REGULAR PAPER

Non-negative matrix factorization for semi-supervised data clustering

Yanhua Chen · Manjeet Rege · Ming Dong · Jing Hua

Abstract   Traditional clustering algorithms are inapplicable to many real-world problems where limited knowledge from domain experts is available. Incorporating the domain knowledge can guide a clustering algorithm, consequently improving the quality of clustering. In this paper, we propose SS-NMF: a semi-supervised non-negative matrix factorization framework for data clustering. In SS-NMF, users are able to provide supervision for clustering in terms of pairwise constraints on a few data objects specifying whether they "must" or "cannot" be clustered together. Through an iterative algorithm, we perform symmetric tri-factorization of the data similarity matrix to infer the clusters. Theoretically, we show the correctness and convergence of SS-NMF. Moveover, we show that SS-NMF provides a general framework for semi-supervised clustering. Existing approaches can be considered as special cases of it. Through extensive experiments conducted on publicly available datasets, we demonstrate the superior performance of SS-NMF for clustering.

Keywords   Non-negative matrix factorization · Semi-supervised clustering · Pairwise constraint

1 Introduction

Clustering or unsupervised learning is a generic name for a variety of procedures designed to find natural groupings, or clusters, in multidimensional data, based on measured or perceived similarities among the patterns [13,19,23,27]. The purpose of clustering is to extract useful information from unlabeled data. Applications of data clustering are found in many fields, such as information discovery, text mining, web analysis, image grouping, medical diagnosis, and bioinformatics. In general, the clustering algorithms can be categorized into two popular techniques: hierarchical clustering and partitional clustering.

Y. Chen · M. Rege · M. Dong (✉) · J. Hua
Department of Computer Science, Wayne State University,
Detroit MI, USA
e-mail: mdong@wayne.edu

Hierarchical clustering [33,35] aims to obtain a hierarchy of clusters, called dendrogram, that shows how the clusters are related to each other. The clustering result can be obtained by cutting the dendrogram at a desired level. Amongst these the agglomerative methods create the cluster dendrogram in a bottom-up fashion, starting with each data object (or sample) in its own cluster and merging clusters successively according to a similarity measure till a convergence criterion is reached [32,55]. Divisive hierarchical clustering methods create the cluster dendrogram in a top-down divisive fashion, where all the data objects initially belong to a single cluster to begin with. This cluster is then split successively according to some measurement till a convergence criterion is reached [6,12,17].

Partitioning methods divide the data in a given number of clusters directly and are typically used more frequently in real-world applications. These methods attempt to obtain a partition which minimizes the within-cluster scatter or maximizes the between-cluster scatter. Amongst these density-based algorithms model clusters as dense regions, use different heuristics to find arbitrary-shaped high-density regions in the input data space and group points accordingly. Well-known methods include Denclue, which tries to analytically model the overall density around a data object [24], and WaveCluster, which uses wavelet-transform to find high-density regions [48]. Density-based methods typically have difficulty scaling up to very high dimensional data. Mixture-based methods assume that the data objects in a cluster are drawn from one of several distributions (usually Gaussian) and attempt to estimate the parameters of all these distributions. The introduction of the expectation maximization (EM) algorithm in Dempster et al. [9] was an important step in solving the parameter estimation problem. Mixture-resolving methods have a high computational complexity and make rather strong assumptions regarding the distribution of the data. Most mixture-based methods view each cluster as a single simple distribution and strongly restrict the shape of the clusters. For example, the k-means algorithm [44] assumes every cluster has a compact shape. Since the actual underlying distribution of the data can be different, these methods are susceptible to their a priori assumptions.

Clustering based on spectral graph partitioning has emerged as a popular method over the years with applications across various domains [8,14–16,20,49]. These methods model the data objects as vertices of a weighted graph with edge weights representing the similarity between two data objects. Clustering is then obtained by "cutting" the graph vertices into different partitions. Partitioning of the graph is obtained by solving an eigenvalue problem where the clustering is inferred from the top eigenvectors. Although, all of the above methods have contributed greatly to the problem of data clustering, they are completely unsupervised. That is, they are inapplicable to many real-world problems where limited knowledge from domain experts is available. Incorporating the domain knowledge can guide a clustering algorithm, consequently improving the quality of clustering.

Semi-supervised clustering uses class labels or pairwise constraints on data objects to aid unsupervised clustering [3,4,31,34,36,51,52]. It can group data using the categories of the initial labeled data as well as unlabeled data in order to modify the existing set of categories which reflect the whole regularities in the data. Two sources of information are usually available to a semi-supervised clustering method: the similarity distance measurement in unsupervised clustering and class labels or some pairwise constraints. For semi-supervised clustering to be profitable, these two sources of information should not completely contradict each other. Existing methods for semi-supervised clustering based on source information generally fall into two categories: *distance-based* and *constraint-based* methods. In distance-based approaches, an existing clustering algorithm that uses a distance measure is employed; however, the distance measure is first trained to satisfy the labels or constraints in the supervised data [31,52]. In constraint-based approaches, the clustering algorithm itself is

modified so that the available labels or constraints are used to bias the search for an appropriate clustering of the data [1,7]. Recent research in semi-supervised clustering tends to combine the constraint-based with distance-based approaches.

In this paper, we propose a non-negative matrix factorization (NMF) [37,38] based framework to incorporate prior knowledge into data clustering. Under the proposed Semi-Supervised NMF (SS-NMF) methodology, user is able to provide pairwise constraints on a few data objects specifying whether they "must" or "cannot" be clustered together. We derive an iterative algorithm to perform symmetric non-negative tri-factorization of the data similarity matrix. The correctness and convergence of the algorithm are proved by showing that the solution satisfied the KKT optimality and the algorithm is guaranteed to converge. We also prove that SS-NMF is a general and unified framework for semi-supervised clustering by establishing the relationship between SS-NMF and other existing semi-supervised clustering algorithms. Experiments performed on various publicly available datasets demonstrate the superior performance of the proposed work.

The rest of the paper is organized as follows. We review related work on semi-supervised data clustering in Sect. 2. The proposed SS-NMF algorithm for data clustering and our theoretical results are presented in Sect. 3. Experimental results appear in Sect. 4. Finally, we conclude in Sect. 5.

2 Related work

In this section, we provide a review of related works on using user provided information to improve data clustering. We first discuss some algorithms in which prior knowledge is in the form of labeled data. Next, we describe other algorithms for which pairwise constraints are required to be known a priori.

SS-constrained-Kmeans [51] and SS-seeded-Kmeans [3] are the two well-known algorithms in semi-supervised clustering with labels. The SS-constrained-Kmeans seeds the k-means algorithm with the given labeled data and keeps that labeling unchanged throughout the algorithm. Moreover, it is appropriate when the initial seed labeling is noise-free, or if the user does not want the labels of the seed data to change. On the other hand, the SS-seeded-Kmeans algorithm changes the given labeling of the seed data during the course of the algorithm. Also, it is applicable in the presence of noisy seeds, since it does not enforce the seed labels to remain unchanged during the clustering iterations and can therefore abandon noisy seed labels after the initialization step. Semi-supervised clustering with labels has been successfully applied to the problem of document clustering. Hotho et al. [25] proposed incorporating background knowledge into document clustering by enriching the text features using WordNet.[1] In Jones et al. [30], some words per class and a class hierarchy were sought from the user in order to generate labels and build an initial text classifier for the class. A similar technique was proposed in Liu et al. [41], where the user is made to select interesting words from automatically selected representative words for each class of documents. These user identified words were then used to re-train the text classifier. Active learning approaches have also found applications in semi-supervised clustering. Godbole et al. [18] has proposed to convert a user recommended feature into a mini-document which is then used to train an SVM classifier. This approach has been extended by Raghavan et al. [47] which adjusts SVM weights of the key features to a predefined value in binary classification tasks. Recently, Huang and Mitchell [26] presented a probabilistic generative model to incorporate

---

[1] http://wordnet.princeton.edu.

extended feedback that allows the user and the algorithm to jointly arrive at coherent clusters that capture the categories of interest to the user. Nigam et al. [46], Blum and Mitchell [5] and Joachims [29] proposed methods where the user provided class labels a priori to some of the documents. These algorithms use the labeled data to generate seed clusters that initialize a clustering algorithm, and use constraints generated from the labeled data to guide the clustering process. Proper seeding biases clustering towards a good region of the search space, while simultaneously producing a clustering similar to the specified labels.

However, in certain applications, supervision in the form of class labels may be unavailable. For example, complete class labels may be unknown in the context of clustering for speaker identification in a conversation [2], or clustering GPS data for lane-finding [51]. In some domains, pairwise constraints occur naturally, e.g., the database of interacting proteins (DIP) dataset contains information about proteins co-occurring in processes, which can be viewed as *must-link* constraints during clustering. Similarly, for document clustering, user knowledge about which few documents are related or unrelated can be incorporated to improve the clustering results. Moreover, it is easier for a user to provide feedback in the form of pairwise constraints than class labels, since providing constraints does not require the user to have significant prior knowledge about the categories in the dataset. Amongst the various methods proposed for utilizing user provided constraints for semi-supervised clustering [3,4], two of the well known include the semi-supervised kernel k-means (SS-KK) [36] and semi-supervised spectral clustering with normalized cuts (SS-SNC) [28]. While, SS-KK transforms the clustering distance measure by weighted kernel k-means with reward and penalty constraints to perform semi-supervised clustering of data given either as vectors or as a graph, SS-SNC utilizes supervision to change the clustering distance measure with pairwise information by spectral methods. The SS-NMF framework presented in this paper, allows the user to provide pairwise constraints on a small percentage of the data points. Specifically, these constraints specify whether the two data points should belong to the same cluster or should strictly belong to different clusters.

3 Semi-supervised non-negative matrix factorization for clustering

In this section, we first formulate the SS-NMF model in Sect. 3.1 and derive it in Sect. 3.2. We prove the correctness and convergence of the algorithm in Sect. 3.3. Equivalence of SS-NMF to SS-KK and SS-SNC is proven in Sect. 3.4, followed by a discussion of advantages of SS-NMF in Sect. 3.5.

3.1 Model formulation

We assume the data consists of n objects, and that m features have been extracted from each of the objects. Correspondingly, the data can be represented using a matrix $\mathbf{X} \in R^{m \times n}$ where columns index the data objects to be clustered and rows denote the features. An entry x_{fi} in this matrix denotes the value of feature f for object i.

We propose a SS-NMF model for data clustering. NMF has received much attention recently and proved to be very useful for applications such as face recognition, text mining, multimedia analysis, and DNA gene expression grouping. It was initially proposed for "parts-of-whole" decomposition [37,38], and later extended to a general framework for data clustering [10]. It can model widely varying data distributions and accomplish both hard and soft clustering simultaneously. When applied to the data matrix $\mathbf{X}$, NMF factorizes it into two non-negative matrices [53],

$$\mathbf{X} \approx \mathbf{P}\mathbf{Q}^{\mathrm{T}} \tag{1}$$

where $\mathbf{P} \in R^{m \times k}$ is cluster centroid, $\mathbf{Q} \in R^{n \times k}$ is cluster indicator, and k is the number of clusters.

