Non-negative Matrix Factorization: a Comprehensive Review

Yu-Xiong Wang, *Student Member, IEEE*, Yu-Jin Zhang, *Senior Member, IEEE*

Abstract—Non-negative matrix factorization (NMF), a relatively novel paradigm for dimensionality reduction, has been in the ascendant since its inception. It incorporates the non-negativity constraint and thus obtains the parts-based representation as well as enhancing the interpretability of the issue correspondingly. This survey paper mainly focuses on the theoretical research into NMF over the last five years, where the principles, basic models, properties, and algorithms of NMF along with its various modifications, extensions, and generalizations are summarized systematically. The existing NMF algorithms are divided into four categories: Basic NMF, Constrained NMF, Structured NMF, and Generalized NMF, upon which the design principles, characteristics, problems, relationships, and evolution of these algorithms are presented and analyzed comprehensively. Some related work not on NMF that NMF should learn from or has connections with is involved too. Moreover, some open issues remained to be solved are discussed. Several relevant application areas of NMF are also briefly described. This survey aims to construct an integrated, state-of-the-art framework for NMF concept, from which the follow-up research may benefit.

Index Terms—data mining, dimensionality reduction, multivariate data analysis, non-negative matrix factorization (NMF).

1 Introduction

One of the basic concepts deeply rooted in science and engineering is that there must be something simple, compact, and elegant playing the fundamental roles under the apparent chaos and complexity. This is also the case in signal processing, data analysis, data mining, pattern recognition, and machine learning. With the increasing quantities of available raw data due to the development in sensor and computer technology, how to obtain such an effective way of representation by appropriate dimensionality reduction technique has become important, necessary, and challenging in multivariate data analysis. Generally speaking, two basic properties are supposed to be satisfied: firstly, the dimension of the original data should be reduced; secondly, the principal components, hidden concepts, prominent features, or latent variables of the data, depending on the application context, should be identified efficaciously.

In many cases, the primitive data sets or observations are organized as data matrices (or tensors), and described by linear (or multilinear) combination models; whereupon the formulation of dimensionality reduction can be regarded as, from the algebraic perspective, decomposing the original data matrix into two factor matrices. The canonical methods, such as principal component analysis (PCA), linear discriminant analysis (LDA), independent component analysis (ICA), and vector quantization (VQ) et al., are the exemplars of such low-rank approximations. They differ from one another in the statistical properties attributable to the different constraints imposed on the component matrices and their underlying structures; however, they have something in common that there is no constraint in the sign of the elements in the factorized matrices. In other words, the negative component or the subtractive combination is allowed in the representation. By contrast, a new paradigm of factorization—Non-negative Matrix Factorization (NMF), which incorporates the non-negativity constraint and thus obtains the parts-based representation as well as enhancing the interpretability of the issue correspondingly, was initiated by Paatero and Tapper in 1994 and 1997 [1], [2] together with Lee and Seung in 1999 and 2001 [3], [4].

As a matter of fact, the notion of NMF has a long history under the name "self modeling curve resolution" in chemometrics, where the vectors are continuous curves rather than discrete vectors [5]. NMF was first introduced by Paatero and Tapper as the concept of Positive Matrix Factorization, which concentrated on a specific application with Byzantine algorithms. These shortcomings limit both the theoretical analysis, such as the convergence of the algorithms or the properties of the solutions, and the generalization of the algorithms in other applications. Fortunately, NMF was popularized by Lee and Seung due to their contributing work of a simple yet effective algorithmic procedure, and more importantly the emphasis on its potential value of parts-based representation.

Far beyond a mathematical exploration, the philosophy underlying NMF, which tries to formulate a feasible model for learning object parts, is closely relevant to perception mechanism. While the parts-

• *Y.-X. Wang and Y.-J. Zhang are with the Tsinghua National Laboratory for Information Science and Technology & Department of Electronic Engineering, Tsinghua University, Beijing 100084, China. E-mail: albertwyx@gmail.com, zhang-yj@tsinghua.edu.cn*

Manuscript received 21 July 2011.

Digital Object Indentifier 10.1109/TKDE.2012.51

based representation seems intuitive, it is indeed on the basis of physiological and psychological evidence: perception of the whole is based on perception of its parts [6], one of the core concepts in certain computational theories of recognition problems. In fact there are two complementary connotations in non-negativity—non-negative component and purely additive combination. On the one hand, the negative values of both observations and latent components are physically meaningless in many kinds of real-world data, such as image, spectra, and gene data, analysis tasks. Meanwhile, the discovered prototypes commonly correspond with certain semantic interpretation. For instance, in face recognition, the learnt basis images are localized rather than holistic, resembling parts of faces, such as eyes, nose, mouth, and cheeks [3]. On the other hand, objects of interest are most naturally characterized by the inventory of its parts, and the exclusively additive combination means that they can be reassembled by adding required parts together similar to identikits. NMF thereupon has achieved great success in real-word scenarios and tasks. In document clustering, NMF surpasses the classic methods, such as spectral clustering, not only in accuracy improvement but also in latent semantic topic identification [7].

To boot, the non-negativity constraint will lead to sort of sparseness naturally [3], which is proved to be a highly effective representation distinguished from both the completely distributed and the solely active component description [8]. When NMF is interpreted as a neural network learning algorithm depicting how the visible variables are generated from the hidden ones, the parts-based representation is obtained from the additive model. A positive number indicates the presence and a zero value represents the absence of some event or component. This conforms nicely to the dualistic properties of neural activity and synaptic strengths in neurophysiology: either excitatory or inhibitory without changing sign [3].

Because of the enhanced semantic interpretability under the non-negativity and the ensuing sparsity, NMF has become an imperative tool in multivariate data analysis, and been widely used in the fields of mathematics, optimization, neural computing, pattern recognition and machine learning [9], data mining [10], signal processing [11], image engineering and computer vision [11], spectral data analysis [12], bioinformatics [13], chemometrics [1], geophysics [14], finance and economics [15]. More specifically, such applications include text data mining [16], digital watermark, image denoising [17], image restoration, image segmentation [18], image fusion, image classification [19], image retrieval, face hallucination, face recognition [20], facial expression recognition [21], audio pattern separation [22], music genre classification [23], speech recognition, microarray analysis, blind source separation [24], spectroscopy [25], gene expression classification [26], cell analysis, EEG signal processing [17], pathologic diagnosis, email surveillance [10], online discussion participation prediction, network security, automatic personalized summarization, identification of compounds in atmosphere analysis [14], earthquake prediction, stock market pricing [15], and so on.

There have been numerous results devoted to NMF research since its inception. Researchers from various fields, mathematicians, statisticians, computer scientists, biologists, and neuroscientists, have explored the NMF concept from diverse perspectives. So a systematic survey is of necessity and consequence. Although there have been such survey papers as [27], [28], [12], [13], [10], [11], [29] and one book [9], they fail to reflect either the updated or the comprehensive results. This review paper will summarize the principles, basic models, properties, and algorithms of NMF systematically over the last five years, including its various modifications, extensions, and generalizations. A taxonomy is accordingly proposed to logically group them, which have not been presented before. Besides these, some related work not on NMF that NMF should learn from or has connections with will also be involved. Furthermore, this survey mainly focuses on the theoretical research rather than the specific applications, the practical usage will also be concerned though. It aims to construct an integrated, state-of-the-art framework for NMF concept, from which the follow-up research may benefit.

In conclusion, the theory of NMF has advanced significantly by now yet is still a work in progress. To be specific: (1) The properties of NMF itself have been explored more deeply; whereas a firm statistical underpinning like those of the traditional factorization methods—PCA or LDA—is not developed fully (partly due to its knottiness). (2) Some problems like the ones mentioned in [29] have been solved, especially those with additional constraints; nevertheless a lot of other questions are still left open.

The existing NMF algorithms are divided into four categories here given in Fig. 1, following some unified criteria: Basic NMF (BNMF), which only imposes the non-negativity constraint; Constrained NMF (CNMF), which imposes some additional constraints as regularization; Structured NMF (SNMF), which modifies the standard factorization formulations; Generalized NMF (GNMF), which breaks through the conventional data types or factorization modes in a broad sense. The model level from Basic to Generalized NMF becomes broader. Therein Basic NMF formulates the fundamental analytical framework upon which all other NMF models are built. We will present the optimization tools and computational methods to efficiently and robustly solve Basic NMF. Moreover, the pragmatic issue of NMF with respect to large-scale data sets and online processing will also be discussed.

Constrained NMF is categorized into four subclass-

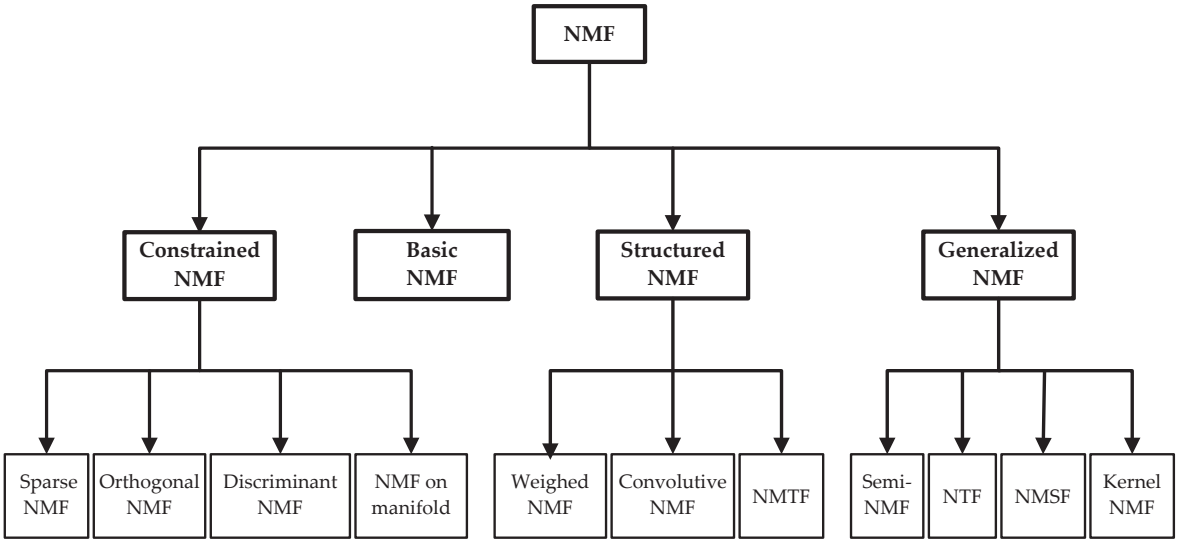

Fig. 1: The categorization of NMF models and algorithms.

es: (1) Sparse NMF (SPNMF), which imposes the sparseness constraint; (2) Orthogonal NMF (ONMF), which imposes the orthogonality constraint; (3) Discriminant NMF (DNMF), which involves the information for classification and discrimination; (4) NMF on manifold (MNMF), which preserves the local topological properties. We will demonstrate why these morphological constraints are essentially necessary and how to incorporate them into the existing solution framework of Basic NMF.

Correspondingly, Structured NMF is categorized into three subclasses: (1) Weighed NMF (WNMF), which attaches different weights to different elements regarding their relative importance; (2) Convolutive NMF (CVNMF), which considers the time-frequency domain factorization; (3) Non-negative Matrix Trifactorization (NMTF), which decomposes the data matrix into three factor matrices.

Besides, Generalized NMF is categorized into four subclasses: (1) Semi-NMF, which relaxes the non-negativity constraint only on the specific factor matrix; (2) Non-negative Tensor Factorization (NTF), which generalizes the matrix-form data to higher dimensional tensors; (3) Non-negative Matrix-set Factorization (NMSF), which extends the data sets from matrices to matrix-sets; (4) Kernel NMF (KNMF), which is the nonlinear model of NMF.

The remainder of this paper is organized as follows. Firstly, the mathematic formulation of NMF model is presented, and the unearthed properties of NMF are summarized. Then the algorithmic details of foregoing categories of NMF are elaborated. Finally, conclusions are drawn, and some open issues remained to be solved are discussed.

2 Concept and Properties of NMF

Definition: Given an M dimensional random vector $\boldsymbol{x}$ with non-negative elements, whose N observations are denoted as $\boldsymbol{x}_{j,j=1,2,\ldots,N}$, let data matrix be $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N] \in \mathbb{R}_{\geq 0}^{M \times N}$, NMF seeks to decompose $\boldsymbol{X}$ into non-negative $M \times L$ basis matrix $\boldsymbol{U}=[\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_L] \in \mathbb{R}_{\geq 0}^{M \times L}$ and non-negative $L \times N$ coefficient matrix $\boldsymbol{V}=[\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_N] \in \mathbb{R}_{\geq 0}^{L \times N}$, such that $\boldsymbol{X} \approx \boldsymbol{UV}$, where $\mathbb{R}_{\geq 0}^{M \times N}$ stands for the set of $M \times N$ element-wise non-negative matrices. This can also be written as the equivalent vector formula $\boldsymbol{x}_j \approx \sum_{i=1}^{L} \boldsymbol{u}_i \boldsymbol{V}_{ij}$.

It is obvious that $\boldsymbol{v}_j$ is the weight coefficient of the observation $\boldsymbol{x}_j$ on the columns of $\boldsymbol{U}$, the basis vectors or the latent feature vectors of $\boldsymbol{X}$. Hence NMF decomposes each data into the linear combination of the basis vectors. Because of the initial condition $L << \min(M, N)$, the obtained basis vectors are incomplete over the original vector space. In other words, this approach tries to represent the high dimensional stochastic pattern with far fewer bases, so the perfect approximation can be achieved successfully only if the intrinsic features are identified in $\boldsymbol{U}$.

Here we discuss the relationship between L and M, N a little more. In most cases, NMF is viewed as a dimensionality reduction and feature extraction technique with $L << M, L << N$; that is, the basis set learnt from NMF model is incomplete, and the energy is compacted. However, in general, L can be smaller, equal or larger than M. But there are fundamental differences in the decomposition for $L < M$ and $L > M$. It is a sort of sparse coding and compressed sensing with over-complete basis when $L > M$. Hence L need not be limited by the dimensionality of the data, which is useful for some applications, like classification. In this situation, it may benefit from the sparseness due to both non-negativity and redundant representation. One approach to obtain this NMF model is to perform the decomposition on the residue

matrix $\boldsymbol{E}=\boldsymbol{X}-\boldsymbol{UV}$ repeatedly and sequentially [30].

As a kind of matrix factorization model, three essential questions need answering: (1) existence, whether the nontrivial NMF solutions exist; (2) uniqueness, under what assumptions NMF is, at least in some sense, unique; (3) effectiveness, under what assumptions NMF is able to recover the "right answer". The existence was showed via the theory of Completely Positive (CP) Factorization for the first time in [31]. The last two concerns were first mentioned and discussed from a geometric viewpoint in [32].

