Exact Combinatorial Multi-Class Graph Cuts for Semi-Supervised Learning

Mohammad Mahdi Omati, Yasin Salajeghe, Mahshad Moradi, Arash Amini

Sharif University of Technology
Tehran, Iran
mohammad.omati@sharif.edu, yasin.salajegheh@gmail.com, mahshadmoradix@gmail.com, aamini@sharif.edu

Abstract

Semi-supervised learning (SSL) on graphs is critical in applications where labeled data are scarce and costly, yet existing graph-based methods often degrade under extreme label sparsity or class imbalance, yielding trivial or unstable solutions. We introduce **CombCut**, the first exact combinatorial optimization framework for multi-class graph-based semi-supervised learning that operates directly on binary one-hot assignments, without any convex relaxation or heuristic volume constraints. By employing a minorization–maximization (MM) scheme, CombCut transforms each step into a structured linear assignment problem solved efficiently via network-flow algorithms. Total unimodularity guarantees integral iterates, and our theoretical analysis establishes both monotonic ascent of the true discrete objective and convergence of every limit point to a Karush–Kuhn–Tucker (KKT) stationary solution of the original combinatorial problem. Our approach requires no hyperparameter tuning and scales near-linearly in the number of vertices. Empirical evaluation on MNIST, Fashion-MNIST, and CIFAR-10 with as few as 1–5 labels per class shows that CombCut excels in worst-case labeling scenarios, significantly outperforming state-of-the-art graph-SSL baselines and yielding more stable and accurate label propagation under severe supervision constraints.

Introduction

In many practical applications—ranging from image classification to regression—annotating data is prohibitively expensive, which has spurred interest in semi-supervised learning (SSL) (Calder et al. 2020; Jacobs, Merkurjev, and Esedoḡlu 2018; Nadler, Srebro, and Zhou 2009; El Alaoui et al. 2016; Zhou et al. 2003; Zhou, Huang, and Schölkopf 2005; Ando and Zhang 2006; Yang et al. 2006; Holtz et al. 2023; Holtz, Tang, and Peyré 2024). In SSL, one is given a small set of labeled examples together with a much larger collection of unlabeled examples; the goal is to leverage the geometry of the unlabeled data to learn a predictor that outperforms one trained on the labeled data alone. Graph-based methods realize this by treating each example as a vertex in a graph and defining a smoothness objective that encourages nearby vertices to share similar labels. A landmark technique in this family is Laplace learning, which finds the harmonic extension of the provided labels over the unlabeled vertices (Zhu, Ghahramani, and Lafferty 2003). Variants of this approach—including Poisson learning (Calder et al. 2020)—have been successfully applied across a range of semi-supervised and graph-structured learning problems (Zhou et al. 2003; Zhou, Huang, and Schölkopf 2005; Ando and Zhang 2006; Yang et al. 2006).

Recent studies (Nadler, Srebro, and Zhou 2009; El Alaoui et al. 2016) have shown that vanilla Laplace learning methods break down when only a handful of labels are available: the harmonic extension tends to concentrate its mass at labeled nodes—producing sharp spikes there—while leaving distant vertices almost unchanged, which leads to highly unreliable estimates near the decision boundary. In classification settings, one then applies a post-hoc threshold to convert the continuous solution into discrete class labels, a step that often amplifies this imbalance. When class distributions are imbalanced, practitioners often resort to heuristic volume-constraint mechanisms to enforce target class proportions (Calder et al. 2020; Jacobs, Merkurjev, and Esedoḡlu 2018). For example, Calder et al. (Calder et al. 2020) adopt an auction-based algorithm (Bertsekas 1988) that alternates between linearizing the quadratic energy and imposing both volume and box constraints. Despite these heuristic fixes, the core task—assigning binary one-hot labels under exact class-size constraints—is inherently a combinatorial optimization problem, prompting recent efforts to address it directly.

More recently, Holtz et al. (Holtz et al. 2023) have sought to address the discrete labeling challenge by framing graph-based SSL as a cardinality-constrained minimum-cut partitioning problem and then leveraging quadratic relaxations to render it tractable. Holtz et al. (Holtz et al. 2023) relax the original nonconvex binary and linear constraints by embedding labels on the Stiefel manifold and solve the resulting problem via a sequential subspace method (SSM). Although this relaxation yields a tractable formulation, it is not tight to the original combinatorial problem and can lead to suboptimal solutions.

To address these challenges, we propose *CombCut*, the first semi-supervised learning framework that preserves the binary one-hot label domain throughout optimization without any relaxation and requires no hyperparameter tuning. Our key insight starts by shifting the graph Laplacian with a sufficiently large diagonal matrix to obtain a positive-semidefinite alternative, resulting in a maximization problem with a convex objec-

tive in the assignment matrix; building on this, we employ a minorization–maximization (MM) scheme that linearizes the surrogate at each iteration and reduces the problem to a structured *linear assignment* subproblem—enforcing one-hot and class-size constraints—solved exactly via network-flow algorithms. Total unimodularity of the constraint system guarantees that every subproblem admits an integral solution, obviating any relaxation or rounding. Moreover, we show that *CombCut* guarantees monotonic ascent of the true combinatorial objective with convergence to Karush–Kuhn–Tucker (KKT) stationary points and delivers superior accuracy and stability under extreme label scarcity, outperforming all relaxation-based graph-SSL baselines.

Contributions

Below we summarize the key innovations of this work:

- We reframe multi-class graph-based semi-supervised learning (SSL) as a purely discrete cardinality-constrained min-cut problem with fixed labels, eliminating any need for continuous relaxations and directly operating within the binary label domain to preserve combinatorial integrity.
- We introduce *CombCut*, a minorization–maximization (MM)–based algorithm that ingeniously shifts the Laplacian spectrum to yield a convex surrogate for the maximization problem, transforms each iteration into an efficiently solvable linear-assignment subproblem via network-flow algorithms, and leverages total unimodularity to guarantee exact solutions without approximations.
- We establish rigorous theoretical foundations, proving monotonic ascent of a global surrogate objective, reliable convergence to a KKT stationary point, and inherent exact integrality of all iterates, thereby avoiding common drawbacks such as rounding errors and reliance on homotopy or heuristic adjustments.
- Through comprehensive experiments on k-NN graphs constructed from MNIST, Fashion-MNIST, and CIFAR-10, we demonstrate that CombCut consistently outperforms its competitors in classification accuracy across diverse label rates, showcasing its practical superiority in low-data regimes.

