Mach Learn (2017) 106:695–712
DOI 10.1007/s10994-016-5601-9

Non-redundant multiple clustering by nonnegative matrix factorization

Sen Yang[1] · **Lijun Zhang**[1]

Received: 3 May 2016 / Accepted: 18 October 2016 / Published online: 23 December 2016

Abstract Clustering is one of the basic tasks in data mining and machine learning which aims at discovering hidden structure in the data. For many real-world applications, there often exist many different yet meaningful clusterings while most of existing clustering methods only produce a single clustering. To address this limitation, multiple clustering, which tries to generate clusterings that are high quality and different from each other, has emerged recently. In this paper, we propose a novel alternative clustering method that generates non-redundant multiple clusterings sequentially. The algorithm is built upon nonnegative matrix factorization, and we take advantage of the nonnegative property to enforce the non-redundancy. Specifically, we design a quadratic term to measure the redundancy between the reference clustering and the new clustering, and incorporate it into the objective. The optimization problem takes on a very simple form, and can be solved efficiently by multiplicative updating rules. Experimental results demonstrate that the proposed algorithm is comparable to or outperforms existing multiple clustering methods.

Keywords Multiple clustering · Alternative clustering · Nonnegative Matrix Factorization · Multiplicative updating

1 Introduction

Clustering, one of the most fundamental tasks in knowledge discovery, plays an important role in investigating the inherent and hidden structures of data (Jain et al. 1999). The goal

Editors: Bob Durrant, Kee-Eung Kim, Geoff Holmes, Stephen Marsland, Zhi-Hua Zhou and Masashi Sugiyama.

✉ Lijun Zhang
zhanglj@lamda.nju.edu.cn

Sen Yang
yangs@lamda.nju.edu.cn

[1] National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China

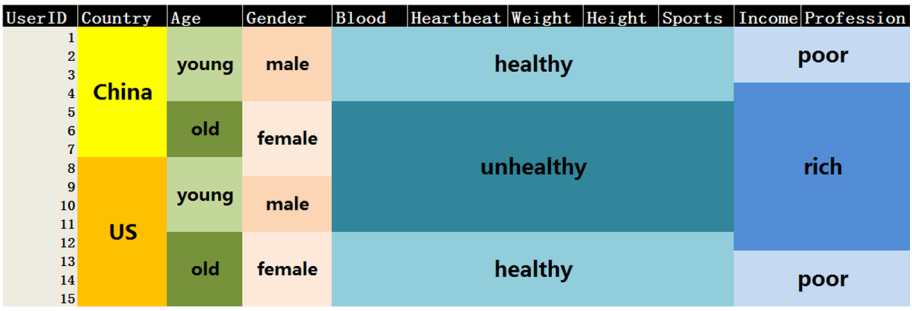

Fig. 1 People can be clustered by different criteria

of clustering is to partition the data points into groups such that those within each group are similar to each other. In the era of big data, clustering can be used as a pre-processing step to deal with large-scale datasets. After clustering, a big dataset becomes a number of small ones, which facilitates subsequent tasks such as summarization and visualization.

Literatures on clustering are vast, and most of them focus on producing a single clustering. However, for many real-world datasets, there often exist different ways to partition them. For example, people can be clustered by inherent properties like country, gender, etc, or by their current state, like health status and profession, as indicated in Fig. 1. It is clear that there exist multiple clusterings of people and each of them has reasonable explanations. Similarly, in bio-informatics, proteins can be clustered by their structures or functions. Thus, for a given dataset, there may exist different ways in which we can divide it into different groups and each of them reflects different aspects of the data. So, it would be helpful if we can present multiple clusterings to users. In addition, these clusterings are required to be not only high quality, but also different from each other.

To get multiple clusterings, the most straightforward approaches include (i) running a clustering algorithm multiple times, using different parameters, (ii) running different clustering algorithms, and (iii) a combination of the above methods (Jain et al. 1999). However their results are unstable and the clusterings may be similar to each other because they do not take the existing clusterings into account. To overcome this drawback, there are two general strategies proposed by researchers to generate multiple non-redundant clusterings (Jain et al. 1999). The first one tries to simultaneously generate multiple clusterings which are required to be different from each other (e.g., Caruana et al. 2006; Dasgupta and Ng 2010). Although in theory these can find the globally optimal solution, the optimization problem is difficult to be solved and in practice most of them can only generate two clusterings simultaneously. The second one generates multiple clustering in a greedy way such that multiple clusterings are produced sequentially and the new clustering is required to be different from the previous clusterings (e.g., Gondek and Hofmann 2003; Bae and Bailey 2006). The second kind of clustering methods are also refereed to as *alternative* clustering. Compared with the first strategy, alternative clustering methods are much more efficient and can generate a large number of different clusterings.

In this paper, we propose a novel alternative clustering method based on Nonnegative Matrix Factorization (NMF) (Lee and Seung 1999). NMF aims at finding two nonnegative matrices U and V whose product is an approximation of the original matrix X. The major difference between NMF and other matrix factorization methods, such as SVD, are the nonnegative constraints on U and V. Although NMF has been applied to generate a single

clustering (Xu et al. 2003; Cai et al. 2011), to the best of our knowledge, this is the first work that investigates NMF in the context of multiple clustering. By exploiting the nonnegative property, we introduce a regularization term, that measures the redundancy between the reference (or previous) clustering(s) and the new clustering, into the objective of NMF. This regularization term is the inner product of the similarity matrices of different clusterings, and can be formulated as a simple quadratic function. The resulting optimization problem is also very simple and can be solved efficiently by two multiplicative updating rules. In this way, the quality of the clustering is guaranteed by the cost function of NMF, and the diversity is ensured by the regularization term. Experimental results on real-life datasets demonstrate the effectiveness of our proposed algorithm for multiple clustering.

The rest of the paper is organized as follows. Section 2 discusses related work on producing multiple clusterings. Section 3 describes the basic NMF algorithm and our clustering algorithm. We present our experimental results on real-life datasets in Sect. 4 and the conclusions are in Sect. 5.

2 Related work

Although the problem of multiple clustering is relatively young, there already exist many multiple clustering methods and we can simply divide them into two categories: unsupervised multiple clustering and semi-supervised multiple clustering (Dang and Bailey 2015). Algorithms belonging to the first category identify multiple clusterings without reference to any existing clusterings. In contrast, algorithms in the second category generate multiple clusterings sequentially with reference to existing clusterings.

2.1 Unsupervised multiple clustering

Caruana et al. (2006) propose an approach called Meta Clustering which is simple and easy to implement. It assigns different weights which agree with the Zipf distribution, to features and applies k-means algorithm in the new feature space. Multiple clusterings are generated by random initializations of centroids in different feature spaces, but these clusterings may be similar to each other. Dasgupta and Ng (2010) use spectral clustering to generate multiple clusterings. They treat each eigenvector of the normalized Laplacian matrix as a clustering dimension, perform clustering in each dimension, and obtain multiple clusterings which tend to be dissimilar with each other as eigenvectors are orthogonal to each other.

