Data Min Knowl Disc
DOI 10.1007/s10618-008-0104-3

A global optimization method for semi-supervised clustering

Yu Xia

Received: 28 September 2006 / Accepted: 27 May 2008
Springer Science+Business Media, LLC 2008

Abstract In this paper, we adapt Tuy's concave cutting plane method to the semi-supervised clustering. We also give properties of local optimal solutions of the semi-supervised clustering. Numerical examples show that this method can give a better solution than other semi-supervised clustering algorithms do.

Keywords Partitional clustering · Semi-supervised clustering · Instance-level constraints · Global optimization · Concavity cuts

1 Introduction

We consider here complete exclusive partitional clustering. For supervised classification, the class labels of the data in a training set are known; and the task is to construct a prediction model based on the training set. In unsupervised classification, data are grouped without any a priori knowledge of the data labels. This paper is concerned with semi-supervised clustering, which are algorithms assuming some knowledge about the class labels of a subset of the data, and using this subset along with other data to partition the whole data set. A priori information about the class labels of the subset in semi-supervised clustering can be in different forms, and may not be the exact class labels themselves. Semi-supervised clustering is favorable in situations where class labels are expensive or impossible to obtain. Previous research on this topic has shown that semi-supervised clustering can produce clusters conforming better to class labels than unsupervised clustering do. There are many papers on this topic, we can only

Responsible editor: Charu Aggarwal.

Y. Xia (✉)
School of Mathematics, University of Birmingham, Edgbaston, Birmingham B15 2TT, UK
e-mail: xiay@maths.bham.ac.uk

sketch a few below. In this paper, we sometimes refer to a data point as an instance or a pattern, and refer to a cluster as a class or a group.

In Demiriz et al. (1999), the labels of a training set are assumed available. The authors modify the objective function of an unsupervised technique, e.g. the k-means clustering, to minimize both the within-cluster variance of the input attributes and a measure of cluster impurity based on the class labels. An unsupervised learning technique that is biased toward producing clusters as pure as possible in terms of class distributions is used to cluster data. These clusters can then be used to predict the classes of future points. In Gao et al. (2006), the authors consider clustering where the background information, specified in the form of labeled examples, has moderate overlapping features with the unlabeled data, but not in the same feature space. They formulate the problem as a constrained optimization problem and propose two learning algorithms to solve the problem based on hard and fuzzy clustering.

In many applications, individual labels are not known a priori; however, some instance-level information about the relationships between some data may be available. For example, a human expert can specify whether two instances are from the same, or different classes, although the expert does not know the exact class labels. Instance-level information includes must-link constraints and cannot-link constraints. Two instances are must-linked together if they should have the same, but unknown, class label. Likewise, two instances are cannot-linked together if they are from different, but unknown, groups. After the work of Wagstaff and Cardie (2000), there have been many papers on clustering with must-link and cannot-link constraints. An early survey on constrained classification is given in Gordon (1996). A recent bibliography on clustering with constraints is available at Ian Davidson's web page. Here, we are only able to mention a few papers related to this topic. Wagstaff et al. (2001) incorporate instance-level constrains in the k-means clustering. They show by numerical examples that their algorithm can reduce the number of iterations and total runtime of clustering, provide a better initial solution, increase clustering accuracy, and generalize the constraint information to improve performance on the unconstrained instances as well. Instance-level constraints are also used to learn a distortion measure. Xing et al. (2002), learn a Mahalanobis distance from the instance-level constrains and numerically show that using a Mahalanobis distance metric with instance-level constraints can improve the accuracy of the k-means clustering. Similar experiments have also been reported in Bar-Hillel et al. (2005) and Chang and Yeung (2006). As well, instance-level constraints have been integrated into EM algorithms to improve the similarity of mixture modeling clusters to class labels (Shental et al. 2003), have been incorporated into complete-link clustering by distorting pairwise proximities between instances (Klein et al. 2002). While the above mentioned algorithms strictly enforce instance-level constraints during each iteration, some algorithms impose a penalty on constraint violation for probabilistic cluster assignments (Basu et al. 2004; Lange et al. 2005). A k-means type algorithm that minimizes the constrained vector quantization error but at each iteration does not attempt to satisfy all constraints is presented in Davidson and Ravi (2005).

There are also semi-supervised clustering algorithms that assume both labeled data and instance-level constraints. In Basu et al. (2003), initial labeled data are used to seed a constrained k-means algorithm. Another class of semi-supervised clustering includes algorithms that allow users to iteratively provide feedback on some data (Cohn et al. 2003; Jain et al. 2006).

The focus of this paper is on clustering with instance-level constraints. In the rest of the paper, we refer semi-supervised clustering to clustering with instance-level constraints. We require constraints to be strictly satisfied in our algorithm. The reason for that requirement is that if two instances are known from the same group, it is reasonable to expect a clustering algorithm to output the same class label for them. Likewise, for cannot-linked instances, it is logical to demand them to have different class labels as a clustering result. As is mentioned in the previous part, numerical results from early research show that instance-level constraints do help improve the performance of clustering. However, semi-supervised clustering algorithms in current literature we are aware of are all some variants of the k-means or the EM algorithm, which can not guarantee to find a global solution. Actually, the k-means algorithm may not even produce a local optimal solution, which we will show later by our analysis and numerical examples. It is desirable that the clustering results be as accurate as possible. For instance, in credit risk analysis, it is bad to assign a low credit risk customer to the high risk group; and it is worse to classify a high risk customer into the low risk group.

In this paper, we use a global optimization approach to solve the semi-supervised clustering problem. Our numerical results show that our algorithm can get a better solution than existing algorithms do. In the absence of must-links and cannot-links, semi-supervised clustering reduces to traditional clustering, and the algorithm presents in this paper reduces to that in Xia and Peng (2005). In addition, we give properties of local optimal solutions for the semi-supervised clustering problem. Locating a local optimal solution is an important step in many clustering algorithms. The k-means algorithm is actually a local search method, i.e., moving to a neighboring point with the smallest objective value. In semi-supervised clustering, it is tempting to replace a set of must-linked points by the centroid of the set to save memory and computational time. Through our analysis, we will show when the must-linked set can be replaced by its centroid, when it can not. Thus, we hope that our results can also help improve other semi-supervised clustering algorithms.

The remainder of the paper is organized as follows. In Sect. 2, we give our mathematical model of the semi-supervised clustering. In Sect. 3, we discuss the properties of a local solution to the mathematical model. In Sect. 4, we describe our concavity cutting plane method for the model and discuss the complexity of our algorithm. In Sect. 5, we give some numerical examples.

2 The mathematical model

To describe our approach to the semi-supervised clustering, in this part, we present our mathematical model of the semi-supervised clustering problem.

Given n patterns represented as $\mathbf{a}_1, \ldots, \mathbf{a}_n \in \mathbb{R}^d$, we are going to partition them into k clusters $C_1, \ldots, C_k$. Let $\mathbf{c}_i$ $(i = 1, \ldots, k)$ denote the centroids of C_i $(i = 1, \ldots, k)$, which are representations of corresponding clusters. We use the k-means (square-error) criteria in this paper, as it is the most commonly used clustering criteria. For the k-means criteria, the dissimilarity measure is the squared Euclidean distance; the task is to assign each point to its closest cluster centroid.

The integer programming model. Let $X = \left[x_{ij}\right]$ denote the cluster membership matrix, i.e.

$$x_{ij} = \begin{cases} 1 & \mathbf{a}_i \in C_j \\ 0 & \text{otherwise} \end{cases} \quad (i = 1, \ldots, n;\ j = 1, \ldots, k).$$

For a positive semidefinite matrix $M \in \mathbb{R}^{d \times d}$, we denote

$$\|\mathbf{v}\|_M^2 = \mathbf{v}^T M \mathbf{v} \quad (\forall\, \mathbf{v} \in \mathbb{R}^d).$$

Note that when M is positive definite, $\|\cdot\|$ defines a norm on $\mathbb{R}^d$. When $M = I$, $\|\mathbf{v}\|_M$ reduces to the Euclidean norm.

Then the semi-supervised k-means clustering problem can be modeled as the following:

$$\min_{x_{ij},\, \mathbf{c}_j} \quad \sum_{j=1}^{k} \sum_{i=1}^{n} x_{ij} \left\| \mathbf{a}_i - \mathbf{c}_j \right\|_{M_j}^2 \tag{1a}$$

$$\text{s.t.} \quad \sum_{j=1}^{k} x_{ij} = 1 \quad (i = 1, \ldots, n) \tag{1b}$$

$$x_{rj} = x_{sj} \quad (\mathbf{a}_r \text{ and } \mathbf{a}_s \text{ must-linked; } j = 1, \ldots, k) \tag{1c}$$

$$x_{pj} + x_{qj} \leq 1 \quad (\mathbf{a}_p \text{ and } \mathbf{a}_q \text{ cannot-linked; } \ j = 1, \ldots, k) \tag{1d}$$

$$x_{ij} \in \{0, 1\} \quad (i = 1, \ldots, n;\ \ j = 1, \ldots, k). \tag{1e}$$

In the model, we use different loss functions for different clusters in the objective (1a). And (1b) is the assignment constraint; (1c) is the must-link constraint; (1d) is the cannot-link constraint. The traditional clustering is a special case of (1), i.e. $M_j = I$ (for $j = 1, \ldots k$), (1c) and (1d) are nonexistent.

Observe that (1) is a nonconvex nonlinear integer programming model (a real-valued function $f(\mathbf{x})$ is convex on its domain D if, and only if for any two points $\mathbf{x}_1, \mathbf{x}_2 \in D$ and any scalar $0 \leq \lambda \leq 1$, $f(\lambda \mathbf{x}_1) + f((1-\lambda)\mathbf{x}_2) \leq f(\lambda \mathbf{x}_1 + (1-\lambda)\mathbf{x}_2)$). For continuous minimization problems (the variables are real and continuous), there are polynomial-time algorithms for some structural convex programs (Nesterov 2004) (a minimization problem is called a convex program if its objective function and its feasible region are convex); otherwise, usually only a local optimal solution can be obtained in reasonable time. For an integer optimization problem (the variables are integer), two common approaches of obtaining an integer feasible solution are (i) iteratively adding cutting planes to remove fractional solutions from the continuous relaxation of the original problem; (ii) branching on the variables to eliminate suboptimal solutions; see Nemhauser and Wolsey (1988). In both approaches, a series of continuous relaxations of the original problem with increasing number of additional linear constraints need to be solved. Because of this, large-scale integer optimization problems without special structure are generally intractable.

Since it is difficult to solve (1) directly, we consider its continuous relaxation. To this end, we first define some notations. Let $\mathbf{x}_j$ represent the jth column of X, which is the membership vector of cluster C_j. We denote the number of patterns in cluster C_j by $n_j \overset{\text{def}}{=} \sum_{i=1}^n x_{ij}$. We first simplify the objective function (1a) by representing $\mathbf{c}_j$ with $\mathbf{x}_j$.

When $n_j > 0$, for fixed $\mathbf{x}_j$, $\sum_{i=1}^n x_{ij} \|\mathbf{a}_i - \mathbf{c}_j\|_{M_j}^2$ is convex in $\mathbf{c}_j$; so the minimum with regard to $\mathbf{c}_i$ is attended at $\frac{\partial\left(\sum_{i=1}^n x_{ij}\|\mathbf{a}_i - \mathbf{c}_j\|_{M_j}^2\right)}{\partial \mathbf{c}_j} = \mathbf{0}$, from which we obtain $\mathbf{c}_j = \frac{\sum_{i=1}^n x_{ij}\mathbf{a}_i}{\sum_{i=1}^n x_{ij}}$; i.e. $\mathbf{c}_j$ is the centroid of C_j. Note that $n_j = 0$ implies $\mathbf{x}_j = \mathbf{0}$. In this case, the minimum of the objective function is attained at $\mathbf{c}_j = \mathbf{0}$. Therefore, in the rest of the paper, we set

$$\mathbf{c}_j = \begin{cases} \frac{\sum_{i=1}^n x_{ij}\mathbf{a}_i}{\sum_{i=1}^n x_{ij}} & n_j > 0, \\ \mathbf{0} & n_j = 0. \end{cases} \tag{2}$$

Denote the square-error, or within-cluster variation, of cluster C_j as

$$\mathrm{SSE}_j(\mathbf{x}_j) = \begin{cases} \sum_{i=1}^n x_{ij} \left\| \mathbf{a}_i - \frac{\sum_{i=1}^n x_{ij}\mathbf{a}_i}{\sum_{i=1}^n x_{ij}} \right\|_{M_j}^2 & n_j > 0, \\ 0 & n_j = 0. \end{cases}$$

Let $\mathrm{SSE}(X) = \sum_{j=1}^k \mathrm{SSE}_j(\mathbf{x}_j)$. Then the objective function (1a) is

$$\min_X \mathrm{SSE}(X).$$

The continuous relaxation. Let the entries of the assignment matrix X be continuous real-valued variables instead of boolean variables in (1). We then get the continuous relaxation of the semi-supervised clustering:

$$\begin{aligned} \min\nolimits_X \ & \mathrm{SSE}(X) \\ \text{s.t.} \ & \sum_{j=1}^k x_{ij} = 1 \quad (i = 1, \ldots, n) \\ & x_{rj} = x_{sj} \quad (\mathbf{a}_r \text{ and } \mathbf{a}_s \text{ must-linked; } r, s = 1, \ldots, n; \ j = 1, \ldots, k) \\ & x_{pj} + x_{qj} \le 1 \quad (\mathbf{a}_p \text{ and } \mathbf{a}_q \text{ cannot-linked; } p, q = 1, \ldots, n; \ j = 1, \ldots, k) \\ & x_{ij} \ge 0 \quad (i = 1, \ldots, n; \ j = 1, \ldots, k). \end{aligned} \tag{3}$$

In the next section, we will study the properties of (1) and (3), and show when (1) can be replaced with (3).

3 The solutions

In this part, we study the properties of (1) and (3). It is known that the must-link represents an equivalence relation, i.e., symmetric, reflexive, and transitive. We take transitive closures over the constrains. A transitive closure of must-links includes all the patterns that are must-linked together. For instance, if $\mathbf{a}_i$ is must-linked to $\mathbf{a}_j$, and $\mathbf{a}_j$ is must-linked to $\mathbf{a}_l$; then $\mathbf{a}_i$, $\mathbf{a}_j$, and $\mathbf{a}_l$ are in the same must-link closure. If a pattern is not must-linked to any other pattern, its must-link closure is a singleton. We also take transitive closure of cannot-links. For instance, if a pattern in a must-link closure is cannot-linked to another pattern in a different must-link closure; then the two must-link closures are cannot-linked to each other.

The rest of this section is organized as follows. In Sect. 3.1, we prove that (3) is a concave program and the extreme points of its feasible region are integer if cannot-links do not exist. We also give some basic equalities that relate the variation of distorted square-error to that of assignment variables. In Sect. 3.2, we discuss whether an optimal solution can have empty clusters and whether the permutation of labels will affect SSE. In Sect. 3.3, we give local optimality conditions of (1) and (3) where cannot-links are nonexistent, and discuss how to find an integer local optimal solution. In Sect. 3.4, we give optimality conditions for (1) and (3) in the presence of cannot-links.

3.1 General properties of the mathematical model

We first show that (3) is a concave program, i.e., $-\operatorname{SSE}(X)$ is convex and the feasible region of (3) is convex; see also Xia 2007. Thus, we can apply concave optimization techniques to it. We use $X \geq 0$ to represent that X is entry-wise nonnegative, i.e. $x_{ij} \geq 0$ (for $i = 1, \ldots, n$; $j = 1, \ldots, k$).

Lemma 1 *The function* $\operatorname{SSE}(X)$ *is concave and continuously differentiable over* $X \geq 0$.

Proof To prove that $\operatorname{SSE}(X)$ is concave over $X \geq 0$, we only need to show that its Hessian exists and is negative semidefinite over $X \geq 0$. To calculate the Hessian of $\operatorname{SSE}(X)$, we first derive its gradient.

Let $\mathbf{e}_i$ denote the vector whose ith entry is 1 and the remaining entries are 0. We calculate the Hessian based on whether $n_j = 0$ or not. When $n_j = 0$, by definition,

$$\frac{\partial \operatorname{SSE}(X)}{\partial x_{lj}} = \lim_{t \to 0} \frac{\operatorname{SSE}(X + t\mathbf{e}_l\mathbf{e}_j^T) - \operatorname{SSE}(X)}{t} = \lim_{t \to 0} \frac{t \left\| \mathbf{a}_l - \frac{t\mathbf{a}_l}{t} \right\|_{M_j}^2}{t} = 0.$$

When $n_j > 0$, by chain rule,

$$\frac{\partial \operatorname{SSE}(X)}{\partial x_{lj}} = \left\| \mathbf{a}_l - \frac{\sum_{i=1}^n x_{ij}\mathbf{a}_i}{\sum_{i=1}^n x_{ij}} \right\|_{M_j}^2 + 2\sum_{i=1}^n x_{ij} \left(\frac{\sum_{p=1}^n x_{pj}\mathbf{a}_p}{\sum_{p=1}^n x_{pj}} - \mathbf{a}_i \right)^T$$

$$M_j \left(\frac{\mathbf{a}_l}{\sum_{p=1}^n x_{pj}} - \frac{\sum_{p=1}^n x_{pj}\mathbf{a}_p}{\left(\sum_{p=1}^n x_{pj}\right)^2} \right).$$

Since $\sum_{i=1}^n x_{ij} \left(\frac{\sum_{p=1}^n x_{pj}\mathbf{a}_p}{\sum_{p=1}^n x_{pj}} - \mathbf{a}_i \right) = \mathbf{0}$ and $\left(\frac{\mathbf{a}_l}{\sum_{p=1}^n x_{pj}} - \frac{\sum_{p=1}^n x_{pj}\mathbf{a}_p}{\left(\sum_{p=1}^n x_{pj}\right)^2} \right)$ is independent of i, the second term in the above equality vanishes.

