Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL

Pacific Journal of Optimization
Vol. 21, No. 1, pp. 51–62, 2025

A GRADIENT PROJECTION METHOD FOR SEMI-SUPERVISED EVEN ORDER HYPERGRAPH CLUSTERING PROBLEMS*

Jingya Chang, Dongdong Liu and Min Xi†

Abstract: Semi-supervised clustering problems focus on clustering data with labels. In this paper, we consider the semi-supervised hypergraph problems. We use the hypergraph related tensor to construct an orthogonal constrained optimization model. The optimization problem is solved by a retraction method, which employs the polar decomposition to map the gradient direction in the tangent space to the Stefiel manifold. A nonmonotone curvilinear search is implemented to guarantee reduction in the objective function value. Convergence analysis demonstrates that the first order optimality condition is satisfied at the accumulation point. Experiments on synthetic hypergraph and hypergraph given by real data demonstrate the effectivity of the proposed method.

Key words: *tensor, hypergraph, semi-supervised clustering, manifold*

Mathematics Subject Classification: *05C65, 15A69, 65K05, 90C35*

1 Introduction

Clustering and classification are two important tasks in machine learning. Clustering approaches aim to divide a number of items without labels into several groups, while classification methods provide a classifier with the help of labeled data and classify other data by using the classifier. On the one hand data labeling in real life, such as getting labels in computer-aided diagnosis or part-of-speech tagging, is usually time-consuming or difficult [16]. On the other hand, sometimes in reality, few annotated points are melded in the unannotated data set in clustering problems and taking advantage of the priori label information often enhances the clustering performances.

Semi-supervised learning is to complete the learning task based on both the labeled and unlabeled data [21]. Semi-supervised clustering approach has wide applications in different areas. In image processing, the semi-supervised clustering approach was employed for image classification and segmentation [5]. A semi-supervised algorithm was proposed in [11] to solve the data de-duplication problem. For microarray expression data analysis, the knowledge

*This work was supported by the National Natural Science Foundation of China (Nos. 12326302, 12326334, 12171168,12101136), Natural Science Foundation of Guangdong Province (2022A515010839), Guangdong Basic and Applied Basic Research Foundations (Nos. 2023A1515011633,2020A1515110967), Guangdong Province Higher Education Foundation (No. 2021ZDZX1071), Project of Science and Technology of Guangzhou (No. 2024A04J2056).

†Corresponding author

 DOI: https://doi.org/10.61208/pjo-2023-025

from gene ontology data set was well utilized to generate the clustering algorithm [4, 6]. An active semi-supervised clustering method was applied to modeling the complex industrial process [13].

Hypergraph is a useful tool to save and describe the high dimensional and complex data arising from reality [17, 20]. Therefore, we consider the semi-clustering problems that are modeled by hypergraphs. We employ the multi way array to generate the clustering costs. By relaxing the label value and imposing the label matrix on the Stefiel manifold, we construct a tensor related optimization model and utilize the projection operator as a retraction to compute the feasible descent direction on the manifold. Numerical experiments show that our method works well on synthetic and real data.

The outline of this paper is as follows. In Section 2, we introduce the preliminary knowledge. The semi-clustering optimization model is given in Section 3, while the computing algorithm and convergent result are presented in Section 4 and Section 5 respectively. The numerical performance of our method is demonstrated in Section 6. Finally, we conclude our work in Section 7.

2 Preliminaries

In this section we demonstrate some useful notions and results on hypergraphs and tensors. Let $\mathbb{R}^{[r,n]}$ be the rth order n-dimensional real-valued tensor space, i.e.,

$$\mathbb{R}^{[r,n]} \equiv \mathbb{R}^{\overbrace{n \times n \times \cdots \times n}^{r\text{-times}}}.$$