Notations In this paper, we use boldface letters for vectors and matrices: vectors are denoted by lowercase boldface (e.g., $\mathbf{a}$) and matrices by uppercase boldface (e.g., $\mathbf{A}$). The transpose operator is indicated by $(\cdot)^T$. The identity matrix is denoted by $\mathbf{I}$, while $\mathbf{0}$ represents an all-zero matrix or vector with sizes inferred from the context. The vector $\mathbf{1}_n$ denotes an n-dimensional vector of all ones. The Frobenius norm of a matrix $\mathbf{A}$ is denoted by $\|\mathbf{A}\|_F$. The number of nonzero entries of a matrix $\mathbf{A}$ is denoted by $\mathrm{nnz}(\mathbf{A})$. The vectorization of a matrix $\mathbf{A}$ is denoted by $\mathrm{vec}(\mathbf{A})$. The largest eigenvalue of the matrix $\mathbf{A}$ is denoted by $\lambda_{\max}(\mathbf{A})$. Finally, all notations are consistently applied throughout the derivations and analyses to ensure clarity and precision.

Problem Formulation

Let $G = (\mathcal{V}, \mathcal{E}, \mathbf{W})$ be an undirected weighted graph with vertex set $\mathcal{V} = \{v_1, \ldots, v_M\}$ and symmetric weight matrix $\mathbf{W} \in \mathbb{R}^{M\times M}$ whose entries $w_{ij} \geq 0$ encode the affinity between v_i and v_j. We define the degree of each vertex as $d_i = \sum_{j=1}^{M} w_{ij}$, and $\mathbf{D} = \mathrm{diag}(d_1, \ldots, d_M)$. Without loss of generality, we assume that the first m vertices $\ell = \{v_1, \ldots, v_m\}$ carry known one-hot labels $\{\mathbf{y}_1, \ldots, \mathbf{y}_m\} \subset \{\mathbf{e}_1, \ldots, \mathbf{e}_k\}$, where $\mathbf{e}_j$ is the j^{th} standard basis vector in $\mathbb{R}^k$ and $0 < m \ll M$. We denote the set of $n = M - m$ unlabeled vertices by $\mathcal{U} = \{v_{m+1}, \ldots, v_M\}$. Graph-based semi-supervised learning seeks to extend the labels on ℓ smoothly over $\mathcal{U}$. A classical method in this framework is Laplace learning (Zhu, Ghahramani, and Lafferty 2003), which finds a soft-label matrix $\mathbf{X}_0 \in \mathbb{R}^{M\times k}$ by solving

$$\min_{\mathbf{X}_0\in\mathbb{R}^{M\times k}} \ \mathrm{tr}\big(\mathbf{X}_0^T\,\mathbf{L}\,\mathbf{X}_0\big) \quad \text{s.t. } (\mathbf{X}_0)_i = \mathbf{y}_i,\ i = 1, \ldots, m, \tag{1}$$

where $\mathbf{L} = \mathbf{D} - \mathbf{W}$ is the combinatorial Laplacian. The predicted discrete label for each v_i is then recovered by

$$\hat{\mathbf{y}}_i = \arg\max_{1\leq j\leq k} (\mathbf{X}_0)_{ij}\,. \tag{2}$$

To isolate the role of the unlabeled vertices in the problem and derive a compact reduced formulation, we partition the Laplacian $\mathbf{L}$ and the soft-label matrix $\mathbf{X}_0$ into labeled and unlabeled blocks:

$$\mathbf{L} = \begin{pmatrix} \mathbf{L}_{\ell\ell} & \mathbf{L}_{\ell u} \\ \mathbf{L}_{u\ell} & \mathbf{L}_{uu} \end{pmatrix}, \qquad \mathbf{X}_0 = \begin{pmatrix} \mathbf{Y} \\ \mathbf{X} \end{pmatrix},$$

where $\mathbf{Y} \in \{0,1\}^{m\times k}$ collects the known one-hot labels and $\mathbf{X} \in \mathbb{R}^{n\times k}$ are the unknown soft labels. Expanding the quadratic form in (1) and dropping constants gives the reduced continuous objective

$$\min_{\mathbf{X}\in\{0,1\}^{n\times k}} \ \frac{1}{2}\,\mathrm{tr}\big(\mathbf{X}^T\,\mathbf{L}_{uu}\,\mathbf{X}\big) \ - \ \mathrm{tr}\big(\mathbf{X}^T\,\mathbf{B}\big), \tag{3}$$

where $\mathbf{B} = -\,\mathbf{L}_{u\ell}\,\mathbf{Y}$. We ensure integral one-hot assignments on $\mathcal{U}$ by requiring $\mathbf{X} \in \{0,1\}^{n\times k}$ and

$$\mathbf{X}\,\mathbf{1}_k = \mathbf{1}_n, \quad \mathbf{X}^T\,\mathbf{1}_n = \mathbf{m}, \quad \mathbf{1}_k^T\mathbf{m} = M,$$

where $\mathbf{m} \in \mathbb{R}^k$ specifies the desired number of examples in each class, and the last equality guarantees that the total number of assignments equals M, the total number of vertices. As a result, our discrete formulation becomes:

$$\begin{aligned} \min_{\mathbf{X}\in\{0,1\}^{n\times k}} \quad & F(\mathbf{X}) = \frac{1}{2}\,\mathrm{tr}\big(\mathbf{X}^T\,\mathbf{L}_{uu}\,\mathbf{X}\big) \ - \ \mathrm{tr}\big(\mathbf{X}^T\,\mathbf{B}\big) \\ \text{s.t.} \quad & \mathbf{X}\,\mathbf{1}_k = \mathbf{1}_n \\ & \mathbf{X}^T\,\mathbf{1}_n = \mathbf{m}. \end{aligned} \tag{4}$$

In the next section, we provide an overview of the MM framework and then explain how to apply it to solve (4).