In order to find the globally optimal multiple clustering solution, Jain et al. (2008), Dang and Bailey (2010) and Niu et al. (2010) try to generate two or more clusterings simultaneously without any background knowledge. The objective function of Jain et al. (2008) is the sum of the error term of k-means method and pairwise dissimilarity terms of the clusterings. It uses the sum of the dot products of the representative vectors and the mean vectors, which belong to different clusterings, to quantify the dissimilarity between each pair of the clusterings. Dang and Bailey (2010) propose a different approach called CAMI which aims to generate two dissimilar clusterings simultaneously in the original data space. Formulating the clustering problem under mixture models, CAMI maximizes the log-likelihood term which accounts for the clustering quality and minimizes the mutual information between two mixture models which accounts for the dissimilarity between the two clusterings. The approach proposed by Niu et al. (2010) transforms the data into independent subspaces and then uses the spectral clustering method to generate clusterings in these subspaces. The independence between different subspaces is quantified by the Hilbert-Schmidt Independence Criterion (HSIC)

(Gretton et al. 2005). The overall objective function consists of the relaxed spectral clustering objective terms in each subspace and the HSIC terms between the subspaces.

2.2 Semi-supervised multiple clustering

Many alternative clustering methods generate alternative clusterings in a sequential manner, e.g., Gondek and Hofmann (2003), Bae and Bailey (2006), Cui et al. (2007), Davidson and Qi (2008), Dang and Bailey (2014), Dang and Bailey (2015), Hu et al. (2015). They are semi-supervised in the sense that they require one or more existing clusterings as input and find an alternative clustering that is uncorrelated to the given ones.

COALA, which is proposed by Bae and Bailey (2006), generates a set of pairwise cannot-link constraints from the given clustering. It attempts to find a different clustering by making a tradeoff between satisfying these cannot-link constraints and ensuring high quality within an agglomerative clustering process. Hu et al. (2015) propose a method called MSC, that requires the clustering results are insensitive to noise and dissimilar with each other. The authors prove that the larger the eigengap of the normalized Laplacian matrix, the more stable the clustering is. This method uses a simplex constraint to generate different sparse weights to the features and then uses spectral clustering to produce multiple stable clusterings.

Information theoretic principles are also used in the generation of alternative clusterings. Gondek and Hofmann (2003) propose an approach called CIB which can be summarized as follows. Let X and Y be two random variables denoting data objects and features, respectively. CIB attempts to find a clustering C such that the shared information between X and C is minimized, while at the same time the shared information between Y and C is maximized conditioning on the information provided by the variable Z which represents provided class labels. Dang and Bailey (2015) propose a framework named MACL for uncovering multiple alternative clusterings. This framework seeks for a novel clustering conditioning on all previous known clusterings. It combines the maximum likelihood principle and the mutual information. The clustering quality is guaranteed by the likelihood maximization over the data and the dissimilarity is ensured by the minimization over the information shared by each pair clusterings.

Some researchers consider the generation of alternative clusterings from the feature space perspective. They use a data space S to characterize the existing clustering(s) and try to find a new feature space which is either orthogonal to S, or independent from S. Once the novel feature space is constructed, any clustering algorithm can be used in this space to generate an alternative clustering. Cui et al. (2007) present a projection-based framework to generate alternative clusterings. The key idea is projecting the data into a space that is orthogonal to the given one and then partition the data into different clusters in the new subspace. Davidson and Qi (2008) propose a subspace multiple clustering method named ADFT which is based on distance matrix learning. It's also a linear transformation method. However, instead of characterizing the known clustering according to the mean vectors or a feature subset, it uses instance must-link and cannot-link constraints to learn a distance function (Xing et al. 2003). Then it makes use of this distance function to get a transformation matrix which gives different weights to the features. Compared with the work of Cui et al. (2007), it can be used even in the case that the data dimension is smaller than the number of clusters, while the algorithm of Cui et al. (2007) can't.

In Dang and Bailey (2014), there are two algorithms that generate multiple clusterings in different subspaces. The first algorithm called RPCA tries to learn a subspace that preserves the global variance property and is independent from the reference clustering. It also uses the HSIC to measure the correlation between different subspaces. This method is suitable for

applications where the boundaries between clusters are linear or close to linear functions. The second algorithm called RegGB tries to deal with the nonlinear case. Its objective function is the same as Laplacian Eigenmap (Belkin and Niyogi 2001). But it incorporates a constraint which requires the new data space to be orthogonal to the subspace S characterizing the reference clustering. The subspace S that characterizes the reference clustering is learned by kernel discriminant analysis (KDA) (Baudat and Anouar 2000).

In this paper, we propose a novel multiple clustering method named Multiple NMF (MNMF). It is also a semi-supervised clustering method (i.e., this method seeks alternative clusterings in sequence conditioning on all the previous clusterings). Based on NMF, we transform the data into different subspaces, and introduce a regularization term to remove redundancy. As a result, the novel clustering generated by our algorithm is different from all the previous clusterings.

3 Our algorithm

In this section, we will introduce our method—MNMF in detail. We use the basic NMF objective function as the measurement of the clustering quality and design a regularization term to quantify the redundancy between different clusterings. Then, we propose an efficient algorithm to solve the optimization problem. We start with the basic NMF.

3.1 NMF

Given a nonnegative matrix, NMF factorizes it into the product of two nonnegative matrices (Lee and Seung 1999). Let $X = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N] \in \mathbb{R}^{M\times N}$ be a data matrix, where each column is an instance. Denote the two new nonnegative matrices by $U = [u_{ik}] \in \mathbb{R}^{M\times K}$ and $V = [v_{jk}] \in \mathbb{R}^{N\times K}$, respectively. Then, we have

$$X \approx UV^\top$$

Generally speaking, the rank of the two matrices U and V is much smaller than the rank of the matrix X, i.e., $K \ll \min(M, N)$.

To measure the quality of the approximation, we need a cost function that quantifies the difference between X and $UV^\top$. The most popular function is the sum of squared errors, or the Frobenius norm of $X - UV^\top$, and the associated optimization problem is given by

$$\min_{U,V\geq 0} J_{sse} = \left\| X - UV^\top \right\|_F^2 = \sum_{i,j}\left(x_{ij} - \sum_{k=1}^{K} u_{ik}v_{jk} \right)^2 \tag{1}$$

Lee and Seung (2001) present an iterative algorithm which optimizes the above problem in the following way

$$u_{ik} \leftarrow u_{ik}\frac{(XV)_{ik}}{(UV^\top V)_{ik}}, \quad v_{jk} \leftarrow v_{jk}\frac{(X^\top U)_{jk}}{(VU^\top U)_{jk}}.$$

It has been proved that the objective function value is nonincreasing under the multiplicative updating rules (Lee and Seung 2001).