Therefore,

$$\frac{\partial\, \mathrm{SSE}(X)}{\partial x_{lj}} = \begin{cases} \left\| \mathbf{a}_l - \frac{\sum_{i=1}^n x_{ij}\mathbf{a}_i}{\sum_{i=1}^n x_{ij}} \right\|_{M_j}^2 & n_j > 0\,, \\ 0 & n_j = 0. \end{cases} \tag{4}$$

Let $\mathbf{v}_{lj} \in \mathbb{R}^d$ denote the difference of the lth pattern from the centroid of cluster C_j, i.e.,

$$\mathbf{v}_{lj} \overset{\text{def}}{=} \mathbf{a}_l - \mathbf{c}_j = \mathbf{a}_l - \frac{\sum_{i=1}^n x_{ij}\mathbf{a}_i}{\sum_{i=1}^n x_{ij}}.$$

Then from (4), we obtain that for any $l, g \in \{1, \dots, n\}$ and $j, m \in \{1, \dots, k\}$:

$$\frac{\partial^2\, \mathrm{SSE}(X)}{\partial x_{lj} x_{gm}} = \begin{cases} -\frac{2}{n_j}\mathbf{v}_{lj}^T M_j \mathbf{v}_{gj} & j = m \text{ and } n_j > 0 \\ 0 & j \neq m \text{ or } n_j = 0. \end{cases}$$

Let V_j denote the matrix whose lth row is the vector $\mathbf{v}_{lj}^T$. Then the Hessian of SSE_j is

$$\nabla^2\, \mathrm{SSE}_j(X) = \begin{cases} -\frac{2}{n_j} V_j M_j V_j^T & n_j > 0 \\ 0 & n_j = 0 \end{cases},$$

from which we obtain that the Hessian of $\mathrm{SSE}(X)$,

$$\nabla^2\, \mathrm{SSE}(X) = \begin{bmatrix} \nabla^2\, \mathrm{SSE}_1(X) & & \\ & \ddots & \\ & & \nabla^2\, \mathrm{SSE}_k(X) \end{bmatrix},$$

is negative semidefinite over $X \geq 0$. This concludes our proof. □

Let D denote the feasible region of (3). It is a polytope, i.e. it is bounded and is the intersection of a finite number of half spaces (linear inequalities). It follows that D is a convex set.

We have proved that (3) is a concave program. Next we describe the extreme points of D, since the minimum of a concave function is achieved at some extreme points of its feasible region.

Proposition 1 *Any integer feasible solution to* (3) *is an extreme point (vertex) of D. And any extreme point of D is an integer feasible solution to* (3) *in the absence of cannot-links* (1d).

Proof We first prove that any integer feasible solution X of (3) is an extreme point of D, i.e., there do not exist two points $Y,\ Z \in D$, $Y \neq Z$, and a scalar $0 < \lambda < 1$, such that $X = \lambda Y + (1 - \lambda) Z$.

For $i = 1, \ldots, n$, assume $x_{ij} = 1$, where j depends on i. From (1b), we have $x_{il} = 0\,(l \neq j)$. Then for any $0 < \lambda < 1$, the equality $x_{il} = \lambda y_{il} + (1 - \lambda) z_{il}$ and $y_{il} \geq 0$, $z_{il} \geq 0$ imply

$$y_{il} = z_{il} = 0 \quad (l \neq j). \tag{5}$$

From (5) and (1b), we have

$$y_{ij} = z_{ij} = 1.$$

Therefore, $Y = Z = X$; in other words, X cannot be represented as a nontrivial convex combination of two points in D. Therefore, X is an extreme point of D.

Next, we prove that any extreme point of D is an integer feasible solution to (3) in the absence of cannot-links. We only need to show that any non-integer feasible solution to (3) is not an extreme point of D.

Let X be a non-integer feasible solution to (3), i.e. there exists a must-link closure $\{\mathbf{a}_{m_p}\}_{p=1}^{r}$ such that for some $j \in \{1, \ldots, k\}$, $0 < x_{ij} < 1\,(i = m_1, \ldots, m_r)$. By (1b), there exists $l \neq j$ such that $0 < x_{il} < 1\,(i = m_1, \ldots, m_r)$. In addition, we have $0 < x_{ij} + x_{il} \leq 1\,(i = m_1, \ldots, m_r)$. Let Y be a matrix with $y_{ij} = x_{ij} + x_{il},\ y_{il} = 0,\ (i = m_1, \ldots, m_r)$, and other components being the same as those of X. Let Z be a matrix with $z_{il} = x_{ij} + x_{il},\ z_{ij} = 0,\ (i = m_1, \ldots, m_r)$, and other components being the same as those of X. Then $Y, Z \in D$. Let $\lambda = \frac{x_{ij}}{x_{ij}+x_{il}}\ (i = m_1, \ldots, m_r)$. We obtain $X = \lambda Y + (1 - \lambda) Z$ and $0 < \lambda < 1$. Therefore, X is not an extreme point of D. □

We have proved that the set of extreme points of D is exactly the set of integer feasible solutions to (1) in the absence of (1d). The minimum of a concave function is attained on the facets of its feasible region; so Proposition 1 indicates that when cannot-links do not exist, we can solve (3) instead of (1). The traditional clustering model fits into the above proposition. Unfortunately, $\mathrm{SSE}(X)$ is not strictly concave over $X \geq 0$; thus, we cannot conclude that all of its minimal solutions are the extreme points of its feasible region. In the next part, we will describe the local solutions to (3) and (1). From the properties of the local optimal solutions, we will derive a procedure of moving from a non-integer solution of (3) to an integer solution with better objective value in the presence of cannot-links. To this end, we first give some observations.

Let's consider $r\,(1 \leq r \leq n)$ patterns: $\mathbf{a}_{m_1}, \ldots, \mathbf{a}_{m_r}$, which may or may not be must-linked together. Let $0 \leq x_{m_p j} \leq 1$ denote the membership (assignment variable) of $\mathbf{a}_{m_p}$ related to cluster C_j. Let $\Delta x_{m_p j}$ denote the variation of $x_{m_p j}$, with $\Delta x_{m_p j} > 0$ (resp. < 0) if $\mathbf{a}_{m_p}$ moves to (resp. away from) cluster C_j. We're interested in the relation between the variation in the square-error and that in the membership of the patterns.

Proposition 2 *Assume that $x_{m_p j}^{old}$ is changed to $x_{m_p j}^{new} \overset{\text{def}}{=} x_{m_p j}^{old} + \Delta x_{m_p j} \geq 0$ ($p = 1, \ldots, r$). Then the sum of square-error of cluster C_j is shifted by*

$$\mathrm{SSE}_j^{new} - \mathrm{SSE}_j^{old} = \begin{cases} -\frac{\left\| \sum_{p=1}^r \Delta x_{m_p j} (\mathbf{a}_{m_p} - \mathbf{c}_j^{old}) \right\|_{M_j}^2}{n_j^{old} + \sum_{p=1}^r \Delta x_{m_p j}} & \\ \quad + \sum_{p=1}^r \Delta x_{m_p j} \left\| \mathbf{a}_{m_p} - \mathbf{c}_j^{old} \right\|_{M_j}^2 & n_j^{new} > 0\,, \\ \sum_{p=1}^r \Delta x_{m_p j} \left\| \mathbf{a}_{m_p} - \mathbf{c}_j^{old} \right\|_{M_j}^2 & n_j^{new} = 0\,, \end{cases} \tag{6}$$

where $\mathbf{c}_j$ is defined in (2).

Proof The number of patterns in the new cluster C_j is $n_j^{\mathrm{new}} = n_j^{\mathrm{old}} + \sum_{p=1}^r \Delta x_{m_p j}$. There are four cases: (1) $n_j^{\mathrm{old}} > 0, n_j^{\mathrm{new}} > 0$; (2) $n_j^{\mathrm{old}} > 0, n_j^{\mathrm{new}} = 0$; (3) $n_j^{\mathrm{old}} = 0, n_j^{\mathrm{new}} = 0$; (4) $n_j^{\mathrm{old}} = 0, n_j^{\mathrm{new}} > 0$.

We first assume $n_j^{\mathrm{old}} > 0$.

Under this assumption, we first consider the case $n_j^{\mathrm{new}} > 0$.

The new centroid of cluster C_j is $\mathbf{c}_j^{\mathrm{new}} = \frac{\mathbf{c}_j^{\mathrm{old}} n_j^{\mathrm{old}} + \sum_{p=1}^r \Delta x_{m_p j} \mathbf{a}_{m_p}}{n_j^{\mathrm{new}}}$.

Therefore,

$$\mathbf{c}_j^{\mathrm{new}} - \mathbf{c}_j^{\mathrm{old}} = \frac{\sum_{p=1}^r \Delta x_{m_p j} \left(\mathbf{a}_{m_p} - \mathbf{c}_j^{\mathrm{old}} \right)}{n_j^{\mathrm{new}}}. \tag{7}$$

Noting $\sum_{i=1}^n x_{ij} (\mathbf{a}_{ij} - \mathbf{c}_j) = 0$ for both old and new cluster C_j by the definition of $\mathbf{c}_j$, we have

$$\begin{aligned} \mathrm{SSE}_j^{\mathrm{new}} &= \sum_{i=1}^n x_{ij}^{\mathrm{new}} \left\| \mathbf{a}_i - \mathbf{c}_j^{\mathrm{new}} \right\|_{M_j}^2 = \sum_{i=1}^n x_{ij}^{\mathrm{old}} \left\| \mathbf{a}_i - \mathbf{c}_j^{\mathrm{old}} + \mathbf{c}_j^{\mathrm{old}} - \mathbf{c}_j^{\mathrm{new}} \right\|_{M_j}^2 \\ &\quad + \sum_{p=1}^r \Delta x_{m_p j} \left\| \mathbf{a}_{m_p} - \mathbf{c}_j^{\mathrm{old}} + \mathbf{c}_j^{\mathrm{old}} - \mathbf{c}_j^{\mathrm{new}} \right\|_{M_j}^2 = \mathrm{SSE}_j^{\mathrm{old}} + n_j^{\mathrm{old}} \left\| \mathbf{c}_j^{\mathrm{old}} - \mathbf{c}_j^{\mathrm{new}} \right\|_{M_j}^2 \\ &\quad + \sum_{p=1}^r \Delta x_{m_p j} \left\| \mathbf{a}_{m_p} - \mathbf{c}_j^{\mathrm{old}} \right\|_{M_j}^2 + \left(\sum_{p=1}^r \Delta x_{m_p j} \right) \left\| \mathbf{c}_j^{\mathrm{old}} - \mathbf{c}_j^{\mathrm{new}} \right\|_{M_j}^2 \\ &\quad + 2 \sum_{p=1}^r \Delta x_{m_p j} \left(\mathbf{a}_{m_p} - \mathbf{c}_j^{\mathrm{old}} \right)^T M_j \left(\mathbf{c}_j^{\mathrm{old}} - \mathbf{c}_j^{\mathrm{new}} \right). \end{aligned}$$

Replacing $\mathbf{c}_j^{\text{new}} - \mathbf{c}_j^{\text{old}}$ in the above equation by (7), we get

$$\begin{aligned}
\text{SSE}_j^{\text{new}} &= \text{SSE}_j^{\text{old}} + \frac{n_j^{\text{old}}}{\left(n_j^{\text{new}}\right)^2} \left\| \sum_{p=1}^{r} \Delta x_{m_p j} \left(\mathbf{a}_{m_p} - \mathbf{c}_j^{\text{old}}\right) \right\|_{M_j}^2 + \sum_{p=1}^{r} \Delta x_{m_p j} \left\| \mathbf{a}_{m_p} - \mathbf{c}_j^{\text{old}} \right\|_{M_j}^2 \\
&\quad + \frac{\sum_{p=1}^{r} \Delta x_{m_p j}}{\left(n_j^{\text{new}}\right)^2} \left\| \sum_{p=1}^{r} \Delta x_{m_p j} \left(\mathbf{a}_{m_p} - \mathbf{c}_j^{\text{old}}\right) \right\|_{M_j}^2 \\
&\quad - \frac{2}{n_j^{\text{new}}} \left\| \sum_{p=1}^{r} \Delta x_{m_p j} \left(\mathbf{a}_{m_p} - \mathbf{c}_j^{\text{old}}\right) \right\|_{M_j}^2 \\
&= \text{SSE}_j^{\text{old}} - \frac{\left\| \sum_{p=1}^{r} \Delta x_{m_p j} \left(\mathbf{a}_{m_p} - \mathbf{c}_j^{\text{old}}\right) \right\|_{M_j}^2}{n_j^{\text{old}} + \sum_{p=1}^{r} \Delta x_{m_p j}} + \sum_{p=1}^{r} \Delta x_{m_p j} \left\| \mathbf{a}_{m_p} - \mathbf{c}_j^{\text{old}} \right\|_{M_j}^2 .
\end{aligned}$$

Now we consider $n_j^{\text{new}} = 0$. Obviously $\text{SSE}_j^{\text{new}} = 0$. If in addition, $\sum_{p=1}^{r} \Delta x_{m_p j} = 0$; then we have $n_j^{\text{old}} = 0$ and $\text{SSE}_j^{\text{old}} = 0$ as well. Now we assume $\sum_{p=1}^{r} \Delta x_{m_p j} \neq 0$. Since $n_j^{\text{new}} = n_j^{\text{old}} + \sum_{p=1}^{r} \Delta x_{m_p j} = 0$, we have

$$\mathbf{c}_j^{\text{old}} = \frac{\sum_{p=1}^{r} \Delta x_{m_p j} \mathbf{a}_{m_p}}{\sum_{p=1}^{r} \Delta x_{m_p j}}, \quad \text{SSE}_j^{\text{old}} = -\sum_{p=1}^{r} \Delta x_{m_p j} \left\| \mathbf{a}_{m_p} - \mathbf{c}_j^{\text{old}} \right\|_{M_j}^2 .$$

Thus, we have proved (6) under $n_j^{\text{old}} > 0$.

Now, let us assume $n_j^{\text{old}} = 0$.

By (2), $\mathbf{c}_j^{\text{old}} = \mathbf{0}$.

If $n_j^{\text{new}} = 0$, from $X \geq 0$, we conclude that $\Delta x_{m_p j} = 0$; hence $\text{SSE}^{\text{new}} = \text{SSE}^{\text{old}} = 0$. That is included in (6).

