Nonnegative Matrix Approximation: Algorithms and Applications

Suvrit Sra Inderjit S. Dhillon

Dept of Computer Sciences
University of Texas at Austin
Austin, TX 78712, USA

21 June, 2006

Technical Report # TR-06-27

Abstract

Low dimensional data representations are crucial to numerous applications in machine learning, statistics, and signal processing. Nonnegative matrix approximation (NNMA) is a method for dimensionality reduction that respects the nonnegativity of the input data while constructing a low-dimensional approximation. NNMA has been used in a multitude of applications, though without commensurate theoretical development. In this report we describe generic methods for minimizing generalized divergences between the input and its low rank approximant. Some of our general methods are even extensible to arbitrary convex penalties. Our methods yield efficient multiplicative iterative schemes for solving the proposed problems. We also consider interesting extensions such as the use of penalty functions, non-linear relationships via "link" functions, weighted errors, and multi-factor approximations. We present some experiments as an illustration of our algorithms. For completeness, the report also includes a brief literature survey of the various algorithms and the applications of NNMA.

Keywords:
Nonnegative matrix factorization, weighted approximation, Bregman divergence, multiplicative updates, link functions, dimensionality reduction.

1 Introduction

A suitable representation of data is central to applications in fields such as machine learning, statistics, and signal processing. The manner in which data are represented determines the course of subsequent processing and analysis, be it pattern recognition, denoising, visualization, compression or anything else. Representation is crucial, for example consider face recognition; to a human viewer a picture makes vastly more sense than an array of numbers, though for computation the latter might be more preferable.

A useful representation has two primary desiderata. First, an amenability to interpretation and second, computational feasibility. Central to obtaining useful representations is the process of dimensionality reduction, wherein one constructs a lower complexity representation of the input data. The reduced dimensionality offers advantages such as denoising, computational efficiency, greater interpretability and easier visualization, among others. While performing dimensionality reduction for inherently nonnegative data such as color intensities, chemical concentrations, frequency counts etc., it makes sense to respect the nonnegativity to

avoid physically absurd and uninterpretable results. This viewpoint has both computational as well as philosophical underpinnings. For example, for the sake of interpretation one would prefer to draw representatives from the same space (or a subspace thereof) as that of the input data. Computationally, nonnegativity leads to a sparser approximation, which in turn facilitates more efficient subsequent processing.

These considerations bring us to the problem of *nonnegative matrix approximation:* Given a set of nonnegative inputs find a small set of nonnegative representative vectors whose nonnegative combinations approximate the input data. That is, given a set $\{\boldsymbol{a}_i : \boldsymbol{a}_i \in \mathbb{R}^M_+,\, 1 \leq i \leq N\}$ of nonnegative inputs, we wish to compute vectors $\boldsymbol{b}_k \in \mathbb{R}^M_+$ and coefficients $c_{kn} \in \mathbb{R}_+$ so that

$$\boldsymbol{a}_n \approx \sum_{k=1}^{K} c_{kn}\boldsymbol{b}_k, \quad 1 \leq n \leq N.$$

Gathering the vectors and coefficients into matrices, this approximation may be written as

$$\boldsymbol{A}_{M\times N} \approx \boldsymbol{B}_{M\times K}\boldsymbol{C}_{K\times N}, \qquad \text{where } \boldsymbol{B}, \boldsymbol{C} \geq 0. \tag{1.1}$$

We remark that by imposing varying constraints on the matrices $\boldsymbol{B}$ and $\boldsymbol{C}$ one can obtain many different problems. For example, when either $\boldsymbol{B}$ or $\boldsymbol{C}$ is unconstrained and one measures approximation errors using any unitarily invariant norm such as the L_2-norm, then (1.1) leads to the truncated singular value decomposition (TSVD). Other measures of approximation error lead to related problems (see [Collins, Dasgupta, and Schapire, 2001], for example). By varying the constraints on $\boldsymbol{B}$ and $\boldsymbol{C}$ one obtains various important problems such as clustering (see [Tropp, 2004, Chapter 8]) and probabilistic latent semantic indexing [Hofmann, 1999], for example.

1.1 Main contributions

This report makes the following main contributions[1].

1. It develops algorithms for minimizing Bregman divergences between the input and its low dimensional approximant. New algorithms as well as details of the derivations for our previous algorithms are included. Our approach is not restricted to merely Bregman divergences, but extensible (in many cases) to arbitrary convex losses. The report discusses extensions to Csiszár's and Young's divergences to illustrate this strength.

2. It presents proofs of convergence for many of the main algorithms, including new proofs of convergence for the Frobenius norm, KL-Divergence and Burg-Entropy based NNMA problems. These proofs demonstrate that the objective function decreases monotonically with each iteration of the algorithm, and since the objective functions are bounded below, one obtains convergence to a fixed point of the objective function.

3. It includes discussion about the use of penalty and "link" functions for NNMA problems. Penalty functions permit one to enforce additional constraints on $\boldsymbol{B}$ and $\boldsymbol{C}$, while link functions allow one to model nonlinear relations such as $\boldsymbol{A} \approx h(\boldsymbol{BC})$. We capitalize on the power of link functions to obtain a new provably convergent algorithm for minimizing $D_\varphi(\boldsymbol{A}; \boldsymbol{BC})$.

4. It derives a few example NNMA problems as special cases to illustrate the power of our methods. Further, it also includes examples showing extension to the multi-factor and weighted NNMA problems—both of which can be useful in many applications.

[1] Preliminary work appeared as [Dhillon and Sra, 2006].

5. It provides a brief literature review to indicate the vast scope and applicability of NNMA. This review includes a large list of references and it can be useful to other researchers in the area. Further, it is our hope that our new algorithms and techniques find use in the numerous applications reviewed.

6. Finally for completeness, for the interested reader, the report includes a bonus section that offers a brief summary of the nonnegative matrix factorization (not approximation) problem, thereby justifying our choice of terminology.

Note: Optimized software written in C++, using BLAS libraries accompanies this report and may be obtained from the following website: *http://www.cs.utexas.edu/~suvrit/work/progs/nnma.html*.

1.2 Summary of the remainder

The rest of the document is organized as follows. Section 2 gives a formal definition of the two main problems to be solved in this report. A large fraction of the theoretical work of this report is contained in Section 3, which derives algorithms for solving Problem (P1) in §3.1, and Problem (P2) in §3.2. Penalty functions are discussed in §§3.1.4 and 3.2.7, while link functions are the topic of §3.1.5. Section 3.3 provides generalizations to other convex penalties as an illustration of the wider applicability of our methods.

Section 4 shows a number of examples that are special cases of our general formulation, including KL-Divergence, constrained, weighted, and multi-factor NNMA problems. Table 2 summarizes many of the algorithms described in this document. Section 5 shows some experimental results to illustrate the behavior of some of our NNMA algorithms. To complement the experiments and theory we include a brief literature review in Section 6 that covers most known algorithms (§6.1) and applications (§6.2) of NNMA. Section 7 makes a minor excursus into the problem of nonnegative matrix factorization problem.

2 Problem formulation

Given a nonnegative matrix $\boldsymbol{A} = [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N]$ as input, the classical NNMA problem seeks to approximate it by a lower rank nonnegative matrix of the form $\boldsymbol{BC}$, where $\boldsymbol{B} = [\boldsymbol{b}_1, ..., \boldsymbol{b}_K]$ and $\boldsymbol{C} = [\boldsymbol{c}_1, ..., \boldsymbol{c}_N]$ are themselves nonnegative. That is, we seek the approximation

$$\boldsymbol{A}_{M\times N} \approx \boldsymbol{B}_{M\times K}\boldsymbol{C}_{K\times N}, \qquad \text{where } \boldsymbol{B}, \boldsymbol{C} \geq 0. \tag{2.1}$$

For estimating the matrices $\boldsymbol{B}$ and $\boldsymbol{C}$ we measure the quality of approximation in (2.1) by using a general class of distortion measures called *Bregman divergences*.

2.1 Bregman divergences

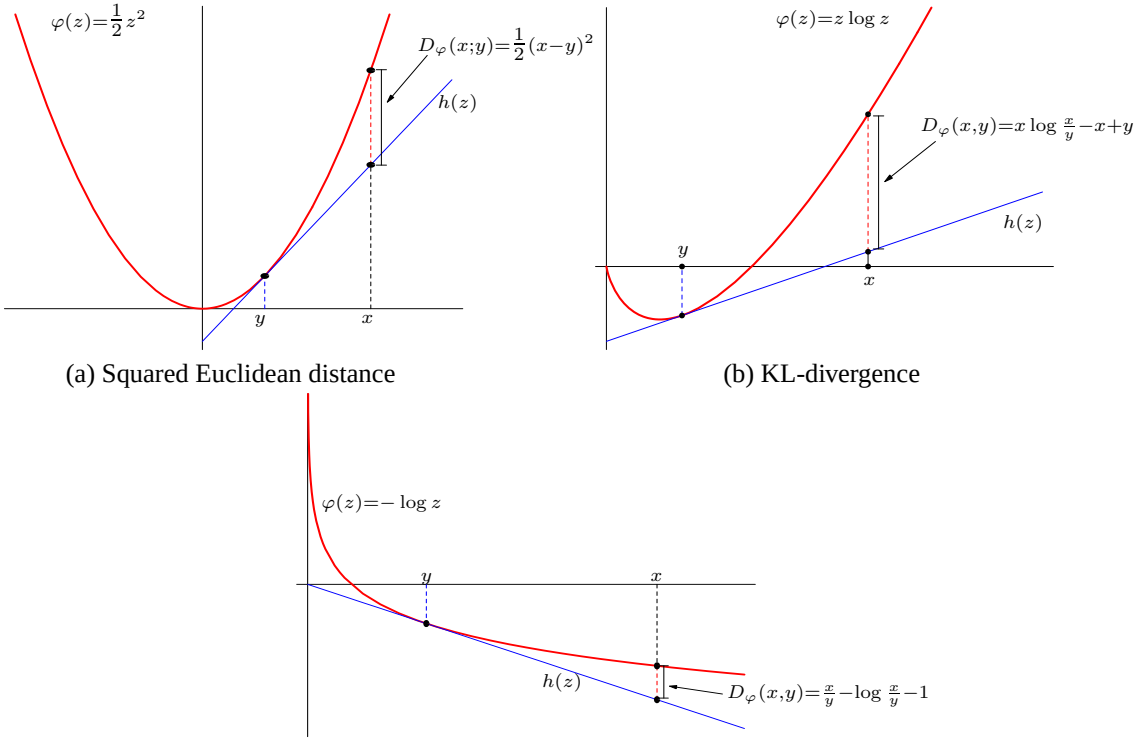
For any strictly convex function $\varphi : S \subseteq \mathbb{R} \to \mathbb{R}$ that has a continuous first derivative, the corresponding **Bregman divergence** $D_\varphi : S \times \text{int}(S) \to \mathbb{R}_+$ is defined as

$$D_\varphi(x; y) \triangleq \varphi(x) - \varphi(y) - \varphi'(y)(x - y), \tag{2.2}$$

where $\text{int}(S)$ is the interior of set S [Censor and Zenios, 1997]. Bregman divergences enjoy many useful properties. For example, they are nonnegative, convex in the first argument and zero if and only if $x = y$. The sum of two Bregman divergences is also a Bregman divergence, hence, we can extend the definition to matrix (elementwise) arguments, so that

$$D_\varphi(\boldsymbol{X}; \boldsymbol{Y}) \triangleq \sum_{ij} D_\varphi(x_{ij}; y_{ij}),$$

with the implicit assumption that $x_{ij}, y_{ij} \in \text{dom}\varphi \cap \mathbb{R}_+$. The well known Euclidean distance, the information theoretic KL-Divergence (unnormalized), and the Itakuro-Saito distance are examples of particular Bregman divergences, illustrated respectively by Figures 1(a), 1(b), and 1(c).



(a) Squared Euclidean distance

(b) KL-divergence

(c) Itakura-Saito distance

Figure 1: Three particular Bregman divergences

Bregman divergences play an important role in convex optimization—they were defined by Bregman [1967] in the context of minimizing a strictly convex function subject to linear inequality constraints. Recently these divergences have also been applied in to clustering [Banerjee et al., 2004b] and co-clustering problems [Banerjee et al., 2004a]. The definition of Bregman divergences can be extended to matrices in a non-elementwise manner [Bauschke and Borwein, 1997]—this extension has been applied to the problem of kernel learning [Kulis et al., 2006, Tsuda et al., 2005].

2.2 The Problems

We focus on separable Bregman divergences of the type described above. The two main generalized NNMA problems that we discuss are

$$\min_{\boldsymbol{B},\boldsymbol{C}\geq 0} \quad D_\varphi(\boldsymbol{BC};\boldsymbol{A}) + \alpha(\boldsymbol{B}) + \beta(\boldsymbol{C}), \quad \text{and} \tag{P1}$$

$$\min_{\boldsymbol{B},\boldsymbol{C}\geq 0} \quad D_\varphi(\boldsymbol{A};\boldsymbol{BC}) + \alpha(\boldsymbol{B}) + \beta(\boldsymbol{C}). \tag{P2}$$

The functions α and β are *penalty* functions, and they allow us to enforce regularization (or other constraints) on $\boldsymbol{B}$ and $\boldsymbol{C}$. We consider both (P1) and (P2) since Bregman divergences are usually asymmetric; further-

more, each version leads to interesting algorithms with differing characteristics. Our formulation is quite general as may be discerned from Table 1, which illustrates how some previously studied NNMA problems turn out to be special cases.

Divergence D_φ	φ	α	β	Remarks
$\|\boldsymbol{A}-\boldsymbol{BC}\|_{\mathrm{F}}^2$	$\frac{1}{2}x^2$	$\mathbf{0}$	$\mathbf{0}$	Lee and Seung [1999, 2000]
$\|\boldsymbol{A}-\boldsymbol{BC}\|_{\mathrm{F}}^2$	$\frac{1}{2}x^2$	$\mathbf{0}$	$\lambda\mathbf{1}^T\boldsymbol{C}\mathbf{1}$	Hoyer [2002]
$\|\boldsymbol{W}\odot(\boldsymbol{A}-\boldsymbol{BC})\|_{\mathrm{F}}^2$	$\frac{1}{2}x^2$	$\mathbf{0}$	$\mathbf{0}$	Paatero et al. [1991]
$\mathrm{KL}(\boldsymbol{A};\boldsymbol{BC})$	$x\log x$	$\mathbf{0}$	$\mathbf{0}$	Lee and Seung [2000]
$\mathrm{KL}(\boldsymbol{A};\boldsymbol{WBC})$	$x\log x$	$\mathbf{0}$	$\mathbf{0}$	Guillamet et al. [2001]
$\mathrm{KL}(\boldsymbol{A};\boldsymbol{BC})$	$x\log x$	$c_1\mathbf{1}^T\boldsymbol{B}^T\boldsymbol{B}\mathbf{1}$	$-c_2\|\boldsymbol{C}\|_{\mathrm{F}}^2$	Feng et al. [2002]

Table 1: Known NNMA problems that may be obtained from (P2). KL(x, y) denotes the generalized KL-Divergence = $\sum_i x_i\log\frac{x_i}{y_i}-x_i+y_i$ (also called I-divergence).

