Multi-View Clustering via Joint Nonnegative Matrix Factorization

Jialu Liu[1], Chi Wang[1], Jing Gao[2], and Jiawei Han[1]

[1]University of Illinois at Urbana-Champaign
[2]University at Buffalo

Abstract

Many real-world datasets are comprised of different representations or views which often provide information complementary to each other. To integrate information from multiple views in the unsupervised setting, multi-view clustering algorithms have been developed to cluster multiple views simultaneously to derive a solution which uncovers the common latent structure shared by multiple views. In this paper, we propose a novel NMF-based multi-view clustering algorithm by searching for a factorization that gives compatible clustering solutions across multiple views. The key idea is to formulate a joint matrix factorization process with the constraint that pushes clustering solution of each view towards a common consensus instead of fixing it directly. The main challenge is how to keep clustering solutions across different views meaningful and comparable. To tackle this challenge, we design a novel and effective normalization strategy inspired by the connection between NMF and PLSA. Experimental results on synthetic and several real datasets demonstrate the effectiveness of our approach.

1 Introduction

Many datasets in real world are naturally comprised of different representations or *views* [5]. For example, the same story can be told in articles from different news sources, one document may be translated into multiple different languages, research communities are formed based on research topics as well as co-authorship links, web pages can be classified based on both content and anchor text leading to hyperlinks, and so on. In these applications, each data set is represented by attributes that can naturally be split into different subsets, any of which suffices for mining knowledge. Observing that these multiple representations often provide compatible and complementary information, it becomes natural for one to integrate them together to obtain better performance rather than relying on a single view. The key of learning from multiple views (*multi-view*) is to leverage each view's own knowledge base in order to outperform simply concatenating views.

As unlabeled data are plentiful in real life and increasing quantities of them come in multiple views from diverse sources, the problem of unsupervised learning from multiple views of unlabeled data has attracted attention [3, 17], referred to as multi-view clustering. The goal of multi-view clustering is to partition objects into clusters based on multiple representations of the object. Existing multi-view clustering algorithms can be roughly classified into three categories. Algorithms in the first category [3, 17] incorporate multi-view integration into the clustering process directly through optimizing certain loss functions. In contrast, algorithms in the second category such as the ones based on Canonical Correlation Analysis [8, 4] first project multi-view data into a common lower dimensional subspace and then apply any clustering algorithm such as k-means to learn the partition. The third category is called *late integration* or *late fusion*, in which a clustering solution is derived from each individual view and then all the solutions are fused base on consensus [7, 13].

In this paper, we propose a new multi-view clustering approach based on a highly effective technique in single-view clustering, i.e., non-negative matrix factorization (NMF) [18]. NMF, which was originally introduced as a dimensionality reduction technique [18], has been shown to be useful in many research areas such as information retrieval [20] and pattern recognition [18]. NMF has received much attention because of its straightforward interpretability for applications, i.e., we can explain each observation as an additive linear combinations of nonnegative basis vectors. Recently, NMF has become a popular technique for data clustering, and it is reported to achieve competitive performance compared with most of the state-of-the-art unsupervised algorithms. For example, Xu et al. [20] applied NMF to text clustering and gained superior performance, and Brunet et al. [6] achieved similar success on biological data clustering. Recent studies [9, 11] show that NMF is closely related to Probabilistic Latent Semantic Analysis (PLSA) [15] which is one of the most popular topic modeling algorithms.

As NMF has shown to generate superior clustering results which are easy to interpret, it will be very useful to have an NMF-based multi-view clustering approach. However, studies on NMF-based multi-view approaches for clustering are still limited. The main challenge of applying NMF to multi-view clustering is how to limit the search of factorizations to those that give meaningful and comparable clustering solutions across multiple views simultaneously. Moreover, traditional normalization strategies proposed for standard NMF are either difficult to be optimized in the multi-view setting [20], or cannot generate meaningful clustering results [21, 10]. In this paper, we approach this problem by proposing a novel normalization strategy and following the principle that factors representing clustering structures learnt from multiple views should be regularized toward a common consensus.

It is worthwhile to highlight several advantages of the proposed approach as follows:

1. As far as we know, this is the first exploration towards a multi-view clustering approach based on joint nonnegative matrix factorization, which is different from traditional approaches simply fixing the shared one-side factor among multiple views.
2. As discussed, existing normalization strategies for standard NMF cannot keep factors from different views comparable and meaningful in the multi-view setting for clustering, making the fusion of views difficult and inconclusive. To tackle this challenge, we develop a novel normalization procedure inspired from connection between NMF and PLSA.
3. We propose an iterative optimization framework which is scalable and convergent. In terms of accuracy, it outperforms the state-of-the-art algorithms in our experiments by 6% on average.

The rest of this paper is organized as follows. In the next section, a brief overview of NMF and its relationship with PLSA is provided. The proposed multi-view NMF algorithm is then presented in Section 3. Extensive experimental results are shown in Section 4. A discussion of related work is given in Section 5. Finally, in Section 6 we provide conclusions.

2 Overview of NMF and PLSA

In this section, we briefly introduce Non-Negative Matrix Factorization (NMF) [18] and its relationship [9, 11] with Probabilistic Latent Semantic Analysis (PLSA) [15]. In the next section, this relationship inspires us to develop the joint non-negative matrix factorization framework for multi-view clustering.