Complete NMF $\boldsymbol{X}=\boldsymbol{UV}$ is considered firstly for the analysis of existence, convexity, and computational complexity. The trivial solution always exists as $\boldsymbol{U}=\boldsymbol{X}$ and $\boldsymbol{V}=\boldsymbol{I}_N$. By relating NMF to CP Factorization, Vasiloglou et al. showed that every non-negative matrix has a nontrivial complete NMF [31]. As such, CP Factorization is a special case, where a non-negative matrix $\boldsymbol{X}\in\mathbb{R}_{\geq 0}^{M\times M}$ is CP if it can be factored in the form $\boldsymbol{X}=\boldsymbol{UU}^T, \boldsymbol{U}\in\mathbb{R}_{\geq 0}^{M\times L}$. The minimum L is called the CP-rank of $\boldsymbol{X}$. When combining that the set of CP matrices forms a convex cone with that the solution to NMF belongs to a CP cone, solving NMF is a convex optimization problem [31]. Nevertheless, finding a practical description of the CP cone is still open, and it remains hard to formulate NMF as a convex optimization problem, despite a convex relaxation to rank reduction with theoretical merit proposed in [31].

Using the bilinear model, complete NMF can be rewritten as linear combination of rank-one non-negative matrices expressed by

$$\boldsymbol{X}=\sum_{i=1}^{L}\boldsymbol{U}_{\bullet i}\boldsymbol{V}_{i\bullet}=\sum_{i=1}^{L}\boldsymbol{U}_{\bullet i}\circ(\boldsymbol{V}_{i\bullet})^T \tag{1}$$

where $\boldsymbol{U}_{\bullet i}$ is the i-th column vector of $\boldsymbol{U}$ while $\boldsymbol{V}_{i\bullet}$ is the i-th row vector of $\boldsymbol{V}$, and $\circ$ denotes the outer product of two vectors. The smallest L making the decomposition possible is called the non-negative rank of the non-negative matrix $\boldsymbol{X}$, denoted as $rank_+(\boldsymbol{X})$. And it satisfies the following trivial bounds [33]

$$rank(\boldsymbol{X})\leq rank_+(\boldsymbol{X})\leq\min(M,N). \tag{2}$$

While PCA can be solved in polynomial time, the optimization problem of NMF, with respect to determining the non-negative rank and computing the associated factorization, is more difficult than its unconstrained counterpart. It is in fact NP-hard when requiring both the dimension and the factorization rank of $\boldsymbol{X}$ to increase, which was proved via relating it to NP-hard intermediate simplex problem by Vavasis [34]. This is also the corollary of CP programming, since the CP cone cannot be described in polynomial time despite its convexity. In the special case when $rank(\boldsymbol{X})=1$, complete NMF can be solved in polynomial time. However, the complexity of complete NMF for fixed factorization rank generally is still unknown [35].

Another related work is so-called Non-negative Rank Factorization (NRF) focusing on the situation of $rank(\boldsymbol{X})=rank_+(\boldsymbol{X})$, i.e. selecting $rank(\boldsymbol{X})$ as the minimum L [33]. This is not always possible, and only non-negative matrix with a corresponding simplicial cone (A polyhedral cone is simplicial if its vertex rays are linearly independent.) existed has an NRF [36].

In most cases, the approximation version of NMF $\boldsymbol{X}\approx\boldsymbol{UV}$ instead of the complete factorization is widely utilized. An alternative generative model is:

$$\boldsymbol{X}=\boldsymbol{UV}+\boldsymbol{E} \tag{3}$$

where $\boldsymbol{E}\in\mathbb{R}^{M\times N}$ is the residue or noise matrix representing the approximation error.

These two modes of NMF are essentially coupled with each other, though much more attention is devoted to the latter. The theoretical results on complete NMF will be helpful to design more efficient NMF algorithms [31], [34]. The selection of the factorization rank L of NMF may be more creditable if tighter bound for the non-negative rank is obtained [37].

In essence, NMF is an ill-posed problem with non-unique solutions [32], [38]. From the geometric perspective, NMF can be viewed as finding a simplicial cone involving all the data points in the positive orthant. Given a simplicial cone satisfying all these conditions, it is not difficult to construct another cone containing the former one to meet the same conditions, so the nesting can work on infinitively thus leading to an ill-defined factorization notion. From the algebraic perspective, if there exists a solution $\boldsymbol{X}\approx\boldsymbol{U}_0\boldsymbol{V}_0$, let $\boldsymbol{U}=\boldsymbol{U}_0\boldsymbol{D}$, $\boldsymbol{V}=\boldsymbol{D}^{-1}\boldsymbol{V}_0$, then $\boldsymbol{X}\approx\boldsymbol{UV}$. If a nonsingular matrix and its inverse are both non-negative, then the matrix is a generalized permutation with the form of $\boldsymbol{PS}$, where $\boldsymbol{P}$ and $\boldsymbol{S}$ are permutation and scaling matrices, respectively. So the permutation and scaling ambiguities for NMF are inevitable. For that matter, NMF is called unique factorization up to a permutation and a scaling transformation when $\boldsymbol{D}=\boldsymbol{PS}$. Unfortunately, there are many ways to select a rotational matrix $\boldsymbol{D}$ which is not necessarily a generalized permutation or even non-negative matrix, so that the transformed factor matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ are still non-negative. In other words, the sole non-negativity constraint in itself will not suffice to guarantee the uniqueness, let alone the effectiveness. Nevertheless, the uniqueness will be achieved if the original data satisfy certain generative model. Intuitively, if $\boldsymbol{U}_0$ and $\boldsymbol{V}_0$ are sufficiently sparse, only generalized permutation matrices are possible rotation matrices satisfying the non-negativity constraint. Strictly speaking, this is called boundary close condition for sufficiency and necessity of the uniqueness of NMF solution [39]. The deep discussions about this issue can be found in [32], [38], [39], [40], [41], [42]. In practice, incorporating additional constraints such as sparseness in the factor matrices or normalizing the columns of $\boldsymbol{U}$ (respectively rows of $\boldsymbol{V}$) to unit length is helpful in alleviating

the rotational indeterminacy [9].

It was hoped that NMF would produce an intrinsically parts-based and sparse representation in unsupervised mode [3], which is the most inspiring benefit of NMF. Intuitively, this can be explained by that the stationary points of NMF solutions will typically be located at the boundary of the feasible domain due to the first order optimality conditions, leading to zero elements [37]. Further experiments by Li et al. have shown, however, that the pure additivity does not necessarily mean sparsity and that NMF will not necessarily learn the localized features [43].

Further more, NMF is equivalent to *k*-means clustering when using Square of Euclidian Distance (SED) [44], [45], while tantamount to probabilistic latent semantic analysis (PLSA) when using Generalized Kullback-Leibler Divergence (GKLD) as the objective function [46], [47].

So far we may conclude that the merits of NMF, parts-based representation and sparseness included, come at the price of more complexity. Besides, SVD or PCA has always a more compact spectrum than NMF [31]. You just can't have the best of both worlds.

3 Basic NMF Algorithms

The cynosure in Basic NMF is trying to find more efficient and effective solutions to NMF problem under the sole non-negativity constraint, which lays the foundation for the practicability of NMF. Due to its NP-hardness and lack of appropriate convex formulations, the non-convex formulations with relatively easy solvability are generally adopted, and only local minima are achievable in a reasonable computational time. Hence, the classic and also more practical approach is to perform alternating minimization of a suitable cost function as the similarity measures between $\boldsymbol{X}$ and the product $\boldsymbol{UV}$. The different optimization models vary from one another mainly in the object functions and the optimization procedures.

These optimization models, even serving to give sight of some possible directions for the solutions to Constrained, Structured, and Generalized NMF, are the kernel discussions of this section. We will first summarize the objective functions. Then the details about the classic Basic NMF framework and the paragon algorithms are presented. Moreover, some new vision of NMF, such as the geometric formulation of NMF, and the pragmatic issue of NMF, such as large-scale data sets, online processing, parallel computing, and incremental NMF, will be discussed. In the last part of this section, some other relevant issues are also involved.

3.1 Similarity Measures or Objective Functions

In order to quantify the difference between the original data $\boldsymbol{X}$ and the approximation $\boldsymbol{UV}$, a similarity measure $D(\boldsymbol{X}\|\boldsymbol{UV})$ needs to be defined firstly. This is also the objective function of the optimization model. These similarity measures can be either distances or divergences, and corresponding objective functions can be either a sole cost function or optionally a set of cost functions with the same global minima to be minimized sequentially or simultaneously.

The most commonly used objective functions are SED (i.e. Frobenius norm) (4) and GKLD (i.e. I-divergence) (5) [4]

$$D_F(\boldsymbol{X}\|\boldsymbol{UV}) = \frac{1}{2}\|\boldsymbol{X}-\boldsymbol{UV}\|_F^2 = \frac{1}{2}\sum_{ij}\left(\boldsymbol{X}_{ij}-[\boldsymbol{UV}]_{ij}\right)^2 \quad (4)$$

$$D_{KL}(\boldsymbol{X}\|\boldsymbol{UV}) = \sum_{ij}\left(\boldsymbol{X}_{ij}\ln\frac{\boldsymbol{X}_{ij}}{[\boldsymbol{UV}]_{ij}} - \boldsymbol{X}_{ij} + [\boldsymbol{UV}]_{ij}\right). \quad (5)$$

There are some drawbacks of GKLD, especially the gradients needed in optimization heavily depend on the scales of factorizing matrices leading to many iterations. Thus the original KLD is renewed for NMF by normalizing the input data in [48]. Other cost functions consist of Minkowski family of metrics known as ℓ_p-norm, Earth Mover's distance metric [18], α-divergence [17], β-divergence [49], γ-divergence [50], Csiszár's φ-divergence [51], Bregman divergence [52], and α-β-divergence [53]. Most of them are element-wise measures. Some similarity measures are more robust with respect to noise and outliers, such as hypersurface cost function [54], γ-divergence [50], and α-β-divergence [53].

Statistically, different similarity measures can be determined based on a prior knowledge about the probability distribution of the noise, which actually reflects the statistical structure of the signals and the disclosed components. For example, the SED minimization can be seen as a maximum likelihood estimator where the difference is due to additive Gaussian noise, whereas GKLD can be shown to be equivalent to the Expectation Maximization (EM) algorithm and maximum likelihood for Poisson processes [9].

Given that while the optimization problem is not jointly convex in both $\boldsymbol{U}$ and $\boldsymbol{V}$, it is separately convex in either $\boldsymbol{U}$ or $\boldsymbol{V}$, the alternating minimizations are seemly the feasible direction. A phenomenon worthy of notice is that although the generative model of NMF is linear, the inference computation is nonlinear.

3.2 Classic Basic NMF Optimization Framework

The prototypical multiplicative update rules originated by Lee and Seung—the SED-MU and GKLD-MU [4] have still been widely used as the baseline. The SED-MU and GKLD-MU algorithms use SED and GKLD as objective functions, respectively, and both apply iterative multiplicative updates as the optimization approach similar to EM algorithms. In essence they can be viewed as adaptive rescaled gradient descent algorithms. Considering the efficiency, they are relatively simple and parameter free with low

cost per iteration, but they converge slowly due to a first-order convergence rate [28], [55]. Regarding the quality of the solutions, Lee and Seung claimed that the multiplicative update rules converge to a local minimum [4]. Gonzales and Zhang indicated that the gradient and properties of continual non-increasing by no means, however, ensure the convergence to a limit point that is also a stationary point, which can be understood under the Karush-Kuhn-Tucker (KKT) optimality conditions [55], [56]. So the accurate conclusion is that the algorithms converge to a stationary point which is not necessarily a local minimum when the limit point is in the interior of the feasible region; its stationarity can not be even determined when the limit point lies on the boundary of the feasible region [10]. However, a minor modification in their step size of the gradient descent formula achieves a first-order stationary point [57]. Another drawback is the strong correlation enforced by the multiplication. Once an element in the factor matrices becomes zero, it must remain zero. This means the gradual shrinkage of the feasible region, which is harmful for getting more superior solution. In practice, to reduce the numerical difficulties, like numerical instabilities or ill-conditioning, the normalization of the ℓ_1 or ℓ_2 norm of the columns in $\boldsymbol{U}$ is often needed as an extra procedure, yet this simple trick has changed the original optimization problem, thereby making searching for the global minimum more complicated. Besides, to preclude the computational difficulty due to division by zero, an extra positive additive value in the denominator is helpful [56].

To accelerate the convergence rate, one popular method is to apply gradient descent algorithms with additive update rules. Other techniques such as conjugate gradient, projected gradient, and more sophisticated second-order scheme like Newton and Quasi-Newton methods et al. are also in consideration. They choose appropriate descent direction, such as the gradient direction, and update the element additively in the factor matrices at a certain learning rate. They differ from one another as for either the descent direction or the learning rate strategy. To satisfy the non-negativity constraint, the updated matrices are brought back to the feasible region, namely the non-negative orthant, by additional projection, like simply setting all negative elements to zero. Usually under certain mild additional conditions, they can guarantee the first-order stationarity. These are the widely developed algorithms in Basic NMF recent years.

Using SED as the objective function, the multiplicative and gradient descent algorithms are special cases of a general framework called "alternating non-negative least squares (ANLS)", which is alternating least squares (ALS) modified under the non-negativity constraint. Considering the separate convexity, the two-variable optimization problem is converted into the non-negative least squares (NLS) optimization subproblems. Given the SED function of two factor matrices, these procedures perform a constrained minimization with respect to one matrix while holding the other matrix fixed; and then the minimization is performed again with the roles of the matrices reversed as follows

$$\min_{\boldsymbol{U}\geq 0} D_F(\boldsymbol{X}\|\boldsymbol{UV}) = \min_{\boldsymbol{U}\geq 0} \frac{1}{2}\|\boldsymbol{X}-\boldsymbol{UV}\|_F^2 \tag{6}$$

$$\min_{\boldsymbol{V}\geq 0} D_F(\boldsymbol{X}\|\boldsymbol{UV}) = \min_{\boldsymbol{V}\geq 0} \frac{1}{2}\|\boldsymbol{X}-\boldsymbol{UV}\|_F^2 . \tag{7}$$

This approach corresponds to the so-called "block coordinate descent" method. The NLS optimization subproblems can be cast as a common form of the existing convex quadratic programming algorithms. The two blocks can be further partitioned into several more basic separable cells upon which it is sometimes possible to perform an exact block coordinate descent, i.e., find a global minimum of subproblems for each block of variables. Of course this general framework is also suitable for the GKLD objective function, as long as the one-variable optimization problems have closed-form solutions or are otherwise tractable.