Later, in this report (§3.3) we will also describe the NNMA problem for two other generalized divergences, namely Csiszár's φ-divergence, and Young's divergence. However, the theoretical ideas for tackling these two will be the same as those developed below for solving (P1) and (P2).

3 Methods for solution

In this section we develop generic methods for obtaining efficient iterative algorithms for solving problems (P1) and (P2). We study Problem (P1) first, as it turns out to be simpler than (P2). We remark that this simplicity is principally due to the convexity of Bregman divergences in their first argument. Section 3.2 discusses methods for solving (P2).

Note that the problems (P1) and (P2) are not jointly convex in $\boldsymbol{B}$ and $\boldsymbol{C}$ simultaneously, making it hard to obtain globally optimal solutions. Our iterative procedures initialize $\boldsymbol{B}$ and $\boldsymbol{C}$ randomly and then alternately update them until there is no further appreciable change in the objective function, yielding locally optimal solutions. Additional initialization strategies could also be used, however, we do not pursue them in this report.

This section includes algorithms for performing the simple multiplicative update of the form $\boldsymbol{c}\leftarrow\eta\boldsymbol{c}$. We show the derivations for both the main NNMA problems.

3.1 Algorithms for (P1)

We derive below a method that yields multiplicative updates for Problem (P1). Since the divergences that we treat are separable, we illustrate our method using a single column of $\boldsymbol{C}$ (or a row of $\boldsymbol{B}$). Explicitly, $D_\varphi(\boldsymbol{BC};\boldsymbol{A})=\sum_j D_\varphi(\boldsymbol{Bc}_j;\boldsymbol{a}_j)$, where $\boldsymbol{c}_j$ and $\boldsymbol{a}_j$ are corresponding columns of $\boldsymbol{C}$ and $\boldsymbol{A}$. Let $F(\boldsymbol{c})$ denote $D_\varphi(\boldsymbol{Bc};\boldsymbol{a})$ for arbitrary columns $\boldsymbol{c}$ and $\boldsymbol{a}$. A multiplicative update for $\boldsymbol{c}$ may be written as

$$c_i\leftarrow\eta_i c_i,\quad \text{where } \eta_i\in\mathbb{R}_+,\quad \text{and } 1\le i\le N. \tag{3.1}$$

A particularly simple special case arises if all the η_i values are selected to be the same. In the derivations below, for simplicity we initially assume $\alpha(\boldsymbol{B})$ and $\beta(\boldsymbol{C})$ to be zero. We also use $\psi(x)$ to denote $\varphi'(x)$.

3.1.1 Simple multiplicative updates

Given $F(\boldsymbol{c}) = D_\varphi(\boldsymbol{B}\boldsymbol{c}; \boldsymbol{a})$, we wish to compute a factor η so that $F(\eta\boldsymbol{c}) \leq F(\boldsymbol{c})$, hence ensuring a monotonic decrease in the objective function. If we set

$$\eta^* = \operatorname*{argmin}_\eta F(\eta\boldsymbol{c}),$$

and update $\boldsymbol{c} \leftarrow \eta^*\boldsymbol{c}$, then clearly $F(\eta^*\boldsymbol{c}) \leq F(\boldsymbol{c})$. Differentiating $F(\eta\boldsymbol{c})$ w.r.t. η and setting the derivative to zero we have

$$\sum_i (\boldsymbol{B}\boldsymbol{c})_i \psi(\eta(\boldsymbol{B}\boldsymbol{c})_i) - \psi(a_i)(\boldsymbol{B}\boldsymbol{c})_i = 0. \tag{3.2}$$

Assuming that[2] $\psi(xy) = \psi(x)\psi(y)$, we solve (3.2) to obtain the minimum (since $F''(\eta\boldsymbol{c}) \geq 0$),

$$\eta^* = \psi^{-1}\left[\frac{\boldsymbol{c}^T\boldsymbol{B}^T\psi(\boldsymbol{a})}{\boldsymbol{c}^T\boldsymbol{B}^T\psi(\boldsymbol{B}\boldsymbol{c})}\right].$$

We derive the update factor for a given row of $\boldsymbol{B}$ in a similar way. Thus, we get the following iterative scheme (for each row $\boldsymbol{b}$ of $\boldsymbol{B}$ and column $\boldsymbol{c}$ of $\boldsymbol{C}$)

$$\boldsymbol{b} \leftarrow \psi^{-1}\left[\frac{\psi(\boldsymbol{a}^T)\boldsymbol{C}^T\boldsymbol{b}}{\psi(\boldsymbol{b}^T\boldsymbol{C})\boldsymbol{C}^T\boldsymbol{b}}\right]\boldsymbol{b} \tag{3.3}$$

$$\boldsymbol{c} \leftarrow \psi^{-1}\left[\frac{\boldsymbol{c}^T\boldsymbol{B}^T\psi(\boldsymbol{a})}{\boldsymbol{c}^T\boldsymbol{B}^T\psi(\boldsymbol{B}\boldsymbol{c})}\right]\boldsymbol{c}. \tag{3.4}$$

It can be proved that using these updates the algorithm terminates after two iterations. Hence, it should not be used independently, but instead in conjunction (using a hybrid approach) with the elementwise scaling (3.7).

3.1.2 Improved multiplicative updates

Evidently the updates (3.3) and (3.4) are overly restrictive, and therefore not very desirable. Hence, we focus on the case where each element c_i is scaled separately. We use the concept of auxiliary functions [Collins et al., 2000, Lee and Seung, 2000] to obtain provably convergent multiplicative updates for $\boldsymbol{c}$ (and $\boldsymbol{b}$) below.

Definition 1 (Auxiliary function). *A function $G(\boldsymbol{c}, \boldsymbol{c}')$ is called an auxiliary function for $F(\boldsymbol{c})$ if:*

1. $G(\boldsymbol{c}, \boldsymbol{c}) = F(\boldsymbol{c})$, *and*
2. $G(\boldsymbol{c}, \tilde{\boldsymbol{c}}) \geq F(\boldsymbol{c})$ *for all* $\tilde{\boldsymbol{c}}$.

Auxiliary functions turn out to be useful primarily due to the following lemma.

Lemma 2 (Iterative minimization). *If $G(\boldsymbol{c}, \tilde{\boldsymbol{c}})$ is an auxiliary function for $F(\boldsymbol{c})$, then F is non-increasing under the update*

$$\boldsymbol{c}^{t+1} = \operatorname{argmin}_{\boldsymbol{c}} G(\boldsymbol{c}, \boldsymbol{c}^t).$$

Proof. $F(\boldsymbol{c}^{t+1}) \leq G(\boldsymbol{c}^{t+1}, \boldsymbol{c}^t) \leq G(\boldsymbol{c}^t, \boldsymbol{c}^t) = F(\boldsymbol{c}^t)$.

Given some initial $\boldsymbol{c}^0$, we can iteratively apply Lemma 2 (with changing $\boldsymbol{B}$) to obtain a sequence $\{\boldsymbol{c}^t\}$ for which $F(\boldsymbol{c}^0) \geq F(\boldsymbol{c}^1) \geq \cdots \geq F(\boldsymbol{c}^{t+1})$ holds. Since F is bounded below, the sequence $\{\boldsymbol{c}^t\}$ converges to a stationary point of F.

[2] More generally, we could assume that $\psi(xy) = \psi_1(x)\psi_2(y)$, i.e., ψ is *factorizable*.

Lemma 3 (Auxiliary function). *The function*

$$G(\boldsymbol{c},\tilde{\boldsymbol{c}}) = \sum_{ij} \lambda_{ij}\varphi\Big(\frac{b_{ij}c_j}{\lambda_{ij}}\Big) - \Big(\sum_i \varphi(a_i) + \psi(a_i)\big((\boldsymbol{B}\boldsymbol{c})_i - a_i\big)\Big), \tag{3.5}$$

with $\lambda_{ij} = (b_{ij}\tilde{c}_j)/(\sum_l b_{il}\tilde{c}_l)$*, is an auxiliary function for* $F(\boldsymbol{c})$*.*

Proof. It is easy to verify that $G(\boldsymbol{c},\boldsymbol{c}) = F(\boldsymbol{c})$. Since $\sum_j \lambda_{ij} = 1$ and $\lambda_{ij} \geq 0$, using the convexity of φ we find that

$$\begin{aligned} F(\boldsymbol{c}) = &\quad \sum_i \varphi\big(\sum_j b_{ij}c_j\big) - \Big(\sum_i \varphi(a_i) + \psi(a_i)\big((\boldsymbol{B}\boldsymbol{c})_i - a_i\big)\Big) \\ \leq &\quad \sum_{ij} \lambda_{ij}\varphi\Big(\frac{b_{ij}c_j}{\lambda_{ij}}\Big) - \Big(\sum_i \varphi(a_i) + \psi(a_i)\big((\boldsymbol{B}\boldsymbol{c})_i - a_i\big)\Big) \\ = &\quad G(\boldsymbol{c},\tilde{\boldsymbol{c}}). \end{aligned}$$

□

Note that we manipulated only the first term of $F(\boldsymbol{c})$. Contributions from the other terms could also be involved, yielding different auxiliary functions.

To obtain an update for $\boldsymbol{c}$, we minimize $G(\boldsymbol{c},\tilde{\boldsymbol{c}})$ with respect to $\boldsymbol{c}$. Let $\psi(\boldsymbol{x})$ denote the vector $[\psi(x_1),\ldots,\psi(x_n)]^T$. The partial derivative of G w.r.t. c_p is

$$\begin{aligned} \frac{\partial G}{\partial c_p} &= \sum_i \lambda_{ip}\psi\Big(\frac{b_{ip}c_p}{\lambda_{ip}}\Big)\frac{b_{ip}}{\lambda_{ip}} - \sum_i b_{ip}\psi(a_i) \\ &= \sum_i b_{ip}\psi\Big(\frac{c_p}{\tilde{c}_p}(\boldsymbol{B}\tilde{\boldsymbol{c}})_i\Big) - (\boldsymbol{B}^T\psi(\boldsymbol{a}))_p. \end{aligned} \tag{3.6}$$

We need to solve (3.6) for c_p by setting $\partial G/\partial c_p = 0$. Solving this equation analytically is not always possible, though in principle we could solve it iteratively in such cases. Let us look at one particular class of functions for which we can obtain an analytic solution. For example, if ψ is multiplicative[3], i.e., $\psi(xy) = \psi(x)\psi(y)$, then we may solve $\partial G/\partial c_p = 0$ as follows,

$$\begin{aligned} \frac{\partial G}{\partial c_p} &= \sum_i b_{ip}\psi\Big(\frac{c_p}{\tilde{c}_p}(\boldsymbol{B}\tilde{\boldsymbol{c}})_i\Big) - (\boldsymbol{B}^T\psi(\boldsymbol{a}))_p = 0 \\ &\Longrightarrow \sum_i b_{ip}\psi\Big(\frac{c_p}{\tilde{c}_p}\Big)\psi\big((\boldsymbol{B}\tilde{\boldsymbol{c}})_i\big) - (\boldsymbol{B}^T\psi(\boldsymbol{a}))_p = 0 \\ &\Longrightarrow \psi\Big(\frac{c_p}{\tilde{c}_p}\Big)[\boldsymbol{B}^T\psi(\boldsymbol{B}\tilde{\boldsymbol{c}})]_p = [\boldsymbol{B}^T\psi(\boldsymbol{a})]_p \\ &\Longrightarrow \psi\Big(\frac{c_p}{\tilde{c}_p}\Big) = \frac{[\boldsymbol{B}^T\psi(\boldsymbol{a})]_p}{[\boldsymbol{B}^T\psi(\boldsymbol{B}\tilde{\boldsymbol{c}})]_p}. \end{aligned}$$

Thus, we obtain the update

$$c_p \leftarrow \tilde{c}_p \cdot \psi^{-1}\Big(\frac{[\boldsymbol{B}^T\psi(\boldsymbol{a})]_p}{[\boldsymbol{B}^T\psi(\boldsymbol{B}\tilde{\boldsymbol{c}})]_p}\Big). \tag{3.7}$$

[3]More generally we could consider functions ψ that are *factorizable*, i.e., $\psi(xy) = \psi_1(x)\psi_2(y)$.

Similarly, we may compute the updates for $\boldsymbol{B}$ one row at a time. Let $\boldsymbol{b}$ denote a row of $\boldsymbol{B}$ and $\boldsymbol{a}$ the corresponding row of $\boldsymbol{A}$. The objective function for this row is

$$H(\boldsymbol{b}) = D_\varphi(\boldsymbol{b}^T\boldsymbol{C}; \boldsymbol{a}^T) = \sum_j D_\varphi(\boldsymbol{b}^T\boldsymbol{c}_j; a_j),$$

where $\boldsymbol{c}_j$ denotes the j-th column of $\boldsymbol{C}$, and a_j denotes the j-th component of the row vector $\boldsymbol{a}^T$. Using the convexity of φ we define an the auxiliary function $K(\boldsymbol{b}, \tilde{\boldsymbol{b}})$ for $H(\boldsymbol{b})$, where

$$K(\boldsymbol{b}, \tilde{\boldsymbol{b}}) = \sum_{jk} \mu_{kj}\varphi\Big(\frac{c_{kj}b_k}{\mu_{kj}}\Big) - \sum_j \varphi(a_j) - \psi(a_j)(\boldsymbol{b}^T\boldsymbol{c}_j - a_j),$$

$\mu_{kj} = c_{kj}\tilde{b}_k/(\sum_l c_{lj}\tilde{b}_l)$, and $\mu_{kl} \geq 0$. As before, we may solve $\partial K/\partial b_p = 0$ to obtain the update (for multiplicative ψ)

$$b_p \leftarrow \tilde{b}_p \cdot \psi^{-1}\Big(\frac{[\psi(\boldsymbol{a}^T)\boldsymbol{C}^T]_p}{[\psi(\tilde{\boldsymbol{b}}^T\boldsymbol{C})\boldsymbol{C}^T]_p}\Big). \tag{3.8}$$

3.1.3 Remarks and observations

1. When φ is a convex function of Legendre type, then ψ^{-1} can be obtained by the derivative of the conjugate function φ^* of φ, i.e., $\psi^{-1} = \nabla\varphi^*$ [Rockafellar, 1970].
2. Since the Frobenius norm is a symmetric Bregman divergence (corresponding with $\varphi(x) = \frac{1}{2}x^2$), it comes as no surprise that (3.7) & (3.8) coincide with the Frobenius norm NNMA updates derived by Lee and Seung [2000]
3. The similarity between (3.7), (3.8) and (3.3), (3.4) is striking, though not unexpected, since the latter updates scale all the elements of $\boldsymbol{c}$ by the same amount.
4. The reader may have observed that the auxiliary functions derived above depend only on the fact that Bregman divergences are convex in their first argument. Therefore, for minimizing a distortion measure $D(\boldsymbol{B}\boldsymbol{c}, \boldsymbol{a}) = \sum_i D_i((\boldsymbol{B}\boldsymbol{c})_i, a_i)$, where each individual distortion function D_i is convex in its first argument, we may use the following general approach:

   (a) Let $\lambda_{ij} = \frac{b_{ij}\tilde{c}_j}{(\boldsymbol{B}\tilde{\boldsymbol{c}})_i}$ (or some other suitable set of coefficients that satisfies $\lambda_{ij} \geq 0$ and $\sum_j \lambda_{ij} = 1$).

