# Fast, Relaxation- and Hyperparameter-Free Pairwise Worst-Case Class Separation

**Mohammad Mahdi Omati**
Department of Electrical Engineering
Sharif University of Technology
mohammad_omati@yahoo.com

**Arash Amini**
Department of Electrical Engineering
Sharif University of Technology
aamini@sharif.edu

**Nezam Mahdavi-Amiri**
Department of Mathematical Sciences
Sharif University of Technology
nezamm@sharif.edu

## Abstract

In this paper, we investigate a novel discriminative dimensionality reduction method based on maximizing the minimum pairwise ratio of between-class to within-class scatter. This objective function enhances class separability by providing critical, adaptive control over the variance within each class pair. The resulting max-min fractional program is non-convex and challenging to solve. Our main contribution is $F^3$-PWCRA (Fast, Free of relaxation, Free of hyperparameters for Pairwise Worst-Case Ratio Analysis), a provably convergent two-level algorithm: an outer generalized Dinkelbach-type loop transforms the fractional objective into subtractive subproblems, ensuring the global convergence of the outer algorithm. Then, we propose a bisection strategy as an alternative, which offers robust convergence and straightforward implementation. For the inner loop, we develop an efficient minorization-maximization (MM) algorithm that tackles the non-convex subproblem by iteratively solving a simple quadratic program (QP), which we derive from the dual of a convex surrogate. $F^3$-PWCRA is computationally efficient, free from semidefinite relaxations and hyperparameter tuning, and extensive benchmarks show it consistently outperforms state-of-the-art methods in classification error.

## 1 Introduction

Modern data acquisition technologies have led to the proliferation of high-dimensional datasets, where the number of measured features can be exceptionally large. This phenomenon, often termed the "curse of dimensionality," introduces significant challenges, including increased computational overhead, a heightened risk of model overfitting due to spurious correlations, and a considerable loss of data interpretability. To address these issues in the context of supervised classification, dimensionality reduction has become an indispensable preprocessing step. Methodologies for dimensionality reduction are broadly categorized into two main paradigms: feature extraction, which constructs a new, smaller set of features by projecting the data onto a lower-dimensional manifold [1–7], and feature selection, which aims to identify and retain only the most informative subset of the original features [8–15]. Both approaches seek to produce a more compact and meaningful data representation, thereby enhancing classification accuracy, preventing overfitting, and improving the comprehensibility of the resulting model.

The core objective of discriminant analysis is to identify a linear projection that optimally separates distinct classes within a dataset. This family of supervised methods has evolved significantly since the

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

seminal work of Fisher, with researchers developing a wide array of class separability metrics over the decades [16–21, 1, 2]. Among these, linear discriminant analysis (LDA) remains the most prominent. Originally conceived by Fisher for binary classification [16] and later extended by Rao to handle multiple classes [17], LDA seeks a subspace projection that maximizes the ratio of between-class variance to within-class variance. Despite its widespread use, classical LDA exhibits several critical limitations. It is susceptible to performance degradation when sample sizes are small, which can yield unreliable covariance matrix estimates and violate its underlying Gaussian assumptions [22, 23]. It is also known to be sensitive to outliers [1]. A particularly significant drawback arises in multi-class scenarios where the target dimension is less than the number of classes minus one. In such cases, the LDA objective, which effectively averages pairwise class separations, becomes biased by the most distant class pair, causing less separated pairs to be projected even closer, potentially leading to class overlap [24]. This phenomenon is known as the "worst-case class separation" problem.

Several algorithms improve multiclass discrimination by adaptively weighting class distances, but they often falter on the closest pair of classes. More recently, researchers have reframed the problem in a worst-case light [25–33, 19, 34]. For example, Bian et al. [34] identify the pair of classes with the smallest distance between their means—the "worst-case" pair that nearly overlaps—and introduce a max–min distance analysis (MMDA) to enlarge that minimal gap in the reduced subspace. Zhang et al. build upon the MMDA with worst-case LDA to further enhance separation under challenging scenarios. Extensions in [21] and [25] move MMDA into heteroscedastic settings, using the Chernoff distance to maximize the smallest inter-class divergence while controlling the intra-class variance.

While these advanced methods improve upon classical LDA, strategies that focus solely on maximizing the minimum distance between class centroids (whether Euclidean or Chernoff-based) are incomplete. They fail to account for the internal dispersion, or within-class scatter, of each class. Consequently, even if the means of two classes are pushed apart, the classes themselves may still overlap if characterized by high variance. To address this interplay between separation and compactness, the worst-case ratio analysis (WCRA) objective was proposed [25, 2], which seeks to maximize the minimum ratio of between-class to within-class scatter.

The authors of [35] proposed a WCRA-based method that maximizes the minimum ratio of pairwise between-class separation with respect to the global within-class scatter matrix. Their approach is based on minorizing the objective and solving the surrogate iteratively, thereby tackling the problem directly and avoiding any relaxation that could lead to suboptimal solutions. Another advantage of their approach is that it is entirely hyperparameter-free (note that, as established in [2, 25, 35], the stopping tolerance is not classified as a hyperparameter in this context). However, by relying on a single global within-class scatter matrix, it does not assess the separability of each class pair relative to its own specific covariance structure. Consequently, its effectiveness is limited in scenarios with significantly heterogeneous class variances.

Wang et al. [2] addressed the limitation of the global scatter matrix by adopting pairwise normalization. Their design evaluates each class pair using its individual within-class scatter, thereby improving robustness to differing class variances. Nevertheless, their proposed algorithm, called worst-case discriminative dimension reduction (WDDR), introduces two notable drawbacks. First, it reformulates the non-convex fractional program into a sequence of quadratic subproblems and applies semidefinite relaxation to obtain a tractable semidefinite program (SDP). However, this relaxation is not guaranteed to be tight, potentially yielding suboptimal solutions that can undermine class separability and the effective handling of class variances. Second, its performance hinges on manually tuning at least two hyperparameters (excluding the stopping tolerance).

In this paper, we introduce F³-PWCRA (Fast, Free of relaxation, Free of hyperparameters for Pairwise Worst-Case Ratio Analysis), a novel and provably convergent iterative algorithm that overcomes the aforementioned limitations. Our approach solves a more adaptive pairwise WCRA formulation: for every class pair, separability is measured with respect to that pair's own within-class scatter matrices. This design naturally inherits the advantages of both [35] and [2], while addressing their respective shortcomings. To solve this problem, our $F^3$-PWCRA algorithm employs a nested iterative process. In the outer loop, we adapt a generalized Dinkelbach-type procedure to transform the challenging max–min fractional objective into a sequence of max–min subtractive subproblems. Theoretically, if the global optimum of each inner subproblem is found, this outer framework—which can be viewed as a specialized Newton's method—is guaranteed to converge to the global optimum of the original ratio. Alternatively, we implement the outer loop via a bisection search with mathematically

derived upper and lower bounds, offering robust convergence without manual tuning. The inner subproblem at each iteration remains a non-convex max–min program. To address this, we employ the minorization–maximization (MM) principle, constructing a tight convex surrogate that lower-bounds the objective to form an SDP at each step. While the MM procedure ensures convergence to a stationary point of the inner subproblem, we demonstrate that this dual-loop structure remains highly effective in practice. Crucially, we solve the inner SDP efficiently by formulating its dual as a simple quadratic program (QP). This overall framework makes F$^3$-PWCRA a robust, computationally efficient, and parameter-free algorithm (excluding the stopping tolerance).

## 2 Problem formulation

Consider a dataset $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$ containing $n$ samples distributed across $C$ distinct classes. The objective is to learn a linear transformation matrix $\mathbf{T} \in \mathbb{R}^{d\times m}$ (where $m \ll d$) that projects the high-dimensional data into a lower-dimensional subspace. The transformation is designed to simultaneously maximize between-class separation and minimize within-class dispersion.

To formalize this, we define class-pairwise scatter matrices. For any pair of classes $(i, j)$, the between-class scatter $\mathbf{S}_b^{ij}$ and within-class scatter $\mathbf{S}_w^{ij}$ are given by:

$$\mathbf{S}_b^{ij} = (\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_j)(\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_j)^T, \qquad \mathbf{S}_w^{ij} = \sum_{h\in\{i,j\}} \sum_{\mathbf{x}_k\in\tau_h} (\mathbf{x}_k - \bar{\mathbf{x}}_h)(\mathbf{x}_k - \bar{\mathbf{x}}_h)^T, \tag{1}$$

where $\tau_h$ denotes the set of samples belonging to the $h$-th class and $\bar{\mathbf{x}}_k$ is the corresponding class mean. Using these definitions, the PWCRA problem is formulated as the following optimization:

$$\max_{\mathbf{T}^T\mathbf{T}=\mathbf{I}_m} \min_{1\le i<j\le C} \frac{\operatorname{tr}\left(\mathbf{T}^T\mathbf{S}_b^{ij}\mathbf{T}\right)}{\operatorname{tr}\left(\mathbf{T}^T\mathbf{S}_w^{ij}\mathbf{T}\right)}. \tag{2}$$

The core of this formulation is the max-min objective, which guarantees pairwise class separability by maximizing the minimum performance ratio across all class pairs, directly addressing the "worst-case" separation problem. The use of pairwise scatter matrices enables adaptive handling of the unique covariance structure of each class pair—a key advantage over methods using a single global within-class scatter matrix. The orthogonality constraint $\mathbf{T}^\top\mathbf{T} = \mathbf{I}_m$ enforces that the basis vectors are orthonormal. As demonstrated in [16, 36, 21, 2], this property improves discriminative performance and maintains numerical stability in high-dimensional settings.

## 3 A Two-Level Optimization Algorithm for PWCRA

The PWCRA problem (2) is a challenging non-convex max-min fractional program. To solve it, we propose the **F$^3$-PWCRA** algorithm (see Algorithm 1), which utilizes a two-level iterative framework. The outer level transforms the fractional objective into an equivalent subtractive root-finding problem; this is then solved using a bisection search (or alternatively, a generalized Dinkelbach procedure). For each outer iteration, the resulting inner max-min subproblem is solved efficiently using an MM scheme (Algorithm 2), which handles the non-convexity through a sequence of tractable quadratic programs. The following subsections describe the derivation of each algorithmic level in detail.

### 3.1 Outer Loop: Generalized Dinkelbach Procedure for Fractional Programming

The PWCRA problem belongs to the class of general max-min ratio problems:

$$\max_{\mathbf{T}\in\mathcal{X}} \min_{1\le i<j\le C} \frac{f_{ij}(\mathbf{T})}{g_{ij}(\mathbf{T})}, \tag{3}$$

where $f_{ij}(\mathbf{T}) = \operatorname{tr}(\mathbf{T}^T\mathbf{S}_b^{ij}\mathbf{T})$, $g_{ij}(\mathbf{T}) = \operatorname{tr}(\mathbf{T}^T\mathbf{S}_w^{ij}\mathbf{T}) > 0$, and the feasible set is the Stiefel manifold $\mathcal{X} = \{\mathbf{T} \in \mathbb{R}^{d\times m} \mid \mathbf{T}^T\mathbf{T} = \mathbf{I}_m\}$. To solve this max-min fractional program, we generalize Dinkelbach's algorithm [37]. The theoretical foundation of this iterative approach is captured in the following theorems.

**Algorithm 1** $F^3$-PWCRA: Outer loop (Bisection)

**Require:** $\{\mathbf{S}_b^{ij}, \mathbf{S}_w^{ij}\}$, target dimension $m$, tolerance $\epsilon = 10^{-4}$.
1: Compute candidates $\delta_{u_1}, \delta_{u_2}, \delta_{u_3}$ via (5) and (6).
2: Set $\delta_u = \min\{\delta_{u_1}, \delta_{u_2}, \delta_{u_3}\}$ and $\delta_l = 0$.
3: **while** $\delta_u - \delta_l > \epsilon$ **do**
4: $\quad\delta^0 = (\delta_l + \delta_u)/2$.
5: $\quad$Solve (4) via Algorithm 2 for $\delta = \delta^0$.
6: $\quad$**if** (4) is feasible **then**
7: $\quad\quad$Save $\mathbf{T}_{\delta^0}$ and set $\delta_l = \delta^0$.
8: $\quad$**else**
9: $\quad\quad\delta_u = \delta^0$.
10: $\quad$**end if**
11: **end while**
12: Set $\delta^* = \delta_l$ and acquire the optimal $\mathbf{T}^*$.
13: **Output:** $\mathbf{T}^*$, $\delta^*$.

**Algorithm 2** $F^3$-PWCRA: Inner loop (MM-QP)

1: **Input:** Random $\mathbf{T}_{\text{init}}$ (projected to satisfy $(\mathbf{T}^0)^T\mathbf{T}^0 = \mathbf{I}_m$), $\{\tilde{\mathbf{S}}_{Cij}\}$, initial weights $\mathbf{z}^0$, tolerance $\epsilon$, $T_{\max}$.
2: **repeat**
3: $\quad\mathbf{A}_{ij} = \tilde{\mathbf{S}}_{Cij}\mathbf{T}^t, \quad c_{ij} = -\operatorname{tr}\left((\mathbf{T}^t)^T\tilde{\mathbf{S}}_{Cij}\mathbf{T}^t\right)$
4: $\quad\mathbf{A}(\mathbf{z}^t) = \sum_{1\le i<j\le C} z_{ij}^t\mathbf{A}_{ij}$
5: $\quad\mathbf{\Phi}^t = \left(\mathbf{A}(\mathbf{z}^t)^T\mathbf{A}(\mathbf{z}^t)\right)^{-\frac{1}{2}}$
6: $\quad$Construct $\mathbf{Q}$ via (21) and solve (20) for $\mathbf{z}^{t+1}$.
7: $\quad$Obtain $\mathbf{T}^{t+1}$ via (15).
8: $\quad t \leftarrow t+1$.
9: **until** $\frac{\|\mathbf{T}^t - \mathbf{T}^{t-1}\|}{\|\mathbf{T}^{t-1}\|} \le \epsilon$ or $t \ge T_{max}$
10: **Output:** $\mathbf{T}_{\delta^k} = \mathbf{T}^t$ (solution for the $k$-th bisection iteration).

**Theorem 3.1.** *The global optimum of the general max-min ratio problem* $\max_{\mathbf{T}\in\mathcal{X}} \min_{ij} \frac{f_{ij}(\mathbf{T})}{g_{ij}(\mathbf{T})}$ *is equivalent to finding the unique root* $\delta^*$ *of the function* $h(\delta) = \max_{\mathbf{T}\in\mathcal{X}} \min_{ij}\{f_{ij}(\mathbf{T}) - \delta g_{ij}(\mathbf{T})\}$ *(Proof in Appendix D).*

A summary of the generalized Dinkelbach procedure is outlined in Algorithm 3, which can be found in the Appendix E. Given the iterate $\mathbf{T}^k$, we compute the current worst-case ratio $\delta_k$. This $\delta_k$ is then used to form a subtractive max-min subproblem (line 4), the solution of which becomes the next iterate $\mathbf{T}^{k+1}$.

**Theorem 3.2.** *The iterative procedure of the generalized Dinkelbach algorithm is equivalent to Newton's method applied to find the root of the function* $h(\delta)$ *(Proof in Appendix E).*

The convergence properties of this algorithm are stated below, guaranteeing its desirable behavior.

**Theorem 3.3.** *The generalized Dinkelbach algorithm monotonically increases the objective value of the PWCRA problem at each iteration and converges to the global optimal solution (Proof in Appendix F).*

### 3.1.1 Bisection Search Strategy

Alternatively, the outer loop can be implemented via a bisection search, which offers robust convergence and straightforward implementation. Observing that the optimal value $\delta^*$ of (2) is the supremum of all $\delta$ for which the following feasibility problem admits a solution:

$$\begin{aligned} \text{Find} \quad & \mathbf{T} \in \mathbb{R}^{d\times m} \\ \text{s.t.} \quad & \mathbf{T}^T\mathbf{T} = \mathbf{I}_m, \\ & \min_{1\le i<j\le C} \operatorname{tr}\left(\mathbf{T}^T\left(\mathbf{S}_b^{ij} - \delta\mathbf{S}_w^{ij}\right)\mathbf{T}\right) \ge 0. \end{aligned} \tag{4}$$

If (4) is feasible for a given $\hat{\delta}$, then $\delta^* \ge \hat{\delta}$; otherwise, $\delta^* < \hat{\delta}$. This monotonic relationship enables efficient location of $\delta^*$ via bisection.

To initialize the procedure, we set the lower bound $\delta_l = 0$, which is always feasible (given a positive semi-definite numerator and positive definite denominator). For the upper bound $\delta_u$, we compute three candidates and take the tightest:

$$\delta_{u_1} = \min_{1\le i<j\le C} \lambda_{\max}\left(\mathbf{S}_b^{ij}, \mathbf{S}_w^{ij}\right), \qquad \delta_{u_2} = \lambda_{\max}\left(\sum_{i<j}\mathbf{S}_b^{ij}, \sum_{i<j}\mathbf{S}_w^{ij}\right), \tag{5}$$

$$\delta_{u_3} = \lambda_{\max}\left(\sum_{i<j}\alpha_{ij}\mathbf{S}_b^{ij}, \sum_{i<j}\alpha_{ij}\mathbf{S}_w^{ij}\right), \qquad \alpha_{ij} = \frac{1}{\lambda_{\max}\left(\mathbf{S}_b^{ij}, \mathbf{S}_w^{ij}\right)}. \tag{6}$$

We then set $\delta_u = \min\{\delta_{u_1},\ \delta_{u_2},\ \delta_{u_3}\}$. The rationale behind these three candidates is provided in Appendix C. At each iteration, we test feasibility at $\delta = (\delta_l + \delta_u)/2$ by solving the inner max-min subproblem described below. If feasible, we set $\delta_l = \delta$; otherwise, $\delta_u = \delta$. The procedure

terminates when $\delta_u - \delta_l \leq \epsilon$, and the optimal value is $\delta^* = \delta_l$. The bisection method converges in $\lceil \log_2((\delta_u - \delta_l)/\epsilon) \rceil$ iterations [37], independently of problem conditioning.

