Front. Electr. Electron. Eng. China 2010, 5(3): 261–273
DOI 10.1007/s11460-010-0106-y

RESEARCH ARTICLE

**Erkki OJA, Zhirong YANG**

# Orthogonal nonnegative learning for sparse feature extraction and approximate combinatorial optimization



**Abstract** Nonnegativity has been shown to be a powerful principle in linear matrix decompositions, leading to sparse component matrices in feature analysis and data compression. The classical method is Lee and Seung's Nonnegative Matrix Factorization. A standard way to form learning rules is by multiplicative updates, maintaining nonnegativity. Here, a generic principle is presented for forming multiplicative update rules, which integrate an orthonormality constraint into nonnegative learning. The principle, called Orthogonal Nonnegative Learning (ONL), is rigorously derived from the Lagrangian technique. As examples, the proposed method is applied for transforming Nonnegative Matrix Factorization (NMF) and its variant, Projective Nonnegative Matrix Factorization (PNMF), into their orthogonal versions. In general, it is well-known that orthogonal nonnegative learning can give very useful approximative solutions for problems involving non-vectorial data, for example, binary solutions. Combinatorial optimization is replaced by continuous-space gradient optimization which is often computationally lighter. It is shown how the multiplicative updates rules obtained by using the proposed ONL principle can find a nonnegative and highly orthogonal matrix for an approximated graph partitioning problem. The empirical results on various graphs indicate that our nonnegative learning algorithms not only outperform those without the orthogonality condition, but also surpass other existing partitioning approaches.

**Keywords** nonnegative factorization, sparse feature extraction, orthogonal learning, clustering

Received March 5, 2010; accepted April 6, 2010

Erkki OJA (✉), Zhirong YANG
Department of Information and Computer Science, Aalto University, FI-00076 Aalto, Espoo, Finland
E-mail: {erkki.oja,zhirong.yang}@tkk.fi

## 1 Introduction

Enforcing nonnegativity in linear factorizations [1] has proven to be a powerful principle for multivariate data analysis, especially sparse feature analysis, as shown by the well-known *Nonnegative Matrix Factorization* (NMF) algorithm by Lee and Seung [2]. There now exists a large literature on NMF; see e.g., Cichocki et al. [3].

Recently, orthogonality as an additional constraint has been introduced in NMF [4]. This is motivated by the observations that orthogonality can significantly reduce the degrees of freedom and lead to much sparser factorizing matrices [4–6]. The primary reason for combining orthogonality and nonnegativity is that two nonnegative vectors are orthogonal if and only if their nonzero parts are non-overlapping. A typical example are binary vectors whose ones appear in different places. An orthogonal and nonnegative matrix is thus generally sparse with lots of zero entries, which may be advantageous in feature extraction. Also, it can sometimes be considered as the approximative substitute for the binary indicator matrix giving the solution of some discrete or combinatorial optimization problem. A typical example is clustering that has a close relation to NMF [7]. For such problems, orthogonal nonnegative learning can be advantageous compared to combinatorial optimization, as it provides a good approximation to the discrete solution while still operating in a continuous space and utilizing gradient descent as a low-complexity optimization tool.

The present authors have earlier introduced a version of NMF called *Projective NMF* (PNMF) [8] that is able to produce highly orthogonal and nonnegative basis matrices. In this paper we review some of this work, focussing on a variety of multiplicative rules for nonnegative learning which also accommodate the orthogonality constraint. The principle is mathematically justified by

using the Lagrangian approach. It is also related to the orthogonal gradient ascent *Principal Component Analysis* (PCA) principle introduced earlier by one of the authors [9].

We start in Sect. 2 by a brief review of PCA, clustering, NMF, and PNMF. We also give an example on sparse feature extraction. We then give the derivation and description of the multiplicative orthogonalization principle for nonnegative learning in Sect. 3 and show how it applies to NMF and some extensions. Then, in Sect. 4 we illustrate the benefits of orthogonal nonnegative learning as an approximate combinatorial optimization method through graph partitioning by trace maximization. These experiments show that orthogonal nonnegative learning algorithms can achieve much better objectives and partitioning accuracies than methods without the orthogonality constraint for a wide range of graph datasets. Our algorithms also defeat two existing partitioning approaches, Kernel $K$-means [10] and POD [11], especially when the derived similarity matrix is far from positive definiteness.

## 2 Linear generative models

Consider a data matrix $\mathbf{X} \in \mathbb{R}^{n\times m}$ whose $m$ columns (or $n$ rows) contain data vectors (signals, images, word histograms, etc.). Let us introduce a generative model

$$\mathbf{X} \approx \mathbf{WH} \tag{1}$$

with sources $\mathbf{H} \in \mathbb{R}^{r\times m}$ and weight matrix $\mathbf{W} \in \mathbb{R}^{n\times r}$. Typically, $r < m, n$ for compression and feature extraction, so an exact solution is not possible if $\mathbf{X}$ is full rank. Therefore, some approximation criterion like least mean squares has to be used. This model is graphically shown in Fig. 1.

The task is now to discover the "optimal" matrices $\mathbf{W}$ and $\mathbf{H}$ given only the observations $\mathbf{X}$.

Of course, the solution depends strongly on the optimality criterion. Even when that is given, we may note at once that this is a severely underdetermined problem: if $(\mathbf{W}^*, \mathbf{H}^*)$ is a solution, so is $\left(\mathbf{W}^*\mathbf{M}, \mathbf{M}^{-1}\mathbf{H}^*\right)$ for any invertible $\mathbf{M}$. Therefore, in addition to the optimality criterion, suitable constraints have to be defined to make the problem tractable.

In the following, let us consider some examples: principal component analysis, $K$-means clustering, nonnegative matrix factorization, and projective nonnegative matrix factorization.

### 2.1 Principal component analysis (PCA)

In PCA, each (centered or zero-mean) column of $\mathbf{X}$, say, $\mathbf{x}_j$, is represented as

$$\mathbf{x}_j \approx \sum_{i=1}^{r} \mathbf{w}_i \left(\mathbf{w}_i^{\mathrm{T}}\mathbf{x}_j\right),$$

where $\mathbf{w}_j$ are the eigenvectors of the data covariance matrix.

PCA gives the best lower-rank approximation to the data vectors: it minimizes [12,13]

$$\mathcal{J} = \sum_{j=1}^{m} \left\| \mathbf{x}_j - \sum_{i=1}^{r} \mathbf{w}_i \left(\mathbf{w}_i^{\mathrm{T}}\mathbf{x}_j\right) \right\|^2.$$

Denoting $\mathbf{W} = (\mathbf{w}_1\ \mathbf{w}_2\ \cdots\ \mathbf{w}_r)$, this can be written as

$$\mathcal{J} = \left\|\mathbf{X} - \mathbf{WW}^{\mathrm{T}}\mathbf{X}\right\|^2 \tag{2}$$

(with matrix Frobenius norm), and thus PCA becomes a special case of the generic problem $\mathbf{X} \approx \mathbf{WH}$ with $\mathbf{H} = \mathbf{W}^{\mathrm{T}}\mathbf{X}$.

If the data vectors are *not* zero-mean, this is still the best approximation, when $\mathbf{w}_j$ are the eigenvectors of the data correlation matrix $\mathbf{XX}^{\mathrm{T}}$ instead of the covariance matrix. The eigenvectors can be solved by standard methods. The eigenvector matrix is not the unique solution, though; any orthogonal rotation in the subspace of the $r$ dominant eigenvectors gives the same minimal error for (2).

Any changes made to the PCA problem such as nonlinearities or extra constraints imply that the eigenvector

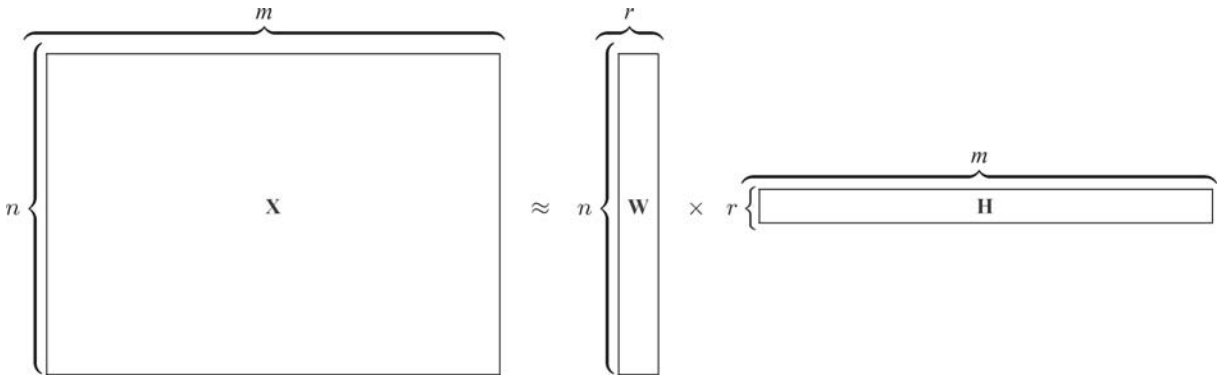


**Fig. 1** Matrix factorization $\mathbf{X} \approx \mathbf{WH}$ with low-rank component matrices

solution is no longer valid. Therefore, gradient learning rules may be needed. One of the present authors suggested such a PCA learning rule [14], that has a neural network flavour (Hebbian learning):

$$\mathbf{W}' = \mathbf{W} + \gamma \left( \mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W} - \mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W} \right) \qquad (3)$$

with $\gamma$ the positive learning rate. This converges to an orthogonal basis matrix of the dominant eigenvector subspace [9].