Let $X = [X_{\cdot,1}, \ldots, X_{\cdot,N}] \in \mathbb{R}_+^{M\times N}$ denote the nonnegative data matrix where each column represents a data point and each row represents one attribute. NMF aims to find two non-negative matrix factors $U = [U_{i,k}] \in \mathbb{R}_+^{M\times K}$ and $V = [V_{j,k}] \in \mathbb{R}_+^{N\times K}$ whose product provides a good approximation to X:

$$X \approx UV^T. \tag{2.1}$$

Here K denotes the desired reduced dimension, and to facilitate discussions, we call U the *basis matrix* and V the *coefficient matrix*.

We can also view this approximation column by column:

$$X_j \approx U(V_{j,\cdot})^T = \sum_{k=1}^{K} U_{\cdot,k} V_{j,k}$$

where $U_{\cdot,k}$ is the k-th column vector of U and $V_{j,\cdot}$ is the j-th row vector of V. Correspondingly, we call $U_{\cdot,k}$ the *basis vector* and $V_{j,\cdot}$ the *coefficient vector*.

One of the common reconstruction processes can be formulated as a *Frobenius norm* optimization problem, defined as:

$$\min_{U,V} ||X - UV^T||_F^2, \ s.t. \ U \geq 0, V \geq 0$$

where $||\cdot||_F$ is the Frobenius norm and $U \geq 0$, $V \geq 0$ represent the constraints that all the matrix elements are non-negative. It is known that the objective function above is not convex in U and V together. Therefore, it is unrealistic to expect an algorithm to find the global minimum. [18] presented "multiplicative update rules" to be executed iteratively to minimize the objective function as follows:

$$U_{i,k} \leftarrow U_{i,k}\frac{(XV)_{i,k}}{(UV^TV)_{i,k}}, \ V_{j,k} \leftarrow V_{j,k}\frac{(X^TU)_{j,k}}{(VU^TU)_{j,k}}. \tag{2.2}$$

Note that given the NMF formulation in Equation 2.1, for arbitrary invertible $K \times K$ matrix Q, we have

$$UV^T = (UQ^{-1})(QV^T). \tag{2.3}$$

Therefore there could be many possible solutions, and it is important to enforce additional constraints to ensure the uniqueness of the factorization in clustering.

We next introduce NMF's relationship with Probabilistic Latent Semantic Analysis (PLSA) [15] and this connection helps clustering. PLSA is a traditional topic modeling technique for document analysis. It models the $M \times N$ term-document co-occurrence matrix X (each entry X_{ij} is the number of occurrences of word w_i in document d_j) as being generated from a mixture model with K components:

$$P(w,d) = \sum_{k=1}^{K} P(w|k)P(d,k)$$

where parameters are estimated by maximizing the likelihood through Expectation Maximization algorithm.

Without loss of generality, we assume $||X||_1 = 1$. In Eq. 2.2, if Q is a diagonal matrix where $Q_{k,k} = \sum_i U_{i,k}$, earlier studies [9, 11] revealed that (UQ^{-1}) (or (QV^T)) has all the formal properties of conditional probability matrix $[P(w|k)] \in \mathbb{R}_+^{M\times K}$ (or $[P(d,k)]^T \in \mathbb{R}_+^{K\times N}$). This provides theoretical foundation for using NMF to conduct clustering. Specifically, in the clustering formulation, each row in V indicates to which degree data point i is associated with cluster k. Note that $||QV^T||_1$ approximately equals to 1 because $\sum P(d,k) = 1$. This property can also be verified from the following equation:

$$||X|| = ||\sum_j X_j|| \approx \sum_{k=1}^{K} ||U_{\cdot,k} \sum_j V_{j,k}|| = \sum_{k=1}^{K} ||\sum_j V_{j,k}|| = ||V||.$$

Now it is clear that QV^T behaves similar to a cluster indicator matrix where each column of it describes the joint probability of one data point and different clusters.

3 Multi-View NMF

In this section, we present the proposed joint matrix factorization formulation for multi-view clustering and effective iterative update rules to solve the optimization problem. The basic idea is as follows. For clustering, we assume that a data point in different views would be assigned to the same cluster with high probability. Therefore, in terms of matrix factorization, we require coefficient matrices learnt from different views to be softly regularized towards a common consensus. This consensus matrix is considered to reflect the latent clustering structure shared by different views.

Additionally, to ensure the uniqueness and correctness of factorizations, we need to adopt some normalization strategies during the optimization. However, traditional strategies proposed for standard NMF are either difficult to be optimized in the multi-view setting [20], or cannot generate meaningful clustering solutions [21, 10, 14], making the fusion of different views difficult and inconclusive. In light of this challenge, we design a novel normalization procedure, which can successfully solve the problem. Specifically, we do ℓ_1 normalization with respect to the basis vectors during the process of optimization[1]. This is inspired by the relationship between NMF and PLSA introduced in the last section, from which we can provide coefficient matrices of different views with probabilistic explanation, rendering them comparable during optimization and meaningful for clustering.

3.1 Objective function Assume that we are now given n_v representations (i.e., views). Let $\{X^{(1)}, X^{(2)}, \ldots, X^{(n_v)}\}$ denote the data of all the views, where for each view $X^{(v)}$, we have factorizations that $X^{(v)} \approx U^{(v)}(V^{(v)})^T$. In standard NMF, coefficient vector $V_{j,\cdot}^{(v)}$ can be regarded as low-rank representation of the jth data point in terms of the new basis $U^{(v)}$. Here for different views, we have the same number of data points but allow for different number of attributes, hence $V^{(v)}$s are of the same shape but $U^{(v)}$s can differ along the row dimension across multiple views.