### 3.2 Inner Loop: Solving the Max-Min Subproblem via MM

Each outer loop iteration requires solving a problem of the form:

$$\max_{\mathbf{T}^T\mathbf{T}=\mathbf{I}_m} \min_{1\leq i<j\leq C} \operatorname{tr}\left(\mathbf{T}^T\tilde{\mathbf{S}}_{Cij}\mathbf{T}\right), \tag{7}$$

where $\tilde{\mathbf{S}}_{Cij} \triangleq \mathbf{S}_b^{ij} - \delta_k \mathbf{S}_w^{ij}$. To facilitate the development of our proposed algorithm for solving the problem in (7), we first introduce and prove a key lemma that requires the underlying matrices to be positive definite. It is worth noting that the potential indefiniteness of the matrices $\tilde{\mathbf{S}}_{ij}$ does not pose a problem. If $\tilde{\mathbf{S}}_{ij}$ is already positive semi-definite, we proceed directly; otherwise, each term can be rendered positive definite—and as a result, convexified—by shifting $\tilde{\mathbf{S}}_{Cij}$ with a suitable scalar matrix $\alpha\mathbf{I}$. Specifically, we can choose any desired scalar $\alpha$ satisfying $\alpha > \max_{i,j} \lambda_{\max}(-\tilde{\mathbf{S}}_{Cij})$. Due to the orthogonality constraint $\mathbf{T}^\top\mathbf{T} = \mathbf{I}_m$, this shift does not alter the optimizer. Moreover, enforcing this strict inequality guarantees that the shifted matrices are full-rank, which proves advantageous in subsequent steps. With the positive definiteness of the shifted matrices $\tilde{\mathbf{S}}_{Cij}$ established, we now present the following lemma.

**Lemma 3.1.** *Given that the matrices $\tilde{\mathbf{S}}_{Cij}$ are positive definite, the original non-convex semi-orthogonality constraint $\mathbf{T}^T\mathbf{T} = \mathbf{I}_m$ in problem* (7) *can be substituted with the relaxed condition $\mathbf{T}^T\mathbf{T} \preccurlyeq \mathbf{I}_m$. This relaxation is valid because the global maximizer of the resulting relaxed problem will inherently satisfy the original equality constraint (see proof in Appendix G).*

Leveraging Lemma 3.1, we can now reformulate the initial problem from (7) into the following relaxed optimization problem:

$$\max_{\mathbf{T}^T\mathbf{T}\preccurlyeq\mathbf{I}_m} \quad \min_{1\leq i<j\leq C} \quad \operatorname{tr}\left(\mathbf{T}^T\tilde{\mathbf{S}}_{Cij}\mathbf{T}\right). \tag{8}$$

A significant advantage of this reformulation is that the new constraint in (8) is convex, since $\mathbf{T}^T\mathbf{T} \preccurlyeq \mathbf{I}_m$ is equivalent to the LMI $\begin{bmatrix} \mathbf{I}_m & \mathbf{T}^T \\ \mathbf{T} & \mathbf{I}_d \end{bmatrix} \succcurlyeq 0$. However, the optimization in (8) is a challenging non-convex problem due to its max-min structure, where each term $h_{ij}(\mathbf{T}) = \operatorname{tr}(\mathbf{T}^T\tilde{\mathbf{S}}_{Cij}\mathbf{T})$ is convex in $\mathbf{T}$. To address (8), we employ the minorization-maximization for max-min (MM4MM) approach [38], an overview of which is detailed in Appendix A.

At each iteration $t$, we construct a tractable surrogate function by minorizing each of the convex quadratic terms, $h_{ij}(\mathbf{T})$. We replace each term with its first-order Taylor expansion around the current estimate $\mathbf{T}^t$. This tangent hyperplane serves as a tight lower bound:

$$h_{ij}(\mathbf{T}) = \operatorname{tr}\left(\mathbf{T}^T\tilde{\mathbf{S}}_{Cij}\mathbf{T}\right) \geq 2\operatorname{tr}\left(\left(\mathbf{T}^t\right)^T\tilde{\mathbf{S}}_{Cij}\mathbf{T}\right) - \operatorname{tr}\left(\left(\mathbf{T}^t\right)^T\tilde{\mathbf{S}}_{Cij}\mathbf{T}^t\right) \triangleq \tilde{h}_{ij}(\mathbf{T}). \tag{9}$$

Substituting these linear lower bounds $\tilde{h}_{ij}(\mathbf{T})$ back into the original problem (8) yields the surrogate problem for the MM update. This problem involves maximizing the minimum of these linear functions:

$$\max_{\mathbf{T}^T\mathbf{T}\preccurlyeq\mathbf{I}_m} \min_{1\leq i<j\leq C} \quad 2\operatorname{tr}\left(\mathbf{A}_{ij}^T\mathbf{T}\right) + c_{ij}, \tag{10}$$

where

$$\mathbf{A}_{ij}^T \triangleq \left(\mathbf{T}^t\right)^T\tilde{\mathbf{S}}_{Cij}, \quad c_{ij} = -\operatorname{tr}\left(\left(\mathbf{T}^t\right)^T\tilde{\mathbf{S}}_{Cij}\mathbf{T}^t\right), \tag{11}$$

which are constants within the current iteration. Although this surrogate problem (10) is convex and can be solved as an SDP, we can devise a more efficient solution method.

To develop this more efficient solver, we begin by reformulating the inner minimization over the discrete indices $(i, j)$. This is accomplished by introducing a set of continuous auxiliary variables, $\{z_{ij}\}$, which are constrained to the probability simplex ($\sum z_{ij} = 1, z_{ij} \geq 0$). This transformation recasts the original problem into the equivalent max-min formulation shown in (12):

$$\max_{\mathbf{T}^T\mathbf{T}\preccurlyeq\mathbf{I}_m} \min_{\{z_{ij}\}} \quad 2\operatorname{tr}\left(\mathbf{A}(\mathbf{z})^T\mathbf{T}\right) + \sum_{1\leq i<j\leq C} z_{ij}c_{ij} \quad \text{s.t.} \quad z_{ij} \geq 0, \sum_{1\leq i<j\leq C} z_{ij} = 1, \tag{12}$$

where the matrix $\mathbf{A}$ is now defined as the convex combination $\mathbf{A}(\mathbf{z}) \triangleq \sum_{1\le i<j\le C} z_{ij}\mathbf{A}_{ij}$.

The objective in (12) is affine (hence concave) in $\mathbf{T}$ and affine (hence convex) in $\mathbf{z}$. Since both domains are compact and convex, Sion's minimax theorem [39] holds, allowing us to interchange the operators and obtain the equivalent formulation:

$$\min_{\{z_{ij}\}} \max_{\mathbf{T}^T\mathbf{T}\preccurlyeq\mathbf{I}_m} \quad 2\,\mathrm{tr}\left(\mathbf{A}(\mathbf{z})^T\mathbf{T}\right) + \sum_{1\le i<j\le C} z_{ij}c_{ij} \quad \text{s.t.} \quad z_{ij}\ge 0, \sum_{1\le i<j\le C} z_{ij} = 1. \tag{13}$$

This dual formulation is highly advantageous, as the inner maximization problem in (13) now admits a closed-form analytical solution. Focusing on the trace term in the objective, the Von Neumann inequality [40] states that $\mathrm{tr}\left(\mathbf{A}^T\mathbf{T}\right) \le \sum_{k=1}^{m}\sigma_k(\mathbf{A})$, where $\sigma_k(\cdot)$ denotes the $k$-th singular value. This upper bound is attained when $\mathbf{T}$ is set to $\mathbf{T}^* = \mathbf{A}\left(\mathbf{A}^T\mathbf{A}\right)^{-\frac{1}{2}}$ (the strict inequality mentioned for $\alpha$ prior to Lemma 3.1 was to ensure this invertibility). Critically, this optimal $\mathbf{T}^*$ inherently fulfills the strict semi-orthogonality constraint $\left(\mathbf{T}^*\right)^T\mathbf{T}^* = \mathbf{I}$. By substituting this analytical solution for $\mathbf{T}^*$ into the dual problem (13), we eliminate the variable $\mathbf{T}$ and arrive at the following optimization problem solely over $\mathbf{z}$:

$$\min_{\{z_{ij}\}} \quad 2\sum_{i=1}^{m}\sigma_i(\mathbf{A}(\mathbf{z})) + \sum_{1\le i<j\le C} z_{ij}c_{ij} \qquad \text{s.t.} \quad z_{ij}\ge 0, \sum_{1\le i<j\le C} z_{ij} = 1, \tag{14}$$

where the dependency of $\mathbf{A}$ on $\mathbf{z}$ is made explicit. The first term in this objective is exactly twice the nuclear norm of $\mathbf{A}(\mathbf{z})$, denoted $\|\mathbf{A}(\mathbf{z})\|_*$. Since the nuclear norm is a convex function, and $\mathbf{A}(\mathbf{z})$ is a linear function of $\mathbf{z}$, problem (14) is convex. While it can be reformulated and solved as an SDP [41], it offers a computational advantage over (10) due to having fewer variables and constraints.

Once the optimal $\mathbf{z}^*$ is found by solving (14), the corresponding update for $\mathbf{T}$ is computed as:

$$\mathbf{T}^{(t+1)} = \mathbf{A}\left(\mathbf{z}^*\right)\left(\mathbf{A}^T\left(\mathbf{z}^*\right)\mathbf{A}\left(\mathbf{z}^*\right)\right)^{-\frac{1}{2}}. \tag{15}$$

This $\mathbf{T}^{(t+1)}$ becomes the input for the subsequent iteration, and the process repeats until convergence.

The primary bottleneck in each iteration is solving the convex problem (14). This can be reformulated as an SDP, but standard interior-point methods scale polynomially with matrix dimensions, costing approximately $\mathcal{O}((d+m)^{4.5})$ per iteration—prohibitive for large-scale datasets. This motivates a more efficient method for (14). To reduce the burden, we introduce an MM-based approach applied directly to the nuclear norm term in (14), replacing the expensive SDP with a sequence of much simpler QPs.

We begin by rewriting problem (14) to explicitly show the nuclear norm and the linear term in summation form:

$$\min_{\{z_{ij}\}} \quad 2\|\mathbf{A}(\mathbf{z})\|_* + \sum_{1\le i<j\le C} z_{ij}c_{ij} \qquad \text{s.t.} \quad z_{ij}\ge 0, \quad \sum_{1\le i<j\le C} z_{ij} = 1. \tag{16}$$

The core idea is to replace the non-smooth nuclear norm $\|\mathbf{A}(\mathbf{z})\|_*$ with a smooth quadratic upper bound at each iteration, using its variational form as a joint minimization:

$$\|\mathbf{X}\|_* = \frac{1}{2}\min_{\mathbf{\Phi}\succ\mathbf{0}} \mathrm{tr}(\mathbf{X}^T\mathbf{X}\mathbf{\Phi}) + \mathrm{tr}(\mathbf{\Phi}^{-1}). \tag{17}$$

Using (17), we can reformulate problem (16) into an equivalent joint minimization problem over both $\mathbf{z}$ and an auxiliary positive definite matrix $\mathbf{\Phi}$:

$$\min_{\{z_{ij}\},\mathbf{\Phi}\succ\mathbf{0}} \quad \mathrm{tr}\left(\mathbf{A}(\mathbf{z})^T\mathbf{A}(\mathbf{z})\mathbf{\Phi}\right) + \mathrm{tr}(\mathbf{\Phi}^{-1}) + \sum_{1\le i<j\le C} z_{ij}c_{ij} \qquad \text{s.t.} \quad z_{ij}\ge 0, \quad \sum_{1\le i<j\le C} z_{ij} = 1. \tag{18}$$

Problem (18) can be tackled using an alternating minimization method. In this scheme, for a given $\mathbf{z}$ at iterate $t$, denoted $\mathbf{z}^t$, we first minimize (18) with respect to $\mathbf{\Phi}$. The problem has a closed-form solution for the optimal $\mathbf{\Phi}$, given by $\mathbf{\Phi}^* = (\mathbf{A}(\mathbf{z}^t)^T\mathbf{A}(\mathbf{z}^t))^{-\frac{1}{2}}$. Substituting $\mathbf{\Phi}$ back into the objective yields the following minimization problem over $\mathbf{z}$:

$$\min_{\{z_{ij}\}} \quad \mathrm{tr}\left(\mathbf{A}(\mathbf{z})^T\mathbf{A}(\mathbf{z})\mathbf{\Phi}^k\right) + \sum_{1\le i<j\le C} z_{ij}c_{ij} \qquad \text{s.t.} \quad z_{ij}\ge 0, \quad \sum_{1\le i<j\le C} z_{ij} = 1. \tag{19}$$

Table 1: Description of datasets.

| Dataset | Samp. | Cls. | Dim. | Dataset | Samp. | Cls. | Dim. | Dataset | Samp. | Cls. | Dim. |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Zernike [42] | 2,000 | 10 | 47 | Digit [43] | 1,797 | 10 | 64 | OSR [44] | 2,688 | 8 | 65,536 |
| MSRA25[1] [45] | 1,799 | 12 | 784 | Yale[2] [46] | 165 | 15 | 1,024 | S-ImgNet[3] | 1,000 | 10 | 786,432 |

This problem can be transformed into a standard QP of the form:

$$\min_{\{z_{ij}\}} \mathbf{z}^T \mathbf{Q} \mathbf{z} + \sum_{1 \le i < j \le C} z_{ij} c_{ij} \qquad \text{s.t.} \quad z_{ij} \ge 0, \sum_{1 \le i < j \le C} z_{ij} = 1, \tag{20}$$

where the matrix $\mathbf{Q}$ is constructed as follows:

$$\mathbf{Q} = \mathbf{S}^{\mathrm{T}} \left( \mathbf{\Phi}^t \otimes \mathbf{I} \right) \mathbf{S}. \tag{21}$$

Here, the symbol $\otimes$ represents the Kronecker product and $\mathbf{S}$ is formed by stacking the vectorized matrices $\mathbf{A}_{ij}$. Specifically, $\mathbf{S} = [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_K]$ with $K = C(C-1)/2$, and each column $\boldsymbol{v}_k = \mathrm{vec}(\mathbf{A}_{ij})$ corresponds to a unique pair $(i, j)$ via the index mapping $k = \frac{(i-1)(2C-i)}{2} + j - i$. The detailed derivation of $\mathbf{Q}$ is available in Appendix H.

This QP formulation (20) is computationally advantageous. It can be solved with standard solvers, and the combined cost of updating $\mathbf{\Phi}^t$ and solving the QP is approximately $\mathcal{O}(C^6 + d^3)$, which is substantially lower than the $\mathcal{O}((d+m)^{4.5})$ complexity of the original SDP approach in (14). While this QP needs to be resolved iteratively for updated $\mathbf{\Phi}^t$ matrices, the process converges rapidly, typically in under 10 iterations [25]. It is worth mentioning that in all standard benchmark datasets evaluated — even across the competing literature — the number of classes $C$ is significantly lower than the feature dimension $d$. Nevertheless, if a scenario with an extremely high $C$ arises, we can bypass the standard solver via MM. By upper-bounding the Hessian $\mathbf{Q}$ with a diagonal majorizer $L\mathbf{I}$ where $L \ge 2\lambda_{\max}(\mathbf{Q})$, the variables decouple. The complex QP collapses into a simple projected Gradient descent update: $\mathbf{z}^{(t+1)} = \mathrm{Proj}_{\Delta}\left(\mathbf{z}^{(t)} - \frac{1}{L}(2\mathbf{Q}\mathbf{z}^{(t)} + \mathbf{c})\right)$, where $\mathbf{c}$ is the vector of linear coefficients $c_{ij}$. Here, $\mathrm{Proj}_{\Delta}$ denotes the Euclidean projection onto the unit simplex $\Delta = \{\mathbf{z} \in \mathbb{R}^K : z_k \ge 0, \sum_{k=1}^K z_k = 1\}$. The Euclidean projection onto the simplex takes only $\mathcal{O}(C^2 \log C)$ time, entirely eliminating the $\mathcal{O}(C^6)$ bottleneck.

# 4 Numerical results

## 4.1 Datasets

We evaluate our algorithm on six datasets spanning digit recognition, face recognition, and scene/object classification across small, medium, and large feature spaces. Table 1 provides an overview; full details are given in Appendix I.

## 4.2 Experimental Protocol and Compared Methods

We compared our method against a comprehensive set of baseline algorithms, including linear discriminant analysis (LDA) [16, 17], heteroscedastic LDA (HLDA) [47], weighted heteroscedastic max-min distance analysis (WHMMDA) [48, 21], weighted distance analysis with eigenvalue decomposition (WDA-EIG) [49], robust LDA based on the $\ell_{2,1}$-norm (RLDA-$\ell_{2,1}$) [1], worst-case discriminative dimension reduction (WDDR) [2], as well as the recent QP-MMDA [25] and QP-MMRA [35] methods for distance and ratio analysis, respectively.