It may be instructive to derive this rule from the cost function, because this is the starting-point for the multiplicative orthogonal learning rules to be presented in Sect. 3. The cost function (2) is a fourth order polynomial in the elements of $\mathbf{W}$ and its gradient with respect to matrix $\mathbf{W}$ can be shown to be the following third order polynomial [15]:

$$\nabla = -4\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W} + 2\mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W} + 2\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{W}. \qquad (4)$$

If we add here the constraint that the columns of $\mathbf{W}$ must be orthonormal, that is, $\mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}$, which holds for the eigenvectors at the minimal point, then formally we get the gradient descent rule (3). A more rigorous derivation is to use the gradient on the Stiefel manifold of orthogonal matrices $\mathbf{W}$

$$\nabla_S = \nabla - \mathbf{W}\nabla^{\mathrm{T}}\mathbf{W}. \qquad (5)$$

Using this in the gradient rule and substituting $\mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}$ again yields the PCA rule (3).

### 2.2 $K$-means clustering

The relation of the classical $K$-means clustering to the generic approximation problem (1) and PCA has been recently emphasized by Ding et al. [7] in the context of NMF. Let us recall the clustering problem: we have $n$ vectors that have to be partitioned into $r$ clusters. An *indicator matrix* is an $n \times r$ binary matrix $\mathbf{U}$ whose element $U_{i,j} = 1$ iff the $i$th vector belongs to the $j$th cluster, otherwise $U_{i,j} = 0$. The indicator matrix thus completely characterizes the clustering result. The question is then, how to find a good clustering, i.e., an optimal indicator matrix.

One of the most classical clustering methods is *K-means clustering* [16]. Each cluster $C_j$ is defined by its centroid $\mathbf{c}_j$ and the algorithm is very simple:

0) centroids are initialized (often arbitrarily);

1) each vector is placed into the cluster with the nearest centroid;

2) centroids are recomputed;

3) if centroids changed, goto 1), otherwise end.

The $K$-means cost function with $r$ clusters is

$$\mathcal{J} = \sum_{j=1}^{r} \sum_{\mathbf{x}_i \in C_j} \left\| \mathbf{x}_i - \mathbf{c}_j \right\|^2,$$

and the Lloyd algorithm finds a local minimum of this.

Following Ref. [7], the cost function can be written as

$$\mathcal{J} = \sum_i \left\| \mathbf{x}_i \right\|^2 - \sum_k \frac{1}{n_k} \sum_{\mathbf{x}_i, \mathbf{x}_j \in C_k} \mathbf{x}_i^{\mathrm{T}} \mathbf{x}_j,$$

where $n_k$ is the number of vectors in cluster $C_k$. This can be further written as

$$\mathcal{J} = \mathrm{Tr}\left( \mathbf{X}^{\mathrm{T}}\mathbf{X} \right) - \mathrm{Tr}\left( \mathbf{U}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{U} \right), \qquad (6)$$

where $\mathbf{U}$ is now the scaled indicator matrix: each $k$th column of the binary indicator matrix is divided by $\sqrt{n_k}$. Note that with this scaling the columns of $\mathbf{U}$ are orthonormal and it holds $\mathbf{U}^{\mathrm{T}}\mathbf{U} = \mathbf{I}$. But then (6) is exactly equivalent to

$$\mathcal{J} = \left\| \mathbf{X}^{\mathrm{T}} - \mathbf{U}\mathbf{U}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}} \right\|^2. \qquad (7)$$

Matrix $\mathbf{X}$ should thus be approximated in the least-mean-square sense by a matrix $\mathbf{X}\mathbf{U}\mathbf{U}^{\mathrm{T}}$. Again, this corresponds to the general problem (1) with $\mathbf{W} = \mathbf{X}\mathbf{U}$ and $\mathbf{H} = \mathbf{U}^{\mathrm{T}}$. Now the constraint is such that $\mathbf{U}$ is an orthogonal, binary matrix (except for the scaling of the columns to unit norm). Therefore, the optimal solution to (7) can only be found by combinatorial optimization, which scales very badly with the dimensions of the data matrix. Due to the binary nature of $\mathbf{U}$, no gradient methods are possible.

### 2.3 Nonnegative matrix factorization (NMF)

In some applications, it is appropriate to assume that data is nonnegative, or $x_{ij} \geqslant 0$; denote $\mathbf{X} \geqslant 0$. We search for an expansion $\mathbf{X} \approx \mathbf{W}\mathbf{H}$ such that also $\mathbf{W} \geqslant 0$, $\mathbf{H} \geqslant 0$. Typical criteria are least square error and divergence between $\mathbf{X}$ and its approximation. This was called *Nonnegative Matrix Factorization* (NMF) by Lee and Seung [2], and earlier considered by Paatero and Tapper [1].

There is now a large literature on NMF; see e.g., Cichocki et al. [3]. Incorporating nonnegativity in linear factorizations has demonstrated great success for multivariate data. The NMF algorithm [2,17], as well as its variants, has been applied to various problems such as feature extraction and clustering for faces [2,18], text [4,19], music [20], environment [1], financial data [21], Scotch whiskies [22], gene expressions [23], etc.

### 2.4 Projective nonnegative matrix factorization (PNMF)

As a motivation for Projective Nonnegative Matrix Factorization, let us again look at the *PCA criterion*

$$\mathcal{J} = \left\| \mathbf{X} - \mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{X} \right\|^2$$

with $\mathbf{W}$ an orthogonal matrix. As mentioned above, the solution is given by a set of eigenvectors, or can be solved with gradient descent. The solution method scales well with the dimensions of the data matrix.

Let us compare this with the *Clustering criterion*

$$\mathcal{J} = \left\|\mathbf{X}^{\mathrm{T}} - \mathbf{U}\mathbf{U}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\right\|^2$$

with $\mathbf{U}$ an orthogonal, (essentially) binary matrix. The optimal solution can be found by combinatorial optimization, which scales badly with the dimensions of the data matrix.

A compromise might be that $\mathbf{W}$ is an orthogonal, *nonnegative* matrix. Then gradient descent related to the NMF algorithms could be used.

The present authors have recently suggested a variant of NMF called *Projective NMF* (PNMF) [8,15]. The difference to NMF is that

$$\mathbf{H} = \mathbf{W}^{\mathrm{T}}\mathbf{X},$$

in other words,

$$\mathbf{X} \approx \mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{X} \text{ with } \mathbf{W} \geqslant 0.$$

Recently, this was also suggested by Ding et al. [24] as special case of their Convex NMF.

The PNMF cost function is the same as the PCA cost:

$$\mathcal{J}(\mathbf{W}) = \left\|\mathbf{X} - \mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{X}\right\|^2$$

(matrix Frobenius norm), however, with the additional constraint that $\mathbf{W} \geqslant 0$. Interestingly, this produces an *almost orthogonal nonnegative matrix* $\mathbf{W}$.

Before presenting the PNMF learning algorithms, let us briefly review some consequences of the combination of orthogonality and nonnegativity. First, consider two vectors $\mathbf{u} = (u_1\ u_2\ \cdots\ u_n)^{\mathrm{T}}$ and $\mathbf{v} = (v_1\ v_2\ \cdots\ v_n)^{\mathrm{T}}$ that are *nonnegative*: $u_i \geqslant 0$, $v_i \geqslant 0$, and *orthogonal*: $\mathbf{u}^{\mathrm{T}}\mathbf{v} = \sum_i^n u_i v_i = 0$. It follows that all $u_i v_i = 0$, or either $u_i = 0$ or $v_i = 0$ for all $i$. Consider then an $n \times r$ matrix $\mathbf{W}$ that is nonnegative: $\mathbf{W} \geqslant 0$, and orthogonal: $\mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}$ (columns are orthonormal). It is *sparse* in the sense that most of its elements must be zero: each row has at most one non-zero element. As no zero columns are allowed, each column has at least one non-zero element. The total number of non-zero elements is thus between $r$ and $n$.

For example, if $n = 2r$, then the ratio of non-zero elements to all elements $(2r^2)$ is between $1/r$ and $1/(2r)$. For large dimensionalities, the matrix is quite sparse.

Note that such a *nonnegative orthogonal matrix is a very good substitute for a clustering indicator matrix.* Recall that an indicator matrix is a binary $n \times r$ matrix $\mathbf{U}$ whose element $U_{i,j} = 1$ iff the $i$th vector belongs to the $j$th cluster. Because every vector belongs to exactly one cluster, each row of $\mathbf{U}$ has exactly one non-zero element. Because there are no empty clusters, each column has at least one non-zero element. These are exactly the properties of an orthogonal nonnegative matrix.

In Sect. 3 we will show by several experiments how the PNMF principle can be effectively used for combinatorial optimization problems.

Ideally, PNMF would perform "nonnegative PCA" by producing an orthogonal rotation of the dominant eigenvector basis of matrix $\mathbf{X}\mathbf{X}^{\mathrm{T}}$ into nonnegative basis vectors. Then the approximation error would be the smallest possible, and yet nonnegativity would hold. This is not possible in the general case, however. Instead, numerical experimentation shows that the minimal solution of the PNMF problem is a nonnegative basis with high degree of orthogonality, spanning a subspace close to the dominant eigenvector subspace. There are several post-processing methods by which a truly orthogonal nonnegative basis could be achieved. For example, in the simplest case, one may put to zero all the elements in any given row except one and then re-normalize the columns to unit norm. Also, from the approximately orthogonal nonnegative matrix $\mathbf{W}$ one may compute a new matrix $\mathbf{W}' = \mathbf{W}\left(\mathbf{W}^{\mathrm{T}}\mathbf{W}\right)^{-1/2}$ and truncate away the nonnegative elements. Note, however, that any such post-processing will increase the value of the cost function somewhat. Thus there is a trade-off of getting minimal approximating error with a $\mathbf{W}$ matrix only approximately orthogonal, and getting a slightly larger approximation error with a fully orthogonal nonnegative matrix.