In brief, the majority of Basic NMF algorithms can be unified as alternating minimizations or block coordinate descent schemes with different block sizes and various optimization approaches for each block. It is mainly because the decoupled subproblems are relatively solvable and can be handled efficiently. For deep understanding one may refer to [9], [37]. There have been no efficient algorithms updating $\boldsymbol{U}$ and $\boldsymbol{V}$ simultaneously so far. Moreover, in most cases, only convergence to stationary points is warranted which satisfies KKT (first-order) optimality conditions as follows

$$\begin{aligned} &\boldsymbol{U}\geq 0, \nabla_{\boldsymbol{U}} D(\boldsymbol{X}\|\boldsymbol{UV})\geq 0, \boldsymbol{U}\otimes\nabla_{\boldsymbol{U}} D(\boldsymbol{X}\|\boldsymbol{UV})=0 \\ &\boldsymbol{V}\geq 0, \nabla_{\boldsymbol{V}} D(\boldsymbol{X}\|\boldsymbol{UV})\geq 0, \boldsymbol{V}\otimes\nabla_{\boldsymbol{V}} D(\boldsymbol{X}\|\boldsymbol{UV})=0 \end{aligned} \tag{8}$$

where $\otimes$ is the Hadamard (component-wise) multiplication. Another simple observation is that $\boldsymbol{U}$ and $\boldsymbol{V}$ are dual variables in the objective functions; so once the update rule for $\boldsymbol{U}$ is obtained, the similar procedure can be performed on $\boldsymbol{V}$.

3.3 Paragon Algorithms

To provide the technological details, some paragon algorithms based on the above the optimization framework will be elaborated here.

Merritt and Zhang [58] proposed an interior-point gradient method using a search direction equivalent to that used in the LS algorithm. The objective function is minimized if the updated value is positive; otherwise a certain proportion of the longest step is chosen as the update step to ensure the positive constraint. This sorting operation guarantees the non-negativity as well as decreasing the objective function as far as possible. Moreover, the convergence of the algorithm was also proved.

Gonzales and Zhang [55] traced the SED-MU model back to the original adaptive additive update formula. To expedite the decreasing speed of the objective function per iteration, they introduced another multiplicative regulatory factor besides the original adaptive learning rate. They also confined the update step into a proportion of the maximum step. This modification of the standard Lee-Seung algorithm is interesting, but the convergence to a local minimum is still not in effect similar to the original algorithm.

Broadly speaking, NMF can be considered as a bound-constrained optimization problem. Chu et al. [28] informed the strict first-order optimization model and the corresponding KKT condition. After this, some typical methods, such as Newton-type approaches, like sequential quadratic programming, alternating direction iteration Newton, and projected Newton method, reduced quadratic model approaches, and gradient approaches are used, respectively. The paper gives sight of some possible directions, where these algorithms are different in the performance and the convergences were not proved.

The "ANLS using projected gradient (PG) methods" proposed by Lin [56] is the crest of the previous work on Basic NMF, which makes headway in the bound-constrained optimization. The standard Lee-Seung algorithms actually minimize the auxiliary functions, leading to the non-increasing of the original objective functions. However, the minimization of the auxiliary functions does not amount to the minimization of the objective functions, and the difference between the two closely depends on the design of the auxiliary functions. This is the deep-seated reason for the slow convergence displayed by the standard algorithms. Thus the direct minimization of the objective functions might be helpful. By using PG methods which select the adaptive gradient decent step size based on the strategy called "Armijo rule along the projection arc", the ANLS optimization (6)(7) with warranted convergence is achieved. The efficiency of the algorithm rests on the convergence rate of the subproblems per iteration. With adjustment between faster convergence and lower complexity per iteration, this mode has also been extended in solving Constrained NMF.

In most situations, the first-order (gradient) optimization scheme is enough for approximate NMF with noise added. If a more accurate solution is necessary, we can then switch to performing optimization on second-order approximations in the Taylor expansion of the objective function; whereupon Zdunek and Cichocki [24] developed a projected quasi-Newton optimization approach. To avoid computing the inverse of the whole Hessians which are usually large and ill-conditioned, the regularized Hessians with the Levenberg-Marquardt approach are inverted by the Q-less QR factorization. Again, replacing the negative values with very small positive values makes theoretical analysis of the convergence difficult. The projection step can not guarantee monotonic decreasing of the objective function, which may lead to inaccurate approximation results. They [59] furthered the work using a more sophisticated hybrid approach based on Gradient Projection Conjugate Gradient (GPCB), where the Gradient Projection method is exploited to find zero-value components (active), and the Newton steps are taken only to compute positive components (inactive) with the Conjugate Gradient method.

Cichocki and Zdunek [60], [61] carried out a systematic evaluation and comparison of several PG based NMF algorithms including projected Landweber, Barzilai-Borwein gradient projection (GPSR-BB), projected sequential subspace optimization (PSESOP), interior-point Newton, and sequential coordinate-wise, with regard to their performance in terms of signal-to-interference ratio and elapsed time using a simple benchmark of mixed partially dependent nonnegative signals. Briefly speaking, the best and most promising algorithms are PSESOP, GPSR-BB, and interior-point gradient method. However, the final selection of the algorithms is problem size dependent.

Berry et al. [10] recommended ALS NMF algorithm by computing the solutions to the subproblems as unconstrained LS problems with multiple right-hand sides and maintaining non-negativity via setting negative values to zero per iteration. This approximate solution to constrained LS subproblem as unconstrained one, although fast, makes it difficult to analyze the convergence of the overall algorithm. In the framework of the two-block coordinate descent method where any limit point will be stationary point, Kim and Park [62] introduced a fast algorithm based on ANLS and the active set. The constrained LS problem in matrix formula with multiple right-hand side vectors is decoupled into several independent nonnegative LS problems with single right-hand side, each of whom is solved by using the active set method of Lawson and Hanson. The KKT condition based convergence criterion is also presented.

The central notion in ANLS or block coordinate descent consists in the partition of variables with convexity preserved. From (1), it is obvious that if every block does not contain $\boldsymbol{U}_{ml}$ and $\boldsymbol{V}_{ln}$ simultaneously, (i.e. an element of a column of $\boldsymbol{U}$ and an element of the corresponding row of $\boldsymbol{V}$), the optimization problem under this partition is convex. More precisely, given a subset of indices $K \subseteq R = \{1, 2, \ldots, L\}$, NMF is convex for the following two subsets of variables [37]

$$P_K = \{\boldsymbol{U}_{\bullet i} \,|\, i \in K\} \cup \{\boldsymbol{V}_{j\bullet} \,|\, j \in R \backslash K\} \tag{9}$$

and its complement

$$Q_K = \{\boldsymbol{U}_{\bullet i} \,|\, i \in R \backslash K\} \cup \{\boldsymbol{V}_{j\bullet} \,|\, j \in K\}. \tag{10}$$

In the standard NLS approach mentioned above, the problem (6) is further decoupled into M independent NLS subproblems in L variables corresponding to

each row of $\boldsymbol{U}$. However, the solutions to these subproblems are still nontrivial and relatively expensive. We may also find that this is not the only partition method as long as the convexity holds.

Li and Zhang [63], [64] proposed an exact block coordinate descent algorithm called FastNMF, which tries to optimize instead one single variable at a time in the above ANLS framework. The closed-form solutions to these subproblems can be directly obtained based on the convexity of a much simpler univariate parabola function. The analytical solutions of several variables can be further unified into a parallel vector formula according to the separability of the objective function. This partition mode results in much simpler iteration rules with easy implementation as well as quick and stable convergence. Cichocki and Phan suggested similar FASTHALS algorithm [65]. FastNMF or FASTHALS along with the accelerated versions is one of the best strategy for solving NMF problems, partly because NMF solutions are expected to be parts-based, which means columns of $\boldsymbol{U}$ (resp. rows of $\boldsymbol{V}$) will be almost disjoint (i.e. share few nonzero entries), makes the coupling between variables in the NLS subproblems rather low, and thereby allows an exact coordinate descent method to be capable of solving the nearly separable NLS subproblems efficiently [37].

3.4 New Vision of Basic NMF

The Basic NMF algorithms mentioned above are all based on the algebraic iterative optimization models, which have some drawbacks in common [38]: the solutions are usually sensitive to the initializations, and thus are not guaranteed to be unique. More importantly, the notion of optimizing an objective function is not obviously and necessarily equivalent to that of identifying the underlying actual components of the data sets, which is instead the ultimate destination of NMF. Hence Klingenberg et al. [38] proposed a seminal formulation coined the extreme vector algorithm (EVA), on the basis of the geometric interpretation of NMF [32]. For one thing, the dimensionality reduction is achieved by other matrix factorization preprocessing like SVD. For another, in the reduced space or nonsingular condition, EVA searches for the smallest simplicial cone containing overall data points, which is the most informative with respect to where the data are located in the positive orthant. This is identical to selecting the vertex vectors of the projected boundary polygon of the original data points on the unit hypersphere as the basis vectors. This manipulation decouples the functions of reducing dimensionality and identifying latent data structure into two independent stages, and thereby might yield better performance. It is original to think from the geometric view, whose formulation is independent of the chosen optimization methods and even the designated objective functions. Besides, EVA identifies more representative components compared with similarity measures based optimization. Meanwhile, the data are supposed to satisfy, at least approximately, the extreme data property, which guarantees the existence of basis vectors yet limits the distribution pattern of the data set. In such case, it achieves lower approximation error than SED-MU at the cost of higher computational consumption. However, to choose the boundary points as the basis vectors (i.e. to select the smallest simplicial cone) will discover the hidden concepts only in certain generative model. And the previous dimensionality reduction procedure, if inappropriate, might lose some information even though EVA itself is effective. Further more, the boundary outliers will strongly influence the final results. In fact, this paper has enlightened the scrutiny of how to choose the optimization goal so as to find the true prominent features.

The above EVA can be regarded as conic coding suggested in [66] under certain constraints imposed on the basis vectors in another sense. Correspondingly, Convex NMF [67] proposed by Ding et al. can be viewed as convex coding also in [66]. To be specific, the data set is generated by the basis vectors $\boldsymbol{u}_i$; reversely, the basis vectors $\boldsymbol{u}_i$ are also generated by the data set. To promote the traceability, $\boldsymbol{U}$ is confined as the convex combination of the columns of $\boldsymbol{X}$, namely $\boldsymbol{U}=\boldsymbol{XG}$ and $\boldsymbol{X}\approx\boldsymbol{XGV}$. For that matter basis vectors are tantamount to the centers of the data points. In addition, there is an appealing property of Convex NMF: without additional constraints, the factor matrices $\boldsymbol{G}$ and $\boldsymbol{V}$ are naturally sparse. The convergence of the algorithm is warranted too. The similar work was also recommended as Concept Factorization [68]. Finally, this generalization is helpful in developing Kernel NMF—$\phi(\boldsymbol{X})=\phi(\boldsymbol{X})\boldsymbol{GV}$.

Linear projection-based NMF (LPBNMF) suggested by Li and Zhang [69], where $\boldsymbol{V}=\boldsymbol{QX}$ and $\boldsymbol{X}\approx\boldsymbol{UQX}$, can be regarded as the dual problem of Convex NMF. This is similar to the subspace PCA under the non-negativity constraint. The main motivations consist in that $\boldsymbol{V}$ is the nonlinear projection of $\boldsymbol{X}$ on $\boldsymbol{U}$, which can only be achieved by iterative procedure thus leading to low efficiency of the algorithm; besides, the sparse features will not be necessarily learnt by Basic NMF. By contrast, the dimensionality reduction in LPBNMF is achieved by linear transformation, which combines the optimization methods of foregoing FastNMF [63], [64] with the multiplicative update rules. The convergence and stability of the algorithm were also proved. Analogous with Convex NMF, LPBNMF has the properties of making $\boldsymbol{U}$ and $\boldsymbol{Q}$ inclined to be orthogonal in the column and row directions, respectively, which means sparseness under non-negativity. Yuan, Yang, and Oja also constructed a similar framework [70], [71].

Given that computing a globally optimal rank-one approximation can be done in polynomial time while the general NMF problem is NP-hard, Gillis

and Glineur [72] introduced non-negative matrix underapproximation (NMU) to solve the higher rank NMF problem in a recursive way. After identification of an optimal rank-one NMF solution $(\boldsymbol{u}, \boldsymbol{v})$, the similar rank-one factorization can be performed on the residue matrix $\boldsymbol{R} = \boldsymbol{X} - \boldsymbol{u}\boldsymbol{v}^T$ sequentially. Besides, to maintain the non-negativity for $\boldsymbol{R}$ so as to make the decomposition keep on, an upper bound constraint $\mathbf{UV} \leq \mathbf{X}$ is introduced. Then an algorithm based on Lagrangian relaxation was proposed to find approximate solutions to NMU. Similar to the previous variants of NMF, the additional constraint of NMU is shown to generate sparser factors, leading naturally to a better parts-based representation with low reconstruction error.

3.5 Pragmatic Issue

In practice for real-world problems with big matrices like Netflix or of ill-conditioned and badly-scaled nature, nothing can be a simple gradient descent of the SED cost functions. The situation varies greatly even for dense and sparse data matrices. Here we mainly focus on two kinds of situations: one is for large-scale data, and the other is online processing.

For large-scale NMF where $L << M, L << N$, $\boldsymbol{X}$ is usually low rank, which implies that the problem $\boldsymbol{X} \approx \boldsymbol{U}\boldsymbol{V}$ becomes highly redundant and thus there is no need to process all elements of $\boldsymbol{X}$ to estimate $\boldsymbol{U}$ and $\boldsymbol{V}$ precisely. In other words, a proper chosen subset of the whole data matrix is enough. To be specific, Cichocki and Phan [65] switched the large-scale factorization to two sets of linked factorizations on much smaller block matrices:

$$\begin{cases} \boldsymbol{X}_r \approx \boldsymbol{U}_r\boldsymbol{V}, & \text{for fixed } \boldsymbol{U}_r \quad (11) \\ \boldsymbol{X}_c \approx \boldsymbol{U}\boldsymbol{V}_c, & \text{for fixed } \boldsymbol{V}_c \quad (12) \end{cases}$$

where $\boldsymbol{X}_r \in \mathbb{R}_{\geq 0}^{R \times N}$ and $\boldsymbol{X}_c \in \mathbb{R}_{\geq 0}^{M \times C}$ are submatrices constructed from the preselected rows and columns of $\boldsymbol{X}$, respectively, and $\boldsymbol{U}_r \in \mathbb{R}_{\geq 0}^{R \times L}$ and $\boldsymbol{V}_c \in \mathbb{R}_{\geq 0}^{L \times C}$ are reduced matrices by using the same indices for the rows and columns as those used in $\boldsymbol{X}_r$ and $\boldsymbol{X}_c$, accordingly. Here $R << M, C << N$. The minimization of the two corresponding cost functions can then be performed sequentially. As for the selection of the submatrices, there are several strategies [65]. For instance, we can randomly select R rows and C columns of $\boldsymbol{X}$, or choose such rows and columns that provide the largest ℓ_p norm values. An alternative is to choose the cluster representatives after clustering $\boldsymbol{X}$ into R rows and C columns.