The tensor $\mathcal{T} = (t_{i_1 \cdots i_r}) \in \mathbb{R}^{[r,n]}$ with $i_j = 1, \ldots, n$ for $j = 1, \ldots, r$, is said to be symmetric if $t_{i_1 \cdots i_r}$ is unchanged under any permutation of indices [3]. Two operations between $\mathcal{T}$ and any vector $\mathbf{x} \in \mathbb{R}^n$ are stipulated as

$$\mathcal{T}\mathbf{x}^r \equiv \sum_{i_1=1}^{n} \cdots \sum_{i_r=1}^{n} t_{i_1 \cdots i_r} \mathbf{x}_{i_1} \cdots \mathbf{x}_{i_r}$$

and

$$(\mathcal{T}\mathbf{x}^{r-1})_i \equiv \sum_{i_2=1}^{n} \cdots \sum_{i_r=1}^{n} t_{i i_2 \cdots i_r} \mathbf{x}_{i_2} \cdots \mathbf{x}_{i_r}, \quad \text{for} \quad i = 1, \ldots, n.$$

Note that, $\mathcal{T}\mathbf{x}^r \in \mathbb{R}$ and $\mathcal{T}\mathbf{x}^{r-1} \in \mathbb{R}^n$ are a scalar and a vector respectively, and $\mathcal{T}\mathbf{x}^r = \mathbf{x}^\top(\mathcal{T}\mathbf{x}^{r-1})$. The tensor outer product of $\mathcal{A} \in \mathbb{R}^{[p,n]}$ and $\mathcal{B} \in \mathbb{R}^{[q,n]}$ is given by

$$\mathcal{A} \circ \mathcal{B} = \left(a_{i_1 i_2 \ldots i_p} b_{j_1 j_2 \ldots j_q}\right) \in \mathbb{R}^{[p+q,n]}.$$

Definition 2.1 (Hypergraph)**.** A hypergraph is defined as $G = (V, E)$, where $V = \{1, 2, \ldots, n\}$ is the vertex set and $E = \{e_1, e_2, \ldots, e_m\} \subseteq 2^V$ (the powerset of V) is the edge set. We call G an r-uniform hypergraph when $|e_t| = r \geq 2$ for $t = 1, \ldots, m$ and $e_i \neq e_j$ in case of $i \neq j$.

If each edge of a hypergraph is linked with a positive number $s(e)$, then this hyperpragh is called a weighted hypergraph and $s(e)$ is the weight associated with the edge e. An ordinary hypergraph can be regarded as a weighted hypergraph with the weight of each edge being 1. The degree of a vertex i is denoted as $d_i = \sum_{i \in e} s(e)$.

The Stiefel manifold is $\mathcal{M}_n^p := \{X : X^T X = I, X \in R^{n \times p}\}$. We take the Euclidean metric as the Riemann metric on the Stiefel manifold and its tangent space. In this paper, we focus on the clustering problems of even order hypergraphs.

3 Semi-supervised Clustering Model

Consider the hypergraph semi-supervised clustering problem. Our task is to cluster the n vertices of the hypergraph into k groups according to the hypergraph structure, while few categorization labels are given. Here k is the number of clusters and is usually much less than n.

Denote an indicator matrix $X \in R^{n\times p}$ as

$$X_{ij} = \begin{cases} 1, & \text{the } i\text{th vertex is in the } j\text{th cluster;} \\ 0, & \text{otherwise.} \end{cases} \tag{3.1}$$

X is a matrix with 0-1 elements and its columns are orthogonal to each other. Consider the two vertices i_1 and i_2 that are contained in the same edge e. If these two vertices are divided in the same cluster, then the cluster cost is 0. Otherwise, a positive cluster cost arises, which can be calculated as

$$\sum_{j=1}^{p} s_e \left(\frac{X_{i_1 j}}{\sqrt[r]{d_{i_1}}} - \frac{X_{i_2 j}}{\sqrt[r]{d_{i_2}}} \right)^r,$$

where r is even. The symbols d_{i_1} and d_{i_2} are degrees of vertices i_1 and i_2 respectively, and r is the order of the hypergraph. Therefore, the total cutting cost of the hypergraph is

$$f(X) = \sum_{j=1}^{p} \sum_{e\in E} s_e \sum_{i_1, i_2 \in e} \left(\frac{X_{i_1 j}}{\sqrt[r]{d_{i_1}}} - \frac{X_{i_2 j}}{\sqrt[r]{d_{i_2}}} \right)^r. \tag{3.2}$$

It is shown in [2] that the cost function can be rewritten as sum of products of a tensor and vectors.