As an experimental proof of the performance of PNMF, let us look at the first application of Lee and Seung's NMF, that was image feature extraction [2]. In this application, the sparsity offered by a nonnegative orthogonal matrix would be of great advantage. If we could have

$$\mathbf{X} \approx \mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{X}$$

with $\mathbf{W}$ orthogonal and nonnegative, then it follows

$$\mathbf{x} \approx \sum_{i=1}^{r} \mathbf{w}_i \left(\mathbf{w}_i^{\mathrm{T}}\mathbf{x}\right)$$

for any column of $\mathbf{X}$ (an individual data item, like an image stacked row-by-row). The $\mathbf{w}_i$ are the orthonormal columns of $\mathbf{W}$.

The PNMF method is now able to produce a highly orthogonal nonnegative matrix $\mathbf{W}$. We can look at the difference in sparsity between NMF and PNMF in the classical example with face images. We use the MIT-CBCL database, having 2429 images of size $19 \times 19 = 361$ pixels. The data matrix $\mathbf{X}$ thus has dimensions $n \times m = 361 \times 2429$. We do compression to rank $r = 49$. The results are given in Figs. 2 and 3. It can be seen in Fig. 2 how PNMF is able to construct a much sparser and more orthogonal basis than NMF. The quality of the PNMF

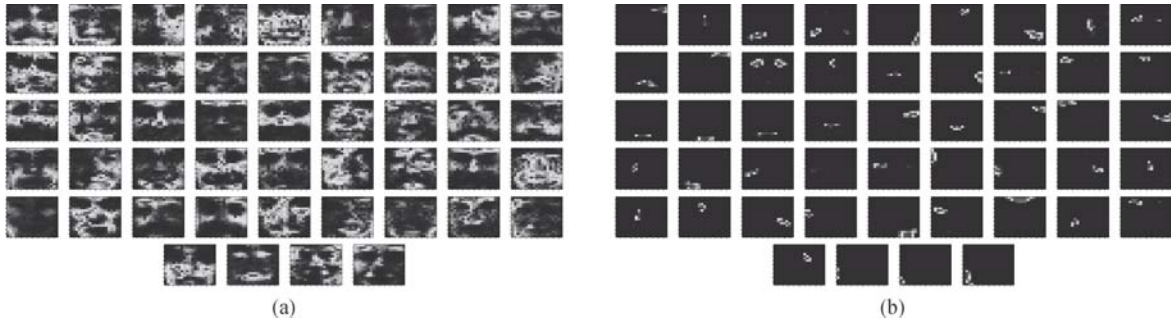


**Fig. 2** Basis images (columns of $\mathbf{W}$) for (a) NMF and (b) PNMF. Number of basis vectors: 49, dimensionality: 361

reconstruction in Fig. 3 is visually not any worse than that of NMF.

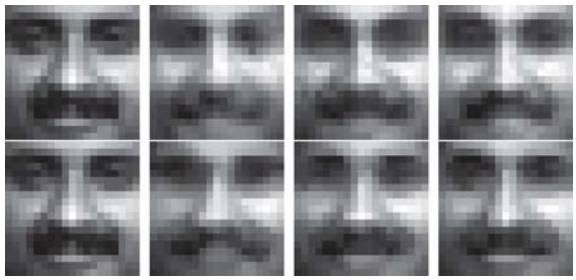

**Fig. 3** Original face image (left) and its reconstructions by NMF (top row) and PNMF (the second row). The dimensions in columns 2, 3, and 4 are 25, 49, and 81, respectively

In the next section, we shall present learning algorithms for PNMF in a slightly more general setting.

## 3 Orthogonality and multiplicative updates

### 3.1 Nonnegative multiplicative learning

In nonnegative learning algorithms, multiplicative updates are widely used. Given the gradient of an objective function, a multiplicative factor can easily be formed by putting part of the gradient terms to the numerator and the rest to the denominator. The updated parameter in learning will remain nonnegative if all separated terms in the multiplicative factor are nonnegative. Such learning algorithms require no user-specified learning rate and many of them have been shown to be locally convergent (e.g., Refs. [17,25,26]).

As a starting-point, consider the standard gradient ascent algorithm for finding a matrix $\mathbf{W}$ maximizing an objective function $\mathcal{J}(\mathbf{W})$:

$$\mathbf{W}' = \mathbf{W} + \gamma\nabla, \tag{8}$$

where $\nabla$ is the gradient of $\mathcal{J}(\mathbf{W})$. Suppose that matrix $\mathbf{W}$ has nonnegative elements, and by some external knowledge, there exists a separation of the gradient matrix into two (elementwise) positive matrices $\nabla^+, \nabla^-$ such that $\nabla = \nabla^+ - \nabla^-$.

Then, by choosing for each matrix element in (8) a separate learning rate

$$\gamma_{ik} = \frac{W_{ik}}{\nabla^-_{ik}},$$

a nonnegative multiplicative update rule directly follows:

$$W'_{ik} = W_{ik}\frac{\nabla^+_{ik}}{\nabla^-_{ik}}. \tag{9}$$

Such multiplicative learning rules are widely used in nonnegative learning, because the above update maintains the nonnegativity of $W_{ik}$. In addition, $W_{ik}$ increases when $\nabla^+_{ik} > \nabla^-_{ik}$, i.e., $\nabla_{ik} > 0$, and decreases if $\nabla_{ik} < 0$. Thus the multiplicative change of $W_{ik}$ indicates how much the direction of that axis conforms to the gradient direction. There exist two kinds of stationary points in the iterative use of the multiplicative update rule (9): one satisfies $\nabla^+_{ik} = \nabla^-_{ik}$, i.e., $\nabla = \mathbf{0}$, which is the same condition for local optima as in the conventional gradient descent updates, and the other one is $W_{ik} = 0$. The latter condition distinguishes nonnegative optimization from conventional ones and often yields sparsity in $\mathbf{W}$, which is desired in many applications. Furthermore, unlike steepest gradient or exponential gradient [27], the multiplicative update rule (9) does not require any user-specified learning rates, which facilitates its application.

As the starting point to derive the Orthogonal Nonnegative Learning (ONL) principle, consider the following constrained optimization problem:

$$\max_{\mathbf{W}\geqslant\mathbf{0}} \quad \mathcal{J}(\mathbf{W}), \tag{10}$$

$$\text{subject to} \quad \mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}. \tag{11}$$

Without the orthogonality constraint, but with the nonnegativity constraint, rule (9) provides a viable solution.

Without the nonnegativity constraint, but with the orthogonality constraint, it was shown by Ref. [9] that the following constrained gradient can be used for maximixation of objective functions with certain symmetry properties:

$$\nabla_\perp = \nabla - \mathbf{W}\mathbf{W}^{\mathrm{T}}\nabla.$$

This gradient is related to the canonical gradient on the Stiefel manifold of orthogonal matrices (5), and it combines maximization and orthonormalization into the

same learning rule. The two gradients are equal if $\mathbf{W}^{\mathrm{T}}\nabla$ is symmetric, as is the case for example with the PCA gradient in (4).

Combining now these two approaches, the multiplicative update rule of the constrained problem (10)–(11) will be

$$W'_{ik} = W_{ik}\frac{\left(\nabla^+ + \mathbf{W}\mathbf{W}^{\mathrm{T}}\nabla^-\right)_{ik}}{\left(\nabla^- + \mathbf{W}\mathbf{W}^{\mathrm{T}}\nabla^+\right)_{ik}}. \tag{12}$$

It was reported in Ref. [5] that the learning rule (12) is able to generate favorable results in facial image analysis. However, the interpretation in Ref. [5] was intuitive.

Here we employ the Lagrangian method to rigorously show that the update using (12) can jointly maximize $\mathcal{J}(\mathbf{W})$ and force $\mathbf{W}$ to approach the nonnegative Stiefel manifold.

**Theorem 1** Suppose there is an auxiliary function (see Appendix A) $G(\mathbf{W}', \mathbf{W})$ that lower bounds $\mathcal{J}(\mathbf{W}')$ and the derivative of $G(\mathbf{W}', \mathbf{W})$ with respect to $\mathbf{W}'$ is

$$\frac{\partial G(\mathbf{W}', \mathbf{W})}{\partial W'_{ik}} = \nabla^+_{ik} - \frac{W'_{ik}}{W_{ik}}\nabla^-_{ik}. \tag{13}$$

The multiplicative update rule (12) is an iterative Lagrangian solution of the orthogonal nonnegative optimization problem (10)–(11).

**Proof** When the objective $\mathcal{J}(\mathbf{W})$ to be maximized is accompanied with the orthonormality constraint $\mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}$, a generalized objective $\tilde{\mathcal{J}}(\mathbf{W}, \mathbf{\Lambda})$ can be formulated by introducing a set of Lagrangian multipliers $\{\Lambda_{kl}\}$. The generalized objective using Lagrangian multipliers is

$$\tilde{\mathcal{J}}(\mathbf{W}, \mathbf{\Lambda}) = \mathcal{J}(\mathbf{W}) + \mathrm{Tr}\left(\mathbf{\Lambda}^{\mathrm{T}}\left(\mathbf{W}^{\mathrm{T}}\mathbf{W} - \mathbf{I}\right)\right). \tag{14}$$

For nonnegative optimization of the Lagrangian, we now construct an auxiliary function $\tilde{G}(\mathbf{W}', \mathbf{W})$ for $\tilde{\mathcal{J}}(\mathbf{W}, \mathbf{\Lambda})$ by using suitable tight lower bounds.