   (b) $D(\boldsymbol{B}\boldsymbol{c}, \boldsymbol{a}) = \sum_i D_i((\boldsymbol{B}\boldsymbol{c})_i, a_i) \leq \sum_{ij} \lambda_{ij} D_i(\frac{b_{ij}c_j}{\lambda_{ij}}, a_i) = G(\boldsymbol{c}, \tilde{\boldsymbol{c}})$.

   (c) Optimize $G(\boldsymbol{c}, \tilde{\boldsymbol{c}})$ w.r.t. each component c_p by setting its derivative to zero and solving

   $$\sum_i b_{ip}\, \nabla D_i\big(\frac{c_p}{\tilde{c}_p}(\boldsymbol{B}\tilde{\boldsymbol{c}})_i, a_i\big) = 0.$$

   With $D_i(x, y) = D_\varphi(x; y)$, we can obtain our method for problem (P1). With $D_i(x, y) = x\varphi(y/x)$, we obtain methods for minimizing Csiszár's generalized divergences. Hence, our method is extensible to a large variety of convex losses.

3.1.4 Nonzero penalty functions

In the derivations above the central task was to develop an auxiliary function and then minimize it. If $G(\boldsymbol{c}, \tilde{\boldsymbol{c}})$ is an auxiliary function for $F(\boldsymbol{c})$ we see that $G(\boldsymbol{c}, \tilde{\boldsymbol{c}}) + \beta(\boldsymbol{c})$ is an auxiliary function for $F(\boldsymbol{c}) + \beta(\boldsymbol{c})$. However, this auxiliary function might not yield simple updates as it does not necessarily lead to a decoupling of the

individual components of $\boldsymbol{c}$. It would be better to find an auxiliary function for $\beta(\boldsymbol{c})$ too, and then proceed as before. Unfortunately, it can often happen that finding an appropriate auxiliary function is not easy. Our simple heuristic below shows how to tackle this difficult case. The procedure is:

1. Let $G(\boldsymbol{c}, \tilde{\boldsymbol{c}}) = \sum_{ij} \lambda_{ij} \varphi\big(\frac{b_{ij} c_j}{\lambda_{ij}}\big) - \big(\sum_i \varphi(a_i) + \psi(a_i)\big((\boldsymbol{B}\boldsymbol{c})_i - a_i\big)\big) + \beta(\boldsymbol{c})$
2. Differentiate $G(\boldsymbol{c}, \tilde{\boldsymbol{c}})$ w.r.t. $\boldsymbol{c}$. The penalty function contributes the term $\nabla_{\boldsymbol{c}} \beta(\boldsymbol{c})$.
3. The resulting system of non-linear equations is hard to solve even if we assume that ψ is "factorizable." To make it easier to solve we approximate $\nabla_{\boldsymbol{c}} \beta(\boldsymbol{c}) \approx \nabla_{\boldsymbol{c}} \beta(\boldsymbol{c}) \,|_{\boldsymbol{c}=\tilde{\boldsymbol{c}}}$
4. Solve the resulting system of equations in a manner similar to that described in the previous section (for factorizable ψ).

This procedure yields the update

$$c_p = \tilde{c}_p \cdot \psi^{-1}\Big(\frac{[\boldsymbol{B}^T \psi(\boldsymbol{a})]_p - [(\nabla \beta)(\tilde{\boldsymbol{c}})]_p}{[\boldsymbol{B}^T \psi(\boldsymbol{B}\tilde{\boldsymbol{c}})]_p}\Big), \tag{3.9}$$

when ψ is multiplicative. Care must be taken to ensure that the argument of ψ^{-1} remains within the domain of ψ^{-1} and to respect the non-negativity of c_p.

3.1.5 Nonlinear models with "link" functions

Certain nonlinear relationships between the input $\boldsymbol{A}$ and its approximant $\boldsymbol{BC}$ may be modeled by a "link" function that describes the nonlinearity. For example the link function h can be used to model a relation of the form $\boldsymbol{A} \approx h(\boldsymbol{BC})$. To obtain $\boldsymbol{BC}$ we may wish to solve

$$\min \;\; D_\varphi(h(\boldsymbol{BC}); \boldsymbol{A}), \qquad \boldsymbol{B}, \boldsymbol{C} \geq 0. \tag{3.10}$$

Clearly, solving (3.10) for arbitrary link functions h can be difficult. However, if $(\varphi \circ h)$ is convex, then we can obtain algorithms for this problem with link functions without too much difficulty. For simplicity we restrict h to be an elementwise function of its matrix argument.

For example, if h is convex (concave) and φ is an increasing (decreasing) function then, $\varphi \circ h$ is also convex as may be verified by considering the second derivative

$$(\varphi \circ h)''(x) = h''(x)\psi(h(x)) + \psi'(h(x))(h'(x))^2,$$

which is nonnegative for the specified h and φ. Writing $g = (\varphi \circ h)$ one can verify that

$$\begin{aligned} F(\boldsymbol{c}) = D_\varphi(h(\boldsymbol{Bc}); \boldsymbol{a}) &= \sum_i g((\boldsymbol{Bc})_i) - \varphi(a_i) - \psi(a_i)(h(\boldsymbol{Bc})_i - a_i) \\ &\leq \sum_{ij} \lambda_{ij} g(\frac{b_{ij} c_j}{\lambda_{ij}}) - \Big(\sum_i g(a_i) + \psi(a_i)(h(\boldsymbol{Bc})_i - a_i)\Big). \end{aligned} \tag{3.11}$$

If we further assume that $\psi(x) \geq 0$ (for $x \geq 0$), then using the convexity of h we may also define the divergence

$$\sum_i \psi(a_i) D_h((\boldsymbol{Bc})_i; (\boldsymbol{B}\tilde{\boldsymbol{c}})_i). \tag{3.12}$$

Adding (3.11) and (3.12) we obtain the function

$$G(\boldsymbol{c},\tilde{\boldsymbol{c}}) = \sum_{ij}\lambda_{ij}g\big(\frac{b_{ij}c_j}{\lambda_{ij}}\big) - \sum_i\Big(\varphi(a_i) + a_i\psi(a_i) + \psi(a_i)\big\{h((\boldsymbol{B}\tilde{\boldsymbol{c}})_i) + h'((\boldsymbol{B}\tilde{\boldsymbol{c}})_i)((\boldsymbol{B}\boldsymbol{c})_i - (\boldsymbol{B}\tilde{\boldsymbol{c}})_i)\big\}\Big),$$

which is clearly an auxiliary function for $F(\boldsymbol{c})$. As before, we differentiate G w.r.t. c_p to obtain

$$\frac{\partial G}{\partial c_p} = \sum_i g'\big(\frac{c_p}{\tilde{c}_p}(\boldsymbol{B}\tilde{\boldsymbol{c}})_i\big)b_{ip} - b_{ip}h'((\boldsymbol{B}\tilde{\boldsymbol{c}})_i)\psi(a_i). \tag{3.13}$$

Finally, to obtain the actual update, we just need to solve $\partial G/\partial c_p = 0$ (which, depending on h, may or may not be analytically solvable). As before, the resulting updates are guaranteed to decrease the objective function (3.10) monotonically.

3.2 Algorithms for Problem (P2)

In this section we derive algorithms for solving Problem (P2). As before, the aim is to obtain multiplicative updates for $\boldsymbol{c}$. Let $F(\boldsymbol{c}) = D_\varphi(\boldsymbol{a};\boldsymbol{B}\boldsymbol{c})$, $\psi(x) = \varphi'(x)$, and $\zeta(x) = \psi'(x)$. We present three different algorithms that illustrate how to iteratively minimize $F(\boldsymbol{c})$. First, in Section 3.2.1 we present a simple multiplicative scheme, then we exploit the concept of link functions to present a new algorithm in Section 3.2.2 followed by algorithms based on approximately solving the KKT necessary conditions in Section 3.2.3.

3.2.1 Simple multiplicative updates for (P2)

Given $F(\boldsymbol{c}) = D_\varphi(\boldsymbol{a};\boldsymbol{B}\boldsymbol{c})$, we wish to compute a factor η so that $F(\eta\boldsymbol{c}) \le F(\boldsymbol{c})$, hence ensuring a monotonic decrease in the objective function. If we set

$$\eta^* = \operatorname*{argmin}_\eta F(\eta\boldsymbol{c}),$$

and update $\boldsymbol{c} \leftarrow \eta^*\boldsymbol{c}$, then clearly $F(\eta^*\boldsymbol{c}) \le F(\boldsymbol{c})$. Differentiating $F(\eta\boldsymbol{c})$ w.r.t. η and setting the derivative to zero we have

$$\sum_i \zeta(\eta(\boldsymbol{B}\boldsymbol{c})_i)(\boldsymbol{B}\boldsymbol{c})_i(\eta(\boldsymbol{B}\boldsymbol{c})_i - a_i) = 0. \tag{3.14}$$

Assuming that $\zeta(xy) = g(x)\zeta(y)$ we solve (3.14) and obtain (note $F''(\eta\boldsymbol{c}) \ge 0$)

$$\eta^* = \frac{\boldsymbol{c}^T\boldsymbol{B}^T Z(\boldsymbol{B}\boldsymbol{c})\boldsymbol{a}}{\boldsymbol{c}^T\boldsymbol{B}^T Z(\boldsymbol{B}\boldsymbol{c})\boldsymbol{B}\boldsymbol{c}},$$

where $Z(\boldsymbol{x}) = \operatorname{diag}(\zeta(\boldsymbol{x}))$. We derive the update factor for a given row of $\boldsymbol{B}$ in a similar way. Thus, we have the following iterative scheme (for each row $\boldsymbol{b}$ of $\boldsymbol{B}$ and column $\boldsymbol{c}$ of $\boldsymbol{C}$)

$$\boldsymbol{b} \leftarrow \frac{\boldsymbol{a}^T Z(\boldsymbol{b}^T\boldsymbol{C})\boldsymbol{C}^T\boldsymbol{b}}{\boldsymbol{b}^T\boldsymbol{C}Z(\boldsymbol{b}^T\boldsymbol{C})\boldsymbol{C}^T\boldsymbol{b}}\boldsymbol{b} \tag{3.15}$$

$$\boldsymbol{c} \leftarrow \frac{\boldsymbol{c}^T\boldsymbol{B}^T Z(\boldsymbol{B}\boldsymbol{c})\boldsymbol{a}}{\boldsymbol{c}^T\boldsymbol{B}^T Z(\boldsymbol{B}\boldsymbol{c})\boldsymbol{B}\boldsymbol{c}}\boldsymbol{c}. \tag{3.16}$$

Remark. This update scheme is reminiscent of the conjugate-gradient method.

3.2.2 Solutions for (P2) via link functions

Link functions arise naturally for Bregman divergences due to the following relation,

$$D_\varphi(x;y) = D_{\varphi^*}(\psi(y);\psi(x)),$$

where $\varphi^*(x)$ is the *Legendre-conjugate*[4] of φ. We use this relation to convert Problem (P2) into an equivalent problem involving ψ as the link function. Observe that if ψ is convex, then $g = (\varphi^* \circ \psi)$ is also convex, since $g''(x) = x\psi''(x) + \zeta(x) \geq 0$, using $\zeta(x)$ to denote $\varphi''(x)$. In such a case we may write

$$F(\boldsymbol{c}) = D_\varphi(\boldsymbol{a}; \boldsymbol{B}\boldsymbol{c}) = D_{\varphi^*}(\psi(\boldsymbol{B}\boldsymbol{c}); \psi(\boldsymbol{a})).$$

Now using $h = \psi$ as the link function, and following the approach of Section 3.1.5 (suitably modifying (3.11) and (3.12)) we obtain

$$G(\boldsymbol{c}, \tilde{\boldsymbol{c}}) = \sum_{ij} \lambda_{ij} g\big(\frac{b_{ij} c_j}{\lambda_{ij}}\big) - \sum_i \bigg(g(a_i) + a_i\big\{\psi(a_i) + \psi_i((\boldsymbol{B}\tilde{\boldsymbol{c}})_i) + a_i\zeta((\boldsymbol{B}\tilde{\boldsymbol{c}})_i)((\boldsymbol{B}\boldsymbol{c})_i - (\boldsymbol{B}\tilde{\boldsymbol{c}})_i)\big\}\bigg),$$

as an auxiliary function for $F(\boldsymbol{c})$. As usual we differentiate G w.r.t. c_p and obtain

$$\frac{\partial G}{\partial c_p} = \sum_i g'\big(\frac{c_p}{\tilde{c}_p}(\boldsymbol{B}\tilde{\boldsymbol{c}})_i\big) b_{ip} - b_{ip}\zeta((\boldsymbol{B}\tilde{\boldsymbol{c}})_i) a_i.$$

Assuming ζ is separable and using $g'(x) = x\zeta(x)$, we can solve $\partial G/\partial c_p = 0$ to obtain the update

$$c_p\zeta(c_p) = \tilde{c}_p\zeta(\tilde{c}_p)\bigg(\frac{[\boldsymbol{B}^T Z(\boldsymbol{B}\tilde{\boldsymbol{c}})\boldsymbol{a}]_p}{[\boldsymbol{B}^T Z(\boldsymbol{B}\tilde{\boldsymbol{c}})\boldsymbol{B}\tilde{\boldsymbol{c}}]_p}\bigg), \tag{3.17}$$

where $Z(\boldsymbol{x}) = \operatorname{diag}(\zeta(x_i))$. This update is somewhat complicated by the ζ terms. In the next section we follow a different approach to derive simpler updates, including those that do not depend upon the separability of ζ, or on the convexity of ψ.

3.2.3 Algorithms based on KKT conditions

The approach in this section is different from the previous sections. As before, we develop our methods with α and β initially set to zero, noting that differentiable functions α and β may be easily incorporated into the updates. We use $Z(\boldsymbol{x}) = \operatorname{diag}(\zeta(x_i))$ for notational convenience. The updates here are based on approximately solving the KKT necessary conditions and they lead to multiplicative updates for each component of $\boldsymbol{c}$.

Consider minimizing $F(\boldsymbol{c}) = D_\varphi(\boldsymbol{a}; \boldsymbol{B}\boldsymbol{c})$ subject to $\boldsymbol{c} \geq \boldsymbol{0}$. The Lagrangian is

$$L(\boldsymbol{c}, \boldsymbol{\lambda}) = F(\boldsymbol{c}) - \boldsymbol{\lambda}^T\boldsymbol{c},$$

where $\boldsymbol{\lambda} \geq \boldsymbol{0}$ is the vector of Lagrange multipliers. The KKT necessary conditions are

$$[\nabla_{\boldsymbol{c}} F(\boldsymbol{c})]_p = \lambda_p \tag{3.18a}$$

$$\lambda_p c_p = 0 \tag{3.18b}$$

$$\lambda_p \geq 0, c_p \geq 0. \tag{3.18c}$$

[4]The Legendre-conjugate of a convex function is defined as $\varphi^*(y) = \sup_x\big(xy - \varphi(x)\big)$.