The following loss function is used as a measure of disagreement between coefficient matrix $V^{(v)}$ and the consensus matrix V^*:

$$D(V^{(v)}, V^*) = \|V^{(v)} - V^*\|_F^2.$$

Note that $V^{(v)}$ in different views might not be comparable at the same scale and they are not meaningful for clustering since the product of $V^{(v)}$ and an arbitrary invertible matrix $Q^{(v)}$ could still be a solution according to Eq. 2.3. To make the disagreement measure proper for different $V^{(v)}$ against the same consensus and let them theoretically meaningful for clustering, a possible solution is to normalize $V^{(v)}$ and then compute the distance measure. However, such a normalization can make the optimization intractable. To solve the problem, we propose to impose the ℓ_1 normalization with respect to the basis vectors $U_{\cdot,k}^{(v)}$. In this way, $||V^{(v)}||_1$ approximately equals to 1 under the assumption that $||X||_1 = 1$ which is discussed in last section. As $V^{(v)}$ is within the same range for different v, we can ensure the comparison between the coefficient matrix $V^{(v)}$ and consensus matrix V^* is reasonable. Additionally, after normalization, each element in $V^{(v)}$ has probabilistic explanation because it can be viewed as $P(d,k)^{(v)}$, making the consensus V^* meaningful in terms of clustering, i.e., each element in the matrix V^* is the consensus of $P(k|d)^{(v)}$ weighted by $P(d)^{(v)}$ from different views.

Incorporating this idea into the NMF framework for individual views, we obtain the following joint minimization problem over $U^{(v)}, V^{(v)}, V^*, 1 \le v \le n_v$:

$$\sum_{v=1}^{n_v} \|X^{(v)} - U^{(v)}(V^{(v)})^T\|_F^2 + \sum_{v=1}^{n_v} \lambda_v \|V^{(v)} - V^*\|_F^2$$

(3.4)

$$s.t.\ \forall 1 \le k \le K, ||U_{\cdot,k}^{(v)}||_1 = 1 \text{ and } U^{(v)}, V^{(v)}, V^* \ge 0$$

where λ_v is the only parameter within the proposed algorithm, which not only tunes the relative weight among different views, but also between standard NMF reconstruction error and disagreement term $D(V^{(v)}, V^*)$.

[1] Within traditional normalization for NMF, ℓ_2 norm or other constraints are adopted and this step is always done during post processing.

The selection of appropriate λ_v will be discussed in the experimental section.

Then the equality constraint on $U^{(v)}$ can be removed by introducing auxiliary variables to simplify the computation. Let

$$Q^{(v)} = Diag\left(\sum_{i=1}^{M} U_{i,1}^{(v)}, \sum_{i=1}^{M} U_{i,2}^{(v)}, \ldots, \sum_{i=1}^{M} U_{i,K}^{(v)}\right) \tag{3.5}$$

where $Diag(\cdot)$ denotes a diagonal matrix with non-zero elements equal to the values in the parenthesis sequentially. According to Eq. 2.3, the problem of minimizing Eq. 3.4 is equivalent to minimizing the following objective function O:

$$\begin{aligned} O = &\sum_{v=1}^{n_v} \|X^{(v)} - U^{(v)}(V^{(v)})^T\|_F^2 \\ &+ \sum_{v=1}^{n_v} \lambda_v \|V^{(v)}Q^{(v)} - V^*\|_F^2 \\ &s.t.\ \forall 1 \le v \le n_v, U^{(v)} \ge 0, V^{(v)} \ge 0, V^* \ge 0. \end{aligned} \tag{3.6}$$

To solve this optimization problem, we propose an iterative update procedure which is convergent. Specifically, the following two steps are repeated until convergence: (1) fixing V^*, minimize O over $U^{(v)}$ and $V^{(v)}$ (Section 3.2), and (2) fixing $U^{(v)}$ and $V^{(v)}$, minimize O over V^* (Section 3.3).

3.2 Fixing V^*, minimize O over $U^{(v)}$ and $V^{(v)}$

When V^* is fixed, for each given v, the computation of $U^{(v)}$ does not depend on $U^{(v')}$ or $V^{(v')}, v' \neq v$. Therefore, we use X, U, V and Q to represent $X^{(v)}, U^{(v)}, V^{(v)}$ and $Q^{(v)}$ for brevity in this subsection. Now Eq. 3.6 reduces to minimize:

$$\|X - UV^T\|_F^2 + \lambda_v \|VQ - V^*\|_F^2 \ s.t.\ U, V \ge 0. \tag{3.7}$$

Then the following multiplicative updating rules for U and V can be used to update their values sequentially and iteratively.