Besides, much better performance can be achieved by using suitable optimization algorithms for large-scale data sets. Some projected gradient methods, such as interior-point gradient, quasi-Newton, and projected sequential subspace optimization, are promising candidates due to their high efficiency [61], [73]. Another consideration to decrease the computational consumption is the parallel implementation of the existing Basic NMF algorithms, which tries to divide and distribute the factorization task block-wisely among several CPUs or GPUs [74].

In other situations, the data steam continuously arrives in a sequential manner. This online processing characteristic demands NMF to update the latent factors incrementally by combining the old factors with the newly arrived data. Apart from reducing the computational load and storage demand caused by the batch nature of conventional static NMF, it will be able to track the dynamic structures and evolutionary patterns of latent components naturally. It can also be viewed as the incremental update mechanism of NMF similar to incremental EM and incremental PCA.

A natural assumption is that the new individual sample will not make a big difference on the optimality of the basis matrix learnt from the previous data set [75], [76]. For that matter it is adequate to update only the basis matrix $\boldsymbol{U}$ along with the coefficient vector $\boldsymbol{v}_{N+1}$ of $\boldsymbol{x}_{N+1}$, while maintaining the coefficient vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_N$ of old samples from $\boldsymbol{x}_1$ to $\boldsymbol{x}_N$ when the latest sample $\boldsymbol{x}_{N+1}$ is added. By assigning different weights to the residuals to differentiate the new sample from the old ones in the objective function, this can be viewed and solved as a special case of Weighted NMF. Another trick called Block NMF applies NMF in the categorized subsets obtained by the prediscriminant procedure [77]. When a new category or a new sample in the existing categories arrives, it is reduced to update only the relevant category rather than the overall data set at the cost of requiring some additional prior classification information. Besides, a theoretical formulation along with the approximate analytical solution to the incremental problem was proposed in [78], where the newly updated basis matrix is linearly transformed from the previous one as a tolerable approximation when the real relation is nonlinear.

3.6 Other Open Issues

Because of the local rather than global minimization characteristic, it is obvious that the initialization of $\boldsymbol{U}$ and $\boldsymbol{V}$ will directly influence the convergence rate and the solution quality. Algorithms with poor starting conditions might converge slowly to incorrect or even irrelevant solution. The random seeding adopted by Lee and Seung [4] is unable to give a desirable initial estimation. Wild et al. utilized spherical k-means clustering [79], which is much more complex as preprocessing and might stop at relatively poor local solution. Some other alternatives include initialization via SVD [80], relaxed k-means clustering [81], PCA, fuzzy clustering, Gabor wavelet [82], population based algorithms [83], and the like [84]. However, the problem of selecting suitable starting matrices can not be put in a nutshell, which is indeed situation dependent. A seeding method appropriate for one

data set is possibly poor for another. There are some extra considerations in large-scale data processing task, such as the computational cost of the initial guess itself. The additional imposed constraints or structures will make things more complicated too. In practice, incorporating different initialization approaches might be helpful.

To mitigate the problem of local minima, Cichocki and Zdunek [85], [60] recommended a simple yet effective approach named multi-layer NMF by replacing the basis matrix $\boldsymbol{U}$ with a set of cascaded factor matrices. That is $\boldsymbol{X} \approx \boldsymbol{U}^{(1)}\boldsymbol{U}^{(2)} \cdots \boldsymbol{U}^{(T)}\boldsymbol{V}$. The initial factorization is the standard NMF $\boldsymbol{X} \approx \boldsymbol{U}^{(1)}\boldsymbol{V}^{(1)}$, and then the decomposition is performed on the last obtained factor matrix $\boldsymbol{V}^{(i)}$ repeatedly and sequentially, layer-by-layer, given the previous factorization result $\boldsymbol{V}^{(i-1)} \approx \boldsymbol{U}^{(i)}\boldsymbol{V}^{(i)}$. Moreover, different update rules and initial conditions are allowed in the cascade system. Due to its distributed structure, this hierarchical multistage procedure combined with multistart initialization will greatly reduce the risk of converging to local minima of cost functions and enhance the performance of NMF algorithms. Interestingly, this is somewhat similar to [30], where the former is accumulation while the latter is continued multiplication.

A systematic mechanism to determine the number L of the basis vectors has not been established effectively by now, partly due to its difficulty. The algorithms designed above assume L is prespecified. In practice, the trial and error approach is often adopted, where L is set in advance and then adjusted according to the feedback of the factorization results, such as the approximation errors. Or an alternative is to choose the value of L, from the candidate pools, corresponding to the best factorization result. Sometimes, the prior knowledge in the specific applications, such as the number of categories in clustering, can be incorporated in the procedure. If there were more adequate theoretical results about the non-negative rank of matrix, this situation might have been ameliorated.

Another ignored issue is the choice of objective functions. Among them, SED is the most widely used and deeply investigated. Different objective functions correspond to varied probability distribution assumptions. However, the original NMF model has no such prior statistical presumptions on the data set, and these objective functions are most effective when the data set matches the statistical distribution. So to what extent these additional assumptions would affect the problem, and how to select and evaluate different objective functions need to be plumbed in depth.

4 Constrained NMF Algorithms

As discussed previously, Basic NMF will not get the unique solution under the sole non-negativity constraint. Hence, to remedy the ill-posedness, it is imperative to introduce additional auxiliary constraints on $\boldsymbol{U}$ and/or $\boldsymbol{V}$ as regularization terms, which will also incorporate prior knowledge and reflect the characteristics of the issues more comprehensively.

The various Constrained NMF models can be unified under the similar extended objective function

$$D_C(\boldsymbol{X} \| \boldsymbol{U}\boldsymbol{V}) = D(\boldsymbol{X} \| \boldsymbol{U}\boldsymbol{V}) + \alpha J_1(\mathbf{U}) + \beta J_2(\boldsymbol{V}) \quad (13)$$

where $J_1(\boldsymbol{U})$ and $J_2(\boldsymbol{V})$ are the penalty terms to enforce certain application dependent constraints, α and β are small regularization parameters balancing the trade-off between the fitting goodness and the constraints. The optimization problem of (13) can be solved by modifying existing Basic NMF algorithms.

According to different formula of $J_1(\boldsymbol{U})$ and $J_2(\boldsymbol{V})$, Constrained NMF algorithms are categorized into four subclasses: (1) Sparse NMF (SPNMF); (2) Orthogonal NMF (ONMF); (3) Discriminant NMF (DNMF); (4) NMF on manifold (MNMF).

4.1 Sparse NMF

The sparseness constraint is helpful in improving the uniqueness of the decomposition along with enforcing a local-based representation. Sparse NMF is the most widely and deeply mined one in overall Constrained NMF problems, and has nearly been a necessity in practice. What must be addressed here is which factor matrix, $\boldsymbol{U}$ or $\boldsymbol{V}$, is selected as the candidate on which the sparseness constraint is imposed [86]. In fact this question is application dependent. If the basis vectors, columns of $\boldsymbol{U}$, are sparse, they themselves are parts-based; in other words, every basis influence only a small part of each observation. If columns of $\boldsymbol{V}$ are sparse, each observation is approximated by a linear combination of a limited number of basis vectors. If rows of $\boldsymbol{V}$ are sparse, then each basis vector is used to approximate a limited number of training data or a limited number of training data are used to infer each basis vector, which is related with clustering closely. The paragon work of Sparse NMF is Non-negative Sparse Coding (NSC) [87], employing the combination of SED and the ℓ_1-norm of $\boldsymbol{V}$, the sparseness penalty term, as the objective function, and NMF with Sparseness Constraints (NMFSC) [88], using SED as the objective function and enforcing sparseness by means of nonlinear projection at each iteration based on the sparseness measure from the relationship between ℓ_1 and ℓ_2 norms, proposed by Hoyer in year 2002 and 2004, respectively.

These two objective functions are usually adopted, such as NSC [89], [90] and NMFSC [86]; meanwhile, some more effective approaches are proposed to resolve the optimization problems. In general, the solution to the factor matrix not to be sparse is still effective by using Basic NMF approaches, while the solution to the candidate matrix to be sparse is supposed to be modified under the sparseness constraint.

One of the highlights in NSC is to employ active set based algorithm to solve the least squares minimization under the non-negativity and sparseness constraints, i.e. modified least angle regression and selection (LARS) [89] and NLS [26]. LARS is generally applied as an effective method for unbounded SED minimization under the ℓ_1 norm constraint. Mørup et al. [89] modified it under the non-negativity constraint as NLARS to solve the NSC optimization problem. Since LARS is to solve the LASSO problem based on the active set algorithm, the modified algorithm merely introduces the non-negativity constraint in the determination of the active and non-active set by removing the elements updated to zero from the active set. This recasts the original Sparse NMF problem into N separable LASSO subproblems. Similarly, NLS is another effective approach to solving SED minimization based on the active set. Then ANLS is selected by Kim and Park to optimize their objective function [26], in which every subproblem is solved by the fast NLS algorithm [25]. They proved that this algorithm converges to a stationary point.

Unlike the above active set based solution methods, Li and Zhang [90] designed a stable and efficient NSC algorithm called SENSC, which is alternating minimization based on the properties of convex hyperparabolic functions, the properties of parabola functions, the projection rule from one point to the unit hypersphere at the origin, and the projection rule from one point to the non-negative data set. In fact, SENSC can be viewed as modified FastNMF [63] under the sparseness constraint. The convergence and stability of SENSC were also proved.

As for NMFSC framework, Mohammadiha and Leijon [86] introduced additional nonlinear projection in the existing PG algorithms for Basic NMF to enforce sparsity. And then they compared these PG algorithms in terms of efficiency and execution time.

Apart from NSC and NMFSC, some other objective functions are also suggested. The ℓ_2 norm of $\boldsymbol{V}$ is selected as the sparseness penalty term in the regular SED objective function by Gao and Church [91]. Unfortunately, the quadratic penalty may lead to low rather than sparse values. In this sense, the sparse formula based on the ℓ_1 norm would be more effective in controlling sparsity than that based on the ℓ_2 norm [92]. Besides, the sparse regulatory factor has a scaling effect because a large value of this factor would suppress the ℓ_2 norm of $\boldsymbol{V}$, rending a larger value of the ℓ_2 norm of $\boldsymbol{U}$. As such, the column normalization of $\boldsymbol{U}$ during iterations is needed. Another a little more complex objective function involves the linear combination of SED, summation of squares of the ℓ_1 norm of the vectors in the candidate matrix to be sparse, and the Frobenius norm of the other matrix not to be sparse [26]. The second term is the sparseness penalty term as usual. And the third term is to suppress the corresponding matrix so as to lower its element values and mitigate the scaling effect mentioned above, which is the conspicuous difference from other Sparse NMF algorithms.

It is another concern to seek for the mechanism to control the sparsity desired. Unlike that based on the row vectors of $\boldsymbol{V}$ given by Hoyer, two sparsity measures are provided based on the ℓ_0 and ℓ_1–ℓ_2 norms of matrix $\boldsymbol{V}$, respectively [89]. Hence, the specific sparsity can be controlled by a single parameter.

Besides relying on additional explicit constraints on the factors like the previous work, sparsity can be also achieved by incorporating some structure information such as Convex NMF [67] and LPBNMF [69], or modifying the factorization mode slightly such as NMU [72] and Affine NMF [93]. Laurberg and Hansen [93] introduced an offset vector in the approximate term and obtained Affine NMF in the form of $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{V} + \boldsymbol{u}_0\mathbf{1}^T + \boldsymbol{E}$ where $\mathbf{1} \in \mathbb{R}^N$ is a vector of all ones and $\boldsymbol{u}_0 \in \mathbb{R}^M$ is the DC bias vector so as to remove the base line from $\boldsymbol{X}$. The offset absorbs the constant values of data matrices, thus making the factorization sparser and enhancing the uniqueness with respect to the possible offset caused by the additive noise.

4.2 Orthogonal NMF

Orthogonal NMF is NMF with orthogonality constraint on either the factor $\boldsymbol{U}$ or $\boldsymbol{V}$. The orthogonality principle was firstly employed by Li et al. [43] to minimize the redundancy between different bases, and then Ding et al. [16] broached the concept of Orthogonal NMF explicitly. In the condition of non-negativity, orthogonality will necessarily result in sparseness. Thus it can be viewed as a special case of Sparse NMF. However, there is a notable difference in the optimization models between these two. Moreover, the result of Orthogonal NMF corresponds to a unique sparse area in the solution region, which learns the most distinct parts. In this sense, it is necessary to probe Orthogonal NMF separately.

If the basis vectors, columns of $\boldsymbol{U}$, are orthogonal, namely $\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}$, it obtains by all means the most distinct parts. If rows of $\boldsymbol{V}$ are orthogonal, that is $\boldsymbol{V}\boldsymbol{V}^T = \boldsymbol{I}$, the orthogonality improves clustering accuracy. The orthogonality can also be imposed on both $\boldsymbol{U}$ and $\boldsymbol{V}$, so-called bi-orthogonality, which nevertheless has poor approximation performance in most cases. Another highlight is the clustering interpretation of Orthogonal NMF, and it has been showed that Orthogonal NMF amounts to *k*-means clustering [44], [45]. Orthogonal NMF on $\boldsymbol{U}$ or $\boldsymbol{V}$ is identical to clustering the rows or columns of an input data matrix, respectively, where one matrix corresponds to the cluster centers (prototypes) and the other is associated with the cluster indicator vectors. Due to this kind of interpretation and relationship, Orthogonal NMF is preferable in clustering tasks [16], [94], [95].

There exist two typical orthogonality penalty terms which are embedded into the SED or GKLD objective function. One involves the trace of the difference matrix between the factor matrix and the identity matrix, i.e. the first order quantity [16]. The other consists of the ℓ_2 norm of the difference matrix between the factor matrix and the identity matrix, i.e. the second order quantity [94], which is similar to the optimization model used in Sparse NMF.

The former is an optimization problem under the orthogonality constraint, which can be obviously solved as unconstrained one by Lagrange multiplier approach. For that matter the corresponding multiplicative update rule similar to the standard algorithms is obtained [16]. However, the Lagrange multiplier is a symmetrical matrix with many parameters, which increases the computational load. As such, the orthogonality in the latter formulation is controlled by a single parameter, decreasing the computational complexity. In like manner, the modified multiplicative update rule can be applied in the solution [94].