Proposition 3.1 ([2]). *For an even uniform weighted hypergraph $G = (V, E, \mathbf{s})$, we define the normalized Laplacian tensor*

$$\mathcal{L} = \sum_{e\in E} s_e \sum_{i,j\in e} \underbrace{\mathbf{u}_{ij} \circ \mathbf{u}_{ij} \circ \cdots \circ \mathbf{u}_{ij}}_{r \quad times}, \tag{3.3}$$

where $\mathbf{u}_{ij} = \frac{\mathbf{e}_i}{\sqrt[r]{d_i}} - \frac{\mathbf{e}_j}{\sqrt[r]{d_j}}$, $\mathbf{e}_i$ and $\mathbf{e}_j$ are column vectors with all elements being zero except the ith and jth entries are one respectively. Then, the cutting cost in (3.2) *can be rewritten as*

$$f(X) = \sum_{j=1}^{p} \mathcal{L}\mathbf{x}_j^r, \tag{3.4}$$

where $\mathbf{x}_j = X(:, j)$.

Next, we take into account the vertices labeled. Denote

$$Y_{ij} = \begin{cases} 1, & \text{the } i\text{th item is known in the } j\text{th cluster;} \\ 0, & \text{otherwise.} \end{cases} \tag{3.5}$$

to save the labels. We call the matrix Y the label matrix herein. It is natural that we try best to retain the labeled vertex in its predetermined cluster during the clustering process. Define

$$(X_Y)_{ij} = \begin{cases} X_{ij}, & \text{if } Y_{ij} \neq 0 \\ 0, & \text{otherwise.} \end{cases}$$

Hence, we solve the semi-supervised hypergraph clustering problem by

$$\begin{cases} \quad \min f(X) = \sum_{j=1}^{p} \mathcal{L}\mathbf{x}_j^r \\ \text{s.t.} \quad X_Y - Y = 0, \\ \quad X(:,j) = \mathbf{x}_j, \ \text{for } j = 1, \dots, p, \\ \quad X(i,j) = 0, \ \text{or } 1, \ \text{for } i = 1, \dots, n, j = 1, \dots, p. \end{cases} \tag{3.6}$$

This is a 0-1 integer programming. To make the optimization model easy, we penalize the constraint $X_Y - Y = 0$ and add a regularization term $\lambda \|X_Y - Y\|_F^2$ to the objective function. The λ here is a parameter. In terms of the discrete 0-1 constraint, it can be deduced from (3.1) that the column vectors of X are unit vectors and orthogonal to each other. Thus, we relax it to the continuous constraint $X^T X = I$. The optimization model is then transformed from discrete to continuous. Finally we get our orthogonal constrained model

$$\begin{cases} \min f(X) = \sum_{j=1}^{p} \mathcal{L}\mathbf{x}_j^r + \lambda \|X_Y - Y\|_F^2 \\ \text{s.t.} \quad X(:,j) = \mathbf{x}_j, \ \text{for } j = 1, \dots, p, \\ \quad X^T X = I. \end{cases} \tag{3.7}$$

for the semi-supervised clustering problem.

4 Computation

In this section, we first review the gradient of a function on the tangent space, and then we introduce the algorithm based on this gradient direction.

For optimization models constrained on the Steifel manifold, there are two threads to follow [9, 10, 1]. One way is transforming the constrained optimization problem into an unconstrained one by using mathematical programming techniques such as the penalty method, the augmented Lagrangian method [19] and then solve it by unconstrained optimization methods. The other way is first finding a descent direction in the tangent space from the current point, then mapping an appropriate point in the descent direction to the Stiefel manifold.

We adopt the second route to compute the orthogonal constrained problem (3.7). The tangent space at a point $X \in \mathcal{M}_n^p$ is

$$T_X := \{Z : X^T Z + Z^T X = 0\}.$$

Suppose the function $f(X) : \mathcal{M}_n^p \to R$ is differentiable. A retraction is a smooth mapping from the tangent bundle to the manifold [1, 18, 2]. Then for any $X \in \mathcal{M}_n^p$, the retraction $h_X(Z)$ is a mapping from $T_X \in R^{n\times p}$ to $\mathcal{M}_n^p$ with $h_X(0) = X$. The function $f(h_X(Z)) : T_X \to R$ is also differentiable. Denote the gradient of function f at the current point X as G. Because the objective function $f(X)$ is separable, we can compute the gradient G in parallel. Take the inner product of two matrices $\langle A, B\rangle$ as $\langle A, B\rangle = tr(A^T B)$. For any vector ξ in the tangent space of T_X the gradient of f at X projected onto the tangent space satisfies

$$\langle \nabla f(X), \xi\rangle = Df(x)[\xi].$$

In this paper, we take the direction $\nabla f(X)$ as $G - X\frac{X^TG+G^TX}{2}$. Next we show that $\nabla f(X) = 0$ for $X^TX = I$ is equivalent to the first order optimality condition of the constrained model (3.7).