MM framework: An Overview

Consider the constrained optimization problem

$$\max_{\mathbf{x}\in\chi} f(\mathbf{X}), \tag{5}$$

where $\mathbf{X}$ denotes the decision variable, $f(\mathbf{X})$ is the objective to be maximized, and χ represents the feasible region. An MM-based method tackles (5) by introducing, at each iteration t, a surrogate function $g\big(\mathbf{X} \mid \mathbf{X}^t\big)$, which underestimates $f(\mathbf{X})$ but matches it exactly at the current point $\mathbf{X}^t$. The next iterate is then found by solving $\mathbf{X}^{t+1} \in \arg\max_{\mathbf{X}\in\chi} g\big(\mathbf{X} \mid \mathbf{X}^t\big)$. These two operations—surrogate construction and maximization—are repeated until convergence to a stationary solution of (5).

For $g(\mathbf{X} \mid \mathbf{X}^t)$ to qualify as a valid minorizer, it must satisfy

$$g\big(\mathbf{X} \mid \mathbf{X}^t\big) \le f(\mathbf{X}) \quad \forall\, \mathbf{X} \in \chi, \tag{6}$$

$$g\big(\mathbf{X}^t \mid \mathbf{X}^t\big) = f\big(\mathbf{X}^t\big). \tag{7}$$

As a result of the surrogate properties, each MM step yields

$$f(\mathbf{X}^{t+1}) \;\ge\; g\big(\mathbf{X}^{t+1} \mid \mathbf{X}^t\big) \;\ge\; g\big(\mathbf{X}^t \mid \mathbf{X}^t\big) \;=\; f(\mathbf{X}^t),$$

which shows the objective value never decreases, ensuring the sequence $f(\mathbf{X}^t)$ converges to a KKT point of (5). To see a more detailed explanation of the MM framework, please refer to (Prabhu and Banerjee 2017).

Solving the Semi-Supervised Cut via MM

In order to apply MM framework to solve (4), it is necessary to have a concave form of the objective function. As a result, we introduce Lemma 0.1.

Lemma 0.1. *Let $\mathbf{X} \in \{0,1\}^{n\times k}$ satisfy $\mathbf{X}\,\mathbf{1}_k = \mathbf{1}_n$. Then*

$$\|\mathbf{X}\|_F^2 \;=\; n.$$

Proof. By definition $\|\mathbf{X}\|_F^2 = \sum_{i=1}^{n}\sum_{j=1}^{k} X_{ij}^2$. Since $X_{ij} \in \{0,1\}$, $X_{ij}^2 = X_{ij}$, hence $\|\mathbf{X}\|_F^2 = \sum_{i,j} X_{ij}$. But $\mathbf{X}\mathbf{1}_k = \mathbf{1}_n$ implies each row sums to one, so $\sum_{i,j} X_{ij} = n$. □

To prepare our problem for the MM framework, let us choose a scalar s satisfying $s \ge \lambda_{\max}(\mathbf{L}_{uu})$, and define $\mathbf{M} = s\,\mathbf{I}_n - \mathbf{L}_{uu}$. As a result, for any feasible $\mathbf{X}$,

$$\begin{aligned}&\frac{1}{2}\mathrm{tr}(\mathbf{X}^T\mathbf{L}_{uu}\,\mathbf{X}) - \mathrm{tr}(\mathbf{X}^T\mathbf{B})\\ &= -\big(\tfrac{1}{2}\mathrm{tr}(\mathbf{X}^T\mathbf{M}\,\mathbf{X}) + \mathrm{tr}(\mathbf{X}^T\mathbf{B})\big) \;+\; \tfrac{1}{2}\,s\,n,\end{aligned} \tag{8}$$

Here, the term $\frac{1}{2}sn$ is constant and can be dropped from the optimization problem. Hence (4) is equivalent to the convex-in-$\mathbf{X}$ maximization

$$\begin{aligned}\max_{\mathbf{X}\in\{0,1\}^{n\times k}} \quad & f(\mathbf{X}) = \frac{1}{2}\mathrm{tr}(\mathbf{X}^T\mathbf{M}\,\mathbf{X}) \;+\; \mathrm{tr}(\mathbf{X}^T\mathbf{B})\\ \text{s.t.} \quad & \mathbf{X}\,\mathbf{1}_k = \mathbf{1}_n\\ & \mathbf{X}^T\mathbf{1}_n = \mathbf{m}.\end{aligned} \tag{9}$$

Since $\mathbf{M} \succeq 0$, $f(.)$ is convex in $\mathbf{X}$; thus, at any current iterate $\mathbf{X}^t$, it can be minorized by its tangent hyperplane:

$$g\big(\mathbf{X} \mid \mathbf{X}^{(t)}\big) = \mathrm{tr}\big((\mathbf{M}\,\mathbf{X}^{(t)} + \mathbf{B})^T\mathbf{X}\big) + \text{const},$$

which by convexity satisfies $g(\mathbf{X} \mid \mathbf{X}^{(t)}) \le f(\mathbf{X})$ for all $\mathbf{X}$ and touches equality at $\mathbf{X} = \mathbf{X}^{(t)}$. Dropping the constant yields the surrogate

$$\begin{aligned}\max_{\mathbf{X}\in\{0,1\}^{n\times k}} \quad & \mathrm{tr}\big((\mathbf{C}^{(t)})^T\mathbf{X}\big)\\ \text{s.t.} \quad & \mathbf{X}\,\mathbf{1}_k = \mathbf{1}_n\\ & \mathbf{X}^T\mathbf{1}_n = \mathbf{m},\end{aligned} \tag{10}$$

where $\mathbf{C}^{(t)} = \mathbf{M}\,\mathbf{X}^{(t)} + \mathbf{B}$.

In (10), all constraints on $\mathbf{X}$, except for the integrality requirement $X_{ij} \in \{0,1\}$, are affine. Relaxing this integrality to the box constraints $0 \le x_{ij} \le 1$ yields a convex feasible set that could, in principle, admit fractional optima. The following definition and theorem—from (Schrijver 1998) and (Wolsey and Nemhauser 1999)—ensure that every extreme point of this polytope is integral, and hence any optimal $\mathbf{X}$ remains binary.

Definition 0.1. A matrix $\mathbf{D}$ is *totally unimodular* (TU) if every square submatrix has determinant in $\{-1, 0, 1\}$.