Another direction in the solution is to directly consider Stiefel manifold, a parameter space possessing the orthogonality constraint itself, and apply the canonical gradient decent in this subspace. Accordingly, Choi [95] converted NMF with the orthogonality constraint into Stiefel manifold with the nonnegativity constraint. And then the corresponding multiplicative update rule with a lower computational complexity is achieved, which simplifies that in Ding's algorithms [16]. Yoo and Choi [96] further utilized this principle to obtain the multiplicative update rule for orthogonal non-negative matrix tri-factorization (ONMTF), i.e. the bi-orthogonality imposed on both $\boldsymbol{U}$ and $\boldsymbol{V}$ (NMTF will be discussed in the following sections).

An intuitive description of the above modified multiplicative update rules under the orthogonality constraint is to replace certain term in the original multiplicative update rules with a new one somewhat implying orthogonality. For instance, when $\boldsymbol{U}^T\boldsymbol{U}=\boldsymbol{I}$, the update rule in [95] substitute $\boldsymbol{X}^T\boldsymbol{U}$ for $\boldsymbol{V}^T$ in the denominator. This makes sense since the coefficients incline towards orthogonal projection in this case.

4.3 Discriminant NMF

From the perspective of pattern recognition, Basic NMF can be considered as unsupervised learning. By coupling discriminant information with the decomposition, Basic NMF is further extended to supervised alternatives—the so-called Discriminant NMF or Fisher-NMF (FNMF) methods—so as to unify the generative model and the classification task into a joint framework. It is now being successfully utilized in classification based applications, such as face recognition and facial expression recognition. Wang et al. [97] at first introduced the Fisher discriminant constraint, i.e. the difference between the within-class scatter and the between-class scatter, as the penalty term in GKLD to construct the objective function.

This formula is often accepted as the basic framework, whereas some modifications are introduced, such as the definitions of the between-class scatters [20] and choosing SED instead of GKLD given that the latter is not well defined on the boundary [21]. It is to be remarked that the definitions of the within-class and between-class scatters in [97], [20] are both solely based on the coefficient matrix $\boldsymbol{V}$, having noting to do with $\boldsymbol{X}$ or $\boldsymbol{U}$. Since the actual classification features are closely relevant to the projection matrix $\boldsymbol{U}^T\boldsymbol{X}$, while only having an indirect connection with $\boldsymbol{V}$, $\boldsymbol{U}^T\boldsymbol{X}$ rather than $\boldsymbol{V}$ is employed in the construction of the discriminant constraint [21]. This operation makes the basis vectors somewhat sparse with distinct parts-based features helpful for classification.

Regarding the fact that the existing Discriminant NMF algorithms cannot be guaranteed to converge, Kotsia et al. [21] suggested PGDNMF algorithm based on the PG methods in Basic NMF to solve this problem. The projected gradient method proposed by Lin [56] is applied so as to warrantee that the limit point is the stationary point.

The above work can be viewed as combining NMF with LDA classifer. Another direction is to incorporate in the NMF model the maximum margin classifier, especially a support vector machine (SVM) [19], which is more informative and preferable for the classification task. It will also benefit from the nonlinear kernels of SVM for linearly inseparable cases.

4.4 NMF on Manifold

In some situations, the real-world data are often sampled from a nonlinear low dimensional submanifold embedded in a high dimensional ambient space, which is locally flat and looks like a Euclidean space. It has been shown that the learning performance can be significantly enhanced if the intrinsic geometrical structure is identified and preserved. There have been numerous manifold learning algorithms, such as ISOMAP, Locally Linear Embedding (LLE), Laplacian Eigenmaps, and the like. They differ from one another in the local topological property to be considered—the local relationship between a point and its neighboring points. While Basic NMF fits data in a Euclidean space, NMF on manifold is to explicitly incorporate NMF with the proper local invariant properties and corresponding manifold learning methods, leading to highly improved performance in tasks like document and image clusterings [98]. The available incorporating approach is to combine the geometrical information in the original NMF objective function as the additional regularization term.

Graph regularized NMF (GRNMF) proposed by Cai et al. [99], [98] modeled the manifold structure by constructing a nearest neighborhood graph on a scatter

of data points. They borrowed the local invariance assumption that the points in the mapped low dimensional space should be close enough with one another if they are neighbors in the original high dimensional space. To access this aim, the weighted squares of the Euclidian distances of the data points in the reduced space are added in the SED objective function as the penalty term, and the modified multiplicative update rules are utilized to resolve it. This is equivalent to integrating NMF with Laplacian Eigenmaps. Zhi et al. [100], [101] developed analogous work, where the linear projection $\boldsymbol{U}^T\boldsymbol{x}_j$ on the basis matrix $\boldsymbol{U}$ rather than $\boldsymbol{v}_j$ is chosen as the low dimensional representation for good classification performance similar to Discriminant NMF [21].

Zhang et al. [102] also considered the local invariance assumption but in a quite different way. Given that the norm of the gradient of a mapping H from the low dimensional manifold to the original high dimensional space provides the measure of how far apart H maps nearby points, a constrained gradient distance minimization problem is formulated, whose goal is to find the map that best preserves local topology. And then an alternating gradient descent algorithm is devised for this topology preserving NMF. Moreover, this optimization model is tantamount to minimizing the square of total variation norm between $\boldsymbol{X}$ and $\boldsymbol{UV}$ under the non-negativity constraint, which preserves finer scale features compared with SED function.

Isometirc NMF developed by Vasiloglou et al. [31] was referred to maximum furthest neighbor unfolding (MFNU) which preserves the local distance and tries to maximize the distance between the furthest neighbors. When the nearest and furthest neighbors are obtained by kd-tree search, the optimization is then cast as a semidefinite programming problem. The convex and non-convex formulations suitable for large-scale data sets are presented.

Another important local topological property is the locally linear embedding assumption that the data point generated as a linear combination of several neighboring points on a specific manifold in the original space should be reconstructed from its neighbors in a similar way or by the same reconstruction coefficients in the reduced low dimensional subspace. Gu and Zhou [103] suggested the neighborhood preserving NMF by exploiting this property, and derived the modified multiplicative update rules accordingly. This is to combine NMF with Locally Linear Embedding. Shen and Si [104] furthered this work by modeling the locally linear relationship from single manifold to multiple manifolds, which approximates the data point by a linear combination of nearby samples only on the same manifold. Technically speaking, these two algorithms are different from each other in the determination of the neighboring sample points. The latter adopts the sparsest linear combination by ℓ_1 norm minimization to approximate the target data point rather than the fixed k-nearest neighborhood which is chosen in the former. This will be preferable when data reside on multiple manifolds which may overlap or intersect.

These constraints are in fact mutually complementary. In practical usage, the sparseness constraint is often considered as a necessity for NMF approaches. Discriminant Sparse NMF [105], Manifold-respecting Discriminant NMF [106], and Manifold Regularized Discriminative NMF [107] are appropriate illustrations of the notion of integrating different constraints, so as to achieve better decomposition quality reflecting the multilateral characteristics of the problems.

5 Structured NMF Algorithms

Structured NMF enforces other characteristics or structures in the solution to NMF learning problem. It usually modifies the regular factorization formulation directly rather than introduces some additional constraints as penalty terms in contrast to Constrained NMF. Formally, it can be written as

$$\boldsymbol{X} \approx F(\boldsymbol{UV}). \tag{14}$$

Specifically, Structured NMF algorithms are divided into three subclasses: (1) Weighed NMF (WNMF); (2) Convolutive NMF (CVNMF); (3) Non-negative Matrix Tri-factorization (NMTF).

5.1 Weighted NMF

Weighed formulations are commonly modified versions of learning algorithms, which can be utilized to emphasize the relative importance of different components. By introducing the weight matrix $\boldsymbol{W}$, the weighted NMF model has the following formula:

$$\boldsymbol{W} \otimes \boldsymbol{X} \approx \boldsymbol{W} \otimes (\boldsymbol{UV}). \tag{15}$$

Generally speaking, Weighted NMF can be viewed as a case of weighted low-rank approximation (WLRA), which seeks for a low-rank matrix that is the closest to the input matrix according to predefined weights.

If the original data matrix is incomplete with some entries missing or unobserved, it is supposed to predict the missing ones when decomposition, which is often referred to as low-rank matrix completion with noise, remarkably employed in collaborative filtering like designing recommendation systems. Such problem can be tackled by assigning binary weights to the data matrix, which is to set observed elements one and unknown elements zero, and constructing the corresponding weight matrix $\boldsymbol{W}$.

Weighted NMF can be solved by introducing the weight matrix in the standard multiplicative update rules, such as the Mult-WNMF algorithm proposed by Mao and Saul [108]. However, this simple trick makes the algorithms converge slowly. An alternative suggested by Zhang et al. is to employ the EM algorithm where missing entries are replaced by the

corresponding values in the current model estimate at the E-step and the standard unweighted multiplicative update rules are applied on the filled-in matrix at the M-step [109]. Basically speaking, EM-WNMF is superior to Mult-WNMF because of more accurate estimation obtained from EM procedure, but it also suffers from slow convergence. Moreover, the filled-in matrix at the E-step, in general, is a dense matrix even if the original matrix is very sparse, increasing the computational load greatly. There is likelihood that the prediction is not accurate enough [110].

To enhance the convergence rate of Weighted NMF, Kim and Choi recommended two approaches [110]. One is to apply ANLS instead of the multiplicative update rules and utilize Projected Newton method to solve the NLS subproblems, which is tantamount to the improvement of the previous Mult-WNMF algorithm. The other highlight is to use the generalized EM model interweaving E-step and partial M-step coined GEM-WNMF. The basic gist involves two aspects. Firstly, ANLS is switched to optimize the problems at the M-step. Secondly, the partial M-step is chosen, in which iterations are stopped when obtaining substantial improvement instead of determining optimal solutions, on the basis of the fact that estimation for missing entries is not accurate at earlier iterations and thus solving M-step exactly is not desirable. This corresponds to the modification of EM-WNMF, which reduces the computational complexity as well as promoting the prediction accuracy.

In addition, although the Weighted NMF algorithms mentioned above are proposed based on the special case of handling incomplete data matrix, they are indeed applicable in the general Weighted NMF models.

5.2 Convolutive NMF

The notion of Convolutive NMF mainly comes from the application of source separation. Conventional Basic NMF can be regarded as a kind of instantaneous decomposition, where each object is described by its spectrum, the basis matrix $\boldsymbol{U}$, and corresponding activation in time, the coefficient matrix $\boldsymbol{V}$. To incorporate the time domain information, in other words, the potential dependency between the neighboring column vectors of the input data matrix $\boldsymbol{X}$, it is necessary to take into account the time-varying characteristic of the spectrum. Typically, the temporal relationship between multiple observations over nearby intervals of time is described using a convolutive generative model. Hence, Basic NMF is further extended to Convolutive NMF form, the summation of products of a sequence of successive basis matrices $\boldsymbol{U}_t$ and corresponding coefficient matrices $\overset{t\rightarrow}{\boldsymbol{V}}$, where $\boldsymbol{U}_t$ varies across time, while $\overset{t\rightarrow}{\boldsymbol{V}}$ satisfies the relationship of right shift and zero padding which can be simplified to a coefficient matrix prototype $\boldsymbol{V}$. Formally, it can be described as

$$\boldsymbol{X} \approx \sum_{t=0}^{T-1} \boldsymbol{U}_t \overset{t\rightarrow}{\mathbf{V}} \tag{16}$$

where $\overset{0\rightarrow}{\boldsymbol{V}} = \boldsymbol{V}$. This amounts to denoting the input data matrix as the convolution of the basis matrix and the coefficient matrix. Hence, Convolutive NMF can be decomposed into a series of Basic NMF problems from the computational perspective. To some extent Convolutive NMF is a case of overcomplete representation.

Smaragdis [111], [112] initiated the above Convolutive NMF model, and cast it as the solutions to the basis matrix sequence $\boldsymbol{U}_t$ and the coefficient primary matrix $\boldsymbol{V}$. Both GKLD [111], [112] and SED [22] objective functions can be employed. By comparison, the SED objective function decreases the computational complexity while possessing better separable performance. As for the solution, the multiplicative update rules are generalized to the case of multiple factor matrices. Nevertheless the simple generalization might not converge.

Similar to Sparse NMF, the NSC sparseness constraint [87] was incorporated in the Convolutive NMF model by O'Grady and Pearlmutter [113] with respect to the separability attributable to sparsity. $\boldsymbol{V}$ is updated as the standard multiplicative rules, while $\boldsymbol{U}_t$ is updated according to the traditional additive gradient descent method, due to the absence of an appropriate adaptive step size like the one in the standard algorithm. Furthermore, the convergence of the algorithm was not guaranteed.

An analogous and complementary framework called sparse shift-invariant NMF (ssiNMF) was suggested by Potluru et al. [114] to learn possibly overcomplete shift-invariant features by extending the original basis matrix $\boldsymbol{U} \in \mathbb{R}_{\geq 0}^{M \times L}$ to $\boldsymbol{G} \in_{\geq 0}^{M \times (M \times L)}$ formed by the circular shift of the feature vectors. While the coefficient matrix $\boldsymbol{V}$ in the previous Convolutive NMF model owns the property of right shift and zero padding, the basis matrix $\boldsymbol{U}$ here becomes the set of the basis vectors and their circular shift counterparts, which is helpful to describe the shifted data, such as the uncalibrated image data. Besides, the NSC sparseness constraint [87] is combined with SED as the objective function.

In a sense conventional NMF belongs to frequency domain analysis, whereas Convolutive NMF is a branch of time-frequency domain analysis. With the convolutive model, the temporal continuity of the signals, whose frequencies vary with time in particular, can be expressed more effectively in the time-frequency domain. However, the scope of application of this generalization form is limited only in audio data analysis. Moreover, the simple modification of the multiplicative update rules adopted by most Convolutive NMF model might not converge.

5.3 NMTF

NMTF extends conventional NMF to the product of three factor matrices, i.e. $\boldsymbol{X} \approx \boldsymbol{USV}$ [96]. Unconstrained 3-factor NMTF makes no sense since it can be merged into unconstrained 2-factor NMF; however, when constrained, 3-factor NMTF provides additional degrees of freedom, thus endowing NMF with new features.