We randomly split each high-dimensional dataset in half, utilizing 50% of the samples for training and the remaining 50% for testing. As a preprocessing step, following the established protocol of [25, 2, 21], we apply PCA to project all feature vectors down to 50 dimensions (except for Stable ImageNet, for which the preprocessing is detailed in Appendix I), thereby retaining over 98% of

[1] `https://github.com/Psy680267/MSNMF`

[2] `http://cvc.cs.yale.edu/cvc/projects/yalefaces/yalefaces.html`

[3] `https://www.kaggle.com/datasets/vitaliykinakh/stable-imagenet1k`

Table 2: Classification error on S-ImageNet and MSRA25 (dim. and std. in parentheses).

| Dataset | Stable ImageNet | | | MSRA-25 | | |
|---|---|---|---|---|---|---|
| Method \ Classifier | 1-NN | NM | QDF | 1-NN | NM | QDF |
| LDA (1936, 1948) [16, 17] | 0.2658 (2, Std:0.0421) | 0.2513 (2, Std:0.0375) | 0.2474 (2, Std:0.0394) | 0.0146 (2, Std:0.0277) | 0.0121 (2, Std:0.0235) | 0.0120 (2, Std:0.0229) |
| HLDA (2004) [47] | 0.2782 (2, Std:0.0377) | 0.2451 (2, Std:0.0396) | 0.2291 (2, Std:0.0384) | 0.0485 (2, Std:0.0301) | 0.0821 (2, Std:0.0421) | 0.0468 (2, Std:0.0290) |
| WHMMDA (2015, 2018)[48, 21] | 0.3461 (2, Std:0.0594) | 0.2974 (2, Std:0.0616) | 0.2858 (2, Std:0.0555) | 0.2211 (2, Std:0.0454) | 0.3400 (2, Std:0.0674) | 0.2456 (2, Std:0.0613) |
| WDA-EIG (2020) [49] | 0.3056 (2, Std:0.0377) | 0.3000 (2, Std:0.0408) | 0.3018 (2, Std:0.0402) | 0.1066 (2, Std:0.0876) | 0.0958 (2, Std:0.0872) | 0.0907 (2, Std:0.0828) |
| RLDA-$\ell_{2,1}$ (2021)[1] | 0.4442 (2, Std:0.0281) | 0.4075 (2, Std:0.0248) | 0.3890 (2, Std:0.0243) | 0.1373 (2, Std:0.0352) | 0.1397 (2, Std:0.0407) | 0.1163 (2, Std:0.0337) |
| WDDR (2024)[2] | 0.3659 (2, Std:0.0372) | 0.2914 (2, Std:0.0363) | 0.2835 (2, Std:0.0353) | 0.0392 (2, Std:0.0493) | 0.0452 (2, Std:0.0599) | 0.0386 (2, Std:0.0528) |
| QP-MMDA (2025)[25] | 0.3552 (2, Std:0.0678) | 0.3058 (2, Std:0.0663) | 0.2928 (2, Std:0.0640) | 0.1735 (2, Std:0.0535) | 0.3276 (2, Std:0.0770) | 0.2268 (2, Std:0.0819) |
| QP-MMRA (2026) [35] | 0.3354 (2, Std:0.1011) | 0.2687 (2, Std:0.0707) | 0.2741 (2, Std:0.0742) | 0.0987 (2, Std:0.1351) | 0.1010 (2, Std:0.1446) | 0.0923 (2, Std:0.1333) |
| F$^3$-PWCRA | **0.2227** (2, Std:0.0494) | **0.1848** (2, Std:0.0367) | **0.2039** (2, Std:0.0497) | **0.0071** (2, Std:0.0061) | **0.0089** (2, Std:0.0074) | **0.0061** (2, Std:0.0059) |

the total variance . We repeat this entire process $100$ times with independent random splits and report the mean classification error alongside its standard deviation. The most rigorous challenge for dimensionality reduction is maintaining performance during an extreme reduction in features. Since many applications prioritize a minimal dimensional footprint, we project data into a strict two-dimensional bottleneck. This effectively tests each algorithm's ability to remain discriminative under substantial information loss while simultaneously allowing for a direct visual comparison of the embedded subspaces. Classification within these subspaces was performed using three models: the nearest neighbor (1-NN), the nearest mean (NM), and the quadratic discriminant function (QDF).

$$\mathbf{x} \in \arg\min_{i=1,\ldots,C} \left\{ (\mathbf{x} - \bar{\mathbf{x}}_i)^T \, \mathbf{\Sigma}_i^{-1} \, (\mathbf{x} - \bar{\mathbf{x}}_i) \; + \; \log|\mathbf{\Sigma}_i| \right\},$$

where $\bar{\mathbf{x}}_i$ represents the mean vector of class $i$, and $\mathbf{\Sigma}_i$ is the covariance matrix of class $i$.

It is worth mentioning that all algorithms were executed in MATLAB R2022b on a dual-socket Intel Xeon E5-2695 v3 system (2×14 cores, 56 threads; 2.3 GHz base, 3.3 GHz turbo; 315 GB RAM).

### 4.3 Results and Analysis

Table 2 presents the quantitative classification performance of the proposed method against various baseline models on the Stable ImageNet and MSRA-25 datasets. On the Stable ImageNet dataset, the proposed F$^3$-PWCRA framework establishes clear superiority, achieving the lowest error rates across all evaluated classifiers. For instance, using a 1-NN classifier, it yields an error of $0.2227$. This not only outperforms strong traditional baselines such as LDA $(0.2658)$ but also demonstrates significant improvements over the most recent robust formulations, including the QP-MMDA $(0.3552)$ and the QP-MMRA $(0.3354)$. This competitive advantage persists across the NM and QDF classifiers, where F$^3$-PWCRA consistently minimizes classification error (e.g., achieving $0.1848$ with the NM classifier) while maintaining narrow standard deviations.

On the MSRA-25 dataset, F$^3$-PWCRA demonstrates exceptional performance, yielding a highly competitive error rate of $0.0071$ under the 1-NN classifier. This result notably surpasses both classic methods like LDA $(0.0146)$ and the latest algorithmic advancements, such as the QP-MMRA, which records an error of $0.0987$. Similar trends are observed for the NM and QDF classifiers, with F$^3$-PWCRA achieving near-perfect classification capabilities (e.g., a remarkably low error of $0.0061$ using QDF). By consistently outperforming past benchmarks and recent methods, F$^3$-PWCRA highlights its robustness, stability, and strong discriminative power.

To provide a qualitative perspective, we examine the two-dimensional projections in Figure 1. Visualizing all classes simultaneously would produce a heavily cluttered space, so we restrict our plots to the first few classes; the source code will enable visualization of any arbitrary subset. As observed in the Stable ImageNet projections, methods such as LDA, HLDA, WHMMDA, WDA-EIG, and QP-MMRA exhibit the poorest visual performance. These algorithms frequently cover one class entirely with another—a failure beyond standard boundary overlap. Specifically, LDA and WDA-EIG cover class 1 under class 5, HLDA covers class 2 under class 5, and WHMMDA covers class 1

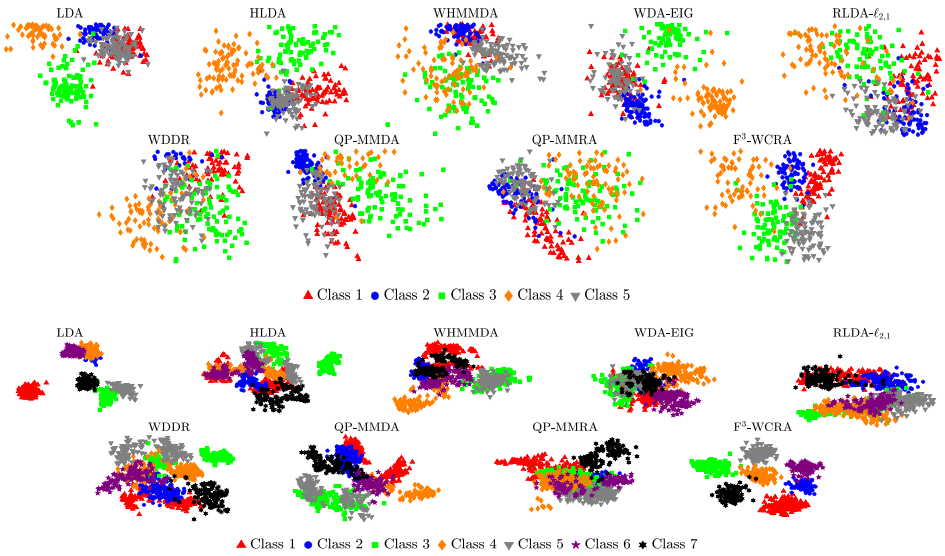


Figure 1: Learned projections for Stable ImageNet (top) and MSRA25 (bottom), $m = 2$.

under classes 2 and 3, and more than half of class 3 under class 4. For RLDA-$\ell_{2,1}$, WDDR, and QP-MMDA, the situation improves somewhat, yet significant overlap persists. RLDA-$\ell_{2,1}$ covers half of class 2 overlapping with classes 1 and 5; WDDR covers half of classes 3 and 4 under class 5; and QP-MMDA covers half of class 2 and half of class 1 under class 5. In contrast, the proposed F$^3$-WCRA achieves improved class separation, substantially reducing the merging and occlusion of classes. This powerful capability is even more apparent on the MSRA25 dataset. For this dataset, F$^3$-WCRA achieves optimal performance, projecting completely isolated and well-separated classes. LDA ranks as the second-best performer; nevertheless, it still projects class 4 entirely onto class 2, alongside a slight overlap between classes 4 and 6. The remaining methods perform extremely poorly on MSRA25, with inter-class occlusion, chaotic overlap, and high intra-class dispersion.

Crucially, these qualitative observations align with the quantitative metrics above. A more comprehensive analysis is provided in Section J of the appendix, including visualizations for the four remaining datasets, along with statistical t-tests and runtime comparisons across all six datasets. To further demonstrate the effectiveness of our approach, the appendix also details all quantitative experiments (1-NN, NM, and QDA metrics, t-tests, and runtime) conducted in three dimensions ($m = 3$).

## 5 Discussion and Limitations

Our proposed method offers a fast, relaxation-free, and hyperparameter-free approach to optimizing pairwise worst-case class separation. However, like prior work, we assume the number of classes, $C$, is fixed and known *a priori*. Extending this framework to an unknown or dynamic $C$ presents theoretical and computational challenges that remain an open research direction.

## 6 Conclusion

In this work, we presented a solution to the worst-case class separation problem in discriminative dimensionality reduction. We proposed an objective that maximizes the minimum pairwise ratio of between-class to within-class scatter, leading to the development of F$^3$-PWCRA, a provably convergent two-level optimization algorithm. By combining a generalized Dinkelbach-type procedure (or, alternatively, bisection search) with an MM-based inner solver, F$^3$-PWCRA efficiently solves the underlying non-convex max-min fractional program without requiring semidefinite relaxations or hyperparameter tuning. Extensive experiments (evaluating classification error using 1-NN, NM, and QDF, alongside t-tests, computational analysis, and visualization) on multiple benchmark datasets substantiated the effectiveness of our approach, demonstrating its consistent superiority over state-of-the-art methods in reducing classification error.

## Acknowledgments and Disclosure of Funding

The authors acknowledge the High Performance Computing (HPC) Center at Sharif University of Technology for providing computational resources that have contributed to this research. This work is supported by INSF, Iran National Science Foundation, grant number 4032041. The authors declare no competing interests.

## References

[1] F. Nie, Z. Wang, R. Wang, Z. Wang, and X. Li, "Towards robust discriminative projections learning via non-greedy $\ell_{2,1}$-norm minmax," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 6, pp. 2086–2100, 2021.

[2] Z. Wang *et al.*, "Worst-case discriminative feature learning via max-min ratio analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 1, pp. 641–658, 2024.

[3] F. Nie, X. Dong, Z. Hu, R. Wang, and X. Li, "Discriminative projected clustering via unsupervised lda," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 11, pp. 9466–9480, 2023.

[4] F. Nie, X. Dong, and X. Li, "Unsupervised and semisupervised projection with graph optimization," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 4, pp. 1547–1559, 2021.

[5] X. Chang, F. Nie, S. Wang, Y. Yang, X. Zhou, and C. Zhang, "Compound rank- $k$ projections for bilinear analysis," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 7, pp. 1502–1513, 2016.

[6] F. Nie, W. Zhu, and X. Li, "Unsupervised large graph embedding," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, p. 2422–2428, AAAI Press, 2017.

[7] Z. Li, F. Nie, X. Chang, L. Nie, H. Zhang, and Y. Yang, "Rank-constrained spectral clustering with flexible embedding," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 12, pp. 6073–6082, 2018.

[8] J. Gui, Z. Sun, S. Ji, D. Tao, and T. Tan, "Feature selection based on structured sparsity: A comprehensive study," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 7, pp. 1490–1507, 2017.

[9] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu, "Feature selection: A data perspective," *ACM Comput. Surv.*, vol. 50, Dec. 2017.

[10] R. Sheikhpour, M. A. Sarram, S. Gharaghani, and M. A. Z. Chahooki, "A survey on semi-supervised feature selection methods," *Pattern Recognition*, vol. 64, pp. 141–158, 2017.

[11] E. Hancer, B. Xue, and M. Zhang, "A survey on feature selection approaches for clustering," *Artificial Intelligence Review*, vol. 53, no. 6, pp. 4519–4545, 2020.

[12] X. Li, Y. Wang, and R. Ruiz, "A survey on sparse learning models for feature selection," *IEEE Transactions on Cybernetics*, vol. 52, no. 3, pp. 1642–1660, 2022.

[13] H. T. Shen, Y. Zhu, W. Zheng, and X. Zhu, "Half-quadratic minimization for unsupervised feature selection on incomplete data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 7, pp. 3122–3135, 2021.

[14] Z. Li, F. Nie, X. Chang, Y. Yang, C. Zhang, and N. Sebe, "Dynamic affinity graph construction for spectral clustering using multiple features," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 12, pp. 6323–6332, 2018.

[15] M. Luo, F. Nie, X. Chang, Y. Yang, A. G. Hauptmann, and Q. Zheng, "Adaptive unsupervised feature selection with structure regularization," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 4, pp. 944–956, 2018.

[16] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of eugenics*, vol. 7, no. 2, pp. 179–188, 1936.

[17] C. R. Rao, "The utilization of multiple measurements in problems of biological classification," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 10, no. 2, pp. 159–203, 1948.

[18] W. Bian and D. Tao, "Max-min distance analysis by using sequential SDP relaxation for dimension reduction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 1037–1050, 2011.

[19] Y. Zhang and D.-Y. Yeung, "Worst-case linear discriminant analysis," in *Advances in Neural Information Processing Systems* (J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, eds.), vol. 23, Curran Associates, Inc., 2010.

[20] Y. Yu, J. Jiang, and L. Zhang, "Distance metric learning by minimal distance maximization," *Pattern Recognition*, vol. 44, no. 3, pp. 639–649, 2011.

[21] B. Su *et al.*, "Heteroscedastic max-min distance analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 4539–4547, 2015.

[22] F. Nie, Z. Wang, R. Wang, and X. Li, "Submanifold-preserving discriminant analysis with an auto-optimized graph," *IEEE Transactions on Cybernetics*, vol. 50, no. 8, pp. 3682–3695, 2020.

[23] F. Nie, Z. Wang, R. Wang, Z. Wang, and X. Li, "Adaptive local linear discriminant analysis," *ACM Trans. Knowl. Discov. Data*, vol. 14, Feb. 2020.

[24] O. C. Hamsici and A. M. Martinez, "Bayes optimality in linear discriminant analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 4, pp. 647–657, 2008.

[25] M. M. Omati, P. Babu, P. Stoica, and A. Amini, "A max-min approach to the worst-case class separation problem," *Transactions on Machine Learning Research*, 2025.

[26] S. Song, Y. Gong, Y. Zhang, G. Huang, and G.-B. Huang, "Dimension reduction by minimum error minimax probability machine," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 1, pp. 58–69, 2017.

[27] B. Xu, K. Huang, and C.-L. Liu, "Dimensionality reduction by minimal distance maximization," in *2010 20th International Conference on Pattern Recognition*, pp. 569–572, 2010.

[28] B. Xu, K. Huang, and C.-L. Liu, "Maxi-min discriminant analysis via online learning," *Neural networks*, vol. 34, pp. 56–64, 2012.

[29] G. Shao and N. Sang, "Max–min distance analysis by making a uniform distribution of class centers for dimensionality reduction," *Neurocomputing*, vol. 143, pp. 208–221, 2014.

[30] J. Ding, G. Li, C. Wen, and C. S. Chua, "Min-max discriminant analysis based on gradient method for feature extraction," in *2014 13th International Conference on Control Automation Robotics & Vision (ICARCV)*, pp. 129–134, 2014.

[31] G. Shao and N. Sang, "Regularized max-min linear discriminant analysis," *Pattern recognition*, vol. 66, pp. 353–363, 2017.

[32] H. Li, C. Shen, A. van den Hengel, and Q. Shi, "Worst case linear discriminant analysis as scalable semidefinite feasibility problems," *IEEE Transactions on Image Processing*, vol. 24, no. 8, pp. 2382–2392, 2015.

[33] J. Hu, W. Deng, J. Guo, and Y. Xu, "Max-K-min distance analysis for dimension reduction," in *2014 22nd International Conference on Pattern Recognition*, pp. 726–731, 2014.

[34] W. Bian and D. Tao, "Max-min distance analysis by using sequential SDP relaxation for dimension reduction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 1037–1050, 2011.

[35] M. M. Omati, A. Amini, and S.-M. Karbasi, "QP-MMRA: A framework for worst-case ratio analysis in class separation." Under review at IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI). Included in the supplementary material, 2026.

[36] C. Shen, H. Li, and M. J. Brooks, "Supervised dimensionality reduction via sequential semidefinite programming," *Pattern Recognition*, vol. 41, no. 12, pp. 3644–3652, 2008.

[37] W. Dinkelbach, "On nonlinear fractional programming," *Management science*, vol. 13, no. 7, pp. 492–498, 1967.

[38] A. Saini, P. Stoica, P. Babu, and A. Arora, "Min-max framework for majorization-minimization algorithms in signal processing applications: An overview," *Foundations and Trends® in Signal Processing*, vol. 18, no. 4, pp. 310–389, 2024.

[39] M. Sion, "On general minimax theorems.," *Pacific Journal of Mathematics*, vol. 8, no. 1, pp. 171 – 176, 1958.

[40] A. Marshall, "Inequalities: Theory of majorization and its applications," 1979.

[41] B. Recht, M. Fazel, and P. A. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM review*, vol. 52, no. 3, pp. 471–501, 2010.

[42] R. Duin, "Multiple features [dataset]." UCI Machine Learning Repository, 1998. `https://doi.org/10.24432/C5HC70`.

[43] V. U. Prabhu, "Kannada-MNIST: A new handwritten digits dataset for the kannada language," *arXiv preprint arXiv:1908.01242*, 2019.

[44] D. Parikh and K. Grauman, "Relative attributes," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 503–510, 2011.

[45] L. Zhang, Q. Zhang, B. Du, J. You, and D. Tao, "Adaptive manifold regularized matrix factorization for data clustering," in *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pp. 3399–3405, 2017.

[46] P. Belhumeur, J. Hespanha, and D. Kriegman, "Eigenfaces vs. Fisherfaces: recognition using class specific linear projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711–720, 1997.

[47] M. Loog and R. P. W. Duin, "Linear dimensionality reduction via a heteroscedastic extension of LDA: The chernoff criterion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 6, pp. 732–739, 2004.

[48] B. Su, X. Ding, C. Liu, and Y. Wu, "Heteroscedastic max–min distance analysis for dimensionality reduction," *IEEE Transactions on Image Processing*, vol. 27, no. 8, pp. 4052–4065, 2018.

[49] H. Liu *et al.*, "Ratio trace formulation of Wasserstein discriminant analysis," in *Adv. Neural Inf. Process. Syst.* (H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, eds.), vol. 33, pp. 16821–16832, Curran Associates, Inc., 2020.

[50] Y. Sun, P. Babu, and D. P. Palomar, "Majorization-minimization algorithms in signal processing, communications, and machine learning," *IEEE Transactions on Signal Processing*, vol. 65, no. 3, pp. 794–816, 2017.