Now we consider $n_j^{\text{new}} > 0$. Since $n_j^{\text{new}} = \sum_{p=1}^{r} \Delta x_{m_p j}$, we have, $\mathbf{c}_j^{\text{new}} = \frac{\sum_{p=1}^{r} \Delta x_{m_p j} \mathbf{a}_{m_p}}{n_j^{\text{new}}}$. Thus,

$$\begin{aligned}
\text{SSE}_j^{\text{new}} &= \sum_{p=1}^{r} \Delta x_{m_p j} \left\| \mathbf{a}_{m_p} - \mathbf{c}_j^{\text{new}} \right\|_{M_j}^2 = \sum_{p=1}^{r} \Delta x_{m_p j} \left\| \mathbf{a}_{m_p} \right\|_{M_j}^2 \\
&\quad -2 \sum_{p=1}^{r} \Delta x_{m_p j} \mathbf{a}_{m_p j}^T M_j \frac{\sum_{p=1}^{r} \Delta x_{m_p j} \mathbf{a}_{m_p}}{\sum_{p=1}^{r} \Delta x_{m_p j}} \\
&\quad + \sum_{p=1}^{r} \Delta x_{m_p j} \left\| \frac{\sum_{p=1}^{r} \Delta x_{m_p j} \mathbf{a}_{m_p}}{\sum_{p=1}^{r} \Delta x_{m_p j}} \right\|_{M_j}^2
\end{aligned}$$

$$= -\frac{\left\|\sum_{p=1}^{r} \Delta x_{m_p j} \mathbf{a}_{m_p}\right\|_{M_j}^2}{\sum_{p=1}^{r} \Delta x_{m_p j}} + \sum_{p=1}^{r} \Delta x_{m_p j} \left\|\mathbf{a}_{m_p}\right\|_{M_j}^2 .$$

Therefore, we have proved (6). □

Corollary 1 *Suppose that $\Delta x_{m_p j}$ of $\mathbf{a}_{m_p}$ $(p = 1, \ldots, r)$ is moved from cluster C_j to cluster C_l. Denote $t = \sum_{p=1}^{r} \Delta x_{m_p l}$. Assume $t \geq 0$. Then the change of the sum of square-error is*

$$\left(\mathrm{SSE}_j^{new} + \mathrm{SSE}_l^{new}\right) - \left(\mathrm{SSE}_j^{old} + \mathrm{SSE}_l^{old}\right) \tag{8}$$

$$= \begin{cases} -\frac{\left\|t\mathbf{c}_l^{old} - \sum_{p=1}^{r} \Delta x_{m_p l} \mathbf{a}_{m_p}\right\|_{M_l}^2}{n_l^{old} + t} - \frac{\left\|t\mathbf{c}_j^{old} - \sum_{p=1}^{r} \Delta x_{m_p l} \mathbf{a}_{m_p}\right\|_{M_j}^2}{n_j^{old} - t} & \\ \quad + \sum_{p=1}^{r} \Delta x_{m_p l} \left[\left\|\mathbf{a}_{m_p} - \mathbf{c}_l^{old}\right\|_{M_l}^2 - \left\|\mathbf{a}_{m_p} - \mathbf{c}_j^{old}\right\|_{M_j}^2\right] & (n_j^{new} > 0,\ n_l^{new} > 0); \\ -\frac{\left\|t\mathbf{c}_l^{old} - \sum_{p=1}^{r} \Delta x_{m_p l} \mathbf{a}_{m_p}\right\|_{M_l}^2}{n_l^{old} + t} & \\ \quad + \sum_{p=1}^{r} \Delta x_{m_p l} \left[\left\|\mathbf{a}_{m_p} - \mathbf{c}_l^{old}\right\|_{M_l}^2 - \left\|\mathbf{a}_{m_p} - \mathbf{c}_j^{old}\right\|_{M_j}^2\right] & (n_j^{new} = 0,\ n_l^{new} > 0); \\ 0 & (n_l^{new} = 0). \end{cases}$$

Proof We have $\Delta x_{m_p j} = -\Delta x_{m_p l}$ $(p = 1, \ldots, r)$, $n_l^{\mathrm{new}} = n_l^{\mathrm{old}} + t$, and $n_j^{\mathrm{new}} = n_j^{\mathrm{old}} - t$. The results follow from (6) by adding the variations in the square-errors of cluster C_j and C_l together. □

Remark 1 We assume $t \geq 0$ in Corollary 1. By symmetry, via swapping l and j we can obtain similar results for $t < 0$.

3.2 Empty cluster and label permutation at a local minimum

Let A represent the data matrix, i.e., let its ith row be the feature vector of pattern $\mathbf{a}_i$. Let $\mathrm{SSE}^*(A; k)$ denote the minimum value of total with-in group square-error sum for partitioning data A into k groups. It is known that $\mathrm{SSE}^*(A; k)$ is a strictly decreasing function of k for the integer model without constraints and under Euclidean metric. That means that in the traditional clustering, there is no empty cluster at an optimum. And it is also obvious that SSE is immune to label permutation for traditional clustering. In other words, let $p(1), \ldots, p(k)$ be a permutation of $1, \ldots, k$ (the set $\{p(1), \ldots, p(k)\}$ equals the set $\{1, \ldots, k\}$); then $\mathrm{SSE}(X) = \mathrm{SSE}(\tilde{X})$, where $\tilde{x}_{ij} = x_{ip(j)}$, $(i = 1, \ldots, n;\ j = 1, \ldots, k)$. In this part, we analyze whether these properties still hold at an optimum in the presence of must-links, cannot-links, and non-Euclidean metrics. We define the neighborhood of a solution X to (1) to be those feasible assignment matrices differing from X in only one must-link closure membership. Let's first give a proposition.

Proposition 3 *Let C_l^{old} be empty. Then the variation of the sum of square-error produced by moving Δx_{m_p} of $\mathbf{a}_{m_p}$ $(p = 1, \ldots, r)$ from C_j to C_l is*

$$\mathrm{SSE}^{new} - \mathrm{SSE}^{old} = \begin{cases} \sum_{p=1}^{r} \Delta x_{m_p} \left\| \mathbf{a}_{m_p} - \frac{\sum_{p=1}^{r} \Delta x_{m_p} \mathbf{a}_{m_p}}{t} \right\|_{M_l - M_j}^2 \\ -\frac{n_j^{old}}{n_j^{old} - t} t \left\| \frac{\sum_{p=1}^{r} \Delta x_{m_p} \mathbf{a}_{m_p}}{t} - \mathbf{c}_j^{old} \right\|_{M_j}^2 & (n_j^{old} > t), \\ \sum_{p=1}^{r} \Delta x_{m_p} \left\| \mathbf{a}_{m_p} - \frac{\sum_{p=1}^{r} \Delta x_{m_p} \mathbf{a}_{m_p}}{t} \right\|_{M_l - M_j}^2 & (n_j^{old} = t) \end{cases} \quad (9)$$

where $t = \sum_{p=1}^{r} \Delta x_{m_p} > 0$. (Here, for a vector $\mathbf{b} \in \mathbb{R}^n$, we use $\|\mathbf{b}\|_{M_l - M_j}^2$ to represent $\mathbf{b}^T (M_l - M_j) \mathbf{b}$, although $(M_l - M_j)$ may not be positive definite.)

Proof From the assumption that C_l^{old} is empty, we have $n_l^{\text{new}} = t$ and $\mathbf{c}_l^{\text{new}} = \frac{\sum_{p=1}^{r} \Delta x_{m_p} \mathbf{a}_{m_p}}{t}$.

We first consider $n_j^{\text{old}} = \sum_{p=1}^{r} \Delta x_{m_p}$. This case implies $n_j^{\text{new}} = 0$ and

$$\mathbf{c}_j^{\text{old}} = \frac{\sum_{p=1}^{r} \Delta x_{m_p} \mathbf{a}_{m_p}}{t}. \quad (10)$$

Therefore, from the second case in (8) we get

$$\begin{aligned} \mathrm{SSE}^{\text{new}} - \mathrm{SSE}^{\text{old}} = -t \left\| \mathbf{c}_j^{\text{old}} \right\|_{M_l}^2 + \sum_{p=1}^{r} \Delta x_{m_p} \|\mathbf{a}_{m_p}\|_{M_l}^2 \\ - \sum_{p=1}^{r} \Delta x_{m_p} \left\| \mathbf{a}_{m_p} - \mathbf{c}_j^{\text{old}} \right\|_{M_j}^2 . \end{aligned}$$

By (10),

$$\begin{aligned} -t \|\mathbf{c}_j^{\text{old}}\|_{M_l}^2 &= -2t \|\mathbf{c}_j^{\text{old}}\|_{M_l}^2 + t \|\mathbf{c}_j^{\text{old}}\|_{M_l}^2 \\ &= -2 \sum_{p=1}^{r} \Delta x_{m_p} \mathbf{a}_{m_p}^T M_l \mathbf{c}_j^{\text{old}} + \sum_{p=1}^{r} \Delta x_{m_p} \|\mathbf{c}_j^{\text{old}}\|_{M_l}^2. \end{aligned}$$

Hence, we have

$$\mathrm{SSE}^{\text{new}} - \mathrm{SSE}^{\text{old}} = \sum_{p=1}^{r} \Delta x_{m_p} \left\| \mathbf{a}_{m_p} - \frac{\sum_{p=1}^{r} \Delta x_{m_p} \mathbf{a}_{m_p}}{t} \right\|_{M_l - M_j}^2 .$$

Now we assume $n_j^{\text{old}} > \sum_{p=1}^{r} \Delta x_{m_p}$, which implies $n_j^{\text{new}} > 0$. By the first case in (8), we obtain the difference in SSE as

$$
\begin{aligned}
\text{SSE}^{\text{new}} - \text{SSE}^{\text{old}} &= -\frac{\left\| \sum_{p=1}^{r} \Delta x_{m_p} \mathbf{a}_{m_p} \right\|_{M_l}^2}{t} - \frac{\left\| t\mathbf{c}_j^{\text{old}} - \sum_{p=1}^{r} \Delta x_{m_p} \mathbf{a}_{m_p} \right\|_{M_j}^2}{n_j^{\text{old}} - t} \\
&\quad + \sum_{p=1}^{r} \Delta x_{m_p} \left(\left\| \mathbf{a}_{m_p} \right\|_{M_l}^2 - \left\| \mathbf{a}_{m_p} - \mathbf{c}_j^{\text{old}} \right\|_{M_j}^2 \right) \\
&= -\frac{1}{t} \left\| \sum_{p=1}^{r} \Delta x_{m_p} \mathbf{a}_{m_p} \right\|_{M_l}^2 - \frac{1}{n_j^{\text{old}} - t} \left\| \sum_{p=1}^{r} \Delta x_{m_p} \mathbf{a}_{m_p} \right\|_{M_j}^2 \\
&\quad - \frac{t^2}{n_j^{\text{old}} - t} \left\| \mathbf{c}_j^{\text{old}} \right\|_{M_j}^2 + \frac{2t}{n_j^{\text{old}} - t} \mathbf{c}_j^{\text{old}\,T} M_j \left(\sum_{p=1}^{r} \Delta x_{m_p} \mathbf{a}_{m_p} \right) \\
&\quad + \sum_{p=1}^{r} \Delta x_{m_p} \left(\left\| \mathbf{a}_{m_p} \right\|_{M_l}^2 - \left\| \mathbf{a}_{m_p} \right\|_{M_j}^2 \right) - t \left\| \mathbf{c}_j^{\text{old}} \right\|_{M_j}^2 \\
&\quad + 2 \sum_{p=1}^{r} \Delta x_{m_p} \mathbf{a}_{m_p}^T M_j \mathbf{c}_j^{\text{old}} \\
&= -\left(\sum_{p=1}^{r} \Delta x_{m_p} \mathbf{a}_{m_p} \right)^T \left(\frac{M_l}{t} + \frac{M_j}{n_j^{\text{old}} - t} \right) \left(\sum_{p=1}^{r} \Delta x_{m_p} \mathbf{a}_{m_p} \right) \\
&\quad + \sum_{p=1}^{r} \Delta x_{m_p} \left[\mathbf{a}_{m_p}^T (M_l - M_j) \mathbf{a}_{m_p} \right] \\
&\quad - \frac{t n_j^{\text{old}}}{n_j^{\text{old}} - t} \mathbf{c}_j^{\text{old}\,T} M_j \mathbf{c}_j^{\text{old}} + \frac{2 n_j^{\text{old}}}{n_j^{\text{old}} - t} \mathbf{c}_j^{\text{old}\,T} M_j \left(\sum_{p=1}^{r} \Delta x_{m_p} \mathbf{a}_{m_p} \right) \\
&= -\left(\sum_{p=1}^{r} \Delta x_{m_p} \mathbf{a}_{m_p} \right)^T \left(\frac{M_l - M_j}{t} \right) \left(\sum_{p=1}^{r} \Delta x_{m_p} \mathbf{a}_{m_p} \right) \\
&\quad + \sum_{p=1}^{r} \Delta x_{m_p} \left[\mathbf{a}_{m_p}^T (M_l - M_j) \mathbf{a}_{m_p} \right] \\
&\quad - \frac{n_j^{\text{old}} t}{n_j^{\text{old}} - t} \left\| \frac{\sum_{p=1}^{r} \Delta x_{m_p} \mathbf{a}_{m_p}}{t} - \mathbf{c}_j^{\text{old}} \right\|_{M_j}^2 \\
&= \sum_{p=1}^{r} \Delta x_{m_p} \left\| \mathbf{a}_{m_p} - \frac{\sum_{p=1}^{r} \Delta x_{m_p} \mathbf{a}_{m_p}}{t} \right\|_{M_l - M_j}^2
\end{aligned}
$$

$$-\frac{n_j^{\text{old}} t}{n_j^{\text{old}} - t}\left\|\frac{\sum_{p=1}^{r} \Delta x_{m_p}\mathbf{a}_{m_p}}{t} - \mathbf{c}_j^{\text{old}}\right\|_{M_j}^2 .$$

We have proved (9). □

Note that the right-hand-side of (9) is the sum of two terms, the first of which is the distorted square-error of the must-link closure $\{\mathbf{a}_{m_p}\}_{p=1}^{r}$ with distortion matrix $M_l - M_j$; so this term may not be zero. This implies that permutations of labels may change SSE if distortion matrices of different clusters are not the same.

Since $\text{SSE}^{\text{new}} - \text{SSE}^{\text{old}}$ may not be negative in (9), we also conclude that if the distance metrics for different clusters are different, some of the clusters may be empty at an optimum. Next, we give some special cases in which no cluster is empty at a local optimum.

Lemma 2 *Assume that either*

1. *there are at least k different singleton-must-link closures; or*
2. *the distortion matrices M_j for different clusters are the same; and there exist at least k must-link closures whose centroids are different from each other.*

Then at a local minimal solution to (3) *or* (1)*, no cluster is empty.*

Proof We first consider solutions to (3). To prove case (1), we will show that an assignment with some empty clusters must not be a local minimal solution to (3) if there are at least k different singleton-must-link closures.

Assume that at a feasible solution to (3), cluster C_l is empty. Since there are at least k different singleton-must-link closures, one of the clusters, say C_j, must include some portions of at least two different singleton-must-link closures; furthermore, at least one of the singleton must-link closures, say $\mathbf{a}_m$, is different from the centroid $\mathbf{c}_j$. By (9), moving $\Delta x_m \in (0, x_{mj}]$ portion of $\mathbf{a}_m$ from C_j to C_l will decrease SSE by $\frac{n_j^{\text{old}}}{n_j^{\text{old}} - \Delta x_m}\,\Delta x_m \left\|\mathbf{a}_m - \mathbf{c}_j^{\text{old}}\right\|_{M_j}^2 > 0$ without violating any feasible constraints. Therefore, a feasible solution with an empty cluster is not a local minimal solution to (3).

The proof for case (2) is similar to that for case (1).

Let C_l be empty in a feasible solution to (3). Since there are at least k must-link closures whose centroids are different from each other, one of the clusters, say C_j, must include some portions of at least two must-link closures whose centroids are different from each other. Therefore, the centroid of at least one must-link closure in C_j, say $\{\mathbf{a}_{m_p}\}_{p=1}^{r}$, is different from the centroid of C_j. Since $\mathbf{a}_{m_1}, \ldots, \mathbf{a}_{m_r}$ are must-linked together, $x_{m_1,j} = \cdots = x_{m_r,j}$ by (1c). By (9), moving $\Delta x_m \in (0, x_{m_1,j}^{\text{old}}]$ portion of $\{\mathbf{a}_{m_p}\}_{p=1}^{r}$ from C_j to C_l will decrease SSE by $\frac{n_j^{\text{old}} r\,\Delta x_m}{n_j^{\text{old}} - r\,\Delta x_m}\left\|\frac{\Delta x_m}{r}\sum_{p=1}^{r}\mathbf{a}_{m_p} - \mathbf{c}_j^{\text{old}}\right\|_M^2 > 0$ without violating any feasible constraints. Therefore, a local minimal solution to (3) must not contain any empty cluster.

Letting $\Delta x_m = 1$ in the above proof for case (1) and case (2), we get the part of the lemma for (1). □

Remark 2 Note that both cases in Lemma 2 include the traditional clustering as a special case, with each must-link closure being singleton and the distance metrics being Euclidean, i.e., M_j's being the identity matrices.

3.3 Local optimum without cannot-links

In this part, we consider semi-supervised clustering with only must-links. We will derive necessary and sufficient conditions for a local optimal solution of (3). We will then give sufficient conditions under which (3) admits only integer local optimal solutions. We will also give local optimality conditions for (1). Finally, we will compare local optimal solutions of (1) and (3).

Lemma 3 *Let* $\{\mathbf{a}_{m_1}, \ldots, \mathbf{a}_{m_r}\}$ *be a must-link closure. Then at a local minimum to* (3) *in the absence of cannot-link constraints* (1d), $x_{m_pj} > 0\,(p = 1, \ldots, r)$ *iff the following conditions hold.*

$$\sum_{p=1}^{r} \left\|\mathbf{a}_{m_p} - \mathbf{c}_j\right\|_{M_j}^2 \leq \sum_{p=1}^{r} \left\|\mathbf{a}_{m_p} - \mathbf{c}_l\right\|_{M_l}^2 \quad (l = 1, \ldots, j-1, j+1, \ldots, k), \tag{11}$$

where the equality can be attained only at $\frac{1}{r}\sum_{p=1}^{r} \mathbf{a}_{m_p} = \mathbf{c}_j = \mathbf{c}_l$.

Proof We first prove the necessity by contradiction.