Theorem 0.2. *If $\mathbf{D}$ is TU and $\mathbf{b}$ is integral, then the polyhedron $\{\mathbf{x} : \mathbf{D}\,\mathbf{x} \le \mathbf{b},\ 0 \le \mathbf{x} \le 1\}$ has only integral vertices, so any LP over this system admits an integral optimum.*

To invoke Theorem 0.2, we vectorize the assignment matrix $\mathbf{X} \in \mathbb{R}^{n\times k}$ into

$$\mathbf{x} \;=\; \mathrm{vec}(\mathbf{X}) \;\in\; \mathbb{R}^{n\,k}.$$

In this form, the affine constraints in (10) become linear inequalities of the form $\mathbf{D}\,\mathbf{x} \le \mathbf{b}$.

First, the row-sum constraint $\mathbf{X}\,\mathbf{1}_k = \mathbf{1}_n$ is equivalent to

$$(\mathbf{I}_n \otimes \mathbf{1}_k^T)\,\mathbf{x} \;=\; \mathbf{1}_n,$$

which we rewrite as the pair

$$(\mathbf{I}_n \otimes \mathbf{1}_k^T)\,\mathbf{x} \le \mathbf{1}_n, \quad -(\mathbf{I}_n \otimes \mathbf{1}_k^T)\,\mathbf{x} \le -\mathbf{1}_n.$$

Next, the column-sum constraint $\mathbf{X}^T\mathbf{1}_n = \mathbf{m}$ can be written in vectorized form as

$$(\mathbf{1}_n^T \otimes \mathbf{I}_k)\,\mathbf{x} = \mathbf{m},$$

which is equivalently enforced by the pair of inequalities

$$(\mathbf{1}_n^T \otimes \mathbf{I}_k)\,\mathbf{x} \le \mathbf{m}, \quad -(\mathbf{1}_n^T \otimes \mathbf{I}_k)\,\mathbf{x} \le -\mathbf{m}.$$

Stacking these into a single matrix yields

$$\mathbf{D} \;=\; \begin{bmatrix} \mathbf{I}_n \otimes \mathbf{1}_k^\top \\ -(\mathbf{I}_n \otimes \mathbf{1}_k^\top) \\ \mathbf{1}_n^\top \otimes \mathbf{I}_k \\ -(\mathbf{1}_n^\top \otimes \mathbf{I}_k) \end{bmatrix} \;\in\; \{0,\pm 1\}^{(2n+2k)\times(nk)}, \quad \mathbf{b} = \begin{bmatrix} \mathbf{1}_n \\ -\mathbf{1}_n \\ \mathbf{m} \\ -\mathbf{m} \end{bmatrix}.$$

Accordingly, the linear programming relaxation of (10) can be expressed as

$$\begin{aligned}\max_{\mathbf{x}\in\mathbb{R}^{nk}} \quad & (\mathrm{vec}(\mathbf{C}^{(t)}))^T\mathbf{x}\\ \text{s.t.} \quad & \mathbf{D}\,\mathbf{x} \le \mathbf{b}\\ & 0 \le \mathbf{x} \le 1.\end{aligned} \tag{11}$$

Algorithm 1: CombCut algorithm

Require: Unlabeled Laplacian $\mathbf{L}_{uu}$, supervision term $\mathbf{B}$, class sizes $\mathbf{m}$, shift $s \geq \lambda_{\max}(\mathbf{L}_{uu})$, step-sizes $\{\eta^{(t)}\}$, tolerance ϵ
Ensure: One-hot labeling $\mathbf{X} \in \{0,1\}^{n\times k}$

Compute PSD shift: $\mathbf{M} \leftarrow s\,\mathbf{I}_n - \mathbf{L}_{uu} \succeq 0$
Initialize $\mathbf{X}^{(0)} \in \{0,1\}^{n\times k}$ s.t. $\mathbf{X}^{(0)}\mathbf{1}_k = \mathbf{1}_n$, $(\mathbf{X}^{(0)})^T\mathbf{1}_n = \mathbf{m}$
for $t = 0, 1, 2, \ldots$ **do**
  $\mathbf{C}^{(t)} \leftarrow \mathbf{M}\,\mathbf{X}^{(t)} + \mathbf{B}$
  Update X: Solve the linear program:

$$\mathbf{X}^{(t+1)} \leftarrow \arg\max_{\mathbf{X}} \quad \mathrm{tr}\big((\mathbf{C}^{(t)})^T\mathbf{X}\big)$$
$$\text{s.t.} \quad \mathbf{X} \in [0,1]^{n\times k},\ \mathbf{X}\,\mathbf{1}_k = \mathbf{1}_n,\ \mathbf{X}^T\mathbf{1}_n = \mathbf{m}.$$

  Check convergence: $\dfrac{\big|F(\mathbf{X}^{(t+1)}) - F(\mathbf{X}^{(t)})\big|}{\big|F(\mathbf{X}^{(t)})\big|} < \epsilon$
  if true **then**
    break
  end if
end for
return $\mathbf{X}^{(t+1)}$

Because $\mathbf{I}_n \otimes \mathbf{1}_k^T$ and $\mathbf{1}_n^T \otimes \mathbf{I}_k$ are both totally unimodular—and this property is preserved under negation and row-stacking (Schrijver 1998)—the assembled matrix $\mathbf{D}$ is TU. Moreover, the entries of $\mathbf{b}$ are clearly integral. As a result, Theorem 0.2, every vertex of the polyhedron

$$\{\mathbf{x} : \mathbf{D}\mathbf{x} \leq \mathbf{b},\ 0 \leq \mathbf{x} \leq 1\}$$

is integral, ensuring that any optimal solution $\mathbf{x}^*$ lies in $\{0,1\}^{nk}$. In other words, the LP relaxation of (10) always admits an integral maximizer.

Consequently, the relaxed surrogate

$$\begin{aligned} \max_{\mathbf{X}\in[0,1]^{n\times k}} \quad & \mathrm{tr}\big((\mathbf{C}^{(t)})^T\mathbf{X}\big) \\ \text{s.t.} \quad & \mathbf{X}\,\mathbf{1}_k = \mathbf{1}_n \\ & \mathbf{X}^T\mathbf{1}_n = \mathbf{m}, \end{aligned} \tag{12}$$

is always tight. Problem (12) admits direct solutions via LP solvers like CVX (Grant and Boyd 2014) or CVXPY (Diamond and Boyd 2016), and can also be tackled through Lagrangian duality approaches (Prabhu and Banerjee 2017; Saini et al. 2024). A succinct outline is presented in Algorithm 1.