# Appendices

## A Preliminary: The Minorization-Maximization (MM) Principle

### A.1 The General MM Framework

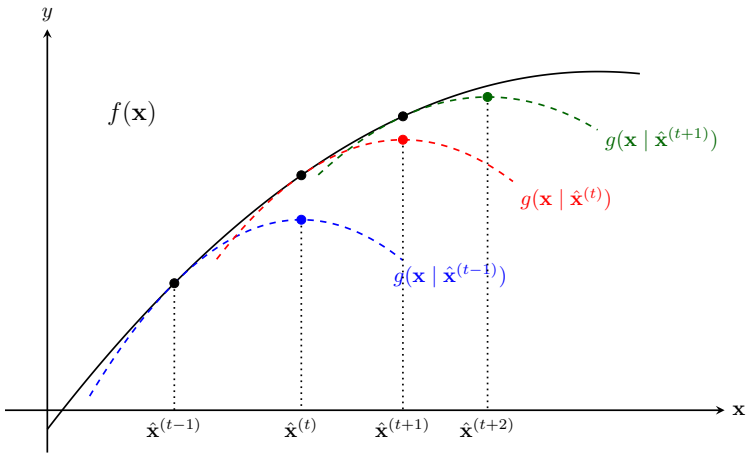


Figure S1: An illustration of the MM algorithm's behavior.

The minorization-maximization (MM) algorithm is an iterative optimization technique for solving constrained maximization problems of the form:

$$\max_{\mathbf{T} \in \chi} f(\mathbf{T}), \tag{S1}$$

where $f(\mathbf{T})$ is the objective function, $\mathbf{T}$ is the optimization variable, and $\chi$ represents the feasible set. The core principle of the MM algorithm involves iteratively solving a sequence of simpler optimization problems. Specifically, at each iteration $t$, a surrogate function $g(\mathbf{T} \mid \mathbf{T}^t)$, termed a minorizer of $f(\mathbf{T})$, is constructed. This surrogate must satisfy two fundamental conditions:

$$g(\mathbf{T} \mid \mathbf{T}^t) \leq f(\mathbf{T}), \quad \forall \mathbf{T} \in \chi, \tag{S2}$$

$$g(\mathbf{T}^t \mid \mathbf{T}^t) = f(\mathbf{T}^t). \tag{S3}$$

The first condition (S2) ensures that the surrogate function provides a global lower bound for the original objective function. The second condition (S3) guarantees that the surrogate is tangent to (or "touches") the objective function at the current iterate $\mathbf{T}^t$.

The subsequent iterate, $\mathbf{T}^{t+1}$, is then obtained by maximizing this surrogate function over the feasible set:

$$\mathbf{T}^{t+1} \in \arg\max_{\mathbf{T} \in \chi} g\left(\mathbf{T} \mid \mathbf{T}^t\right). \tag{S4}$$

This iterative process, encompassing the construction and maximization of the surrogate, is repeated until a convergence criterion is met, typically when the relative change in the objective function value falls below a predefined tolerance $\epsilon$. Figure S1 illustrates the behavior of the MM approach.

A key property of the MM algorithm is the guaranteed monotonic improvement of the objective function value at each step. This ascent property is readily established through the following sequence of inequalities:

$$f(\mathbf{T}^{t+1}) \geq g(\mathbf{T}^{t+1} \mid \mathbf{T}^t) \geq g(\mathbf{T}^t \mid \mathbf{T}^t) = f(\mathbf{T}^t). \tag{S5}$$

The first inequality holds due to the surrogate condition in (S2), the second follows from the maximization step in (S4), and the final equality is a direct consequence of the tangency condition in (S3). This guarantees that the sequence of objective values $\{f(\mathbf{T}^t)\}$ is non-decreasing.

Table S1: An overview of the ablation dimensions distinguishing the evaluated baseline methods. In this table, ✓ shows that a method employs the specified feature, and × indicates it does not.

| Method | Ratio based | Worst case | Direct | Hyperparameter free | Pairwise scatter (pairwise covariance control) |
|---|---|---|---|---|---|
| LDA [16, 17] | ✓ | × | ✓ | ✓ | × |
| HLDA [47] | ✓ | × | ✓ | ✓ | × |
| WHMMDA [48, 21] | × | ✓ | × | × | × |
| WDA-EIG (2020) [49] | ✓ | × | ✓ | × | × |
| RLDA-$\ell_{2,1}$ [1] | ✓ | × | ✓ | ✓ | × |
| WDDR [2] | ✓ | ✓ | × | × | ✓ |
| QP-MMDA [25] | × | ✓ | ✓ | ✓ | × |
| QP-MMRA [35] | ✓ | ✓ | ✓ | ✓ | × |
| $F^3$-PWCRA (Ours) | ✓ | ✓ | ✓ | ✓ | ✓ |

### A.2 An MM Framework for Max-Min Problems (MM4MM)

The MM principle can be effectively extended to address max-min optimization problems, which are structured as:

$$\max_{\mathbf{T}\in\mathcal{X}} \left\{ f(\mathbf{T}) \triangleq \min_{i=1,\ldots,K} f_i(\mathbf{T}) \right\}, \tag{S6}$$

where the overall objective $f(\mathbf{T})$ is defined by the pointwise minimum of a set of functions $\{f_i(\mathbf{T})\}_{i=1}^K$. To solve this problem using an MM approach, we construct a composite surrogate function for $f(\mathbf{T})$.

Let $g_i(\mathbf{T} \mid \mathbf{T}^t)$ be a valid minorizer for each individual function $f_i(\mathbf{T})$, satisfying the standard MM conditions:

$$g_i(\mathbf{T} \mid \mathbf{T}^t) \leq f_i(\mathbf{T}), \tag{S7}$$

$$g_i(\mathbf{T}^t \mid \mathbf{T}^t) = f_i(\mathbf{T}^t). \tag{S8}$$

A natural choice for the overall surrogate function $g(\mathbf{T} \mid \mathbf{T}^t)$ is the pointwise minimum of the individual surrogates:

$$g\left(\mathbf{T} \mid \mathbf{T}^t\right) \triangleq \min_{i=1,\ldots,K} g_i\left(\mathbf{T} \mid \mathbf{T}^t\right). \tag{S9}$$

It can be verified that this construction yields a valid minorizer for the max-min objective $f(\mathbf{T})$. The lower-bound property is established as follows:

$$g(\mathbf{T} \mid \mathbf{T}^t) = \min_i g_i(\mathbf{T} \mid \mathbf{T}^t) \leq \min_i f_i(\mathbf{T}) = f(\mathbf{T}), \tag{S10}$$

and the tangency condition is similarly met:

$$g(\mathbf{T}^t \mid \mathbf{T}^t) = \min_i g_i(\mathbf{T}^t \mid \mathbf{T}^t) = \min_i f_i(\mathbf{T}^t) = f(\mathbf{T}^t). \tag{S11}$$

By applying the standard MM update rule with the surrogate defined in (S9), the resulting sequence of iterates $\{\mathbf{X}^t\}$ is guaranteed to monotonically increase the max-min objective function and converge to a stationary point. For a more detailed exposition of the Minorization-Maximization (MM) approach and its diverse applications, including methods for constructing surrogate functions, we refer readers to [50, 38].

## B Methodological Ablation Study

This section details how the performance evaluation in the main text functions as a comprehensive ablation study of discriminant analysis techniques. The selection of baseline techniques is deliberate, ensuring that they encompass the key structural variations and methodological divergences within the field.

### B.1 Design Dimensions for Ablation

The algorithms evaluated in this study vary systematically across five distinct methodological axes:

#### B.1.1 Distance-Based versus Ratio-Based Formulation

Methods relying solely on distance aim to enlarge the absolute gap between classes. This works well for tight distributions but falters when classes exhibit varying degrees of dispersion. On the other hand, ratio-based formulations normalize the objective by simultaneously maximizing the distance between classes and minimizing their internal spread. This dual optimization ensures that data points within a class remain compact, preventing the overlap that high intra-class variance might otherwise cause.

#### B.1.2 Average-Case versus Worst-Case Optimization

Traditional average-case strategies evaluate class separation holistically, meaning the overall metric can be skewed by distant classes while neglecting the pairs closest to one another. Worst-case optimization, however, targets the "weakest link" by specifically maximizing the margin between the least separated class pairs. This focused approach is vital for multi-class scenarios where certain classes are highly prone to overlapping.

#### B.1.3 Relaxation-Based versus Direct Optimization

Advanced discriminant criteria, such as worst-case ratio problems, are often highly non-convex and difficult to solve. To sidestep this, relaxation-based techniques often use approximations like semidefinite relaxation to guarantee a solvable convex substitute, though this can lead to sub-optimal, loose solutions. Direct optimization techniques tackle the non-convex problem in its exact form without relaxing the constraints, thereby guaranteeing true convergence.

#### B.1.4 Hyperparameter Dependence

Algorithms that depend on manually tuned hyperparameters demand rigorous cross-validation to function properly, which hinders their deployment in large or automated pipelines. In contrast, hyperparameter-free approaches establish their parameters inherently from the data itself. This self-tuning property ensures robust, reproducible results and significantly simplifies real-world application.

#### B.1.5 Global versus Pairwise Scatter Normalization

Some ratio-based methods utilize a global within-class scatter matrix to represent overall data compactness. This non-pairwise normalization fails to adapt to the unique variances of individual class pairs, causing degraded performance when class dispersions differ substantially. Conversely, methods utilizing pairwise scatter evaluate the separability of each specific class pair relative to its own unique within-class variance. This adaptive design ensures that the distinct compactness of the most challenging class pairs is accurately accounted for, leading to superior discrimination when classes exhibit significantly different spread or variance structures.

### B.2 Summary of Ablation Dimensions

Table S1 outlines exactly how each compared algorithm maps onto these five critical design dimensions. This structured breakdown clarifies that our baseline selection encompasses the full spectrum of methodological choices. By isolating these specific variables, the experimental results demonstrate the tangible impact of each architectural decision, ultimately validating $F^3$-PWCRA as a unique and optimal integration of ratio-based formulation, worst-case focus, direct problem-solving, parameter-free execution, and adaptive pairwise scatter control.

## C Rationale for Upper Bound Candidates in Bisection

When implementing the bisection search as an alternative outer loop, we require an initial upper bound $\delta_u$ that satisfies $\delta_u \geq \delta^*$, where $\delta^*$ is the optimal value of (2). To obtain a tight upper bound that minimizes the initial search interval, we compute three candidate bounds and take their minimum. Each candidate is derived from a different perspective on the problem structure.

### C.1 Candidate $\delta_{u_1}$: Pairwise Min-Max Bound

For any function $f(x, y)$, the following inequality holds for both continuous and discrete settings:

$$\max_x \min_y f(x, y) \leq \min_y \max_x f(x, y). \tag{S12}$$

Applying this to our objective, we treat $\mathbf{T}$ as $x$ and the class pair $(i, j)$ as $y$:

$$\delta^* = \max_{\mathbf{T}^T\mathbf{T}=\mathbf{I}_m} \min_{i<j} \frac{\operatorname{tr}(\mathbf{T}^T\mathbf{S}_b^{ij}\mathbf{T})}{\operatorname{tr}(\mathbf{T}^T\mathbf{S}_w^{ij}\mathbf{T})} \leq \min_{i<j} \max_{\mathbf{T}^T\mathbf{T}=\mathbf{I}_m} \frac{\operatorname{tr}(\mathbf{T}^T\mathbf{S}_b^{ij}\mathbf{T})}{\operatorname{tr}(\mathbf{T}^T\mathbf{S}_w^{ij}\mathbf{T})}. \tag{S13}$$

For each fixed pair $(i, j)$, the inner maximization is a standard generalized Rayleigh quotient, whose maximum is given by the largest generalized eigenvalue $\lambda_{\max}(\mathbf{S}_b^{ij}, \mathbf{S}_w^{ij})$. Thus:

$$\delta^* \leq \min_{i<j} \lambda_{\max}\left(\mathbf{S}_b^{ij}, \mathbf{S}_w^{ij}\right) \triangleq \delta_{u_1}. \tag{S14}$$

### C.2 Candidate $\delta_{u_2}$: Standard LDA Bound

Since the minimum of a set of numbers is always less than or equal to their weighted average, for any projection matrix $\mathbf{T}$ we have:

$$\min_{i<j} \frac{\operatorname{tr}(\mathbf{T}^T\mathbf{S}_b^{ij}\mathbf{T})}{\operatorname{tr}(\mathbf{T}^T\mathbf{S}_w^{ij}\mathbf{T})} \leq \frac{\sum_{i<j} \operatorname{tr}(\mathbf{T}^T\mathbf{S}_b^{ij}\mathbf{T})}{\sum_{i<j} \operatorname{tr}(\mathbf{T}^T\mathbf{S}_w^{ij}\mathbf{T})} = \frac{\operatorname{tr}\left(\mathbf{T}^T\left(\sum_{i<j}\mathbf{S}_b^{ij}\right)\mathbf{T}\right)}{\operatorname{tr}\left(\mathbf{T}^T\left(\sum_{i<j}\mathbf{S}_w^{ij}\right)\mathbf{T}\right)}. \tag{S15}$$

Therefore:

$$\delta^* = \max_{\mathbf{T}} \min_{i<j} \frac{\operatorname{tr}(\mathbf{T}^T\mathbf{S}_b^{ij}\mathbf{T})}{\operatorname{tr}(\mathbf{T}^T\mathbf{S}_w^{ij}\mathbf{T})} \leq \max_{\mathbf{T}} \frac{\operatorname{tr}\left(\mathbf{T}^T\mathbf{S}_b^{\text{total}}\mathbf{T}\right)}{\operatorname{tr}\left(\mathbf{T}^T\mathbf{S}_w^{\text{total}}\mathbf{T}\right)} = \lambda_{\max}\left(\mathbf{S}_b^{\text{total}}, \mathbf{S}_w^{\text{total}}\right) \triangleq \delta_{u_2}, \tag{S16}$$

where $\mathbf{S}_b^{\text{total}} = \sum_{i<j} \mathbf{S}_b^{ij}$ and $\mathbf{S}_w^{\text{total}} = \sum_{i<j} \mathbf{S}_w^{ij}$.

### C.3 Candidate $\delta_{u_3}$: Inverse-Weighted LDA Bound

For any set of positive weights $\alpha_{ij} > 0$, the minimum is bounded above by the weighted average with those weights:

$$\min_{i<j} \frac{\operatorname{tr}(\mathbf{T}^T\mathbf{S}_b^{ij}\mathbf{T})}{\operatorname{tr}(\mathbf{T}^T\mathbf{S}_w^{ij}\mathbf{T})} \leq \frac{\sum_{i<j} \alpha_{ij} \operatorname{tr}(\mathbf{T}^T\mathbf{S}_b^{ij}\mathbf{T})}{\sum_{i<j} \alpha_{ij} \operatorname{tr}(\mathbf{T}^T\mathbf{S}_w^{ij}\mathbf{T})} = \frac{\operatorname{tr}\left(\mathbf{T}^T\left(\sum_{i<j}\alpha_{ij}\mathbf{S}_b^{ij}\right)\mathbf{T}\right)}{\operatorname{tr}\left(\mathbf{T}^T\left(\sum_{i<j}\alpha_{ij}\mathbf{S}_w^{ij}\right)\mathbf{T}\right)}. \tag{S17}$$

It follows that:

$$\delta^* \leq \max_{\mathbf{T}} \frac{\operatorname{tr}\left(\mathbf{T}^T\tilde{\mathbf{S}}_b\mathbf{T}\right)}{\operatorname{tr}\left(\mathbf{T}^T\tilde{\mathbf{S}}_w\mathbf{T}\right)} = \lambda_{\max}\left(\tilde{\mathbf{S}}_b, \tilde{\mathbf{S}}_w\right), \tag{S18}$$

where $\tilde{\mathbf{S}}_b = \sum_{i<j} \alpha_{ij}\mathbf{S}_b^{ij}$ and $\tilde{\mathbf{S}}_w = \sum_{i<j} \alpha_{ij}\mathbf{S}_w^{ij}$. To obtain the tightest possible bound, we choose weights that down-weight easy pairs. Specifically, we set:

$$\alpha_{ij} = \frac{1}{\lambda_{\max}\left(\mathbf{S}_b^{ij}, \mathbf{S}_w^{ij}\right)}, \tag{S19}$$

which assigns smaller weights to pairs that already achieve large separation. This yields the candidate bound $\delta_{u_3}$.

### C.4 Taking the Minimum

Since each candidate provides a valid upper bound, we set:

$$\delta_u = \min\{\delta_{u_1},\ \delta_{u_2},\ \delta_{u_3}\}. \tag{S20}$$

Choosing the minimum among them yields the tightest possible initial upper bound, which minimizes the number of bisection iterations required to achieve the desired tolerance $\epsilon$.