It can be shown that

$$\begin{aligned}\tilde{G}(\mathbf{W}', \mathbf{W}) &= G(\mathbf{W}', \mathbf{W}) + \sum_{ik}\frac{(\mathbf{W}\mathbf{\Lambda})_{ik}\,{W'_{ik}}^2}{W_{ik}} \\ &+ \sum_{ik}\left(\mathbf{W}\mathbf{W}^{\mathrm{T}}\nabla^+\right)_{ik}\left(W'_{ik} - \frac{{W'_{ik}}^2 + W_{ik}^2}{2W_{ik}}\right) \\ &- \mathrm{Tr}(\mathbf{\Lambda})\end{aligned} \tag{15}$$

is an auxiliary function of $\tilde{\mathcal{J}}(\mathbf{W}', \mathbf{\Lambda})$. That is, $\tilde{G}(\mathbf{W}', \mathbf{W}) \leqslant \tilde{\mathcal{J}}(\mathbf{W}', \mathbf{\Lambda})$ where the equality holds if and only if $\mathbf{W}' = \mathbf{W}$. Here the second and third terms in the right-hand side of (15) are tight lower bounds of $\mathrm{Tr}\left(\mathbf{\Lambda}^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\mathbf{W}\right)$ and zero, respectively, which can be proven [4,17] by writing $W'_{ik} = u_{ik}W_{ik}$, where $u_{ik} \in \mathbb{R}$. The third term, called *moving-term* technique [15], is introduced to add the same term $\left(\mathbf{W}\mathbf{W}^{\mathrm{T}}\nabla^+\right)_{ik}$ to both numerator and denominator of the resulting multiplicative update rule and thus maintains the nonnegativity of $\mathbf{W}$.

Setting $\partial\tilde{G}(\mathbf{W}', \mathbf{W})/\partial W'_{ik} = 0$, we obtain

$$W'_{ik} = W_{ik}\frac{\left(\nabla^+ + 2\mathbf{W}\mathbf{\Lambda} + \mathbf{W}\mathbf{W}^{\mathrm{T}}\nabla^+\right)_{ik}}{\left(\nabla^- + \mathbf{W}\mathbf{W}^{\mathrm{T}}\nabla^+\right)_{ik}}. \tag{16}$$

The Lagrangian multipliers $\mathbf{\Lambda}$ can next be determined by using the Karush-Kuhn-Tucker (K.K.T.) conditions and substituted into (16).

From $\partial\tilde{\mathcal{J}}(\mathbf{W}, \mathbf{\Lambda})/\partial\mathbf{W} = \mathbf{0}$, we get $2\mathbf{W}\mathbf{\Lambda} = \nabla^- - \nabla^+$. Using $\partial\tilde{\mathcal{J}}(\mathbf{W}, \mathbf{\Lambda})/\partial\mathbf{\Lambda} = \mathbf{0}$, i.e., $\mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}$, one obtains

$$\mathbf{\Lambda} = \frac{1}{2}\mathbf{W}^{\mathrm{T}}\left(\nabla^- - \nabla^+\right). \tag{17}$$

Inserting (17) to (16), the multiplicative update rule becomes identical to (12). □

The update rule (12) is called an *iterative Lagrangian solution* for the constrained optimization problem. The definition of the auxiliary function guarantees that $\tilde{\mathcal{J}}(\mathbf{W}, \mathbf{\Lambda})$ is monotonically increasing if one repeatedly applies (12). The iterations will converge to a local maximum if there is an upper bound of $\tilde{\mathcal{J}}(\mathbf{W}, \mathbf{\Lambda})$. An iterative Lagrangian solution jointly maximizes the original objective and the orthogonality.

However, notice that the iterative Lagrangian solution does not keep matrix $\mathbf{W}$ orthogonal in the optimization. Instead, it jointly maximizes $\mathcal{J}(\mathbf{W})$ and forces $\mathbf{W}$ to approach orthogonality. Actually $\mathbf{W}$ never exists in the nonnegative Stiefel manifold because strict orthogonality would require many entries of a nonnegative matrix to become exactly zero. If $\mathbf{W}$ is initialized in such a manifold, the convergence will be very poor, because the multiplicative updates cannot recover a zero entry to be positive.

### 3.2 An alternative update rule

Recently, another type of multiplicative update rules of the form

$$W'_{ik} = W_{ik}\sqrt{\frac{\nabla^+_{ik}}{\nabla^-_{ik}}} \tag{18}$$

have been proposed [24,25]. This is usually derived by constructing an auxiliary function $F(\mathbf{W}', \mathbf{W})$ for $\mathcal{J}(\mathbf{W}')$ and with the derivative

$$\frac{\partial F(\mathbf{W}', \mathbf{W})}{\partial W'_{ik}} = \frac{W_{ik}}{W'_{ik}}\nabla^+_{ik} - \frac{W'_{ik}}{W_{ik}}\nabla^-_{ik}. \tag{19}$$

We can similarly derive the alternative multiplicative rule

$$W'_{ik} = W_{ik}\sqrt{\frac{\left(\nabla^+ + \mathbf{W}\mathbf{W}^{\mathrm{T}}\nabla^-\right)_{ik}}{\left(\nabla^- + \mathbf{W}\mathbf{W}^{\mathrm{T}}\nabla^+\right)_{ik}}} \tag{20}$$

for the constrained optimization problem (10)–(11).

**Theorem 2** The multiplicative update rule (20) is an iterative Lagrangian solution of the orthogonal nonnegative optimization problem (10)–(11).

**Proof** We employ the alternative auxiliary function

$$\tilde{F}(\mathbf{W}', \mathbf{W}) = F(\mathbf{W}', \mathbf{W}) + \sum_{ik} \frac{(\mathbf{W}\mathbf{\Lambda})_{ik} {W'_{ik}}^2}{W_{ik}} + \sum_{ik} \left(\mathbf{W}\mathbf{W}^{\mathrm{T}}\nabla^+\right)_{ik} \left( W_{ik} + W_{ik} \log \frac{W'_{ik}}{W_{ik}} - \frac{{W'_{ik}}^2 + W_{ik}^2}{2W_{ik}} \right) + \mathrm{Tr}(\mathbf{\Lambda}) \tag{21}$$

for $\tilde{\mathcal{J}}(\mathbf{W})$. Compared with (15), $G(\mathbf{W}', \mathbf{W})$ is replaced with $F(\mathbf{W}', \mathbf{W})$ in the first term in the right-hand side of (21), while the second and the fourth terms remain the same. The third term is again for adding $\mathbf{W}\mathbf{W}^{\mathrm{T}}\nabla^+$ to both numerator and denominator. To see it is a tight lower bound of zero, one can separate this term into

$$\sum_{ik} \left(\mathbf{W}\mathbf{W}^{\mathrm{T}}\nabla^+\right)_{ik} \left( W'_{ik} - \frac{{W'_{ik}}^2 + W_{ik}^2}{2W_{ik}} \right), \tag{22}$$

$$- \sum_{ik} \left(\mathbf{W}\mathbf{W}^{\mathrm{T}}\nabla^+\right)_{ik} \left( W'_{ik} - W_{ik} - W_{ik} \log \frac{W'_{ik}}{W_{ik}} \right). \tag{23}$$

(22) is exactly the third term in the right-hand side of (15), which has been shown to a tight lower bound of zero. (23) is also no greater than zero because

$$W'_{ik} - W_{ik} - W_{ik} \log \frac{W'_{ik}}{W_{ik}} \geqslant 0 \tag{24}$$

due to the inequality $z \geqslant 1 + \log z$ for all $z \geqslant 0$, where the equality holds if and only if $z = 1$. The multiplicative update rule (20) can thus be obtained by setting $\partial \tilde{F}(\mathbf{W}', \mathbf{W}) / \partial W'_{ik} = 0$ and with $\mathbf{\Lambda}$ substituted using (17). □

### 3.3 Remarks

Our derivation of the multiplicative rules for learning nonnegative projections is based on the Lagrangian approach. That is, one should apply the K.K.T. condition $\mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}$ to simplify the resulting multiplicative update rule after transforming (9) to its orthogonal counterpart (12) or (20).

The same term $\mathbf{W}\mathbf{W}^{\mathrm{T}}\nabla^-$ is added to both the numerator and denominator in our derivation, which aims to move the unsigned negative terms of $\mathbf{W}\mathbf{\Lambda}$ to the numerator. In some special cases, such negative terms may cancel with $\nabla^-$ (or part of it) after applying $\mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}$ and thus the denominator is already nonnegative. The *moving-term* technique therefore becomes unnecessary in such cases. Actually provided both update rules are convergent, the updates without adding $\mathbf{W}\mathbf{W}^{\mathrm{T}}\nabla^-$ to both the numerator and denominator would converge faster because $(a + c)/(b + c)$ is closer to one than $a/b$ for $a, b, c > 0$.

We are now ready to collect the above observations into a generic principle for turning a nonnegative multiplicative update rule into the corresponding version, where also the orthogonality constraint is incorporated. We call this the *Orthogonal Nonnegative Learning* (ONL) *principle*, and it is given in Fig. 4. Some examples of applying the ONL principle are given in the following.

### 3.4 Example 1: NMF

Denote the squared Frobenius norm $\|\mathbf{A}\|_F^2 = \sum_{ij} A_{ij}^2$. *Nonnegative Matrix Factorization* (NMF) based on the Frobenius norm seeks two nonnegative matrices $\mathbf{W} \in \mathbb{R}_+^{m \times r}$ and $\mathbf{H} \in \mathbb{R}_+^{r \times N}$ that minimize $\mathcal{L}(\mathbf{W}, \mathbf{H}) = \frac{1}{2}\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2$ for a nonnegative input matrix $\mathbf{X} \in \mathbb{R}_+^{m \times N}$. Of course, the minimization problem can be equivalently converted to a maximization one by introducing $\mathcal{J}(\mathbf{W}, \mathbf{H}) = -\mathcal{L}(\mathbf{W}, \mathbf{H})$.