For the fact that $K \ll \min(M, N)$, NMF can be treated as a technique of dimension reduction. And we can view the approximation column by column as follows

$$\mathbf{x}_j \approx \sum_{k=1}^{K} \mathbf{u}_k v_{jk}$$

where $\mathbf{u}_k$ is the k-th column vector of U. Thus, each data point $\mathbf{x}_j$ is approximated by a linear combination of the columns of U, with the coefficient given in the k-th row of V. Therefore, U can be regarded as a basis consisting of nonnegative vectors and each row in V is a new representation of an instance with respect to U. For the purpose of clustering, we can set K to be the number of clusters and assign $\mathbf{x}_i$ to cluster $c_i = \underset{k}{\operatorname{argmax}}\, v_{ik}$.

The most important difference between NMF and the other matrix factorization methods, like SVD, is the nonnegative constraints on U and V which only allow additive combinations among different basis vectors. For this reason, it is believed that NMF can learn a part-based representation which reveals the inherent structure of the original data.

3.2 Multiple NMF

Suppose, there exists a clustering C_1 which partition the original data into different groups. How can we make use of it to generate a new clustering, which is on one hand different from C_1 and on the other hand has high quality?

First, from the reference clustering C_1, we can extract a similarity matrix $S \in \mathbb{R}^{N \times N}$ between N data points. Specifically, we have

$$S_{ij} = \begin{cases} 1, & \text{if } \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ are in the same cluster} \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

Then, the similarity matrix S can be used to guide the generation of the new clustering. Next, we discuss how to modify the standard NMF to exploit this additional information. Our goal is to generate a new clustering C_2 by NMF. As discussed in Sect. 3.1, column vectors of U consist of a set of basis vectors of the new subspace, and rows of V provide new presentations of data points.

Define

$$W = VV^\top \in \mathbb{R}^{N \times N}$$

where W_{ij} is the inner product of the i-th row and the j-th row of V. Since V is nonnegative, $W_{ij} \geq 0$ represents the similarity between new representations of $\mathbf{x}_i$ and $\mathbf{x}_j$. Because the new clustering C_2 is also derived from V, we can use W to approximate the similarity matrix of C_2. Note that in the ideal case that V is an indicator matrix, W is equal to the similarity matrix of C_2.

Given two similarity matrices S and W, we measure the redundancy between C_1 and C_2 as the inner produce of S and W, i.e.,

$$\langle S, W \rangle = \sum_{i,j=1}^{N} W_{ij} S_{ij}$$

In order to minimize the redundancy, we want the value of $\sum_{ij} W_{ij} S_{ij}$ to be as small as possible. From the property of trace operation, we can formulate the above quantity as a simple quadratic term

$$R = \sum_{i,j}^{N} W_{ij} S_{ij} = \mathrm{tr}(W^\top S) = \mathrm{tr}(VV^\top S) = \mathrm{tr}(V^\top SV) \tag{3}$$

By minimizing R, we expect that if two data points $\mathbf{x}_i$ and $\mathbf{x}_j$ are in the same cluster in the reference clustering C_1 (i.e., $S_{ij} = 1$), they would be in different clusters in the new clustering C_2. Then, we incorporate R as a regularization term into Eq. (1), and obtain the objective function of Multiple NMF as follows:

$$\min_{U,V\geq 0} \phi = \left\| X - UV^\top \right\|_F^2 + \lambda \,\mathrm{tr}(V^\top SV). \tag{4}$$

In the above equation, the regularization parameter $\lambda > 0$ controls the trade off between the clustering quality and the dissimilarity between different clusterings. By minimizing ϕ, we can get an alternative clustering C_2 with respect to the reference clustering C_1. Although the objective function ϕ is not convex in U and V jointly, it is convex in them separately. Thus, a local minima can be found by optimizing U and V alternatively, similar to the optimization of the basic NMF. In the mathematical form, our optimization problem is similar to that of graph regularized NMF (GNMF) (Cai et al. 2011), and thus we can borrow techniques of GNMF to optimize it.

The objective function ϕ in Eq. (4) can be rewritten as follows:

$$\begin{aligned}\phi &= \mathrm{tr}((X - UV^\top)(X - UV^\top)^\top) + \lambda \,\mathrm{tr}(V^\top SV) \\ &= \mathrm{tr}(XX^\top) - 2\,\mathrm{tr}(XVU^\top) + \mathrm{tr}(UV^\top VU^\top) + \lambda \,\mathrm{tr}(V^\top SV)\end{aligned}$$

There are two nonnegative constraints that $U \geq 0$ and $V \geq 0$. In order to eliminate the constraints, we derive the Lagrange function of ϕ. Let $A = [a_{ik}] \in \mathbb{R}^{M\times K}$ and $B = [b_{jk}] \in \mathbb{R}^{N\times K}$ be the matrices of dual variables. The Lagrange function L is

$$\begin{aligned}L =& \,\mathrm{tr}(XX^\top) - 2\,\mathrm{tr}(XVU^\top) + \mathrm{tr}(UV^\top VU^\top) \\ &+ \lambda \,\mathrm{tr}(V^\top SV) + \mathrm{tr}(AU^\top) + \mathrm{tr}(BV^\top)\end{aligned}$$

The partial derivatives of L with respect to U and V are:

$$\begin{aligned}\frac{\partial L}{\partial U} &= -2XV + 2UV^\top V + A \\ \frac{\partial L}{\partial V} &= -2X^\top U + 2VU^\top U + 2\lambda SV + B\end{aligned}$$

From the KKT conditions that $A_{ik}u_{ik} = 0$ and $B_{jk}v_{jk} = 0$, we obtain the following equations for u_{ik} and v_{jk}:

$$\begin{aligned}-(XV)_{ik}u_{ik} + (UV^\top V)_{ik}u_{ik} &= 0 \\ -(X^\top U)_{jk}v_{jk} + (VU^\top U)_{jk}v_{jk} + \lambda(SV)_{jk}v_{jk} &= 0\end{aligned}$$

leading to the following multiplicative updating rules:

$$u_{ik} \leftarrow u_{ik}\frac{(XV)_{ik}}{(UV^\top V)_{ik}} \tag{5}$$

$$v_{jk} \leftarrow v_{jk}\frac{(X^\top U)_{jk}}{(VU^\top U + \lambda SV)_{jk}} \tag{6}$$

For these two updating rules, we have the following theorem.