Proposition 4.1. *Assume $X \in \mathcal{M}_n^p$. Let $\nabla f(X) = G - X\frac{X^TG+G^TX}{2}$. The first order optimality conditions of* (3.7) *hold if and only if*

$$\nabla f(X) = 0, \tag{4.1}$$

with $X^\top X = I$.

Proof. Let the symmetric matrix Λ be the Lagrangian multiplier of the constraint $X^\top X = I$. The Lagrangian function is $L = f(X) + \frac{1}{2}tr(\Lambda^\top(X^\top X - I))$. The linear independence constraint qualification is satisfied and the Karush-Kuhn-Tucker Conditions

$$\nabla L_X = G + X\Lambda = 0 \tag{4.2}$$

$$X^\top X = I \tag{4.3}$$

hold.

Similar to the proof in [18, Lemma 1], we multiply both sides of equation (4.2) by $X^\top$ and obtain $\Lambda = -X^\top G$, which is symmetric. Then we get the first order optimality condition of (3.7)

$$G - XG^\top X = 0, \qquad X^\top X = I. \tag{4.4}$$

Furthermore,

$$\nabla f = G - X\frac{X^TG + G^TX}{2} = G - XG^\top X = 0.$$

On the other hand if $\nabla f = 0$ and $X^\top X = I$, we get $X^\top G = G^\top X$ by multiplying both sides of ∇f by $X^\top$. Then

$$X^\top(G - XG^\top X) = 0.$$

Because X is full column rank, the above equation indicates that $G - XG^\top X = 0$. The first optimality conditions (4.4) hold. ☐

The direction $-\nabla f$ is employed to find a new point in the tangent space. In the iteration process, we find

$$\widetilde{X} = X_k - \alpha\nabla f$$

in the tangent space from X_k. The next step is to map $\widetilde{X}$ to the manifold by using a retraction. We project $\widetilde{X}$ to the manifold by utilizing the polar decomposition [1, 14]. The projection of any matrix Z onto the Stefiel manifold is defined as

$$h(Z) = \underset{Q\in\mathcal{M}_n^p}{\arg\min}\|Z - Q\|_F^2.$$

If the SVD of Z is $Z = U\Sigma V^T$, then the optimal solution of $h(Z)$ can be computed by UV^T [8]. Also, $h(Z)$ can be expressed as $Z(Z^TZ)^{-\frac{1}{2}}$ equivalently.

Proposition 4.2. *Suppose $X \in \mathcal{M}_n^p$ and Z is a vector in the tangent space T_X. Consider the univariate function*

$$h_X^Z(t) = \underset{Q\in\mathcal{M}_n^p}{\arg\min}\|Q - (X + tZ)\|_F^2$$

with its domain $t \in R$, which is the projection of $X + tZ$ onto the Stiefel manifold. The derivative of $h_X^Z(t)$ at $t = 0$ is

$$(h_X^Z)'(0) = Z. \tag{4.5}$$

When $Z = -\nabla f(X)$, the derivative of $f(h_X^Z(t))$ at $t = 0$ is

$$f_t'(h_X^Z(0)) = -\|\nabla f(X)\|^2. \tag{4.6}$$

Proof. For any arbitrary matrix $\tilde{Z} \in R^{n\times p}$, it can be decomposed as $\tilde{Z} = XA + X_\perp B + XC$, in which $A \in R^{p\times p}$ is skew-symmetric and $C \in R^{p\times p}$ is symmetric. It is proved in [14, Lemma 8] that,

$$h_X^{\tilde{Z}}(t) = X + t(XA + X_\perp B) + O(t^2). \tag{4.7}$$

On the other hand, the tangent vector space at X is $T_X = \{X\Omega + X_\perp K, \Omega = -\Omega, K \in R^{(n-p)\times p}\}$ [1]. Therefore, for $Z \in T_X$

$$h_X^Z(t) = X + tZ + O(t^2). \tag{4.8}$$

Since $h_X^Z(0) = X$, we get $(h_X^Z)'(0) = Z$.