## D Proof of Theorem 3.1

**Proof.** Supposing $\mathbf{T}^*$ and $\delta^*$ are the optimal solution and corresponding objective function value of problem (2), then the following holds:

$$\min_{1\le i<j\le C} \frac{f_{ij}(\mathbf{T}^*)}{g_{ij}(\mathbf{T}^*)} = \delta^*.$$

Moreover, for any feasible solution $\mathbf{T} \in \mathcal{X}$, since $g_{ij}(\mathbf{T}) > 0$, we have:

$$\min_{1\le i<j\le C} \frac{f_{ij}(\mathbf{T})}{g_{ij}(\mathbf{T})} \le \delta^* \implies \min_{1\le i<j\le C} (f_{ij}(\mathbf{T}) - \delta^* g_{ij}(\mathbf{T})) \le 0.$$

So we can determine that:

$$h(\delta^*) = \max_{\mathbf{T}\in\mathcal{X}} \min_{1\le i<j\le C} (f_{ij}(\mathbf{T}) - \delta^* g_{ij}(\mathbf{T})) \le 0.$$

On the other hand, for the optimal solution $\mathbf{T}^*$:

$$\min_{1\le i<j\le C} \frac{f_{ij}(\mathbf{T}^*)}{g_{ij}(\mathbf{T}^*)} = \delta^* \implies \min_{1\le i<j\le C} (f_{ij}(\mathbf{T}^*) - \delta^* g_{ij}(\mathbf{T}^*)) = 0.$$

Consequently, we have $h(\delta^*) = 0$. This demonstrates that the optimal value $\delta^*$ of the problem in (2) is precisely the root of the function $h(\delta)$, thereby completing the proof of Theorem 3.1. □

## E Proof of Theorem 3.2

**Proof.** Algorithm 3 is a direct implementation of the generalized Dinkelbach procedure described in Section 3.1. This procedure can be interpreted as an application of Newton's method to find the root of the function $h(\delta) = \max_{\mathbf{T}\in\mathcal{X}} \min_{ij}(f_{ij}(\mathbf{T}) - \delta g_{ij}(\mathbf{T}))$. The first-order Taylor expansion of $h(\delta)$ around the current estimate $\delta_k$ is given by:

$$h(\delta) \approx h(\delta_k) + h'(\delta_k)(\delta - \delta_k).$$

The derivative of $h(\delta)$ with respect to $\delta$ is $h'(\delta) = -g_{ab}(\mathbf{T}^{k+1})$, where $\mathbf{T}^{k+1}$ is the argument that maximizes the inner expression for a given $\delta$, and $(a, b)$ is the index pair corresponding to the minimum value for that $\mathbf{T}^{k+1}$. Newton's method finds the root by setting this linear approximation to zero:

$$0 = h(\delta_k) - g_{ab}(\mathbf{T}^{k+1})(\delta - \delta_k).$$

Solving for $\delta$ yields the update rule for the next iterate, which we denote $\delta_{k+1}$:

$$\delta_{k+1} = \delta_k + \frac{h(\delta_k)}{g_{ab}(\mathbf{T}^{k+1})} = \delta_k + \frac{f_{ab}(\mathbf{T}^{k+1}) - \delta_k g_{ab}(\mathbf{T}^{k+1})}{g_{ab}(\mathbf{T}^{k+1})} = \frac{f_{ab}(\mathbf{T}^{k+1})}{g_{ab}(\mathbf{T}^{k+1})}.$$

This formulation is precisely the update rule presented in Step 2 of Algorithm 3. Thus, the algorithm implements Newton's method to solve $h(\delta) = 0$. □

## F Proof of Theorem 3.3

**Proof. Monotonic Convergence:** In the $k$-th iteration of Algorithm 3, let $\delta_k = \min_{ij} \frac{f_{ij}(\mathbf{T}^k)}{g_{ij}(\mathbf{T}^k)}$. This implies $f_{ij}(\mathbf{T}^k) - \delta_k g_{ij}(\mathbf{T}^k) \ge 0$ for all pairs $(i, j)$, and consequently, $\min_{ij}(f_{ij}(\mathbf{T}^k) - \delta_k g_{ij}(\mathbf{T}^k)) = 0$. The subproblem solved in Step 4 yields $\mathbf{T}^{k+1}$, defined as:

$$h(\delta_k) = \max_{\mathbf{T}\in\mathcal{X}} \min_{ij}(f_{ij}(\mathbf{T}) - \delta_k g_{ij}(\mathbf{T})) = \min_{ij}(f_{ij}(\mathbf{T}^{k+1}) - \delta_k g_{ij}(\mathbf{T}^{k+1})).$$

Since $\mathbf{T}^k$ is a feasible candidate for this maximization, we have $h(\delta_k) \ge 0$. This implies that for the new iterate $\mathbf{T}^{k+1}$, the inequality $f_{ij}(\mathbf{T}^{k+1}) - \delta_k g_{ij}(\mathbf{T}^{k+1}) \ge 0$ holds for all pairs $(i, j)$. Given $g_{ij}(\mathbf{T}) > 0$, it follows that:

$$\frac{f_{ij}(\mathbf{T}^{k+1})}{g_{ij}(\mathbf{T}^{k+1})} \ge \delta_k \quad \forall (i, j).$$

**Algorithm 3** Generalized Dinkelbach Algorithm

1: **Initialize:** Feasible $\mathbf{T}^0$, set $k = 0$, tolerance $\epsilon = 10^{-4}$.
2: **while** not converged **do**
3: Compute worst-case ratio: $\delta_k = \min_{1\le i<j\le C} \frac{\mathrm{tr}\left((\mathbf{T}^k)^T \mathbf{S}_b^{ij} \mathbf{T}^k\right)}{\mathrm{tr}\left((\mathbf{T}^k)^T \mathbf{S}_w^{ij} \mathbf{T}^k\right)}$.
4: Solve the max-min subproblem for the next iterate via Algorithm 2:

$$\mathbf{T}^{k+1} = \arg \max_{\mathbf{T}^T\mathbf{T}=\mathbf{I}_m} \min_{1\le i<j\le C} \left\{ \mathrm{tr}\big(\mathbf{T}^T(\mathbf{S}_b^{ij} - \delta_k \mathbf{S}_w^{ij})\mathbf{T}\big) \right\}.$$

5: Check convergence: $|\delta_k - \delta_{k-1}| < \epsilon$.
6: $k \leftarrow k + 1$.
7: **end while**
8: **Output:** Optimal transformation $\mathbf{T}^* = \mathbf{T}^k$.

Since $\delta_{k+1}$ is defined as the minimum of these ratios, i.e., $\delta_{k+1} = \min_{ij} \frac{f_{ij}(\mathbf{T}^{k+1})}{g_{ij}(\mathbf{T}^{k+1})}$, it must satisfy $\delta_{k+1} \ge \delta_k$. This proves that the objective value is non-decreasing at each iteration of Algorithm 3.

**Optimality:** Suppose the algorithm converges at iteration $k$, such that $\delta_k = \delta_{k+1}$. This convergence is mathematically guaranteed because the sequence $\{\delta_k\}$ is both monotonically non-decreasing (as established previously) and bounded from above. Specifically, for any $\mathbf{T} \in \mathcal{S}$ and class pair $(i, j)$, the Rayleigh-Ritz theorem and the orthogonality constraint $\mathbf{T}^T\mathbf{T} = \mathbf{I}_m$ ensure that $\frac{f_{ij}(\mathbf{T})}{g_{ij}(\mathbf{T})} \le \frac{\lambda_{\max}(\tilde{\mathbf{S}}_C^{ij})}{\lambda_{\min}(\tilde{\mathbf{S}}_W^{ij})}$. Consequently, $f(\mathbf{T})$ is bounded by the maximum value $M := \max_{ij} \frac{\lambda_{\max}(\tilde{\mathbf{S}}_C^{ij})}{\lambda_{\min}(\tilde{\mathbf{S}}_W^{ij})}$. By the monotone convergence theorem, the sequence must therefore converge to a limit, $\delta^*$.

At convergence, the update rule dictates that $\delta_k = \min_{ij} \frac{f_{ij}(\mathbf{T}^{k+1})}{g_{ij}(\mathbf{T}^{k+1})}$. Let $(c, d)$ represent the specific index pair that achieves this minimum for $\mathbf{T}^{k+1}$. It follows that $\delta_k = \frac{f_{cd}(\mathbf{T}^{k+1})}{g_{cd}(\mathbf{T}^{k+1})}$, which rearranges to $f_{cd}(\mathbf{T}^{k+1}) - \delta_k g_{cd}(\mathbf{T}^{k+1}) = 0$. This is mathematically equivalent to stating that $h(\delta_k) = 0$. According to Theorem 3.1, a solution $\delta^*$ is optimal if and only if $h(\delta^*) = 0$. Because the converged solution $\delta_k$ satisfies this exact condition, it is the optimal solution.

We can formally verify this via contradiction. Assume convergence has occurred, yet $h(\delta_k) > 0$. This assumption implies that $f_{cd}(\mathbf{T}^{k+1}) - \delta_k g_{cd}(\mathbf{T}^{k+1}) > 0$, which would force the subsequent update to be $\delta_{k+1} = \min_{ij} \frac{f_{ij}(\mathbf{T}^{k+1})}{g_{ij}(\mathbf{T}^{k+1})} > \delta_k$. This strictly contradicts our initial convergence premise that $\delta_k = \delta_{k+1}$. Therefore, the generalized Dinkelbach algorithm must converge to the global optimum.

□

## G The proof of Lemma 3.1

**Proof.** Consider the trace term $\mathrm{tr}\left(\mathbf{T}^T\tilde{\mathbf{S}}_{Cij}\mathbf{T}\right)$ for any pair $(i, j)$. Using the cyclic property of the trace, we can write:

$$\mathrm{tr}\left(\mathbf{T}^T\tilde{\mathbf{S}}_{Cij}\mathbf{T}\right) = \mathrm{tr}\left(\tilde{\mathbf{S}}_{Cij}\mathbf{T}\mathbf{T}^T\right). \tag{S21}$$

Furthermore, for any orthogonal matrix $\mathbf{U}$ (where $\mathbf{U}\mathbf{U}^T = \mathbf{I}$), we can insert it into the expression without changing its value:

$$\mathrm{tr}\left(\tilde{\mathbf{S}}_{Cij}\mathbf{T}\mathbf{T}^T\right) = \mathrm{tr}\left(\tilde{\mathbf{S}}_{Cij}\mathbf{T}\mathbf{U}\mathbf{U}^T\mathbf{T}^T\right). \tag{S22}$$

Let $\mathbf{T}$ be any matrix that satisfies the relaxed constraint $\mathbf{T}^T\mathbf{T} \preccurlyeq \mathbf{I}$. We can always choose a specific orthogonal matrix $\mathbf{U}$ such that it diagonalizes $\mathbf{T}^T\mathbf{T}$, a result from the singular value decomposition. This gives $\mathbf{U}^T\mathbf{T}^T\mathbf{T}\mathbf{U} = \mathbf{\Lambda}$, where $\mathbf{\Lambda}$ is a diagonal matrix of the eigenvalues of $\mathbf{T}^T\mathbf{T}$. The constraint $\mathbf{T}^T\mathbf{T} \preccurlyeq \mathbf{I}$ ensures that these eigenvalues satisfy $\mathbf{\Lambda}_{kk} \le 1$ for all $k$.

Since the matrices $\mathbf{T}^T\mathbf{T}$ and $\mathbf{T}\mathbf{T}^T$ share the same non-zero eigenvalues, it follows that the eigendecomposition of $\mathbf{T}\mathbf{T}^T$ is given by:

$$\mathbf{T}\mathbf{T}^T = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T, \tag{S23}$$

where $\mathbf{V}$ contains the principal eigenvectors of $\mathbf{T}\mathbf{T}^T$ and satisfies $\mathbf{V}^T\mathbf{V} = \mathbf{I}$. Using (S23), we can expand the trace term:

$$\operatorname{tr}\left(\mathbf{T}^T\tilde{\mathbf{S}}_{Cij}\mathbf{T}\right) = \operatorname{tr}\left(\tilde{\mathbf{S}}_{Cij}\mathbf{V}\mathbf{\Lambda}\mathbf{V}^T\right) = \operatorname{tr}\left((\mathbf{V}^T\tilde{\mathbf{S}}_{Cij}\mathbf{V})\mathbf{\Lambda}\right) \tag{S24}$$

$$= \sum_{k=1}^{m}\left(\mathbf{V}^T\tilde{\mathbf{S}}_{Cij}\mathbf{V}\right)_{kk}\mathbf{\Lambda}_{kk} \tag{S25}$$

$$\leq \sum_{k=1}^{m}\left(\mathbf{V}^T\tilde{\mathbf{S}}_{Cij}\mathbf{V}\right)_{kk}, \tag{S26}$$

where the inequality in (S26) holds because $(\mathbf{V}^T\tilde{\mathbf{S}}_{Cij}\mathbf{V})_{kk} \geq 0$ and $\mathbf{\Lambda}_{kk} \leq 1$.

The inequality shows that the objective function takes its maximum value when $\mathbf{\Lambda}_{kk} = 1$ for all $k$, which corresponds to $\mathbf{\Lambda} = \mathbf{I}$. This indicates that the global maximizer of the problem under the relaxed constraint $\mathbf{T}^T\mathbf{T} \preccurlyeq \mathbf{I}$ must inherently satisfy the original, stricter constraint $\mathbf{T}^T\mathbf{T} = \mathbf{I}$. Therefore, the relaxation does not alter the solution, completing the proof of Lemma 3.1. □

## H Proof of (21)

**Proof.** Our objective is to find an explicit expression for the matrix $\mathbf{Q}$ in the equality $\operatorname{tr}\left(\mathbf{A}^{\mathrm{T}}\mathbf{A}\mathbf{\Phi}^t\right) = \mathbf{z}^{\mathrm{T}}\mathbf{Q}\mathbf{z}$. We begin by re-indexing the terms. Let $\tilde{\mathbf{A}}_k = \mathbf{A}_{ij}$ and $\tilde{z}_k = z_{ij}$, where the index $k$ corresponds to the pair $(i, j)$ for $1 \leq i < j \leq C$ via the mapping $k = \frac{(i-1)(2C-i)}{2} + j - i$. The total number of such indices is $K = \frac{C(C-1)}{2}$. The matrix $\mathbf{A}$ can then be written as $\mathbf{A} = \sum_{k=1}^{K}\tilde{z}_k\tilde{\mathbf{A}}_k$.

First, we expand the term $\mathbf{A}^{\mathrm{T}}\mathbf{A}$:

$$\mathbf{A}^{\mathrm{T}}\mathbf{A} = \left(\sum_{k=1}^{K}\tilde{z}_k\tilde{\mathbf{A}}_k\right)^{\mathrm{T}}\left(\sum_{l=1}^{K}\tilde{z}_l\tilde{\mathbf{A}}_l\right) = \sum_{k=1}^{K}\sum_{l=1}^{K}\tilde{z}_k\tilde{z}_l\tilde{\mathbf{A}}_k^{\mathrm{T}}\tilde{\mathbf{A}}_l.$$

Taking the trace after right-multiplying by $\mathbf{\Phi}^t$ yields:

$$\operatorname{tr}\left(\mathbf{A}^{\mathrm{T}}\mathbf{A}\mathbf{\Phi}^t\right) = \sum_{k=1}^{K}\sum_{l=1}^{K}\tilde{z}_k\tilde{z}_l\operatorname{tr}\left(\tilde{\mathbf{A}}_k^{\mathrm{T}}\tilde{\mathbf{A}}_l\mathbf{\Phi}^t\right). \tag{S27}$$

The quadratic form $\mathbf{z}^{\mathrm{T}}\mathbf{Q}\mathbf{z}$ can be expanded as:

$$\mathbf{z}^{\mathrm{T}}\mathbf{Q}\mathbf{z} = \sum_{k=1}^{K}\sum_{l=1}^{K}\tilde{z}_kQ_{k,l}\tilde{z}_l. \tag{S28}$$

By equating the coefficients of $\tilde{z}_k\tilde{z}_l$ in (S27) and (S28), we identify the entries of $\mathbf{Q}$ as:

$$Q_{k,l} = \operatorname{tr}\left(\tilde{\mathbf{A}}_k^{\mathrm{T}}\tilde{\mathbf{A}}_l\mathbf{\Phi}^t\right).$$

To express $\mathbf{Q}$ in a compact matrix form, we use a property of the Kronecker product $\otimes$. Let $\boldsymbol{v}_k = \operatorname{vec}(\tilde{\mathbf{A}}_k)$. The trace term can be written as:

$$\operatorname{tr}\left(\tilde{\mathbf{A}}_k^{\mathrm{T}}\tilde{\mathbf{A}}_l\mathbf{\Phi}^t\right) = \boldsymbol{v}_k^{\mathrm{T}}\left(\mathbf{\Phi}^t \otimes \mathbf{I}\right)\boldsymbol{v}_l,$$

Substituting this into the sum gives:

$$\operatorname{tr}\left(\mathbf{A}^{\mathrm{T}}\mathbf{A}\mathbf{\Phi}^t\right) = \sum_{k=1}^{K}\sum_{l=1}^{K}\tilde{z}_k\left(\boldsymbol{v}_k^{\mathrm{T}}\left(\mathbf{\Phi}^t \otimes \mathbf{I}\right)\boldsymbol{v}_l\right)\tilde{z}_l.$$

Let us construct a matrix $\mathbf{S}$ by stacking the vectors $\boldsymbol{v}_k$ as its columns: $\mathbf{S} = [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_K]$. The expression above can then be rewritten as a matrix-vector product:

$$\operatorname{tr}\left(\mathbf{A}^{\mathrm{T}}\mathbf{A}\boldsymbol{\Phi}^{t}\right) = \mathbf{z}^{\mathrm{T}}\mathbf{S}^{\mathrm{T}}\left(\boldsymbol{\Phi}^{t} \otimes \mathbf{I}\right)\mathbf{S}\mathbf{z}. \tag{S29}$$

Comparing (S29) with $\mathbf{z}^{\mathrm{T}}\mathbf{Q}\mathbf{z}$, we deduce the final form of $\mathbf{Q}$:

$$\mathbf{Q} = \mathbf{S}^{\mathrm{T}}\left(\boldsymbol{\Phi}^{t} \otimes \mathbf{I}\right)\mathbf{S},$$

where $\mathbf{S} = [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_K]$ and $\boldsymbol{v}_k = \operatorname{vec}(\mathbf{A}_{ij})$ with $k = \dfrac{(i-1)(2C-i)}{2} + j - i$. This concludes the proof. □

# I Detailed Dataset Descriptions

We evaluate the performance of our proposed algorithm on six diverse datasets encompassing handwritten digit recognition, face recognition, and natural scene classification. To thoroughly assess the algorithm's scalability, robustness, and feature extraction capabilities, we selected datasets across a wide spectrum of feature space sizes, categorizing them into small-dimensional, medium-dimensional, and large-dimensional subsets.

For our small-dimensional benchmarks, we utilize the Zernike [42] and Digit [43] datasets. The Zernike dataset consists of 47-dimensional Zernike moment features extracted from 2,000 handwritten numerals, evenly distributed across 10 classes. This evaluates the algorithm's ability to handle scale- and rotation-invariant shape representations rather than raw pixel intensities. The Digit dataset serves as another compact benchmark, comprising 1,797 samples of grayscale handwritten digit. As detailed in the baseline parameters, each image is represented by a 64-dimensional feature vector, providing a foundational baseline for evaluating classification performance in tightly constrained low-dimensional spaces.

Moving to medium-dimensional feature spaces, we employ MSRA25 [45] and Yale [46]. The MSRA25 dataset contains 1,799 grayscale face images distributed across 12 identity classes. Each image is represented in a 784-dimensional space, introducing challenges related to complex geometric and appearance characteristics typical in facial imagery. The Yale Face Database consists of 165 grayscale face images from 15 different individuals mapped to a 1,024-dimensional space, specifically testing discriminative dimensionality reduction under varying lighting conditions and facial expressions. Notably, for datasets such as Yale, where the feature dimension significantly exceeds the total number of samples, the pairwise within-class scatter matrices are inherently singular due to the small sample size (SSS) problem. However, this rank deficiency does not pose a concern in our approach because we employ PCA as a preprocessing step. This pre-projection removes the null space of the data, thereby ensuring stable computation.