In the proposed model, we perform symmetric non-negative tri-factorization of the similarity matrix $\mathbf{A} = \mathbf{X}^{\mathrm{T}}\mathbf{X} \in R^{n \times n}$ as,

$$\mathbf{A} \approx \mathbf{G}\mathbf{S}\mathbf{G}^{\mathrm{T}} \tag{2}$$

where $\mathbf{G} \in R^{n \times k}$ is the cluster indicator matrix. An entry g_{ih} in $\mathbf{G}$ gives the degree of association of object $\mathbf{x}_i$ with cluster h. The cluster membership of an object is given by finding the cluster with the maximum association value. $\mathbf{S} \in R^{k \times k}$ is the cluster centroid matrix that gives a compact $k \times k$ representation of $\mathbf{X}$.

Supervision is provided as two sets of pairwise constraints on the data objects: *must-link* constraints C_{ML} and *cannot-link* constraints C_{CL}. Every pair, $(\mathbf{x}_i, \mathbf{x}_j) \in C_{\mathrm{ML}}$ implies that $\mathbf{x}_i$ and $\mathbf{x}_j$ must belong to the same cluster. Similarly, all possible pairs $(\mathbf{x}_i, \mathbf{x}_j) \in C_{\mathrm{CL}}$ implies that the two objects should belong to different clusters. The constraints are accompanied by associated violation cost matrix $\mathbf{W}$. An entry w_{ij} in this matrix denotes the cost of violating the constraint between $\mathbf{x}_i$ and $\mathbf{x}_j$, if such a constraint exists, that is, either $(\mathbf{x}_i, \mathbf{x}_j) \in C_{\mathrm{ML}}$ or $(\mathbf{x}_i, \mathbf{x}_j) \in C_{\mathrm{CL}}$. The model relies on a distortion measure $D : R^m \rightarrow R$, to compute distance between the data objects. For a given k, the goal is to partition the data objects into k disjoint clusters $\{\mathbf{X}_h\}_{h=1}^{k}$, such that the total distortion between the objects and the corresponding cluster representatives is (locally) minimized according to the given distortion measure D, while constraint violations are kept to a minimum.

3.2 Algorithm derivation

We define the objective function of SS-NMF as follows:

$$J_{\mathrm{SS-NMF}} = \min_{\mathbf{S} \geq 0, \mathbf{G} \geq 0} \|\widetilde{\mathbf{A}} - \mathbf{G}\mathbf{S}\mathbf{G}^{\mathrm{T}}\|^2 \tag{3}$$

where $\widetilde{\mathbf{A}} = \mathbf{A} - \mathbf{W}_{\mathrm{reward}} + \mathbf{W}_{\mathrm{penalty}}$ is affinity or similarity matrix $\mathbf{A}$ with constraints $\mathbf{W}_{\mathrm{reward}} = \{w_{ij} | (\mathbf{x}_i, \mathbf{x}_j) \in C_{ML}, \text{s.t. } y_i = y_j\}$ and $\mathbf{W}_{\mathrm{penalty}} = \{w_{ij} | (\mathbf{x}_i, \mathbf{x}_j) \in C_{CL}, \text{s.t. } y_i = y_j\}$, w_{ij} is the penalty cost for violating a constraint between $\mathbf{x}_i$ and $\mathbf{x}_j$, and y_i is the cluster label of $\mathbf{x}_i$. $\mathbf{S} \in R^{k \times k}$ is the cluster centroid, and $\mathbf{G} \in R^{n \times k}$ is the cluster indicator.

We propose an iterative procedure for the minimization of Eq. (3) where we update one factor while fixing the others. The updating rules are,

$$\mathbf{S}_{ih} \leftarrow \mathbf{S}_{ih} \sqrt[2]{\frac{(\mathbf{G}^{\mathrm{T}}\widetilde{\mathbf{A}}\mathbf{G})_{ih}}{(\mathbf{G}^{\mathrm{T}}\mathbf{G}\mathbf{S}\mathbf{G}^{\mathrm{T}}\mathbf{G})_{ih}}} \tag{4}$$

$$\mathbf{G}_{ih} \leftarrow \mathbf{G}_{ih} \sqrt[4]{\frac{(\widetilde{\mathbf{A}}\mathbf{G}\mathbf{S})_{ih}}{(\mathbf{G}\mathbf{S}\mathbf{G}^{\mathrm{T}}\mathbf{G}\mathbf{S})_{ih}}} \tag{5}$$

Thus, the SS-NMF algorithm for document clustering can be illustrated in Algorithm 1.

3.3 Algorithm correctness and convergence

We now prove the theoretical correctness and convergence of SS-NMF. Motivated by Long et al. [42,43] and Ding et al. [11], we render the proof based on optimization theory, auxiliary function and several matrix inequalities.

Algorithm 1 SS-NMF Algorithm

INPUT: Data similarity matrix $\mathbf{A}$, number of clusters k, constraint penalty matrix $\mathbf{W}_{penalty}$, and constraint reward matrix $\mathbf{W}_{reward}$

OUTPUT: Clusters $\{\mathbf{X}_h\}_{h=1}^{k}$ with $\mathbf{Y}_h = \{i | \mathbf{x}_i \in \mathbf{X}_h\}$

METHOD:

1. Initialize $\mathbf{S}$ and $\mathbf{G}$ with non-negative values.
2. Construct $\widetilde{\mathbf{A}} = \mathbf{A} - \mathbf{W}_{reward} + \mathbf{W}_{penalty}$
3. Iterate for each i and h until *convergence*

    (a) Cluster centroid

$$S_{ih} \leftarrow S_{ih} \sqrt[2]{\frac{(\mathbf{G}^T\widetilde{\mathbf{A}}\mathbf{G})_{ih}}{(\mathbf{G}^T\mathbf{GSG}^T\mathbf{G})_{ih}}}$$

    (b) Cluster indicator

$$G_{ih} \leftarrow G_{ih} \sqrt[4]{\frac{(\widetilde{\mathbf{A}}\mathbf{GS})_{ih}}{(\mathbf{GSG}^T\mathbf{GS})_{ih}}}$$

3.3.1 Correctness

First, we prove the correctness of the algorithm, which can be stated as,

Proposition 1 *If the solution converges based on the updating rules in Eqs.* (4) *and* (5)*, the solution satisfies the KKT optimality condition.*

Proof Following the standard theory of constrained optimization, we introduce the Lagrangian multipliers λ_1 and λ_2 to minimize the lagrangian function,

$$L(\mathbf{S}, \mathbf{G}, \lambda_1, \lambda_2) = \min_{\mathbf{S}\geq 0, \mathbf{G}\geq 0} \|\widetilde{\mathbf{A}} - \mathbf{GSG}^T\|^2 - \mathrm{Tr}(\lambda_1\mathbf{S}^T) - \mathrm{Tr}(\lambda_2\mathbf{G}^T) \qquad (6)$$

Based on the KKT complementarity conditions,

$$\frac{\partial J}{\partial \mathbf{S}} = 0$$

$$\frac{\partial J}{\partial \mathbf{G}} = 0$$

we obtain the following two equations,

$$2\mathbf{G}^T\widetilde{\mathbf{A}}\mathbf{G} - 2\mathbf{G}^T\mathbf{GSG}^T\mathbf{G} + \lambda_1 = 0 \qquad (7)$$

$$4\widetilde{\mathbf{A}}\mathbf{GS} - 4\mathbf{GSG}^T\mathbf{GS} + \lambda_2 = 0 \qquad (8)$$

Applying the Hadamard multiplication on both sides of Eqs. (7) and (8) by $\mathbf{S}$ and $\mathbf{G}$, respectively, and using KKT conditions of

$$\lambda_1 \odot \mathbf{S}^2 = 0$$

$$\lambda_2 \odot \mathbf{G}^4 = 0$$

where $\odot$ denotes the Hadamard product of two matrices, we can prove that if $\mathbf{S}$ and $\mathbf{G}$ are a local minimizer of the objective function in Eq. (6), the following equations are satisfied,

$$(\mathbf{G}^T\widetilde{\mathbf{A}}\mathbf{G}) \odot \mathbf{S}^2 - (\mathbf{G}^T\mathbf{GSG}^T\mathbf{G}) \odot \mathbf{S}^2 = 0 \qquad (9)$$

$$(\widetilde{\mathbf{A}}\mathbf{GS}) \odot \mathbf{G}^4 - (\mathbf{GSG}^T\mathbf{GS}) \odot \mathbf{G}^4 = 0 \qquad (10)$$

Based on Eqs. (9) and (10), we derive the proposed updating rules of Eqs. (4) and (5). If the updating rules converge, the solution satisfies the KKT optimality condition. Proof is completed. □

3.3.2 Convergence

Next, we prove the convergence of the algorithm. In Propositions 2 and 3, we show that the objective function decreases monotonically under the two updating rules. This can be done by making use of an auxiliary function similar to that used in Lee and Seung [38].

Proposition 2 *If $\mathbf{G}$ is a fixed matrix, then $J(S) = \|\widetilde{A} - GSG^{\mathrm{T}}\|^2 = Tr(\widetilde{A}^{\mathrm{T}}\widetilde{A} - 2G^{\mathrm{T}}\widetilde{A}^{\mathrm{T}}GS + G^{\mathrm{T}}GSG^{\mathrm{T}}GS^{\mathrm{T}})$ decreases monotonically under the updating rule of Eq. (4).*

Proof A function $F(\mathbf{S}^{(t+1)}, \mathbf{S}^{(t)})$ is called an auxiliary function of $L(\mathbf{S}^{(t+1)})$ if it satisfies $F(\mathbf{S}^{(t+1)}, \mathbf{S}^{(t)}) \geq L(\mathbf{S}^{(t+1)})$ and $F(\mathbf{S}^{(t+1)}, \mathbf{S}^{(t)}) = L(\mathbf{S}^{(t+1)})$ for any $\mathbf{S}^{(t+1)}, \mathbf{S}^{(t)}$. Define $\mathbf{S}^{(t+1)} = \arg\min_s F(\mathbf{S}^{(t+1)}, \mathbf{S}^{(t)})$. By construction, $L(\mathbf{S}^{(t)}) = F(\mathbf{S}^{(t)}, \mathbf{S}^{(t)}) \geq F(\mathbf{S}^{(t+1)}, \mathbf{S}^{(t)}) \geq L(\mathbf{S}^{(t+1)})$. Thus, $L(\mathbf{S}^{(t)})$ is monotonic decreasing (non-increasing).