By the chain rule, when $Z = -\nabla f(X)$ the derivative of $f(h_X^Z(t))$ with respect to t at $t = 0$ is

$$\begin{aligned}
f_t^{'}(h_X^Z(0)) &= \langle f'(X), (h_X^Z)'(0)\rangle = \langle G, -\nabla f(X)\rangle \\
&= -\langle \nabla f(X), \nabla f(X)\rangle - \frac{1}{2}\langle \nabla f(X), X(X^TG + G^TX)\rangle \\
&= -\langle \nabla f(X), \nabla f(X)\rangle - \frac{1}{2}[\langle X^T\nabla f(X), X^TG + G^TX\rangle] \\
&= -\langle \nabla f(X), \nabla f(X)\rangle - \frac{1}{4}[\langle X^TG - G^TX, X^TG + G^TX\rangle] \\
&= -\|\nabla f(X)\|^2.
\end{aligned}$$

□

Based on the above analysis, we employ a nonmonotone line search method to find a proper step size along $-\nabla f(X_k)$. Given X_k and the descent direction $Z = -\nabla f(X_k)$, by using the adaptive feasible BB-like method proposed in [10], we find a step size t_k such that

$$f(h_X^Z(t_k)) \leq f_r + \delta t_k f'(h_X^Z(0)), \tag{4.9}$$

where f_r is a reference objective function value. The new iteration point

$$X_{k+1} = h_X^Z(t_k) \tag{4.10}$$

is obtained. Let $S_{k-1} = X_k - X_{k-1}, Y_{k-1} = -\nabla f(X_k) + \nabla f(X_{k-1})$. The parameter t_k takes the following BB stepsizes alternately:

$$t_k^1 = \frac{\langle S_{k-1}, S_{k-1}\rangle}{|\langle S_{k-1}, Y_{k-1}\rangle|}, t_k^2 = \frac{|\langle S_{k-1}, Y_{k-1}\rangle|}{\langle Y_{k-1}, Y_{k-1}\rangle}.$$

Let L be a preassigned positive integer and f_{best} be the current best function value. Denote by f_C the maximum objective function value after f_{best} is found. The reference function value f_r is updated only when the best function value is not improved in L iterations. The detailed steps are shown below.

$$\begin{array}{ll} \texttt{if} & f_{k+1} < f_{best} \\ & f_{best} = f_{k+1}, f_c = f_{k+1}, l = 0 \\ \texttt{else} & \\ & f_c = \max\{f_{k+1}, f_c\}, l = l+1, \\ & \text{if } l = L, f_r = f_c, f_c = f_{k+1}, l = 0 \text{ end} \\ \texttt{end} & \end{array} \tag{4.11}$$

The proposed method is shown in Algorithm 1.

Algorithm 1 SSHC: Semi-supervised Hypergraph Clustering Algorithm

Require: $0 < \rho < 1, X_0 \in \mathcal{M}_n^p, 0 < \beta < 1, \epsilon > 0, L$ be a positive integer.

while do

Compute G at the point X_k in parallel.

Compute $\nabla f(X_k), Z = -\nabla f(X_k)$.

Find the smallest nonnegative integer m, such that

$$f(h^Z_{X_k}(\beta^m \hat{t_k})) \leq f_r + \delta \beta^m \hat{t_k} f'(h^Z_{X_k}(0)). \tag{4.12}$$

holds and set $t_k = \beta^m \hat{t_k}$ with $\hat{t_k} = t^1_k$ or t^2_k alternately.