Assume that (11) is not satisfied at a feasible solution X to (3), i.e.,

(1) $x_{m_pj} > 0\,(p = 1, \ldots, r)$, and there exists $l \neq j$ with $\sum_{p=1}^{r} \left\|\mathbf{a}_{m_p} - \mathbf{c}_l\right\|_{M_l}^2 = \sum_{p=1}^{r} \left\|\mathbf{a}_{m_p} - \mathbf{c}_j\right\|_{M_j}^2$; in addition, $\frac{1}{r}\sum_{p=1}^{r} \mathbf{a}_{m_p} \neq \mathbf{c}_j$ or $\frac{1}{r}\sum_{p=1}^{r} \mathbf{a}_{m_p} \neq \mathbf{c}_l$. Or

(2) $x_{m_pj} > 0\,(p = 1, \ldots, r)$, there exists $l \neq j$ with $\sum_{p=1}^{r} \left\|\mathbf{a}_{m_p} - \mathbf{c}_l\right\|_{M_l}^2 < \sum_{p=1}^{r} \left\|\mathbf{a}_{m_p} - \mathbf{c}_j\right\|_{M_j}^2$; in addition, $\frac{1}{r}\sum_{p=1}^{r} \mathbf{a}_{m_p} = \mathbf{c}_j = \mathbf{c}_l$.

By Corollary 1, moving Δx_m of $\mathbf{a}_{m_p}$ $(p = 1, \ldots, r)$ from C_j to C_l with $\Delta x_m \in (0, x_{m_pj}^{\text{old}}]$, i.e., letting

$$x_{m_pj}^{\text{new}} = x_{m_pj}^{\text{old}} - \Delta x_m, \quad x_{m_pl}^{\text{new}} = x_{m_pl}^{\text{old}} + \Delta\, x_m, \quad (p = 1, \ldots, r),$$

will decrease the total sum of square-error without violating any constraints. Therefore, X is not a local minimal solution.

Now we prove sufficiency.

Let $\tilde{X}$ be a feasible solution to (3) satisfying (11), we need to show that $\text{SSE}(\tilde{X} + \Delta X) \geq \text{SSE}(\tilde{X})$ for $\|\Delta X\|_{\text{F}}$ sufficiently small and $(\tilde{X} + \Delta X)$ feasible to (3).

Because of the must-link constraints (1c), we have for each must-link closure $\{\mathbf{a}_{m_p}\}_{p=1}^{m_r}$ and cluster C_l $(l = 1, \ldots, k)$,

$$x_{m_1l} = \cdots = x_{m_rl} \overset{\text{def}}{=} x_{ml}, \quad \Delta x_{m_1l} = \cdots = \Delta x_{m_rl} \overset{\text{def}}{=} \Delta x_{ml}.$$

Therefore,

$$\sum_{p=1}^{m_r} \Delta x_{m_p l} \|\mathbf{a}_{m_p} - \mathbf{c}_l\|_{M_l}^2 = \Delta x_{ml} \sum_{p=1}^{m_r} \|\mathbf{a}_{m_p} - \mathbf{c}_l\|_{M_l}^2. \tag{12}$$

Define the index set

$$MC_m \overset{\text{def}}{=} \arg\min_l \left(\sum_{p=1}^{m_r} \left\|\mathbf{a}_{m_p} - \mathbf{c}_l\right\|_{M_l}^2 \right).$$

Since X and $X + \Delta X$ satisfy constraints (1b), we also obtain

$$\sum_{l \notin MC_m} \Delta x_{ml} = - \sum_{l \in MC_m} \Delta x_{ml}. \tag{13}$$

The conditions (11) and (1b) imply that $\sum_{l \in MC_m} x_{ml} = 1$ and $x_{ml} = 0\,(l \notin MC_m)$; together with X and ΔX being assignment matrices, we get

$$\sum_{l \in MC_m} \Delta x_{ml} \le 0\,, \quad \Delta x_{ml} \ge 0\,(l \notin MC_m). \tag{14}$$

Fixing an index $m_g \in MC_m$, from (11) and (12), we also have

$$\sum_{l \in MC_m} \sum_{p=1}^{m_r} \Delta x_{ml} \left\|\mathbf{a}_{m_p} - \mathbf{c}_l^{\text{old}}\right\|_{M_l}^2 = \left(\sum_{l \in MC_m} \Delta x_{ml} \right) \left(\sum_{p=1}^{m_r} \left\|\mathbf{a}_{m_p} - \mathbf{c}_{m_g}^{\text{old}}\right\|_{M_{m_g}}^2 \right). \tag{15}$$

The total variation in SSE is the sum of variations in SSE resulting from the variation of memberships of each must-link closures. For each must-link closure $\{\mathbf{a}_{m_p}\}_{p=1}^{m_r}$, we fix an index $m_g \in MC_m$. Let $\text{SSE}_{\{\mathbf{a}_{m_p}\}}(\tilde{X} + \Delta X) - \text{SSE}_{\{\mathbf{a}_{m_p}\}}(\tilde{X})$ denote the change of SSE caused by the membership change in $\{\mathbf{a}_{m_p}\}_{p=1}^{m_r}$. Then, by (6), (12), (13), and (15), we have

$$\begin{aligned} &\underset{\{\mathbf{a}_{m_p}\}}{\text{SSE}}(\tilde{X} + \Delta X) - \underset{\{\mathbf{a}_{m_p}\}}{\text{SSE}}(\tilde{X}) \\ &\quad = \left[\sum_{l \notin MC_m} \Delta x_{ml} \sum_{p=1}^{m_r} \left(\left\|\mathbf{a}_{m_p} - \mathbf{c}_l^{\text{old}}\right\|_{M_l}^2 - \left\|\mathbf{a}_{m_p} - \mathbf{c}_{m_g}^{\text{old}}\right\|_{M_{m_g}}^2 \right) \right] \\ &\qquad - \sum_{l=1}^{k} \frac{\left\| \Delta x_{ml} \sum_{p=1}^{m_r} (\mathbf{a}_{m_p} - \mathbf{c}_l^{\text{old}}) \right\|_{M_l}^2}{n_l^{\text{old}} + m_r\, \Delta x_{ml}}. \end{aligned}$$

When $\sum_{l \notin MC_m} \Delta x_{ml} > 0$, because of (11) and (14), the first summation in the right-hand-side is strictly positive and is of order $\mathcal{O}(\Delta x_{ml})$; the second summation is of order $\mathcal{O}[(\Delta x_{ml})^2]$. Therefore, $\mathrm{SSE}_{\{\mathbf{a}_{m_p}\}}(\tilde{X} + \Delta X) - \mathrm{SSE}_{\{\mathbf{a}_{m_p}\}}(\tilde{X}) > 0$ for sufficiently small ΔX.

If $\sum_{l \notin MC_m} \Delta x_{ml} = 0$, because of (14), we get $\Delta x_{ml} = 0\,(l \notin MC_m)$. If MC_m is a singleton, by (13), we have $\Delta x_{m_g} = 0$; hence $\Delta X = 0$; otherwise, the equality in (11) holds. It follows that $\sum_{p=1}^{m_r}(\mathbf{a}_{m_p} - \mathbf{c}_l^{\text{old}}) = 0$ for $l \in MC_m$. Therefore, $\mathrm{SSE}_{\{\mathbf{a}_{m_p}\}}(\tilde{X} + \Delta X) - \mathrm{SSE}_{\{\mathbf{a}_{m_p}\}}(\tilde{X}) = 0$.

We have proved that $\tilde{X}$ is a local minimal solution to (3). □

In the proof of the lemma we have shown that (3) admits a non-integer local optimal solution only when a must-link closure has more than one closest cluster centroids in the distorted distance. The next lemma gives cases in which the solution of (3) must be integer.

Lemma 4 *Assume that the distortion matrices M_j for different clusters are the same. Also suppose that there exist at least k must-link closures whose centroids are different from each other. Then at a local minimal solution to* (3)*, X is an integer matrix. A must-link closure $\{\mathbf{a}_{m_p}\}_{p=1}^{r}$ is assigned to cluster C_j, i.e. $x_{m_p j} = 1$, iff*

$$\sum_{p=1}^{r} \left\| \mathbf{a}_{m_p} - \mathbf{c}_j \right\|_{M_j}^2 < \sum_{p=1}^{r} \left\| \mathbf{a}_{m_p} - \mathbf{c}_l \right\|_{M_l}^2 \quad (l = 1, \ldots, j-1, j+1, \ldots, k).$$

Proof We only need to show that under the assumptions of the lemma, the equality in (11) is not satisfied at a local minimal solution to (3), since other parts are proved in Lemma 3.

We use contradiction. Assume that at an optimal solution to (3), there existed a must-link closure $\{\mathbf{a}_{m_p}\}$, clusters C_j and C_l, such that

$$\sum_{p=1}^{r} \|\mathbf{a}_{m_p} - \mathbf{c}_j\|_{M_j}^2 = \sum_{p=1}^{r} \|\mathbf{a}_{m_p} - \mathbf{c}_l\|_{M_l}^2.$$

Then by Lemma 3, $\mathbf{c}_j = \mathbf{c}_l$. Since $M_l = M_j$, merging the two clusters C_l and C_j would not change the total SSE. However, after merging, we would have an empty cluster; which contradicts Lemma 2. □

In the traditional clustering, all the distance metrics are the same, and all the must-link closures are singleton. The above lemma applies to this case. We thus conclude that in the traditional clustering, all patterns are assigned to their closest centroid at a local optimal solution to (3); and for any pattern, there is only one cluster centroid closest to it.

Remark 3 Lemma 3 shows how to move from a non-integer solution of (3) to a local integer optimal solution with the same or better SSE: simply assign each must-link closure to one of the cluster from whose centroid the must-link closure has the smallest square-error.

Now we study local minimal solutions to (1).

Lemma 5 *At a local optimal solution to* (1) *where cannot-link constraints* (1d) *are nonexistent, a must-link closure* $\{\mathbf{a}_{m_p}\}_{p=1}^r$ *is assigned to* C_j *iff the following conditions are satisfied: for each* $l \neq j$,

$$\begin{cases} \sum_{p=1}^r \left[\left\| \mathbf{a}_{m_p} - \mathbf{c}_l \right\|_{M_l}^2 - \left\| \mathbf{a}_{m_p} - \mathbf{c}_j \right\|_{M_j}^2 \right] \\ \quad \geq \frac{\left\| r\mathbf{c}_l - \sum_{p=1}^r \mathbf{a}_{m_p} \right\|_{M_l}^2}{n_l + r} + \frac{\left\| r\mathbf{c}_j - \sum_{p=1}^r \mathbf{a}_{m_p} \right\|_{M_j}^2}{n_j - r} & (n_j > r); \\ \sum_{p=1}^r \left[\left\| \mathbf{a}_{m_p} - \mathbf{c}_l \right\|_{M_l}^2 - \left\| \mathbf{a}_{m_p} - \mathbf{c}_j \right\|_{M_j}^2 \right] \\ \quad \geq \frac{\left\| r\mathbf{c}_l - \sum_{p=1}^r \mathbf{a}_{m_p} \right\|_{M_l}^2}{n_l + r} & (n_j = r). \end{cases} \tag{16}$$

Proof From (8), we have that under (16), no re-assignment of a single must-link closure can reduce SSE. □

Remark 4 Note that (16) is stronger than (11). A local minimal solution to (3) is not necessarily a local minimal solution to (1); on the other hand, a local minimal solution to (1) must be a local minimal solution to (3).

To see this, consider the following example. We want to partition 3 one-dimensional patterns $\mathbf{a}_1 = -2,\ \mathbf{a}_2 = 0,\ \mathbf{a}_3 = 3$ into 2 groups, where all the must-link closures are singleton. The optimal clustering is $(\{\mathbf{a}_1, \mathbf{a}_2\},\ \{\mathbf{a}_3\})$. However, the suboptimal assignment $(\{\mathbf{a}_1\},\ \{\mathbf{a}_2, \mathbf{a}_3\})$ also satisfies (11).

3.4 Local optimum with cannot-links

In this part, we consider semi-supervised clustering with both must-links and cannot-links.

For a must-link closure $L_m = \{\mathbf{a}_{m_p}\}_{p=1}^r$, we define an index set

$$F_m \overset{\text{def}}{=} \left\{ i \in \{1, \ldots, k\} \colon \text{No pattern in } C_i \text{ is cannot-linked to } \{\mathbf{a}_{m_p}\}_{p=1}^r \right\}. \tag{17}$$

Note that when cannot-link constraints do not exist, the results of swapping two sets of patterns in two clusters are the same as re-assigning the two sets of patterns sequentially. However, in the presence of cannot-links, swapping two sets of patterns in two clusters may further change a local minimum because F_m may not equal to $\{1, \ldots, k\}$.

In this part, we first describe local optimality conditions without swapping, then consider local optimality conditions with swapping.

3.4.1 Local minimum without swapping

We consider local optimal solutions without swapping here. We modify Lemma 3 and Lemma 5 to include cannot-links.

Lemma 6 *Let $\{\mathbf{a}_{m_p}\}_{p=1}^r$ be a must-link closure. Let F_m be defined in (17). Then at a local minimum to (3), the assignment variables $x_{m_p j} > 0\, (p = 1, \ldots, r)$, iff the following conditions hold.*

$$\sum_{p=1}^{r} \left\|\mathbf{a}_{m_p} - \mathbf{c}_j\right\|_{M_j}^2 \leq \sum_{p=1}^{r} \left\|\mathbf{a}_{m_p} - \mathbf{c}_l\right\|_{M_l}^2 \quad (l \in F_m \backslash \{j\}), \tag{18}$$

where equality is possible only when $\frac{1}{r}\sum_{p=1}^r \mathbf{a}_{m_p} = \mathbf{c}_j = \mathbf{c}_l$.

Further assume that all the distortion matrices equal to M. Then (18) is equivalent to

$$\left\|\frac{\sum_{p=1}^r \mathbf{a}_{m_p}}{r} - \mathbf{c}_j\right\|_M^2 \leq \left\|\frac{\sum_{p=1}^r \mathbf{a}_{m_p}}{r} - \mathbf{c}_l\right\|_M^2 \quad (l \in F_m \backslash \{j\}). \tag{19}$$

Proof The first part of the lemma is stated in Lemma 3, except that we consider F_m instead of all the indices $\{1, \ldots, k\}$ for comparison. Next, we prove the second part. Since $M_j = M_l = M$, (18) is equivalent to

$$-\sum_{p=1}^{r} 2\mathbf{a}_{m_p}^T M\mathbf{c}_j + r\|\mathbf{c}_j\|_M^2 \leq -\sum_{p=1}^{r} 2\mathbf{a}_{m_p}^T M\mathbf{c}_l + r\|\mathbf{c}_l\|_M^2 \quad (l \in F_m \backslash \{j\}).$$

Dividing both sides of the above inequality by r and adding $\left\|\frac{1}{r}\sum_{p=1}^r \mathbf{a}_{m_p}\right\|_M^2$ to both sides, we obtain (19). □

Lemma 7 *Let $\{\mathbf{a}_{m_p}\}_{p=1}^r$ be a must-link closure. Let F_m be defined in (17). Then at a local optimal solution to* (1), *$\{\mathbf{a}_{m_p}\}_{p=1}^r$ is assigned to a cluster C_j iff the following conditions are satisfied: for each $l \neq j$ and $l \in F_m$,*

$$\begin{cases} \sum_{p=1}^r \left[\left\|\mathbf{a}_{m_p} - \mathbf{c}_l\right\|_{M_l}^2 - \left\|\mathbf{a}_{m_p} - \mathbf{c}_j\right\|_{M_j}^2\right] \\ \quad \geq \frac{\left\|r\mathbf{c}_l - \sum_{p=1}^r \mathbf{a}_{m_p}\right\|_{M_l}^2}{n_l + r} + \frac{\left\|r\mathbf{c}_j - \sum_{p=1}^r \mathbf{a}_{m_p}\right\|_{M_j}^2}{n_j - r} & (n_j > r)\,; \\ \sum_{p=1}^r \left[\left\|\mathbf{a}_{m_p} - \mathbf{c}_l\right\|_{M_l}^2 - \left\|\mathbf{a}_{m_p} - \mathbf{c}_j\right\|_{M_j}^2\right] \geq \frac{\left\|r\mathbf{c}_l - \sum_{p=1}^r \mathbf{a}_{m_p}\right\|_{M_l}^2}{n_l + r} & (n_j = r). \end{cases} \tag{20}$$

Further assume that all the distance metrics are the same, i.e. $M_1 = \cdots, M_k = M$. Then (20) is equivalent to

$$\begin{cases} \left\|\frac{\sum_{p=1}^{r}\mathbf{a}_{m_p}}{r}-\mathbf{c}_l\right\|_M^2-\left\|\frac{\sum_{p=1}^{r}\mathbf{a}_{m_p}}{r}-\mathbf{c}_j\right\|_M^2 & \\ \quad\geq\frac{r\left\|\mathbf{c}_l-\frac{\sum_{p=1}^{r}\mathbf{a}_{m_p}}{r}\right\|_M^2}{n_l+r}+\frac{r\left\|\mathbf{c}_j-\frac{\sum_{p=1}^{r}\mathbf{a}_{m_p}}{r}\right\|_M^2}{n_j-r} & (n_j>r); \\ \left\|\frac{\sum_{p=1}^{r}\mathbf{a}_{m_p}}{r}-\mathbf{c}_l\right\|_M^2-\left\|\frac{\sum_{p=1}^{r}\mathbf{a}_{m_p}}{r}-\mathbf{c}_j\right\|_M^2\geq\frac{r\left\|\mathbf{c}_l-\frac{\sum_{p=1}^{r}\mathbf{a}_{m_p}}{r}\right\|_M^2}{n_l+r} & (n_j=r). \end{cases}$$