Taking into account that the bi-orthogonality both on $\boldsymbol{U}$ and $\boldsymbol{V}$ in Orthogonal NMF is very restrictive which will lead to a rather poor low-rank matrix approximation, an extra factor $\boldsymbol{S}$ is introduced [96] to absorb the different scales of $\boldsymbol{U}$ and $\boldsymbol{V}$ such that the low-rank matrix representation remains accurate while satisfying the orthogonality constraint. Hence the rows and columns of $\boldsymbol{X}$ can be clustered simultaneously, which is quite useful in text analysis and clustering. In fact this is identical to the previous work of Non-smooth NMF (NSNMF) [115], where the incorporation of a very smooth factor $\boldsymbol{S}$ makes $\boldsymbol{U}$ and $\boldsymbol{V}$ quite sparse, and thus reconciles the contradiction between approximation and sparseness.

Convex NMF [67] and LPBNMF [69] mentioned previously can also be considered as extending NMF to tri-factorization; however, their goals are different from the above models in essence. Besides, NMTF can be somewhat viewed as a special case of more general formulation of multi-layer NMF [85].

6 Generalized NMF Algorithms

Generalized NMF might be considered as extensions of NMF in a broad sense. In contrast to introducing some additional constraints as penalty terms in Constrained NMF, Generalized NMF has extended the decomposition model itself in depth a little bit similar to Structured NMF. It breaches the intrinsic non-negativity constraint to some extent, or changes the data types, or alters the factorization pattern, and so on. This is the latest emerging field with several preliminary results compared with Basic, Constrained, and Structured NMF. Here Generalized NMF algorithms are summarized into four subclasses for the present: (1) Semi-NMF; (2) Non-negative Tensor Factorization (NTF); (3) Non-negative Matrix-set factorization (NMSF); (4) Kernel NMF (KNMF), where (1) breaches the non-negativity constraint, (2) and (3) popularize the data type into high dimensionality, and (4) alters the factorization pattern into nonlinear formulation. The details will be discussed in the following sections.

6.1 Semi-NMF

Conventional NMF restricts every element in data matrix $\boldsymbol{X}$ to be non-negative. When $\boldsymbol{X}$ is unconstrained, which may have mixed signs, Ding et al. [67] suggested an extended version referred to as Semi-NMF which remains some kernel concepts of NMF, where $\boldsymbol{V}$ is still restricted to be non-negative while placing no restriction on the signs of $\boldsymbol{U}$.

This form of generalization makes sense in that the candidate data in practical applications are not always non-negative, so the latent features or principal components might also have some negative elements reflecting the phase information. This indeed has the physical interpretation as NMF. However, the non-subtractive combinations are still effective.

Under such conditions, the analogous status of $\boldsymbol{U}$ and $\boldsymbol{V}$ in Basic NMF are undermined. Ding et al. employed an alternating iteration approach to solve the optimization problem, where the positive and negative parts are separated from the mixed-sign matrix. $\boldsymbol{V}$ is updated using multiplicative rules while holding $\boldsymbol{U}$ fixed, and then the analytical local optimal solution for $\boldsymbol{U}$ is obtained with $\boldsymbol{V}$ fixed. The convergence was also proved. Further more, Semi-NMF applies equally to the foregoing Convex NMF model [67].

6.2 NTF

Conventional methods preprocess multiway data by arranging them into a matrix, which might lose the original multiway structure of the data. By contrast, a natural generalization of matrix factorization is tensor factorization. And NMF is a particular case of non-negative n-dimensional tensor factorization (n-NTF) when $n = 2$. In fact Welling and Weber [116] first put forward the concept of Positive Tensor Factorization (PTF) not long after the recommendation of NMF. Compared with other formulations of GNMF, NTF has attracted widely attention recently.

This kind of generalization is indeed not trivial since NTF possesses many new properties varying from NMF [117], [118]. Firstly, the data to be processed in NMF are vectors in essence. However, in some applications the original data may not be vectors, and the vectorization might result in some undesirable problems. For instance, the vectorization of image data, which is two dimensional, will lose the local spatial and structural information. Secondly, one of the core concerns in NMF is the uniqueness issue, and to remedy the ill-posedness some strong constraints have to be imposed. Nevertheless, tensor factorization will be unique under only some weak conditions. Besides, the uniqueness of the solution will be enhanced as the tensor order increases.

There are generally two types of NTF model—NTD [119] and more restricted NPARAFAC [117], whose main difference lies in the core factor tensor. As for the solution, there are some feasible approaches. For example, NTF can be restated as regular NMF by matricizing the array [116], [119]. Or the alternating iteration method can be utilized directly on the outer product definition of tensors [117], [118], [23]. Similarly, SED, GKLD and other forms of divergence can also be used as the objective functions [23], [120], [121].

And some specific update models can adopt the existing conclusions in NMF. For thorough understanding one may refer to [9], [122]. What must be scrutinized here is that the convergence of these algorithms is not guaranteed by the simple generation from matrix to tensor forms in itself.

What's more, the concepts in Constrained NMF can also be incorporated in NTF, such as sparse NTF [123], [124], [125], discriminant NTF [126], NTF on manifold [127], and the like.

6.3 NMSF

Li and Zhang [128] proposed the generalization formulation referred to as Non-negative Matrix-Set Factorization (NMSF), with respect to the fact that the corresponding learning problem will become a notorious small sample problem if vectorizing the original matrix-type, like image, data, leading to unsatisfactory approximation, poor generality, and high computational load. NMSF is implemented directly on the matrix set, whose candidates to be processed are the set of sample matrices. Each sample matrix is decomposed into the product of K factor matrices, where the public $K-1$ factor matrices represent the learnt features which generalize the feature matrix in NMF to a feature matrix set, and the remaining factor matrix varying from individual sample matrix describes the activation patterns which generalizes the coefficient vector in NMF to a coefficient matrix. As such, Li and Zhang established Bilinear Form-based NMSF algorithm (BFBNMSF) [129] to solve the optimization problem. NMSF has so far not been fathomed fully.

NTF and NMSF have developed a generalized non-negative factorization framework, upon which the previous Basic, Constrained, and Structured NMF algorithms are able to be extended appropriately. Besides, NMSF only concentrates on the 3-dimensional situation, where it is more flexible and thorough than non-negative 3-dimensional tensor factorization, while NTF involves much broader cases.

6.4 Kernel NMF

Essentially, NMF and its variants mentioned above are linear models, which are unable to extract nonlinear structures and relationships hidden in the data. This restricts the scope of application of NMF. To overcome these limitations, a nature extension is to apply kernel-based methods by mapping input data into an implicit feature space using nonlinear functions, just as kernel PCA, kernel LDA, and kernel ICA. Besides, it is potential in processing data with negative values by using some specific kernel functions and allowing high-order dependencies between basis vectors [130], [131].

Given a nonlinear mapping $\phi : \mathbb{R}^M \to \Re, \boldsymbol{x} \mapsto \phi(\boldsymbol{x})$, which maps the input data space $\mathbb{R}^M$ into the feature space $\Re$, the original data matrix is transformed into $\boldsymbol{X} \to \boldsymbol{Y} = \phi(\boldsymbol{X}) = [\phi(\boldsymbol{x}_1), \phi(\boldsymbol{x}_2), \ldots, \phi(\boldsymbol{x}_N)]$. Kernel NMF seeks to find factor matrices $\boldsymbol{Z} = [\phi(\boldsymbol{u}_1), \phi(\boldsymbol{u}_2), ..., \phi(\boldsymbol{u}_L)]$ and $\boldsymbol{V}$, such that $\boldsymbol{Y} \approx \boldsymbol{ZV}$, where $\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_L$ are basis vectors in the original space. To avoid expressing explicitly the nonlinear mapping, SED is chosen as the object function:

$$\begin{aligned} &D_{KF}(\boldsymbol{Y} \,\|\boldsymbol{UV}) = \frac{1}{2}\|\boldsymbol{Y} - \boldsymbol{ZV}\|_F^2 \\ &= \frac{1}{2}tr(\boldsymbol{Y}^T\boldsymbol{Y}) - tr(\boldsymbol{Y}^T\boldsymbol{ZV}) + \frac{1}{2}tr(\boldsymbol{V}^T\boldsymbol{Z}^T\boldsymbol{ZV}). \end{aligned} \tag{17}$$

Using kernel function $k(x,y) = \langle\phi(x), \phi(y)\rangle$ where $\langle\cdot,\cdot\rangle$ denotes the inner product and kernel matrices $\boldsymbol{K}^{xx}$, $\boldsymbol{K}^{uu}$, and $\boldsymbol{K}^{xu}$, where $\mathbf{K}_{ij}^{xx} = \langle\phi(x_i), \phi(x_j)\rangle$, $\mathbf{K}_{ij}^{uu} = \langle\phi(u_i), \phi(u_j)\rangle$, and $\mathbf{K}_{ij}^{xu} = \langle\phi(x_i), \phi(u_j)\rangle$, (17) can be written as

$$\begin{aligned} &D_{KF}(\boldsymbol{Y} \,\|\boldsymbol{UV}) \\ &= \frac{1}{2}tr(\boldsymbol{K}^{xx}) - tr(\boldsymbol{K}^{xu}\boldsymbol{V}) + \frac{1}{2}tr(\boldsymbol{V}^T\boldsymbol{K}^{uu}\boldsymbol{V}). \end{aligned} \tag{18}$$

Thus the model depends only on the kernel matrices.

Buciu et al. proposed the above kernel NMF model in polynomial feature space, and adopted modified multiplicative rules as the update algorithm [131]. This is only applicable for polynomial kernels. By resorting to the PG method, it is generalized to any kernel function [132]. However, the non-negativity of bases in kernel feature space is not warranted by these two methods. To handle this, a Mercer kernel is constructed to preserve the non-negativity on both bases and coefficients in kernel feature space [133].

An alternative direction is to choose an appropriate NMF model firstly, and then implement the kernel generation. In the case of kernel convex NMF $\phi(\boldsymbol{X}) \approx \phi(\boldsymbol{X})\boldsymbol{GV}$ [67], the corresponding cost function is

$$\begin{aligned} &\|\phi(\boldsymbol{X}) - \phi(\boldsymbol{X})\boldsymbol{GV}\|_F^2 \\ &= tr(\boldsymbol{K}) - 2tr(\boldsymbol{KGV}) + tr(\boldsymbol{V}^T\boldsymbol{G}^T\boldsymbol{KGV}) \end{aligned} \tag{19}$$

which only depends on the kernel matrix $\boldsymbol{K} = \phi^T(X)\phi(X)$. The solution to this problem is much easier than the previous one that directly obtains basis vectors, and thereby reduces the computational consumption [132]. It will also incorporate the merit of convex NMF and enhance the sparsity. Besides, SVM Discriminant NMF [19] is advisable.

However, the current kernel NMF results are just preliminary with objective function dependent, so a systematic construction and evaluation framework for kernel NMF needs developing in the future.

7 Conclusion

In short, as a multivariate data analysis and dimensionality reduction technique, NMF enhances compression and interpretability due to its parts-based and sparse representation from the non-negativity or

purely additive constraint. It outperforms classic low-rank approximation approaches such as PCA in some cases, and makes the post-processing much easier.

Although there have been some notable results on NMF, they are far to be perfect with lots of open questions remained to be solved. Here we just list a few possibilities as follows:

(1) Statistical underpinning. Although NMF can be interpreted as the maximum likelihood algorithm in different residual distribution, a solid statistical underpinning has not been developed adequately by now, which is an essential yet neglected, to some extent, issue.

(2) Complete NMF. Compared with its approximate counterpart, there are currently not abundant theoretical results about complete NMF from the matrix factorization perspective, such as non-negative rank and complexity. It is indeed hard; anyhow, it will provide some worthy suggestions for approximate NMF. Points (1) and (2) can be viewed as two complementary directions from statistical and algebraic standpoints, respectively.

(3) Posedness or uniqueness. Many problems in NMF can be traced back to the ill-posedness of NMF. There have been some discussions, but this has not been resolved satisfactorily. What Constrained NMF has done is to reduce the degrees of freedom and enhance the uniqueness of the solution by imposing various additional constraints. In another viewpoint, this reflects the relationship and difference between optimizing the objective functions and identifying the true feature structures. Some concerns from the geometric perspective providing heuristic attempt might be able to solve this problem to some extent by combining the objective function optimization with the intuitive traceability.

(4) Formulations. The formulation of a problem directly determines the solution to it. The majority of NMF and its variants are based on the SED objective function, which basically sets the keynote of the whole NMF framework. It claims attention that the notion of optimizing the objective functions is not obviously and necessarily equivalent to that of identifying the underlying actual components of the data set, which is the ultimate destination of NMF. So how to select and evaluate different objective functions, and even try to formulate new paradigms of NMF, need further consideration. The discussion in Section 3.4, although discrete, might show the possible way.

(5) Global optimal solution. Most existing algorithms only obtain the local optimal solutions, so some global optimization techniques can be introduced. This is highly pertinent to the problems discussed in (3) and (4).

(6) Determination of the number L of the basis vectors. There has been no systematic mechanism for the choice of L, which brings inconvenience for practical applications. Bounds for the non-negative rank or incorporating some manifold learning techniques could help select L.

(7) Initialization. Proper seeding mechanism of factor matrices both accelerates the convergence and enhances solution quality, since NMF is sensitive to the initial values.

(8) Pragmatic issue. In order to make possible the practical usage of NMF on large-scale data set, more efficient, highly scalable, more effectively incremental NMF algorithms need further studying.

(9) Generalized NMF. Reasonable variations on NMF will extend the applicable range of NMF methods. Compared with the other three, Generalized NMF is the most premature area. There is a lot of work worthy to be done. Besides, the corresponding basic, constrained, and structured models as well as algorithms can be constructed upon the generalized framework similarly.

Finally, the research in NMF has led to some modifications on canonical decomposition methods, such as non-negative PCA [134] and non-negative ICA [135]. This illustrates that we can either incorporate other constraints into NMF model, or introduce the non-negativity constraint in the existing decomposition framework. In one word, they are the explorations for the same issue from different perspectives.

ACKNOWLEDGMENTS

We would like to thank Dr. Le Li for his comments. This work was supported by National Natural Science Foundation of China under Grant 61171118.

REFERENCES

[1] P. Paatero and U. Tapper, "Positive Matrix Factorization: a Non-negative Factor Model with Optimal Utilization of Error Estimates of Data Values," *Environmetrics*, vol. 5, no. 2, pp. 111–126, 1994.

[2] P. Paatero, "Least Squares Formulation of Robust Non-negative Factor Analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 37, no. 1, pp. 23–35, 1997.

[3] D. Lee and H. Seung, "Learning the Parts of Objects by Non-negative Matrix Factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[4] ——, "Algorithms for Non-negative Matrix Factorization," in *Proc. Advances in neural information processing systems 13*, 2001, pp. 556–562.

[5] "Non-negative Matrix Factorization," *Wikipedia*, http://en.wikipedia.org/wiki/Non-negative_matrix_factorization.