The key step is to find appropriate auxiliary function $F(\mathbf{S}^{(t+1)}, \mathbf{S}^{(t)})$. Assuming $\mathbf{G}$ is fixed, we write

$$L(\mathbf{S}^{(t+1)}) = \mathrm{Tr}(\widetilde{\mathbf{A}}^{\mathrm{T}}\widetilde{\mathbf{A}} - 2\mathbf{G}^{\mathrm{T}}\widetilde{\mathbf{A}}^{\mathrm{T}}\mathbf{G}\mathbf{S} + \mathbf{G}^{\mathrm{T}}\mathbf{G}\mathbf{S}\mathbf{G}^{\mathrm{T}}\mathbf{G}\mathbf{S}^{\mathrm{T}}) \tag{11}$$

and show that,

$$F(\mathbf{S}^{(t+1)}, \mathbf{S}^{(t)}) = \| \widetilde{\mathbf{A}} \|^2 - \sum_{ih} 2(\mathbf{G}^{\mathrm{T}}\widetilde{\mathbf{A}}\mathbf{G})_{ih}\mathbf{S}_{ih}^{(t)}(1 + \log \frac{\mathbf{S}_{ih}^{(t+1)}}{\mathbf{S}_{ih}^{(t)}})$$
$$+ \sum_{ih} \frac{(\mathbf{G}^{\mathrm{T}}\mathbf{G}\mathbf{S}^{(t)}\mathbf{G}^{\mathrm{T}}\mathbf{G})_{ih}\mathbf{S}_{ih}^{2(t+1)}}{\mathbf{S}_{ih}^{(t)}} \tag{12}$$

is an auxiliary function of $L(\mathbf{S}^{(t+1)})$.

First, we show that the inequality $F(\mathbf{S}^{(t+1)}, \mathbf{S}^{(t)}) \geq L(\mathbf{S}^{(t+1)})$ holds. We can see the second term in $F(\mathbf{S}^{(t+1)}, \mathbf{S}^{(t)})$ (aside from the negative sign) is always smaller than the second term in $L(\mathbf{S}^{(t+1)})$ because of the inequality $\frac{\mathbf{S}_{ih}^{(t+1)}}{\mathbf{S}_{ih}^{(t)}} \geq 1 + log\left(\frac{\mathbf{S}_{ih}^{(t+1)}}{\mathbf{S}_{ih}^{(t)}}\right), \forall \frac{\mathbf{S}_{ih}^{(t+1)}}{\mathbf{S}_{ih}^{(t)}} > 0$. In addition, the third term in $F(\mathbf{S}^{(t+1)}, \mathbf{S}^{(t)})$ is always bigger than the third term in $L(\mathbf{S}^{(t+1)})$ [11]. Thus, the condition $F(\mathbf{S}^{(t+1)}, \mathbf{S}^{(t)}) \geq L(\mathbf{S}^{(t+1)})$ holds. Second, we show the equality $F(\mathbf{S}^{(t+1)}, \mathbf{S}^{(t)}) = L(\mathbf{S}^{(t+1)})$ holds. It is obvious when $\mathbf{S}^{(t+1)} = \mathbf{S}^{(t)}$, the equality $F(\mathbf{S}^{(t+1)}, \mathbf{S}^{(t)}) = L(\mathbf{S}^{(t+1)})$ holds.

Therefore, $F(\mathbf{S}^{(t+1)}, \mathbf{S}^{(t)})$ is an auxiliary function of $L(\mathbf{S}^{(t+1)})$. Since we have,

$$\mathbf{S}^{(t+1)} = \arg\min_{\mathbf{S}} F\left(\mathbf{S}^{(t+1)}, \mathbf{S}^{(t)}\right) \tag{13}$$

$\mathbf{S}^{(t+1)}$ is given by the minimum of $F(\mathbf{S}^{(t+1)}, \mathbf{S}^{(t)})$ while fixing $\mathbf{S}^{(t)}$. The minimum value is obtained by setting,

$$\frac{\partial F(\mathbf{S}^{(t+1)}, \mathbf{S}^{(t)})}{\partial \mathbf{S}_{ih}^{(t+1)}} = -\sum_{ih} 2(\mathbf{G}^{\mathrm{T}}\widetilde{\mathbf{A}}\mathbf{G})_{ih} \frac{\mathbf{S}_{ih}^{(t)}}{\mathbf{S}_{ih}^{(t+1)}} + 2\sum_{ih} \frac{(\mathbf{G}^{\mathrm{T}}\mathbf{G}\mathbf{S}^{(t)}\mathbf{G}^{\mathrm{T}}\mathbf{G})_{ih}\mathbf{S}_{ih}^{(t+1)}}{\mathbf{S}_{ih}^{(t)}} = 0 \quad (14)$$

Thus, we can derive the updating rule of Eq. (4) as $\mathbf{S}_{ih} \leftarrow \mathbf{S}_{ih}\sqrt[2]{\frac{(\mathbf{G}^{\mathrm{T}}\widetilde{\mathbf{A}}\mathbf{G})_{ih}}{(\mathbf{G}^{\mathrm{T}}\mathbf{G}\mathbf{S}\mathbf{G}^{\mathrm{T}}\mathbf{G})_{ih}}}$. Under this updating rule, $J(\mathbf{S})$ decreases monotonically. Proof is completed. □

Proposition 3 *If S is a fixed matrix, $J(\mathbf{G}) = \|\widetilde{A} - \mathbf{GSG}^{\mathrm{T}}\|^2 = Tr(\widetilde{A}^{\mathrm{T}}\widetilde{A} - 2\mathbf{G}^{\mathrm{T}}\widetilde{A}^{\mathrm{T}}\mathbf{GS} + \mathbf{G}^{\mathrm{T}}\mathbf{GSG}^{\mathrm{T}}\mathbf{GS}^{\mathrm{T}})$ decreases monotonically under the updating rule of Eq. (5).*

Proof A function $F(\mathbf{G}^{(t+1)}, \mathbf{G}^{(t)})$ is called an auxiliary function of $L(\mathbf{G}^{(t+1)})$ if it satisfies $F(\mathbf{G}^{(t+1)}, \mathbf{G}^{(t)}) \geq L(\mathbf{G}^{(t+1)})$ and $F(\mathbf{G}^{(t+1)}, \mathbf{G}^{(t)}) = L(\mathbf{G}^{(t+1)})$ for any $\mathbf{G}^{(t+1)}, \mathbf{G}^{(t)}$. Define $\mathbf{G}^{(t+1)} = \arg\min_{\mathbf{G}} F(\mathbf{G}^{(t+1)}, \mathbf{G}^{(t)})$. By construction, $L(\mathbf{G}^{(t)}) = F(\mathbf{G}^{(t)}, \mathbf{G}^{(t)}) \geq F(\mathbf{G}^{(t+1)}, \mathbf{G}^{(t)}) \geq L(\mathbf{G}^{(t+1)})$. Thus, $L(\mathbf{G}^{(t)})$ is monotonic decreasing (non-increasing).

The key step is to find appropriate auxiliary function $F(\mathbf{G}^{(t+1)}, \mathbf{G}^{(t)})$. Assuming $\mathbf{S}$ is fixed, we write

$$L(\mathbf{G}^{(t+1)}) = \mathrm{Tr}(\widetilde{\mathbf{A}}^{\mathrm{T}}\widetilde{\mathbf{A}} - 2\widetilde{\mathbf{A}}^{\mathrm{T}}\mathbf{GSG}^{\mathrm{T}} + \mathbf{G}^{\mathrm{T}}\mathbf{SG}^{\mathrm{T}}\mathbf{GS}^{\mathrm{T}}\mathbf{G}^{\mathrm{T}}) \tag{15}$$

and show that,

$$F(\mathbf{G}^{(t+1)}, \mathbf{G}^{(t)}) = \| \widetilde{\mathbf{A}} \|^2 - \sum_{ih} 2(\widetilde{\mathbf{A}}\mathbf{G}^{(t)}\mathbf{S})_{ih}\mathbf{G}_{ih}^{(t)}\left(1 + 2log\frac{\mathbf{G}_{ih}^{(t+1)}}{\mathbf{G}_{ih}^{(t)}}\right) \tag{16}$$
$$+ \sum_{ih} \frac{(\mathbf{G}^{(t)}\mathbf{SG}^{(t)T}\mathbf{G}^{(t)}\mathbf{S})_{ih}\mathbf{G}_{ih}^{4(t+1)}}{\mathbf{G}_{ih}^{4(t)}}$$

is an auxiliary function of $L(\mathbf{G}^{(t+1)})$.

Following the proof of Proposition 2, it is not difficult to prove $F(\mathbf{G}^{(t+1)}, \mathbf{G}^{(t)})$ is an auxiliary function of $L(\mathbf{G}^{(t+1)})$. Since $\mathbf{G}^{(t+1)} = \arg\min_{\mathbf{G}} F(\mathbf{G}^{(t+1)}, \mathbf{G}^{(t)})$, $\mathbf{G}^{(t+1)}$ is given by the minimum of $F(\mathbf{G}^{(t+1)}, \mathbf{G}^{(t)})$ while fixing $\mathbf{G}^{(t)}$. The minimum value is obtained by setting $\frac{\partial F(\mathbf{G}^{(t+1)}, \mathbf{G}^{(t)})}{\partial \mathbf{G}_{ih}^{(t+1)}} = 0$. Thus, we can derive the updating rule of Eq. (5) as $\mathbf{G}_{ih} \leftarrow \mathbf{G}_{ih}\sqrt[4]{\frac{(\widetilde{\mathbf{A}}\mathbf{GS})_{ih}}{(\mathbf{GSG}^{\mathrm{T}}\mathbf{GS})_{ih}}}$. Under this updating rule, $J(\mathbf{G})$ decreases monotonically. Proof is completed. □

3.4 Equivalence of SS-NMF and other semi-supervised clustering methods

We now show that SS-NMF is a general and unified framework for semi-supervised clustering by establishing the relationship between SS-NMF and other well-known semi-supervised clustering algorithms, i.e., semi-supervised kernel *k*-means (SS-KK) [36] and semi-supervised spectral clustering with normalized cuts (SS-SNC) [28]. In fact, both these algorithms can be considered to be special cases of SS-NMF.