3.2.1 Fixing V^* and $V^{(v)}$, compute $U^{(v)}$

Let Ψ be the Lagrange multiplier matrix for the constraint $U \ge 0$, and L be the Lagrange $L = O + Tr(\Psi U)$, where $Tr(\cdot)$ is the trace function. We only care about terms that are relevant to $U^{(v)}$ at this step, and thus minimizing L is equivalent to minimizing L_1 as follows:

$$L_1 = Tr(UV^TVU^T - 2XVU^T) + \lambda_v R + Tr(\Psi U)$$

Algorithm 1 Multi-View NMF (MultiNMF)

```
Input: Nonnegative Matrix $\{X^{(1)}, X^{(2)}, \ldots, X^{(n_v)}\}$,
    parameters $\{\lambda_1, \lambda_2, \ldots, \lambda_{n_v}\}$, number of clusters $K$
Output: Basis Matrices $\{U^{(1)}, U^{(2)}, \ldots, U^{(n_v)}\}$, Coefficient Matrices $\{V^{(1)}, V^{(2)}, \ldots, V^{(n_v)}\}$ and Consensus Matrix $V^*$
Normalize each view $X^{(v)}$ such that $||X^{(v)}||_1 = 1$
Initialize $U^{(v)}$,$V^{(v)}$ and $U^*$ $(1 \le v \le n_v)$
repeat
    for $v = 1$ to $n_v$ do
        repeat
            Fixing $V^*$ and $V^{(v)}$, update $U^{(v)}$ by Eq. 3.8
            Normalize $U^{(v)}$ and $V^{(v)}$ as in Eq. 3.9
            Fixing $V^*$ and $U^{(v)}$, update $V^{(v)}$ by Eq. 3.10
        until Eq. 3.7 converges.
    end for
    Fixing $U^{(v)}$ and $V^{(v)}$ $(1 \le v \le n_v)$, update $V^*$ by Eq. 3.11.
until Eq. 3.6 converges.
```

where $R = Tr(VQQ^TV^T - 2VQ(V^*)^T)$ contains the relevant terms in the regularizer $\|VQ - V^*\|_F^2$. With substitution of Eq. 3.5, we have

$$\begin{aligned} R = &\sum_{j=1}^{N}\sum_{k=1}^{K}(V_{j,k}\sum_{i=1}^{M}U_{i,k}\sum_{i=1}^{M}U_{i,k}V_{j,k}) \\ &- \sum_{j=1}^{N}\sum_{k=1}^{K}(V_{j,k}\sum_{i=1}^{M}U_{i,k}V_{j,k}^*). \end{aligned}$$

Taking derivative of R with respect to U gives

$$P_{i,k} = \frac{\partial R}{\partial U_{i,k}} = 2\left(\sum_{l=1}^{M} U_{l,k} \sum_{j=1}^{N} V_{j,k}^2 - \sum_{j=1}^{N} V_{j,k}V_{j,k}^*\right).$$

Using Karush-Kuhn-Tucker (KKT) conditions, we have

$$\begin{aligned} \frac{\partial L_1}{\partial U} &= -2XV + 2UV^TV + \lambda_v P + \Psi = 0 \\ \Psi_{i,k}U_{i,k} &= 0, \forall 1 \le i \le M, 1 \le k \le K. \end{aligned}$$

Based on this condition, we can derive the following update rule:

$$U_{i,k} \leftarrow U_{i,k} \frac{(XV)_{i,k} + \lambda_v \sum_{j=1}^{N} V_{j,k}V_{j,k}^*}{(UV^TV)_{i,k} + \lambda_v \sum_{l=1}^{M} U_{l,k} \sum_{j=1}^{N} V_{j,k}^2}. \tag{3.8}$$

It is easy to see that $U_{i,k}$ remains non-negative after each update.

3.2.2 Fixing V^* and $U^{(v)}$, compute $V^{(v)}$

For each $1 \le v \le n_v$, we first normalize the column vectors of U using Q as in Eq. 3.5:

$$U \leftarrow UQ^{-1},\ V \leftarrow VQ. \tag{3.9}$$

Table 1: Computational operation counts for each iteration in MultiNMF

	addition	multiplication	division	overall
NMF: U	$MNK + (M+N)K^2$	$MNK + (M+N)K^2$	MK	$O(MNK)$
NMF: V	$MNK + (M+N)K^2$	$MNK + (M+N)K^2$	NK	$O(MNK)$
MultiNMF: U	$MNK + (M+N)K^2 + (2N+3M-3)K$	$MNK + (M+N)K^2 + (2N+3)K$	MK	$O(MNK)$
MultiNMF: Q	$(M-1)K$	$MK + NK$	K	$O(MK+NK)$
MultiNMF: V	$MNK + (M+N)K^2 + 2NK$	$MNK + (M+N)K^2 + 2NK$	NK	$O(MNK)$

[NMF: U] Update rule for basis matrix U in NMF (similar meaning for MultiNMF: U)
[NMF: V] Update rule for coefficient matrix V in NMF (similar meaning for MultiNMF: V)
[MultiNMF: Q] Normalization step as in Eq. 3.9

Note that this normalization does not change the value of Eq. 3.7, and now we just need to minimize $\|X - UV^T\|_F^2 + \lambda_v\|V - V^*\|_F^2$ with the constraint $V \geq 0$. Let Φ be the Lagrange multiplier matrix for the constraint $V \geq 0$. The Lagrange writes as:

$$\begin{aligned} L_2 = & Tr(UV^TVU^T - 2XVU^T) \\ & + \lambda_v Tr(VV^T - 2V(V^*)^T) + Tr(\Phi V). \end{aligned}$$

Using KKT conditions, we have:

$$\begin{aligned} \frac{\partial L_2}{\partial V} &= 2VU^TU - 2X^TU + 2\lambda_v(V - V^*) + \Phi = 0 \\ \Phi_{j,k}V_{j,k} &= 0, \forall 1 \leq j \leq N, 1 \leq k \leq K. \end{aligned}$$

The solution leads to the following update rule:

$$V_{j,k} \leftarrow V_{j,k}\frac{(X^TU)_{j,k} + \lambda_v V^*_{j,k}}{(VU^TU)_{j,k} + \lambda_v V_{j,k}}. \tag{3.10}$$