Algorithm 1 Multiple NMF

Input: Data matrix X, cluster number k, parameter λ, a reference clustering C_1, threshold ε
Output: Alternative clustering C_2
1: Initialize matrix U and V randomly;
2: Calculate S by Eq. (2);
3: **while** the difference between successive objectives $\geq \varepsilon$ **do**
4: Update U by Eq. (5);
Update V by Eq. (6);
Normalize U and V by Eq. (7);
5: **end while**
6: **for** each instance $\mathbf{x}_i$ **do**
7: $c_i = \text{argmax}_k\, v_{ik}$;
8: **end for**
9: return $C_2 = \{c_1, \ldots, c_N\}$;

Theorem 1 *The objective function ϕ in Eq. (4) is nonincreasing under the updating rules in Eqs. (5) and (6).*

The proof is given in the next section. The analysis essentially follows that of NMF and GNMF (Lee and Seung 2001; Cai et al. 2011). We note that the above theorem cannot guarantee the final solution is a stationary point. To obtain a stronger theoretical guarantee, one can adopt the technique of Lin (2007) to modify the updating rule.

In practice, to prevent elements of V being unbounded, we will normalize the columns of U to make them of unit length (Xu et al. 2003). The matrix V also needs to be adjusted accordingly. The normalization steps are as follows

$$u_{ik} \leftarrow \frac{u_{ik}}{\sqrt{\sum_i u_{ik}^2}}, \quad v_{jk} \leftarrow v_{jk} \sqrt{\sum_i v_{ik}^2} \tag{7}$$

After obtaining the new representation V of the data, we get an alternative clustering by either assigning the instance $\mathbf{x}_i$ to the cluster $\max_k v_{ik}$ or applying any clustering method like k-means to V. Given a reference clustering C_1, the whole process of generating an alternative clustering C_2 by MNMF is summarized in Algorithm 1.

When $\lambda = 0$, the two updating rules are the same as NMF, and the algorithm reduces to the traditional clustering by NMF. In addition, our multiple clustering method can also be extended to deal with the case that we need to generate more than two clusterings. Each time we obtain a clustering C_i, we get the corresponding similarity matrix S_i. Then, we can simply calculate the accumulated similarity matrix by $S = \sum_i S_i$, and use S to generate another clustering.

3.3 Proof of Theorem 1

The objective function ϕ of MNMF in Eq. (4) is bounded from below by zero. In order to prove the algorithm converges to a stable state, we need to show that ϕ is non-increasing under the updating rules in Eqs. (5) and (6). Since the second term of ϕ is only related to V, we have exactly the same updating rule for U in MNMF as in the original NMF. Thus, we can use the convergence proof of NMF to show that ϕ is non-increasing under the update rule in Eq. (5). Please see Lee and Seung (2001) for details.

Then, we need to prove that ϕ is non-increasing under the updating step in Eq. (6). We follow the similar procedure described in Cai et al. (2011). Define

$$F(V) = \phi(U, V) = \left\| X - UV^\top \right\|_F^2 + \lambda \operatorname{tr}(V^\top SV)$$

We will construct an auxiliary function which satisfies the following conditions:

$$G(v, v^t) \geq F(v), \quad G(v, v) = F(v).$$

Lemma 1 *If G satisfies the conditions above, then F is non-increasing under the updating rule:*

$$v^{t+1} = \arg\min_v G(v, v^t) \tag{8}$$

Proof

$$F(v^{t+1}) \leq G(v^{t+1}, v^t) \leq G(v^t, v^t) = F(v^t)$$

Considering any element v_{ab} in V, we use F_{ab} to denote the part of ϕ which is only relevant to v_{ab}. It is easy to check that

$$F'_{ab} = \left(\frac{\partial \phi}{\partial V}\right)_{ab} = (-2X^\top U + 2VU^\top U + 2\lambda SV)_{ab} \tag{9}$$

$$F''_{ab} = 2(U^\top U)_{ab} + 2\lambda S_{aa} \tag{10}$$

Since our updating rule is essentially element-wise, it is sufficient to show that each F_{ab} is non-increasing under the updating step in Eq. (8).

Lemma 2

$$\begin{aligned} G(v, v_{ab}^t) = {} & F_{ab}(v_{ab}^t) + F'_{ab}(v_{ab}^t)(v - v_{ab}^t) \\ & + \frac{(VU^\top U)_{ab} + \lambda(SV)_{ab}}{v_{ab}^t}(v - v_{ab}^t)^2 \end{aligned} \tag{11}$$

is an auxiliary function for F_{ab} which satisfies the conditions in Lemma 1.

Proof Obviously, $G(v, v) = F_{ab}(v)$. So we only need to prove that $G(v, v_{ab}^t) \geq F_{ab}(v)$. To do this, we use the Taylor series expansion of $F_{ab}(v)$:

$$F_{ab}(v) = F_{ab}(v_{ab}^t) + F'_{ab}(v_{ab}^t)(v - v_{ab}^t) + [(U^\top U)_{bb} + \lambda S_{aa}](v - v_{ab}^t)^2$$

Compared with Eq. (11), we observe that $G(v, v_{ab}^t) \geq F_{ab}(v)$ is equivalent to

$$\frac{(VU^\top U)_{ab} + \lambda(SV)_{ab}}{v_{ab}^t} \geq (U^\top U)_{bb} + \lambda S_{aa} \tag{12}$$

We have

$$(VU^\top U)_{ab} = \sum_{l=1}^{k} v_{al}^t (U^\top U)_{lb} \geq v_{ab}^t (U^\top U)_{bb}$$

and

$$\lambda(SV)_{ab} = \lambda \sum_{j=1}^{M} S_{aj} v_{jb}^t \geq \lambda S_{aa} v_{ab}^t.$$

As a result, Eq. (12) holds and we have $G(v, v_{ab}^t) \geq F_{ab}(v)$ □

We can now demonstrate the convergence of Theorem 1:

Proof Replacing $G(v, v_{ab}^t)$ in Eq. (8) with Eq. (11), then we get the updating rule for v_{ab}:

$$v_{ab}^{t+1} = v_{ab}^t - v_{ab}^t \frac{F_{ab}^{'}(v_{ab}^t)}{2(VU^\top U)_{ab} + 2\lambda(SV)_{ab}} = v_{ab}^t \frac{(X^\top U)_{ab}}{(VU^\top U + \lambda SV)_{ab}}.$$

Since Eq. (11) is an auxiliary function, F_{ab} is nonincreasing under this updating rule. □

In summary, we conclude that the objective function ϕ is non-increasing under the updating rules in Eqs. (5) and (6). Furthermore, the convergence analysis of GNMF (Yang et al. 2014) implies our algorithm still converges under the additional normalization steps.

4 Experiment

In this section, we present experimental comparisons of our algorithm, multiple NMF against the following methods: COALA (Bae and Bailey 2006), two methods from Cui et al. (2007) denoted by Algo1 and Algo2, ADFT (Davidson and Qi 2008), SC (Dasgupta and Ng 2010), two subspace methods from Dang and Bailey (2014) denoted by RPCA and RegGB, and MSC (Hu et al. 2015). The parameters of different methods are set to be the default values that the authors suggested in their papers. We run each algorithm (except for COALA) ten times and the average values are reported. Because COALA is an hierarchical method and there is no randomness in the agglomerative clustering process, we just run it once. We choose three datasets from the UCI KDD repository (Asuncion and Newman 2007), i.e., Pen digit dataset, CMUFace and Semeion handwritten dataset, and a text dataset—Webkb dataset to conduct the experiments.