Proposition 4 *Orthogonal SS-NMF clustering is equivalent to SS-KK clustering.*

Proof The SS-NMF objective function is,

$$J_{\mathrm{SS-NMF}} = \min_{\mathbf{S}\geq 0, \mathbf{G}\geq 0} \|\widetilde{\mathbf{A}} - \mathbf{GSG}^{\mathrm{T}}\|^2 \tag{17}$$

The equation can be written as, $J_{\mathrm{SS-NMF}} = \|\widetilde{\mathbf{A}} - \mathbf{GSG}^{\mathrm{T}}\|^2 = \|\widetilde{\mathbf{A}} - \mathbf{G}'\mathbf{G}'^{\mathrm{T}}\|^2 = \mathrm{Tr}(\widetilde{\mathbf{A}}^{\mathrm{T}}\widetilde{\mathbf{A}} - 2\mathbf{G}'^{\mathrm{T}}\widetilde{\mathbf{A}}\mathbf{G}' + \mathbf{G}'^{\mathrm{T}}\mathbf{G}')$ if let $\mathbf{S} = \mathbf{Q}^{\mathrm{T}}\mathbf{Q}$ and $\mathbf{G}' = \mathbf{GQ}^{\mathrm{T}}$. Since $\mathrm{Tr}(\widetilde{\mathbf{A}}^{\mathrm{T}}\widetilde{\mathbf{A}} + \mathbf{G}'^{\mathrm{T}}\mathbf{G}')$ is a constant, the minimization of J becomes a maximization problem as,

$$\max_{\mathbf{G}'\geq 0} \mathrm{Tr}(\mathbf{G}'^{\mathrm{T}}\widetilde{\mathbf{A}}\mathbf{G}') \text{ s.t. } \mathbf{G}'^{\mathrm{T}}\mathbf{G}' = \mathbf{I} \tag{18}$$

The SS-KK objective function is [36],

$$J_{SS-KK} = \min \sum_{h=1}^{k} \sum_{i \in \mathbf{X}_h} \|\phi(\mathbf{x}_i) - \overline{\phi_h}\|^2$$
$$- \sum_{(\mathbf{x}_i, \mathbf{x}_j) \in C_{ML}, s.t. y_i = y_j} w_{ij} + \sum_{(\mathbf{x}_i, \mathbf{x}_j) \in C_{CL}, s.t. y_i = y_j} w_{ij} \qquad (19)$$

where $\phi(\cdot)$ is the kernel function and $\overline{\phi_h}$ the centroid. Let $\mathbf{E}$ be the matrix of pairwise squared Euclidean distances among the data points, $\mathbf{W}$ the constraint matrix and $\mathbf{G}$ the cluster indicator. Equation (19) becomes the minimization of the following function,

$$\min_{\mathbf{G} \geq 0} \text{Tr}(\mathbf{G}^{\mathrm{T}}(\mathbf{E} - 2\mathbf{W})\mathbf{G}) \text{ s.t. } \mathbf{G}^{\mathrm{T}}\mathbf{G} = \mathbf{I} \qquad (20)$$

We can convert the minimization of Eq. (20) to a maximization of the problem,

$$\max_{\mathbf{G} \geq 0} \text{Tr}(\mathbf{G}^{\mathrm{T}}\mathbf{K}\mathbf{G}) \text{ s.t. } \mathbf{G}^{\mathrm{T}}\mathbf{G} = \mathbf{I} \qquad (21)$$

where $\mathbf{K} = \mathbf{A} + \mathbf{W}$ and $\mathbf{A}$ the similarity matrix.

It is clear that the objective function of SS-NMF (Eq. (18)) is equivalent to that of SS-KK (Eq. (21)) if $\mathbf{K} = \widetilde{\mathbf{A}}$. The $\mathbf{G}'$ in Eq. (18) represents the same clustering as $\mathbf{G}$ of Eq. (21) does. Proof is completed. □

Proposition 5 *Orthogonal SS-NMF clustering is equivalent to SS-SNC clustering.*

Proof The objective function of SS-SNC is [28],

$$J_{SS-SNC} = \min \sum_{h=1}^{k} \frac{\mathbf{g}_h^{\mathrm{T}}(\widetilde{\mathbf{D}} - \widetilde{\mathbf{A}})\mathbf{g}_h}{\mathbf{g}_h^{\mathrm{T}}\widetilde{\mathbf{D}}\mathbf{g}_h} = \sum_{h=1}^{k} \mathbf{z}_h^{\mathrm{T}}(\mathbf{I} - \dot{\mathbf{A}})\mathbf{z}_h \qquad (22)$$

where $\widetilde{\mathbf{A}} = \mathbf{A} - \mathbf{W}_{\text{reward}} - \mathbf{W}_{\text{penalty}}$ is the pairwise similarity matrix with constraints, $\widetilde{\mathbf{D}} = \text{diag}(\widetilde{\mathbf{x}}_1, \ldots, \widetilde{\mathbf{x}}_n)$ is the diagonal matrix, $\mathbf{g}_h$ is the cluster indictor, scaled cluster indicator vector $\mathbf{z}_h = \widetilde{\mathbf{D}}^{1/2}\mathbf{g}_h / \|\widetilde{\mathbf{D}}^{1/2}\mathbf{g}_h\|$, and $\dot{\mathbf{A}} = \mathbf{D}^{-1/2}\widetilde{\mathbf{A}}\mathbf{D}^{-1/2}$.

It can be shown that the minimization of Eq. (22) becomes a maximization problem as,

$$\max_{\mathbf{Z} \geq 0} \text{Tr}(\mathbf{Z}^{\mathrm{T}}\dot{\mathbf{A}}\mathbf{Z}) \text{ s.t. } \mathbf{Z}^{\mathrm{T}}\mathbf{Z} = \mathbf{I} \qquad (23)$$

Also, it can be seen that Eq. (18) is equivalent to Eq. (23) if $\widetilde{\mathbf{A}} = \dot{\mathbf{A}}$. Moreover, the $\mathbf{G}'$ in Eq. (18) represents the same clustering as $\mathbf{Z}$ of Eq. (23) does. Proof is completed. □

From the above two proofs, we can see that SS-NMF, SS-KK, and SS-SNC are mathematically equivalent. However, notice that in SS-NMF, the matrix $\widetilde{\mathbf{A}}$ might have some negative values, which is not permitted in traditional NMF [37,38]. In this case, one possible solution is to perform some normalization techniques to guarantee non-negative values. Alternatively, we can simply relax the non-negative constraint to allow negative values as in Semi-NMF [40]. In either of the approaches, the clustering result will not get affected. In SS-NMF, the cluster indicator $\mathbf{G}'$ is near-orthogonal and can produce soft clustering results. The cluster centroid $\mathbf{S}$ can provide good characterization of the quality of data clustering because the residue of the matrix approximation $J = \min \|\widetilde{\mathbf{A}} - \mathbf{G}\mathbf{S}\mathbf{G}^{\mathrm{T}}\|$ is smaller than $J = \min \|\widetilde{\mathbf{A}} - \mathbf{G}\mathbf{G}^{\mathrm{T}}\|$. On the other hand, for SS-KK and SS-SNC, if input matrix is added with constraint weight $\mathbf{W}$, in order to ensure positive definiteness, certain additive constraints need to be enforced. Moreover, these constraints are difficult to be relaxed. Also, the cluster indicator $\mathbf{G}$ or $\mathbf{Z}$ is required
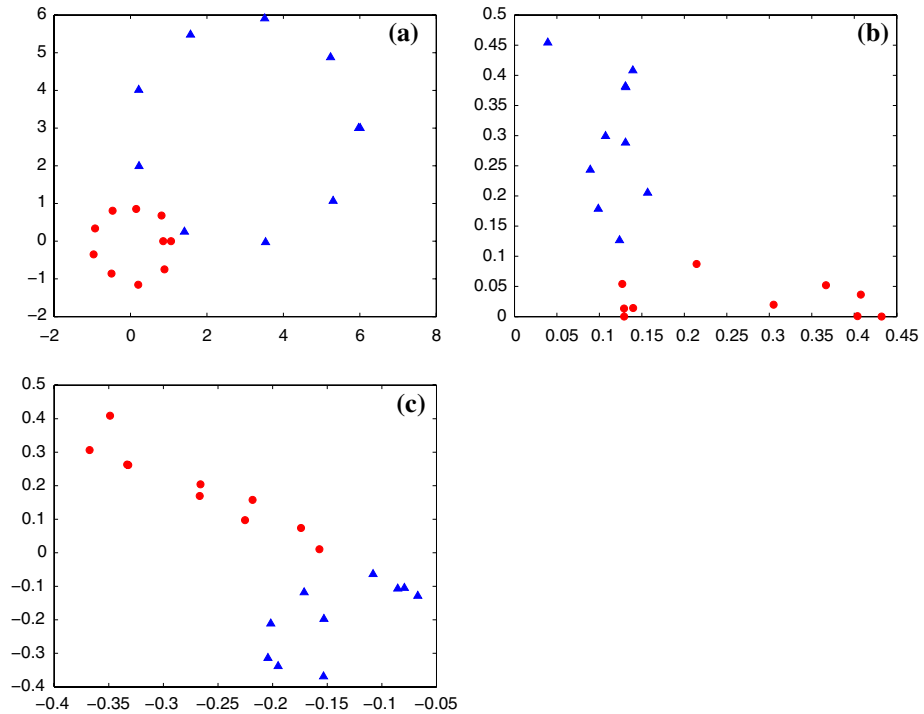
Fig. 1 **a** An artificial toy dataset consisting of two natural clusters. **b** Data distribution in the SS-NMF subspace of the two column vectors of **G**. The data points from the two clusters get distributed along the two axes. **c** Data distribution in the SS-SNC subspace of the first two singular vectors. There is no relationship between the axes and the clusters

to be orthogonal, leading to only hard clustering results. Hence, both SS-KK and SS-SNC can be viewed as special cases of SS-NMF with orthogonal space constraints. Thus, SS-NMF essentially provides a general and unified mathematical framework for semi-supervised data clustering.

3.5 Advantages of SS-NMF

In this section, we further illustrate the advantages of SS-NMF using a toy dataset shown in Fig. 1a, which follows an extreme distribution consisting of 20 data points forming two natural clusters: two circular rings with 10 data points each. Traditional unsupervised clustering methods, such as (kernel) k-means, spectral normalized cut or NMF, are unable to produce satisfactory results on this dataset. However, after incorporating knowledge from the user in the form of constraints, we are able to achieve much better results.

Unlike SS-SNC, SS-NMF maps the samples into a non-negative latent semantic space. Moreover, SS-NMF does not require the derived space to be orthogonal. Figure 1b, c shows, the data distributions in the two spaces for SS-NMF and SS-SNC, respectively. Data points belonging to the same cluster are depicted by the same symbol. For SS-NMF, we plot the data points in the space of two column vectors of **G**, while for SS-SNC the first two singular vectors are used. Clearly, in the SS-NMF space, every data point takes non-negative values in both the directions. Furthermore, in SS-NMF space, each axis corresponds to a cluster, and

Table 1 Cluster indicator **G** of SS-KK and SS-NMF for the toy dataset

| G	SS-KK		SS-NMF	
g_1	1	0	0.2778	0.0820
g_2	1	0	0.2977	0.0486
g_3	1	0	0.4301	0.0009
g_4	1	0	0.1295	0.0494
g_5	1	0	0.1377	0.0021
g_6	1	0	0.3845	0.0000
g_7	1	0	0.1281	0.0001
g_8	1	0	0.1426	0.0097
g_9	1	0	0.3119	0.0023
g_{10}	1	0	0.4691	0.0080
g_{11}	0	1	0.0651	0.3959
g_{12}	0	1	0.0599	0.4449
g_{13}	0	1	0.1161	0.4108
g_{14}	0	1	0.0978	0.2985
g_{15}	0	1	0.0592	0.2506
g_{16}	1	0	0.1220	0.1233
g_{17}	0	1	0.1047	0.1735
g_{18}	0	1	0.1503	0.2028
g_{19}	0	1	0.1233	0.2866
g_{20}	0	1	0.1181	0.3800

all the data points belonging to the same cluster are nicely spread along the axis. The cluster label for a data point can be determined by finding the axis with which the data point has the largest projection value. However, in the SS-SNC space, there is no direct relationship between the axes (singular vectors) and the clusters.