3.3 Fixing $U^{(v)}$ and $V^{(v)}$, minimize O over V^*.

We take the derivative of the objective function O in Eq. 3.6 over V^*:

$$\begin{aligned} \frac{\partial O}{\partial V^*} = & \frac{\partial \sum_{v=1}^{n_v} \lambda_v \|V^{(v)}Q^{(v)} - V^*\|_F^2}{\partial V^*} \\ = & \sum_{v=1}^{n_v} \lambda_v(-2V^{(v)} + 2V^*) = 0. \end{aligned}$$

Solving it, we have an exact solution for V^*:

$$V^* = \frac{\sum_{v=1}^{n_v} \lambda_v V^{(v)}Q^{(v)}}{\sum_{v=1}^{n_v} \lambda_v} \geq 0. \tag{3.11}$$

The two-step procedure is summarized in Algorithm 1. We call the proposed algorithm **MultiNMF**, which stands for a joint non-negative matrix factorization procedure for multi-view clustering. Note that once we obtain the consensus matrix V^*, the cluster label of data point i could be computed as $\arg\max_k V^*_{i,k}$.

Due to the connection between NMF and PLSA, the proposed algorithm has a nice probabilistic interpretation: each element in the matrix V^* is the consensus of $P(d|k)^{(v)}$ weighted by $P(d)^{(v)}$ from different views. Therefore, it is especially effective for data collections in which data points from different clusters do not lie along the same direction in the vector space. For example, as shown in topic modeling algorithms, document-word co-occurrence matrix possesses this property and thus can be modeled by this proposed approach.

However, for those data which is do not have the property mentioned above, one can simply use k-means directly on V^* where V^* is viewed as a latent representation of the original data points.

3.4 Computational Complexity Analysis

In this subsection, we discuss the computational complexity of the proposed algorithm in comparison to standard NMF. Besides expressing the complexity of the algorithm using big O notation, we also count the number of arithmetic operations to provide more details about running time. We show the result in Table 1.

Based on the updating rules summarized in Algorithm 1, it is not hard to count the arithmetic operations of inner loop[2] in MultiNMF for each single view, which is quite similar to the multiplicative updating rule for single-view NMF as in Eq. 2.2. Suppose the multiplicative updates stops after t_{in} iterations, the time cost of multiplicative updates then becomes $O(n_v t_{in} MNK)$. Besides the multiplicative updates, MultiNMF also needs $O(n_v NK)$ to compute the consensus matrix according to Eq. 3.11 after the convergence of inner loop. Assume the outer loop[3] which pushes all views toward the consensus stops after t_{out} iterations, the overall cost for MultiNMF is

$$O(t_{out} t_{in} n_v MNK).$$

Therefore, overall the running time of MultiNMF is linear with respect to the number of data points, clusters and views. Note that within each iteration of the outer loop, we actually start multiplicative updates from the convergent points in the previous iteration. This usually saves a lot of time and ensures the descent of the objective function value. It is also worth noting that

[2] line 5-9 in Algorithm 1
[3] line 1-11 in Algorithm 1

the out loop converges very fast, which is demonstrated in the experimental section.

4 Experiment

In this section, experiments were conducted to demonstrate the effectiveness of the proposed MultiNMF in discovering the underlying clustering structure shared by multiple views of data.

4.1 Datasets One synthetic and three real world datasets are used in the experiment. Among the three real world datasets, the first two are text data, and the last one is handwritten digit data. The important statistics of them are summarized in Table 2.

Table 2: Statistics of the four datasets

dataset	size	# view	# cluster
Synthetic	10000	2	4
3-Sources	169	3	6
Reuters	600	3	6
Digit	2000	2	10

- **Synthetic dataset:** It is a toy example consisting of two views for generated data in a two dimensional space. In either view, two of the randomly picked clusters' centers are highly overlapped and therefore difficult to distinguish. This dataset is also designed for the complexity analysis which will be covered in Section 4.6.
- **3-Sources Text dataset**[4]**:** It is collected from three online news sources: BBC, Reuters, and The Guardian. In total there are 948 news articles covering 416 distinct news stories from the period February to April 2009. Of these stories, 169 were reported in all three sources. Each story was manually annotated with one of the six topical labels: business, entertainment, health, politics, sport and technology [13].
- **Reuters Multilingual dataset**[5]**:** This test collection contains feature characteristics of documents originally written in five different languages , and their translations, over a common set of 6 categories [2]. We use documents originally in English as the first view and their French and German translations as the second and third view. We randomly sample 600 documents from this collection in a balanced manner, with each of the 6 clusters having 100 documents.
- **UCI Handwritten Digit dataset**[6]**:** This handwritten digits (0-9) data is from the UCI repository. The dataset consists of 2000 examples, with view-1 being the 76 Fourier coefficients and view-2 being the 240 pixel averages in 2×3 windows.