Finally, to rigorously test the algorithm's performance on large-dimensional data, we include the Outdoor Scene Recognition (OSR) dataset [44] and Stable-ImageNet. The OSR dataset poses the challenge of natural scene classification and consists of 2,688 color images spanning 8 outdoor categories. These images are represented in the original $65{,}536$-dimensional pixel space ($256 \times 256$ resolution), presenting a highly complex high-dimensional setting. Furthermore, the Stable-ImageNet dataset provides an even more demanding benchmark in terms of dimensionality. Inspired by [2], we consider a subset of 10 classes from Stable-ImageNet, with each class containing 1,000 samples. These RGB images are stored at a $512 \times 512$ resolution. Since the raw $786{,}432$-dimensional pixel space is impractical to work with directly, we follow [2] and employ a deep learning-based feature extraction pipeline. Specifically, we use ResNet-50 as the backbone network. By extracting features from the global average pooling layer, we obtain $2{,}048$-dimensional feature vectors. These vectors are further reduced to $128$ dimensions using PCA.

# J Additional Results of four datasets

In this section, we expand upon our primary findings by providing supplementary visualizations of the learned projections, conducting a rigorous statistical significance analysis of the classification improvements, and evaluating the computational efficiency (runtime) of the proposed method against competing worst-case discriminative baselines.

Table S2: Classification error for OSR and Digit datasets. Best projection dimension and standard deviation shown in parentheses.

| Dataset | OSR | | | Digit | | |
|---|---|---|---|---|---|---|
| Method \ Classifier | 1-NN | NM | QDF | 1-NN | NM | QDF |
| LDA (1936, 1948) [16, 17] | 0.2490 (2, Std:0.0174) | 0.2193 (2, Std:0.0118) | 0.2039 (2, Std:0.0123) | 0.1159 (2, Std:0.0255) | 0.0892 (2, Std:0.0223) | 0.0877 (2, Std:0.0226) |
| HLDA (2004) [47] | 0.2677 (2, Std:0.0180) | 0.2368 (2, Std:0.0150) | 0.2155 (2, Std:0.0167) | 0.1512 (2, Std:0.0244) | 0.1214 (2, Std:0.0214) | 0.1211 (2, Std:0.0223) |
| WHMMDA (2015, 2018) [48, 21] | 0.2801 (2, Std:0.0278) | 0.2312 (2, Std:0.0240) | 0.2164 (2, Std:0.0240) | 0.2429 (2, Std:0.0441) | 0.2174 (2, Std:0.0558) | 0.2047 (2, Std:0.0438) |
| WDA-EIG (2020) [49] | 0.2407 (2, Std:0.0159) | 0.2053 (2, Std:0.0129) | 0.1945 (2, Std:0.0141) | 0.1197 (2, Std:0.0289) | 0.0928 (2, Std:0.0218) | 0.0913 (2, Std:0.0221) |
| RLDA-$\ell_{2,1}$ (2021) [1] | 0.3982 (2, Std:0.0233) | 0.3426 (2, Std:0.0177) | 0.3224 (2, Std:0.0180) | 0.1496 (2, Std:0.0141) | 0.1358 (2, Std:0.0153) | 0.1226 (2, Std:0.0144) |
| WDDR (2024) [2] | 0.4572 (2, Std:0.0297) | 0.3801 (2, Std:0.0316) | 0.3729 (2, Std:0.0310) | 0.1570 (2, Std:0.0158) | 0.1945 (2, Std:0.0340) | 0.1186 (2, Std:0.0141) |
| QP-MMDA (2025) [25] | 0.2579 (2, Std:0.0309) | 0.2041 (2, Std:0.0334) | 0.1963 (2, Std:0.0264) | 0.2486 (2, Std:0.0411) | 0.2279 (2, Std:0.0566) | 0.2172 (2, Std:0.0525) |
| QP-MMRA (2026) [35] | 0.4799 (2, Std:0.0365) | 0.3806 (2, Std:0.0301) | 0.3815 (2, Std:0.0332) | 0.1516 (2, Std:0.0136) | 0.1203 (2, Std:0.0109) | 0.1149 (2, Std:0.0120) |
| $F^3$-PWCRA | **0.0719** (2, Std:0.0390) | **0.0130** (2, Std:0.0055) | **0.0827** (2, Std:0.0310) | **0.0645** (2, Std:0.0278) | **0.0540** (2, Std:0.0248) | **0.0486** (2, Std:0.0242) |

Table S3: Classification error for Zernike and Yale datasets. Best projection dimension and standard deviation shown in parentheses.

| Dataset | Zernike | | | Yale | | |
|---|---|---|---|---|---|---|
| Method \ Classifier | 1-NN | NM | QDF | 1-NN | NM | QDF |
| LDA (1936, 1948) [16, 17] | 0.2821 (2, Std:0.0220) | 0.2548 (2, Std:0.0249) | 0.2513 (2, Std:0.0258) | 0.2400 (2, Std:0.1516) | 0.2760 (2, Std:0.1596) | 0.3508 (2, Std:0.1832) |
| HLDA (2004) [47] | 0.3340 (2, Std:0.0310) | 0.2949 (2, Std:0.0305) | 0.2625 (2, Std:0.0284) | 0.4072 (2, Std:0.2075) | 0.3888 (2, Std:0.2161) | 0.4004 (2, Std:0.2024) |
| WHMMDA (2015, 2018) [48, 21] | 0.3373 (2, Std:0.0441) | 0.2690 (2, Std:0.0387) | 0.2618 (2, Std:0.0397) | 0.4016 (2, Std:0.1737) | 0.4184 (2, Std:0.1607) | 0.4520 (2, Std:0.1698) |
| WDA-EIG (2020) [49] | 0.3133 (2, Std:0.0908) | 0.2533 (2, Std:0.0765) | 0.2516 (2, Std:0.0726) | 0.6584 (2, Std:0.1105) | 0.6376 (2, Std:0.1043) | 0.7012 (2, Std:0.0831) |
| RLDA-$\ell_{2,1}$ (2021) [1] | 0.3326 (2, Std:0.0177) | 0.2793 (2, Std:0.0150) | 0.2829 (2, Std:0.0149) | 0.1868 (2, Std:0.0735) | 0.1712 (2, Std:0.0710) | 0.2260 (2, Std:0.0914) |
| WDDR (2024) [2] | 0.2466 (2, Std:0.0553) | 0.1912 (2, Std:0.0502) | 0.1801 (2, Std:0.0460) | 0.2676 (2, Std:0.2031) | 0.2788 (2, Std:0.2060) | 0.2700 (2, Std:0.1987) |
| QP-MMDA (2025) [25] | 0.3462 (2, Std:0.0410) | 0.2716 (2, Std:0.0355) | 0.2610 (2, Std:0.0355) | 0.2763 (2, Std:0.0426) | 0.4193 (2, Std:0.0438) | 0.3210 (2, Std:0.0487) |
| QP-MMRA (2026) [35] | 0.3475 (2, Std:0.0236) | 0.2932 (2, Std:0.0216) | 0.2986 (2, Std:0.0206) | 0.5632 (2, Std:0.2493) | 0.6568 (2, Std:0.2078) | 0.7000 (2, Std:0.1526) |
| $F^3$-PWCRA | **0.1349** (2, Std:0.0415) | **0.1020** (2, Std:0.0268) | **0.0993** (2, Std:0.0280) | **0.0916** (2, Std:0.0597) | **0.1168** (2, Std:0.0672) | **0.1652** (2, Std:0.0886) |

### J.1 Examination of 1NN, NM, and QDA

Table S2 reports the classification error rates of the proposed $F^3$-PWCRA framework alongside several baseline and state-of-the-art methods on the OSR and Digit datasets. On the OSR dataset, $F^3$-PWCRA achieves the lowest error rates across all classifiers, obtaining 0.0719 (1-NN), 0.0130 (NM), and 0.0827 (QDF). These results represent a substantial improvement over classical LDA (0.2039–0.2490) as well as more recent approaches such as WDA-EIG (0.1945–0.2407) and QP-MMDA (0.1963–0.2579), with a particularly notable gain observed under the NM classifier.

On the Digit dataset, the proposed method similarly outperforms all competing techniques, recording error rates of 0.0645 (1-NN), 0.0540 (NM), and 0.0486 (QDF). These values are consistently lower than those obtained by LDA (0.0877–0.1159), QP-MMDA, and QP-MMRA, demonstrating the strong discriminative power of the learned projections for handwritten digit recognition.

Table S3 presents the classification performance on the Zernike and Yale datasets. For the Zernike dataset, $F^3$-PWCRA achieves the best results with error rates of 0.1349 (1-NN), 0.1020 (NM), and 0.0993 (QDF), clearly outperforming LDA (0.2513–0.2821) and the competitive WDDR method (0.1801–0.2466). This highlights the framework's effectiveness in capturing discriminative information from orthogonal moment-based features.

On the more challenging Yale dataset, characterized by significant illumination and pose variations, $F^3$-PWCRA maintains its superiority by yielding the lowest error rates of 0.0916 (1-NN), 0.1168 (NM), and 0.1652 (QDF). The proposed method substantially outperforms both classical and recent

robust approaches, many of which show considerable degradation in performance under these conditions.

Overall, the results on these four datasets further validate the robustness and generalization capability of F$^3$-PWCRA across diverse data domains and classification schemes.

### J.2 Additional Visualizations of Learned Projections

Figures S2 and S3 present the 2D projected spaces for the few classes of the OSR, Digit, Zernike, and Yale datasets. Beginning with the OSR dataset (Figure S2, top), methods such as LDA, HLDA, WDA-EIG, WHMMDA, RLDA-$\ell_{2,1}$, and QP-MMDA yield the poorest visual performance. These algorithms exhibit severe shortcomings, frequently projecting one class entirely over another — a failure that extends far beyond standard boundary overlap. For instance, under LDA, HLDA, and WDA-EIG, class 2 is entirely subsumed by class 4. Similarly, for WHMMDA and QP-MMDA, class 1 is completely overshadowed by class 3, while RLDA-$\ell_{2,1}$ forces class 2 to be entirely enveloped by class 3. Methods like WDDR and QP-MMRA demonstrate slightly better structural retention, yet still suffer from excessive inter-class overlapping. In the WDDR projection, for example, while class 1 is visually more distinct than in the worst-performing methods, it still exhibits extreme overlap with class 3. QP-MMRA provides the second-best projection on this dataset, but approximately half of class 2 remains heavily occluded by classes 3 and 4. This trend is strongly reinforced in the Digit (Figure S2, bottom) and Zernike (Figure S3, top) datasets. Across both datasets, the baseline methods exhibit similar structural flaws, characterized by extensive class occlusion and boundary merging. However, F$^3$-PWCRA successfully achieves near-perfect class separation without any overlap, highlighting its robust capability to tightly cluster samples within their respective categories while simultaneously pushing the classes apart. Finally, the Yale dataset (Figure S3, bottom) presents a unique challenge due to its naturally high intra-class variance and dispersion. As seen in the projections, almost all competing methods yield highly chaotic and dispersed distributions. This failure stems directly from their algorithmic designs, which either lack an explicit penalty for intra-class dispersion or rely on mathematical relaxations that inevitably result in suboptimal projection matrices. Notably, while WDDR demonstrates better structural coherence than traditional baselines by explicitly penalizing intra-class variance, its performance is constrained by suboptimal formulation relaxations. In contrast, F$^3$-PWCRA operates without these loose relaxations, directly and rigorously bounding the variance of each class. This exact formulation ensures superior structural integrity and tighter clustering, successfully preventing class overlap even in highly dispersed datasets like Yale.

### J.3 Statistical Significance Analysis

To assess whether the observed performance improvements of F$^3$-PWCRA are statistically significant, we performed paired t-tests against all competing methods across the $100$ independent runs for each dataset and classifier (with the projection dimension fixed at $2$). Table S4 reports results against the *statistically strongest competitor* in each setting — defined as the baseline yielding the largest $p$-value (i.e., the most challenging comparison).

Note that this statistically closest competitor does not always coincide with the method having the second-lowest mean error in Tables 2, S2, and S3. Mean error rankings reflect average performance, whereas the paired t-test evaluates the consistency and magnitude of per-run differences. We adopt this approach for rigor: it demonstrates that F$^3$-PWCRA remains superior even against the competitor to which it is most statistically similar.

As shown in Table S4, F$^3$-PWCRA achieves statistically significant improvements over the strongest competing method in the large majority of cases. Highly significant differences ($p \ll 0.05$) are observed for most dataset–classifier combinations, particularly with the 1-NN and QDF classifiers. The only exceptions are Stable ImageNet with the NM classifier ($p = 0.215$) and MSRA-25 with the NM classifier ($p = 0.162$), where the differences did not reach statistical significance at the $0.05$ level. Overall, these results provide strong evidence supporting the effectiveness and reliability of the proposed framework.

### J.4 Computational Complexity Analysis

A major challenge in worst-case discriminant analysis lies in the substantial computational cost associated with solving the underlying minimax optimization problems. To enable a fair and meaningful

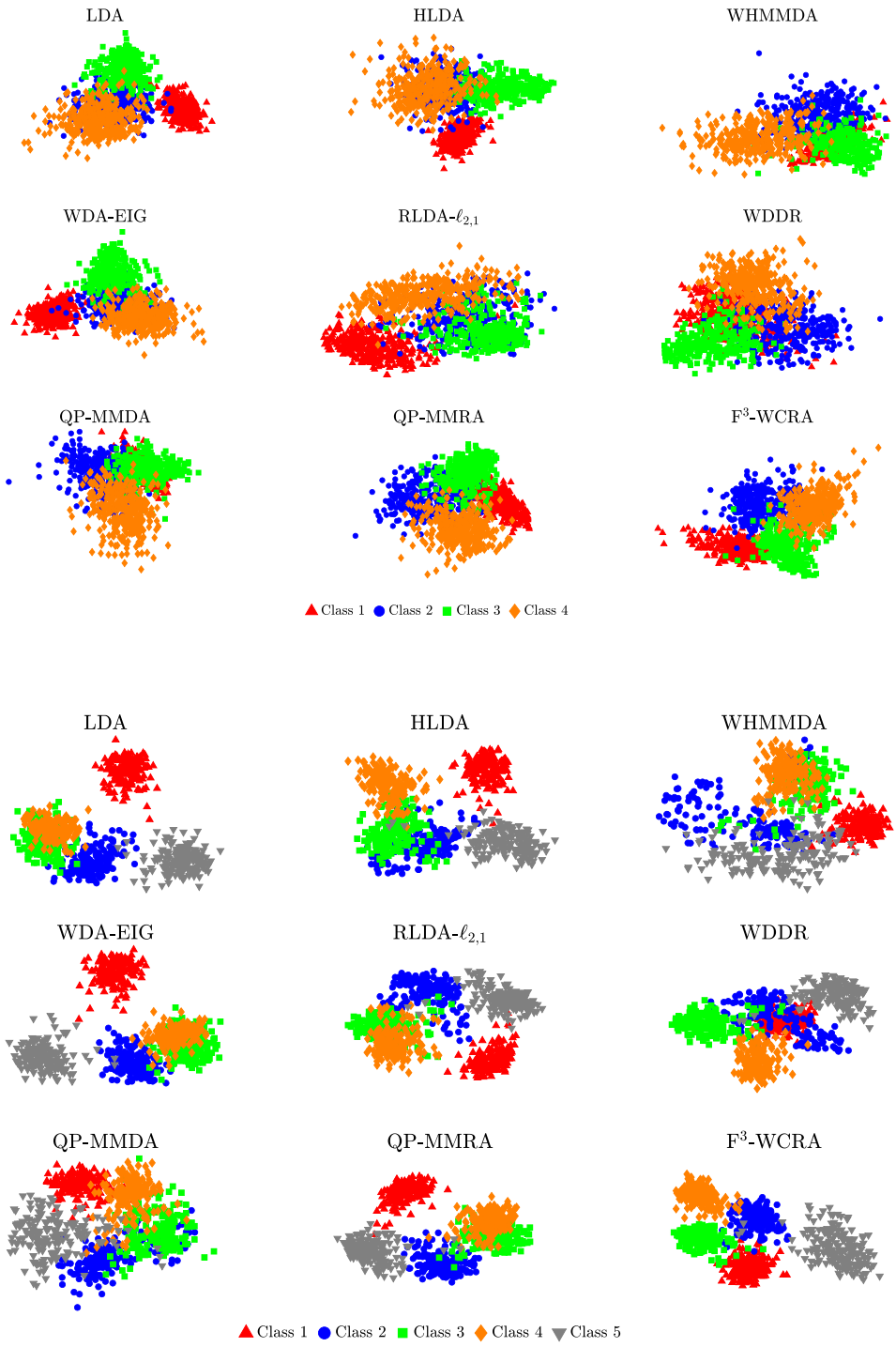


Figure S2: Visualization of the learned projections of the OSR dataset (top) and the Digit dataset (bottom), where the target subspace dimensionality for all methods was set to $m = 2$.