Table 1 shows the difference of cluster indicator between the hard clustering of SS-KK and soft clustering of SS-NMF. An exact orthogonality in SS-KK means that each row of cluster indicator **G** has only one nonzero element, which implies that each data object belongs to only 1 cluster. The near-orthogonality of cluster indicator **G** in SS-NMF relaxes this a bit, i.e., each data object could belong fractionally to more than 1 cluster. This can help in knowledge discovery in the cases where the data point is evenly projected along the different axes. For instance, $g_{16} = \{0.1220, 0.1233\}$ indicates that this data point may belong to any one of the two clusters.

SS-NMF uses an efficient iterative algorithm instead of solving a computationally expensive constrained eigen decomposition problem as in SS-SNC. The time complexity of SS-NMF is $\mathcal{O}(tkn^2)$ where k is the number of clusters, n is the number of documents, and t is the number of iterations. In fact, the time complexity is similar to that of the classical SS-KK clustering algorithm. However, compared to SS-KK, SS-NMF algorithm is simple as it only involves some basic matrix operations and hence can be easily deployed over a distributed computing environment when dealing with large datasets. Another advantage in favor of SS-NMF is that a partial answer can be obtained at intermediate stages of the solution by specifying a fixed number of iterations.

In Fig. 2, we demonstrate the computational speed of SS-NMF with respect to SS-KK and SS-SNC. This experiment was performed on a machine with a 3 GHz Intel Pentium 2

Fig. 2 Computational speed
comparison for SS-KK, SS-SNC
and SS-NMF
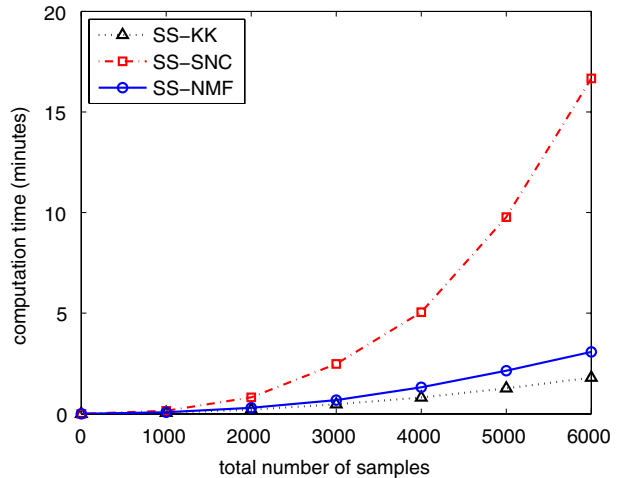
processor with 2 GB RAM. As the number of data samples increase, SS-SNC turns out to be the slowest of the three algorithms. SS-KK is the quickest with SS-NMF closely following it. In the next section, we show the superior performance of SS-NMF in terms of clustering accuracy in comparison with other clustering algorithms.

4 Experiments and results

In this section, we empirically demonstrate the performance of SS-NMF for data clustering. we present the details of our experiments, starting with the descriptions of the data sets (Sect. 4.1), the methodology and evaluation metrics (Sect 4.2), followed by thorough performance comparisons with leading unsupervised and semi-supervised clustering algorithms (Sect 4.3).

4.1 Data description

We have thoroughly evaluated the proposed algorithm on a variety of datasets, with number of classes ranging from 2 to 10, having between 27 and 500 data samples, and the dimensionality (attributes) ranging from 4 to 12,600. These datasets represent applications from different domains such as text mining and bioinformatics.

1. Text datasets
    We have used eight text datasets for document clustering. In particular, we created the datasets by mixing some of the datasets used in Han and Karypis [21].[2] Datasets *oh*0 and *oh*5 are from OHSUMED collection [22], a subset of MEDLINE database, which contains 233, 445 documents indexed using 14, 321 unique categories. Dataset *re*0 is from Reuters-21578 text categorization collection Distribution 1.0 [39]. Dataset *Fbis* is from the Foreign Broadcast Information Service data of TREC-5 [50]. For all datasets, the common words are removed and the words are stemmed using Porter's suffix-stripping algorithm.

---

[2] http://www.cs.umn.edu/~han/data/tmdata.tar.gz.

Table 2 Summary of text datasets used in the experiments

Datasets	No. of clusters	No. of words	No. of docs
Graft-Phos	2	2,432	293
England-Heart	2	2,504	375
Interest-Trade	2	2,682	438
Fbis2	2	2,000	200
Fbis3	3	2,000	300
Fbis4	4	2,000	400
Fbis5	5	2,000	500
Fbis10	10	2,000	500

Table 2 shows the document datasets used in our experiments. These datasets were created as follows:

– Classes $Graft\text{-}Survival$ and $Phospholipids$ from $oh5$ were mixed to form the $Graft\text{-}Phos$ dataset.
– Dataset $England\text{-}Heart$ was created by mixing classes $England$ and $Heart\text{-}Valve\text{-}Prosthesis$ from $oh0$.
– $Interest\text{-}Trade$ was formed by mixing $Interest$ and $Trade$ classes of $re0$ dataset.
– We randomly selected 2, 3, 4, 5, and 10 classes from *Fbis* to form datasets *Fbis2*, *Fbis3*, *Fbis4*, *Fbis5* and *Fbis10*, respectively.

In addition, we performed feature selection on the words according to Yang and Pedersen [54] by retaining the top 10% of the words based on mutual information in each of the datasets.

2. Gene expression datasets
   We have used five datasets from Kent Ridge Biomedical Dataset Repository[3] for gene expression clustering, including *AML/ALL*, *Colon Tumor*, *Prostate Cancer*, *ALL/MLL/AML*, and *Central Nervous System (CNS)*.
   – The *ALL/AML* dataset includes two types of human tumor-acute myelogenous leukemia (*AML*, 11 samples) and acute lymphoblastic leukemia (*ALL*, 27 samples).
   – The *Colon Tumor* dataset contains 62 samples collected from colon-cancer patients. Among them, 40 tumor biopsies are from tumors and 22 normal biopsies are from healthy parts of the colons of the same patients; 2, 000 out of around 6, 500 genes were selected based on the confidence in the measured expression levels.
   – The *Prostate Cancer* dataset contains 52 prostate tumor samples and 50 non-tumor prostate samples with around 12, 600 genes.
   – The *ALL/MLL/AML* dataset contains 57 leukemia samples which are divided into 20 *ALL*, 17 *MLL* and 20 *AML*.
   – The *Central Nervous System (CNS)* dataset consists of 34 samples: 10 classic medulloblastoms, 10 malignantgliomas, 10 rhabdoids and 4 normals.
   These datasets are summarized in Table 3.

3. UCI datasets
   We have utilized three datases from the UCI data repository [45]: *Iris*, *LettersIJL*, and *Soybean*.

---

[3] http://research.i2r.a-star.edu.sg/rp/

Table 3 Summary of gene expression datasets used in the experiments

Datasets	No. of clusters	No. of genes	No. of samples
ALL/AML	2	7,129	38
Colon Tumor	2	2,000	62
Prostate Tumor	2	12,600	102
ALL/MLL/AML	3	12,582	57
CNS	4	7,129	34

Table 4 Summary of UCI datasets used in the experiments

Datasets	No. of clusters	No. of attributes	No. of samples
Iris	3	4	150
LettersIJL	3	16	300
Soybean	4	35	47

– *Iris* plant data contains three classes: Iris Setosa, Iris Versicolour and Iris Virginica with four attributes sepal length, sepal width, petal length and petal width.
– *LettersIJL* is a randomly sampled subset of three letters I, J, L with 300 samples from Letters dataset.
– *Soybean* comes from Soybean Small data with 4 classes: D1, D2, D3 and D4.

The datasets are summarized in Table 4.

4.2 Methodology and evaluation metrics

We compared the performance of SS-NMF model on all the 15 datasets with the following 6 clustering methods: (1) k-means, (2) kernel k-means, (3) spectral normalized cuts, (4) NMF, (5) SS-KK, (6) SS-SNC. The first four methods are the most popular unsupervised data clustering methods, whereas SS-KK and SS-SNC are the representative semi-supervised ones. Through these comparison studies, we demonstrate the relative position of SS-NMF with respect to unsupervised and semi-supervised approaches in real-world data clustering.

We evaluated the clustering results using confusion matrix and the accuracy metric AC. Each entry (i, j) in the confusion matrix represents the number of objects in cluster i that belong to true class j. The AC metric measures how accurately a learning method assigns labels $\hat{y}_i$ to the ground truth y_i, and is defined as,

$$AC = \frac{\sum_{i=1}^{n} \delta(y_i, \hat{y}_i)}{n}. \tag{24}$$

where n denotes the total number of objects in the experiment, and δ is the delta function that equals one if $\hat{y}_i = y_i$, else its zero. Since iterative algorithm is not guaranteed to find the global minimum, it is beneficial to run the algorithm several times with different initial values and choose one trial with a minimal objective value. In reality, usually a few number of trials is sufficient. In the case of NMF and k-means, for a given k, we conducted 20 test runs. Three trials are performed in each of the 20 test runs and final accuracy value is the average of all the test runs.

Table 5 Comparison of document clustering accuracy between k-means, kernel k-means (KK), spectral normalized cuts (SNC), NMF and, SS-NMF with 3% constraints

Datasets	G-P	E-H	I-T	Fbis2	Fbis3	Fbis4	Fbis5	Fbis10
k-means	0.684	0.710	0.722	0.565	0.472	0.462	0.418	0.232
KK	0.798	0.714	0.742	0.570	0.553	0.552	0.514	0.378
SNC	0.655	0.632	0.703	0.990	0.636	0.597	0.542	0.392
NMF	0.815	0.784	0.956	0.995	0.653	0.612	0.590	0.416
SS-NMF	0.993	0.997	1.000	1.000	0.883	0.877	0.752	0.674

The names of the datasets have been abbreviated as G-P (Graft-Phos), E-H (England-Heart), I-T (Interest-Trade)

4.3 Results

4.3.1 Document clustering

We first performed comparison of the four unsupervised clustering approaches with SS-NMF having pairwise constraints on only 3% pairs of all the possible document pairs, which is $\binom{\text{total docs}}{2}$. Each of the constraints were generated by randomly selecting a pair of documents. If both the documents have the same class label (*must-link*) , then the constraint is assigned maximum weight in the document-document similarity matrix. On the other hand, if they belong to different classes (*cannot-link*), then the minimum weight in the similarity matrix is used for the constraint. For kernel k-means, we used a Gaussian (exponential) kernel $K(\mathbf{x}_1, \mathbf{x}_2) = \exp(-\|\mathbf{x}_1 - \mathbf{x}_2\|^2/2\sigma^2)$, with variance $\sigma = 0.00001$ for two clusters and $\sigma = 0.01$ for more than two clusters. In Table 5, we compared the algorithms on all the datasets using AC values. The performance of the first three methods is similar with NMF proving to be the best amongst the unsupervised methods. However, the accuracy of NMF greatly deteriorates and is unable to produce meaningful results on datasets having more than two clusters. On the other hand, the superior performance of SS-NMF is evident across all the datasets. We can see that in general a semi-supervised method can greatly enhance the document clustering results by benefitting from the user provided knowledge. Moreover, SS-NMF is able to generate significantly better results by quickly learning from the few pairwise constraints provided. Table 6 demonstrates the performance of SS-NMF when varying amounts of pairwise constraints were available a priori. We reported the results in terms of the confusion matrix $\mathbf{C}$ and the cluster centroid matrix $\mathbf{S}$. As the available prior knowledge increases from 0 to 5%, we can make the following two key observations. Firstly, the confusion matrices tend to become perfectly diagonal indicating higher clustering accuracy. Second observation pertains to the cluster centroid matrix $\mathbf{S}$ which represents the similarity or distance between the clusters. Increasing values of the diagonal elements of $\mathbf{S}$ indicate higher inter-cluster similarities. As expected, when the amount of prior knowledge available is more, the performance of the algorithm clearly gets better.