4.1 Clustering measurements

The clustering results are evaluated by the quality and the dissimilarity of the clusterings. For clustering quality, we use the Dunn Index denoted as DI, which measures the minimum distance between clusters normalized by the maximum cluster diameter. Mathematically, the Dunn Index is defined by: $DI(C) = \frac{\min_{i \neq j}\{\delta(c_i, c_j)\}}{x_{1 \leq l \leq k}\{\Delta(c_l)\}}$ with $\delta : C \times C \to \mathbb{R}_0^+$ is the distance between different clusters and $\Delta : C \to \mathbb{R}_0^+$ is the diameter of one of the clusters. The bigger the Dunn Index is, the higher the quality of the clustering is.

For measuring the dissimilarity between alternative clusterings, there exist a large number of measurements. In our experiments, we choose Rand Index (RI) (Rand 1971), Adjusted Rand Index (AR) (Hubert and Arabie 1985), Jaccard Index (JI) (Hamers et al. 1989), Mutual Information (MI) and Normalized Mutual Information (NMI) (Meilă 2007) to measure the dissimilarity between different clusterings. Notice that different from the Dunn Index, all these dissimilarity measurements are desired to be smaller, which indicates higher dissimilarity between clusterings.

4.2 Impact of parameter λ

Before showing the capability of MNMF to generate multiple clusterings, we need to choose a proper value of λ which controls the trade off between the quality and dissimilarity of the clusterings generated by MNMF. We apply MNMF to CMUFace and Webkb with the varying value of λ from 0 to 0.2. To compare the impact of different values of λ, we use Dunn

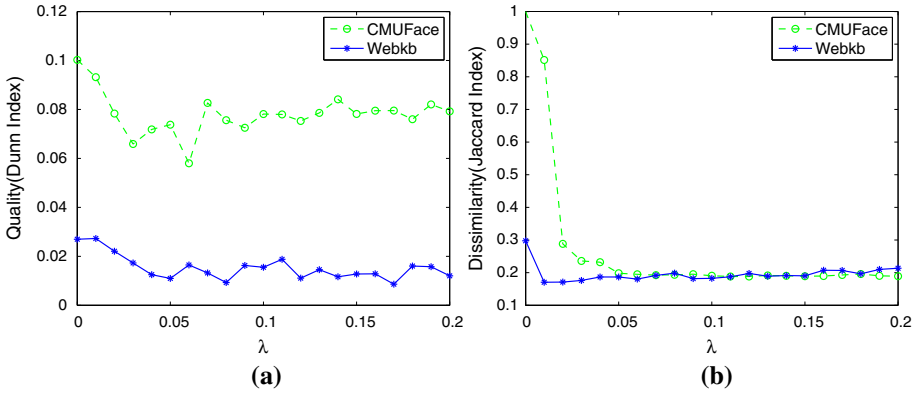

Fig. 2 Quality and dissimilarity given by Dunn Index and Jaccard Index as the value of λ changes

Index as the quality measurement and Jaccard Index as the dissimilarity measurement which is the same as Bae and Bailey (2006). Fig. 2a shows that the quality of the novelly generated clustering decreases as the value of λ increases and Fig. 2b shows that the dissimilarity of the novelly generated clustering increases as the value of λ increases. However, we hope the quality of the novel clustering to be high and the dissimilarity of the novel clustering also to be high, which means that we need a big Dunn Index value and a small Jaccard Index value. The behaviors of the curves imply that we need to make a trade off between these two measurement. From Fig. 2, we can see that $\lambda = 0.1$ could be a good choice, so we choose 0.1 as the default value of λ in the following experiments.

4.3 Pen digit dataset

Pen digit dataset obtained from the UCI KDD repository consists of handwritten digits recorded on a pen-based pressure sensitive tablet. Each instance corresponds to a single digit from 0 to 9 and has 16 attributes, which represent the 8 two-dimensional positions of the pen as the digit is being written. Each pair of co-ordinates is sampled as the digit is being written. Users are free to write the digits in any form that they are accustomed to. Certainly, the most prominent partition over this dataset is the one based on the ten digits. Nonetheless, for the purpose of generating multiple clusterings, we just take care of the ways that the digits have been written rather than which digit the instances belong to. And we will illustrate how our algorithm can interpret the ways that the digits have been written, analogous to the result in Davidson and Qi (2008).

We set the cluster number k to be 2 and run k-means on the original pen dataset to obtain the first clustering C_1. Each group's centroids are shown in Fig. 3. As seen from the first clustering C_1, the writing style of the digits seems to follow clockwise trend with slightly constant speed but having a slow speed for initial strokes and increasingly high speed for later strokes. Notice the time interval between any two adjacent points is the same, so a shorter distance between two adjacent points indicates a slower writing speed and inversely, a longer one reveals a faster speed of strokes' writing.

With the reference clustering C_1, we apply our algorithm to generate an alternative clustering C_2. The centroids of each cluster are shown in Fig. 4. The writing style of the digits in clustering C_2 also seems to follow clockwise trend, but the distribution of the speed is quite different from clustering C_1. In Fig. 4a, the writers start with a slow speed and increase