Computational Complexity and Convergence

Computational Complexity

We now analyze the computational complexity of the proposed *CombCut* algorithm. At iteration t, the main operations are as follows:

- **Computation of $\mathbf{C}^{(t)}$:** The update $\mathbf{C}^{(t)} = \mathbf{M}\,\mathbf{X}^{(t)} + \mathbf{B}$ requires multiplying the $n \times n$ sparse matrix $\mathbf{M}$ with the $n \times k$ label matrix $\mathbf{X}^{(t)}$. Since $\mathbf{M}$ inherits the sparsity pattern of $\mathbf{L}_{uu}$ from the k-nearest neighbor graph with $O(kn)$ nonzero entries, the multiplication cost is

$$O(\mathrm{nnz}(\mathbf{M}) \cdot k) \;=\; O(k^2 n),$$

  which is linear in n for small, fixed k.
- **Projection onto the constraint set $\mathcal{C}$:** The Euclidean projection onto the intersection of row- and column-sum constraints is carried out via alternating normalization steps with sorting. This requires $O(nk)$ operations per sweep, and a small constant number of sweeps is sufficient in practice, giving an overall projection cost of $O(nk)$.
- **Objective evaluation:** The computation of $F(\mathbf{X}^{(t)})$ involves the quadratic form $\mathbf{X}^T\mathbf{M}\mathbf{X}$ and a trace term, both evaluated in $O(k^2 n)$ time using sparse $\mathbf{M}$.

Therefore, the per-iteration cost is

$$O(k^2 n) + O(nk) = O(k^2 n),$$

dominated by the sparse matrix–label multiplication. Given that k is typically small (e.g., $k = 10$), the method scales linearly with the number of unlabeled vertices n.

In terms of memory, $\mathbf{M}$ and $\mathbf{W}$ are stored in sparse format, requiring $O(kn)$ space, while $\mathbf{X}$ and $\mathbf{C}$ require $O(nk)$ storage. This yields a total memory footprint of $O(kn)$, making *CombCut* memory-efficient for large-scale graphs.

Since each MM iteration monotonically increases the objective, convergence is reached in a modest number of iterations T (typically tens in our experiments), leading to an overall complexity $O(T\,k^2 n)$, which is near-linear in n and competitive with, or faster than, existing graph-based SSL methods that require solving large linear systems or semidefinite programs.

Convergence Analysis

In this section, we prove that the MM iterates generated by Algorithm 1 produce a non-decreasing objective sequence that converges, and that every limit point of the iterates satisfies the first-order (KKT) stationarity condition. To make it clear what is a stationary point in our case, we first introduce a first-order optimality condition for minimizing a smooth function over an arbitrary constraint set, which follows from (Bertsekas, Nedic, and Ozdaglar 2003).

First-Order Optimality for Maximization

Proposition 1. *Let $f : \mathbb{R}^{n\times k} \to \mathbb{R}$ be continuously differentiable, and let $\mathbf{X}^*$ be a local maximizer of f over a closed set $\mathcal{C} \subseteq \mathbb{R}^{n\times k}$. Then*

$$\mathrm{tr}\big(\nabla_{\mathbf{X}} f(\mathbf{X}^*)^T\,(\mathbf{Z} - \mathbf{X}^*)\big) \;\leq\; 0, \quad \forall\,\mathbf{Z} \in T_{\mathcal{C}}(\mathbf{X}^*),$$

where $T_{\mathcal{C}}(\mathbf{X}^*)$ denotes the tangent cone of $\mathcal{C}$ at $\mathbf{X}^*$.

Monotonicity and Stationarity: Recall the original maximization problem (1):

$$\max_{\mathbf{X}\in\mathcal{C}} f(\mathbf{X}) \;=\; \tfrac{1}{2}\mathrm{tr}\big(\mathbf{X}^T\mathbf{M}\mathbf{X}\big) + \mathrm{tr}\big(\mathbf{X}^T\mathbf{B}\big),$$

and define at iterate $\mathbf{X}^{(t)}$ the MM surrogate

$$g\big(\mathbf{X} \mid \mathbf{X}^{(t)}\big) = \mathrm{tr}\big((\mathbf{M}\,\mathbf{X}^{(t)} + \mathbf{B})^T\mathbf{X}\big) + \mathrm{const.}$$

Since $\mathbf{M} \succeq 0$, F is convex in $\mathbf{X}$. Therefore, for all $\mathbf{X} \in \mathcal{C}$,

$$g\big(\mathbf{X} \mid \mathbf{X}^{(t)}\big) \;\leq\; f(\mathbf{X}),$$

with equality at $\mathbf{X} = \mathbf{X}^{(t)}$.

The MM update chooses

$$\mathbf{X}^{(t+1)} = \arg\max_{\mathbf{X}\in\mathcal{C}} g\big(\mathbf{X} \mid \mathbf{X}^{(t)}\big).$$

Hence

$$f\big(\mathbf{X}^{(t+1)}\big) \;\geq\; g\big(\mathbf{X}^{(t+1)} \mid \mathbf{X}^{(t)}\big) \geq g\big(\mathbf{X}^{(t)} \mid \mathbf{X}^{(t)}\big) \;=\; f\big(\mathbf{X}^{(t)}\big). \tag{13}$$

This shows the sequence $\{f(\mathbf{X}^{(t)})\}$ is non-decreasing. Since $\mathcal{C}$ is finite, $f(.)$ is bounded above, and thus $f(\mathbf{X}^{(t)}) \to f^* < \infty$.