Combining the fact

$$[\nabla_{\boldsymbol{c}} F(\boldsymbol{c})]_p = [\boldsymbol{B}^T Z(\boldsymbol{B}\boldsymbol{c})(\boldsymbol{B}\boldsymbol{c} - \boldsymbol{a})]_p,$$

with (3.18a) and (3.18b), we obtain

$$[\boldsymbol{B}^T Z(\boldsymbol{B}\boldsymbol{c})\boldsymbol{B}\boldsymbol{c}]_p c_p = [\boldsymbol{B}^T Z(\boldsymbol{B}\boldsymbol{c})\boldsymbol{a}]_p c_p. \tag{3.19}$$

Since $\boldsymbol{c}$ occurs on both sides of (3.19) we could solve for $\boldsymbol{c}$ iteratively. Hence, we are led to the following simple update

$$c_p \leftarrow \tilde{c}_p \frac{[\boldsymbol{B}^T Z(\boldsymbol{B}\tilde{\boldsymbol{c}})\boldsymbol{a}]_p}{[\boldsymbol{B}^T Z(\boldsymbol{B}\tilde{\boldsymbol{c}})\boldsymbol{B}\tilde{\boldsymbol{c}}]_p}. \tag{3.20}$$

Update (3.20) is of the form $c_p \leftarrow \eta_p c_p$, and the close similarity with update (3.16) (which is of the form $\boldsymbol{c} \leftarrow \eta \boldsymbol{c}$) is unmistakable. For $\boldsymbol{b}$ we similarly derive the update

$$b_p \leftarrow \tilde{b}_p \frac{[\boldsymbol{a}^T Z(\tilde{\boldsymbol{b}}^T \boldsymbol{C})\boldsymbol{C}^T]_p}{[\tilde{\boldsymbol{b}}^T \boldsymbol{C} Z(\tilde{\boldsymbol{b}}^T \boldsymbol{C})\boldsymbol{C}^T]_p}. \tag{3.21}$$

3.2.4 Monotonicity

We now assess the correctness of the multiplicative updates (3.20) and (3.21). Let $\boldsymbol{d}$ denote the vector obtained by updating $\boldsymbol{c}$ as per (3.20). To establish correctness, we need to show that $F(\boldsymbol{d}) \leq F(\boldsymbol{c})$. The difference

$$\begin{aligned} F(\boldsymbol{c}) - F(\boldsymbol{d}) &= \sum_i \varphi((\boldsymbol{B}\boldsymbol{d})_i) - \varphi((\boldsymbol{B}\boldsymbol{c})_i) - \psi((\boldsymbol{B}\boldsymbol{c})_i)(a_i - (\boldsymbol{B}\boldsymbol{c})_i) + \psi((\boldsymbol{B}\boldsymbol{d})_i)(a_i - (\boldsymbol{B}\boldsymbol{d})_i) \\ &= \sum_i (a_i - (\boldsymbol{B}\boldsymbol{d})_i)(\psi((\boldsymbol{B}\boldsymbol{d})_i) - \psi((\boldsymbol{B}\boldsymbol{c})_i)) + D_\varphi((\boldsymbol{B}\boldsymbol{d})_i; (\boldsymbol{B}\boldsymbol{c})_i), \end{aligned}$$

which is just the generalized Pythagorean theorem for Bregman divergences [Censor and Zenios, 1997]. In vector notation we may write

$$\Delta(\boldsymbol{d}) = F(\boldsymbol{c}) - F(\boldsymbol{d}) = (\boldsymbol{a} - \boldsymbol{B}\boldsymbol{d})^T \big(\psi(\boldsymbol{B}\boldsymbol{d}) - \psi(\boldsymbol{B}\boldsymbol{c})\big) + D_\varphi(\boldsymbol{B}\boldsymbol{d}; \boldsymbol{B}\boldsymbol{c}).$$

Thus we need to prove that the change $\Delta(\boldsymbol{d}) \geq 0$. Unfortunately, without further assumptions on φ or ψ it seems difficult to prove $\Delta(\boldsymbol{d}) \geq 0$. However for three important cases we are able to prove $\Delta(\boldsymbol{d}) \geq 0$ (in fact we prove a stronger statement) as Lemma 5 below shows. Further, if we linearize ψ we can obtain a simple proof of monotonicity, as shown in Section 3.2.5 below.

Lemma 4 (Auxiliary inequality). *Let $\boldsymbol{c}$, $\boldsymbol{d}$ be nonnegative, and $\boldsymbol{X} = \boldsymbol{Y}^T\boldsymbol{Y}$, where $\boldsymbol{Y} \geq 0$. Then*

$$\sum_i (\boldsymbol{X}\boldsymbol{c})_i \frac{d_i^2}{c_i} \geq \boldsymbol{d}^T \boldsymbol{X} \boldsymbol{d}. \tag{3.22}$$

Proof. We have the following

$$\begin{aligned} \boldsymbol{d}^T\boldsymbol{X}\boldsymbol{d} \qquad\qquad &= \boldsymbol{d}^T\boldsymbol{Y}^T\boldsymbol{Y}\boldsymbol{d} = \sum_k (\boldsymbol{Y}\boldsymbol{d})_k^2 \\ \leq \sum_{ki} \lambda_{ki} \left(\frac{y_{ki} d_i}{\lambda_{ki}}\right)^2 \qquad &= \sum_{ki} (\boldsymbol{Y}\boldsymbol{c})_k y_{ki} d_i^2 / c_i, \quad \text{using } \lambda_{ki} = \frac{y_{ki} c_i}{(\boldsymbol{Y}\boldsymbol{c})_k} \\ = \sum_l \left(\sum_k y_{ik}^T (\boldsymbol{Y}\boldsymbol{c})_k\right) \frac{d_i^2}{c_i} \;&= \sum_l \left(\sum_{kj} y_{ik}^T y_{kj} c_j\right) \frac{d_i^2}{c_i} = \quad \sum_i (\boldsymbol{X}\boldsymbol{c})_i \frac{d_i^2}{c_i}. \end{aligned}$$

□

Lemma 5 (Monotonicity). *For $\varphi(x) = \frac{1}{2}x^2$, $\varphi(x) = x\log x$, or $\varphi(x) = -\log x$, if $\boldsymbol{d}$ is obtained by updating $\boldsymbol{c}$ as per* (3.20)*, then*

$$\hat{\Delta}(\boldsymbol{d}) = \Delta(\boldsymbol{d}) - D_\varphi(\boldsymbol{Bd};\boldsymbol{Bc}) \geq 0.$$

Proof. We treat each case separately.

CASE I. For the Frobenius norm NNMA problem we have $\varphi(x) = \frac{1}{2}x^2$. Hence $\psi(x) = x$ and we need to show that

$$0 \leq \hat{\Delta}(\boldsymbol{d}) = \sum_i \big(a_i - (\boldsymbol{Bd})_i\big)\big((\boldsymbol{Bd})_i - (\boldsymbol{Bc})_i\big).$$

For this problem the update (3.20) simplifies to

$$d_i = c_i \frac{(\boldsymbol{B}^T\boldsymbol{a})_i}{(\boldsymbol{B}^T\boldsymbol{Bc})_i}. \tag{3.23}$$

Cross-multiplying and summing (3.23) over i we see that

$$\sum_i d_i(\boldsymbol{B}^T\boldsymbol{Bc})_i = \sum_i c_i(\boldsymbol{B}^T\boldsymbol{a})_i \Leftrightarrow (\boldsymbol{Bd})^T(\boldsymbol{Bc}) = \boldsymbol{a}^T(\boldsymbol{Bc}),$$

whereby $\hat{\Delta}(\boldsymbol{d}) = \boldsymbol{d}^T\boldsymbol{B}^T\boldsymbol{a} - \boldsymbol{d}^T\boldsymbol{B}^T\boldsymbol{Bd}$. Eliminating $(\boldsymbol{B}^T\boldsymbol{a})_i$ we get

$$\hat{\Delta}(\boldsymbol{d}) = \sum_i \frac{d_i^2}{c_i}(\boldsymbol{B}^T\boldsymbol{Bc})_i - d_i(\boldsymbol{B}^T\boldsymbol{Bd})_i.$$

Now, using Lemma 4 with $\boldsymbol{X} = \boldsymbol{B}^T\boldsymbol{B}$ we immediately conclude the non-negativity of $\hat{\Delta}(\boldsymbol{d})$.

CASE II. For the KL-Divergence NNMA problem we have $\psi(x) = 1 + \log x$. Hence, we need to prove that

$$0 \leq \hat{\Delta}(\boldsymbol{d}) = \sum_i (a_i - (\boldsymbol{Bd})_i)\log\frac{(\boldsymbol{Bd})_i}{(\boldsymbol{Bc})_i}.$$

We proceed by analyzing individual terms in the summation above. We have

$$\begin{aligned}
\log\frac{(\boldsymbol{Bd})_i}{(\boldsymbol{Bc})_i} &= \frac{1}{(\boldsymbol{Bc})_i}\Big((\boldsymbol{Bc})_i \log\frac{(\boldsymbol{Bd})_i}{(\boldsymbol{Bc})_i}\Big) \\
&= \frac{1}{(\boldsymbol{Bc})_i}\Big(\sum_j b_{ij}c_j \log\frac{\sum_l b_{il}d_l}{\sum_l b_{il}c_l}\Big) \\
&\geq \frac{1}{(\boldsymbol{Bc})_i}\sum_j b_{ij}c_j\log\frac{d_j}{c_j},
\end{aligned} \tag{3.24}$$

where the latter inequality follows from the log-sum inequality

$$\sum_i x_i\log\frac{x_i}{y_i} \geq (\sum_i x_i)\log\frac{\sum_i x_i}{\sum_i y_i}.$$

A second application of this log-sum inequality allows us to conclude that

$$-(\boldsymbol{Bd})_i\log\frac{(\boldsymbol{Bd})_i}{(\boldsymbol{Bc})_i} \geq -\sum_j b_{ij}d_j\log\frac{d_j}{c_j}. \tag{3.25}$$

Using (3.24) and (3.25) we conclude that

$$\begin{aligned}\Delta(\boldsymbol{d}) &\geq \sum_{ij} b_{ij} \log \frac{d_j}{c_j}\Big(\frac{a_i}{(\boldsymbol{Bc})_i}c_j - d_j\Big)\\ &= \sum_{j} \log \frac{d_j}{c_j}\Big((c_j \sum_i b_{ij}\frac{a_i}{(\boldsymbol{Bc})_i}) - \sum_i b_{ij} d_j\Big)\\ &= 0,\end{aligned}$$

where the last equality follows from the update (3.20) for d_j given by

$$d_j = c_j \frac{\sum_i b_{ij} a_i/(\boldsymbol{Bc})_i}{\sum_i b_{ij}}.$$

Hence the proof is complete.

CASE III. For the Burg-entropy NNMA problem $\varphi(x) = -\log x$. Here the domain of φ is $\mathbb{R}_{++}$. We use $\hat{\Delta}(\boldsymbol{d}) = (\boldsymbol{a} - \boldsymbol{Bd})^T(\psi(\boldsymbol{Bd}) - \psi(\boldsymbol{Bc}))$, where $\psi(x) = -1/x$. Thus, the aim is to prove

$$0 \leq \hat{\Delta}(\boldsymbol{d}) = \sum_i \big(a_i - (\boldsymbol{Bd})_i\big)\left(\frac{1}{(\boldsymbol{Bc})_i} - \frac{1}{(\boldsymbol{Bd})_i}\right).$$

The update (3.20) simplifies to

$$d_i = c_i \frac{(\boldsymbol{B}^T\boldsymbol{Za})_i}{(\boldsymbol{B}^T\boldsymbol{ZBc})_i}, \tag{3.26}$$

where $\boldsymbol{Z} = \operatorname{diag}(1/(\boldsymbol{Bc})_i^2)$. Cross-multiplying and summing (3.26) over i we obtain

$$\sum_{ij} d_i b_{ij}/(\boldsymbol{Bc})_j = \sum_{ij} c_i b_{ij} a_j/(\boldsymbol{Bc})_j^2 \quad \Leftrightarrow \quad \sum_j \frac{(\boldsymbol{Bd})_j}{(\boldsymbol{Bc})_i} = \sum_j \frac{a_j}{(\boldsymbol{Bc})_i}.$$

Hence proving $\Delta(\boldsymbol{d}) \geq 0$ boils down to proving

$$\sum_i \left(1 - \frac{a_i}{(\boldsymbol{Bd})_i}\right) \geq 0.$$

Applying convexity of $1/x$ to $1/(\boldsymbol{Bd})_i$ we obtain

$$\frac{a_i}{(\boldsymbol{Bd})_i} = \frac{a_i}{\sum_j b_{ij} d_j} \leq a_i \sum_j \frac{\lambda_{ij}}{b_{ij} d_j/\lambda_{ij}},$$

where $\sum_j \lambda_{ij} = 1$. Letting $\lambda_{ij} = b_{ij}c_j/(\boldsymbol{Bc})_i$ we have

$$\begin{aligned}\sum_i \frac{a_i}{(\boldsymbol{Bd})_i} &\leq \sum_{ij} \frac{a_i b_{ij} c_j^2}{(\boldsymbol{Bc})_i^2 d_j}\\ = \sum_{ij} \frac{a_i b_{ij} c_j (\boldsymbol{B}^T\boldsymbol{ZBc})_j}{(\boldsymbol{B}^T\boldsymbol{Za})_j (\boldsymbol{Bc})_i^2} &= \sum_j \frac{(\boldsymbol{B}^T\boldsymbol{ZBc})_j c_j}{(\boldsymbol{B}^T\boldsymbol{Za})_j} \sum_i b_{ij} \frac{1}{(\boldsymbol{Bc})_i^2} a_i\\ = \sum_j (\boldsymbol{B}^T\boldsymbol{ZBc})_j c_j &= \sum_{jk} b_{kj} (\boldsymbol{ZBc})_k c_j\\ = \sum_{jk} \frac{b_{kj} c_j}{(\boldsymbol{Bc})_k} &= \sum_k 1,\end{aligned}$$

which is what we needed to show, hence the proof is complete. □

We remark that even though convergence for the first two cases ($\varphi(x) = \frac{1}{2}x^2$ or $\varphi(x) = x\log x$) is already known [Lee and Seung, 2000], our proofs are *new* and direct. Proving $\hat{\Delta}(\boldsymbol{d}) \geq 0$ for other interesting functions φ remains a challenge.

3.2.5 Monotonicity with linearization

Consider the first order Taylor-series expansion

$$\psi((\boldsymbol{Bd})_i) \approx \psi((\boldsymbol{Bc})_i) + \zeta((\boldsymbol{Bc})_i)((\boldsymbol{Bd})_i - (\boldsymbol{Bc})_i), \tag{3.27}$$

which provides a good approximation to ψ when $\boldsymbol{Bd}$ is sufficiently close to $\boldsymbol{Bc}$ (which is usually the case after a few iterations, as revealed by our experiments). Thus, (3.27) allows us to write

$$\Delta(\boldsymbol{d}) \approx (\boldsymbol{a} - \boldsymbol{Bd})^T Z(\boldsymbol{Bc})(\boldsymbol{Bd} - \boldsymbol{Bc}) + D_\varphi(\boldsymbol{Bd}; \boldsymbol{Bc}).$$

Lemma 6 shows that

$$(\boldsymbol{a} - \boldsymbol{Bd})^T Z(\boldsymbol{Bc})(\boldsymbol{Bd} - \boldsymbol{Bc}) \geq 0,$$

from which, using nonnegativity of $D_\varphi(\boldsymbol{Bd}; \boldsymbol{Bc})$ we immediately conclude that $\Delta(\boldsymbol{d}) \geq 0$.