Let us consider the update rule of $\mathbf{W}$ for example. The gradient $\nabla_{\mathbf{W}} \mathcal{J}(\mathbf{W}, \mathbf{H}) = \nabla^+ - \nabla^-$ where $\nabla^+ = \mathbf{X}\mathbf{H}^{\mathrm{T}}$ and $\nabla^- = \mathbf{W}\mathbf{H}\mathbf{H}^{\mathrm{T}}$ suggests the NMF multiplicative update rule [2,17]

$$W'_{ik} = W_{ik} \frac{\nabla^+_{ik}}{\nabla^-_{ik}} = W_{ik} \frac{\left(\mathbf{X}\mathbf{H}^{\mathrm{T}}\right)_{ik}}{\left(\mathbf{W}\mathbf{H}\mathbf{H}^{\mathrm{T}}\right)_{ik}}. \tag{25}$$

It has been shown that $\mathcal{J}(\mathbf{W}, \mathbf{H})$ is not decreasing with the update (25) [17]. If $\mathbf{W}$ is constrained to be orthonormal, one can apply the ONL principle to obtain a learning algorithm as follows.

Given a multiplicative update rule for nonnegative learning without orthogonality constraint:

$$W'_{ik} = W_{ik} \frac{\nabla^+_{ik}}{\nabla^-_{ik}} \quad \text{or} \quad W'_{ik} = W_{ik} \sqrt{\frac{\nabla^+_{ik}}{\nabla^-_{ik}}},$$

change this into the corresponding update rule with orthogonality constraint:

1) Rewrite the update rule as

$$W'_{ik} = W_{ik} \frac{\left(\nabla^+ + \mathbf{W}\mathbf{W}^{\mathrm{T}}\nabla^-\right)_{ik}}{\left(\nabla^- + \mathbf{W}\mathbf{W}^{\mathrm{T}}\nabla^+\right)_{ik}} \quad \text{or} \quad W'_{ik} = W_{ik} \sqrt{\frac{\left(\nabla^+ + \mathbf{W}\mathbf{W}^{\mathrm{T}}\nabla^-\right)_{ik}}{\left(\nabla^- + \mathbf{W}\mathbf{W}^{\mathrm{T}}\nabla^+\right)_{ik}}}.$$

2) Simplify the update rule by substituting $\mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}$.

3) Remove the duplicates that appear in both numerator and denominator.

**Fig. 4** Orthogonal Nonnegative Learning (ONL) principle for reforming a multiplicative update rule into its orthogonal version

1) Rewrite the multiplicative update rule according to (12):

$$W'_{ik} = W_{ik}\frac{\left(\nabla^+ + \mathbf{W}\mathbf{W}^{\mathrm{T}}\nabla^-\right)_{ik}}{\left(\nabla^- + \mathbf{W}\mathbf{W}^{\mathrm{T}}\nabla^+\right)_{ik}} = W_{ik}\frac{\left(\mathbf{X}\mathbf{H}^{\mathrm{T}} + \mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{W}\mathbf{H}\mathbf{H}^{\mathrm{T}}\right)_{ik}}{\left(\mathbf{W}\mathbf{H}\mathbf{H}^{\mathrm{T}} + \mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{X}\mathbf{H}^{\mathrm{T}}\right)_{ik}}. \quad (26)$$

2) Simplify the above rule by using $\mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}$:

$$W'_{ik} = W_{ik}\frac{\left(\mathbf{X}\mathbf{H}^{\mathrm{T}} + \mathbf{W}\mathbf{H}\mathbf{H}^{\mathrm{T}}\right)_{ik}}{\left(\mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{X}\mathbf{H}^{\mathrm{T}} + \mathbf{W}\mathbf{H}\mathbf{H}^{\mathrm{T}}\right)_{ik}}. \quad (27)$$

3) Remove the duplicate $\mathbf{W}\mathbf{H}\mathbf{H}^{\mathrm{T}}$ in both numerator and denominator:

$$W'_{ik} = W_{ik}\frac{\left(\mathbf{X}\mathbf{H}^{\mathrm{T}}\right)_{ik}}{\left(\mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{X}\mathbf{H}^{\mathrm{T}}\right)_{ik}}. \quad (28)$$

This is exactly the multiplicative update rule for *Orthogonal Nonnegative Matrix Factorization* presented in Ref. [4].

### 3.5 Example 2: PNMF based on Frobenius norm

A variant of NMF, called *Projective Nonnegative Matrix Factorization* (PNMF), was proposed in Refs. [8,15,28]. As reviewed in Sect. 2, PNMF replaces the matrix $\mathbf{H}$ with $\mathbf{W}^{\mathrm{T}}\mathbf{X}$. The resulting objective function $\mathcal{J}_{\mathrm{PNMF}}(\mathbf{W}) = -\frac{1}{2}\left\|\mathbf{X} - \mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{X}\right\|_F^2$ has the gradient parts $\nabla^+ = \mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}$, $\nabla^- = \mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W} + \mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{W} - \mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}$. In Appendix B, we show an auxiliary function for $\mathcal{J}_{\mathrm{PNMF}}(\mathbf{W})$, which leads to the multiplicative update rule

$$W'_{ik} = W_{ik}\frac{\left(\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}\right)_{ik}}{\left(\mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W} + \mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{W} - \mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}\right)_{ik}}. \quad (29)$$

The ONL principle transforms the above rule to its orthogonal version by the following steps.

1) Rewrite the update rule (29) according to (12):

$$W'_{ik} = W_{ik}\frac{\left(\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W} + \mathbf{W}\mathbf{W}^{\mathrm{T}}\left(\mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W} + \mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{W} - \mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}\right)\right)_{ik}}{\left(\mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W} + \mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{W} - \mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W} + \mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}\right)_{ik}}. \quad (30)$$

2) Simplified with $\mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}$, the rule becomes

$$W'_{ik} = W_{ik}\frac{\left(\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}\right)_{ik}}{\left(\mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}\right)_{ik}}, \quad (31)$$

which is identical to the one proposed as NLHN in Ref. [5] and as OPNMF in Ref. [15]. Note the analogy between this rule for PNMF and the classical gradient descent learning rule for PCA given in (3). The rule above is clearly the multiplicative equivalent of the PCA learning rule, and it results in a nonnegative basis matrix $\mathbf{W}$.

### 3.6 Example 3: PNMF based on divergence

Alternatively, the difference between the input matrix $\mathbf{X}$ and its approximate $\hat{\mathbf{X}} = \mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{X}$ in PNMF can be measured by the Kullback-Leibler (KL) divergence:

$$D\left(\mathbf{X}||\hat{\mathbf{X}}\right) = \sum_{i,j}\left(X_{ij}\log\frac{X_{ij}}{\hat{X}_{ij}} - X_{ij} + \hat{X}_{ij}\right). \quad (32)$$

Denote $Z_{ij} = X_{ij}/\hat{X}_{ij}$ and $\mathcal{J}(\mathbf{W}) = -D\left(\mathbf{X}||\mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{X}\right)$. The derivative

$$\frac{\partial\mathcal{J}(\mathbf{W})}{\partial W_{ik}} = \mathbf{X}\mathbf{Z}^{\mathrm{T}}\mathbf{W} + \mathbf{Z}\mathbf{X}^{\mathrm{T}}\mathbf{W} - \sum_j\left(\mathbf{W}^{\mathrm{T}}\mathbf{X}\right)_{kj} - \left(\sum_j X_{ij}\right)\left(\sum_a W_{ak}\right) \quad (33)$$

suggests the multiplicative update rule

$$W'_{ik} = W_{ik}\frac{\left(\mathbf{X}\mathbf{Z}^{\mathrm{T}}\mathbf{W} + \mathbf{Z}\mathbf{X}^{\mathrm{T}}\mathbf{W}\right)_{ik}}{\sum_j\left(\mathbf{W}^{\mathrm{T}}\mathbf{X}\right)_{kj} + \left(\sum_j X_{ij}\right)\left(\sum_a W_{ak}\right)} \quad (34)$$

of PNMF based on KL-divergence [8,28]. If $\mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}$ is enforced, we can apply the ONL principle and directly obtain

$$W'_{ik} = W_{ik}\frac{B_{ik} + \left(\mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{C}\right)_{ik}}{C_{ik} + \left(\mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{B}\right)_{ik}}, \quad (35)$$

where $\mathbf{B} = \mathbf{Z}\mathbf{X}^{\mathrm{T}}\mathbf{W} + \mathbf{X}\mathbf{Z}^{\mathrm{T}}\mathbf{W}$ and $C_{ik} = \sum_j\left(\mathbf{W}^{\mathrm{T}}\mathbf{X}\right)_{kj} + \left(\sum_j X_{ij}\right)\left(\sum_a W_{ak}\right)$ for notational brevity. Besides the ONL principle, a more dedicated and detailed justification of the update rule (35) using the Lagrangian technique can be found in Ref. [15].

## 4 Graph partitioning by orthogonal nonnegative learning

The advantages of using the orthogonality constraint have been demonstrated for feature extraction and for clustering of multi-dimensional vectorial data [4–6,15]. In this section we further confirm the benefits in a non-vectorial application — graph partitioning. Given a positive integer $r$, graph partitioning divides the vertices of a graph into $r$ disjoint sets such that a certain objective is optimized. Graph partitioning has a wide range of

applications such as data clustering and finding communities in a social network. In this paper we only consider undirected graphs.