4.2 Baseline Algorithms To demonstrate how the clustering performance can be improved by the proposed approach, we compared with the following algorithms:

- Single View (**BSV** and **WSV**): Runing each view using the NMF technique. We normalize U and V after convergence according to [20]. Then both the best and the worst single view results are reported, which are referred to as BSV and WSV respectively.
- Feature Concatenation (**ConcatNMF**): Concatenating the features of all the views, and then run NMF directly on this concatenated view representation. The normalization strategy is the same as that of the single-view NMF method.
- Collective NMF (**ColNMF**): Using the shared coefficient matrix but different basis matrices across views as shown below[19, 1]:
$$\sum_{v=1}^{n_v} \lambda_v \|X^{(v)} - U^{(v)}(V^{(*)})^T\|_F^2.$$
It is easy to verify that ColNMF is equivalent to ConcatNMF when no normalization is involved in the latter algorithm.
- Co-regularized Spectral clustering (**Co-reguSC**): Adopting co-regularization framework to spectral clustering [17]. We used gaussian kernel to build the affinity matrix for each view and set the parameter in this algorithm to be 0.01 as suggested.
- Multi-View NMF (**MultiNMF**): This is the proposed algorithm. In our experiments, we empirically set λ_v to 0.01 for all views and datasets. The parameter study will be later discussed.

For the sake of comparison, multiple views are considered with equivalent importance in the evaluation of all the multi-view algorithms. The clustering result is evaluated by comparing the obtained label of each data point with the label provided by the dataset. Two metrics, the accuracy (AC) and the normalized mutual information (NMI) are used to measure the clustering performance. Please refer to [20] for detailed definitions.

4.3 Results Table 3 shows the clustering performance of different algorithms on all the four datasets. In order to randomize the experiments, 20 test runs with different random initializations were conducted and the average performance as well as the standard deviation are reported. As we can see, MultiNMF outperforms the second best algorithm in terms of accuracy/normalized mutual information as 16.6%/12.8% on

[4] http://mlg.ucd.ie/datasets

[5] http://multilingreuters.iit.nrc.ca

[6] http://archive.ics.uci.edu/ml/datasets.html

Table 3: Clustering performance on four real datasets (%)

Algorithm	Accuracy(%)				Normalized Mutual Information(%)			
	Synthetic	3-Sources	Reuters	Digit	Synthetic	3-Sources	Reuters	Digit
BSV	66.0±.09	60.8±.01	46.8±.02	68.5±.05	56.2±.10	53.0±.01	38.8±.02	63.4±.03
WSV	51.7±.11	49.1±.03	46.4±.00	63.4±.04	54.3±.05	44.1±.02	34.2±.00	60.3±.03
ConcatNMF	68.4±.14	58.6±.03	47.3±.00	67.8±.06	60.9±.14	51.7±.03	34.1±.00	62.4±.04
ColNMF	61.8±.08	61.3±.02	51.2±.00	66.0±.05	47.3±.07	55.2±.02	34.6±.00	62.1±.03
Co-reguSC	75.4±.00	47.8±.01	50.6±.02	86.6±.00	71.2±.00	41.4±.01	35.7±.01	77.0±.00
MultiNMF	**92.0±.10**	**68.4±.06**	**53.5±.00**	**88.1±.01**	**84.0±.15**	**60.2±.06**	**40.9±.00**	**80.4±.01**

* Result of MultiNMF is obtained when $\lambda_v = 0.01$. For other values, it still outperforms others in most cases.

synthetic dataset, 7.6%/5.0% on 3-Source, 2.3%/2.1% on Reuters and 1.5%/3.4% on Digit.

On the synthetic dataset, surprisingly, MultiNMF outperforms the baseline methods with a large margin about 16%/12.8%. One of the possible reasons is that the two views are generated independently with complementary information, which satisfies the multi-view assumption well. The difference is significant especially between NMF-based algorithms and MultiNMF which is as large as 23%/23%. The proposed MultiNMF can find the true clustering effectively and efficiently because it restricts matrix factorization towards the right direction instead of simply concatenating all the features together, which loses information of "view" itself.

On the two relational datasets, i.e., 3-Sources and Reuters, MultiNMF still achieves much improvement in AC and NMI compared with the baselines. The difference is especially noticeable on 3-sources, where MultiNMF has 68.4%/60.2% when the best view's AC/NMI is 60.8%/53.0% and the highest AC/NMI obtained by baseline methods is 61.3%/55.2%. Therefore, it supports our claim that the proposed MultiNMF algorithm is quite useful in clustering relational data and its power is amplified by its connection to PLSA.

On the handwritten digit dataset, both Co-reguSC and MultiNMF are demonstrating encouraging clustering results. MultiNMF performs slightly better than Co-reguSC and achieves about 20%/17% performance gain over the other NMF-based algorithms. On this dataset, it is essential to utilize the constraint to regularize clustering solutions obtained from multiple views towards a consensus solution. Therefore, NMF-based algorithms that fail to utilize the complementary perspectives of multiple views cannot find the underlying clustering solution. In contrast, by adopting co-regularization framework to learn the nonnegative factors across different views, MultiNMF succeeds in capturing such knowledge.

4.4 Parameter Study There are n_v parameters in our MultiNMF algorithm: the regularization parameters λ_v for each view. The relative value of λ_v among multiple views reflects each view's importance. If we have some prior knowledge that some views are noisy, then it is better to set relatively small λ_v for such views. In [12], the authors suggest to set the relative weight according to the disagreement between each single view and the consensus. Meanwhile, the absolute value of λ_v reflects how much we want to enforce the regularization constraint. A large λ_v focuses on reaching consensus across views, while a small λ_v cannot tolerate matrix factorization error. In the extreme case, when λ_vs are all 0, the problem reduces to the same as doing NMF with normalization for each view seperately; when λ_vs go to infinity, $V^{(v)}Q^{(v)}$ for different views share the same value. Note that it is still different from directly fixing the shared factor as ColNMF.