Because $\mathcal{C}$ is compact, the iterates $\{\mathbf{X}^{(t)}\}$ admit at least one limit point. Let $\mathbf{X}^{(\infty)}$ be such a point, and extract a convergent subsequence $\{\mathbf{X}^{(t_j)}\}$ with

$$\mathbf{X}^{(t_j)} \;\to\; \mathbf{X}^{(\infty)}.$$

By definition of the update,

$$\mathbf{X}^{(t_j+1)} = \arg\max_{\mathbf{X}\in\mathcal{C}} g\big(\mathbf{X} \mid \mathbf{X}^{(t_j)}\big),$$

so for any $\mathbf{Z} \in \mathcal{C}$,

$$g\big(\mathbf{X}^{(t_j+1)} \mid \mathbf{X}^{(t_j)}\big) \;\geq\; g\big(\mathbf{Z} \mid \mathbf{X}^{(t_j)}\big). \tag{14}$$

Letting $j \to \infty$ and using continuity of $g(.)$ yields

$$g\big(\mathbf{X}^{(\infty)} \mid \mathbf{X}^{(\infty)}\big) \;\geq\; g\big(\mathbf{Z} \mid \mathbf{X}^{(\infty)}\big), \quad \forall\, \mathbf{Z} \in \mathcal{C}.$$

Thus $\mathbf{X}^{(\infty)}$ globally maximizes the linear surrogate $g(\cdot \mid \mathbf{X}^{(\infty)})$ over $\mathcal{C}$. By Proposition 1, the first-order condition for this maximization is

$$\operatorname{tr}\big(\nabla_{\mathbf{X}}\, g\big(\mathbf{X} \mid \mathbf{X}^{(\infty)}\big)\big|_{\mathbf{X}=\mathbf{X}^{(\infty)}}(\mathbf{Z}-\mathbf{X}^{(\infty)})\big) \;\leq\; 0, \quad \forall\, \mathbf{Z} \in \mathcal{C}.$$

Noting

$$\nabla_{\mathbf{X}}\, g\big(\mathbf{X} \mid \mathbf{X}^{(\infty)}\big)\big|_{\mathbf{X}=\mathbf{X}^{(\infty)}} = \mathbf{M}\,\mathbf{X}^{(\infty)} + \mathbf{B}, \tag{15}$$

we conclude

$$\operatorname{tr}\big((\mathbf{M}\,\mathbf{X}^{(\infty)} + \mathbf{B})^T(\mathbf{Z} - \mathbf{X}^{(\infty)})\big) \;\leq\; 0, \quad \forall\, \mathbf{Z} \in \mathcal{C},$$

which is exactly the KKT stationarity condition for f at $\mathbf{X}^{(\infty)}$. This completes the proof.

Initialization Strategy

Due to the non-convex nature of the problem, the selection of an initial value is critical for obtaining a high-quality solution. For our initialization, we employ the approach proposed in (Holtz, Tang, and Peyré 2024), setting the perturbation parameter to $s = 0.2$. This choice is motivated by the observation in their work that for $0 \leq s \leq 1$, the constrained quadratic minimizer encourages solutions where predictions at unlabeled, high-degree vertices have a small norm. Although Holtz et al. (Holtz, Tang, and Peyré 2024) require $s > 1$ to theoretically guarantee a discrete solution, we found that adhering to this condition can lead to extreme solutions that are not necessarily optimal or of high quality.

Numerical Results

Dataset

In this paper, we conduct experiments on three widely recognized image datasets—MNIST (Deng 2012), Fashion-MNIST (Xiao, Rasul, and Vollgraf 2017), and CIFAR-10 (Krizhevsky, Hinton et al. 2009)—chosen to evaluate the proposed method across varying levels of complexity and to compare its performance against a diverse set of baselines. For MNIST and Fashion-MNIST, variational autoencoders (VAEs) with 3 fully connected layers of sizes (784, 400, 20) and (784, 400, 30), respectively, were trained for 100 epochs on each dataset using a symmetrically defined decoder architecture, loss, and training procedure similar to (Calder et al. 2020). These VAEs served as feature extractors, and graphs over the latent feature space were constructed with $n = 70{,}000$ nodes for both datasets, utilizing all available data. For CIFAR-10, SimCLR (Simple Framework for Contrastive Learning of Visual Representations) (Chen et al. 2020) was employed to pretrain representations on the full dataset of $n = 60{,}000$ nodes, leveraging self-supervised contrastive learning with data augmentations such as random cropping and color distortion.

The graphs for all datasets were constructed as k-nearest neighbor graphs with Gaussian edge weights, calculated as

$$w_{ij} = \exp\big(-4\|v_i - v_j\|^2/d_k(v_i)^2\big),$$

where v_i and v_j are the latent variables for images i and j, and $d_k(v_i)$ is the distance in the latent space between v_i and its k-th nearest neighbor. We set $k = 10$ consistently across all experiments, and symmetrized the weight matrix $\mathbf{W}$ by replacing it with $\frac{1}{2}(\mathbf{W} + \mathbf{W}^T)$.

Evaluations focused on low-label regimes (1–5 labels per class), a critical scenario for SSL where traditional methods often degenerate. We report average top-1 accuracy (%) over 20 independent trials, with standard deviations indicating variability due to random label selection. All experiments are conducted using MATLAB R2022b on a dual-socket Intel Xeon E5-2695 v3 system (2 × 14 physical cores, 56 threads total, 2.3 GHz base frequency, up to 3.3 GHz turbo boost, 70 MiB L3 cache) with 256 GB of RAM.

Compared Methods

To evaluate our proposed combCut method, we compare it against several graph-based SSL approaches. Below are brief introductions to each baseline:

1. **Laplace Learning** (Zhu, Ghahramani, and Lafferty 2003): Propagates labels via graph Laplacian smoothness, effective but prone to over-smoothing with few labels.
2. **Poisson Learning** (Calder et al. 2020): Enhances Laplace learning with class size priors for better balance in low-label settings.
3. **p-Laplace** (Flores, Calder, and Lerman 2022): Uses a nonlinear p-Laplacian (e.g., p=3) for robust label propagation against noise.
4. **AMLE** (El Alaoui et al. 2016): Extends labels with minimal Lipschitz constant, approximating infinity-Laplacian behavior.

5. **Volume-MBO** (Jacobs, Merkurjev, and Esedoḡlu 2018): Applies threshold dynamics with volume constraints for balanced multi-class partitions.
6. **SSM** (Holtz et al. 2023): Optimizes SSL on Stiefel manifolds to address low-label degeneracy.