### 4.1 Graph partitioning by trace maximization

Consider a graph $\mathcal{G}$ with $N$ vertices and $n$ edges to be partitioned. The connections in the graph are represented by a real-valued $N \times N$ affinity matrix $\mathbf{A}$, whose element $A_{ij}$ gives the weight of the edge connecting vertices $i, j$. The optimization of many graph partitioning objectives, for example, minimizing the number of removed edges in partitioning, is NP-complete due to the complexity of algorithms in a discrete space. It is therefore advantageous to employ a continuous approximation of the objectives, which enables the development of efficient optimization algorithms by using differential calculus.

A promising approximation is to reformulate the graph partitioning problem to

$$\max_{\mathbf{W} \geqslant \mathbf{0}} \quad \mathcal{J}(\mathbf{W}) = \mathrm{Tr}\left(\mathbf{W}^{\mathrm{T}}\mathbf{S}\mathbf{W}\right), \tag{36}$$

$$\text{subject to} \quad \mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}, \tag{37}$$

where $\mathbf{S} \in I\!R^{N \times N}$ is a symmetric matrix derived from $\mathbf{A}$. Here $\mathbf{W} = \mathbf{C}\left(\mathbf{C}^{\mathrm{T}}\mathbf{C}\right)^{-1/2}$ is the reweighted version of the partition indicator matrix $\mathbf{C} \in \{0,1\}^{N \times r}$ where $C_{ik} = 1$ if the $i$th vertex belongs to the $k$th partition and $C_{ik} = 0$ otherwise. The reweighting enforces orthonormality of $\mathbf{W}$, which does not affect the partition indication but facilitates the optimization.

Some popular choices of $\mathbf{S}$ include the negative *Laplacian matrix* [29], the *modularity matrix* [30], the *vertex similarity matrix* [31], and the *discriminative K-means matrix* [32]. Here we employ the last one, which is given by

$$\mathbf{S} = \mathbf{I} - \left(\mathbf{I} + \frac{1}{\lambda}\mathbf{A}\right)^{-1}, \tag{38}$$

as it has been shown to have favorable discriminative power for data clustering [32]. We empirically set the regularization parameter $\lambda = 10$ in our work.

Neglecting the nonnegativity constraint, the problem (36)–(37) becomes the *Principal Component Analysis* problem and can be solved by e.g., eigenvalue decomposition of $\mathbf{S}$. However, the resulting eigenvectors may contain negative values which are difficult to interpret in terms of indicating partitions. Therefore, the trace maximization based on eigenvalue decomposition is only suitable for two-way partitioning [30]. Repeated subdivision of a network into two parts may easily fail to find the optimal objective [30].

### 4.2 ONL update rules

Without the orthonormality constraint (37), the nonnegative learning problem (36) has the multiplicative update rule

$$W'_{ik} = W_{ik}\frac{(\mathbf{S}^{+}\mathbf{W})_{ik}}{(\mathbf{S}^{-}\mathbf{W})_{ik}} \tag{39}$$

or

$$W'_{ik} = W_{ik}\sqrt{\frac{(\mathbf{S}^{+}\mathbf{W})_{ik}}{(\mathbf{S}^{-}\mathbf{W})_{ik}}}, \tag{40}$$

where $S^{+}_{st} = (|S|_{st} + S_{st})/2$ and $S^{-}_{st} = (|S|_{st} - S_{st})/2$ are the positive and unsigned negative elements of $\mathbf{S}$. In Appendix C, we show that the updates using the above rules converge to a local maximum of $\mathrm{Tr}\left(\mathbf{W}^{\mathrm{T}}\mathbf{S}\mathbf{W}\right)$.

Applying now the ONL principle, we obtain the corresponding multiplicative update rules for (36)–(37):

$$W'_{ik} = W_{ik}\frac{\left(\mathbf{S}^{+}\mathbf{W} + \mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{S}^{-}\mathbf{W}\right)_{ik}}{\left(\mathbf{S}^{-}\mathbf{W} + \mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{S}^{+}\mathbf{W}\right)_{ik}} \tag{41}$$

or

$$W'_{ik} = W_{ik}\sqrt{\frac{\left(\mathbf{S}^{+}\mathbf{W} + \mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{S}^{-}\mathbf{W}\right)_{ik}}{\left(\mathbf{S}^{-}\mathbf{W} + \mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{S}^{+}\mathbf{W}\right)_{ik}}}. \tag{42}$$

We note in passing that if the matrix $\mathbf{S}$ in (38) is positive definite, the above update rules can alternatively be derived by using the kernel technique of PNMF [15,24]. Denote $\mathbf{\Phi} = [\phi(\mathbf{x}_1)\ \phi(\mathbf{x}_2)\ \cdots\ \phi(\mathbf{x}_n)]^{\mathrm{T}}$, where $\mathbf{x}_i$ are the data vectors and they are implicitly mapped into another vector space $\mathcal{V}$ by a function $\phi$. The objective of kernel PNMF thus is $-\frac{1}{2}\left\|\mathbf{\Phi} - \mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{\Phi}\right\|_F^2 = \frac{1}{2}\mathrm{Tr}\left(\mathbf{W}^{\mathrm{T}}\mathbf{S}\mathbf{W} - \mathbf{W}\mathbf{W}^{\mathrm{T}}\mathbf{S}\mathbf{W}\mathbf{W}^{\mathrm{T}}\right) - \mathrm{Tr}(\mathbf{S})$ by writing $\mathbf{S} = \mathbf{\Phi}\mathbf{\Phi}^{\mathrm{T}}$. In Ref. [15], the Lagrangian technique was then applied to decouple the auto-association of $\mathbf{W}$ to obtain the same update rules (41) and (42). However, such kernel derivation is theoretically problematic because the kernel decomposition $\mathbf{S} = \mathbf{\Phi}\mathbf{\Phi}^{\mathrm{T}}$ is invalid if matrix $\mathbf{S}$ in (38) is not positive definite. Instead, the proposed ONL principle directly works on $\mathbf{S}$ and has no such restriction of positive definiteness. Actually, we will show in Sect. 4.3 that the orthogonal nonnegative update rules (41) and (42) are particularly advantageous for those discriminative $K$-means matrices that are far from positive definiteness.

### 4.3 Experiments on graph partitioning

In our experiments, we have used a wide range of undirected graphs from Newman's collection[1)] and the Pajek

1) http://www-personal.umich.edu/~mejn/netdata/

database[2]. The graph size ranges from 24 to 937 vertices and the number of partitions from 2 to 14. The graph affinity matrix $\mathbf{A}$ can be either sparse or not. The context descriptions of these datasets are as follows.

- *Scotland*. The dataset contains the corporate interlocks in Scotland in the beginning of the twentieth century. The involved 108 companies are classified to eight industry types: oil and mining, railway, engineering and steel, electricity and chemicals, domestic products, banks, insurance, and investment.
- *Cities*. This dataset consists of the service values (indicating the importance of a city in the office network of a firm) of 46 global advanced producer service firms over 55 world cities. The firms are grouped into four categories: accountancy, advertising, banking/finance, and law.
- *WorldCities*. A more comprehensive version of *Cities* where 100 global service firms across 315 cities worldwide are included. Two additional firm categories, insurance and management consultancy, are added.
- *Journals*. The dataset contains 124 magazines and journals of 14 classes in Slovenia. Each pair of magazines/journals is associated with the number of common readers.
- *DutchElite*. A 2-mode network of 3810 persons in 937 administrative bodies which are classified according to their main task. We have used only the three largest classes (advice, administration, and inspection) and merged the other classes into a miscellaneous one.
- *Strike* and *Sawmill*. Communication networks of employees who were on strike in a wood-processing facility. The classification is according to age and ethnic group: Spanish-speaking employees, young (30 or younger) English-speaking employees, and old (over 30) English-speaking employees.
- *Korea*. A communication network of 39 women in Korea about family planning. The women are classified by their membership in a Mothers' Club.
- *Football*. A network of American football games between Division IA colleges during regular season Fall 2000. The 115 teams are divided into 11 groups as well as an additional miscellaneous group.
- *A99m*. The data corresponds to a graph of the characters and their relations in the long-running German soap opera called 'Lindenstrasse'. The characters are classified into three groups: inactive, active, and others.

The statistics of the selected datasets are summarized in Table 1. If the discriminative $K$-means matrix $\mathbf{S}$ is not positive definite, $p$ is the largest integer such that there is an upper triangular matrix $\mathbf{R}$ with $\mathbf{R}^{\mathrm{T}}\mathbf{R} = \mathbf{S}_{1:p-1,1:p-1}$[3]. Therefore a smaller $p/N$ value usually indicates that $\mathbf{S}$ is farther from positive definiteness. The $p/N$ value is neglected if the discriminative $K$-means matrix $\mathbf{S}$ is positive definite, which is the case for the datasets *Cities*, *WorldCities*, and *Journals*. In our experiments, the number of clusters $r$ is empirically set to a double of the number of classes as we focus on the multi-way graph partitioning.

**Table 1** Statistics of selected graph datasets

| datasets | #vertices | sparse | $p/N$ | #classes |
|---|---|---|---|---|
| Scotland | 108 | yes | 0.17 | 8 |
| Cities | 46 | no | – | 4 |
| WorldCities | 100 | no | – | 6 |
| Journals | 124 | no | – | 14 |
| DutchElite | 937 | no | 0.66 | 4 |
| Strike | 24 | yes | 0.04 | 3 |
| Sawmill | 36 | yes | 0.03 | 3 |
| Korea | 39 | yes | 0.03 | 2 |
| Football | 115 | yes | 0.01 | 12 |
| A99m | 234 | yes | 0.00 | 3 |

The datasets *Cities*, *WorldCities*, *DutchElite*, and *A99m* are originally given in a rectangular matrix, with each vertex given by a row. We construct the affinity matrix $\mathbf{A}$ by simply computing the normalized dot products, i.e., the cosines, between the row vectors. The raw vectorial data are then discarded after the graph is created.