Proof Similar to the proof for Lemma 6. The condition (20) is equivalent to

$$\begin{cases} \left(-\sum_{p=1}^{r}2\mathbf{a}_{m_p}^T M\mathbf{c}_l+r\|\mathbf{c}_l\|_M^2\right)+\left(\sum_{p=1}^{2}2\mathbf{a}_{m_p}^T M\mathbf{c}_j-r\|\mathbf{c}_j\|_M^2\right) & \\ \quad\geq\frac{r^2\left\|\mathbf{c}_l-\frac{1}{r}\sum_{p=1}^{r}\mathbf{a}_{m_p}\right\|_M^2}{n_l+r}+\frac{r^2\left\|\mathbf{c}_j-\frac{1}{r}\sum_{p=1}^{r}\mathbf{a}_{m_p}\right\|_M^2}{n_j-r} & (n_j>r); \\ \left(-\sum_{p=1}^{r}2\mathbf{a}_{m_p}^T M\mathbf{c}_l+r\|\mathbf{c}_l\|_M^2\right)+\left(\sum_{p=1}^{2}2\mathbf{a}_{m_p}^T M\mathbf{c}_j-r\|\mathbf{c}_j\|_M^2\right) & \\ \quad\geq\frac{r^2\left\|\mathbf{c}_l-\frac{1}{r}\sum_{p=1}^{r}\mathbf{a}_{m_p}\right\|_M^2}{n_l+r} & (n_j=r). \end{cases}$$

Dividing both sides of the above inequalities by r and then adding $\left\|\frac{\sum_{p=1}^{r}\mathbf{a}_{m_p}}{r}\right\|_M^2$ and $-\left\|\frac{\sum_{p=1}^{r}\mathbf{a}_{m_p}}{r}\right\|_M^2$ to the first and second terms of the left-hand-side respectively, we get the lemma. □

3.4.2 *Local minimum with swapping*

Let $\{\mathbf{a}_{m_p}\}_{p=1}^r$ be a must-link closure assigned to cluster C_j. We calculate its cannot-link closures in C_j and C_l, represented as $T_{(m)j}$ and $T_{(m)l}$, as follows:

Cannot-link closure calculation

1. Let $T_{(m)j}^0=\{\mathbf{a}_{m_p}\}_{p=1}^r,\ T_{(m)l}^0=\emptyset,\ q=0.$
2. Let $T_{(m)l}^{q+1}=T_{(m)l}^q\cup$ { must-link closures in C_l that have some patterns cannot-linked to any pattern in $T_{(m)j}^q$ }.
3. Let $T_{(m)j}^{q+1}=T_{(m)j}^q\cup$ { must-link closures in C_j that have some patterns cannot-linked to any pattern in $T_{(m)l}^q$ }.
4. If $T_{(m)_j}^{q+1}=T_{(m)_j}^q$ and $T_{(m)_l}^{q+1}=T_{(m)_l}^q$, output $T_{(m)_j}=T_{(m)_j}^{q+1}$ and $T_{(m)_l}=T_{(m)_l}^{q+1}$, stop; otherwise, let $q\leftarrow q+1$, goto step 2.

One way to re-assign $\{\mathbf{a}_{m_P}\}_{p=1}^r$ from C_l to C_j is to switch $T_{(m)j}$ and $T_{(m)l}$. The lemma below gives conditions under which SSE can be reduced by switching $T_{(m)_j}$ and $T_{(m)_l}$.

Lemma 8 *Re-assigning* $\{\mathbf{a}_{u_p}\}_{p=1}^{w}$ *from* C_j *to* C_l *and* $\{\mathbf{a}_{h_q}\}_{q=1}^{s}$ *from* C_l *to* C_j *changes the* SSE *by*

$$\begin{aligned}\mathrm{SSE}^{new} - \mathrm{SSE}^{old} = & \sum_{p=1}^{w}\left(\|\mathbf{a}_{u_p} - \mathbf{c}_l\|_{M_l}^2 - \|\mathbf{a}_{u_p} - \mathbf{c}_j\|_{M_j}^2\right) + \sum_{q=1}^{s}\left(\|\mathbf{a}_{h_q} - \mathbf{c}_j\|_{M_j}^2\right.\\ & \left. - \|\mathbf{a}_{h_q} - \mathbf{c}_l\|_{M_l}^2\right) - \frac{1}{n_l^{old} + w - s}\left\|(w-s)\mathbf{c}_l^{old} - \sum_{p=1}^{w}\mathbf{a}_{u_p}\right.\\ & \left. + \sum_{q=1}^{s}\mathbf{a}_{h_q}\right\|_{M_l}^2 - \frac{1}{n_j^{old} - w + s}\left\|(w-s)\mathbf{c}_j^{old} - \sum_{p=1}^{w}\mathbf{a}_{u_p}\right.\\ & \left. + \sum_{q=1}^{s}\mathbf{a}_{h_q}\right\|_{M_j}^2, \end{aligned} \tag{21}$$

with either of the last two terms vanishes if $n_l^{old} + w - s = 0$ *or* $n_j^{old} - w + s = 0$ *respectively.*

Proof The results follow from Corollary 1 with $\Delta x_{u_p j} = -1, \Delta x_{u_p l} = 1\,(p = 1, \ldots, w)$, $\Delta x_{h_q l} = -1, \Delta x_{h_q j} = 1\,(q = 1, \ldots, s)$. □

Remark 5 From the analysis in this part we conclude that to calculate the centroid of a cluster, a must-link closure can be replaced by its centroid weighted by its number of patterns; however, to search for a local minimum, a must-link closure cannot be replaced by any single point, including its centroid.

4 Concavity cuts

By Lemma 1, (3) is a concave program. Therefore, we can apply concave optimization techniques to it. We adapt Tuy's cutting algorithm (Tuy 1964) to (3). A sketch of our algorithm is given in Xia (2007). Here, we give a detailed description of the algorithm.

We will briefly describe Tuy's cuts in the first part of this section for completeness. However, Tuy's cuts can't be applied directly to (3), because its feasible region does not have full dimension. In the second part of this section, we will show how to adapt Tuy's concavity cuts to (3) and prove that this method can find a global minimum of (3) in finite steps. We will also discuss the complexity of our algorithm.

4.1 Tuy's concavity cuts

For self-completeness, we sketch Tuy's cuts (also known as concavity cuts) below; see Horst and Tuy (1993) for details.

Tuy's cutting plane method solves $\min_{\mathbf{v}\in D} f(\mathbf{v})$, where

1. $D \subseteq \mathbb{R}^m$ is a full dimensional polyhedron, i.e. int $D \neq \emptyset$;
2. $f(\mathbf{v})$ is concave; and for any real number α, the level set $\{\mathbf{v} \in \mathbb{R}^m : f(\mathbf{v}) \geq \alpha\}$ is bounded.

Because f is concave, its local minimum is attained at some vertices of D. Let $\mathbf{v}^0$ be a local minimum and a vertex of D. Since D has full dimension, $\mathbf{v}^0$ has m linearly independent edges. Assume that $\mathbf{y}^1 - \mathbf{v}^0, \ldots, \mathbf{y}^m - \mathbf{v}^0$ are linearly independent. Then the cone originating at $\mathbf{v}^0$ generated by the half lines in the directions $\mathbf{y}^i - \mathbf{v}^0$ covers D. Define

$$\gamma = f(\mathbf{v}^0).$$

Let

$$\theta_i \overset{\text{def}}{=} \sup\{t : t \geq 0,\ f\left(\mathbf{v}^0 + t(\mathbf{y}^i - \mathbf{v}^0)\right) \geq \gamma\}, \quad (i = 1, \ldots, m). \tag{22}$$

Let

$$\mathbf{z}^i \overset{\text{def}}{=} \mathbf{v}^0 + \theta_i(\mathbf{y}^i - \mathbf{v}^0), \quad (i = 1, \ldots, m).$$

Note that (1) $\theta_i \geq 1$; so Spx $\overset{\text{def}}{=}$ conv$\{\mathbf{v}^0, \mathbf{z}^1, \ldots, \mathbf{z}^n\}$ contains $\mathbf{v}^0$ and all its neighbor vertices; (2) the larger θ_i, the larger Spx.

Because f is concave, any point in the simplex Spx has objective value no less than γ. Therefore, to find whether there is any solution with objective value less than $\mathbf{v}^0$, one only needs to search in $D \backslash$Spx. Let

$$U = [\mathbf{y}^1 - \mathbf{v}^0, \ldots, \mathbf{y}^n - \mathbf{v}^0], \quad \pi = \mathbf{e}^T \text{Diag}\left(\frac{1}{\theta_1}, \ldots, \frac{1}{\theta_n}\right) U^{-1}. \tag{23}$$

Then the inequality

$$\pi(\mathbf{v} - \mathbf{v}^0) > 1 \tag{24}$$

excludes Spx. In other words, (24) provides a γ-valid cut for (f, D), i.e., any $\mathbf{v} \in D$ having objective value less than γ must satisfy (24). Therefore, if (24) does not intersect with D, $\mathbf{v}^0$ must be a global minimum. Below is Tuy's original pure concave cutting algorithm based on the above idea.

Tuy's Cutting Algorithm (Algorithm V.1., Chapter V, Horst and Tuy 1993).

Initialization Find a vertex $\mathbf{v}^0$ of D which is a local minimal solution of $f(\mathbf{v})$. Set $\gamma = f(\mathbf{v}^0)$, $D_0 = D$.
Iteration $\mathbf{i} = \mathbf{0}, \mathbf{1}, \mathbf{2}, \ldots$

1. At $\mathbf{v}^i$ construct a γ-valid cut π^i for (f, D_i).

2. Solve the linear program (LP)

$$\max\ \pi^i(\mathbf{v} - \mathbf{v}^i) \quad \text{s.t. } \mathbf{v} \in D_i. \tag{25}$$

Let ω^i be a basic optimal solution of this LP. If $\pi^i(\omega^i - \mathbf{v}^i) \leq 1$, then stop: $\mathbf{v}^0$ is a global minimum; otherwise, go to step 3.

3. Let $D_{i+1} = D_i \cap \{\mathbf{v}: \pi^i(\mathbf{v} - \mathbf{v}^i) \geq 1\}$. Starting from ω^i find a vertex $\mathbf{v}^{i+1}$ of D_{i+1} which is a local minimum of $f(\mathbf{v})$ over D_{i+1}. If $f(\mathbf{v}^{i+1}) \geq \gamma$, then go to iteration $i+1$. Otherwise, set $\gamma \leftarrow f(\mathbf{v}^{i+1})$, $\mathbf{v}^0 \leftarrow \mathbf{v}^{i+1}$, $D_0 \leftarrow D_{i+1}$, and go to iteration 0.

Theorem 1 (Theorem V.2 Horst and Tuy 1993) *If the sequence* $\{\pi^i\}$ *is bounded, the above cutting algorithm is finite.*

Figure 1 gives an illustration of Tuy's concavity cuts. In the figure, we use $\mathbf{x}$ to represent the variable. The polytope is the feasible region in a two-dimensional space. The dotted curve is the level set $\{\mathbf{x} \in \mathbb{R}^2: f(\mathbf{x}) = f(\mathbf{x}^*) = \gamma\}$. And $\mathbf{x}^*$ is a local optimal solution of $f(\mathbf{x})$. The vertex $\mathbf{x}^*$ has two adjacent vertices: $\mathbf{x}_1$ and $\mathbf{x}_2$. The cone generated by $\mathbf{x}^*$, $\mathbf{x}_1$, and $\mathbf{x}_2$ covers the whole feasible region. The shadowed region is the part cut off by a Tuy's concavity cut.

4.2 Concavity cuts for semi-supervised clustering

Note that (3) does not have full dimension, which means that it does not satisfy the assumptions of Tuy's concavity cuts. In this section, we will show how to adapt Tuy's algorithm to semi-supervised clustering. We will first give our procedure of finding a local minimum. Our algorithm searches for a local minimum of (1), since it is stronger than that of (3) by Remark 4. Then, we will describe how we construct the concavity cuts. Finally, we will prove the finite convergence of our algorithm and discuss its complexity.

Based on Remark 5, to reduce the number of variables, we consider the following equivalent form of (3).

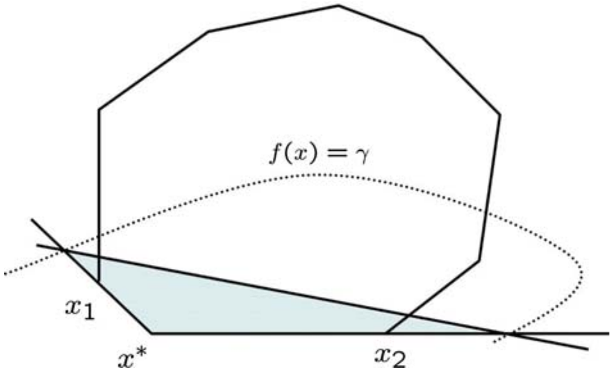

Fig. 1 Tuy's Cut (adapted from Horst and Tuy (1993))

Let L_i $(i = 1, \ldots, N)$ represent the must-link closures, i.e., $\cap_{i=1}^N L_i = \emptyset$, $\cup_{i=1}^N L_i = \{\mathbf{a}_1, \ldots, \mathbf{a}_n\}$, and all the patterns in L_i are must-linked together. Let r_i denote the number of patterns in L_i. Note that $r_i = 1$ means that L_i has only a singleton.

$$\begin{array}{ll}\min_{y_{ij}} & \sum_{j=1}^k \sum_{i=1}^N y_{ij} \sum_{l\in L_i} \left\| \mathbf{a}_l - \frac{\sum_{s=1}^N y_{sj} \sum_{t\in L_s} \mathbf{a}_l}{\sum_{s=1}^N y_{sj} r_s} \right\|_{M_j}^2 \\ \text{s.t.} & \sum_{j=1}^k y_{ij} = 1 \quad (i = 1, \ldots, N) \\ & y_{pj} + y_{qj} \le 1 \quad (\mathbf{a}_p \text{ and } \mathbf{a}_q \text{ cannot-linked; } j = 1, \ldots, k) \\ & y_{ij} \ge 0 \quad (i = 1, \ldots, N;\ j = 1, \ldots, k). \end{array} \tag{26}$$

4.2.1 *Finding a local minimum*

To find a local minimum of (1), we use pivot: moving from one vertex of the feasible domain to an adjacent one that has the least SSE based on Lemma (5, 7, 8). Note that by the results in Sect. 3, we do not need to calculate the total SSE.

Routine for finding a local minimum At the mth iteration $(m = 0, 1, \ldots)$, let D_m denote the feasible region. Do **Loop** until Lemma 7 and Lemma 8 are satisfied.
Loop For $l = 1, \ldots, N$:

1. Assume that the must-link closure $L_l = \{\mathbf{a}_{l_1}, \ldots, \mathbf{a}_{l_{r_l}}\}$ is assigned to cluster C_j.

   Let $s_l = \min_{\substack{q\ne j \\ q\in D_m \\ q\in F_l}} \sum_{p=1}^{r_l} \left\| \mathbf{a}_{l_p} - \mathbf{c}_q \right\|_{M_q}^2 - \frac{\left\| r_l \mathbf{c}_q - \sum_{p=1}^{r_l} \mathbf{a}_{l_p} \right\|_{M_q}^2}{n_q + r_l}$. If

   $$s_l < \begin{cases} \sum_{p=1}^{r_l} \left\| \mathbf{a}_{l_p} - \mathbf{c}_j \right\|_{M_j}^2 + \frac{\left\| r_l \mathbf{c}_j - \sum_{p=1}^{r_l} \mathbf{a}_{l_p} \right\|_{M_j}^2}{n_j - r_l} & n_j > r_l \\ \sum_{p=1}^{r_l} \left\| \mathbf{a}_{l_p} - \mathbf{c}_j \right\|_{M_j}^2 & n_j = r_l \end{cases},$$

   move L_l to any cluster in argmin s_l and update SSE according to Corollary 1.
2. Calculate cannot-link closures of L_l for each $i = 1, \ldots, k$ based on the procedure on p. 20. Swapping according to Lemma 8, until no swapping can reduce SSE.

Remark 6 Note that in the above routine, we only search among vertices. The number of vertices of the feasible region of (1) is finite. And the objective value SSE is strictly decreased after each pivot. Hence, the number of steps for finding a local minimum is finite.