[6] S. Ullman, *High-Level Vision: Object Recognition and Visual Cognition*. Cambridge: MIT Press, 2000.

[7] W. Xu, X. Liu, and Y. Gong, "Document clustering based on non-negative matrix factorization," in *Proc. Int. ACM SIGIR Conf. Research and development in informaion retrieval*, 2003, pp. 267–273.

[8] D. Field, "What is the goal of sensory coding?" *Neural computation*, vol. 6, no. 4, pp. 559–601, 1994.

[9] A. Cichocki, R. Zdunek, A. Phan, and S. Amari, *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation*. West Sussex, United Kingdom: John Wiley & Sons, 2009.

[10] M. Berry, M. Browne, A. Langville, V. Pauca, and R. Plemmons, "Algorithms and Applications for Approximate Non-negative Matrix Factorization," *Computational Statistics & Data Analysis*, vol. 52, no. 1, pp. 155–173, 2007.

[11] I. Buciu, "Non-negative Matrix Factorization, a New Tool for Feature Extraction: Theory and Applications," *Int. J. Computers, Communications and Control*, vol. 3, Suppl. S, pp. 67–74, 2008.

[12] M. Chu and R. Plemmons, "Nonnegative Matrix Factorization and Applications," *Bulletin of the International Linear Algebra Society*, vol. 34, pp. 2–7, 2005.

[13] S. Sra and I. Dhillon, "Nonnegative Matrix Approximation: Algorithms and Applications," University of Texas at Austin, Tech. Rep., 2006, http://www.cs.utexas.edu/ftp/ftp/pub/techreports/tr06-27.pdf.

[14] I. Ulbrich, M. Canagaratna, Q. Zhang, D. Worsnop, and J. Jimenez, "Interpretation of organic components from positive matrix factorization of aerosol mass spectrometric data," *Atmospheric Chemistry and Physics*, vol. 9, pp. 2891–2918, 2009.

[15] K. Drakakis, S. Rickard, R. de Fréin, and A. Cichocki, "Analysis of financial data using non-negative matrix factorization," vol. 3, no. 38, pp. 1853–1870, 2008.

[16] C. Ding, T. Li, W. Peng, and H. Park, "Orthogonal Nonnegative Matrix Tri-Factorizations for Clustering," in *Proc.12th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD)*, 2006, pp. 126–135.

[17] A. Cichocki, H. Lee, Y. Kim, and S. Choi, "Non-negative Matrix Factorization with α-Divergence," *Pattern Recognition Letters*, vol. 29, no. 9, pp. 1433–1440, 2008.

[18] R. Sandler and M. Lindenbaum, "Nonnegative matrix factorization with earth mover's distance metric for image analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2011, to be published. Digital Object Identifier: 10.1109/TPAMI.2011.18.

[19] M. Gupta and J. Xiao, "Non-Negative Matrix Factorization as a Feature Selection Tool for Maximum Margin Classifiers," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2011.

[20] S. Zafeiriou, A. Tefas, I. Buciu, and I. Pitas, "Exploiting Discriminant Information in Nonnegative Matrix Factorization with Application to Frontal Face Verification," *IEEE Trans. Neural Network*, vol. 17, no. 3, pp. 683–695, 2006.

[21] I. Kotsia, S. Zafeiriou, and I. Pitas, "A Novel Discriminant Non-negative Matrix Factorization Algorithm with Applications to Facial Image Characterization Problems," *IEEE Trans. Infor. Forensics & Security*, vol. 2, no. 3, pp. 588–595, 2007.

[22] W. Wang, "Squared Euclidean Distance Based Convolutive Non-negative Matrix Factorization with Multiplicative Learning Rules for Audio Pattern Separation," in *Proc. IEEE Int. Symposium on for Audio Pattern Separation Signal Processing and Information Technology*, 2007, pp. 347–352.

[23] E. Benetos and C. Kotropoulos, "Non-negative Tensor Factorization Applied to Music Genre Classification," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 18, no. 8, pp. 1955–1967, 2010.

[24] R. Zdunek and A. Cichocki, "Non-negative Matrix Factorization with Quasi-Newton Optimization," in *Proc. 8th Int. Conf. Artificial Intelligence and Soft Computing*, 2006, pp. 870–879.

[25] M. Van Benthem and M. Keenan, "Fast Algorithm for the Solution of Large-Scale Non-negativity-Constrained Least Squares Problems," *J. chemometrics*, vol. 18, no. 10, pp. 441–450, 2004.

[26] H. Kim and H. Park, "Sparse Non-negative Matrix Factorizations via Alternating Non-negativity-Constrained Least Squares for Microarray Data Analysis," *Bioinformatics*, vol. 23, no. 12, p. 1495, 2007.

[27] J. Tropp, "Literature Survey: Nonnegative Matrix Factorization," Institute for Computational Engineering and Sciences, University of Texas at Asutin, Tech. Rep., 2003, http://www.acm.caltech.edu/~jtropp/notes/Tro03LiteratureSurvey.pdf.

[28] M. Chu, F. Diele, R. Plemmons, and S. Ragni, "Optimality, Computation, and Interpretation of Nonnegative Matrix Factorizations," *SIAM J. Matrix Anal. and Appl.*, pp. 4–21, 2004.

[29] L. Li and Y.-J. Zhang, "A Survey on Algorithms of Non-negative Matrix Factorization," *Chinese J. Electronics*, vol. 36, no. 4, pp. 737–743, 2008.

[30] M. Gupta, "Additive Non-negative Matrix Factorization for Missing Data," *Arxiv preprint arXiv:1007.0380*, 2010.

[31] N. Vasiloglou, A. Gray, and D. Anderson, "Non-negative Matrix Factorization, Convexity and Isometry," in *Proc. SIAM Data Mining Conf.*, 2009, pp. 673–684.

[32] D. Donoho and V. Stodden, "When does Non-negative Matrix Factorization Give a Correct Decomposition into Parts?" in *Proc. Advances in neural information processing systems 16*, 2004, pp. 1141–1148.

[33] B. Dong, M. Lin, and M. Chu, "Nonnegative Rank Factorization via Rank Reduction," *preprint*, 2008, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.157.8036&rep=rep1&type=pdf.

[34] S. Vavasis, "On the Complexity of Nonnegative Matrix Factorization," *SIAM J. Opt.*, vol. 20, no. 3, pp. 1364–1377, 2009.

[35] N. Gillis and F. Glineur, "Nonnegative Factorization and the Maximum Edge Biclique Problem," *Arxiv preprint arXiv:0810.4225*, 2008.

[36] L. Thomas, "Solution to Problem 73-14: Rank Factorization of Nonnegative Matrices by A. Berman and R. J. Plemmons," *SIAM Review*, vol. 16, pp. 393–394, 1974.

[37] N. Gillis, "Nonnegative Matrix Factorization: Complexity, Algorithms and Applications," Ph.D. dissertation, Dept. Mathematical Engineering, Université catholique de Louvain, Belgium, Feb. 2011.

[38] B. Klingenberg, J. Curry, and A. Dougherty, "Non-negative Matrix Factorization: Ill-Posedness and a Geometric Algorithm," *Pattern Recognition*, vol. 42, no. 5, pp. 918–928, 2009.

[39] S. Moussaoui, D. Brie, and J. Idier, "Non-negative Source Separation: Range of Admissible Solutions and Conditions for the Uniqueness of the Solution," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, vol. 5, 2005, pp. 289–292.

[40] H. Laurberg, M. Christensen, M. Plumbley, L. Hansen, and S. Jensen, "Theorems on Positive Data: On the Uniqueness of NMF," *Computational Intelligence and Neuroscience*, 2008.

[41] F. Theis, K. Stadlthanner, and T. Tanaka, "First Results on Uniqueness of Sparse Non-negative Matrix Factorization," in *Proc. 13th European Signal Processing Conf. (EUSIPCO05)*, 2005.

[42] S. Rickard and A. Cichocki, "When is Non-negative Matrix Decomposition Unique?" in *Proc. 42nd Annual Conference on Information Sciences and Systems (CISS)*, 2008, pp. 1091–1092.

[43] S. Li, X. Hou, H. Zhang, and Q. Cheng, "Learning Spatially Localized, Parts-based Representation," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2001, pp. 207–212.

[44] C. Ding, X. He, and H. Simon, "On the Equivalence of Nonnegative Matrix Factorization and Spectral Clustering," in *Proc. SIAM Data Mining Conf*, 2005, pp. 606–610.

[45] T. Li and C. Ding, "The Relationships among Various Nonnegative Matrix Factorization Methods for Clustering," in *Proc. 6th IEEE Int. Conf. Data Mining*, 2006, pp. 362–371.

[46] E. Gaussier and C. Goutte, "Relation between PLSA and NMF and Implications," in *Proc. 28th ACM Conf. Research and Development in Information Retrieval*, 2005, pp. 601–602.

[47] C. Ding, T. Li, and W. Peng, "Nonnegative Matrix Factorization and Probabilistic Latent Semantic Indexing: Equivalence Chi-square Statistic, and a Hybrid Method," in *Proc. 21st Nat'l Conf. Artificial Intelligence (AAAI)*, 2006, pp. 342–347.

[48] Z. Yang, H. Zhang, Z. Yuan, and E. Oja, "Kullback-Leibler Divergence for Nonnegative Matrix Factorization," *Artificial Neural Networks and Machine Learning*, vol. 6791, pp. 250–257, 2011.

[49] R. Kompass, "A Generalized Divergence Measure for Nonnegative Matrix Factorization," *Neural computation*, vol. 19, no. 3, pp. 780–791, 2007.

[50] A. Cichocki and S. Amari, "Families of alpha-beta-and gamma-divergences: Flexible and robust measures of similarities," *Entropy*, vol. 12, no. 6, pp. 1532–1568, 2010.

[51] A. Cichocki, R. Zdunek, and S. Amari, "Csiszárs Divergences for Non-negative Matrix Factorization: Family of New Algorithms," *Lecture Notes in Computer Science*, pp. 32–39, 2006.

[52] I. Dhillon and S. Sra, "Generalized Nonnegative Matrix Approximations with Bregman Divergences," in *Proc. Advances in neural information processing systems 18*, 2006, pp. 283–290.

[53] A. Cichocki, S. Cruces, and S. Amari, "Generalized alpha-beta divergences and their application to robust nonnegative matrix factorization," *Entropy*, vol. 13, no. 1, pp. 134–170, 2011.

[54] A. Hamza and D. Brady, "Reconstruction of reflectance spectra using robust nonnegative matrix factorization," *IEEE trans. Signal Processing*, vol. 54, no. 9, pp. 3637–3642, 2006.

[55] E. Gonzalez and Y. Zhang, "Accelerating the Lee-Seung Algorithm for Non-negative Matrix Factorization," Department of Computational and Applied Mathematics, Rice University, Tech. Rep., 2005, http://www.caam.rice.edu/tech_reports/2005/TR05-02.ps.

[56] C.-J. Lin, "Projected Gradient Methods for Nonnegative Matrix Factorization," *Neural Computation*, vol. 19, no. 10, pp. 2756–2779, 2007.

[57] ——, "On the Convergence of Multiplicative Update Algorithms for Nonnegative Matrix Factorization," *IEEE Trans. Neural Networks*, vol. 18, no. 6, pp. 1589–1596, 2007.

[58] M. Merritt and Y. Zhang, "An Interior-Point Gradient Method for Large-Scale Totally Nonnegative Least Squares Problems," *J. Opt. Theo. & Appl.*, vol. 126, no. 1, pp. 191–202, 2005.

[59] R. Zdunek and A. Cichocki, "Nonnegative Matrix Factorization with Constrained Second-Order Optimization," *Signal Processing*, vol. 87, no. 8, pp. 1904–1916, 2007.

[60] A. Cichocki and R. Zdunek, "Multilayer nonnegative matrix factorization using projected gradient approaches," *Int. J. Neural Systems*, vol. 17, no. 6, pp. 431–446, 2007.

[61] R. Zdunek and A. Cichocki, "Fast Nonnegative Matrix Factorization Algorithms Using Projected Gradient Approaches for Large-Scale Problems," *Computational intelligence and neuroscience*, 2008.

[62] H. Kim and H. Park, "Non-negative Matrix Factorization based on Alternating Non-negativity Constrained Least Squares and Active Set Method," *SIAM J. Matrix Anal. and Appl.*, vol. 30, no. 2, pp. 713–730, 2008.

[63] L. Li and Y.-J. Zhang, "FastNMF: a Fast Monotonic Fixed-Point Non-negative Matrix Factorization Algorithm with High Ease of Use," in *Proc.19th Int. Conf. Pattern Recognition*, 2008, pp. 1–4.

[64] ——, "FastNMF: Highly Efficient Monotonic Fixed-Point Nonnegative Matrix Factorization Algorithm with Good Applicability," *J. Electronic Imaging*, vol. 18, no. 3, p. 033004, 2009.

[65] A. Cichocki and A. Phan, "Fast Local Algorithms for Large Scale Nonnegative Matrix and Tensor Factorizations," *IEICE Trans. Fundamentals of Electronics*, vol. 92, pp. 708–721, 2008.

[66] D. Lee and H. Seung, "Unsupervised Learning by Convex and Conic Coding," in *Proc. Advances in Neural Information Processing Systems 9*, 1997, pp. 515–521.

[67] C. Ding, T. Li, and M. Jordan, "Convex and Semi-nonnegative Matrix Factorizations," *IEEE Trans. Pattern Analysis and Machine Intelligence*, no. 1, pp. 45–55, 2010.

[68] W. Xu and Y. Gong, "Document Clustering by Concept Factorization," in *Proc. 27th annual Int. Conf. Research and development in information retrieval*, 2004, pp. 202–209.

[69] L. Li and Y.-J. Zhang, "Linear Projection-Based Non-negative Matrix Factorization," *Acta Automatica Sinica*, vol. 36, no. 1, pp. 23–39, 2010.

[70] Z. Yuan and E. Oja, "Projective Nonnegative Matrix Factorization for Image Compression and Feature Extraction," in *Proc. 14th Scandinavian Conf. Image Anal.*, 2005, pp. 333–342.

[71] Z. Yang and E. Oja, "Linear and Nonlinear Projective Nonnegative Matrix Factorization," *IEEE Trans. Neural Networks*, vol. 21, no. 5, pp. 734–749, 2010.

[72] N. Gillis and F. Glineur, "Using Underapproximations for Sparse Nonnegative Matrix Factorization," *Pattern Recognition*, vol. 43, no. 4, pp. 1676–1687, 2010.

[73] A. Cichocki, R. Zdunek, and S. Amari, "Nonnegative Matrix and Tensor Factorization," *IEEE Signal Processing Magazine*, vol. 25, no. 1, pp. 142–145, 2008.