In Fig. 3a, the sparsity pattern of a typical document-document matrix $\mathbf{A} = \mathbf{X}^T\mathbf{X}$ (*England-Heart* in the figure) before clustering is shown. The SS-NMF algorithm is applied to the modified similarity matrix $\tilde{\mathbf{A}}$. Document clustering leads to re-ordering of the rows and columns of the matrix. Figure 3b,c shows the $\tilde{\mathbf{A}}$ matrices for *England-Heart* and *Fbis5* datasets after clustering with 5% pairwise constraints. Document clusters are indicated by the dense sub-matrices in these matrices.
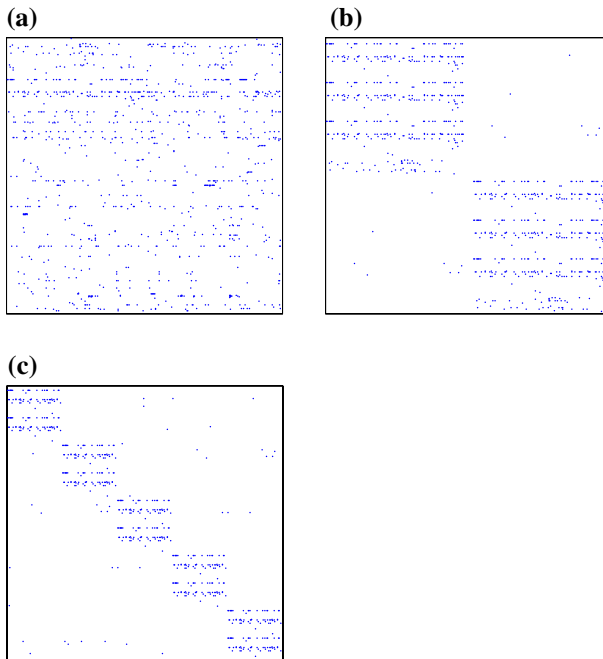
(a)

(b)

(c)

Fig. 3 **a** Typical document-document matrix (shown here *England-Heart*) before clustering. **b** *England-Heart* similarity matrix after clustering with SS-NMF. **c** *Fbis5* similarity matrix after clustering with SS-NMF

We now compare SS-NMF with the other two semi-supervised clustering approaches. As before, for SS-KK, a Gaussian kernel was used. In Fig. 4, we plotted the AC values against increasing percentage of pairwise constraints available, for the algorithms on all the datasets. On the whole, all three algorithms perform better as the percentage of pairwise constraints increases. While the performance of SS-KK is close to that of SS-SNC on the datasets in Figs. 4a–4c, it is clearly left out of the race completely in Figs. 4d–4h. This is mainly because of the fact that SS-KK is unable to maintain its accuracy when producing more than two clusters. While, the performance of SS-SNC is head-to-head with SS-NMF on *Fbis2* and *Fbis3*, it is consistently outperformed by SS-NMF on the rest of the datasets. Another noticeable fact is that the curve for SS-KK and SS-SNC might take a slow rise in some cases indicating that they need more amount of prior knowledge to improve the performance. Comparatively, SS-NMF gets better accuracy than the other two algorithms even for minimum percentage of pairwise constraints.

4.3.2 Gene expression clustering

We now present the comparison of SS-NMF with the other algorithms on real-world gene expression datasets. We first compared the four unsupervised clustering approaches with SS-NMF having pairwise constraints on only 3% pairs of all the possible sample pairs. For kernel k-means, we used a Gaussian (exponential) kernel, with variance $\sigma = 0.00001$ for *ALL/AML* and *Colon Tumor* datasets and a polynomial kernel $K(\mathbf{x}_1, \mathbf{x}_2) = (1 + \mathbf{x}_1 * \mathbf{x}_2')^p$ with polynomial parameter $p = 1$ for the other datasets. In Table 7, we have compared the algorithms on all the five gene expression datasets with AC values. As was the case with

Table 6 The comparison of confusion matrix **C** and cluster centroid matrix **S** of SS-NMF for different percentages of document pairs constrained

constr. (%)	Matrix	G-P dataset		E-H dataset		Fbis5 dataset				
0	**C**	116	21	181	81	1	1	4	1	4
		33	123	0	113	84	95	0	0	1
						14	1	11	1	0
						0	0	0	96	3
						1	3	85	2	92
	S	0.777	0	1.036	0	1.069	0	0	0	0
		0	0.773	0	1.150	0	0.869	0	0	0
						0	0	1.039	0	0
						0	0	0	0.87	0
						0	0	0	0	1.041
1	**C**	130	3	181	31	92	17	0	8	0
		19	141	0	163	0	0	22	0	0
						0	0	64	0	1
						0	0	1	89	0
						8	83	13	3	99
	S	0.914	0	1.216	0	2.520	0	0	0	0
		0	0.944	0	1.534	0	2.475	0	0	0
						0	0	2.425	0	0
						0	0	0	2.653	0
						0	0	0	0	2.823
3	**C**	147	0	193	0	55	0	0	7	0
		2	144	1	181	33	99	0	0	0
						0	0	0	0	0
						0	0	90	89	0
						72	1	10	4	100
	S	1.231	0	2.581	0	4.257	0	0	0	0
		0	1.300	0	2.798	0	4.678	0	0	0
						0	0	4.234	0	0
						0	0	0	4.089	0
						0	0	0	0	4.095
5	**C**	149	0	194	0	100	0	0	0	0
		0	144	0	181	0	100	0	0	0
						0	0	100	0	0
						0	0	0	100	0
						0	0	0	0	100
	S	1.609	0	3.427	0	6.517	0	0	0	0
		0	1.598	0	2.564	0	6.311	0	0	0
						0	0	6.042	0	0
						0	0	0	6.731	0
						0	0	0	0	5.922

The names of the datasets have been abbreviated as G-P (Graft-Phos), E-H (England-Heart)
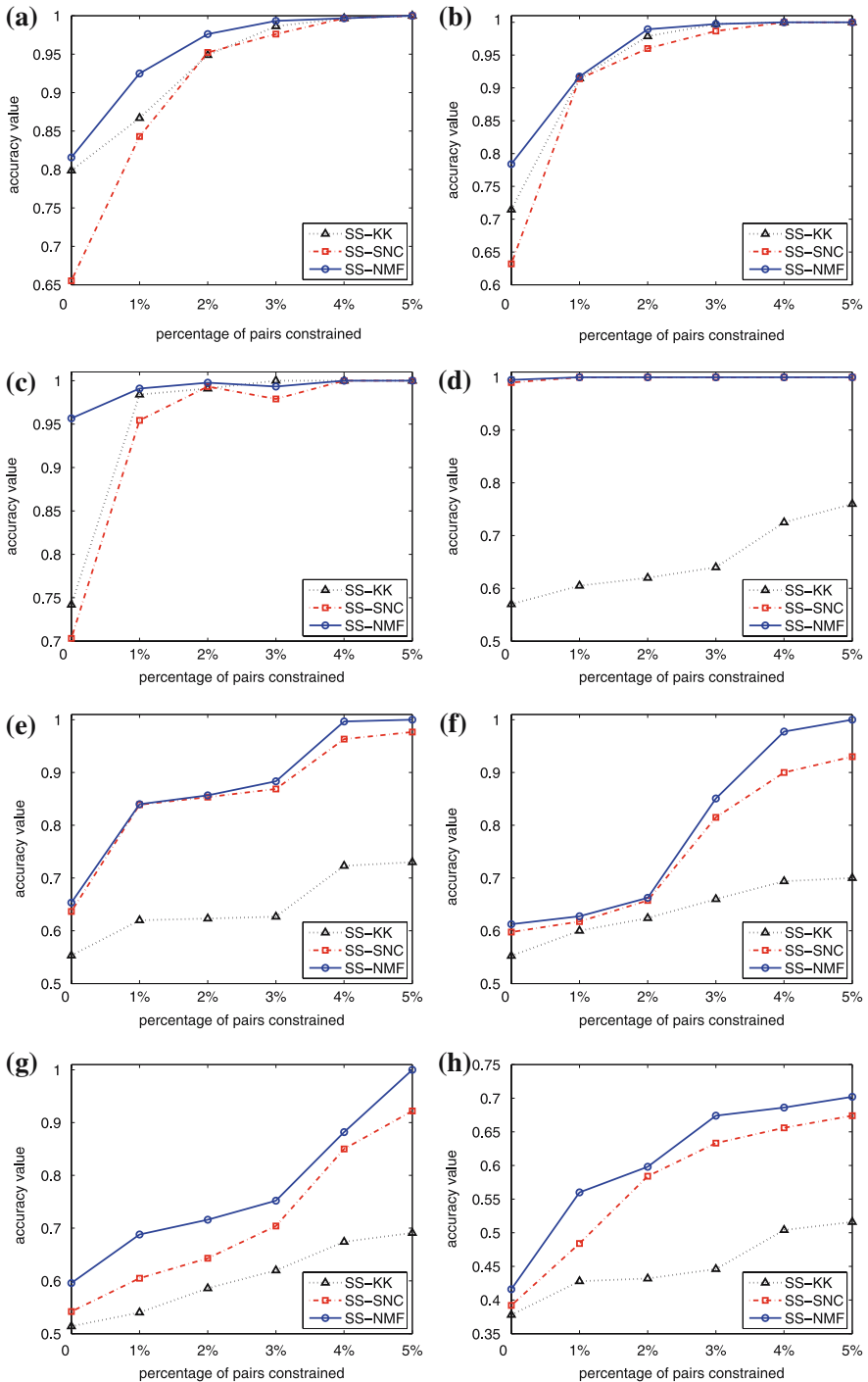
Fig. 4 Comparison of document clustering accuracy between SS-KK, SS-SNC, and SS-NMF for different percentages of document pairs constrained **a** *Graft-Phos*, **b** *England-Heart*, **c** *Interest-Trade*, **d** *Fbis2*, **e** *Fbis3*, **f** *Fbis4*, **g** *Fbis5*, and **h** *Fbis10* dataset

Table 7 Comparison of gene expression clustering accuracy between *k*-means, kernel *k*-means (KK), spectral normalized cuts (SNC), NMF and, SS-NMF with 3% constraints

Datasets	ALL/AML	CT	PC	ALL/MLL/AML	CNS
k-means	0.526	0.629	0.578	0.631	0.529
KK	0.526	0.532	0.607	0.649	0.676
SNC	0.631	0.596	0.598	0.560	0.655
NMF	0.684	0.661	0.647	0.666	0.767
SS-NMF	0.763	0.758	0.666	0.736	0.852

The names of the datasets have been abbreviated as CT (Colon Tumor), PC (Prostate Cancer)

document clustering, SS-NMF performs to be the best across all the datasets. It is evident that the algorithm learns quickly in spite of having few constraints. Table 8 demonstrates the performance of SS-NMF improves when the number of pairwise constraints on the gene expression datasets increase from 0 to 5%. These results are reported in terms of the confusion matrix **C** and the normalized cluster centroid matrix **S** as before.