4.2.2 *Construction concavity cuts*

In this part, we show how to construct the concavity cut (23). At the mth iteration, let $Y^0 \in \mathbb{R}^{N\times k}$ be a vertex of D_m and a local optimal solution to (26). Let γ be the smallest SSE obtained from the previous iterations. Next, we will give the formulations of U and θ_i in (23).

(1) *Adjacent vertices*. The feasible region of (3) does not have full dimension: each vertex is adjacent to at most $N \times (k-1)$ other vertices. In order to form U in (23) so that U is invertible, we add artificial vertices outside D_m so that Y^0 has $N \times k$ adjacent vertices.

Let $E_{i,j}$ denote the matrix whose (i, j) entry is 1, the other entries are 0. Let $E_{(i,\cdot)}$ denote the matrix whose entries are all 0 except the ith row being a vector of all 1's. For $l = 1, \ldots, N$, assume that must-link $L_l = \{\mathbf{a}_{l_i}\}_{i=1}^{r_l}$ is assigned to cluster C_{l_j}. Let $Z^{l,i}$ $(i = 1, \ldots, k;\ i \neq l_j)$ denote the assignment matrix different from Y^0 only in the assignment of L_l: L_l is assigned to cluster C_i in $Z^{l,i}$ instead of to cluster C_{l_j} in Y^0. Fix any $1 \leq l_p \leq k$, $l_p \neq l_j$. Let Z^{l,l_j} denote the matrix different from Y^0 only in its (l, l_p) entry being 1 as well, i.e.,

$$Z^{l,i} = \begin{cases} Y^0 - E_{l,l_j} + E_{l,i} & i \neq l_j \\ Y^0 + E_{l,l_p} & i = l_j \end{cases}.$$

Then $Z^{l,i}$ $(l = 1, \ldots, N;\ i = 1, \ldots, k)$ are $N \times k$ adjacent vertices of Y^0, although some of them may not be feasible due to cannot-links. We form the vector $\mathbf{y}^0$ by stacking all the columns of Y^0 together. Similarly, we form the vectors $\mathbf{z}^{l,i}$ by stacking all the columns of $Z^{l,i}$ together correspondingly. It is not hard to see that $U = [\mathbf{z}^{1,1} - \mathbf{y}^0, \ldots, \mathbf{y}^{1,k} - \mathbf{y}^0, \ldots, \mathbf{z}^{N,k} - \mathbf{y}^0]$ has full rank. Let I represent the identity matrix. Then it is straightforward to verify that U^{-1} is a block diagonal matrix with its lth block being

$$I + E_{(l_j,\cdot)} - E_{(l_p,\cdot)} - E_{l_j l_j}.$$

Because Z^{l,l_j} and some Z^{l,l_i} are not feasible to (3), a part of the simplex conv$\{\mathbf{y}^0, \mathbf{z}^{1,1}, \ldots, \mathbf{z}^{1,k}, \ldots, \mathbf{z}^{N,k}\}$ lies outside the feasible region of (26); nevertheless, the concavity cut can exclude some parts of the feasible region of (3).

(2) *The cutting plane*. From the adjacent vertices $Z^{l,i}$ of Y^0, we obtain that the coefficients of (25) are

$$\pi^{l,i} = \begin{cases} \frac{1}{\theta^{l,i}} - \frac{1}{\theta^{l,l_p}} & i \neq l_j \\ -\frac{1}{\theta^{l,l_p}} & i = l_j \end{cases}, \quad \pi \mathbf{y}^0 = -\sum_{l=1}^{n} \frac{1}{\theta^{l,l_p}}.$$

Next, let's determine the θ's.

Let F_l be defined as in (17) for the must-link L_l. Then $\theta^{l,l_j} = \infty$, and $\theta^{l,i} = 1$ for $i \in (\{1, \ldots, K\} \backslash F_l)$. By Corollary 1, for $i \in F_l \backslash \{l_j\}$, $\theta^{l,i}$ is a solution to the problem below.

$$\begin{aligned} \max\ & s \\ \text{s.t.}\ & 0 \leq s \leq \frac{n_{l_j}}{r_l} \\ & \text{SSE}(Y^0) + \mathbf{u} \geq \gamma, \end{aligned} \tag{27}$$

where

$$\mathbf{u} = \begin{cases} -\frac{\left\| s r_l \mathbf{c}_i^{\text{old}} - s \sum_{p=1}^{r_l} \mathbf{a}_{l_p} \right\|_{M_i}^2}{n_i^{\text{old}} + s r_l} - \frac{\left\| s r_l \mathbf{c}_j^{\text{old}} - s \sum_{p=1}^{r_l} \mathbf{a}_{l_p} \right\|_{M_j}^2}{n_j^{\text{old}} - s r_l} & \\ \quad + s \sum_{p=1}^{r_l} \left[\left\| \mathbf{a}_{l_p} - \mathbf{c}_i^{\text{old}} \right\|_{M_i}^2 - \left\| \mathbf{a}_{l_p} - \mathbf{c}_j^{\text{old}} \right\|_{M_j}^2 \right] & (n_j^{\text{new}} > 0,\ n_i^{\text{new}} > 0); \\ -\frac{\left\| s r_l \mathbf{c}_i^{\text{old}} - s \sum_{p=1}^{r_l} \mathbf{a}_{l_p} \right\|_{M_i}^2}{n_i^{\text{old}} + s r_l} & \\ \quad + s \sum_{p=1}^{r_l} \left[\left\| \mathbf{a}_{l_p} - \mathbf{c}_i^{\text{old}} \right\|_{M_i}^2 - \left\| \mathbf{a}_{l_p} - \mathbf{c}_j^{\text{old}} \right\|_{M_j}^2 \right] & (n_j^{\text{new}} = 0,\ n_i^{\text{new}} > 0); \\ 0 & (n_i^{\text{new}} = 0). \end{cases}$$

For the sake of notation simplification, we use j to represent l_j in the formulation for $\mathbf{u}$. The first constraint in (27) keeps the assignment matrix in the feasible region where SSE is a concave function by Lemma 1.

It is not difficult to solve (27). When $n_j^{\text{new}} = 0$ and $n_i^{\text{new}} > 0$, we have $s r_l = n_j$, i.e. $s = \frac{n_j}{r_l}$; when $n_i^{\text{new}} = 0$, we have $s = 0$. From Y^0 being a local minimum, we have $\theta^{l,i} \geq 1$. It is also easy to verify that

$$\text{SSE}(Y^0) + \mathbf{u} - \gamma \text{ is continuous on } [0, \frac{n_{l_j}}{r_l}] \text{ and is nonnegative at } s = 1. \tag{28}$$

When $n_j^{\text{new}} > 0$ and $n_i^{\text{new}} > 0$, multiplying $(n_i^{\text{old}} + s r_l)(n_j^{\text{old}} - s r_l)$ to both sides of $\text{SSE}(Y^0) + \mathbf{u} - \gamma \geq 0$ will reduce it to a cubic polynomial inequality in s, i.e. $b_3 s^3 + b_2 s^2 + b_1 s + b_0 \geq 0$ with

$$\begin{aligned} b_3 &= r_l \left\| r_l \mathbf{c}_i^{\text{old}} - \sum_{p=1}^{r_l} \mathbf{a}_{l_p} \right\|_{M_i}^2 - r_l \left\| r_l \mathbf{c}_j^{\text{old}} - \sum_{p=1}^{r_l} \mathbf{a}_{l_p} \right\|_{M_j}^2 \\ &\quad - r_l^2 \sum_{p=1}^{r_l} \left[\left\| \mathbf{a}_{l_p} - \mathbf{c}_i^{\text{old}} \right\|_{M_i}^2 - \left\| \mathbf{a}_{l_p} - \mathbf{c}_j^{\text{old}} \right\|_{M_j}^2 \right], \\ b_2 &= r_l^2 \left[\gamma - \text{SSE}(Y^0) \right] - n_j^{\text{old}} \left\| r_l \mathbf{c}_i^{\text{old}} - \sum_{p=1}^{r_l} \mathbf{a}_{l_p} \right\|_{M_i}^2 - n_i^{\text{old}} \left\| r_l \mathbf{c}_j^{\text{old}} - \sum_{p=1}^{r_l} \mathbf{a}_{l_p} \right\|_{M_j}^2 \\ &\quad + r_l (n_j^{\text{old}} - n_i^{\text{old}}) \sum_{p=1}^{r_l} \left[\left\| \mathbf{a}_{l_p} - \mathbf{c}_i^{\text{old}} \right\|_{M_i}^2 - \left\| \mathbf{a}_{l_p} - \mathbf{c}_j^{\text{old}} \right\|_{M_j}^2 \right], \\ b_1 &= (n_j^{\text{old}} - n_i^{\text{old}}) r_l \left[\text{SSE}(Y^0) - \gamma \right] + n_i^{\text{old}} n_j^{\text{old}} \sum_{p=1}^{r_l} \left[\left\| \mathbf{a}_{l_p} - \mathbf{c}_i^{\text{old}} \right\|_{M_i}^2 - \left\| \mathbf{a}_{l_p} - \mathbf{c}_j^{\text{old}} \right\|_{M_j}^2 \right], \\ b_0 &= n_i^{\text{old}} n_j^{\text{old}} \left[\text{SSE}(Y^0) - \gamma \right]. \end{aligned}$$

All the coefficients of $\mathrm{SSE}(Y^0)+\mathbf{u}-\gamma$ are real; so it can only have one or three real roots with the possibility of some equal roots if $b_3 \neq 0$. The three roots of the corresponding cubic equation can be obtained by Cardano's formula. The representation of $\theta^{i,l}$ depends on the coefficients $b_3, \ldots, b_0$. We categorize the representations based on the signs of b_3 below.

Case 1 $b_3 > 0$.
When $\mathrm{SSE}(Y^0)+\mathbf{u}-\gamma$ has only one root s_1, by (28), we have $s_1 \leq 1$; so

$$\theta^{l,i} = \frac{n_{l_j}}{r_l}.$$

When $\mathrm{SSE}(Y^0)+\mathbf{u}-\gamma$ has three roots $s_1 \geq s_2 \geq s_3$, by (28), we get either $s_1 \leq 1$ or $s_3 \leq 1 \leq s_2$. Then

$$\theta^{l,i} = \begin{cases} \frac{n_{l_j}}{r_l} & s_1 \leq 1 \\ \min(s_2, \frac{n_{l_j}}{r_l}) & s_3 \leq 1 \leq s_2. \end{cases}$$

Case 2 $b_3 < 0$.
When $\mathrm{SSE}(Y^0)+\mathbf{u}-\gamma$ has only one root s_1, by (28), we obtain $s_1 \geq 1$. Therefore,

$$\theta^{l,i} = \min\left(s_1, \frac{n_{l_j}}{r_l}\right).$$

When $\mathrm{SSE}(Y^0)+\mathbf{u}-\gamma$ has three roots $s_1 \geq s_2 \geq s_3$, by (28), we have either $s_3 \geq 1$ or $s_2 \leq 1 \leq s_1$. Then

$$\theta^{l,i} = \begin{cases} \min(s_3, \frac{n_{l_j}}{r_l}) & s_3 \geq 1 \\ \min(s_1, \frac{n_{l_j}}{r_l}) & s_2 \leq 1 \leq s_1 \ . \end{cases}$$

Case 3 $b_3 = 0$.
By the definition of b_3, we have

$$\left\| r_l \mathbf{c}_i^{\mathrm{old}} - \sum_{p=1}^{r_l} \mathbf{a}_{l_p} \right\|_{M_i}^2 - \left\| r_l \mathbf{c}_j^{\mathrm{old}} - \sum_{p=1}^{r_l} \mathbf{a}_{l_p} \right\|_{M_j}^2 = r_l \sum_{p=1}^{r_l} \left[\left\| \mathbf{a}_{l_p} - \mathbf{c}_i^{\mathrm{old}} \right\|_{M_i}^2 - \left\| \mathbf{a}_{l_p} - \mathbf{c}_j^{\mathrm{old}} \right\|_{M_j}^2 \right].$$

Plugging the above equality into the definition of b_2, we get

$$b_2 = r_l^2 \left[\gamma - \mathrm{SSE}(Y^0) \right] - n_i^{\mathrm{old}} \left\| r_l \mathbf{c}_i^{\mathrm{old}} - \sum_{p=1}^{r_l} \mathbf{a}_{l_p} \right\|_{M_i}^2 - n_j^{\mathrm{old}} \left\| r_l \mathbf{c}_j^{\mathrm{old}} - \sum_{p=1}^{r_l} \mathbf{a}_{l_p} \right\|_{M_j}^2 ,$$

from which we obtain $b_2 \leq 0$.

(i) If $b_2 = 0$, we have

$$c_i^{\text{old}} = c_j^{\text{old}} = \frac{1}{r_l}\sum_{p=1}^{r_l} \mathbf{a}_{l_p}, \quad \text{SSE}(Y^0) = \gamma,$$

which implies $b_0 = 0$. By (28),

$$\theta^{l,i} = \frac{n_{l_j}}{r_l}.$$

(ii) When $b_2 < 0$, along with $b_0 \geq 0$, we have $b_1^2 - 4b_2b_0 \geq 0$. So $\text{SSE}(Y^0) + \mathbf{u} - \gamma$ has two roots; and 1 is between the two roots by (28). Therefore,

$$\theta^{l,i} = \min\left(\frac{-b_1 - \sqrt{b_1^2 - 4b_2b_0}}{2b_2}, \frac{n_{l_j}}{r_l}\right).$$

When L_l is a singleton $\{\mathbf{a}_l\}$, we get closed form solution for the second inequality in (27): $s \leq s^*$ with

$$s^* = -\frac{\left(\text{SSE}(Y^0) - \gamma\right)\left(n_i - n_{l_j}\right) + n_i n_{l_j}\left(\|\mathbf{v}_{l,l_j}\|_{M_{l_j}}^2 - \|\mathbf{v}_{l,i}\|_{M_i}^2\right) - \sqrt{\omega}}{2\left(\text{SSE}(Y^0) - \gamma + n_{l_j}\|\mathbf{v}_{l,l_j}\|_{M_{l_j}}^2 + n_i\|\mathbf{v}_{l,i}\|_{M_i}^2\right)},$$

where

$$\begin{aligned}\omega = &\left[(\text{SSE}(Y^0) - \gamma)(n_i + n_{l_j}) + n_i n_{l_j}(\|\mathbf{v}_{l,l_j}\|_{M_{l_j}}^2 - \|\mathbf{v}_{l,i}\|_{M_i}^2)\right]^2\\ &+ 4(\text{SSE}(Y^0) - \gamma) n_{l_j} n_i (n_{l_j} + n_i)\|\mathbf{v}_{l,i}\|_{M_i}^2.\end{aligned}$$

Observe that when $\text{SSE}(Y^0) = \gamma$, since Y^0 is a local minimum of (1), by Lemma 3, we have $\|\mathbf{v}_{l,i}\|_{M_i} \geq \|\mathbf{v}_{l,l_j}\|_{M_{l_j}}$; hence

$$s^* = \frac{n_{l_j} n_i(\|\mathbf{v}_{l,i}\|_{M_i}^2 - \|\mathbf{v}_{l,l_j}\|_{M_{l_j}}^2)}{n_{l_j}\|\mathbf{v}_{l,l_j}\|_{M_{l_j}}^2 + n_i\|\mathbf{v}_{l,i}\|_{M_i}^2} \leq n_{l_j}.$$

Therefore, in this case, $\theta^{l,i} = s^*$.

The distance from the concavity cut (25) to Y^0 is $\frac{1}{\|\pi\|^2}$. The smaller $\|\pi\|$, the deeper the cut, i.e., the larger the simplex Spx. For fixed $\theta^{l,j}$, the minimal solution to the convex univariate function $\sum_{j=1}^k(\frac{1}{\theta^{l,j}} - x)^2$ is achieved at $x^* = \frac{\sum_{j=1}^k \frac{1}{\theta^{l,j}}}{k}$. To minimize $\|\pi\|$, we choose l_p satisfying

$$l_p \in \arg \min_{j \in \{1,\ldots k\}} \left| x^* - \frac{1}{\theta^{l,j}} \right|.$$

4.2.3 *Finite convergence*

One of the vertex of the simplex Spx cut off by a concavity cut in our algorithm is a local minimum of (1); so each concavity cut eliminates at least one vertex of (1). Since the number of vertices of (1) is finite, the number of concavity cuts can be added is finite. In addition, remark 6 states that only finite pivots are needed to reach a local minimum of (1). Therefore, our method can find a global minimum of (1) in finite steps.

4.2.4 *Complexity*

The k-means problem is NP-hard for $k \geq 2$ (Drineas et al. 2004). For semi-supervised clustering, it is proved in Davidson and Ravi (2005); Klein et al. (2002) that the feasibility problem, i.e. determining whether there is a feasible solution satisfying all cannot-link constraints, is NP-complete. Therefore, we do not attempt to prove the polynomiality of our algorithm.