Update X_{k+1} and f_r, f_{best}, f_c by (4.10) and (4.11) respectively.

end while

5 Convergence Analysis

The sequence $\{\nabla f(X_k)\}$ generated from Algorithm 1 either terminates with $\nabla f(X_k) = 0$ or is infinite. By reductio and absurdum, we prove that when the iteration is infinite, a subsequence of $\{\nabla f(X_k)\}$ converges to 0, which means

$$\lim_{k\to\infty} \inf \|\nabla f(X_k)\| = 0.$$

Lemma 5.1. *Assume there exists a constant $\varepsilon > 0$ such that*

$$\|\nabla f(X_k)\| \geq \varepsilon. \tag{5.1}$$

Under this assumption, the step size t_k generated by (4.12) satisfies

$$t_k \geq c, \tag{5.2}$$

where c is a constant.

Proof. Suppose the conclusion does not hold. Then we can find a subsequence $\{k_i\}$ satisfies that

$$t_{k_i} \to 0 \quad \text{as } k_i \to \infty.$$

We use the symbol k instead of k_i for simplicity. According to Proposition 4.2, when $Z = -\nabla f(X_k)$ the Taylor expansion of $f(h^Z_{X_k}(t))$ at the point $t = 0$ is

$$\begin{aligned} f(h^Z_{X_k}(t)) &= f(h^Z_{X_k}(0)) + t f'_t(h^Z_{X_k}(0)) + o(t) \\ &= f(X_k) - t\|\nabla f(X_k)\|^2 + o(t). \end{aligned} \tag{5.3}$$

If $\hat{t_k}$ is not accepted in the Armijo-type search (4.12), then we have $\hat{t_k} \geq \beta^{-1} t_k$ and

$$\begin{aligned} f(h^Z_{X_k}(\beta^{-1}t_k)) &> f_r + \delta\beta^{-1}t_k f'(h^Z_{X_k}(0) \\ &\geq f(X_k) - \delta\beta^{-1}t_k\|\nabla f(X_k)\|^2. \end{aligned} \tag{5.4}$$

Substituting $\beta^{-1}t_k$ for t in (5.3), we have

$$f(h^Z_{X_k}(\beta^{-1}t_k)) = f(X_k) - \beta^{-1}t_k\|\nabla f(X_k)\|^2 + o(t_k). \tag{5.5}$$

Since $t_k \to 0$ as $k \to \infty$, by combining (5.4) and (5.5), we obtain

$$(1-\delta)\|\nabla f(X_k)\|^2 + o(1) \leq 0.$$

This inequity is impossible, when $\|\nabla f(X_k)\| \geq \varepsilon$. The proof is then completed. □

Theorem 5.2. *If the sequence $\{X_k\}$ given by Algorithm 1 is infinite, then we have*

$$\liminf_{k\to\infty} \|\nabla f(X_k)\| = 0. \tag{5.6}$$

Proof. Suppose this conclusion is not true, then the assumption (5.1) holds. We save all values of f_r in (4.11) in the sequence $\{f_r^m\}$, where the index m means the mth value of f_r. Denote the index of the first iteration that is produced from the line search (4.9) related with f_r^m as k_m. Let l_m be the index number that satisfies $f(X_{l_m}) = \max_{k_m \leq j < k_{m+1}} f(X_j)$. From (4.12), we have

$$f(X_{l_m}) \leq f_r^m - \delta t_{l_m}\|\nabla f(X_{l_m})\|^2. \tag{5.7}$$

Also from the updating process we have

$$f_r^{m+1} \leq f(X_{l_m}). \tag{5.8}$$

By (5.7), (5.8) and (5.2), we obtain

$$f_r^{m+1} \leq f_r^m - \delta c\|\nabla f(X_{l_m})\|^2. \tag{5.9}$$

If $\{f_r^m\}$ is finite, the sequence of $\{f_{best}\}$ is infinite which contradicts with the fact that $f(X)$ is bounded below. Therefore, $\{f_r^m\}$ is an infinite sequence. Then based on Lemma 5.1 and (5.9) we get

$$+\infty > \sum_{m=1}^{+\infty}[f_r^m - f_r^{m+1}] \geq \sum_{m=1}^{+\infty} \delta c\|\nabla f(X_{l_m})\|^2, \tag{5.10}$$

which indicates that the assumption (5.1) is impossible. The conclusion (5.6) is finally proved. □

6 Numerical Experiments

In this section, we demonstrate the numerical performance of SSHC method for clustering synthetic and real data. For each problem, we run 200 times and report the average values of different evaluation indices produced by the 200 runs. The stopping criteria is set as

$$\|\nabla f(x)\| \leq 10^{-5}, \text{or} \ \ \text{IterNum} > 1000,$$

where "IterNum" means the total iteration number in each run.