Next, we compare SS-NMF with the other two semi-supervised clustering approaches on the gene expression datasets. Figure 5 shows a plot of the AC values against increasing percentage of pairwise constraints for the three semi-supervised algorithms on all the five datasets. All three algorithms perform better as the percentage of pairwise constraints increases. SS-NMF performs significantly better than the other two algorithms with any percentage of constraints when distinguishing between tumor and non-tumor samples, as in Figs. 5b-c. Also, for clustering subtypes of tumors, although the differences are small, SS-NMF outperforms the other two algorithms as seen from Figs. 5a, d-e.

4.3.3 UCI datasets clustering

Table 9 shows the comparison of SS-NMF with the unsupervised clustering algorithms on all three UCI datasets. As before, for kernel *k*-means, we used a Gaussian (exponential) kernel with variance $\sigma = 1$ for *Iris* data and polynomial kernel with polynomial parameter $p = 1$ for the other datasets. As can be seen, with just 5% constraints, SS-NMF yields significantly better results than the unsupervised approaches. For instance, on Soybean data, SS-NMF improves the accuracy over 25%. Similar trends can also be observed for other two datasets.

Figure 6 illustrates the performance of SS-NMF and the two semi-supervised algorithms for increasing number of pairwise constraints on the UCI datasets. We can observe that SS-NMF clustering always produces best accuracy performance when the dimensionality of the datasets is high (Fig. 6b,c). However, it is unable to achieve quality clustering on low dimensionality datasets for fewer constraints. For *Iris* dataset which has dimensionality of 4 (Fig. 6a) SS-NMF yields low accuracy initially and tends to slowly catch up with SS-KK as the percentage of pairwise constraints increase. This shows that SS-NMF is a viable proposition for low-dimensional data as well but needs higher percentage of constraints.

5 Conclusions

We presented SS-NMF: a semi-supervised approach for clustering based on non-negative matrix factorization. In the proposed framework, users are able to provide supervision in

Table 8 The comparison of confusion matrix **C** and cluster centroid matrix **S** of SS-NMF for different percentages of gene expression sample pairs constrained

constr. (%)	Matrix	ALL/AML dataset		ALL/MLL/AML dataset			CNS dataset			
0	**C**	17	2	17	6	0	7	1	2	0
		10	9	1	6	5	0	7	0	0
				2	5	15	2	0	8	0
							1	2	0	4
	S	1.367	0	1.350	0	0	3.773	0	0	0
		0	1.306	0	1.076	0	0	4.671	0	0
				0	0	1.349	0	0	2.921	0
							0	0	0	3.395
1	**C**	18	1	17	3	0	9	1	1	1
		9	10	1	8	6	1	7	0	0
				2	6	14	0	0	8	0
							0	2	1	3
	S	1.373	0	1.388	0	0	3.897	0	0	0
		0	1.314	0	1.081	0	0	4.685	0	0
				0	0	1.350	0	0	2.934	0
							0	0	0	3.412
3	**C**	19	1	16	1	7	9	1	1	1
		8	10	4	13	0	1	8	0	0
				0	3	13	0	0	9	0
							0	1	0	3
	S	1.382	0	1.361	0	0	4.046	0	0	0
		0	1.327	0	1.100	0	0	5.056	0	0
				0	0	1.357	0	0	3.188	0
							0	0	0	3.518
5	**C**	21	1	15	3	0	10	0	1	0
		6	10	5	14	1	0	9	0	0
				0	0	19	0	0	8	0
							0	1	1	4
	S	1.391	0	1.391	0	0	4.736	0	0	0
		0	1.333	0	1.122	0	0	5.255	0	0
				0	0	1.358	0	0	3.351	0
							0	0	0	3.612

Table 9 Comparison of UCI data clustering accuracy between k-means, kernel k-means (KK), spectral normalized cuts (SNC), NMF and, SS-NMF with 5% constraints

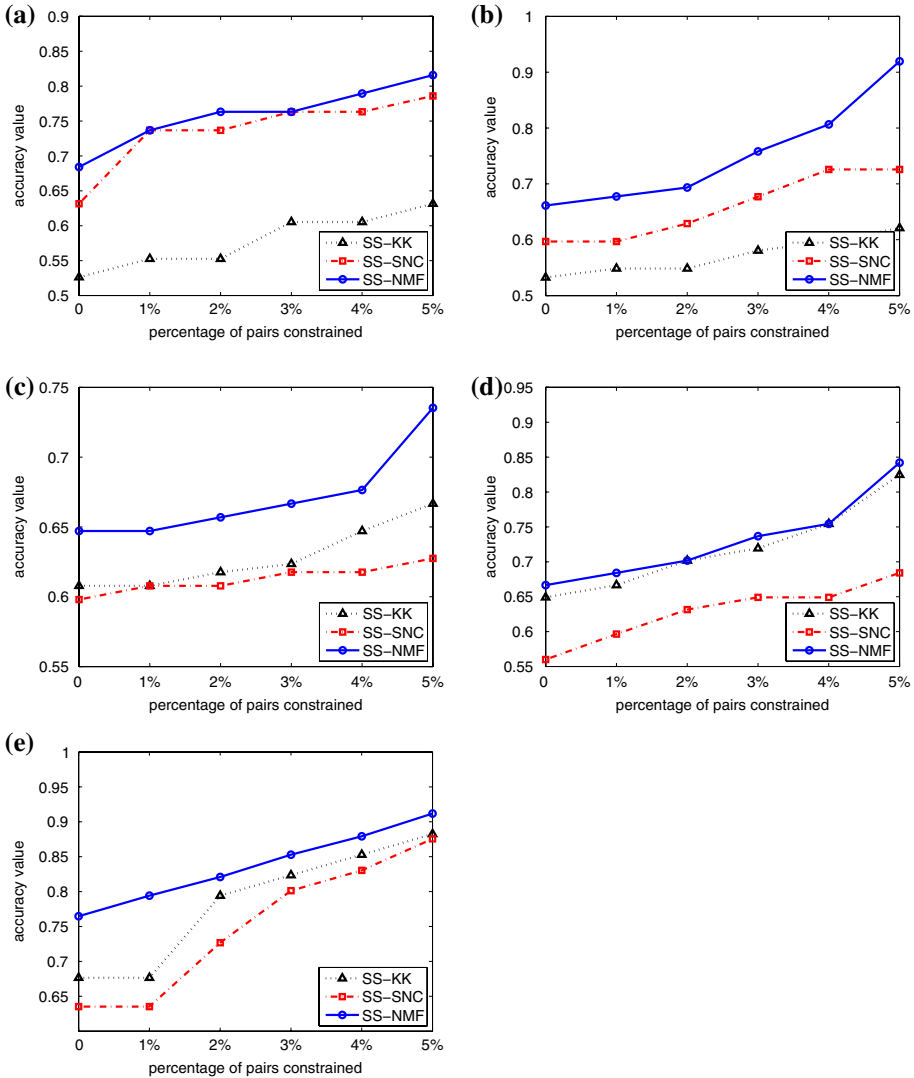| Datasets	Iris	LettersIJL	Soybean
k-means	0.8263	0.5167	0.7234
KK	0.6933	0.5167	0.7021
SNC	0.6667	0.4467	0.7234
NMF	0.6733	0.5200	0.7447
SS-NMF	0.9267	0.6300	0.9149

Fig. 5 Comparison of gene expression clustering accuracy between SS-KK, SS-SNC, and SS-NMF for different percentages of sample pairs constrained **a** *ALL/AML*, **b** *Colon Tumor*, **c** *Prostate Cancer*, **d** *ALL/MLL/AML* and **e** *CNS* dataset

terms of *must-link* and *cannot-link* pairwise constraints on the data objects. We derived an iterative algorithm to perform symmetric tri-factorization of the data similarity matrix. We have mathematically shown the correctness and convergence of SS-NMF. Moveover, we proved that SS-NMF provides a general and unified framework for semi-supervised data clustering. Existing approaches can be considered as special cases of it. Empirically, we showed that SS-NMF outperforms well-established unsupervised and semi-supervised clustering methods in grouping publicly available datasets.
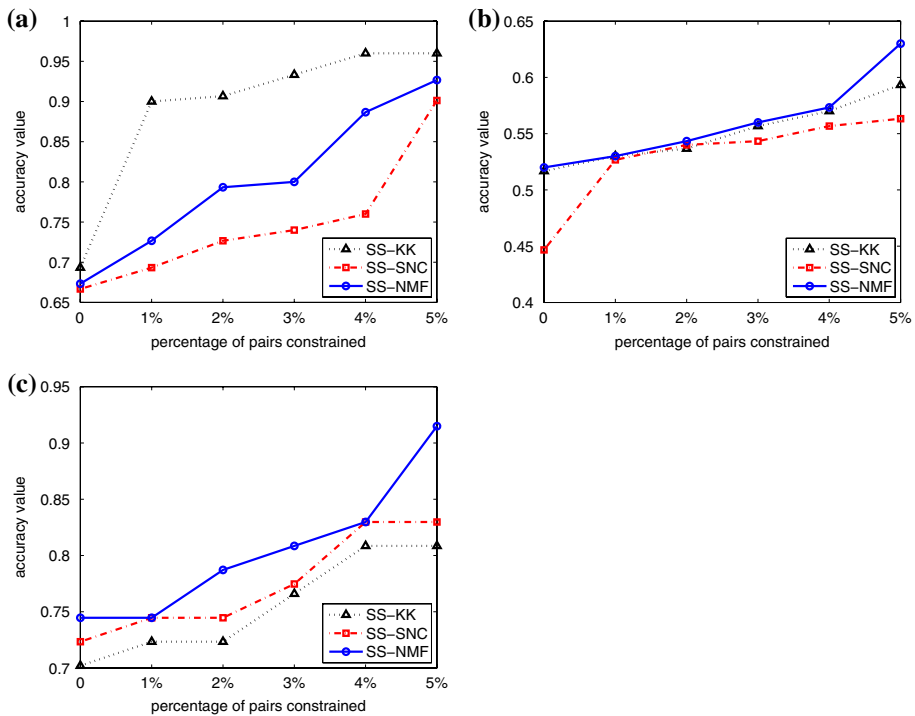
Fig. 6 Comparison of UCI data clustering accuracy between SS-KK, SS-SNC, and SS-NMF for different percentages of sample pairs constrained **a** *Iris*, **b** *LettersIJL* and **c** *Soybean* dataset