The main computations of our cutting algorithm involve two steps: (1) the routine for finding a local minimum on p. 24, and (2) solving the linear program (25). In step (1), about $\mathcal{O}(p^i \cdot w^i)$ operations are required, where p^i is the total number of iterations and w^i is the number of nonzeros in the assignment matrix after adding $i-1$ concavity cuts. For step (2), since D_i is the intersection of the cutting planes $\pi^j(\mathbf{y} - \mathbf{y}^j)$ (for $j = 0, \ldots, i$) and the feasible region of (26), the maximal number of nonzeros in the coefficient matrix of the polyhedral representation of D_i is $w^i + 2v^i + iw^i$, where v^i is the number of remaining cannot-links after adding $i-1$ cutting planes. Note that $w^i \leq kN$ and v^i is no more than the original number of cannot-links. If N and k are not too large, an approximate solution within any accuracy to (25) can be obtained in polynomial time in the size of input data by interior-point methods (Nesterov and Nemirovskii 1994). For very large scale instances, Krylov-subspace iterations can be incorporated in the linear programming solver; see, for instance Freund and Jarre (1997). The main computations of each iteration of Krylov-subspace solver are matrix-vector multiplications. And the number of multiplications is linear in the nonzeros of the coefficient matrix of the polyhedral representation of D_i, i.e. $\mathcal{O}\left((i+1)w^i + 2v^i\right)$. Note that D_{i+1} can be represented as D_i plus a concavity cut. And we have a sequence of feasible points obtained from the process of finding a minimum solution in D_i. For this type of linear programs, the dual simplex method (Forrest and Goldfarb 1992) takes only a few iterations to reach an optimal solution of (25). The majority of operations of the simplex method for large-scale sparse problems consist of sparse inner products and additions to a dense vector of length m, where m is the number of variables. The number of operations is linear in m; see for instance Forrest and Tomlin (1992).

We can also incorporate our cutting algorithm into methods for large-scale clustering. For instance, we can use parallel and distributed computation, sample the data, partition the data into disjoint sets and then cluster these sets separately. For high dimensional data, we can apply dimensionality reduction techniques such as PCA (principle component analysis), SVD (singular value decomposition), to map the data to a new space.

5 Numerical examples

We have implemented the above algorithm in Ansi C with the linear programming subroutines of our cutting plane method solved by the CPLEX 91 callable library (www.ilog.com/products/cplex). The code is available on the author's web page.

Our algorithm stops if (1) a global solution is obtained; or (2) more than 21 cuts are added; or (3) no improvement in SSE after 8 consecutive cuts.

Numerical results on traditional clustering show that our cutting method can get a better solution than the k-means algorithm; see Xia and Peng (2005). Test results on semi-supervised clustering (Xia 2007) demonstrate that we can generate local minimal solutions better than solutions obtained by the constrained k-means algorithm; and our algorithm produces better solutions than these local minima. The computation was done on a Toshiba satellite notebook, with Intel Pentium M processor of 1.70 GHz, 496 MB of RAM, Windows XP home edition operating system. The test data sets in Xia (2007) are from the UCI machine learning repository (Murphy and Aha 1994). And our algorithm stopped within four seconds on all these test data sets with up to 1,500 instances.

In this part, we compare our method with some popular semi-supervised clustering algorithms. All the computations here were done on a shared Linux machine with an Intel XEON processor of 3.2 GHz (dual CPU) with HT, 4G RAM. We compare our algorithm with the cop kmeans algorithm (Wagstaff et al. 2001), the cop kmeans + XNJR metric algorithm (the algorithm of Xing et al. 2002), the cop kmeans + RCA metric algorithm (the algorithm of Bar-Hillel et al. 2005), the EM RCA algorithm (Bar-Hillel et al. 2005), the wekaUT algorithm (Bilenko et al. 2004). We coded the cop kmeans algorithm based on (Wagstaff et al. 2001). We downloaded the codes for learning the XNJR and the RCA metrics from their authors's web pages. We downloaded the EM RCA from Bar-Hillel's web page, downloaded the wekaUT from University of Texas's RISC web page (Repository of Information on Semi-supervised Clustering). Unless specified otherwise, data sets in this part are downloaded from the UCI machine learning repository.

In the tables below, 'cop kmeans' means that the results are generated by the cop kmeans algorithm; 'cut' means that the results are obtained by the cutting algorithm with the distance metric being the Euclidean metric. 'cop XNJR' means that the results are generated by the cop kmeans + XNJR metric; 'cut XNJR' means that the results are obtained by our cutting algorithm + XNJR metric; 'cop RCA' means that the results are generated by the cop kmeans + RCA metric; 'cut RCA' means that the results are obtained from our cutting algorithm + RCA metric; 'EM RCA' means that the results are generated by the EM RCA; 'wekaUT' means that the results are generated by the wekaUT.

Since the 'cop kmeans' and the 'cut' have the same objective function, we can compare them based on the objective values of their solutions. For each data set, we ran the 'cop kmeans' and the 'cut' 100 times with random starts. For each run, the starting points of the 'cop kmeans' and the 'cut' are the same. Likewise, we compare the 'cop XNJR' with the 'cut XNJR', the 'cop RCA' with the 'cut RCA', based on the objective values of their solutions, since each pair has the same objective function. For each metric, we ran corresponding algorithms 100 times with random starts. For each run, the starting points of corresponding pair of algorithms are the same.

For each data set, we compare the algorithms with various numbers of must-links and cannot-links. We describe the comparison results in two different tables. In the first table for each data set, the row 'Obj' is the average ratio of the objective value of the solution to the initial objective value of the 100 runs. For instance, the value in the column 'cop kmeans' and the row 'Obj' is

$$\frac{1}{100}\sum_{i=1}^{100}\frac{\text{SSE from the cop k-means algorithm at the } i\text{th run}}{\text{SSE from the initial partition at the } i\text{th run}}.$$

The smaller the value in 'Obj', the better. The row 'CPU' is the average CPU time in seconds of the 100 runs. Because the metrics in the objective functions of the EM RCA and the wekaUT change at each iteration, it is difficult to compare the cutting algorithm with them by objective values. We compare them by some clustering validating indices in the second table. In the second table for each data set, 'AI' represents the adjusted Rand index; 'RI' is the unadjusted Rand index; 'MI' is the Mirkin's index; and 'HI' is the Hubert's index. The higher 'AI', 'RI', 'HI', and the lower 'MI', the better. The MATLAB code for measuring the indices is written by David Corney, downloaded with the CVT-NC package from the MATLAB central (http://www.mathworks.com/matlabcentral/).

The test results from the EM RCA are sensitive to initial parameters. We used the code 'best_params.m' included in the EM RCA package to generate initial parameters. Some times this code didn't converge; we then supplied 0 as the initial center and the identity matrix as the initial covariance matrix.

Balance scale weight & distance database This data set was generated to model psychological experiments reported by Siegler, R. S. (1976): Three Aspects of Cognitive Development, Cognitive Psychology, 8, 481–520. Number of instances: 625; number of attributes: 4; number of classes: 3.

1. Number of instances must-linked: 156; number of instances cannot-linked: 94.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.741386	0.731789	0.431341	0.425626	0.585552	0.579776
CPU	0.012500	0.636300	0.007700	0.446800	0.009300	0.473200

	AR	RI	MI	HI
EM RCA	0.354083	0.692215	0.307785	0.384431
wekaUT	0.196011	0.616297	0.383703	0.232595
cut	0.326255	0.678277	0.321723	0.356554
cut XNJR[a]	0.539150	0.779759	0.220241	0.559518
cut RCA	0.533037	0.776738	0.223262	0.553477

[a] The best of all the clustering results

2. Number of instances must-linked: 218; number of instances cannot-linked: 126.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.752434	0.741034	0.565527	0.556787	0.630256	0.620857
CPU	0.012800	0.543900	0.010400	0.517400	0.011100	0.520500

	AR	RI	MI	HI
EM RCA[a]	0.026413	0.473169	0.526831	0.053662
wekaUT	0.172593	0.605067	0.394933	0.210133
cut	0.430807	0.728328	0.271672	0.456656
cut XNJR[b]	0.551682	0.785631	0.214369	0.571262
cut RCA	0.481501	0.752559	0.247441	0.505118

[a] The code 'best_params.m' does not work

[b] The best of all the clustering results

Synthetic Control Chart Time Series This data set contains 600 examples of control charts synthetically generated by the process in the paper by Alcock and Manolopoulos (1999): Time-Series Similarity Queries Employing a Feature-Based Approach, 7th Hellenic Conference on Informatics. There are six different classes of control charts. Each chart has 60 attributes.

1. Number of charts must-linked: 60; number of charts cannot-linked: 30.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.259643	0.250588	0.027610	0.023449	0.632576	0.629154
CPU	0.958700	15.892900	1.498100	19.082100	1.400300	17.638700

	AR	RI	MI	HI
EM RCA	0.427319	0.840751	0.159249	0.681503
wekaUT	0.591439	0.876138	0.123862	0.752276
cut	0.539254	0.857078	0.142922	0.714157
cut XNJR	0.498063	0.847629	0.152371	0.695259
cut RCA[a]	0.751622	0.930534	0.069466	0.861068

[a] The best of all the clustering results

2. Number of charts must-linked: 90; number of charts cannot-linked: 60.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.264281	0.252792	0.026742	0.024789	0.600107	0.594348
CPU	0.998700	16.480100	1.273900	18.185800	1.881200	24.372200

	AR	RI	MI	HI
EM RCA	0.416484	0.812526	0.187474	0.625053
wekaUT	0.533932	0.846850	0.153150	0.693701
cut	0.538185	0.865982	0.134018	0.731964
cut XNJR	0.443760	0.845170	0.154830	0.690339
cut RCA[a]	0.627171	0.896767	0.103233	0.793534

[a] The best of all the clustering results

Johns Hopkins University Ionosphere Database This radar data was collected by a system in Goose Bay, Labrador. Instances in this data set are complex electromagnetic signals of free electrons in the ionosphere. Number of instances: 351; number of attributes: 34. There are a total of two classes, "good" and "bad", defined by whether there is evidence of certain types of structure in the ionosphere.

1. Number of items must-linked: 52; number of items cannot-linked: 36.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.800101	0.776615	0.456632	0.365377	0.947485	0.930310
CPU	0.035200	0.543100	0.031200	0.331100	0.067600	0.880300

	AR	RI	MI	HI
EM RCA[a]				
wekaUT	0.136504	0.568417	0.431583	0.136834
cut	0.158586	0.579487	0.420513	0.158974
cut XNJR	0.187015	0.593846	0.406154	0.187692
cut RCA[b]	0.499588	0.754986	0.245014	0.509972

[a] The code 'best_params.m' does not work

[b] The best of all the clustering results

2. Number of items must-linked: 122; number of items cannot-linked: 64.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.863829	0.808420	0.595327	0.410939	0.966635	0.950421
CPU	0.034500	0.629300	0.023800	0.201100	0.053700	0.897700

	AR	RI	MI	HI
EM RCA[a]				
wekaUT	0.017393	0.509158	0.490842	0.018315
cut	0.145639	0.572747	0.427253	0.145495
cut XNJR	0.178015	0.588930	0.411070	0.177859
cut RCA[b]	0.673459	0.838502	0.161498	0.677004

[a] The code 'best_params.m' does not work

[b] The best of all the clustering results

Iris Plants Database This famous data set is from Fisher,R.A. (1936): The use of multiple measurements in taxonomic problems, Annual Eugenics, 7, Part II, 179–188. The data set contains 3 types of iris plants. Number of instances: 150 (50 in each of three classes); number of attributes: 4 numeric.

1. Number of items must-linked: 12; number of items cannot-linked: 12.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.122784	0.117699	0.124550	0.118705	0.091468	0.088823
CPU	0.001700	0.129300	0.001800	0.106700	0.001500	0.134600

	AR	RI	MI	HI
EM RCA	0.903874	0.957494	0.042506	0.914989
wekaUT	0.707698	0.869978	0.130022	0.739955
cut	0.744526	0.885906	0.114094	0.771812
cut XNJR	0.744526	0.885906	0.114094	0.771812
cut RCA[a]	0.922177	0.965638	0.034362	0.931275

[a] The best of all the clustering results

2. Number of items must-linked: 16; number of items cannot-linked: 8.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.130293	0.125586	0.054175	0.050117	0.040254	0.031396
CPU	0.001600	0.141800	0.001200	0.122400	0.000600	0.107900

	AR	RI	MI	HI
EM RCA	0.801880	0.912394	0.087606	0.824787
wekaUT	0.765873	0.896644	0.103356	0.793289
cut	0.7570	0.8923	0.1077	0.7845
cut XNJR[a]	0.9222	0.9656	0.0344	0.9313
cut RCA[a]	0.9222	0.9656	0.0344	0.9313

[a] The best of all the clustering results

Letter Image Recognition Data This data set is downloaded from the wekaUT data directory (letter-0.05.arff). The data were generated from randomly distorted fonts of 26 English capital alphabetic letters. Number of instances: 1,000; number of attributes: 16; number of classes: 26.

1. Number of items must-linked: 100; number of items cannot-linked: 500.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.384073	0.368775	0.130848	0.120495	0.379098	0.361797
CPU	1.428200	33.079400	1.683100	35.727200	1.421100	33.983600

	AR	RI	MI	HI
EM RCA[a]	0.000248	0.386887	0.613113	–0.226226
wekaUT	0.132058	0.932264	0.067736	0.864529
cut	0.139454	0.931029	0.068971	0.862058
cut XNJR	0.176416	0.937808	0.062192	0.875616
cut RCA[b]	0.257559	0.939159	0.060841	0.878318

[a] The code ‘best_params.m’ does not work

[b] The best of all the clustering results

2. Number of items must-linked: 150; number of items cannot-linked: 100.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.396869	0.380331	0.176503	0.153552	0.416388	0.398808
CPU	1.368200	29.133700	1.637600	32.498500	1.313200	27.977900

	AR	RI	MI	HI
EM RCA[a]	0.000113	0.131093	0.868907	–0.737814
wekaUT	0.133798	0.931309	0.068691	0.862619
cut	0.166441	0.934012	0.065988	0.868024
cut XNJR	0.147764	0.932440	0.067560	0.864881
cut RCA[b]	0.304402	0.944276	0.055724	0.888553

[a] The code ‘best_params.m’ does not work

[b] The best of all the clustering results

MAGIC gamma telescope data 2004 This data set includes simulated data generated by a Monte Carlo program about the pulses left by high energy gamma particles on the photomultiplier tubes in a ground-based atmospheric Cherenkov gamma telescope, originated from R. K. Bock, Major Atmospheric Gamma Imaging Cherenkov Telescope project (MAGIC) (http://wwwmagic.mppmu.mpg.de). The data set has 19020 instances. Each instance has 10 attributes. There are two classes: gamma (signal) and hadron (background).

1. [a]Number of instances must-linked: 1,902; number of instances cannot-linked: 952.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.764353	0.741350	0.772339	0.748044	0.914651	0.905605
CPU	2.487100	25.138800	1.329600	16.327400	1.537800	16.671500

	AR	RI	MI	HI
EM RCA	0.082045	0.545102	0.454898	0.090205
wekaUT	0.017006	0.509375	0.490625	0.018750
cut	0.051625	0.534667	0.465333	0.069334
cut XNJR	0.086965	0.556589	0.443411	0.113177
cut RCA[b]	0.352629	0.681564	0.318436	0.363128

[a] We stop the cuttting algorithm if (1) a global solution is obtained; or (2) more than 11 cuts are added; or (3) no improvement in SSE after 5 consecutive cuts

[b] The best of all the clustering results

2. Number of items must-linked: 2,854; number of items cannot-linked: 1,902.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.800935	0.758894	0.822353	0.772112	0.929089	0.911230
CPU	2.289600	63.566800	1.613300	86.026800	2.007400	51.418300

	AR	RI	MI	HI
EM RCA	0.111918	0.560841	0.439159	0.121682
wekaUT	0.029971	0.516065	0.483935	0.032130
cut	0.044623	0.527542	0.472458	0.055084
cut XNJR	0.167939	0.599181	0.400819	0.198362
cut RCA[a]	0.366041	0.686926	0.313074	0.373852

[a] The best of all the clustering results

Optical Recognition of Handwritten Digits This data set is from E. Alpaydin, C. Kaynak of the Department of Computer Engineering, Bogazici University, Istanbul Turkey. The data are normalized bitmaps of handwritten digits from 30 people extracted by NIST preprocessing programs. 32×32 bitmaps are divided into nonoverlapping blocks of 4×4 and the number of on pixels are counted in each block. We use the training set of the data set which has 3,823 instances. Each instance has 64 attributes. Each attribute is an integer in the range 0–16. It has 10 classes for digits $0, \ldots, 9$.

For this data set, the XNJR metric generated by its authors' code is complex. As a result, there is no output for 'cop kmeans + XNJR' and 'cutting algorithm + XNJR'.