Results

Performance on MNIST: As shown in Table 1(a), CombCutSSL achieves superior accuracy across all label rates, starting at 95.94% ± 2.38 for 1 label per class and rising to 97.36% ± 0.10 at 5 labels—marking improvements of 3.07% and 1.07% over the next-best baseline, Poisson Learning (92.87% ± 3.73 and 96.29% ± 0.84, respectively). This substantial lead highlights CombCutSSL's strength in leveraging exact integer partitions through its minorization-maximization scheme, effectively countering the degeneracy seen in Laplace Learning, which starts at a mere 18.93% ± 7.84 at 1 label due to its reliance on harmonic extensions that fail to propagate labels effectively with sparse supervision. Poisson Learning and Volume MBO, with accuracies of 92.87% ± 3.73 and 87.19% ± 4.13 at 1 label, benefit from volume constraints that enforce class balance, yet their suboptimal relaxations limit further gains compared to CombCutSSL's discrete approach. p-Laplace (72.70% ± 6.29) and AMLE (64.69% ± 4.01) offer moderate improvements through nonlinearity, but their higher variance at low labels suggests sensitivity to initial label placement. StiefelSSL (62.36% ± 5.37) underperforms, likely due to its manifold-based relaxation not fully capturing cluster boundaries. The upward accuracy trend with increasing labels is expected, reflecting the growing availability of supervisory signals. CombCutSSL's standard deviation dropping from 2.38 to 0.10 underscores its enhanced stability and convergence to near-optimal Karush-Kuhn-Tucker (KKT) points, suggesting a robust optimization process that adapts well as more labels are introduced.

Performance on FashionMNIST: Results in Table 1(b) on the more challenging FashionMNIST dataset show Poisson Learning leading at 1 label per class with 60.16% ± 6.02 compared to CombCutSSL's 57.38% ± 5.98, a gap of 2.78%, while CombCutSSL surpasses Poisson Learning after 3 labels per class, achieving 72.05% ± 1.98 at 5 labels against Poisson Learning's 71.92% ± 2.15, a 0.13% improvement. The dataset's inherent ambiguity, due to visually similar apparel items, exacerbates baseline degradation, as seen with Laplace Learning dropping to 17.60% ± 5.91 at 1 label, reflecting its struggle with sparse labels. Volume MBO (54.10% ± 6.00) and p-Laplace (53.77% ± 5.11) provide moderate lifts, leveraging threshold dynamics and nonlinearity, respectively, yet fall short of CombCutSSL's discrete cut formulation, which better delineates cluster boundaries (e.g., distinguishing between shirts and coats). StiefelSSL (46.20% ± 3.57) lags, possibly due to its manifold constraints not fully aligning with the data's geometry.

Performance on CIFAR-10: On the CIFAR-10 dataset, characterized by complex visual features, CombCutSSL again demonstrates superior performance, achieving 75.72% ± 5.69 at 1 label per class, outperforming Poisson Learning's 71.86% ± 7.52 by 3.86%, and reaching 81.60% ± 2.76 at 5 labels, a 1.59% improvement over Poisson's 80.01% ± 2.31, as shown in Table 1(c). The lower baseline accuracies (e.g., AMLE at 60.79% ± 5.89, StiefelSSL at 55.23% ± 6.53) highlight the difficulty of this dataset, where Volume MBO (68.88% ± 7.69) gains from threshold dynamics but incurs higher variance, yet CombCutSSL's consistent gains and stabilizing deviations (from 5.69 to 2.76) indicate its adaptability to richer feature spaces.

Conclusion

In this paper, we introduced *CombCut*, a novel framework for graph-based semi-supervised learning that directly optimized over binary one-hot label assignments, eliminating the need for hyperparameter tuning. We shifted the graph Laplacian with a diagonal matrix to create a convex, positive-semidefinite surrogate objective and employed a minorization-maximization scheme that reduced each iteration to a linear assignment subproblem, solved exactly via network-flow algorithms. The total unimodularity of the constraint system ensured integral solutions, preserving exact class sizes without approximation. Moreover, we proved theoretical guarantees of monotonic ascent of the combinatorial objective and convergence to Karush-Kuhn-Tucker stationary points. Our empirical evaluations on MNIST, FashionMNIST, and CIFAR-10 showed that CombCut delivered superior accuracy and stability, particularly in low-label regimes, even under class imbalance. This work advanced semi-supervised learning by integrating combinatorial optimization with efficient convex techniques, providing a robust solution for label-scarce scenarios across image classification and broader applications.

Acknowledgments. This work is supported by INSF, Iran National Science Foundation, grant number 4032041.

References

Ando, R.; and Zhang, T. 2006. Learning on graph with Laplacian regularization. *Advances in neural information processing systems*, 19.

Bertsekas, D.; Nedic, A.; and Ozdaglar, A. 2003. *Convex analysis and optimization*, volume 1. Athena Scientific.

Bertsekas, D. P. 1988. The auction algorithm: A distributed relaxation method for the assignment problem. *Annals of operations research*, 14(1): 105–123.

Calder, J.; Cook, B.; Thorpe, M.; and Slepcev, D. 2020. Poisson learning: Graph based semi-supervised learning at very low label rates. In *International conference on machine learning*, 1306–1316. PMLR.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, 1597–1607. PmLR.

Deng, L. 2012. The MNIST Database of Handwritten Digit Images for Machine Learning Research [Best of the Web]. *IEEE Signal Processing Magazine*, 29(6): 141–142.

Table 1: Accuracy (%) for 1–5 labels per class on MNIST-VAE, FashionMNIST-VAE, and CIFAR-SimCLR.

(a) MNIST-VAE

# Labels per class	1	2	3	4	5
Laplace Learning	18.93 ± 7.84	32.39 ± 14.58	47.19 ± 17.68	63.21 ± 10.62	75.19 ± 7.56
Poisson Learning	92.87 ± 3.73	95.61 ± 1.02	95.72 ± 0.84	96.20 ± 0.41	96.29 ± 0.84
p-Laplace (p=3.00)	72.70 ± 6.29	86.02 ± 3.62	88.90 ± 1.71	91.25 ± 1.15	92.54 ± 1.05
AMLE	64.69 ± 4.01	75.53 ± 4.32	78.79 ± 2.82	82.34 ± 1.78	84.43 ± 1.74
Volume MBO (T=0.10, V=0.50)	87.19 ± 4.13	93.61 ± 2.92	93.78 ± 1.89	95.23 ± 1.48	96.12 ± 1.18
StiefelSSL	62.36 ± 5.37	73.05 ± 4.27	76.32 ± 3.03	79.39 ± 1.99	81.54 ± 1.77
CombCutSSL (Exact Method)	$\mathbf{95.94 \pm 2.38}$	$\mathbf{97.14 \pm 0.19}$	$\mathbf{97.26 \pm 0.11}$	$\mathbf{97.33 \pm 0.08}$	$\mathbf{97.36 \pm 0.10}$