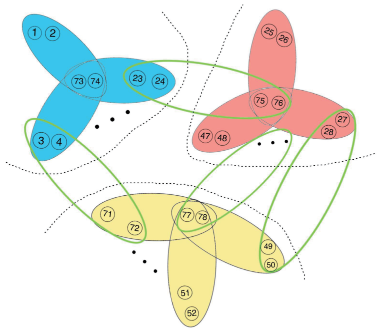

Figure 1: The 4-uniform hyerpgraph

Table 1: The clustering results with different proportions of labels

Methods \ Error	Max	Min	Average	Median
SHC	0.125000	0.125000	0.125000	0.125000
SSHC	0.145833	0.041667	0.098125	0.104167

6.1 Semi-supervised and supervised clustering of artificial hypergraphs

In this subsection, we employ the proposed SSHC method to cluster an artificial hypergraph. We compare the clustering results of SSHC method and the SHC method, which is in fact the unsupervised model by replacing the objective function in (3.7) as

$$f(X) = \sum_{j=1}^{p} \mathcal{L}\mathbf{x}_j^r.$$

In order to construct the 4-uniform hyerpgraph, we first generate three 4-uniform sub-hypergraphs. Each of the sub-hypergraph has 12 edges which share two common vertices. The three sub-hypergraphs are shown in Figure 1 with different colors. Next, we produce 4 more edges by choosing two vertices from the sub-hypergraphs and putting two pairs of the two vertices into one edge. These 4 edges are marked in green. The weight of this hypergraph is an all one vector. The final hypergraph is shown in Figure 1. Thus, the construction of the hypergraph in Figure 1 implies that its vertices can be divided into three clusters with the vertices in each sub-hypergraph belonging to the same cluster. Next we use SSHC and SHC method to cluster the given hypergraph. For the semi-supervised problem, 10 percent of the vertices are labeled. We report the maximum, minimum, average and median clustering errors in Table 1. It can be seen that the clustering accuracy is promoted by the proposed SSHC method.

6.2 Yale face data clustering

The extended Yale face data base B contains 600 face images of 30 persons under 20 lighting conditions [7, 12]. Figure 2 displays 8 images of 4 persons as an example. Our task is to group the images of each person into a cluster. Before computation, each image is resized into 48×42 pixels and expressed as a vector. Regard each image as a vertex in the hypergraph.

Figure 2: 8 images of 4 person from the Yale data set

Table 2: The clustering results with different proportions of labels

Ratio	Error	Iteration number	CPU time
0	0.06250	79	0.04115
0.1	0.05984	179	0.06804
0.2	0.05563	146	0.04886
0.3	0.04813	93	0.03542

For each vertex, we utilize the Nearest Subspace Neighbor (NSN) approach [15] to put the vertices that are closest to it into one edge. The weight of an edge are given based on the pairwise similarities of vertices that belong to this edge.

We cluster the images for 200 times and record the average clustering error, iteration numbers and runtime. The ratio of labeled images is shown in the Ratio column. The initial point is randomly chosen. In each trail, SSHC method computing the optimization problem with different ratios from the same initial point. Since the subsequence of SSHC method is guaranteed to converge to a stationary point theoretically, the average clustering accuracy, iteration number and computing time may vary slightly when the initial point changes. However, the consistent observation from our experiment is that the average values of clustering error, the iteration number as well as the computing time decrease when the ratio of labels increases from 0.1 to 0.3. Table 2 shows the numerical results of SSHC method for clustering the images in Figure 2.

7 Conclusion

In this paper, we give a tensor related optimization model to compute the hypergraph clustering problems with little part of labels provided. We use the polar decomposition as a retraction on the Stiefel manifold. The convergence analysis shows that an accumulation point of the iteration sequence is a stationary point. Numerical experiments indicate that the method improves the computation accuracy when compared to the unsupervised model. However, the effectiveness of the hypregraph clustering method relies on an appropriate hypergraph of the data. The construction of a hypergraph that reasonably reflects the data structure and relationship is a meaningful topic in our future research.

References

[1] P.-A. Absil, R. Mahony and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*, Princeton: Princeton University Press, 2009.