Lemma 6 (Monotonicity). *If $\boldsymbol{d}$ is obtained from $\boldsymbol{c}$ as per (3.20), then*

$$(\boldsymbol{a} - \boldsymbol{Bd})^T Z(\boldsymbol{Bc})(\boldsymbol{Bd} - \boldsymbol{Bc}) \geq 0.$$

Proof. From (3.20) we have

$$d_i[\boldsymbol{B}^T Z(\boldsymbol{Bc})\boldsymbol{a}]_i = [\boldsymbol{B}^T Z(\boldsymbol{Bc})\boldsymbol{Bc}]_i \frac{d_i^2}{c_i} \quad \Longrightarrow \quad \boldsymbol{a}^T Z(\boldsymbol{Bc})\boldsymbol{Bd} = \sum_i [\boldsymbol{B}^T Z(\boldsymbol{Bc})\boldsymbol{Bc}]_i \frac{d_i^2}{c_i}.$$

Similarly (3.20) also yields $(\boldsymbol{Bd})^T Z(\boldsymbol{Bc})\boldsymbol{Bc} = \boldsymbol{a}^T Z(\boldsymbol{Bc})\boldsymbol{Bc}$. Thus the claim of this lemma reduces to

$$\sum_i [\boldsymbol{B}^T Z(\boldsymbol{Bc})\boldsymbol{Bc}]_i \frac{d_i^2}{c_i} - (\boldsymbol{Bd})^T Z(\boldsymbol{Bc})\boldsymbol{Bd} \geq 0.$$

Using Lemma 4 with $\boldsymbol{X} = \boldsymbol{B}^T Z(\boldsymbol{Bc})\boldsymbol{B}$ we conclude the truth of this inequality. □

3.2.6 Miscellaneous monotonicity approaches

Monotonicity in general is easier to show given additional assumptions. For example, if $\psi(x)$ is convex and $a_i \geq (\boldsymbol{Bd})_i$, then $\hat{\Delta}(\boldsymbol{d}) \geq 0$. This claim follows from Lemma 6 upon using the convexity of ψ since

$$\psi((\boldsymbol{Bd})_i) \geq \psi((\boldsymbol{Bc})_i) + \zeta((\boldsymbol{Bc})_i)((\boldsymbol{Bd})_i - (\boldsymbol{Bc})_i).$$

However, the requirement $a_i \geq (\boldsymbol{Bd})_i$ is overly restrictive and without additional adjustments to the algorithm, not always easy to guarantee. A more interesting variant arises if we assume that $\boldsymbol{a}$ *majorizes* $\boldsymbol{Bc}$.

Convergence for ℓ_p-norms Encouraged by the proofs centered around Lemma 4 we could apply the same techniques (and some additional manipulations) to obtain a proof of convergence for the case where $\varphi(x) = \frac{1}{p}x^p$, for $p \geq 2$. As before we use $\hat{\Delta}(\boldsymbol{d}) = (\boldsymbol{a} - \boldsymbol{B}\boldsymbol{d})^T(\psi(\boldsymbol{B}\boldsymbol{d}) - \psi(\boldsymbol{B}\boldsymbol{c}))$, where $\psi(x) = x^{p-1}$. We wish to prove

$$0 \leq \hat{\Delta}(\boldsymbol{d}) = \sum_i \big(a_i - (\boldsymbol{B}\boldsymbol{d})_i\big)\big((\boldsymbol{B}\boldsymbol{d})_i^{p-1} - (\boldsymbol{B}\boldsymbol{c})_i^{p-1}\big).$$

For the present case (3.20) simplifies to

$$d_i = c_i \frac{(\boldsymbol{B}^T\boldsymbol{Z}\boldsymbol{a})_i}{(\boldsymbol{B}^T\boldsymbol{Z}\boldsymbol{B}\boldsymbol{c})_i}, \tag{3.28}$$

where $\boldsymbol{Z} = \operatorname{diag}((\boldsymbol{B}\boldsymbol{c})_i^{p-2})$ (we have dropped the factor $(p-2)$ because it cancels out). Cross-multiplying (3.28) and summing over i we obtain

$$\begin{aligned} & \sum_{ij}(p-2)d_i b_{ji}(\boldsymbol{B}\boldsymbol{c})_j^{p-2}(\boldsymbol{B}\boldsymbol{c})_j = \sum_{ij} c_i(p-2)b_{ji}(\boldsymbol{B}\boldsymbol{c})_j^{p-2}a_j \\ \Leftrightarrow \quad & \sum_i (\boldsymbol{B}\boldsymbol{d})_i(\boldsymbol{B}\boldsymbol{c})_i^{p-1} = \sum_i (\boldsymbol{B}\boldsymbol{c})_i^{p-1}a_i. \end{aligned}$$

Thus, proving $\hat{\Delta}(\boldsymbol{d}) \geq 0$ boils down to proving

$$\sum_i \big(a_i - (\boldsymbol{B}\boldsymbol{d})_i\big)(\boldsymbol{B}\boldsymbol{d})_i^{p-1} \geq 0. \tag{3.29}$$

Inequality (3.29) appears difficult on account of the $p-1$ exponent. Exploiting Conjecture 7 (empirically verified) the following sequence of inequalities and equalities establishes (3.29).

$$\begin{aligned} \sum_i (\boldsymbol{B}\boldsymbol{d})_i^p = & \sum_i (\boldsymbol{B}\boldsymbol{d})_i \hat{z}_{ii} (\boldsymbol{B}\boldsymbol{d})_i \\ \leq & \sum_i (\boldsymbol{B}^T\hat{\boldsymbol{Z}}\boldsymbol{B}\boldsymbol{c})_i \frac{d_i^2}{c_i}, && \text{Lemma 4} \\ = & \sum_i \frac{(\boldsymbol{B}^T\hat{\boldsymbol{Z}}\boldsymbol{B}\boldsymbol{c})_i(\boldsymbol{B}^T\boldsymbol{Z}\boldsymbol{a})_i}{(\boldsymbol{B}^T\boldsymbol{Z}\boldsymbol{B}\boldsymbol{c})_i} d_i, && \text{Using (3.20)} \\ = & \sum_i \frac{(\boldsymbol{B}^T\hat{\boldsymbol{Z}}\boldsymbol{B}\boldsymbol{c})_i(\boldsymbol{B}^T\boldsymbol{Z}\boldsymbol{a})_i}{(\boldsymbol{B}^T\boldsymbol{Z}\boldsymbol{B}\boldsymbol{c})_i(\boldsymbol{B}^T\hat{\boldsymbol{Z}}\boldsymbol{a})_i} (\boldsymbol{B}^T\hat{\boldsymbol{Z}}\boldsymbol{a})_i d_i \\ \leq & \sum_i (\boldsymbol{B}\boldsymbol{d})_i \hat{z}_{ii} a_i, && \text{Conjecture 7} \\ = & \sum_i a_i (\boldsymbol{B}\boldsymbol{d})_i^{p-1}. \end{aligned}$$

Conjecture 7 (Ratio conjecture). *Let $\hat{\boldsymbol{Z}} = \operatorname{diag}\big((\boldsymbol{B}\boldsymbol{d})_i^{p-2}\big)$, where $\boldsymbol{d}$ is as given by* (3.28)*. Then,*

$$\frac{(\boldsymbol{B}^T\hat{\boldsymbol{Z}}\boldsymbol{B}\boldsymbol{c})_i}{(\boldsymbol{B}^T\boldsymbol{Z}\boldsymbol{B}\boldsymbol{c})_i} \leq \frac{(\boldsymbol{B}^T\hat{\boldsymbol{Z}}\boldsymbol{a})_i}{(\boldsymbol{B}^T\boldsymbol{Z}\boldsymbol{a})_i}. \tag{3.30}$$

Further, equality holds in (3.30) *if and only if $\boldsymbol{d}$ and $\boldsymbol{c}$ are proportional, i.e., $\frac{d_i}{c_i} = \eta$.*

Proof. First we prove the iff condition. If $\boldsymbol{d} = \eta \boldsymbol{c}$ then clearly since $\hat{\boldsymbol{Z}} = \eta^{p-2}\boldsymbol{Z}$ (essentially depending on *separability* of ζ), equality holds. For the other direction assume that equality holds. Then,

$$\begin{aligned} \sum_i (\boldsymbol{B}^T \hat{\boldsymbol{Z}} \boldsymbol{B}\boldsymbol{c})_i (\boldsymbol{B}^T \boldsymbol{Z}\boldsymbol{a})_i &= \sum_i (\boldsymbol{B}^T \boldsymbol{Z}\boldsymbol{B}\boldsymbol{c})_i (\boldsymbol{B}^T \hat{\boldsymbol{Z}}\boldsymbol{a})_i \\ \text{i.e.,} \quad \boldsymbol{c}^T \boldsymbol{B}^T \hat{\boldsymbol{Z}} \boldsymbol{B}\boldsymbol{B}^T \boldsymbol{Z}\boldsymbol{a} &= \boldsymbol{c}^T \boldsymbol{B}^T \boldsymbol{Z}\boldsymbol{B}\boldsymbol{B}^T \hat{\boldsymbol{Z}}\boldsymbol{a}. \end{aligned}$$

Since $\boldsymbol{a}$ and $\boldsymbol{B}\boldsymbol{c}$ are arbitrary, we must have $\hat{\boldsymbol{Z}}\boldsymbol{B}\boldsymbol{B}^T\boldsymbol{Z} = \boldsymbol{Z}\boldsymbol{B}\boldsymbol{B}^T\hat{\boldsymbol{Z}}$. Since $\boldsymbol{Z}$ and $\hat{\boldsymbol{Z}}$ are diagonal, this implies that

$$\hat{z}_{ii} z_{jj} = z_{ii} \hat{z}_{jj}, \qquad \text{for all } i, j,$$

which is possible only when $\hat{z}_{ii}/z_{ii} = \eta$, for some constant η. This, in turn implies that $d_i = \eta c_i$.
Now we need to prove the inequality—as of now we do not have a proof. □

3.2.7 Nonzero penalty functions

We now look at the case with nonzero (differentiable) penalty functions by proceeding along the same lines as Section 3.2. Other authors have also obtained similar updates with penalty functions, e.g., [Cichocki et al., 2006a,b]. Consider minimizing $F(\boldsymbol{c}) + \beta(\boldsymbol{c})$ (the penalty function need not be separable, but for notational convenience we write it here as a function of the column $\boldsymbol{c}$ under consideration) subject to $\boldsymbol{c} \geq 0$. We form the Lagrangian, differentiate it, and write out the KKT necessary conditions for optimality

$$\nabla_{\boldsymbol{c}}[F(\boldsymbol{c}) + \beta(\boldsymbol{c})] = \boldsymbol{\lambda} \tag{3.31a}$$

$$\lambda_p c_p = 0 \tag{3.31b}$$

$$\lambda_p \geq 0, c_p \geq 0. \tag{3.31c}$$

Once again we use (3.31a), (3.31b) in conjunction with the gradient

$$\nabla_{\boldsymbol{c}}(F(\boldsymbol{c}) + \beta(\boldsymbol{c})) = \boldsymbol{B}^T Z(\boldsymbol{B}\boldsymbol{c})(\boldsymbol{B}\boldsymbol{c} - \boldsymbol{a}) + \nabla_{\boldsymbol{c}}\beta(\boldsymbol{c}),$$

to obtain the iterative update

$$c_p \leftarrow \tilde{c}_p \frac{[\boldsymbol{B}^T Z(\boldsymbol{B}\tilde{\boldsymbol{c}})\boldsymbol{a}]_p}{[\boldsymbol{B}^T Z(\boldsymbol{B}\tilde{\boldsymbol{c}})\boldsymbol{B}\tilde{\boldsymbol{c}}]_p + [\nabla_{\boldsymbol{c}}\beta(\boldsymbol{c})]_p}. \tag{3.32}$$

The update for $\boldsymbol{b}$ may be derived similarly and we skip it for brevity. If $[\nabla_{\boldsymbol{c}}\beta(\boldsymbol{c})]_p \geq 0$ we do not have to do any additional work to enforce nonnegativity of c_p. Otherwise, we might have to resort to additional heuristics to respect the nonnegativity. Furthermore, one must ensure that the denominator remains nonzero. Since the problem with penalties is more general than the one without, proofs of convergence can be difficult to furnish, and are deferred to future work.

3.3 Further generalizations

Our methods can be used to minimize other divergence measures too. Below we illustrate two such possibilities as an example. We derive updates for minimizing Csiszár's φ-divergences and Young's divergences (defined below). We remark that additional work on minimizing Csiszár's divergences for NNMA problems has appeared previously, e.g., [Cichocki et al., 2006b].

3.3.1 Csiszár's φ-divergences.

Let $\varphi(x)$ be a convex function defined on $x > 0$ with $\varphi(1) = 0$. Csiszár's φ-divergence between $\boldsymbol{x} \geq 0$ and $\boldsymbol{y} \geq 0$ is defined as

$$D_{C\varphi}(\boldsymbol{x};\, \boldsymbol{y}) = \sum_i y_i \varphi\left(\frac{x_i}{y_i}\right).$$

The two associated NNMA problems are

$$\min_{\boldsymbol{c}\geq 0} \; D_{C\varphi}(\boldsymbol{B}\boldsymbol{c};\, \boldsymbol{a}) = \sum_i (\boldsymbol{B}\boldsymbol{c})_i \varphi\left(\frac{a_i}{(\boldsymbol{B}\boldsymbol{c})_i}\right), \tag{3.33}$$

and

$$\min_{\boldsymbol{c}\geq 0} \; D_{C\varphi}(\boldsymbol{a}; \boldsymbol{B}\boldsymbol{c}) = \sum_i a_i \varphi\left(\frac{(\boldsymbol{B}\boldsymbol{c})_i}{a_i}\right). \tag{3.34}$$

We may follow the auxiliary function method suggested in Section 3.1.3 or the KKT technique of Section 3.2 to obtain updates for minimizing the above two functions. For example, when minimizing (3.34) using the auxiliary function method of § 3.1.3 we can obtain the updated value of c_p by solving ($\partial G/\partial c_p = 0$)

$$\sum_i b_{ip} a_i \psi\left(\frac{c_p (\boldsymbol{B}\tilde{\boldsymbol{c}})_i}{\tilde{c}_p a_i}\right) = 0. \tag{3.35}$$

The update (3.35) will decrease the objective function (3.34) monotonically (by construction). However, as before, it is not always possible to solve (3.35) analytically. In principle, one can solve (3.35) iteratively in such cases.