Figure 1 shows how the accuracy of MultiNMF on four datasets varies with changes in parameters λ_v, respectively. Figures on NMI measure are omitted due to space limit. Considering the convenience of comparison, we set λ_v to be the same for all the views. As we can see, MultiNMF performs relatively stable when λ_v is around 0.01, which is the value we set to get the experimental results in this section. Moreover, most of the time, MultiNMF still outperforms the baseline methods when λ_v takes various values. It is not strange that all the datasets share the same preference for the value of λ_v simply because $||V^*||_1 \approx ||X||_1 = 1$, implying that the difference in scales of two terms in Eq. 3.6 should not vary significantly for different datasets.

4.5 Convergence Study The updating rules for minimizing the objective function of MultiNMF in Eq. 3.6 are essentially iterative and it can be proved that these rules are convergent. Figure 2 shows the convergence curve together with its performance. The black solid line shows the value of the objective function and the red dashed line indicates the accuracy of the method. As can be seen, only after around 15 iterations, the algorithm will converge.

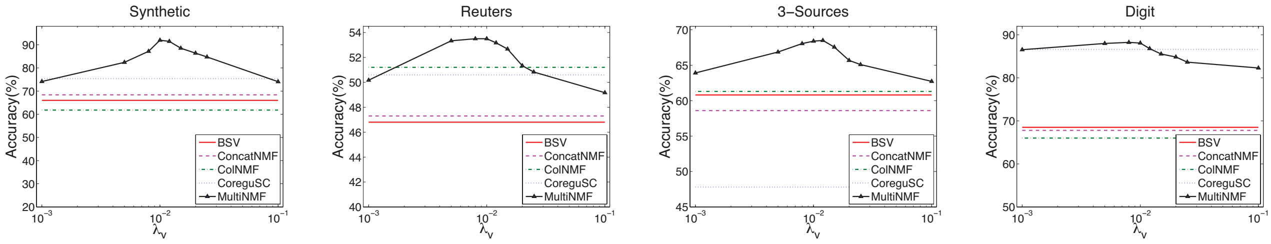

Figure 1: Performance of MultiNMF w.r.t. parameters λ_v.

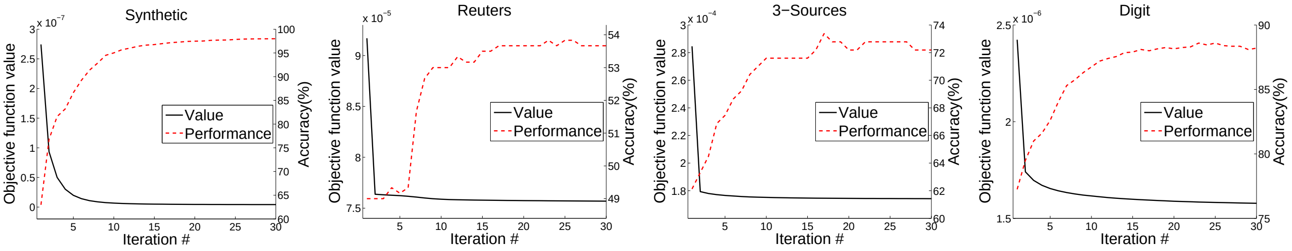

Figure 2: Convergence and corresponding performance curve.

4.6 Computational Complexity Study As discussed in subsection 4.6, MultiNMF has linear time complexity in the number of data points, clusters, and views. In this part, we verify this claim on the synthetic data mentioned earlier in this section.

We conduct the experiments on a desktop with Intel Core I7 2600 and 16GB memory. We change the number of data points, clusters, and views respectively. The default setting is 10000 data points, 4 clusters, and 2 views. During the experiment, we fix two aspects and change the remaining one. Figure 3 shows the running time of MultiNMF and CoreguSC in terms of varying data points. Additional figures regarding the varying number of clusters and views are put in the supplementary material. Clearly, MultiNMF is linear in execution time and can scale well to large data sets. Although CoreguSC is proposed based on the similar co-regularization framework, it requires eigendecompositions of non-sparse matrices which costs too much time and space[7].

5 Related work

We have discussed three categories of multi-view clustering algorithms in the introduction, and here we particularly discuss several algorithms proposed for spectral clustering [22, 16, 17]. In [22], the normalized cut approach is generalized to multiple views via a random walk formulation. [16] approaches the problem by adopting an co-training framework [5] such that the similarity matrix in one view is affected by the similarity estimated based on the eigenvectors of Laplacian matrix in the other view. In [17], a co-regularization framework is proposed to enforce the *pairwise similarities* computed from the eigenvectors learnt from different views to be close. However, due to non-sparseness of the intermediate matrices for eigendecomposition, the computational complexity becomes an issue, which can be observed in Figure 3.

A few clustering algorithms have been proposed to apply NMF on multi-relation (heterogeneous) or multi-view data. Collective NMF [19] is designed for relational learning based on multi-relations where each entity type is represented by one factor. In this setting, data is in a single view but contains multiple relations, which is different from what we are trying to solve. In [1], this idea is extended to multi-view clustering scenario by enforcing a shared coefficient matrix among different views. This is equivalent to first concatenating features of different views together and then applying NMF to factorize. However, whether this approach is optimized for clustering is questionable. First, this hard assumption seems too strong and many times we prefer relatively soft constraints. Second, with proper normalization, previous work [20] has shown to achieve better performance in terms of clustering. For these two purposes, the proposed MultiNMF algorithm maintains meaningful clustering results obtained from each view but only biases their findings towards a common clustering solution via regularization constraints. [13] assumes that clustering solutions have been obtained from multiple views separately, and employs matrix factorization techniques over the clustering results, and thus it is a late integration strategy. Therefore, it is quite different from the proposed MultiNMF approach as we aim at conducting clustering instantaneously on the raw data of multiple views.