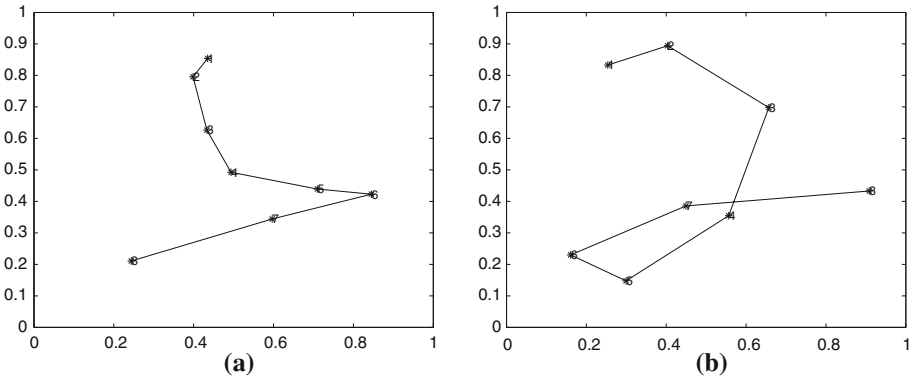

Fig. 3 Reference clustering C_1

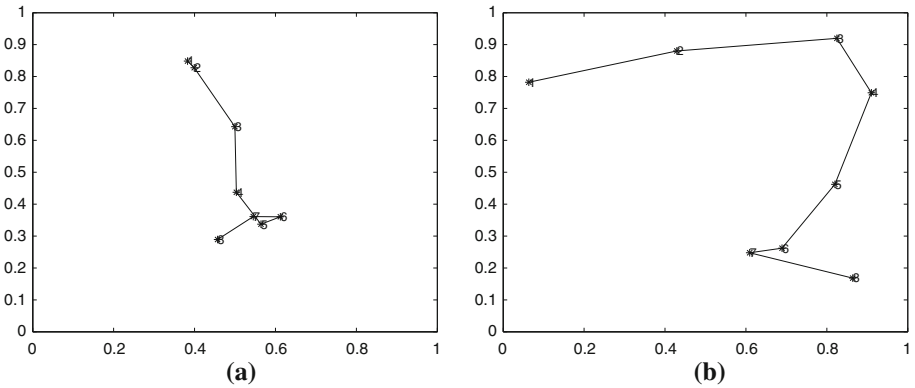

Fig. 4 One alternative clustering C_2 generated by MNMF

Table 1 One alternative clustering on the Pen digit dataset

Method	RI	AR	MI	NMI	JI	DI
MNMF	**0.5007**	**0.0021**	**0.0003**	**0.0004**	0.3634	0.1476
Algo1	0.5040	0.0030	0.0086	0.0089	0.3689	**0.1542**
Algo2	0.5114	0.0099	0.0065	0.0068	0.3899	0.1444
ADFT	0.5131	0.1020	0.1639	0.3093	**0.1455**	0.1431
RPCA	0.6791	0.3542	0.3494	0.3513	0.5859	0.1505
RegGB	0.6825	0.3516	0.3022	0.3206	0.5790	0.1257
MSC	0.5806	0.1270	0.1342	0.1624	0.4982	0.1072
SC	0.9930	0.5002	0.4330	0.4781	0.9930	0.0029

Bold values indicate the best performance of each column

writing speed greatly in the following stroke. Then the writers decrease the writing speed slowly until finishing writing the digits. In Fig. 4b, the writers start with a high speed and then decrease the writing speed slowly until the last stroke.

Quantitative results are provided in Table 1. We omit the result of COALA because it is very slow on this dataset. From this table, we can see that our method ranks first on most

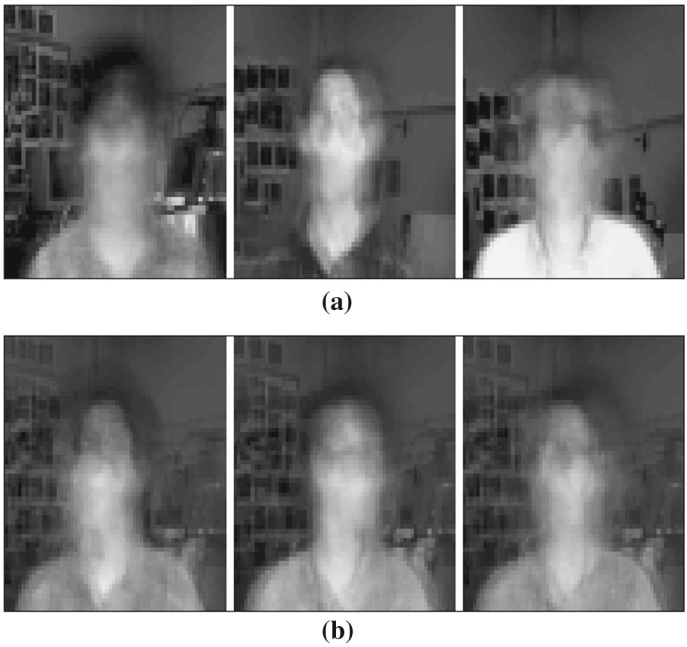

Fig. 5 MNMF's clustering result on the CMUFace dataset. **a** Cluster means of the reference clustering. **b** Cluster means of the alternative clustering generated by MNMF

of the dissimilarity measurements and ranks third on the quality measurement. Overall, our algorithm performs the best on the Pen digit dataset, which shows the strong capability of our method to generate no-redundant multiple clusterings.

4.4 CMUFace dataset

CMUFace dataset, which is also obtained from the UCI KDD repository, consists of images from 20 people taken with various features such as facial expressions (neutral, happy, sad, angry), head positions (left, right, or straight), and eye states (open or with sunglass). Each person has 32 images captured in every combination of these features. For this dataset, images can be partitioned by different ways easily (by individual, pose, etc). But the clustering result might be affected by the chosen number of clusters k. For example, if we set $k = 20$, the clustering algorithm tends to partition these images according to individuals. On the other hand, if we set $k = 3$, the clustering algorithm will partition the dataset based on head positions. In order to alleviate the effect of k on the alternative clustering result, we follow the setting of Dang and Bailey (2014) and randomly select 3 people along with all their images to create a smaller dataset. So, this subset can be partitioned into 3 clusters either by individuals or by head positions.

We use the partition based on individuals as the reference clustering and apply MNMF to find an alternative clustering of the dataset. For visualization purpose, we show the cluster means for each clustering in Fig. 5. While pictures in Fig. 5a correspond to the clustering based on different persons, the pictures in Fig. 5b correspond to the clustering based on different head positions. In Fig. 5b, the person in different pictures has different head positions. The person in the left image looks to his left while the person in the medium image looks to his

Table 2 One alternative clustering on the CMUFace dataset

Method	RI	AR	MI	NMI	JI	DI
MNMF	0.5550	**0.0106**	**0.0096**	**0.0096**	**0.1907**	**0.0776**
COALA	0.7939	0.5478	0.5817	0.6860	0.5446	0.0055
Algo1	**0.5266**	0.0154	0.0639	0.0713	0.2374	0.0312
Algo2	0.5419	0.0533	0.1068	0.1231	0.2544	0.0418
ADFT	0.7248	0.4496	0.5794	0.6583	0.5004	0.0158
RPCA	0.7272	0.4554	0.5794	0.6612	0.5048	0.0118
RegGB	0.5711	0.2017	0.3319	0.3980	0.3737	0.0134
MSC	0.7083	0.4251	0.4880	0.5325	0.5631	0.0293
SC	0.7432	0.4820	0.6367	0.6640	0.5828	0.0371

Bold values indicate the best performance of each column

right and the person in the right image looks forward. This alternative clustering provides a different yet meaningful interpretation about the data.

To make a comparison on this dataset, we report the dissimilarity and quality of the alternative clusterings generated by all the methods in Table 2. Because SC and MSC can't make use of the reference clustering, we simply apply them to generate two clusterings and report the dissimilarity and quality of the two clusterings. From this table, we can see our method achieves the best performance in the all the measurement except Rand Index in which our method ranks third. It proves that our method performs better than the others both on the dissimilarity measurement and the quality measurement. The result confirms the strong capability of MNMF to generate multiple non-redundant and meaningful clusterings.