If we follow the KKT approach, we obtain the following update rule for minimizing (3.33)

$$c_p \leftarrow \tilde{c}_p \frac{[\boldsymbol{B}^T \varphi(\boldsymbol{r})]_p}{[\boldsymbol{B}^T(\psi(\boldsymbol{r}) \odot \boldsymbol{r})]_p}, \qquad \boldsymbol{r} = \boldsymbol{a}/\boldsymbol{B}\boldsymbol{c}. \tag{3.36}$$

Experiments reveal that for $\boldsymbol{r} \leq \mathbf{1}$ update (3.36) leads to a monotonic decrease as long as the nonnegativity of all the elements can be maintained. More work is needed to determine the conditions under which (3.36) yields monotonically decreasing updates. Related work on minimizing Csiszár's divergence for NNMA may be found in [Cichocki et al., 2006b].

3.3.2 Young's Divergence

Fenchel's inequality [see Boyd and Vandenberghe, 2004, pg. 94] states that

$$\varphi(\boldsymbol{x}) + \varphi^*(\boldsymbol{y}) \geq \boldsymbol{x}^T \boldsymbol{y},$$

where φ^* is the convex-conjugate of φ. When φ is differentiable (which we assume it to be) this inequality is called *Young's inequality*. Thus, one may define a divergence

$$D_Y^{\varphi}(\boldsymbol{x}; \boldsymbol{y}) = \varphi(\boldsymbol{x}) - \boldsymbol{x}^T \boldsymbol{y} + \varphi^*(\boldsymbol{y}). \tag{3.37}$$

This divergence was previously called the *Generalized Bregman's divergence* by Gordon [2003], but owing to its genesis from Young's inequality we prefer to call it *Young's divergence*. This divergence does not in general satisfy $D_Y^{\varphi}(\boldsymbol{x}; \boldsymbol{x}) = 0$, and there may exist $\boldsymbol{y} \neq \boldsymbol{x}$ such that $D_Y^{\varphi}(\boldsymbol{x}; \boldsymbol{y}) = 0$. Assume for simplicity that both $\operatorname{dom}\varphi$ and $\operatorname{dom}\varphi^* \subseteq \mathbb{R}_+$. Thus, we redefine (3.37) as

$$D_Y^{\varphi}(\boldsymbol{x}; \boldsymbol{y}) = \sum_i \varphi(x_i) - x_i y_i + \varphi^*(y_i).$$

Young's divergence is particularly conducive to optimization, since both φ and φ^* are convex. Thus, it is easy to obtain provably convergent iterative algorithms for minimizing both $D_Y^{\varphi}(\boldsymbol{a}; \boldsymbol{Bc})$ and $D_Y^{\varphi}(\boldsymbol{Bc}; \boldsymbol{a})$ via our auxiliary function technique. We illustrate both cases below, which are essentially special cases of discussion in Section 3.1.3.

Minimizing $D_Y^{\varphi}(\boldsymbol{a}; \boldsymbol{Bc})$ For minimizing $D_Y^{\varphi}(\boldsymbol{a}; \boldsymbol{Bc})$ we again use the auxiliary function technique. Owing to the convexity of φ^* it is easy to construct an auxiliary function. As before, letting $\lambda_{ij} = b_{ij}\tilde{c}_j/(\boldsymbol{B}\tilde{\boldsymbol{c}})_i$ we construct the auxiliary function

$$G(\boldsymbol{c}, \tilde{\boldsymbol{c}}) = \sum_{ij} \lambda_{ij} \varphi^*\big(\frac{c_j}{\tilde{c}_j}(\boldsymbol{B}\tilde{\boldsymbol{c}})_i\big) + \sum_i \varphi(a_i) - (\boldsymbol{Bc})_i a_i. \tag{3.38}$$

Minimizing (3.38) w.r.t. c_p amounts to solving

$$\sum_i b_{ip} \nabla\varphi^*\big(\frac{c_p}{\tilde{c}_p}(\boldsymbol{B}\tilde{\boldsymbol{c}})_i\big) = [\boldsymbol{B}^T\boldsymbol{a}]_p \ .$$

For example, if $\nabla\varphi^* = \psi^{-1}$ is "factorizable" then we obtain the update (cf. Update 3.7)

$$c_p \leftarrow \tilde{c}_p \psi\bigg(\frac{[\boldsymbol{B}^T\boldsymbol{a}]_p}{[\boldsymbol{B}^T\psi^{-1}(\boldsymbol{B}\tilde{\boldsymbol{c}})]_p}\bigg). \tag{3.39}$$

Minimizing $D_Y^{\varphi}(\boldsymbol{Bc}; \boldsymbol{a})$ This case is tackled with equal ease as the previous one. Observe the (expected) duality between the two problems. It is easily verified that

$$G(\boldsymbol{c}, \tilde{\boldsymbol{c}}) = \sum_{ij} \lambda_{ij} \varphi\big(\frac{c_j}{\tilde{c}_j}(\boldsymbol{B}\tilde{\boldsymbol{c}})_i\big) + \sum_i \varphi^*(a_i) - (\boldsymbol{Bc})_i a_i, \tag{3.40}$$

serves as an auxiliary function. As before, for factorizable ψ we obtain the update (cf. Update 3.7)

$$c_p \leftarrow \tilde{c}_p \psi^{-1}\bigg(\frac{[\boldsymbol{B}^T\boldsymbol{a}]_p}{[\boldsymbol{B}^T\psi(\boldsymbol{B}\tilde{\boldsymbol{c}})]_p}\bigg). \tag{3.41}$$

Observe that when minimizing $D_Y^{\varphi}(\boldsymbol{Bc}; \psi(\boldsymbol{a}))$, (3.40) leads to an update identical with update (3.7) (that minimizes $D_{\varphi}(\boldsymbol{Bc}; \boldsymbol{a})$).

4 Examples of NNMA problems

In this section we present some specific NNMA problems and their solutions as obtained by the methods discussed above. We also motivate simple generalizations such as weighted and multi-factor NNMA problems by means of examples. Our (new) contributions are highlighted with a $\star$ suffixed to the section name.

4.1 New KL-Divergence NNMA*

The original NNMA problem [Lee and Seung, 1999] focused on minimizing $\mathrm{KL}(\boldsymbol{a}; \boldsymbol{Bc})$. We look at the corresponding asymmetric case that minimizes

$$\mathrm{KL}(\boldsymbol{Bc}, \boldsymbol{a}) \ = \ \sum_i (\boldsymbol{Bc})_i \log \frac{(\boldsymbol{Bc})_i}{a_i} - (\boldsymbol{Bc})_i + a_i, \qquad \boldsymbol{B}, \boldsymbol{c} \geq 0. \tag{4.1}$$

Let $\varphi(x) = x\log x - x$. Then, $\psi(x) = \log x$, and since $\psi(xy) = \psi(x) + \psi(y)$, upon substituting for ψ in (3.6) and setting the resultant to zero we obtain

$$\begin{aligned}
\frac{\partial G}{\partial c_p} &= \sum_i b_{ip}\log(c_p(\boldsymbol{B}\tilde{\boldsymbol{c}})_i/\tilde{c}_p) - \sum_i b_{ip}\log a_i = 0,\\
&\implies (\boldsymbol{B}^T\mathbf{1})_p \log\frac{c_p}{\tilde{c}_p} = [\boldsymbol{B}^T\log\boldsymbol{a} - \boldsymbol{B}^T\log(\boldsymbol{B}\tilde{\boldsymbol{c}})]_p\\
&\implies c_p = \tilde{c}_p\cdot\exp\left(\frac{[\boldsymbol{B}^T\log\big(\boldsymbol{a}/(\boldsymbol{B}\tilde{\boldsymbol{c}})\big)]_p}{[\boldsymbol{B}^T\mathbf{1}]_p}\right).
\end{aligned}$$

Similarly the update for $\boldsymbol{b}$ is derived to be

$$b_p = \tilde{b}_p\cdot\exp\left(\frac{[\big(\log(\boldsymbol{a}/\tilde{\boldsymbol{b}}^T\boldsymbol{C})\big)^T\boldsymbol{C}^T]_p}{[\mathbf{1}^T\boldsymbol{C}^T]_p}\right).$$

Due to the $\exp(\cdot)$ function it is obvious that the updates maintain the nonnegativity of c_p and b_p, provided the iteration is primed with nonnegative $\boldsymbol{c}$ and $\boldsymbol{b}$.

4.2 Constrained NNMA and Maximum Entropy*

Consider the following problem with additional linear constraints

$$\begin{aligned}
\min_{\boldsymbol{c}} \quad & D_\varphi(\boldsymbol{B}\boldsymbol{c};\boldsymbol{a})\\
\text{s.t.} \quad & \boldsymbol{P}\boldsymbol{c}\le\mathbf{0}, \quad \boldsymbol{c}\ge\mathbf{0}.
\end{aligned} \tag{4.2}$$

We introduce a differentiable penalty function for enforcing the constraints $\boldsymbol{P}\boldsymbol{c}\le\mathbf{0}$. Let,

$$F(\boldsymbol{c}) = D_\varphi(\boldsymbol{B}\boldsymbol{c};\boldsymbol{a}) + \rho\|\max(0,\boldsymbol{P}\boldsymbol{c})\|^2, \tag{4.3}$$

where $\rho > 0$ is some penalty constant. Assuming multiplicative ψ and following the auxiliary function technique described in Section 3.1.4, we obtain the following updates for $\boldsymbol{c}$,

$$c_p \leftarrow c_p\cdot\psi^{-1}\left(\frac{[\boldsymbol{B}^T\psi(\boldsymbol{a})]_p - \rho[\boldsymbol{P}^T(\boldsymbol{P}\boldsymbol{c})^+]_p}{[\boldsymbol{B}^T\psi(\boldsymbol{B}\boldsymbol{c})]_p}\right),$$

where $(\boldsymbol{P}\boldsymbol{c})^+ = \max(\mathbf{0},\boldsymbol{P}\boldsymbol{c})$. Note that care must be taken to ensure that the addition of the penalty term does not violate the nonnegativity of $\boldsymbol{c}$, and that the argument of ψ^{-1} lies in its domain.

Maximum Entropy. Incorporating additional constraints into (4.1) is easier, since the exponential updates ensure nonnegativity. Given $\boldsymbol{a} = \mathbf{1}$, consider the problem

$$\min_{\boldsymbol{c}}\mathrm{KL}(\boldsymbol{B}\boldsymbol{c},\mathbf{1}) \qquad \text{s.t. } \boldsymbol{P}\boldsymbol{c}\le 0, \quad \boldsymbol{c}\ge\mathbf{0}.$$

Using the penalty function as described above along with the derivation in §4.1 we obtain

$$c_p \quad \leftarrow \; c_p\cdot\exp\left(\frac{[-\boldsymbol{B}^T\log(\boldsymbol{B}\boldsymbol{c}) - \rho\boldsymbol{P}^T(\boldsymbol{P}\boldsymbol{c})^+]_p}{[\boldsymbol{B}^T\mathbf{1}]_p}.\right)$$

Note that one may use other penalty functions to enforce $\boldsymbol{P}\boldsymbol{c}\le\mathbf{0}$, provided that these functions are differentiable.

4.3 Lee and Seung's Algorithms.

Let $\alpha \equiv 0$, $\beta \equiv 0$. When $\varphi(x) = \frac{1}{2}x^2$, then (3.21) and (3.20) yield

$$b_{mk} \leftarrow b_{mk} \frac{(\boldsymbol{A}\boldsymbol{C}^T)_{mk}}{(\boldsymbol{B}\boldsymbol{C}\boldsymbol{C}^T)_{mk}}, \qquad c_{kn} \leftarrow c_{kn} \frac{(\boldsymbol{B}^T\boldsymbol{A})_{kn}}{(\boldsymbol{B}^T\boldsymbol{B}\boldsymbol{C})_{kn}},$$

while for $\varphi(x) = x \log x$ we obtain

$$b_{mk} \leftarrow b_{mk} \left\{ \frac{\left([\frac{\boldsymbol{A}}{\boldsymbol{B}\boldsymbol{C}}]\boldsymbol{C}^T\right)_{mk}}{(\mathbf{1}_M \mathbf{1}_N^T \boldsymbol{C}^T)_{mk}} \;=\; \frac{\sum_s c_{ks} a_{ms}/(\boldsymbol{B}\boldsymbol{C})_{ms}}{\sum_n c_{kn}} \right\},$$
$$c_{kn} \leftarrow c_{kn} \left\{ \frac{\left(\boldsymbol{B}^T[\frac{\boldsymbol{A}}{\boldsymbol{B}\boldsymbol{C}}]\right)_{kn}}{(\boldsymbol{B}^T \mathbf{1}_M \mathbf{1}_N^T)_{kn}} \;=\; \frac{\sum_t b_{tk} a_{tn}/(\boldsymbol{B}\boldsymbol{C})_{tn}}{\sum_m b_{mk}} \right\}.$$

These updates are the same as the ones originally derived by Lee and Seung [2000]. It can easily be shown that these updates are equivalent (see [Gaussier and Goutte, 2005]) to those for probabilistic latent semantic indexing (PLSI) [Hofmann, 1999].

4.4 The Multifactor NNMA Problem* and an Application

Both the NNMA problems (P1) and (P2) can be extended to the "multi-factor" problem, wherein one seeks an approximation of the type $\boldsymbol{A} \approx \boldsymbol{B}_1\boldsymbol{B}_2 \ldots \boldsymbol{B}_R$. As a simple example, consider minimizing

$$D_\varphi(\boldsymbol{A}; \boldsymbol{B}_1\boldsymbol{B}_2 \ldots \boldsymbol{B}_R),$$

where all matrices involved are nonnegative. We compute the gradient of the divergence D_φ w.r.t. each $\boldsymbol{B}_r$. Let $\widehat{\boldsymbol{B}} = \boldsymbol{B}_1\boldsymbol{B}_2 \ldots \boldsymbol{B}_{r-1}$, $\widehat{\boldsymbol{C}} = \boldsymbol{B}_{r+1}\boldsymbol{B}_{r+2} \ldots \boldsymbol{B}_R$, and $\boldsymbol{H} = \boldsymbol{B}_1\boldsymbol{B}_2 \ldots \boldsymbol{B}_R$. Let b^r_{pq} denote the (p,q)-th element of $\boldsymbol{B}_r$. It is easy to verify that

$$\frac{\partial D_\varphi}{\partial b^r_{pq}} = \left[\widehat{\boldsymbol{B}}^T(\zeta(\boldsymbol{H}) \odot (\boldsymbol{H} - \boldsymbol{A}))\widehat{\boldsymbol{C}}^T\right]_{pq}.$$

Following the derivation in Section 3.2 we obtain the update

$$\boldsymbol{B}_r \leftarrow \boldsymbol{B}_r \odot \frac{\widehat{\boldsymbol{B}}^T(\zeta(\boldsymbol{H}) \odot \boldsymbol{A})\widehat{\boldsymbol{C}}^T}{\widehat{\boldsymbol{B}}^T(\zeta(\boldsymbol{H}) \odot \boldsymbol{H})\widehat{\boldsymbol{C}}^T}. \tag{4.4}$$