1. [a]Number of items must-linked: 496; number of items cannot-linked: 306.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.552918	0.546343	[b]	[b]	0.699775	0.691483
CPU	30.978900	215.844500	[b]	[b]	22.208700	169.439300

	AR	RI	MI	HI
EM RCA[c]				
wekaUT	0.603633	0.926090	0.073910	0.852179
cut	0.709085	0.946841	0.053159	0.893681
cut RCA[d]	0.743931	0.950519	0.049481	0.901038

[a] We stop the cuttting algorithm if (1) a global solution is obtained; or (2) more than 11 cuts are added; or (3) no improvement in SSE after 5 consecutive cuts

[b] Complex number

[c] No output

[d] The best of all the clustering results

2. Number of items must-linked: 686; number of items cannot-linked: 420.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.554776	0.547201	[a]	[a]	0.689020	0.677558
CPU	28.024000	428.404800	[a]	[a]	19.070700	314.095100

	AR	RI	MI	HI
EM RCA[b]				
wekaUT	0.490691	0.900861	0.099139	0.801722
cut	0.711620	0.946219	0.053781	0.892439
cut RCA[c]	0.879813	0.978382	0.021618	0.956764

[a] Complex number

[b] No output

[c] The best of all the clustering results

Page Blocks Classification Each observation of the data set consists of some numerical attributes of one block from 54 distinct documents that has been detected by a segmentation process. The aim is to classify all the blocks. There are 5 types of documents: text, horizontal line, picture, vertical line and graphic. Number of instances: 5,473; number of attributes: 10; number of classes: 5.

1. Number of blocks must-linked: 548; number of blocks cannot-linked: 274.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.124673	0.124331	0.088712	0.088484	0.156298	0.140886
CPU	0.917800	15.837600	0.834700	15.885600	0.695000	13.855700

	AR	RI	MI	HI
EM RCA[a]	0.096336	0.404766	0.595234	−0.190469
wekaUT	0.015314	0.414114	0.585886	−0.171773
cut	0.013137	0.567416	0.432584	0.134833
cut XNJR	0.028545	0.583985	0.416015	0.167970
cut RCA[b]	0.591593	0.889803	0.110197	0.779605

[a] The code ‘best_params.m’ does not work

[b] The best of all the clustering results

2. Number of blocks must-linked: 1,204; number of blocks cannot-linked: 820.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.136029	0.134312	0.119725	0.118266	0.187115	0.166168
CPU	1.041000	14.505300	0.983700	15.843300	0.906500	17.130800

	AR	RI	MI	HI
EM RCA	0.094252	0.401777	0.598223	0.196445
wekaUT	0.015314	0.414114	0.585886	0.171773
cut	0.031681	0.533806	0.466194	0.067613
cut XNJR	0.040625	0.544503	0.455497	0.089006
cut RCA[a]	0.590672	0.871905	0.128095	0.743811

[a] The best of all the clustering results

Protein This data set is downloaded from Eric Xing’s web page. It has 116 instances with 20 attributes each. There are 6 classes.

1. Number of items must-linked together: 18; number of items cannot-linked together: 12.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.681749	0.654520	0.175389	0.164417	0.637941	0.607605
CPU	0.017700	0.515400	0.021500	0.458000	0.021000	0.545100

	AR	RI	MI	HI
EM RCA[a]	0.004838	0.242729	0.757271	−0.514543
wekaUT	0.276166	0.773163	0.226837	0.546327
cut	0.312395	0.800750	0.199250	0.601499
cut XNJR[b]	0.318041	0.804648	0.195352	0.609295
cut RCA	0.221767	0.776462	0.223538	0.55292

[a] The code ‘best_params.m’ does not work

[b] The best of all the clustering results

2. Number of items must-linked together: 26; number of items cannot-linked together: 18.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.681749	0.654520	0.175389	0.164417	0.637941	0.607605
CPU	0.017700	0.515400	0.021500	0.458000	0.021000	0.545100

	AR	RI	MI	HI
EM RCA[a]	0.011151	0.276762	0.723238	0.446477
wekaUT	0.183484	0.767616	0.232384	0.535232
cut[b]	0.329930	0.809295	0.190705	0.618591
cut XNJR	0.256252	0.782159	0.217841	0.564318
cut RCA	0.103120	0.735382	0.264618	0.470765

[a] The code ‘best_params.m’ does not work

[b] The best of all the clustering results

Statlog (Landsat Satellite) Data Set The database consists of the multi-spectral values of pixels in 3 × 3 neighborhoods in a satellite image generated from Landsat Multi-Spectral Scanner image date purchased from NASA by the Australian Center for Remote Sensing. We use the training set of the data set, which has 4,435 instances, 36 attributes, and 6 classes,

1. Number of items must-linked together: 444; number of items cannot-linked: 222.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.229574	0.224957	0.069141	0.065167	0.581613	0.578327
CPU	7.948400	136.879700	7.681300	113.289700	8.606100	125.390900

	AR	RI	MI	HI
EM RCA	0.366415	0.801548	0.198452	0.603095
wekaUT	0.378833	0.793589	0.206411	0.587178
cut	0.535568	0.858873	0.141127	0.717746
cut XNJR	0.584852	0.863922	0.136078	0.727843
cut RCA[a]	0.654945	0.883567	0.116433	0.767135

[a] The best of all the clustering results

2. [a]Number of items must-linked together: 666; number of items cannot-linked: 444.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.234156	0.228742	0.040589	0.039941	0.587995	0.583399
CPU	7.853300	57.579300	9.855400	64.118500	6.521000	52.846500

	AR	RI	MI	HI
EM RCA	0.375987	0.804853	0.195147	0.609707
wekaUT	0.345386	0.783161	0.216839	0.566323
cut	0.539893	0.859985	0.140015	0.719971
cut XNJR	0.530561	0.844788	0.155212	0.689577
cut RCA[b]	0.676902	0.891386	0.108614	0.782772

[a] We stop the cuttting algorithm if (1) a global solution is obtained; or (2) more than 11 cuts are added; or (3) no improvement in SSE after 5 consecutive cuts

[b] The best of all the clustering results

Small Soybean Database This is a subset of the data from R.S. Michalski and R.L. Chilausky (1980): Learning by Being Told and Learning from Examples: An Experimental Comparison of the Two Methods of Knowledge Acquisition in the Context of Developing an Expert System for Soybean Disease Diagnosis, International Journal of Policy Analysis and Information Systems, Vol. 4, No. 2.

Number of instances: 47; number of attributes: 35; number of classes: 4.

1. Number of items must-linked together: 4; number of items cannot-linked: 24.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.344980	0.322840	0.049682	0.036741	0.287293	0.269655
CPU	0.010600	0.225600	0.007900	0.215100	0.008900	0.223700

	AR	RI	MI	HI
EM RCA[a]				
wekaUT	0.548885	0.803885	0.196115	0.607771
cut	0.551276	0.833488	0.166512	0.666975
cut XNJR	0.315361	0.744681	0.255319	0.489362
cut RCA[b]	0.581409	0.842738	0.157262	0.685476

[a] No output

[b] The best of all the clustering results

2. Number of items must-linked together: 8; number of items cannot-linked: 4.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.365990	0.341289	0.044405	0.044304	0.246952	0.200295
CPU	0.008300	0.207900	0.010600	0.085700	0.009600	0.108900

	AR	RI	MI	HI
EM RCA[a]				
wekaUT	0.524844	0.817761	0.182239	0.635523
cut	0.625730	0.859389	0.140611	0.71877
cut XNJR	0.409350	0.774283	0.225717	0.548566
cut RCA[b]	1.000000	1.000000	0.000000	1.000000

[a] No output

[b] The best of all the clustering results

Wine Recognition Data This data set is from Forina, M. et al. PARVUS—An Extendible Package for Data Exploration, Classification and Correlation. Each instance in the data set consists of the quantities of 13 constituents found in each of the three types of wines grown in the same region in Italy but derived from three different cultivars. There are 178 instances, each has 13 attributes, and a total of 3 classes.

1. Number of items must-linked together: 44; number of items cannot-linked together: 26.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.170750	0.169710	0.168806	0.167999	0.554292	0.536157
CPU	0.007200	0.236700	0.005800	0.216700	0.008100	0.295900

	AR	RI	MI	HI
EM RCA	0.607339	0.825176	0.174824	0.650352
wekaUT	0.699030	0.865486	0.134514	0.730972
cut	0.447852	0.753634	0.246366	0.507268
cut XNJR	0.444582	0.752301	0.247699	0.504602
cut RCA[a]	0.739974	0.884149	0.115851	0.768298

[a] The best of all the clustering results

2. Number of items must-linked together: 72; number of items cannot-linked together: 44.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.217126	0.194179	0.214235	0.193356	0.502998	0.472176
CPU	0.004900	0.228600	0.005200	0.187300	0.005400	0.181300

	AR	RI	MI	HI
EM RCA	0.734012	0.881673	0.118327	0.763347
wekaUT	0.441603	0.750651	0.249349	0.501301
cut	0.452178	0.754967	0.245033	0.509935
cut XNJR	0.474542	0.764235	0.235765	0.528471
cut RCA[a]	0.948716	0.977084	0.022916	0.954167

[a] The best of all the clustering results

Protein Localization Sites (Yeast) This data set is from Kenta Nakai (http://psort.ims.u-tokyo.ac.jp/). Each instance is the information of an amino acid sequence and its source origin. The classes are the Cellular Localization Sites of Proteins. Number of instances: 1,484, number of attributes: 9, and number of classes: 10.

1. Number of items must-linked: 296; number of items cannot-linked: 178.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.433904	0.419116	0.287329	0.272538	0.393354	0.379990
CPU	0.393600	10.847700	0.388100	10.748400	0.362800	12.091300

	AR	RI	MI	HI
EM RCA[a]	0.012431	0.295460	0.704540	0.409080
wekaUT	0.166856	0.745925	0.254075	0.49184
cut	0.159939	0.752909	0.247091	0.505817
cut XNJR	0.145082	0.751556	0.248444	0.503113
cut RCA[b]	0.173515	0.750714	0.249286	0.501428

[a] The code ‘best_params.m’ does not work

[b] The best of all the clustering results

2. Number of items must-linked: 520; number of items cannot-linked: 296.

	cop kmeans	cut	cop XNJR	cut XNJR	cop RCA	cut RCA
Obj	0.462089	0.448656	0.246241	0.240499	0.427164	0.408590
CPU	0.325000	11.546800	0.363000	11.932900	0.359400	12.136200

	AR	RI	MI	HI
EM RCA[a]	0.010443	0.337053	0.662947	0.325895
wekaUT	0.161922	0.739867	0.260133	0.479733
cut	0.177142	0.750328	0.249672	0.500655
cut XNJR	0.156789	0.755752	0.244248	0.511504
cut RCA[b]	0.195498	0.758225	0.241775	0.516450

[a] The code ‘best_params.m’ does not work

[b] The best of all the clustering results

Conclusion For all the data sets we have tested, out ‘cutting algorithm’ generates better solutions than the ‘cop kmeans’ does; the ‘cut + metric’ generates better solutions than the ‘cop kmeans + metric’ does, where the metric is the RCA or the XNJR. And the ‘cut’, or the ‘cut RCA’, or the ‘cut XNJR’ generates the best solutions among all the algorithms tested here.

Acknowledgements This research was carried out when the author was a postdoctoral research fellow at the Institute of Statistical Mathematics, Japan. I thank the support from JSPS (Japan Society for the Promotion of Science). The paper was revised when I was a research fellow with the Center of Operations Research and Econometrics (CORE), Université catholique de Louvain, Belgium. I also thank the support from CORE. Finally, I would like to thank the helpful suggestions and comments of two anonymous referees which help improve the paper.

References

Bar-Hillel A, Hertz T, Shental N, Weinshall D (2005) Learning a mahalanobis metric from equivalence constraints. Journal of Machine Learning Research 6:937–965

Basu S, Banerjee A, Mooney RJ (2003) Semi-supervised clustering by seeding. In: Sammut C, Hoffmann AG (eds) ICML: Machine learning, proceedings of the nineteenth international conference (ICML 2002). University of New South Wales, Sydney, Australia, Morgan Kaufmann, July 8–12, 2002, pp 27–34

Basu S, Bilenko M, Mooney RJ (2004) A probabilistic framework for semi-supervised clustering. In: Kim W, Kohavi R, Gehrke J, DuMouchel W (eds) KDD: proceedings of the tenth ACM SIGKDD international conference on knowledge discovery and data mining. Seattle, Washington, USA, ACM, August 22–25, pp 59–68

Bilenko M, Basu S, Mooney RJ (2004) Integrating constraints and metric learning in semi-supervised clustering. In: ICML'04: proceedings of the twenty-first international conference on machine learning. ACM Press, New York, NY, USA, p 11

Chang H, Yeung D-Y (2006) Locally linear metric adaptation with application to semi-supervised clustering and image retrieval. Pattern Recognition 39(7):1253–1264

Cohn D, Caruana R, McCallum A (2003) Semi-supervised clustering with user feedback. Technical report, Cornell University

Davidson I, Ravi SS (2005) Clustering with constraints: feasibility issues and the k-means algorithm. In: Proceedings of the 2005 SIAM international conference on data mining

Demiriz A, Bennett KP, Embrechts MJ (1999) Semi-supervised clustering using genetic algorithms. In: Proceedings of ANNIE'99 (Artificial Neural Networks in Engineering). R.P.I. Math Report No. 9901, Rensselaer Polytechnic Institute, Troy, New York

Drineas P, Frieze AM, Kannan R, Vempala S, Vinay V (2004) Clustering large graphs via the singular value decomposition. Mach Learn 56(1–3):9–33

Forrest JJ, Goldfarb D (1992) Steepest-edge simplex algorithms for linear programming. Math Programming 57(3, Ser. A):341–374

Forrest JJH, Tomlin JA (1992) Implementing the simplex method for the optimization subroutine library. IBM Syst J 31(1):11–25

Freund RW, Jarre F (1997) A QMR-based interior-point algorithm for solving linear programs. Math Program 76(1, Ser. B):183–210. Interior point methods in theory and practice (Iowa City, IA, 1994)

Gao J, Tan P-N, Cheng H (2006) Semi-supervised clustering with partial background information. In: Ghosh J, Lambert D, Skillicorn DB, Srivastava J (eds) SDM'06: proceedings of the sixth SIAM international conference on data mining. SIAM, Bethesda, MD, USA, April 20–22

Gordon AD (1996) A survey of constrained classification. Comput Stat Data Anal 21(1):17–29

Horst R, Tuy H (1993) Global optimization. Springer-Verlag, Berlin

Jain AK, Mallapragada PK, Law M (2006) Bayesian feedback in data clustering. In: ICPR. IEEE Computer Society, pp 374–378

Klein D, Kamvar SD, Manning CD (2002) From instance-level constraints to space-level constraints: making the most of prior knowledge in data clustering. In: Proceedings of the international conference on machine learning

Lange T, Law MHC, Jain AK, Buhmann JM (2005) Learning with constrained and unlabelled data. In: CVPR'05: proceedings of the 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05), vol 1. IEEE Computer Society, Washington, DC, USA, pp 731–738

Murphy PM, Aha DW (1994) UCI repository of machine learning databases. Technical report, University of California, Department of Information and Computer Science, Irvine, CA. http://www.ics.uci.edu/~mlearn/MLRepository.html

Nemhauser GL, Wolsey LA (1988) Integer and combinatorial optimization Wiley-Interscience Series in Discrete Mathematics and Optimization. Wiley, A Wiley-Interscience Publication, New York

Nesterov Y (2004) Introductory lectures on convex optimization, volume 87 of Applied Optimization. Kluwer Academic Publishers, Boston, MA (A basic course)

Nesterov Y, Nemirovskii A (1994) Interior-point polynomial algorithms in convex programming, volume 13 of SIAM studies in applied mathematics. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA

Shental N, Bar-Hillel A, Hertz T, Weinshall D (2003) Computing Gaussian mixture models with EM using equivalence constraints. In: Thrun S, Saul LK, Schölkopf B (eds) NIPS. MIT Press

Tuy H (1964) Concave programming under linear constraints. Soviet Math 5:1437–1440

Wagstaff K, Cardie C (2000) Clustering with instance-level constraints. In: Proceedings of the 17th international conference on machine learning. Morgan Kaufmann, San Francisco, CA, pp 1103–1110

Wagstaff K, Cardie C, Rogers S, Schroedl S (2001) Constrained k-means clustering with background knowledge. In: ICML'01: proceedings of the eighteenth international conference on machine learning. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp 577–584

Xia Y (2007) Constrained clustering via concavity cuts. In: CPAIOR'07: proceedings of the fourth international conference on integration of AI and OR techniques in constraint programming for combinatorial optimization problems, pp 318–331. http://dx.doi.org/10.1007/978-3-540-72397-4_23, http://dblp.uni-trier.de

Xia Y, Peng J (2005) A cutting algorithm for the minimum sum-of-squared error clustering. In: Proceedings of the fifth SIAM international conference on data mining, pp 150–160

Xing EP, Ng AY, Jordan MI, Russell S (2002) Distance metric learning with application to clustering with side-information. In: Thrun S, Becker S, Obermayer K (eds) Advances in neural information processing systems, vol 15. MIT Press, Cambridge, MA, pp 505–512