4.5 Webkb dataset

Webkb dataset is a text dataset and the documents in Webkb are webpages collected by the World Wide Knowledge Base project of the CMU text learning group (Cardoso-Cachopo 2007). These pages were collected from computer science departments of various universities in 1997, manually classified into four different classes: student, faculty, course, and project. We extract 1000 features by TF-IDF for this document dataset. We use the given label as the reference label. This clustering is based upon where the document are collected. We apply our algorithm on this dataset to find the main topics.

To visualize the generated clustering, we need to identify the most informative words that characterise each cluster. Following the method in Dasgupta and Ng (2010), we rank them by their weighted log-likelihood ratio (WLLR): $P(w_i|\pi_j) \cdot log\frac{P(w_i|\pi_j)}{P(w_i|\neg\pi_j)}$, where w_i denotes the i-th feature and π_j denotes the j-th cluster. In addition, each probability is add-one smoothed. Informally, w_i will have a high rank with regard to π_j if it appears frequently in π_j and infrequently in $\neg\pi_j$. After ranking the correlated features, we select the top 10 words to represent the corresponding cluster. The result is shown in Fig. 6. In each clustering, the first row is the words we extracted to describe the clusters. From the alternative clustering, we can see that there are four main topics in the documents: software, hardware, military and research. It is not surprising that these documents mainly talk about computers and research since the documents come from the computer science departments of several universities.

	student	faculty	course	project
reference clustering C_1	compute	associate	assign	application
	depart	compute	class	base
	graduate	depart	exam	develop
	home	fax	final	faculty
	page	phone	grade	group
	research	professor	homework	include
	science	public	hour	project
	student	research	instructor	relate
	universe	science	lecture	research
	work	universe	syllabus	support
	software	**hardware**	**military**	**research**
alternative clustering C_2	application	architecture	alternative	compute
	environment	cache	army	depart
	implement	exploit	concern	home
	include	memory	director	inform
	language	multiprocessor	foundation	office
	level	perform	massive	page
	provide	processor	prerequisite	research
	support	scalable	serve	science
	type	share	theoretic	student
	user	supercompute	tradition	universe

Fig. 6 MNMF's clustering result on the Webkb dataset

Table 3 One alternative Clustering on the Webkb dataset

Method	RI	AR	MI	NMI	JI	DI
MNMF	0.6031	0.0344	0.0328	0.0335	**0.1854**	**0.0225**
COALA	**0.2887**	**0.0000**	**0.0010**	0.0120	0.2876	0.0000
Algo1	0.5569	0.0058	0.0124	0.0137	0.1948	0.0057
Algo2	0.5570	0.0071	0.0073	**0.0079**	0.1921	0.0074
ADFT	0.5685	0.1887	0.2053	0.2131	0.3097	0.0162
RPCA	0.6791	0.2228	0.3300	0.3348	0.2895	0.0165
RegGB	0.4315	0.0044	0.0154	0.0261	0.2489	0.0003
MSC	0.8612	0.2200	0.1694	0.2360	0.8577	0.0000
SC	0.9262	0.7278	0.7070	0.7609	0.9164	0.0000

Bold values indicate the best performance of each column

To make a comparison with the other methods, quantitative results are given in Table 3. For the same reason, we simply apply SC and MSC to generate two clusterings and report the dissimilarity and quality of the two clusterings. Although COALA seems to have higher dissimilarity between the clusterings than our methods, the quality of it is the worst. Although our method does't preform better than all the others, it achieves a good performance both in the dissimilarity measurement and quality measurement. Thus, MNMF can be used to find multiple clusterings on text corpus.

4.6 Semeion handwritten digit dataset

Semeion handwritten digit dataset is also obtained from the UCI KDD repository, consists of 1593 data samples, where each sample has 256 features. The samples come from around 80 people and the features are stretched in a rectangular box 16 × 16 in a gray scale of 256

Table 4 One alternative clustering on the Semeion handwritten digit dataset

Method	RI	AR	MI	NMI	JI	DI
MNMF	0.8206	**0.0005**	0.0109	**0.0113**	**0.0524**	**0.0067**
COALA	**0.4019**	0.0083	0.1033	0.1763	0.0981	0.0001
Algo1	0.8259	0.0584	0.1424	0.1433	0.0842	0.0038
Algo2	0.8287	0.0585	0.1322	0.1368	0.0833	0.0047
ADFT	0.8859	0.3716	0.5175	0.5184	0.2812	0.0044
RPCA	0.8395	0.1565	0.3025	0.3265	0.1401	0.0023
RegGB	0.8862	0.4414	0.5867	0.6202	0.3375	0.0015
MSC	0.8703	0.4325	0.5646	0.5740	0.3397	0.0013
SC	0.4557	0.0014	**0.0085**	0.0201	0.1091	0.0012

Bold values indicate the best performance of each column

Table 5 Two alternative clusterings on the Semeion handwritten digit dataset

Method	JI_{12}	JI_{13}	JI_{23}	NMI_{12}	NMI_{13}	NMI_{23}	DI_2	DI_3
MNMF	**0.0558**	0.0872	0.0924	0.0154	0.0367	**0.0295**	**0.0046**	0.0005
COALA	0.0981	0.0815	0.2438	0.1763	0.1095	0.4466	0.0000	0.0000
Algo1	0.0880	**0.0737**	**0.0632**	0.1562	0.0928	0.0406	0.0035	0.0029
Algo2	0.0816	0.0765	0.0669	0.1316	0.1043	0.0564	0.0037	0.0025
ADFT	0.2928	0.3114	0.3904	0.5366	0.5419	0.6294	0.0033	**0.0037**
RPCA	0.1409	0.1373	0.5769	0.3264	0.3173	0.8041	0.0027	0.0023
RegGB	0.3363	0.3627	0.6237	0.6202	0.6363	0.8250	0.0011	0.0023
MSC	0.2565	0.1760	0.1962	0.5305	0.3745	0.4167	0.0012	0.0010
SC	0.1117	0.0977	0.1644	**0.0142**	**0.0112**	0.3122	0.0013	0.0028

JI_{ij} stands for the JI between C_i and C_j clusterings. The same interpretation is applied to NMI_{ij} and DI_i
Bold values indicate the best performance of each column

values. This is also a dataset of handwritten digits, but each observation in it comprises a digit image, rather than the co-ordinates of the pen. Since it has been labeled by different digits, we just use the given label as the reference clustering to generate alternative clusterings.

First, we only generate one alternative clustering on Semeion handwritten digit dataset and the experimental result is in Table 4. We can see that our method MNMF has the highest Dunn Index, and most of the dissimilarity measurements are the smallest among them.