[2] J. Chang, Y. Chen, L. Qi and H. Yan, Hypergraph clustering using a new Laplacian tensor with applications in image processing, *SIAM J. Imag. Sci.* 13 (2020) 1157–1178.

[3] Y. Chen, L. Qi and Q. Wang, Positive semi-definiteness and sum-of-squares property of fourth order four dimensional Hankel tensors, *J. Comput. Appl. Math.* 302 (2015) 356–368.

[4] J. Cheng, M. Cline, J. Martin, D. Finkelstein, T. Awad, D. Kulp and M.A. Siani-Rose, A knowledge-based clustering algorithm driven by gene ontology, *J. Biopharm. Stat.* 14 (2004) 687–700.

[5] J. Enguehard, P. O'Halloran and A. Gholipour, Semi-supervised learning with deep embedded clustering for image classification and segmentation, *IEEE Access.* 7 (2019) 11093-11104.

[6] Z. Fang, J. Yang, Y. Li, Q. Luo and L. Liu, Knowledge guided analysis of microarray data, *J Biomed Inform.* 39 (2006) 401–11.

[7] A. S. Georghiades, P. N. Belhumeur and D. J. Kriegman, From few to many: illumination cone models for face recognition under variable lighting and pose, *IEEE Trans. Pattern Anal. Mach. Intell.* 23 (2001) 643–660.

[8] G. H. Golub and C. F. Van Loan, *Matrix computations*, Baltimore: JHU press, 2013.

[9] J. Hu, X. Liu, Z. Wen and Y. Yuan, A brief introduction to manifold optimization, *J. Oper. Res. Soc. China* 8 (2020) 199–248.

[10] B. Jiang and Y. Dai, A framework of constraint preserving update schemes for optimization on Stiefel manifold, *Math. Program.* 153 (2015) 535–575.

[11] S. Kushagra, S. David and I. Ilyas, Semi-supervised clustering for de-duplication, In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1659–1667. PMLR, 2019

[12] K. C. Lee, J. Ho and D. J. Kriegman, Acquiring linear subspaces for face recognition under variable lighting, *IEEE Trans. Pattern Anal. Mach. Intell.* 27 (2005) 684–698.

[13] Q. Lei, H. Yu, M. Wu and J. She, Modeling of complex industrial process based on active semi-supervised clustering, *Eng. Appl. Artif. Intell.* 56 (2016) 131–141.

[14] J. H. Manton, Optimization algorithms exploiting unitary constraints, *IEEE Trans. Signal Process.* 50 (2002) 635–650.

[15] D. Park, C. Caramanis and S. Sanghavi, Greedy subspace clustering, *Advances in neural information processing systems*, 27, 2014.

[16] J.E. van Engelen and H.H. Hoos, A survey on semi-supervised learning, *Mach. Learn.* 109 (2020) 373–440.

[17] N. Veldt, A. R. Benson and J. Kleinberg, Hypergraph cuts with general splitting functions, *SIAM Rev.* 64 (2022) 650–685.

[18] Z. Wen and W. Yin, A feasible method for optimization with orthogonality constraints, *Math. Program.* 142 (2013) 397–434.

[19] N. Xiao and X. Liu, Solving optimization problems over the stiefel manifold by smooth exact penalty function, *arXiv preprint arXiv:2110.08986*, 2021.

[20] P. Zhou, X. Wang, L. Du and X. Li, Clustering ensemble via structured hypergraph learning, *Inf. Fusion* 78 (2022) 171–179.

[21] Z. Zhou, A brief introduction to weakly supervised learning. *Natl. Sci. Rev.* 5 (2018) 44–53.

Manuscript received 11 January 2023
revised 21 July 2023
accepted for publication 21 July 2023

Jingya Chang
School of Mathematics and Statistics
Guangdong University of Technology
Guangzhou, 510520, China
E-mail address: jychang@gdut.edu.cn

Dongdong Liu
School of Mathematics and Statistics
Guangdong University of Foreign Studies
Guangzhou, 510006, China
E-mail address: ddliu@gdut.edu.cn

Min Xi
School of Mathematics and Statistics
Guangdong University of Foreign Studies
Guangzhou, 510006, China
E-mail address: mxi@gdufs.edu.cn