J. Rauss, "The feret database and evaluation procedure for face-recognition algorithms," *Image and vision computing*, vol. 16, no. 5, pp. 295–306, 1998.

[28] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[29] J. J. Hull, "A database for handwritten text recognition research," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 5, pp. 550–554, 1994.

[30] H. Zhao, Z. Wang, and F. Nie, "A new formulation of linear discriminant analysis for robust dimensionality reduction," *IEEE Transactions on Knowledge and data engineering*, vol. 31, no. 4, pp. 629–640, 2018.

[31] J. H. Oh and N. Kwak, "Generalization of linear discriminant analysis using lp-norm," *Pattern Recognition Letters*, vol. 34, no. 6, pp. 679–685, 2013.

[32] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

[33] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P.-A. Manzagol, and L. Bottou, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion." *Journal of machine learning research*, vol. 11, no. 12, 2010.