To demonstrate the benefit brought by the orthonormality constraint, we have compared the nonnegative learning algorithms *NL* (39) and $\sqrt{NL}$ (40) with their orthogonal versions *ONL* (41) and $\sqrt{ONL}$ (42). Furthermore, we also compare two other algorithms for optimizing (36)–(37). One is the *kernel K-means* method (KKmeans) that applies the classical *K-means* algorithm with the Euclidean distance measure in the space $\mathcal{V}$ between the $i$th vertex and the $k$th cluster mean [10]:

$$\mathrm{dist}(i,k) = S_{ii} - \frac{2}{N_k}\sum_{t\in\mathcal{C}_k} S_{it} + \frac{1}{N_k^2}\sum_{s\in\mathcal{C}_k}\sum_{t\in\mathcal{C}_k} S_{st}, \quad (43)$$

where $\mathcal{C}_k$ and $N_k$ are the indices and size of the $k$th partition, respectively. The other is a two-stage approach called *POD* [11]. It first obtains the eigenvectors associated with the largest eigenvalues. Denote $\tilde{\mathbf{W}}$ the matrix composed of the eigenvectors as its columns. POD then finds the reweighted partition indicating matrix $\mathbf{W}$ and a rotation matrix $\mathbf{V}$ such that $\left\|\mathbf{W} - \tilde{\mathbf{W}}\mathbf{V}\right\|_F$ is locally minimized.

For each graph, we ran the KKmeans and POD 50 times and took the $\mathbf{W}^*$ with the largest trace. Next, we initialize $W = W^* + 0.2$ according to Ref. [4] for the four nonnegative learning algorithms. The binary partition

2) http://vlado.fmf.uni-lj.si/pub/networks/data/

3) The $p$ value can be calculated with the MATLAB function for Cholesky factorization.

indicating matrix $\mathbf{C}$ is obtained by taking the maximum along each row of trained nonnegative matrix after 10000 iterations. The objective achieved by each nonnegative learning algorithm is calculated with the reweighted version of $\mathbf{C}$.

The resulting objectives are shown in Table 2. ONL or $\sqrt{\text{ONL}}$ wins seven out of ten among the selected datasets, while POD achieves the best for *Scotland*, *Journals*, and *DutchElite*. We divide the ten datasets into two groups: *Scotland*, *Cities*, *WorldCities*, *Journals*, and *DutchElite* whose $\mathbf{S}$ is positive definite or nearly positive definite, and the other five with $\mathbf{S}$ far from positive definiteness. It is worth to notice that the results achieved by POD, KKmeans, ONL or $\sqrt{\text{ONL}}$ are quite close for the first group, whereas ONL or $\sqrt{\text{ONL}}$ significantly outperform POD or KKmeans for the second group of datasets. That is, orthogonal nonnegative learning is more advantageous for the problem (36)–(37) when the discriminative $K$-means matrix $\mathbf{S}$ is far from positive definiteness, where by contrast POD and KKmeans may even return a negative objective. NL and $\sqrt{\text{NL}}$, the two nonnegative learning algorithms without the orthogonality constraint, work poorly for all datasets. Although they return positive objectives for both dataset groups, their converged objectives are far from the best ones for all graphs.

**Table 2** Discriminative $K$-means objectives using the six compared methods. Boldface numbers are the best (largest) in their rows

| datasets | group | NL | $\sqrt{\text{NL}}$ | ONL | $\sqrt{\text{ONL}}$ | KKmeans | POD |
|---|---|---|---|---|---|---|---|
| Scotland | 1 | 0.18 | 0.18 | 2.64 | 2.65 | 2.06 | **2.66** |
| Cities | 1 | 0.70 | 0.70 | **1.50** | **1.50** | 1.46 | 1.49 |
| WorldCities | 1 | 0.81 | 0.81 | **2.24** | **2.24** | 2.10 | 2.19 |
| Journals | 1 | 0.97 | 0.97 | 27.05 | 27.11 | 27.01 | **27.17** |
| DutchElite | 1 | 0.16 | 0.16 | 4.89 | 4.90 | 3.16 | **4.97** |
| Strike | 2 | 0.23 | 0.23 | **0.98** | **0.98** | 0.33 | 0.17 |
| Sawmill | 2 | 0.23 | 0.23 | **1.01** | **1.01** | −0.10 | −0.46 |
| Korea | 2 | 0.24 | 0.24 | 0.75 | **0.80** | 0.15 | −0.32 |
| Football | 2 | 0.52 | 0.52 | **4.60** | **4.60** | −1.43 | −3.24 |
| A99m | 2 | 0.15 | 0.15 | 2.18 | **2.21** | −0.65 | −1.07 |

Besides the objectives, we also computed the purities of the resulting partitions by using the ground truth class information. The purities are given by

$$\text{purity} = \frac{1}{N}\sum_{k=1}^{r}\max_{1\leqslant l\leqslant q} N_k^l \tag{44}$$

with $N_k^l$ is the number of vertices in the partition $k$ that belong to original class $l$. The resulting purities are shown in Table 3. It can be seen that ONL or $\sqrt{\text{ONL}}$ achieves the best for all selected datasets, where three of ten are tied with POD. The purities confirm that KKmeans, NL and $\sqrt{\text{NL}}$ are not satisfactory for graph partitioning.

**Table 3** Graph partitioning purities using the six compared methods. Boldface numbers are the best (largest) in their rows

| datasets | group | NL | $\sqrt{\text{NL}}$ | ONL | $\sqrt{\text{ONL}}$ | KKmeans | POD |
|---|---|---|---|---|---|---|---|
| Scotland | 1 | 0.20 | 0.20 | 0.46 | **0.48** | 0.36 | 0.47 |
| Cities | 1 | 0.35 | 0.35 | **0.83** | **0.83** | 0.67 | 0.65 |
| WorldCities | 1 | 0.23 | 0.23 | **0.63** | **0.63** | 0.59 | **0.63** |
| Journals | 1 | 0.19 | 0.19 | **0.56** | 0.53 | 0.36 | **0.56** |
| DutchElite | 1 | 0.42 | 0.42 | **0.49** | 0.48 | 0.43 | 0.48 |
| Strike | 2 | 0.46 | 0.46 | **1.00** | **1.00** | 0.79 | **1.00** |
| Sawmill | 2 | 0.50 | 0.50 | **0.58** | **0.58** | 0.53 | 0.56 |
| Korea | 2 | 0.66 | 0.66 | **0.77** | 0.74 | 0.66 | 0.74 |
| Football | 2 | 0.11 | 0.11 | **0.95** | **0.95** | 0.57 | 0.59 |
| A99m | 2 | 0.49 | 0.49 | **0.53** | **0.53** | 0.52 | 0.52 |

## 5 Conclusions

We reviewed some classical linear expansions of data matrices and emphasized how an orthogonal and nonnegative basis of the data vectors would be highly useful. When data vectors are given in this kind of basis, the representation is "part-based" and, because of orthogonality, very sparse. Another advantage of orthogonal nonnegative basis matrices is that they can stand for the binary indicator matrices widely used to define combinatorial optimization problems like clustering or graph partitioning.

We proposed a common principle to derive multiplicative update rules for orthogonal nonnegative learning tasks from the ones without the orthogonality constraint. The ONL principle was mathematically justified by using the Lagrangian multipliers, auxiliary functions and Karush-Kuhn-Tucker conditions. By applying the principle, one can for example obtain the orthogonal versions of NMF and PNMF algorithms in a very straightforward manner.

The principle can be used to develop algorithms for either vectorial or non-vectorial data, with wide applicability. One of the latter applications, graph partitioning by using trace maximization, was demonstrated in this work. The derivation using the ONL principle does not require the symmetric matrix in analysis to be positive definite, which is more favorable in practical use. The derived algorithms using the proposed principle achieve the best or close to the best graph partitioning objective and accuracy for a wide range of graph datasets, which consolidates the advantages of using orthogonal nonnegative learning.

## Appendix A Auxiliary function

The *auxiliary function* method has widely been used for convergence analysis of optimization algorithms

such as the nonnegative multiplicative updates and Expectation-Maximization (EM). Given an objection function $\mathcal{J}(\mathbf{W})$ to be maximized, $G(\mathbf{W}', \mathbf{W})$ is called an auxiliary function if it is a tight lower bound of $\mathcal{J}(\mathbf{W}')$, i.e.,

$$G(\mathbf{W}', \mathbf{W}) \leqslant \mathcal{J}(\mathbf{W}'), \quad G(\mathbf{W}, \mathbf{W}) = \mathcal{J}(\mathbf{W}) \tag{A.1}$$

for any $\mathbf{W}$ and $\mathbf{W}'$. Define

$$\mathbf{W}' = \arg\max_{\tilde{\mathbf{W}}} G\left(\tilde{\mathbf{W}}, \mathbf{W}\right). \tag{A.2}$$

By construction,

$$\begin{aligned} \mathcal{J}(\mathbf{W}) = G(\mathbf{W}, \mathbf{W}) &\leqslant G(\mathbf{W}', \mathbf{W}) \\ &\leqslant G(\mathbf{W}', \mathbf{W}') = \mathcal{J}(\mathbf{W}'), \end{aligned} \tag{A.3}$$

where the left inequality is the result of maximization and the right one comes from the lower bound. Iteratively applying the update rule (A.2) thus results in a monotonically increasing sequence of $\mathcal{J}$. Besides the tight lower bound, it is often desired that the maximization (A.2) has a closed-form solution. In particular, setting $\partial G/\partial \mathbf{W}' = 0$ should lead to the iterative update rule in analysis.