References

1. Bansal N, Blum A, Chawla S (2002) Correlation clustering. Proceedings of the 43rd symposium on foundations of computer science, pp 238–247
2. Bar-Hillel A, Hertz T, Shental N, Weinshall D (2003) Learning distance functions using equivalence relations. Proceedings of the 20th international conference on machine learning, pp 11–18
3. Basu S, Banerjee A, Mooney RJ (2002) Semi-supervised clustering by seeding. Proceedings of the 19th international conference on machine learning, pp 27–34
4. Basu S, Bilenko M, Mooney RJ (2004) A probabilistic framework for semi-supervised clustering. Proceedings of the 10th ACM SIGKDD international conference on knowledge discovery and data mining, pp 59–68
5. Blum A, Mitchell TM (1998) Combining labeled and unlabeled data with co-training. Annual workshop on computational learning theory, Proceedings of the 11th annual conference on Computational learning theory, pp 92–100
6. Boley D (1998) Principal direction divisive partitioning. Data Mining Knowledge Discovery 2(4):325–344
7. Charikar M, Guruswami V, Wirth A (2003) Clustering with qualitative information. Proceedings of the 44th IEEE symposium on foundations of computer science, pp 524–533
8. Chung FRK (1997) Spectral Graph Theory, American Mathematical Society
9. Dempster AP, Laird NM, Rubin DB (1977) Maximum likelihood from incomplete data via the em algorithm. J R Stat Soc B pp 1–38

10. Ding C, He X, Simon HD (2005) On the equivalence of nonnegative matrix factorization and spectral clustering. Proceedings of SIAM international conference on data mining, pp 606–610
11. Ding C, Li T, Peng W, Park H (2006) Orthogonal nonnegative matrix tri-factorizations for clustering. Proceedings of the 12th ACM SIGKDD international conference on knowledge discovery and data mining, ACM Press, New York, pp 126–135
12. Dubnov S, EI-Yaniv R, Gdalyahu Y, Schneidman E, Tishby N, Yona G (2002) A new nonparametric pairwise clustering algorithm based on iterative estimation of distance profiles. Machine Learning 47(1):35–61
13. Duda RO, Hart PE, Stork DG (2000) Pattern Classification. Wiley, New York
14. Fiedler M (1973) Algebraic connectivity of graphs. Czechoslovak Math J 23:298–305
15. Fiedler M (1975a) Eigenvectors of acyclic matrices. Czechoslovak Math J 25:607–618
16. Fiedler M (1975b) A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory. Czechoslovak Math J 25:619–633
17. Fisher D (1987) Knowledge acquisition via incremental conceptual clustering. Machine Learning 2:139–172
18. Godbole S, Harpale A, Sarawagi S, Chakrabarti S (2004) Document classification through interactive supervision of document and term labels. Proceedings of the 8th European conference on principles and practice of knowledge discovery in databases, pp 185–196
19. Gondek D, Hofmann T (2007) Non-redundant data clustering. Knowledge Information Systems 12(1):1–24
20. Hagen L, Kahng AB (1992) New spectral methods for ratio cut partitioning and clustering. IEEE Trans CAD Integrated Circuits Systems 11(9):1074–1085
21. Han E-H, Karypis G (2000) Centroid-based document classification: Analysis and experimental results. Proceedings of the 4th European conference on principles of data mining and knowledge discovery, pp 424–431
22. Hersh W, Buckley C, Leone T, Hickam D (1994) Ohsumed: an interactive retrival evaluation and new large test collection for research. Proceedings of 17th ACM SIGIR conference on research and development in information retrieval, pp 192–201
23. Hinneburg A, Keim D (2003) A general approach to clustering in large databases with noise. Knowledge Information Systems 5(4):387–415
24. Hinneburg A, Keim DA (1998) An efficient approach to clustering in large multimedia databases with noise. Proceedings of the 4th international conference on knowledge discovery and data mining, pp 58–65
25. Hotho A, Staab S, Stumme G (2003) Text clustering based on background knowledge, Technical report 425. University of Karlsruhe, Institute AIFB, Karlsruhe
26. Huang Y, Mitchell TM (2006) Text clustering with extended user feedback. Proceedings of the 29th ACM SIGIR conference on research and development in information retrieval, pp 413–420
27. Jain AK, Murty MN, Flynn PJ (1999) Data clustering: a review. ACM computing surveys 31(3):264–323
28. Ji X, Xu W (2006) Document clustering with prior knowledge. Proceedings of the 29th ACM SIGIR conference on research and development in information retrieval, pp 405–412
29. Joachims T (1999) Transductive inference for text classification using support vector machines. Proceedings of the 16th international conference on machine learning, pp 200–209
30. Jones R, McCallum A, Nigam K, Riloff E (1999) Bootstrapping for text learning tasks. Workshop on text mining: foundations, techniques and applications, proceedings of international joint conference on artifical intelligence, pp 52–63
31. Kamvar SD, Klein D, Manning CD (2003) Spectral learning. Proceedings of the 18th international joint conference on artificial intelligence, pp 561–566
32. Kaufman L, Rousseeuw P (1990) Finding Groups in Data: an introduction to cluster analysis. Wiley, New York
33. Kim H, Lee S (2004) An intelligent information system for organizing online text documents. Knowledge Information Systems 6(2):125–149
34. Klein D, Kamvar S, Manning C (2002) From instance-level constraints to space-level constraints: Making the most of prior knowledge in data clustering. Proceedings of the 19th international conference on machine learning, pp 307–314
35. Koga H, Ishibashi T, Watanabe T (2007) Fast agglomerative hierarchical clustering algorithm using locality-sensitive hashing. Knowledge Information Systems 12(1):25–53
36. Kulis B, Basu S, Dhillon I, Mooney R (2005) Semi-supervised graph clustering: a kernel approach. Proceedings of the 22nd international conference on machine learning, pp 457–464
37. Lee DD, Seung HS (1999) Learning the parts of objects by non-negative matrix factorization. Nature 401:788–791

38. Lee D, Seung H (2001) Algorithms for non-negative matrix factorization. Proceedings of annual conference on neural information processing systems 13, pp 556–562
39. Lewis DD (1999) Reuters-21578 text categorization test collection distribution 1.0, http://www.research.att/lewis
40. Li T, Ding C (2006) The relationships among various nonnegative matrix factorization methods for clustering. Proceedings of the 6th IEEE international conference on data mining, pp 362–371
41. Liu B, Li X, Lee WS, Yu PS (2004) Text classification by labeling words. Proceedings of AAAI conference on artificial intelligence, pp 425–430
42. Long B, Zhang Z, Yu PS (2005) Co-clustering by block value decomposition. Proceedings of the 11th ACM SIGKDD international conference on knowledge discovery and data mining, pp 635–640
43. Long B, Zhang Z, Yu PS (2007) Relational clustering by symmetric convex coding. Proceedings the 24th annual international conference on machine learning, pp 569–576
44. MacQueen J (1967) Some methods for classsification and analysis of multivariate observations. Proceedings of 5th Berkeley symposium on mathematical statistics and probability, pp 281–297
45. Newman C, Hettich S, Merz C (1998) UCI repository of machine learning databases
46. Nigam K, McCallum AK, Thrun S, Mitchell TM (1998) Learning to classify text from labeled and unlabeled documents. Proceedings of AAAI conference on artificial intelligence, pp 792–799
47. Raghavan H, Madani O, Jones R (2005) Interactive feature selection. Proceedings of international joint conference on artificial intelligence, pp 841–846
48. Sheikholesami G, Chatterjee S, Zhang A (1998) Wavecluster: a multi-resolution clustering approach for very large spatial databases. Proceedings of the international conference on very large databases, pp 428–439
49. Shi J, Malik J (2000) Normalized cuts and image segmentation. IEEE Trans on PAMI 22(8):888–905
50. TREC (n.d.) Text retrieval conference, http://trec.nist.gov
51. Wagstaff K, Cardie C, Rogers S, Schroedl S (2001) Constrained k-means clustering with background knowledge. Proceedings of the 18th international conference on machine learning, pp 577–584
52. Xing EP, Ng AY, Jordan M, Russell S (2002) Distance metric learning, with application to clustering with side-information. Advances in neural information processing systems 15, pp 502–512
53. Xu W, Liu X, Gong Y (2003) Document clustering based on non-negative matrix factorization. Proceedings of the 26th ACM SIGIR conference on research and development in information retrieval, pp 267–273
54. Yang Y, Pedersen JO (1997) A comparative study on feature selection in text categorization. Proceedings of the 14th international conference on machine learning, pp 412–420
55. Zhang T, Ramakrishnan R, Livny M (1996) Birch: An efficient data clustering method for very large databases. Proceedings of the ACM SIGMOD international conference on management of data, pp 103–114

Author Biographies

Yanhua Chen received the MS degree in Computer Science and Engineering from Michigan State University, East Lansing, MI, in 2004. She is currently a PhD student at Machine Vision and Pattern Recognition Laboratory of the Department of Computer Science, Wayne State University, Detroit, MI. Her research interests are in the areas of pattern recognition, machine learning, data mining, graph theory, and information retrieval.

Manjeet Rege received his PhD from the Department of Computer Science, Wayne State University, MI. He has a MS in Computer Information Systems from Eastern Michigan University, MI and BS in Mathematics from the University of Mumbai, India. His research interests lie in the areas of Data Mining, Information Retrieval, Machine Learning, and Multimedia Analysis.

Ming Dong received his BS degree from Shanghai Jiao Tong University, Shanghai, P.R. China in 1995 and his PhD degree from the University of Cincinnati, Ohio, in 2001, both in electrical engineering. He joined the faculty of Wayne State University, Detroit, MI in 2002. He is currently an assistant professor of Computer Science and the Director of the Machine Vision and Pattern Recognition Laboratory. His research interests include pattern recognition, data mining, and multimedia analysis. He has published over 60 technical articles, many in premium journals and conferences such as IEEE Trans. on Neural Networks, IEEE Trans. on Computers, IEEE Trans. on Fuzzy Systems, IEEE ICDM, IEEE CVPR, ACM Multimedia, and WWW. He is an associate editor of the Pattern Analysis and Applications Journal and was on the editorial board of International Journal of Semantic Web and Information Systems, 2005-2006. He also serves as a program committee member for many related conferences.

Jing Hua is an assistant professor of computer science at Wayne State University and the Director of the Graphics and Imaging Laboratory. He received his PhD degree (2004) in computer science from the State University of New York at Stony Brook. He also received his MS degree (1999) in pattern recognition and artificial intelligence from the Institute of Automation, the Chinese Academy of Sciences in Beijing, P.R. China and BS degree (1996) in electrical engineering from the Huazhong University of Science and Technology, P.R. China. His research interests include computer graphics, geometric modeling, and image informatics. His research is currently supported by the NSF, MTTC, and 21st Century Jobs Funds. Dr. Hua is on the editorial board for the International Journal of Technology Enhanced Learning and serves as a program committee member for many international conferences. He is a member of the IEEE and the IEEE Computer Society.