[7] It is difficult to estimate CoreguSC when number of data points is over 20,000 due to memory limitation.

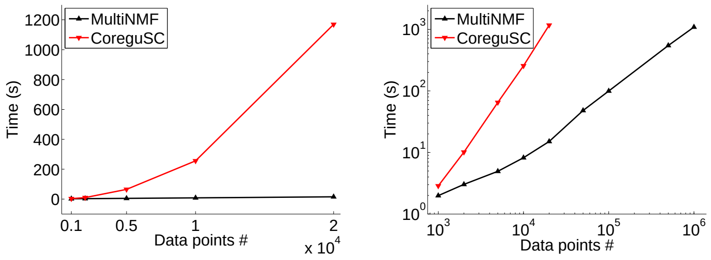

Figure 3: Running time of MultiNMF v.s. CoreguSC on synthetic dataset.

6 Conclusions

In this paper, we introduced a novel algorithm for multi-view clustering based on nonnegative matrix factorization. In order to efficiently learn the underlying clustering structure embedded in multiple views, we require coefficient matrices learnt from factorizations of different views to be regularized towards a common consensus. To achieve this, we develop a joint matrix factorization algorithm to incorporate not only individual matrix factorizations but also inconsistency between each view's coefficient matrix and the consensus. Moreover, we design a novel and effective normalization procedure to keep different factors comparable and meaningful in terms of clustering. We also show that the proposed method converges with linear time. Experiments on both synthetic and three real world datasets demonstrate its effectiveness.

7 Acknowledgement

This work was supported in part by the U.S. National Science Foundation grants IIS0905215, U.S. Army Research Laboratory under Cooperative Agreement No. W911NF0920053 (NS-CTA). Chi Wang was supported by Microsoft Graduate Fellowship.

References

[1] Z. Akata, C. Thurau, and C. Bauckhage. Non-negative Matrix Factorization in Multimodality Data for Segmentation and Label Prediction. In *16th Computer Vision Winter Workshop*, pages 1–8, 2011.

[2] M. Amini, N. Usunier, and C. Goutte. Learning from multiple partially observed views - an application to multilingual text categorization. In *NIPS*, pages 28–36, 2009.

[3] S. Bickel and T. Scheffer. Multi-view clustering. In *ICDM*, pages 19–26, 2004.

[4] M. Blaschko and C. Lampert. Correlational spectral clustering. In *CVPR*, pages 1 –8, 2008.

[5] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *COLT*, pages 92–100, 1998.

[6] J. Brunet, P. Tamayo, T. Golub, and J. Mesirov. Metagenes and molecular pattern discovery using matrix factorization. *PNAS*, 101(12):4164–4169, 2004.

[7] E. Bruno and S. Marchand-Maillet. Multiview clustering: a late fusion approach using latent models. In *SIGIR*, pages 736–737, 2009.

[8] K. Chaudhuri, S. Kakade, K. Livescu, and K. Sridharan. Multi-view clustering via canonical correlation analysis. In *ICML*, pages 129–136, 2009.

[9] C. Ding, T. Li, and W. Peng. On the equivalence between non-negative matrix factorization and probabilistic latent semantic indexing. *Computational Statistics Data Analysis*, 52:3913 – 3927, 2008.

[10] C. Ding, T. Li, W. Peng, and H. Park. Orthogonal nonnegative matrix t-factorizations for clustering. In *KDD*, pages 126–135, 2006.

[11] E. Gaussier and C. Goutte. Relation between PLSA and NMF and implications. In *SIGIR*, pages 601–602, 2005.

[12] G. Greco, A. Guzzo, and L. Pontieri. Coclustering multiple heterogeneous domains: Linear combinations and agreements. *TKDE*, 22:1649–1663, 2010.

[13] D. Greene and P. Cunningham. A matrix factorization approach for integrating multiple data views. In *ECML PKDD*, pages 423–438, 2009.

[14] Q. Gu, C. Ding, and J. Han. On trivial solution and scale transfer problems in graph regularized nmf. In *AAAI*, pages 1288–1293, 2011.

[15] T. Hofmann. Probabilistic latent semantic indexing. In *SIGIR*, pages 50–57, 1999.

[16] A. Kumar and H. Daumé III. A co-training approach for multi-view spectral clustering. In *ICML*, pages 393–400, 2011.

[17] A. Kumar, P. Rai, and H. Daumé III. Co-regularized multi-view spectral clustering. In *NIPS*, pages 1413–1421. 2011.

[18] D. Lee and S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

[19] A. Singh and G. Gordon. Relational learning via collective matrix factorization. In *KDD*, pages 650–658, 2008.

[20] W. Xu, X. Liu, and Y. Gong. Document clustering based on non-negative matrix factorization. *SIGIR*, pages 267–273, 2003.

[21] J. Yang, S. Yang, Y. Fu, X. Li, and T. Huang. Nonnegative graph embedding. In *CVPR*, pages 1 –8, june 2008.

[22] D. Zhou and C. Burges. Spectral clustering and transductive learning with multiple views. *ICML*, pages 1159–1166, 2007.