Application to relaxed Co-clustering. A typical usage of multi-factor NNMA problem would be to obtain a three-factor NNMA, namely $\boldsymbol{A} \approx \boldsymbol{R}\boldsymbol{B}\boldsymbol{C}^T$. Such an approximation is closely tied to the problem of co-clustering [Cho et al., 2004], and can be used to produce relaxed co-clustering solutions, wherein the matrices $\boldsymbol{R}$, $\boldsymbol{C}$, $\boldsymbol{B}$ represent row-clustering, column-clustering, and co-cluster prototypes, respectively. For example, for the problems

$$\begin{aligned} &\min \ \|\boldsymbol{A} - \boldsymbol{R}\boldsymbol{B}\boldsymbol{C}^T\|_{\mathrm{F}}^2 && \text{(Euclidean)} \\ &\min \ \mathrm{KL}(\boldsymbol{A};\, \boldsymbol{R}\boldsymbol{B}\boldsymbol{C}^T) && \text{(Information-theoretic)}, \end{aligned}$$

an application of (4.4) yields the iterative schemes

$$\boldsymbol{R} \leftarrow \quad \boldsymbol{R} \odot \frac{\boldsymbol{A}\boldsymbol{C}\boldsymbol{B}^T}{\boldsymbol{R}\boldsymbol{B}\boldsymbol{C}^T\boldsymbol{C}\boldsymbol{B}^T}, \quad \boldsymbol{B} \leftarrow \quad \boldsymbol{B} \odot \frac{\boldsymbol{R}^T\boldsymbol{A}\boldsymbol{C}}{\boldsymbol{R}^T\boldsymbol{R}\boldsymbol{B}\boldsymbol{C}^T\boldsymbol{C}}, \quad \boldsymbol{C} \leftarrow \quad \boldsymbol{C} \odot \frac{\boldsymbol{A}^T\boldsymbol{R}\boldsymbol{B}}{\boldsymbol{C}\boldsymbol{B}^T\boldsymbol{R}^T\boldsymbol{R}\boldsymbol{B}}$$
$$\boldsymbol{R} \leftarrow \quad \boldsymbol{R} \odot \frac{[\frac{\boldsymbol{A}}{\boldsymbol{R}\boldsymbol{B}\boldsymbol{C}^T}]\boldsymbol{C}\boldsymbol{B}^T}{\mathbf{1}\mathbf{1}^T\boldsymbol{C}\boldsymbol{B}^T}, \quad \boldsymbol{B} \leftarrow \quad \boldsymbol{B} \odot \frac{\boldsymbol{R}^T[\frac{\boldsymbol{A}}{\boldsymbol{R}\boldsymbol{B}\boldsymbol{C}^T}]\boldsymbol{C}}{\boldsymbol{R}^T\mathbf{1}\mathbf{1}^T\boldsymbol{C}}, \quad \boldsymbol{C} \leftarrow \quad \boldsymbol{C} \odot \frac{[\frac{\boldsymbol{A}^T}{\boldsymbol{C}\boldsymbol{B}^T\boldsymbol{R}^T}]\boldsymbol{R}\boldsymbol{B}}{\mathbf{1}\mathbf{1}^T\boldsymbol{R}\boldsymbol{B}},$$

for the relaxations of the Euclidean and Information-theoretic clustering, respectively. In practice we can also normalize both $\boldsymbol{R}$ and $\boldsymbol{C}$ while adjusting the matrix $\boldsymbol{B}$ accordingly. It is evident that we can exploit the generality of the update (4.4) to obtain relaxed co-clustering solutions to problems of the form $\min D_\varphi(\boldsymbol{A}; \boldsymbol{RBC})$. We remark that in practice, the above updates should be implemented to exploit the sparsity of $\boldsymbol{A}$. A recent paper [Badea, 2005] also describes a co-clustering procedure based on NNMA.

4.5 Weighted NNMA Problems*

There are three main ways in which weighting may be incorporated into the NNMA model. First, we weight the objective function elementwise, second we weight the low-rank approximant elementwise, and third we weight the approximant by multiplying it with weighting matrices. The corresponding NNMA problems are

$$\begin{array}{llll} \min \sum_{ij} w_{ij} D_\varphi(a_{ij}; (\boldsymbol{BC})_{ij}) & & \text{and} & \min \sum_{ij} w_{ij} D_\varphi((\boldsymbol{BC})_{ij}; a_{ij}), \\ \min \quad D_\varphi(\boldsymbol{A}; \boldsymbol{W} \odot (\boldsymbol{BC})) & & \text{and} & \min \quad D_\varphi(\boldsymbol{W} \odot (\boldsymbol{BC}); \boldsymbol{A}), \\ \min \quad D_\varphi(\boldsymbol{A}; \boldsymbol{W}_1 \boldsymbol{BC} \boldsymbol{W}_2) & & \text{and} & \min \quad D_\varphi(\boldsymbol{W}_1 \boldsymbol{BC} \boldsymbol{W}_2; \boldsymbol{A}), \end{array}$$

where the weighting matrices $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$ are also nonnegative. All of these problems may be solved easily by the techniques developed above though to avoid repetition we skip the derivations. Table 2, however, summarizes some of the associated updates.

Example: The PMF Problem. Here we wish to minimize $\|\boldsymbol{W} \odot (\boldsymbol{A} - \boldsymbol{BC})\|_F^2$. Using $\boldsymbol{X} \leftarrow \sqrt{\boldsymbol{W}} \odot \boldsymbol{X}$, and $\boldsymbol{A} \leftarrow \sqrt{\boldsymbol{W}} \odot \boldsymbol{A}$ in (3.21) and (3.20) one obtains

$$\boldsymbol{B} \leftarrow \boldsymbol{B} \odot \frac{(\boldsymbol{W} \odot \boldsymbol{A})\boldsymbol{C}^T}{(\boldsymbol{W} \odot (\boldsymbol{BC}))\boldsymbol{C}^T}, \qquad \boldsymbol{C} \leftarrow \boldsymbol{C} \odot \frac{\boldsymbol{B}^T(\boldsymbol{W} \odot \boldsymbol{A})}{\boldsymbol{B}^T(\boldsymbol{W} \odot (\boldsymbol{BC}))}.$$

These iterative updates are significantly simpler than the PMF algorithms of Paatero and Tapper [1994] and may be used as alternatives to them.

Example: Weighted KL-Divergence Problem. Here we wish to minimize $\mathrm{KL}(\boldsymbol{A}; \boldsymbol{PBCQ})$, where $\boldsymbol{P}$ and $\boldsymbol{Q}$ are positive diagonal matrices. This problem is a slight generalization of the diagonally weighted problem considered by [Guillamet et al., 2001, 2003]. Using (4.4), we obtain

$$\begin{aligned} b_{mk} \leftarrow b_{mk} \left\{ \frac{\left(\boldsymbol{P}\left[\frac{\boldsymbol{A}}{\boldsymbol{PBCQ}}\right]\boldsymbol{QC}^T\right)_{mk}}{\left(\boldsymbol{P}(\mathbf{1}_M \mathbf{1}_N^T)\boldsymbol{QC}^T\right)_{mk}} \right. &= \left. \frac{\sum_s c_{ks} a_{ms} / (p_{mm}(\boldsymbol{BC})_{ms})}{\sum_n c_{kn} q_{nn}} \right\} \\ c_{kn} \leftarrow c_{kn} \left\{ \frac{\left(\boldsymbol{B}^T\boldsymbol{P}\left[\frac{\boldsymbol{A}}{\boldsymbol{PBCQ}}\right]\boldsymbol{Q}\right)_{kn}}{\left(\boldsymbol{B}^T\boldsymbol{P}(\mathbf{1}_M \mathbf{1}_N^T)\boldsymbol{Q}\right)_{kn}} \right. &= \left. \frac{\sum_t b_{tk} a_{tn} / (q_{nn}(\boldsymbol{BC})_{tn})}{\sum_m b_{mk} p_{mm}} \right\}. \end{aligned}$$

Observe that when $\boldsymbol{P} = \boldsymbol{I}_M$ and $\boldsymbol{Q} = \boldsymbol{I}_N$ then these updates simplify to those given in Section 4.3.

Objective function	Update			Reference
$\|\boldsymbol{A}-\boldsymbol{BC}\|_{\mathrm{F}}^2$	c_{kn}	$\leftarrow$	$c_{kn}\frac{(\boldsymbol{B}^T\boldsymbol{A})_{kn}}{(\boldsymbol{B}^T\boldsymbol{BC})_{kn}}$	§ 4.3
$\mathrm{KL}(\boldsymbol{A};\boldsymbol{BC})$	c_{kn}	$\leftarrow$	$c_{kn}\frac{\sum_t b_{tk}a_{tn}/(\boldsymbol{BC})_{tn}}{\sum_m b_{mk}}$	§ 4.3
$\mathrm{KL}(\boldsymbol{Bc};\boldsymbol{a})$	c_p	$\leftarrow$	$c_p\exp\Big(\frac{[\boldsymbol{B}^T\log\big(\boldsymbol{a}/(\boldsymbol{Bc})\big)]_p}{[\boldsymbol{B}^T\boldsymbol{1}]_p}\Big)$	§ 4.1
$D_\varphi(\boldsymbol{Bc};\boldsymbol{a})$	$\boldsymbol{c}$	$\leftarrow$	$\psi^{-1}\big[\frac{\boldsymbol{c}^T\boldsymbol{B}^T\psi(\boldsymbol{a})}{\boldsymbol{c}^T\boldsymbol{B}^T\psi(\boldsymbol{Bc})}\big]\boldsymbol{c}$	(3.4)
$D_\varphi(\boldsymbol{Bc};\boldsymbol{a})$	c_p	$\leftarrow$	$c_p\cdot\psi^{-1}\big(\frac{[\boldsymbol{B}^T\psi(\boldsymbol{a})]_p}{[\boldsymbol{B}^T\psi(\boldsymbol{Bc})]_p}\big)$	(3.7)
$D_\varphi(\boldsymbol{a};\boldsymbol{Bc})$	$\boldsymbol{c}$	$\leftarrow$	$\frac{\boldsymbol{c}^T\boldsymbol{B}^TZ(\boldsymbol{Bc})\boldsymbol{a}}{\boldsymbol{c}^T\boldsymbol{B}^TZ(\boldsymbol{Bc})\boldsymbol{Bc}}\boldsymbol{c}$	(3.16)
$D_\varphi(\boldsymbol{a};\boldsymbol{Bc})$	$g'(c_p)$	$\leftarrow$	$g'(c_p)\frac{[\boldsymbol{B}^TZ(\boldsymbol{Bc})\boldsymbol{a}]_p}{[\boldsymbol{B}^TZ(\boldsymbol{Bc})\boldsymbol{Bc}]_p}$	(3.17)
$D_\varphi(\boldsymbol{a};\boldsymbol{Bc})$	c_p	$\leftarrow$	$c_p\frac{[\boldsymbol{B}^TZ(\boldsymbol{Bc})\boldsymbol{a}]_p}{[\boldsymbol{B}^TZ(\boldsymbol{Bc})\boldsymbol{Bc}]_p}$	(3.20)
$D_\varphi(\boldsymbol{A};\boldsymbol{B}_1\boldsymbol{B}_2\ldots\boldsymbol{B}_R)$	$\boldsymbol{B}_r$	$\leftarrow$	$\boldsymbol{B}_r\odot\frac{\widehat{\boldsymbol{B}}^T(\zeta(\boldsymbol{H})\odot\boldsymbol{A})\widehat{\boldsymbol{C}}^T}{\widehat{\boldsymbol{B}}^T(\zeta(\boldsymbol{H})\odot\boldsymbol{H})\widehat{\boldsymbol{C}}^T}$	(4.4)
$\|\boldsymbol{W}\odot(\boldsymbol{A}-\boldsymbol{BC})\|_F^2$	$\boldsymbol{C}$	$\leftarrow$	$\boldsymbol{C}\odot\frac{\boldsymbol{B}^T(\boldsymbol{W}\odot\boldsymbol{A})}{\boldsymbol{B}^T(\boldsymbol{W}\odot(\boldsymbol{BC}))}$	§ 4.5
$\mathrm{KL}(\boldsymbol{A};\boldsymbol{PBCQ})$	c_{kn}	$\leftarrow$	$c_{kn}\frac{\sum_t b_{tk}a_{tn}/(q_{nn}(\boldsymbol{BC})_{tn})}{\sum_m b_{mk}p_{mm}}$	§ 4.5
$\sum_i w_iD_\varphi(a_i;(\boldsymbol{Bc})_i)$	c_p	$\leftarrow$	$c_p\frac{[\boldsymbol{B}^TZ(\boldsymbol{Bc})(\boldsymbol{w}\odot\boldsymbol{a})]_p}{[\boldsymbol{B}^TZ(\boldsymbol{Bc})(\boldsymbol{w}\odot(\boldsymbol{Bc}))]_p}$	§ 4.5
$\sum_i w_iD_\varphi((\boldsymbol{Bc})_i;a_i)$	c_p	$\leftarrow$	$c_p\cdot\psi^{-1}\big(\frac{[\boldsymbol{B}^T(\boldsymbol{w}\odot\psi(\boldsymbol{a}))]_p}{[\boldsymbol{B}^T(\boldsymbol{w}\odot(\psi(\boldsymbol{Bc})))]_p}\big)$	§ 4.5
$D_\varphi(\boldsymbol{a};\boldsymbol{w}\odot(\boldsymbol{Bc}))$	c_p	$\leftarrow$	$c_p\frac{[\boldsymbol{B}^TZ(\boldsymbol{w}\odot(\boldsymbol{Bc}))(\boldsymbol{w}\odot\boldsymbol{a})]_p}{[\boldsymbol{B}^TZ(\boldsymbol{w}\odot(\boldsymbol{Bc}))(\boldsymbol{w}^2\odot\boldsymbol{Bc})]_p}$	§ 4.5
$D_\varphi(\boldsymbol{A};\boldsymbol{W}_1\boldsymbol{BCW}_2)$	$\boldsymbol{C}$	$\leftarrow$	$\boldsymbol{C}\odot\frac{\boldsymbol{B}^T\boldsymbol{W}_1^T\big(\zeta(\boldsymbol{Z})\odot\boldsymbol{A}\big)\boldsymbol{W}_2^T}{\boldsymbol{B}^T\boldsymbol{W}_1^T\big(\zeta(\boldsymbol{Z})\odot(\boldsymbol{Z})\big)\boldsymbol{W}_2^T}$	$\boldsymbol{Z}=\boldsymbol{W}_1\boldsymbol{BCW}_2$
$D_{C\varphi}(\boldsymbol{Bc};\,\boldsymbol{a})$	c_p	$\leftarrow$	$c_p\frac{[\boldsymbol{B}^T\varphi(\boldsymbol{r})]_p}{[\boldsymbol{B}^T\mathrm{diag}(\boldsymbol{r})\psi(\boldsymbol{r})]_p}$	§ 3.3.1
$D_Y^\varphi(\boldsymbol{Bc};\boldsymbol{a})$	c_p	$\leftarrow$	$\tilde{c}_p\cdot\psi^{-1}\left(\frac{[\boldsymbol{B}^T\boldsymbol{a}]_p}{[\boldsymbol{B}^T\psi(\boldsymbol{B}\tilde{\boldsymbol{c}})]_p}\right)$	§ 3.3.2
$D_Y^\varphi(\boldsymbol{a};\boldsymbol{Bc})$	c_p	$\leftarrow$	$\tilde{c}_p\cdot\psi\left(\frac{[\boldsymbol{B}^T\boldsymbol{a}]_p}{[\boldsymbol{B}^T\psi^{-1}(\boldsymbol{B}\tilde{\boldsymbol{c}})]_p}\right)$	§ 3.3.2

Table 2: Summary of some NNMA algorithms. The updates are shown either for individual elements of $\boldsymbol{C}$ (c_{kn}), the entire matrix $\boldsymbol{C}$, or individual elements (c_p) of an arbitrary column of $\boldsymbol{c}$. The corresponding updates for $\boldsymbol{B}$ are similar and have been omitted for brevity.