Then, we use each method to generate two alternative clusterings on this dataset and compare the dissimilarity and the quality between each pair of the different clusterings. In this experiment, we use JI and NMI to measure the dissimilarity between clusterings and the result is shown in Table 5. Recall that COALA and ADFT cannot discover multiple alternative clusterings. For these algorithms, the results related to C_3 in Table 5, are computed by providing clustering C_2 as the reference clustering. Such methods have the risk that the similarity between C_1 and C_3 may be much higher than the others. However, our method can generate a novel clustering with the reference of all the existing clusterings and can avoid this case. For SC and MSC which can't make use of the given label, we just use them to generate three clusterings. From this table, there are 3 measurements on which our method ranks first and 2 measurements on which our method ranks second, which shows that our method can perform better than the others.

5 Conclusion

Clustering is one of the fundamental techniques of data analysis. In this paper, we propose a novel method named MNMF to generate multiple clusterings that are both of high quality and different from each other. It is based upon NMF and we incorporate the inner product of similarity matrices corresponding to different clusterings as a regularization term to the original cost function. This regularization term is meaningful due to the nonnegative constraint, and can enforce a newly-generated clustering to be different from the reference clustering. In order to solve this problem, we design two multiplicative updating rules. Our approach is effective, with a computational complexity of $O(t(mnk + n^2))$ where t denotes the number of iterations, m denotes the dimension, n denotes the number of samples and k denotes the rank of U and V. In addition, the experimental results have shown the appealing performance of MNMF when dealing with the text document datasets and image datasets.

In the experiments, we use Dunn Index as the quality measurement. Although Dunn Index has been effective for measuring quality, it is known to be sensitive to outliers and prefers compact and well-separated clusters (Bezdek and Pal 1998). In the future, we would like to investigate other unsupervised measurements for validating quality. Another future work is to design stopping criterion for multiple clusterings. In addition, it is also interesting to study whether the nonnegative constraints can be relaxed (Ding et al. 2010).

Acknowledgments This work was partially supported by the NSFC (61603177, 61422304), JiangsuFS (BK20160658), and the Collaborative Innovation Center of Novel Software Technology and Industrialization of Nanjing University.

References

Asuncion, A., & Newman, D. (2007). UCI machine learning repository.

Bae, E., & Bailey, J. (2006). Coala: A novel approach for the extraction of an alternate clustering of high quality and high dissimilarity. In *International Conference on Data Mining*, (pp. 53–62).

Baudat, G., & Anouar, F. (2000). Generalized discriminant analysis using a kernel approach. *Neural Computation*, *12*(10), 2385–2404.

Belkin, A., & Niyogi, P. (2001). Laplacian eigenmaps and spectral techniques for embedding and clustering. *Advances in Neural Information Processing Systems*, *14*, 585–591.

Bezdek, J. C., & Pal, N. R. (1998). Some new indexes of cluster validity. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, *28*(3), 301–315.

Cai, D., He, X., Han, J., & Huang, T. S. (2011). Graph regularized nonnegative matrix factorization for data representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *33*(8), 1548–1560.

Cardoso-Cachopo, A. (2007). *Improving methods for single-label text categorization*. PdD Thesis, Instituto Superior Tecnico, Universidade Tecnica de Lisboa.

Caruana, R., Elhawary, M.F., Nguyen, N., & Smith, C. (2006). Meta clustering. In *International Conference on Data Mining*, (pp. 107–118).

Cui, Y., Fern, X.Z., Dy, J.G. (2007). Non-redundant multi-view clustering via orthogonalization. In *International Conference on Data Mining*, (pp. 133–142).

Dang, X.H., & Bailey, J. (2010). Generation of alternative clusterings using the CAMI approach. In *International Conference on Data Mining*, (pp. 118–129).

Dang, X. H., & Bailey, J. (2014). Generating multiple alternative clusterings via globally optimal subspaces. *Data Mining and Knowledge Discovery*, *28*(3), 569–592.

Dang, X. H., & Bailey, J. (2015). A framework to uncover multiple alternative clusterings. *Machine Learning*, *98*(1–2), 7–30.

Dasgupta, S., & Ng, V. (2010). Mining clustering dimensions. In *International Conference on Machine Learning*, (pp. 263–270).

Davidson, I., & Qi, Z. (2008). Finding alternative clusterings using constraints. In *International Conference on Data Mining*, (pp. 773–778).

Ding, C. H. Q., Li, T., & Jordan, M. I. (2010). Convex and semi-nonnegative matrix factorizations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *32*(1), 45–55.

Gondek, D., & Hofmann, T. (2003). Conditional information bottleneck clustering. In *International Conference on Data Mining*, (pp. 36–42).

Gretton, A., Bousquet, O., Smola, A.J., Schölkopf, B. (2005). Measuring statistical dependence with hilbert-schmidt norms. In *International conference on Algorithmic Learning Theory*, (pp. 63–77).

Hamers, L., Hemeryck, Y., Herweyers, G., Janssen, M., Keters, H., Rousseau, R., et al. (1989). Similarity measures in scientometric research: The jaccard index versus salton's cosine formula. *Information Processing and Management*, *25*(3), 315–318.

Hu, J., Qian, Q., Pei, J., Jin, R., & Zhu, S. (2015). Finding multiple stable clusterings. In *International Conference on Data Mining*, (pp. 171–180).

Hubert, L., & Arabie, P. (1985). Comparing partitions. *Journal of Classification*, *2*(1), 193–218.

Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: A review. *ACM Computing Surveys*, *31*(3), 264–323.

Jain, P., Meka, R., & Dhillon, I. S. (2008). Simultaneous unsupervised learning of disparate clusterings. *Statistical Analysis and Data Mining*, *1*(3), 195–210.

Lee, D. D., & Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, *401*(6755), 788–791.

Lee, D.D., & Seung, H.S. (2001). Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems*, (pp. 556–562).

Lin, C. (2007). On the convergence of multiplicative update algorithms for nonnegative matrix factorization. *IEEE Trans Neural Networks*, *18*(6), 1589–1596.

Meilă, M. (2007). Comparing clusteringsan information based distance. *Journal of Multivariate Analysis*, *98*(5), 873–895.

Niu, D., Dy, J.G., & Jordan, M.I. (2010). Multiple non-redundant spectral clustering views. In *International Conference on Machine Learning*, (pp. 831–838).

Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, *66*(336), 846–850.

Xing, E. P., Ng, A. Y., Jordan, M. I., & Russell, S. (2003). Distance metric learning with application to clustering with side-information. *Advances in Neural Information Processing Systems*, *15*, 505–512.

Xu, W., Liu, X., Gong, Y. (2003). Document clustering based on non-negative matrix factorization. In *Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, (pp. 267–273).

Yang, S., Yi, Z., Ye, M., & He, X. (2014). Convergence analysis of graph regularized non-negative matrix factorization. *IEEE Transactions on Knowledge and Data Engineering*, *26*(9), 2151–2165.