[74] C. Liu, H. Yang, J. Fan, L. He, and Y. Wang, "Distributed nonnegative matrix factorization for web-scale dyadic data analysis on mapreduce," in *Proc. ACM Int. Conf. World Wide Web*, 2010, pp. 681–690.

[75] S. Bucak and B. Gunsel, "Incremental Subspace Learning via Non-negative Matrix Factorization," *Pattern recognition*, vol. 42, no. 5, pp. 788–797, 2009.

[76] ——, "Video Content Representation by Incremental Non-negative Matrix Factorization," in *Proc. IEEE Int. Conf. Image Processing (ICIP)*, vol. 2, 2007, pp. 113–116.

[77] W. Chen, B. Pan, B. Fang, and J. Zou, "A Novel Constraint Non-negative Matrix Factorization Criterion Based Incremental Learning in Face Recognition," in *Int. Conf. Wavelet Anal. & Pattern Recog.*, vol. 1, 2008, pp. 292–297.

[78] B. Cao, D. Shen, J. Sun, X. Wang, Q. Yang, and Z. Chen, "Detect and Track Latent Factors with Online Nonnegative Matrix Factorization," in *Proc. 20th Int. Joint Conf. Artificial Intelligence (IJCAI)*, 2007, pp. 2689–2694.

[79] S. Wild, J. Curry, and A. Dougherty, "Improving Non-negative Matrix Factorizations Through Structured Initialization," *Pattern Recognition*, vol. 37, no. 11, pp. 2217–2232, 2004.

[80] C. Boutsidis and E. Gallopoulos, "SVD based Initialization: A Head Start for Nonnegative Matrix Factorization," *Pattern Recognition*, vol. 41, no. 4, pp. 1350–1362, 2008.

[81] Y. Xue, C. Tong, Y. Chen, and W. Chen, "Clustering-based Initialization for Non-negative Matrix Factorization," *Applied Mathe. & Compu.*, vol. 205, no. 2, pp. 525–536, 2008.

[82] Z. Zheng, J. Yang, and Y. Zhu, "Initialization Enhancer for Non-negative Matrix Factorization," *Engineering Applications of Artificial Intelligence*, vol. 20, no. 1, pp. 101–110, 2007.

[83] A. Janecek and Y. Tan, "Using Population Based Algorithms for Initializing Nonnegative Matrix Factorization," *Advances in Swarm Intelligence*, vol. 6729, pp. 307–316, 2011.

[84] A. Langville, C. Meyer, R. Albright, J. Cox, and D. Duling, "Initializations for the Nonnegative Matrix Factorization," in *Proc.12th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, 2006.

[85] A. Cichocki and R. Zdunek, "Multilayer Nonnegative Matrix Factorization," *Elec. Lett.*, vol. 42, no. 16, pp. 947–948, 2006.

[86] N. Mohammadiha and A. Leijon, "Nonnegative Matrix Factorization Using Projected Gradient Algorithms with Sparseness Constraints," in *Proc. IEEE Int. Symposium on Signal Processing and Information Technology*, 2009, pp. 418–423.

[87] P. Hoyer, "Non-negative Sparse Coding," in *Proc. IEEE Workshop on Neur. Networks for Signal Pro.*, 2002, pp. 557–565.

[88] ——, "Non-negative Matrix Factorization with Sparseness Constraints," *J. Machine Learning Research*, vol. 5, no. 9, pp. 1457–1469, 2004.

[89] M. Mørup, K. Madsen, and L. Hansen, "Approximate L0 Constrained Non-negative Matrix and Tensor Factorization," in *Proc. IEEE International Symposium on Circuits and Systems*, 2008, pp. 1328–1331.

[90] L. Li and Y.-J. Zhang, "SENSC: A Stable and Efficient Algorithm for Non-negative Sparse Coding," *Acta Automatica Sinica*, vol. 35, no. 10, pp. 1257–1271, 2009.

[91] Y. Gao and G. Church, "Improving Molecular Cancer Class Discovery Through Sparse Non-negative Matrix Factorization," *Bioinformatics*, vol. 21, no. 21, pp. 3970–3975, 2005.

[92] R. Tibshirani, "Regression Shrinkage and Selection via the Lasso," *J. Royal Statistical Society. Series B*, vol. 58, no. 1, pp. 267–288, 1996.

[93] H. Laurberg and L. Hansen, "On Affine Non-negative Matrix Factorization," in *Proc. IEEE Int. Conf. Acoustics Speech and Signal Processing*, vol. 2, 2007, pp. 653–656.

[94] Z. Li, X. Wu, and H. Peng, "Nonnegative Matrix Factorization on Orthogonal Subspace," *Pattern Recognition Letters*, vol. 31, no. 9, pp. 905–911, 2010.

[95] S. Choi, "Algorithms for Orthogonal Nonnegative Matrix Factorization," in *Proc. IEEE Int. Joint Conf. Neural Networks (IJCNN)*, 2008, pp. 1828–1832.

[96] J. Yoo and S. Choi, "Orthogonal Nonnegative Matrix Tri-Factorization for Co-Clustering: Multiplicative Updates on Stiefel Manifolds," *Information Processing & Management*, vol. 46, no. 5, pp. 559–570, 2010.

[97] Y. Wang, Y. Jia, C. Hu, and M. Turk, "Fisher Non-negative Matrix Factorization for Learning Local Features," in *Proc. Asian Conf. Comp Vision*, 2004, pp. 27–30.

[98] D. Cai, X. He, J. Han, and T. Huang, "Graph Regularized Non-negative Matrix Factorization for Data Representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1548–1560, 2011.

[99] D. Cai, X. He, X. Wu, and J. Han, "Non-negative Matrix Factorization on Manifold," in *Proc. 8th IEEE Int. Conf. Data Mining (ICDM)*, 2008, pp. 63–72.

[100] R. Zhi, M. Flierl, Q. Ruan, and W. Kleijn, "Facial Expression Recognition based on Graph-Preserving Sparse Non-negative Matrix Factorization," in *Proc. 16th IEEE Int. Conf. Image Processing (ICIP)*, 2009, pp. 3293–3296.

[101] ——, "Graph-Preserving Sparse Nonnegative Matrix Factorization with Application to Facial Expression Recognition,"

IEEE Trans. Systems, Man, and Cybernetics, Part B: Cybernetics, vol. 41, no. 1, pp. 38–52, 2011.

[102] T. Zhang, B. Fang, Y. Tang, G. He, and J. Wen, "Topology Preserving Non-negative Matrix Factorization for Face Recognition," *IEEE Trans. Image Processing*, vol. 17, no. 4, pp. 574–584, 2008.

[103] Q. Gu and J. Zhou, "Neighborhood Preserving Nonnegative Matrix Factorization," in *Proc. 20th British Machine Vision Conference*, 2009.

[104] B. Shen and L. Si, "Nonnegative Matrix Factorization Clustering on Multiple Manifolds," in *Proc. 24th AAAI Conf. Artificial Intelligence (AAAI)*, 2010, pp. 575–580.

[105] R. Zhi and Q. Ruan, "Discriminant Sparse Nonnegative Matrix Factorization," in *Proc. IEEE Int. Conf. Multimedia and Expo (ICME)*, 2009, pp. 570–573.

[106] S. An, J. Yoo, and S. Choi, "Manifold-Respecting Discriminant Nonnegative Matrix Factorization," *Pattern Recognition Letters*, vol. 32, no. 6, pp. 832–837, 2011.

[107] N. Guan, D. Tao, Z. Luo, and B. Yuan, "Manifold Regularized Discriminative Non-negative Matrix Factorization with Fast Gradient Descent," *IEEE trans. Image Processing*, vol. 20, no. 2030–2048, 2011.

[108] Y. Mao and L. Saul, "Modeling Distances in Large-Scale Networks by Matrix Factorization," in *Proc. 4th ACM SIGCOMM Conf. Internet Measurement*, 2004, pp. 278–287.

[109] S. Zhang, W. Wang, J. Ford, and F. Makedon, "Learning from Incomplete Ratings using Non-negative Matrix Factorization," in *Proc.6th SIAM Int. Conf. Data Mining (SDM)*, 2006, pp. 549–553.

[110] Y. Kim and S. Choi, "Weighted Nonnegative Matrix Factorization," in *AProc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, 2009, pp. 1541–1544.

[111] P. Smaragdis, "Non-negative Matrix Factor Deconvolution; Extraction of Multiple Sound Sources from Monophonic Inputs," in *Proc. 5th Int. Conf. Independent Component Analysis and Blind Signal Separation*, 2004, pp. 494–499.

[112] ——, "Convolutive Speech Bases and Their Application to Supervised Speech Separation," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 15, no. 1, pp. 1–12, 2007.

[113] P. O'Grady and B. Pearlmutter, "Convolutive Non-negative Matrix Factorisation with a Sparseness Constraint," in *Proc. 16th IEEE Signal Processing Society Workshop on Machine Learning for Signal Processing*, 2006, pp. 427–432.

[114] V. Potluru, S. Plis, and V. Calhoun, "Sparse Shift-Invariant NMF," in *Proc. IEEE Southwest Symposium on Image Analysis and Interpretation (SSIAI)*, 2008, pp. 69–72.

[115] A. Pascual-Montano, J. Carazo, K. Kochi, D. Lehmann, and R. Pascual-Marqui, "Nonsmooth Nonnegative Matrix Factorization (nsNMF)," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 3, pp. 403–415, 2006.

[116] M. Welling and M. Weber, "Positive Tensor Factorization," *Pattern Recognition Letters*, vol. 22, no. 12, pp. 1255–1261, 2001.

[117] T. Hazan, S. Polak, and A. Shashua, "Sparse Image Coding Using a 3D Non-negative Tensor Factorization," in *Proc. 10th IEEE Int. Conf. Computer Vision (ICCV)*, vol. 1, 2005, pp. 50–57.

[118] A. Shashua and T. Hazan, "Non-negative Tensor Factorization with Applications to Statistics and Computer vision," in *Proc. 22nd Int. Conf. Machine Learning*, 2005, pp. 792–799.

[119] M. Mørup, L. Hansen, and S. Arnfred, "Algorithms for Sparse Nonnegative Tucker Decompositions," *Neural computation*, vol. 20, no. 8, pp. 2112–2131, 2008.

[120] A. Cichocki, R. Zdunek, S. Choi, R. Plemmons, and S. Amari, "Non-negative Tensor Factorization Using α and β Divergences," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, vol. 3, 2007, pp. 1393–1396.

[121] S. Zafeiriou and M. Petrou, "Nonnegative tensor factorization as an alternative csiszar–tusnady procedure: Algorithms, convergence, probabilistic interpretations and novel probabilistic tensor latent variable analysis algorithms," *Data Mining and Knowledge Discovery*, vol. 22, no. 3, pp. 419–466, 2011.

[122] S. Zafeiriou, "Algorithms for nonnegative tensor factorization," *Tensors in Image Processing and Computer Vision*, pp. 105–124, 2009.

[123] M. Heiler and C. Schnörr, "Controlling Sparseness in Non-negative Tensor Factorization," in *Proc. 9th European Conf. Computer Vision (ECCV)*, Graz, Austria, 2006, pp. 56–67.

[124] A. Phan, P. Tichavsk, and A. Cichocki, "Fast damped gauss-newton algorithm for sparse and nonnegative tensor factorization," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, 2011, pp. 1988–1991.

[125] J. Liu, J. Liu, P. Wonka, and J. Ye, "Sparse non-negative tensor factorization using columnwise coordinate descent," *Pattern Recognition*, vol. 45, no. 1, pp. 649–656, 2012.

[126] S. Zafeiriou, "Discriminant Nonnegative Tensor Factorization Algorithms," *IEEE Trans. Neural networks*, vol. 20, no. 2, pp. 217–235, 2009.

[127] C. Wang, X. He, J. Bu, Z. Chen, C. Chen, and Z. Guan, "Image representation using laplacian regularized nonnegative tensor factorization," *Pattern Recognition*, vol. 44, no. 10-11, pp. 2516–2526, 2011.

[128] L. Li and Y.-J. Zhang, "Non-negative Matrix-Set Factorization," *Chinese J. Electronics and Information Technology*, vol. 31, no. 2, pp. 255–260, 2009.

[129] ——, "Bilinear Form-Based Non-Negative Matrix Set Factorization," *Chinese J. Computers*, vol. 32, no. 8, pp. 1536–1549, 2009.

[130] D. Zhang, Z. Zhou, and S. Chen, "Non-negative Matrix Factorization on Kernels," in *Proc. 9th Pacific Rim Int. Conf. Artificial Intelligence (PRICAI)*, vol. 4099, 2006, pp. 404–412.

[131] I. Buciu, N. Nikolaidis, and I. Pitas, "Nonnegative Matrix Factorization in Polynomial Feature Space," *IEEE Trans. Neural networks*, vol. 19, no. 6, pp. 1090–1100, 2008.

[132] Z. Liang, Y. Li, and T. Zhao, "Projected Gradient Method for Kernel Discriminant Nonnegative Matrix Factorization and the Applications," *Signal Processing*, vol. 90, no. 7, pp. 2150–2163, 2010.

[133] B. Pan, J. Lai, and W. Chen, "Nonlinear Nonnegative Matrix Factorization Based on Mercer Kernel Construction," *Pattern Recognition*, vol. 44, no. 10, pp. 2800–2810, 2011.

[134] M. Plumbley and E. Oja, "A nonnegative PCA Algorithm for Independent Component Analysis," *IEEE Trans. Neural networks*, vol. 15, no. 1, pp. 66–76, 2004.

[135] M. Plumbley, "Algorithms for Nonnegative Independent Component Analysis," *IEEE Trans. Neural networks*, vol. 14, no. 3, pp. 534–543, 2003.

Yu-Xiong Wang (M'12) received the B.E. degree in Electronic Engineering from Beijing Institute of Technology (BIT), Beijing, China, in 2009. Currently, he is a master's student in the Department of Electronic Engineering at Tsinghua University, Beijing, China. His research interests include image processing, computer vision, and machine learning.

Yu-Jin Zhang (M'98-SM'99) received the Ph.D. degree in Applied Science from Montefiore Institute at the State University of Liège, Liège,' Belgium, in 1989. He was post-doc fellow and research fellow with the Department of Applied Physics and Department of Electrical Engineering at the Delft University of Technology, Delft, the Netherlands from 1989 to 1993. In 1993, he joined the Department of Electronic Engineering at Tsinghua University, Beijing, China, where he has been a professor of Image Engineering since 1997. He has authored more than 20 books and published more than 400 papers in the areas of image processing, image analysis, and image understanding. Professor Zhang is director of academic committee of China Society of Image and Graphics, and is a Fellow of SPIE.