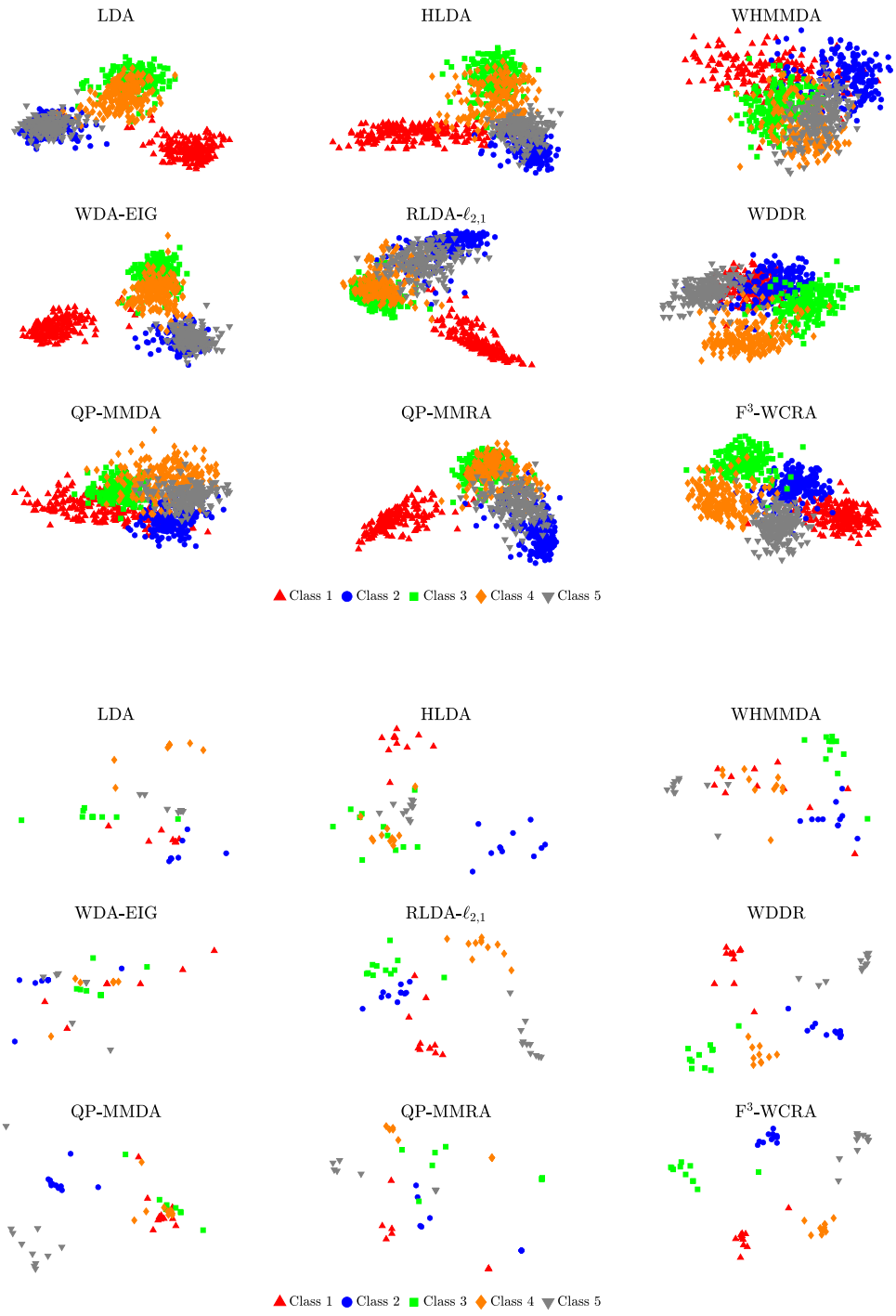


Figure S3: Visualization of the learned projections of the Zernike dataset (top) and the Yale dataset (bottom), where the target subspace dimensionality for all methods was set to $m = 2$.

Table S4: Statistical significance analysis using paired t-tests over 100 runs (dimension = 2). We report p-values comparing the proposed method ($F^3$-PWCRA) against the next best competing method for each dataset and classifier. Statistically significant results ($p < 0.05$) are highlighted in bold.

| Dataset | Classifier | Next Best Method | p-value | Significance |
|---|---|---|---|---|
| Stable ImageNet | 1-NN | HLDA | $\mathbf{1.09 \times 10^{-6}}$ | Significant |
| | NM | HLDA | $\mathbf{6.93 \times 10^{-9}}$ | Significant |
| | QDF | HLDA | $2.15 \times 10^{-1}$ | Not Significant |
| MSRA-25 | 1-NN | LDA | $\mathbf{2.36 \times 10^{-3}}$ | Significant |
| | NM | LDA | $1.62 \times 10^{-1}$ | Not Significant |
| | QDF | LDA | $\mathbf{3.13 \times 10^{-3}}$ | Significant |
| OSR | 1-NN | WDA-EIG | $\mathbf{2.66 \times 10^{-35}}$ | Significant |
| | NM | WDA-EIG | $\mathbf{1.88 \times 10^{-17}}$ | Significant |
| | QDF | WDDR | $\mathbf{1.18 \times 10^{-43}}$ | Significant |
| Digit | 1-NN | WDA-EIG | $\mathbf{4.70 \times 10^{-42}}$ | Significant |
| | NM | WDA-EIG | $\mathbf{4.90 \times 10^{-32}}$ | Significant |
| | QDF | WDA-EIG | $\mathbf{4.53 \times 10^{-38}}$ | Significant |
| Zernike | 1-NN | WDDR | $\mathbf{2.12 \times 10^{-40}}$ | Significant |
| | NM | WDDR | $\mathbf{6.75 \times 10^{-37}}$ | Significant |
| | QDF | WDDR | $\mathbf{3.87 \times 10^{-37}}$ | Significant |
| Yale | 1-NN | RLDA-$\ell_{2,1}$ | $\mathbf{1.51 \times 10^{-8}}$ | Significant |
| | NM | RLDA-$\ell_{2,1}$ | $\mathbf{1.55 \times 10^{-4}}$ | Significant |
| | QDF | WDDR | $\mathbf{4.45 \times 10^{-3}}$ | Significant |

Table S5: Runtime (seconds, mean $\pm$ std over 100 runs) for worst-case discriminative methods on Stable ImageNet, MSRA-25, and OSR datasets for dimension = 2.

| Method | Stable ImageNet | MSRA-25 | OSR |
|---|---|---|---|
| WHMMDA (2015, 2018) [48, 21] | 1132.7399 $\pm$ 97.7042 | 21.8363 $\pm$ 2.5277 | 21.2594 $\pm$ 2.6933 |
| WDDR (2024) [2] | 732.2010 $\pm$ 93.8295 | 9.4861 $\pm$ 0.7409 | 13.7357 $\pm$ 1.2432 |
| QP-MMDA (2025) [25] | 1.4750 $\pm$ 0.3218 | 2.3618 $\pm$ 0.4447 | 1.8242 $\pm$ 0.3391 |
| QP-MMRA (2026) [35] | 1.6166 $\pm$ 1.0087 | **1.5342 $\pm$ 0.1941** | 1.5980 $\pm$ 0.1287 |
| $F^3$-PWCRA (2026) | **1.0274 $\pm$ 0.1231** | 1.5612 $\pm$ 0.1445 | **1.5514 $\pm$ 0.2013** |

Table S6: Runtime (seconds, mean $\pm$ std over 100 runs) for worst-case discriminative methods on Digit, Zernike, and Yale datasets for dimension = 2.

| Method | Digit | Zernike | Yale |
|---|---|---|---|
| WHMMDA (2015, 2018) [48, 21] | 21.0371 $\pm$ 2.6901 | 17.5927 $\pm$ 2.4552 | 20.2891 $\pm$ 3.4288 |
| WDDR (2024) [2] | 8.0569 $\pm$ 0.7353 | 8.4768 $\pm$ 1.0714 | 9.1750 $\pm$ 1.4087 |
| QP-MMDA (2025) [25] | 2.1400 $\pm$ 0.3962 | 1.9354 $\pm$ 0.3257 | 2.3334 $\pm$ 0.3437 |
| QP-MMRA (2026) [35] | 1.6697 $\pm$ 0.2359 | 1.8534 $\pm$ 0.2537 | 1.8575 $\pm$ 0.4210 |
| $F^3$-PWCRA | **1.4729 $\pm$ 0.1130** | **1.6940 $\pm$ 0.1798** | **1.4491 $\pm$ 0.1304** |

comparison of runtime performance, the analysis in Tables S5 and S6 is restricted exclusively to methods specifically designed for worst-case class separability. Traditional mean-based discriminant analysis methods (e.g., LDA) are deliberately excluded.

While mean-based approaches rely on aggregated class statistics and lead to standard eigenvalue or least-squares problems, worst-case methods must explicitly consider the separation margin between every pair of classes under the most adverse conditions. This inherently leads to a more demanding minimax ($\max - \min$ or $\min - \max$) optimization framework. Consequently, direct runtime comparisons between mean-based and worst-case methods would be unfair. By focusing solely on worst-case algorithms, we provide a consistent and equitable assessment of how efficiently each method tackles the intrinsic computational challenges of minimax optimization.

All runtime comparisons were conducted using the same per-iteration examination inspired by [25]. Each method employs its own convergence criterion (some based on the change in objective value, others on the difference between successive solutions) and tolerance thresholds (e.g., $10^{-3}$, $10^{-4}$, etc.). This ensures that every algorithm is evaluated under its own convergence conditions, allowing a fair efficiency assessment.

The experimental results reveal that earlier worst-case methods suffer from considerable computational overhead. In particular, WHMMDA requires over 1000 seconds on the Stable ImageNet dataset and approximately 17–21 seconds on the remaining datasets. Although WDDR offers a noticeable improvement, it still demands roughly 8–13 seconds per execution across most datasets. In contrast, QP-MMDA, QP-MMRA, and the proposed F$^3$-PWCRA significantly reduce the runtime to approximately one to two seconds. Notably, F$^3$-PWCRA consistently ranks as the fastest or among the fastest methods across all six datasets. It achieves the lowest runtime on Stable ImageNet (1.0274 s), Digit (1.4729 s), Zernike (1.6940 s), and Yale (1.4491 s). These results correspond to roughly an order-of-magnitude speedup relative to WHMMDA and a 5–6$\times$ improvement compared to WDDR. While QP-MMRA also demonstrates competitive efficiency, F$^3$-PWCRA outperforms it on several datasets, most notably on Stable ImageNet.

Overall, the runtime analysis demonstrates that F$^3$-PWCRA achieves an excellent balance between computational efficiency and discriminative power, delivering state-of-the-art classification performance at remarkably low computational cost.

### J.5 Performance Evaluation with Three-Dimensional Projections

To further examine the behavior of the competing methods in a higher-dimensional subspace, we repeated the experimental protocol with a target dimensionality of $m = 3$.

Tables S7, S8, and S9 present the classification error rates for $m = 3$. On the Stable ImageNet dataset, F$^3$-PWCRA achieves the lowest error rates across all three classifiers, with errors of 0.1248 (1-NN), 0.0958 (NM), and 0.1148 (QDF). These results outperform the best baseline, LDA, by approximately 10% to 27% depending on the classifier. For the MSRA-25 dataset, all methods yield very low error rates owing to its relative simplicity. Nevertheless, F$^3$-PWCRA attains near-perfect classification with an error of 0.0015 for both 1-NN and QDF, matching or slightly surpassing the strongest competing methods.

On the OSR dataset, performance differences are more nuanced. While WDA-EIG and WDDR achieve the best results for 1-NN and QDF, respectively, F$^3$-PWCRA obtains the lowest error for the NM classifier (0.0126). On the Digit dataset, F$^3$-PWCRA yields the lowest classification errors across all classifiers. Similarly, on the Zernike dataset, it consistently outperforms all other methods, reducing the error relative to the best baseline (WDDR) by 17% to 22%. Finally, on the more challenging Yale dataset, which exhibits large intra-class variations due to differences in illumination, expression, and pose, F$^3$-PWCRA achieves substantially lower error rates (0.0644 for 1-NN, 0.0928 for NM, and 0.1384 for QDF) compared to the strongest baselines.

Overall, these results indicate that F$^3$-PWCRA maintains its competitive advantage when moving from a two-dimensional to a three-dimensional projection subspace, with particularly pronounced improvements on more difficult datasets such as Stable ImageNet, Zernike, and Yale.

To evaluate the statistical significance of the performance differences at a higher projection dimension, we performed paired t-tests across the 100 independent runs for each dataset and classifier when $m = 3$. Table S10 reports the $p$-values comparing F$^3$-PWCRA against the statistically strongest competitor in each setting (i.e., the baseline yielding the largest $p$-value). A significance level of $p < 0.05$ was adopted.

The analysis reveals that F$^3$-PWCRA achieves statistically significant improvements over the strongest competing method in the vast majority of dataset–classifier combinations. Significant gains are consistently observed on Stable ImageNet, OSR, Zernike, and Yale, as well as across all classifiers on the MSRA-25 dataset. The only exceptions are found on the Digit dataset, where the performance differences do not reach statistical significance at the $p < 0.05$ level.

Tables S11 and S12 present the runtime performance (in seconds, mean $\pm$ std over 100 runs) of the worst-case discriminative methods at projection dimension $m = 3$. Exactly as in the $m = 2$ case, all runtime comparisons were conducted using per-iteration examination. While WHMMDA and

Table S7: Classification error for Stable ImageNet and MSRA-25 datasets. Best projection dimension and standard deviation shown in parentheses.

| Dataset | Stable ImageNet | | | MSRA-25 | | |
|---|---|---|---|---|---|---|
| Method \ Classifier | 1-NN | NM | QDF | 1-NN | NM | QDF |
| LDA (1936, 1948) [16, 17] | 0.1396 (3, Std:0.0213) | 0.1312 (3, Std:0.0230) | 0.1278 (3, Std:0.0235) | 0.0017 (3, Std:0.0014) | 0.0017 (3, Std:0.0014) | 0.0015 (3, Std:0.0014) |
| HLDA (2004) [47] | 0.1440 (3, Std:0.0327) | 0.1391 (3, Std:0.0293) | 0.1283 (3, Std:0.0312) | 0.0109 (3, Std:0.0051) | 0.0315 (3, Std:0.0125) | 0.0090 (3, Std:0.0042) |
| WHMMDA (2015, 2018)[48, 21] | 0.1887 (3, Std:0.0346) | 0.1643 (3, Std:0.0294) | 0.1558 (3, Std:0.0302) | 0.0495 (3, Std:0.0222) | 0.1494 (3, Std:0.0462) | 0.0761 (3, Std:0.0286) |
| WDA-EIG (2020) [49] | 0.1983 (3, Std:0.0370) | 0.1919 (3, Std:0.0357) | 0.1962 (3, Std:0.0367) | 0.0043 (3, Std:0.0084) | 0.0040 (3, Std:0.0072) | 0.0040 (3, Std:0.0076) |
| RLDA-$\ell_{2,1}$ (2021)[1] | 0.3211 (3, Std:0.0291) | 0.2681 (3, Std:0.0270) | 0.2516 (3, Std:0.0225) | 0.0270 (3, Std:0.0108) | 0.0504 (3, Std:0.0159) | 0.0326 (3, Std:0.0122) |
| WDDR (2024)[2] | 0.2421 (3, Std:0.0952) | 0.2066 (3, Std:0.0630) | 0.2056 (3, Std:0.0588) | 0.0066 (3, Std:0.0073) | 0.0072 (3, Std:0.0072) | 0.0057 (3, Std:0.0066) |
| QP-MMDA (2025)[25] | 0.2073 (3, Std:0.0365) | 0.1793 (3, Std:0.0307) | 0.1698 (3, Std:0.0316) | 0.0480 (3, Std:0.0186) | 0.1141 (3, Std:0.0364) | 0.0732 (3, Std:0.0219) |
| QP-MMRA (2026) [35] | 0.3927 (3, Std:0.0504) | 0.2865 (3, Std:0.0390) | 0.3019 (3, Std:0.0384) | 0.0018 (3, Std:0.0155) | 0.0022 (3, Std:0.0198) | 0.0031 (3, Std:0.0197) |
| $F^3$-PWCRA | **0.1248** (3, Std:0.0248) | **0.0958** (3, Std:0.0193) | **0.1148** (3, Std:0.0246) | **0.0015** (3, Std:0.0022) | 0.0018 (3, Std:0.0027) | **0.0015** (3, Std:0.0022) |

Table S8: Classification error for Outdoor Scene Recognition (OSR) and Digit datasets. Best projection dimension and standard deviation shown in parentheses.

| Dataset | Outdoor Scene Recognition (OSR) | | | Digit | | |
|---|---|---|---|---|---|---|
| Method \ Classifier | 1-NN | NM | QDF | 1-NN | NM | QDF |
| LDA (1936, 1948) [16, 17] | 0.0161 (3, Std:0.0069) | 0.0245 (3, Std:0.0074) | 0.0142 (3, Std:0.0059) | 0.0291 (3, Std:0.0085) | 0.0275 (3, Std:0.0083) | 0.0241 (3, Std:0.0087) |
| HLDA (2004) [47] | 0.0146 (3, Std:0.0063) | 0.0252 (3, Std:0.0092) | 0.0186 (3, Std:0.0076) | 0.0531 (3, Std:0.0153) | 0.0769 (3, Std:0.0125) | 0.0584 (3, Std:0.0134) |
| WHMMDA (2015, 2018) [48, 21] | 0.0152 (3, Std:0.0069) | 0.0215 (3, Std:0.0067) | 0.0178 (3, Std:0.0080) | 0.0761 (3, Std:0.0187) | 0.1037 (3, Std:0.0148) | 0.0759 (3, Std:0.0163) |
| WDA-EIG (2020) [49] | **0.0137** (3, Std:0.0058) | 0.0193 (3, Std:0.0063) | 0.0280 (3, Std:0.0102) | 0.0289 (3, Std:0.0075) | 0.0248 (3, Std:0.0078) | 0.0230 (3, Std:0.0081) |
| RLDA-$\ell_{2,1}$ (2021) [1] | 0.0864 (3, Std:0.0156) | 0.1012 (3, Std:0.0136) | 0.0751 (3, Std:0.0129) | 0.0412 (3, Std:0.0083) | 0.0720 (3, Std:0.0097) | 0.0460 (3, Std:0.0085) |
| WDDR (2024) [2] | 0.0156 (3, Std:0.0059) | 0.0220 (3, Std:0.0065) | **0.0131** (3, Std:0.0061) | 0.0202 (3, Std:0.0059) | 0.0199 (3, Std:0.0063) | 0.0188 (3, Std:0.0068) |
| QP-MMDA (2025) [25] | 0.0168 (3, Std:0.0075) | 0.0234 (3, Std:0.0081) | 0.0185 (3, Std:0.0072) | 0.0751 (3, Std:0.0158) | 0.1040 (3, Std:0.0149) | 0.0776 (3, Std:0.0140) |
| QP-MMRA (2026) [35] | 0.0377 (3, Std:0.0290) | 0.0843 (3, Std:0.0368) | 0.3257 (3, Std:0.1739) | 0.0206 (3, Std:0.0060) | 0.0210 (3, Std:0.0071) | 0.0173 (3, Std:0.0065) |
| $F^3$-PWCRA | 0.0730 (3, Std:0.0349) | **0.0126** (3, Std:0.0057) | 0.0954 (3, Std:0.0295) | **0.0197** (3, Std:0.0056) | **0.0187** (3, Std:0.0070) | **0.0171** (3, Std:0.0065) |

Table S9: Classification error for Zernike and Yale datasets. Best projection dimension and standard deviation shown in parentheses.