(b) FashionMNIST-VAE

# Labels per class	1	2	3	4	5
Laplace Learning	17.60 ± 5.91	28.48 ± 9.17	45.25 ± 11.04	52.93 ± 8.24	56.82 ± 6.37
Poisson Learning	$\mathbf{60.16 \pm 6.02}$	$\mathbf{66.87 \pm 3.73}$	68.69 ± 2.82	70.84 ± 1.44	71.92 ± 2.15
p-Laplace (p=3.00)	53.77 ± 5.11	60.96 ± 4.07	65.07 ± 3.08	67.85 ± 2.09	69.45 ± 2.11
AMLE	48.52 ± 4.30	55.70 ± 3.67	59.65 ± 3.04	62.24 ± 1.93	63.64 ± 2.38
Volume MBO (T=0.10, V=0.50)	54.10 ± 6.00	61.89 ± 5.04	66.48 ± 2.60	68.45 ± 2.69	70.67 ± 2.29
StiefelSSL	46.20 ± 3.57	52.61 ± 3.12	56.72 ± 3.39	59.21 ± 2.11	60.53 ± 2.44
CombCutSSL (Exact Method)	57.38 ± 5.98	66.34 ± 3.93	$\mathbf{69.03 \pm 3.14}$	$\mathbf{71.12 \pm 2.07}$	$\mathbf{72.05 \pm 1.98}$

(c) CIFAR-SimCLR

# Labels per class	1	2	3	4	5
Laplace Learning	13.95 ± 6.61	24.36 ± 9.77	40.01 ± 9.83	48.98 ± 8.09	58.77 ± 8.48
Poisson Learning	71.86 ± 7.52	76.08 ± 3.62	80.08 ± 2.10	80.12 ± 1.65	80.01 ± 2.31
p-Laplace (p=3.00)	63.84 ± 7.36	71.63 ± 3.45	76.57 ± 1.82	76.69 ± 1.55	77.26 ± 2.40
AMLE	60.79 ± 5.89	66.09 ± 3.56	70.78 ± 2.92	71.15 ± 1.93	71.86 ± 2.73
Volume MBO (T=0.10, V=0.50)	68.88 ± 7.69	74.43 ± 4.94	77.91 ± 2.49	77.98 ± 1.99	79.17 ± 2.08
StiefelSSL	55.23 ± 6.53	62.77 ± 3.45	67.22 ± 3.08	68.15 ± 2.07	68.95 ± 2.99
CombCutSSL (Exact Method)	$\mathbf{75.72 \pm 5.69}$	$\mathbf{78.77 \pm 3.77}$	$\mathbf{81.99 \pm 1.73}$	$\mathbf{81.60 \pm 1.88}$	$\mathbf{81.60 \pm 2.76}$

Diamond, S.; and Boyd, S. 2016. CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83): 1–5.

El Alaoui, A.; Cheng, X.; Ramdas, A.; Wainwright, M. J.; and Jordan, M. I. 2016. Asymptotic behavior ofℓ_p-based Laplacian regularization in semi-supervised learning. In *Conference on Learning Theory*, 879–906. PMLR.

Flores, M.; Calder, J.; and Lerman, G. 2022. Analysis and algorithms for ℓ_p-based semi-supervised learning on graphs. *Applied and Computational Harmonic Analysis*, 60: 77–122.

Grant, M.; and Boyd, S. 2014. CVX: Matlab Software for Disciplined Convex Programming, version 2.1.

Holtz, C.; Chen, P.; Cloninger, A.; Cheng, C.-K.; and Mishne, G. 2023. Semi-Supervised Laplace Learning on Stiefel Manifolds. *arXiv preprint arXiv:2308.00142*.

Holtz, C.; Tang, R.; and Peyré, G. 2024. Continuous Partitioning for Graph-Based SSL. In *Adv. in Neural Information Processing Systems 37*.

Jacobs, M.; Merkurjev, E.; and Esedoḡlu, S. 2018. Auction dynamics: A volume constrained MBO scheme. *Journal of Computational Physics*, 354: 288–310.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.

Nadler, B.; Srebro, N.; and Zhou, X. 2009. Semi-supervised learning with the graph Laplacian: The limit of infinite unlabelled data. *Advances in neural information processing systems*, 22: 1330–1338.

Prabhu, V.; and Banerjee, A. 2017. An MM-Based Approach to Label Propagation. In *Proc. of the 30th AAAI Conf. on Artificial Intelligence*, 2839–2847.

Saini, A.; Stoica, P.; Babu, P.; Arora, A.; et al. 2024. Min-Max Framework for Majorization-Minimization Algorithms in Signal Processing Applications: An Overview. *Foundations and Trends® in Signal Processing*, 18(4): 310–389.

Schrijver, A. 1998. *Theory of linear and integer programming*. John Wiley & Sons.

Wolsey, L. A.; and Nemhauser, G. L. 1999. *Integer and combinatorial optimization*. John Wiley & Sons.

Xiao, H.; Rasul, K.; and Vollgraf, R. 2017. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*.

Yang, X.; Fu, H.; Zha, H.; and Barlow, J. 2006. Semi-supervised nonlinear dimensionality reduction. In *Proceed-*

ings of the 23rd international conference on Machine learning, 1065–1072.

Zhou, D.; Bousquet, O.; Lal, T.; Weston, J.; and Schölkopf, B. 2003. Learning with local and global consistency. *Advances in neural information processing systems*, 16.

Zhou, D.; Huang, J.; and Schölkopf, B. 2005. Learning from labeled and unlabeled data on a directed graph. In *Proceedings of the 22nd international conference on Machine learning*, 1036–1043.

Zhu, X.; Ghahramani, Z.; and Lafferty, J. D. 2003. Semi-supervised learning using Gaussian fields and harmonic functions. In *Proceedings of the 20th International conference on Machine learning (ICML-03)*, 912–919.