## Appendix B Auxiliary function for PNMF

The PNMF objective function $\mathcal{J}(\mathbf{W}') = -\frac{1}{2}\|\mathbf{X} - \mathbf{W}'\mathbf{W}'^{\mathrm{T}}\mathbf{X}\|_F^2$ can be rewritten as

$$\mathcal{J}(\mathbf{W}') = \frac{1}{2}\mathrm{Tr}\left(\mathbf{W}'^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}'\right) - \frac{1}{2}\mathrm{Tr}\left(\mathbf{W}'\mathbf{W}'^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}'\mathbf{W}'^{\mathrm{T}} - \mathbf{W}'^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}'\right) \tag{B.1}$$

and tightly lower bounded by

$$G(\mathbf{W}', \mathbf{W}) = \frac{1}{2}\sum_{abk}\left(\mathbf{X}\mathbf{X}^{\mathrm{T}}\right)_{ab} W_{ak}W_{bk}\left(1 + \log\frac{W'_{ak}W'_{bk}}{W_{ak}W_{bk}}\right) - \frac{1}{2}\mathrm{Tr}\left(\mathbf{W}'\mathbf{W}'^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}'\mathbf{W}'^{\mathrm{T}} - \mathbf{W}'^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{W}'\right). \tag{B.2}$$

Setting $\partial G(\mathbf{W}', \mathbf{W})/\partial W'_{ik} = 0$, we obtain the multiplicative update rule (29).

## Appendix C Auxiliary function for trace maximization

The trace objective

$$\mathcal{J}(\mathbf{W}') = \mathrm{Tr}\left(\mathbf{W}'^{\mathrm{T}}\mathbf{S}^{+}\mathbf{W}'\right) - \mathrm{Tr}\left(\mathbf{W}'^{\mathrm{T}}\mathbf{S}^{-}\mathbf{W}'\right) \tag{C.1}$$

can be tightly lower bounded by

$$G(\mathbf{W}', \mathbf{W}) = \sum_{abk} S^{+}_{ab}W_{ak}W_{bk}\left(1 + \log\frac{W'_{ak}W'_{bk}}{W_{ak}W_{bk}}\right) \tag{C.2}$$

$$-\mathrm{Tr}\left(\mathbf{W}'^{\mathrm{T}}\mathbf{S}^{-}\mathbf{W}'\right) \tag{C.3}$$

or

$$F(\mathbf{W}', \mathbf{W}) = \sum_{abk} S^{+}_{ab}W_{ak}W_{bk}\left(1 + \log\frac{W'_{ak}W'_{bk}}{W_{ak}W_{bk}}\right) \tag{C.4}$$

$$-\sum_{ik}\frac{\left(\mathbf{S}^{-}\mathbf{W}\right)_{ik}W'^{\,2}_{ik}}{W_{ik}}. \tag{C.5}$$

Setting $\partial G(\mathbf{W}', \mathbf{W})/\partial W'_{ik} = 0$ leads to the multiplicative update rule (39) and $\partial F(\mathbf{W}', \mathbf{W})/\partial W'_{ik} = 0$ to (40).

## References


1. Paatero P, Tapper U. Positive matrix factorization: A nonnegative factor model with optimal utilization of error estimates of data values. Environmetrics, 1994, 5(2): 111–126
2. Lee D D, Seung H S. Learning the parts of objects by nonnegative matrix factorization. Nature, 1999, 401(6755): 788–791
3. Cichocki A, Zdunek R, Phan A H, Amari S-I. Nonnegative Matrix and Tensor Factorizations. Singapore: Wiley, 2009
4. Ding C, Li T, Peng W, Park H. Orthogonal nonnegative matrix t-factorizations for clustering. In: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2006, 126–135
5. Yang Z, Laaksonen J. Multiplicative updates for nonnegative projections. Neurocomputing, 2007, 71(1–3): 363–373
6. Choi S. Algorithms for orthogonal nonnegative matrix factorization. In: Proceedings of IEEE International Joint Conference on Neural Networks. 2008, 1828–1832
7. Ding C, He X, Simon H D. On the equivalence of nonnegative matrix factorization and spectral clustering. In: Proceedings of SIAM International Conference of Data Mining. 2005, 606–610
8. Yuan Z, Oja E. Projective nonnegative matrix factorization for image compression and feature extraction. In: Proceedings of the 14th Scandinavian Conference on Image Analysis (SCIA 2005). Joensuu, Finland, 2005, 333–342
9. Oja E. Principal components, minor components, and linear neural networks. Neural Networks, 1992, 5(6): 927–935
10. Dhillon I, Guan Y, Kulis B. Kernel $k$-means, spectral clustering and normalized cuts. In: Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Seattle, WA, USA, 2004, 551–556
11. Yu S X, Shi J. Multiclass spectral clustering. In: Proceedings of the 9th IEEE International Conference on Computer Vision. 2003, 2: 313–319

12. Jolliffe I T. Principal Component Analysis. Berlin: Springer-Verlag, 2002
13. Diamantaras K, Kung S Y. Principal Component Neural Networks. New York: Wiley, 1996
14. Oja E. Subspace Methods of Pattern Recognition. Letchworth: Research Studies Press, 1983
15. Yang Z, Oja E. Linear and nonlinear projective nonnegative matrix factorization. IEEE Transactions on Neural Networks, 2009, accepted, to appear
16. Lloyd S P. Least square quantization in PCM. IEEE Transactions on Information Theory, 1982, 28(2): 129–137
17. Lee D D, Seung H S. Algorithms for non-negative matrix factorization. In: Proceedings of the Conference on Advances in Neural information Processing Systems. 2000, 556–562
18. Hoyer P O. Non-negative matrix factorization with sparseness constraints. Journal of Machine Learning Research, 2004, 5: 1457–1469
19. Xu W, Liu X, Gong Y. Document clustering based on nonnegative matrix factorization. In: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval. 2003, 267–273
20. Févotte C, Bertin N, Durrieu J-L. Nonnegative matrix factorization with the itakura-saito divergence: With application to music analysis. Neural Computation, 2009, 21(3): 793–830
21. Drakakis K, Rickard S, de Fréin R, Cichocki A. Analysis of financial data using non-negative matrix factorization. International Mathematical Forum, 2008, 3: 1853–1870
22. Young S S, Fogel P, Hawkins D M. Clustering scotch whiskies using non-negative matrix factorization. Joint Newsletter for the Section on Physical and Engineering Sciences and the Quality and Productivity Section of the American Statistical Association, 2006, 14(1): 11–13
23. Brunet J-P, Tamayo P, Golub T R, Mesirov J P. Metagenes and molecular pattern discovery using matrix factorization. Proceedings of the National Academy of Sciences of the United States of America, 2004, 101(12): 4164–4169
24. Ding C, Li T, Jordan M I. Convex and semi-nonnegative matrix factorizations. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2010, 32(1): 45–55
25. Sha F, Saul L K, Lee D D. Multiplicative updates for large margin classifiers. In: Proceedings of the 16th Annual Conference on Learning Theory and the 7th Kernel Workshop, COLT. 2003, 188–202
26. Cichocki A, Lee H, Kim Y-D, Choi S. Non-negative matrix factorization with $\alpha$-divergence. Pattern Recognition Letters, 2008, 29(9): 1433–1440
27. Kivinen J, Warmuth M K. Exponentiated gradient versus gradient descent for linear predictors. Information and Computation, 1997, 132(1): 1–63
28. Yang Z, Yuan Z, Laaksonen J. Projective non-negative matrix factorization with applications to facial image processing. International Journal of Pattern Recognition and Artificial Intelligence, 2007, 21(8): 1353–1362
29. Chung F R K. Spectral Graph Theory. American Methematical Society, 1997
30. Newman M E J. Finding community structure in networks using the eigenvectors of matrices. Physical Review E, 2006, 74(3): 036104
31. Leicht E A, Holme P, Newman M E J. Vertex similarity in networks. Physical Review E, 2006, 73(2): 026120
32. Ye J, Zhao Z, Wu M. Discriminative $K$-means for clustering. In: Proceedings of the Conference on Advances in Neural Information Processing Systems. 2007, 1649–1656

Erkki Oja is Director of the Adaptive Informatics Research Centre and Professor of Computer Science at the Department of Information and Computer Science, Aalto University, Finland. He received his Dr. Sc. degree in 1977 and a honorary doctorate from Uppsala University, Sweden, in 2008. He has been post-doctoral research associate at Brown University, Providence, RI, and visiting professor at Tokyo Institute of Technology and Universite Paris I — Pantheon Sorbonne. Dr. Oja is the author or coauthor of more than 300 articles and book chapters on pattern recognition, image and signal processing, and machine learning, as well as three books: *Subspace Methods of Pattern Recognition* (RSP and J. Wiley, 1983), which has been translated into Chinese and Japanese, *Kohonen Maps* (Elsevier, 1999), and *Independent Component Analysis* (J. Wiley, 2001; Japanese translation, 2005; Chinese translation, 2006).

Dr. Oja's research interests are in the study of statistical signal processing, pattern recognition, and data analysis. He is member of the editorial boards of several journals and program committees of several recent and upcoming conferences including ICANN, IJCNN, NIPS, and ICONIP. He is Chairman of the Research Council for Natural Sciences and Engineering of the Academy of Finland, member of the Finnish Academy of Sciences, Fellow of the IEEE, Founding Fellow of the International Association of Pattern Recognition (IAPR), Fellow of the INNS, and Past President of the European Neural Network Society (ENNS). He was recipient of the 2004 IAPR Pierre Devijver Award and the 2006 IEEE Computational Intelligence Society Pioneer Award.