| Dataset | Zernike | | | Yale | | |
|---|---|---|---|---|---|---|
| Method \ Classifier | 1-NN | NM | QDF | 1-NN | NM | QDF |
| LDA (1936, 1948) [16, 17] | 0.1074 (3, Std:0.0316) | 0.0862 (3, Std:0.0283) | 0.0885 (3, Std:0.0285) | 0.1308 (3, Std:0.0777) | 0.1960 (3, Std:0.1112) | 0.2244 (3, Std:0.0993) |
| HLDA (2004) [47] | 0.2135 (3, Std:0.0310) | 0.1938 (3, Std:0.0279) | 0.1723 (3, Std:0.0257) | 0.3552 (3, Std:0.2187) | 0.3484 (3, Std:0.2328) | 0.4100 (3, Std:0.2172) |
| WHMMDA (2015, 2018) [48, 21] | 0.2319 (3, Std:0.0453) | 0.1943 (3, Std:0.0394) | 0.1810 (3, Std:0.0360) | 0.3044 (3, Std:0.2191) | 0.3224 (3, Std:0.2237) | 0.3880 (3, Std:0.1992) |
| WDA-EIG (2020) [49] | 0.1086 (3, Std:0.0182) | 0.0893 (3, Std:0.0170) | 0.0883 (3, Std:0.0174) | 0.6300 (3, Std:0.1069) | 0.5712 (3, Std:0.1094) | 0.6404 (3, Std:0.0789) |
| RLDA-$\ell_{2,1}$ (2021) [1] | 0.1992 (3, Std:0.0176) | 0.1761 (3, Std:0.0116) | 0.1684 (3, Std:0.0144) | 0.1396 (3, Std:0.0558) | 0.1344 (3, Std:0.0541) | 0.2232 (3, Std:0.0883) |
| WDDR (2024) [2] | 0.0926 (3, Std:0.0264) | 0.0735 (3, Std:0.0190) | 0.0704 (3, Std:0.0177) | 0.2428 (3, Std:0.1756) | 0.2776 (3, Std:0.2107) | 0.3284 (3, Std:0.2082) |
| QP-MMDA (2025) [25] | 0.2254 (3, Std:0.0361) | 0.1890 (3, Std:0.0311) | 0.1769 (3, Std:0.0302) | 0.2832 (3, Std:0.2116) | 0.3236 (3, Std:0.2226) | 0.4456 (3, Std:0.1861) |
| QP-MMRA (2026) [35] | 0.1089 (3, Std:0.0441) | 0.0911 (3, Std:0.0366) | 0.0860 (3, Std:0.0366) | 0.6604 (3, Std:0.1797) | 0.7152 (3, Std:0.1698) | 0.7640 (3, Std:0.1333) |
| $F^3$-PWCRA | **0.0718** (3, Std:0.0129) | **0.0596** (3, Std:0.0105) | **0.0582** (3, Std:0.0110) | **0.0644** (3, Std:0.0545) | **0.0928** (3, Std:0.0703) | **0.1384** (3, Std:0.0867) |

Table S10: Statistical significance analysis using paired t-tests over 100 runs (dimension = 3). We report p-values comparing the proposed method against the next best competing method for each dataset and classifier. Statistically significant results ($p < 0.05$) are highlighted in bold.

| Dataset | Classifier | Next Best Method | p-value | Significance |
|---|---|---|---|---|
| Stable ImageNet | 1-NN | LDA | $\mathbf{1.41 \times 10^{-3}}$ | Significant |
| | NM | LDA | $\mathbf{4.79 \times 10^{-24}}$ | Significant |
| | QDF | LDA | $\mathbf{4.25 \times 10^{-5}}$ | Significant |
| MSRA-25 | 1-NN | QP-MMRA | $\mathbf{2.99 \times 10^{-9}}$ | Significant |
| | NM | QP-MMRA | $\mathbf{3.67 \times 10^{-9}}$ | Significant |
| | QDF | QP-MMRA | $\mathbf{2.40 \times 10^{-4}}$ | Significant |
| OSR | 1-NN | WDA-EIG | $\mathbf{7.35 \times 10^{-33}}$ | Significant |
| | NM | WDA-EIG | $\mathbf{1.86 \times 10^{-14}}$ | Significant |
| | QDF | WDDR | $\mathbf{1.72 \times 10^{-46}}$ | Significant |
| Digit | 1-NN | WDDR | $3.34 \times 10^{-1}$ | Not Significant |
| | NM | WDDR | $2.38 \times 10^{-1}$ | Not Significant |
| | QDF | QP-MMRA | $3.77 \times 10^{-1}$ | Not Significant |
| Zernike | 1-NN | WDDR | $\mathbf{2.73 \times 10^{-8}}$ | Significant |
| | NM | WDDR | $\mathbf{6.40 \times 10^{-9}}$ | Significant |
| | QDF | WDDR | $\mathbf{5.35 \times 10^{-7}}$ | Significant |
| Yale | 1-NN | RLDA-$\ell_{2,1}$ | $\mathbf{8.89 \times 10^{-20}}$ | Significant |
| | NM | RLDA-$\ell_{2,1}$ | $\mathbf{2.74 \times 10^{-7}}$ | Significant |
| | QDF | RLDA-$\ell_{2,1}$ | $\mathbf{7.10 \times 10^{-8}}$ | Significant |

Table S11: Runtime (seconds, mean $\pm$ std over 100 runs) for worst-case discriminative methods on Stable ImageNet, MSRA-25, and OSR datasets for dimension = 3.

| Method | Stable ImageNet | MSRA-25 | OSR |
|---|---|---|---|
| WHMMDA (2015, 2018) [48, 21] | 1078.1602 $\pm$ 241.4341 | 19.4446 $\pm$ 2.4912 | 12.9040 $\pm$ 1.3834 |
| WDDR (2024) [2] | 697.4449 $\pm$ 34.9674 | 7.9422 $\pm$ 1.1850 | 7.6056 $\pm$ 0.4243 |
| QP-MMDA (2025) [25] | 1.4309 $\pm$ 0.3157 | 1.4316 $\pm$ 0.2922 | 1.0947 $\pm$ 0.2678 |
| QP-MMRA (2026) [35] | 1.6115 $\pm$ 1.0009 | 1.0499 $\pm$ 0.0844 | **0.8259 $\pm$ 0.0559** |
| F$^3$-PWCRA (2026) | **0.9993 $\pm$ 0.0968** | **0.9547 $\pm$ 0.0888** | 0.8793 $\pm$ 0.0628 |

Table S12: Runtime (seconds, mean $\pm$ std over 100 runs) for worst-case discriminative methods on Digit, Zernike, and Yale datasets for dimension = 3.

| Method | Digit | Zernike | Yale |
|---|---|---|---|
| WHMMDA (2015, 2018) [48, 21] | 18.1788 $\pm$ 2.2179 | 11.4318 $\pm$ 1.5325 | 14.7381 $\pm$ 2.1096 |
| WDDR (2024) [2] | 10.3236 $\pm$ 1.1582 | 5.2012 $\pm$ 0.5114 | 11.3669 $\pm$ 1.7527 |
| QP-MMDA (2025) [25] | 1.7997 $\pm$ 0.3425 | 1.3068 $\pm$ 0.2875 | 2.0974 $\pm$ 0.4217 |
| QP-MMRA (2026) [35] | 1.3923 $\pm$ 0.1377 | 1.0340 $\pm$ 0.0788 | 1.5459 $\pm$ 0.2677 |
| F$^3$-PWCRA (2026) | **1.3746 $\pm$ 0.1196** | **0.9750 $\pm$ 0.0793** | **1.3760 $\pm$ 0.1100** |

WDDR require several hundred to over a thousand seconds on the larger datasets (most notably Stable ImageNet), F$^3$-PWCRA consistently achieves runtimes below 1.4 seconds across all six datasets. Furthermore, the proposed method ranks first or second in computational speed on every dataset, with consistently low standard deviations, highlighting its superior efficiency and numerical stability.

Overall, these results demonstrate that similarly for a reduced dimension of $m = 3$, F$^3$-PWCRA serves as a highly efficient solution among worst-case discriminative methods, achieving significant speedups without compromising performance.

## K Convergence Analysis and Sensitivity to Initialization

We empirically evaluate the convergence behavior of the proposed optimization framework and examine whether the final solution depends on the choice of initial projection matrix.

### K.1 Outer Loop Convergence

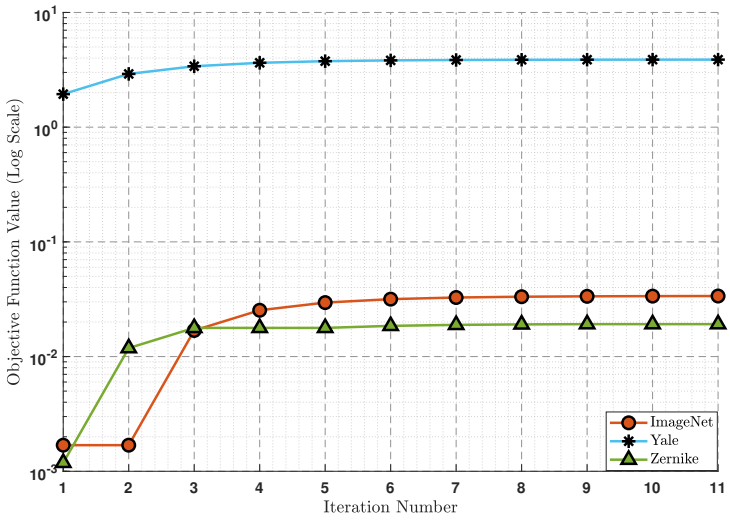


Figure S4: Outer loop convergence on ImageNet, Yale, and Zernike datasets.

Figure S4 illustrates the evolution of the outer loop objective, $\delta$, across iterations for the ImageNet, Yale, and Zernike datasets. In each case, the objective increases monotonically at every bisection step, directly reflecting the theoretical guarantee that the feasible lower bound improves as the search interval narrows. Convergence is consistently achieved within 10 to 15 iterations, after which the objective stabilizes within the prescribed tolerance.While Figure S4 confirms the convergence of the objective, it is essential to investigate the stability of the underlying optimization. Since the outer loop is governed by a deterministic bisection (or Dinkelbach-type) logic, the primary source of potential sensitivity lies within the inner MM-based solver, which handles the non-convex quadratic subproblem. Consequently, we focus our sensitivity analysis on the inner loop's behavior in the following subsection to determine if it consistently reaches the same solution regardless of initialization.

### K.2 Inner Loop Sensitivity to Initialization

A fundamental question regarding the proposed algorithm is whether the inner MM solver—which addresses a non-convex max-min quadratic subproblem—is sensitive to its starting point. To rigorously evaluate this, we examine the convergence behavior under diverse initialization conditions.

Figures S5 and S6 provide a direct empirical assessment of this issue for the Yale and Zernike datasets, respectively. For each dataset, the inner MM solver was executed from four distinct, randomly generated initial projection matrices. The results are compelling: despite the wide disparity in starting points, every trial converges to the same optimal objective value.

These results demonstrate that the inner solver is robust against local optima and consistently identifies the same solution regardless of the initial guess. The objective rises monotonically and smoothly throughout the iterations, consistent with the guaranteed ascent property of the MM framework, and stabilizes within 25 to 30 iterations for both datasets. Notably, the Yale dataset — which exhibits high intra-class variance — and the Zernike dataset — which consists of compact shape features — represent very different data regimes, yet the inner loop displays equally robust convergence behavior in both cases.

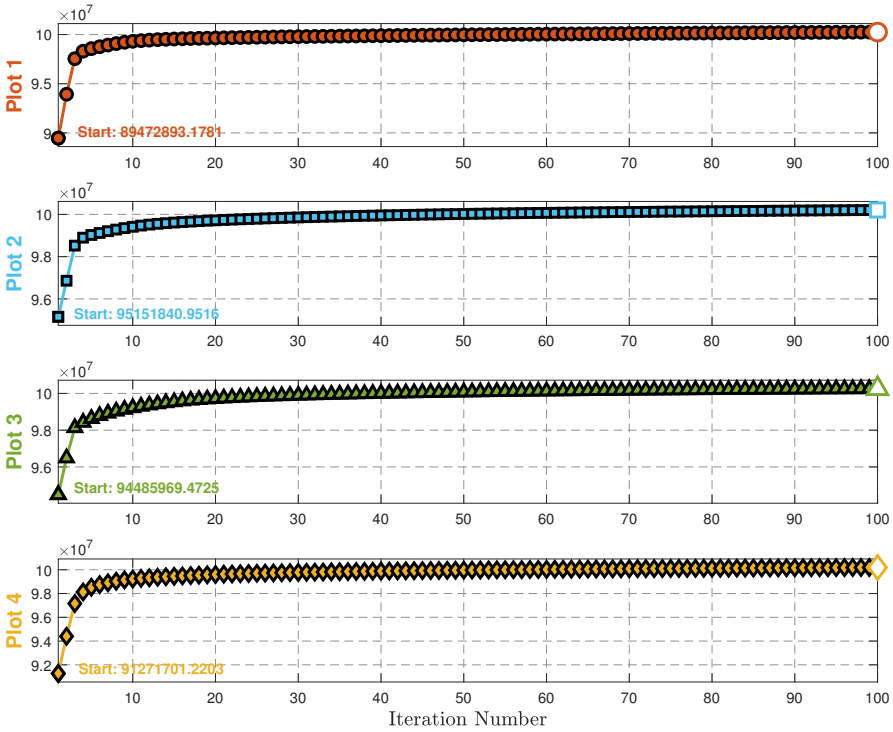


Figure S5: Inner MM convergence on Yale.

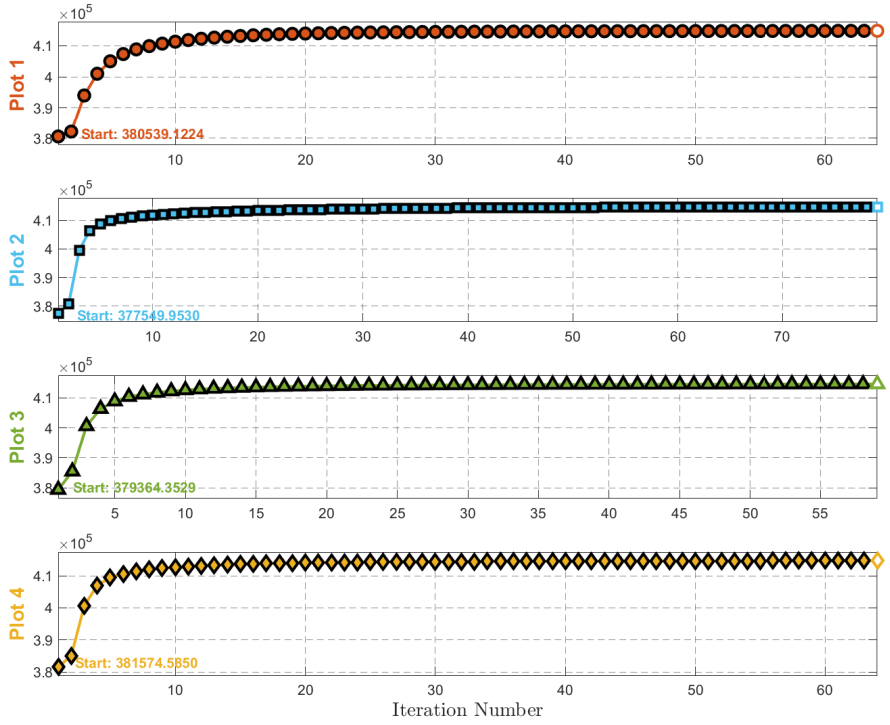


Figure S6: Inner MM convergence on Zernike.

## NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state our main contribution: the introduction of $F^3$-PWCRA (Fast, Free of relaxation, Free of hyperparameters for Pairwise Worst-Case Ratio Analysis), a novel discriminative dimensionality reduction method based on maximizing the minimum pairwise ratio of between-class to within-class scatter. The claims that $F^3$-PWCRA is computationally efficient, provably convergent, requires no semidefinite relaxations or hyperparameter tuning, and outperforms state-of-the-art methods in classification error are directly supported by our theoretical and empirical results.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We have included a dedicated "Discussion and Limitations" section preceding the conclusion section, which details the limitations of our proposed method. Particularly, the number of classes needs to be known a priori.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best

judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: We explain the theoretical components in detail, clearly stating all assumptions, lemmas, and theorems in the main text, and we provide their complete proofs in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We have included all the information necessary to reproduce the main experimental results. Details regarding the datasets (dimensions, samples, classes) are provided in the main text, with a more detailed description in the appendix. Additionally, within the numerical results section, we included a subsection dedicated to the experimental protocol to ensure other researchers can easily reproduce our work. Besides, all codes are already available as a supplementary file.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We have included the complete source code for our proposed method in the supplementary material, and the datasets are publicly available from the sources cited in the paper. To ensure that other researchers can faithfully reproduce the main experimental results, we have thoroughly documented the source code with explanatory comments detailing how to run it.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We detailed all training and test splits, as well as the preprocessing steps necessary to understand the results. Regarding hyperparameters, for the baseline methods, we used the exact hyperparameters specified in their respective papers or open-source repositories to guarantee a fair comparison.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: In this paper, we ran each experiment 100 times to ensure a highly reliable comparison (notably, competing papers typically evaluate over at most 20 runs). We report the classification error rates using mean and standard deviation across diverse datasets and two target dimensions. Furthermore, to explicitly demonstrate statistical significance, we performed t-tests comparing our proposed method against the baseline competitors.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: In the main text, we specify the computing resources used, stating that all algorithms were executed in MATLAB R2022b on a dual-socket Intel Xeon E5-2695 v3 system ($2\times14$ cores, 56 threads; 2.3 GHz base, 3.3 GHz turbo; 315 GB RAM). Moreover, in the appendix, we provide a comparison of execution times (runtime) to demonstrate the computational efficiency of our proposed approach.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: We have thoroughly reviewed the NeurIPS Code of Ethics and ensured that our research fully conforms to its guidelines. Additionally, all potential uncertainties were resolved by consulting the NeurIPS organizers prior to submission via the "Contact NeurIPS" form on the website.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [N/A]

Justification: We are currently unaware of the societal impacts of our work, if any.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: Used data in this paper were publicly available even before this submission. Besides, our contributions are not related to generative models.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.

- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We have properly credited the creators of all existing assets (datasets and baseline codes) used in this work by citing their original publications. Additionally, where applicable, we have provided the corresponding URLs for these assets.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We did not introduce any new datasets; all datasets used are from previously published papers. The only new asset introduced is our optimization technique. We have provided comprehensive comments throughout the source code to ensure it is well-documented. Currently, the code is not released publicly elsewhere but is included in the supplementary material attached to this submission.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

Justification: Our work does not include crowdsourcing or research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: No human subject is involved in this work.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [No]

    Justification: Large Language Models were only utilized for editing and refining the manuscript's text. They did not contribute to any core research methods, experiments, or original contributions in this work.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.