5 Experiments

This section presents simple experiments to illustrate some of the properties of our algorithms. We do not focus on any particular application and point the interested reader to the vast list of applications in Section 6.2).

5.1 Monotonic convergence

First we illustrate the monotonic convergence behavior of some of our algorithms that were implemented in MATLAB. However, this implementation is for illustrative purposes only; we refer the reader to our high performance implementation in C++ for tackling real world datasets.

Figure 2 reports how the respective objective functions decreased monotonically while performing a rank-3 decomposition for a 20×8 nonnegative input matrix. The first subfigure in the second row shows that the simple multiplicative scaling procedure (3.4)) hits its stationary point within two iterations, and no further

improvement to the divergence is achieved thereafter. Comparing it with the second subfigure in row two, we see that the elementwise multiplicative update (3.7) leads to a better local minimum for the same objective function value.

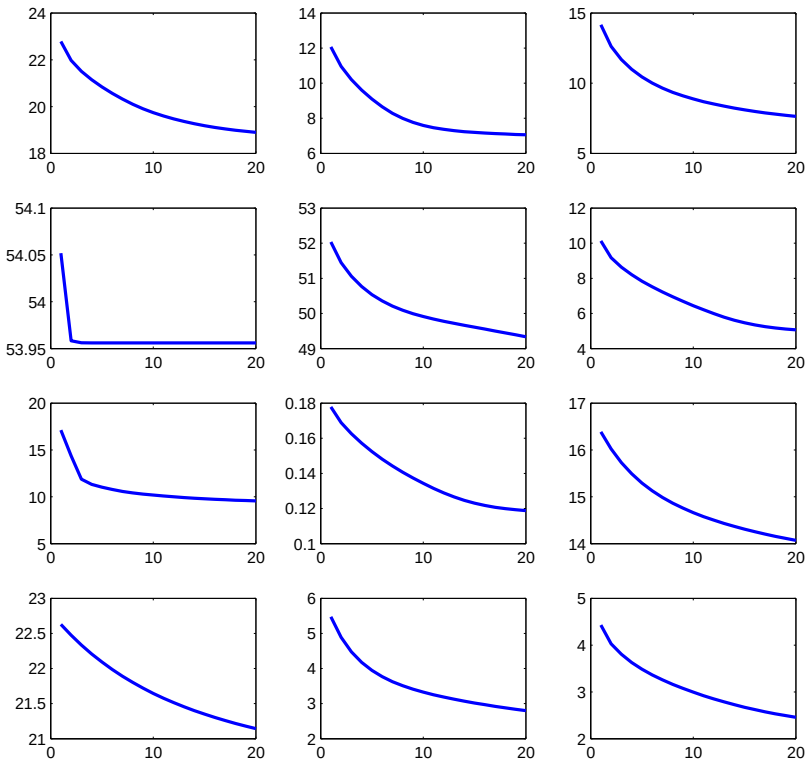


Figure 2: Illustration of monotonic decrease in objective function values for a various instances of NNMA problems. A rank-3 decomposition of a 20×8 matrix was obtained for each case. The first row (left to right) shows NNMA for $\|\boldsymbol{A} - \boldsymbol{BC}\|_{\mathrm{F}}^2$, KL$(\boldsymbol{A};\, \boldsymbol{BC})$, and KL$(\boldsymbol{BC};\, \boldsymbol{A})$. The second row shows NNMA update (3.4) ($D_\varphi(\boldsymbol{BC}; \boldsymbol{A})$) with $\varphi = \frac{1}{3}x^3$, update (3.7) ($D_\varphi(\boldsymbol{BC}; \boldsymbol{A})$) with $\varphi = \frac{1}{3}x^3$, and update (3.20) ($D_\varphi(\boldsymbol{A}; \boldsymbol{BC})$) with $\varphi = \exp(x)$. The third row shows NNMA update (3.20) with $\varphi = x^8$ followed by updates for $\|\boldsymbol{W} \odot (\boldsymbol{A} - \boldsymbol{BC})\|_{\mathrm{F}}^2$, and KL$(\boldsymbol{A};\, \boldsymbol{W}_1\boldsymbol{BCW}_2)$ with random weighting matrices. The last row shows Csiszár's divergence update (3.36) with $\varphi = \sqrt{x-1}$, Young's divergence updates (3.39), and (3.41) with $\varphi = \frac{1}{3}x^3$. The absolute values of the objective functions cannot of course be compared to each other; the essential point being the monotonicity.

While producing these figures, we noticed another interesting trend. Our algorithm for minimizing $D_\varphi(\boldsymbol{A}; \boldsymbol{BC})$ also monotonically decreases $D_\varphi(\boldsymbol{BC}; \boldsymbol{A})$ at the same time, and vice-versa. The same holds true for our algorithms that minimize Young's divergences. This curious effect is somewhat unexpected considering the innate asymmetry of these divergences. However, it remains to be investigated in greater detail.

5.2 Effect of objective function

We remark that if the model fits the data well then the particular objective function selected does not play a very important role. However, if one has an *a priori* assumption on the noise corrupting the observed data, then minimizing the Bregman divergence corresponding to the assumed noise distribution is expected to given a better reconstruction. Banerjee et al. [2005] illustrate clustering results on data following different distributions, demonstrating that if one the matching Bregman divergence the resulting clustering accuracy is higher. Their observation lends support to our recommendation for selecting an appropriate divergence for minimization.

The application being studied can govern the selection of the objective function, see for example the recent work [Chen et al., 2006, Cichocki et al., 2006a,c]. In addition, the summary of applications in Section 6.2 can provide some additional information about the choice of divergence.

Another important factor governing the choice of objective function is the ease of minimization and computational complexity, especially in the presence of additional regularization terms and/or constraints. Furthermore, the sparsity pattern of the input can govern which objective function we choose to use. However, just as selecting the appropriate kernel is not always easy, it is difficult to give general prescriptions for which particular divergence measure is most suited to a given problem. Usually experience and knowledge about the data determine the choice of the divergence measure.

6 Brief Literature Review

Since its introduction, NNMA has been increasingly applied as a technique for dimensionality reduction and data analysis. Correspondingly, there has been a significant amount of research related to it. The aim of this section is to provide a brief summary about the various algorithms and applications of NNMA that have appeared in the literature. While attempt has been made to be as complete as possible, the sheer magnitude of the task renders it impossible to attain completeness. We apologize in advance to the authors whose work we might have inadvertently missed.

The origin of the *approximate* nonnegative factorization problem or NNMA may be credited to [Paatero et al., 1991] who called it Positive Matrix Factorization (PMF), and to [Lee and Seung, 1999] who called it Nonnegative Matrix Factorization. The *exact* factorization problem is however older, and Section 7 digresses briefly into it.

6.1 Algorithms

There exist a few different algorithms for NNMA. Some of them are based on solving suitably modified non-linear least squares problems, while others are simple iterative procedures. We summarize procedures of both types below.

6.1.1 Paatero's methods

Paatero et al. [1991] introduced the term PMF and sought to construct a factor model with two nonnegative matrices by minimizing

$$\|\boldsymbol{W} \odot (\boldsymbol{A} - \boldsymbol{BC})\|_{\mathrm{F}}^2, \tag{6.1}$$

where $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, and $\boldsymbol{W}$ are all nonnegative. The matrix $\boldsymbol{W}$ consists of weights reflecting confidence in the measurements in $\boldsymbol{A}$. In the same paper Paatero et al. [1991] also introduced a three factor NNMA model. However, they did not provide any algorithm to actually compute the presented models. Paatero and Tapper [1993] suggested using alternating least squares (ALS), wherein one holds $\boldsymbol{B}$ fixed while obtaining the optimal $\boldsymbol{C}$ and vice versa, for PMF. Nonnegativity is enforced in an ad-hoc fashion by simply discarding the

entries smaller than zero. NNMA may also be performed with alternating non-negative least squares instead of ALS by using the NNLS algorithm of [Lawson and Hanson, 1974]. While doing ALS or Alternating NNLS, the least squares subroutines can prove to be a bottleneck. Hence, in practice it is better to combine the least square approach with the faster Lee/Seung type updates.

Later [Paatero and Tapper, 1994] proposed another approach for PMF, claiming it to be superior to the one based on ALS. In this approach one iteratively solves $(\boldsymbol{B}+\Delta\boldsymbol{B})\boldsymbol{C} \approx \boldsymbol{A}$ for $\Delta\boldsymbol{B}$ (likewise for $\Delta\boldsymbol{C}$), followed by solving for the coefficient α in $(\boldsymbol{B}+\Delta\boldsymbol{B})(\boldsymbol{C}+\Delta\boldsymbol{C}) \approx \boldsymbol{A}$. However, in practice Paatero and Tapper [1994] recommend neglecting the product $\Delta\boldsymbol{B}\Delta\boldsymbol{C}$ while minimizing $\|\boldsymbol{A} - (\boldsymbol{B} + \Delta\boldsymbol{B})(\boldsymbol{C} + \Delta\boldsymbol{C})\|_{\mathrm{F}}$ to obtain $\Delta\boldsymbol{B}$ and $\Delta\boldsymbol{C}$. In [Paatero, 1997b], yet another algorithm for PMF is introduced under the name PMF2 (the two standing for a two-factor model). However, from its description, the PMF2 algorithm seems to have expanded upon the just described method of [Paatero and Tapper, 1994] and it enforces nonnegativity using logarithmic penalty functions.

Paatero [1997a] went on to consider a three-way factor analytic model (also called PARAFAC, a factor model introduced in 1970 by Harshman [Harshman and Lundy, 1984]). The corresponding algorithm for computing nonnegative factors was called PMF3 and pseudocode is provided in the paper [Paatero, 1997a]. However, the algorithm requires a significant amount of engineering effort to implement and is rather obscure. As an application to the same problem (PARAFAC) Bro and de Jong [1997] presented a faster NNLS algorithm. In order to solve more general "multi-factor" problems Paatero [1999] developed another algorithm called the Multi-linear Engine that allows solving n-way models. The solution is computed using a method based on conjugate gradients.

Other methods based on Least Squares Pauca et al. [2004b] presented an algorithm that combines a constrained least squares problem with the multiplicative update procedures of Lee and Seung [1999]. The procedure solves the least square problem $\min \|\boldsymbol{A} - \boldsymbol{B}\boldsymbol{C}\|_{\mathrm{F}}^2 + \lambda\|\boldsymbol{C}\|_{\mathrm{F}}^2$ using ordinary least squares. The nonnegativity of $\boldsymbol{C}$ is enforced by setting the negative elements to 0. The matrix $\boldsymbol{B}$ is updated using the standard updates (§ 4.3). Langville and Meyer [2005] suggest using alternating constrained least squares for both $\boldsymbol{B}$ and $\boldsymbol{C}$. The λ term influences the sparsity of the resulting solution. Langville and Meyer [2005] also discuss other measures of sparsity that one could incorporate. Other related work dealing with sparsity in NNMA is [Paatero et al., 2002] (controlling rotations by influencing sparsity) and [Heiler and Schnörr, 2006a] (for nonnegative tensors). In a vein similar to alternating NNLS Lawrence et al. [2004a] describe an alternating constrained nonnegative least squares procedure for NNMA built on top of linearly constrained least squares. The brief survey paper [Berry et al., 2006] describes some other approaches.

6.1.2 Lee & Seung and Related Methods

Lee and Seung also developed the problem of NNMA and introduced a specially constrained version of it in the context of unsupervised learning by convex and conic coding [Lee and Seung, 1997]. In that paper, they considered learning encodings so that the reconstruction error over the ensemble of inputs is minimized. The method of choice was an alternating projected gradient approach in which first $\boldsymbol{B}$ is fixed and a gradient descent is done w.r.t. $\boldsymbol{C}$ and vice versa. Nonnegativity constraints were implemented by zeroing out the negative entries and the normalization constraints were enforced using quadratic penalty functions. However, NNMA finally gained popularity after the two papers [Lee and Seung, 1999, 2000] introduced the problem under the name *nonnegative matrix factorization*. Lee and Seung [1999] provided efficient iterative algorithms for NNMA, which were developed and analyzed further in [Lee and Seung, 2000].

Hoyer [2002] added an ℓ_1-norm based regularization term to the original Frobenius norm objective function in order achieve sparser solutions. The resultant NNMA problem, which he named Nonnegative Sparse

Coding, was

$$\min_{\boldsymbol{B},\boldsymbol{C}\geq 0} \|\boldsymbol{A}-\boldsymbol{B}\boldsymbol{C}\|_{\mathrm{F}}^2 + \lambda \sum_{ij} c_{ij},$$

where $\lambda > 0$ is a regularization parameter. Subsequently, Hoyer [2004] extended the enforcement of sparsity by minimizing $\|\boldsymbol{A}-\boldsymbol{B}\boldsymbol{C}\|_{\mathrm{F}}^2$ under additional sparsity constraints of the form sparsity$(\boldsymbol{c}_j^T) = S_C$, sparsity$(\boldsymbol{b}_i) = S_B$. Hoyer [2004] uses the function

$$\text{sparsity}(\boldsymbol{x}) = \frac{\sqrt{n} - \|\boldsymbol{x}\|_1/\|\boldsymbol{x}\|_2}{\sqrt{n}-1},$$

to measure the sparsity and uses a combination of projected gradient descent and Lee/Seung's iterative updates for carrying out the minimization. Evidently, one can use other measures of sparsity (See [Langville and Meyer, 2005], for further examples).

Feng et al. [2002] added additional constraints to the KL-Divergence NNMA problem to model spatial locality in the input matrix $\boldsymbol{A}$. Locality is encouraged by enforcing constraints on $\boldsymbol{B}$, and sparsity by imposing constraints on $\boldsymbol{C}$. The resultant objective function was

$$\mathrm{KL}(\boldsymbol{A};\boldsymbol{B}\boldsymbol{C}) + c_1\mathbf{1}^T\boldsymbol{B}^T\boldsymbol{B}\mathbf{1} - c_2\|\boldsymbol{C}\|_{\mathrm{F}}^2, \tag{6.2}$$

where $c_1, c_2 > 0$ are some constants.

Sajda et al. [2003] modified Lee/Seung's algorithm by forcing small values in $\boldsymbol{C}$ to $\epsilon > 0$, and named their modification cNMF (constrained NMF). They initialized $\boldsymbol{B}$ randomly, and $\boldsymbol{C}$ using a constrained least squares solution. Thereafter, they updated $\boldsymbol{B}$ and $\boldsymbol{C}$ as usual with the exception of clamping down small values in $\boldsymbol{C}$ to the fixed constant ϵ.

Guillamet et al. [2001, 2003] suggest that one should weight the input vectors (columns of $\boldsymbol{A}$) and consider the approximation $\boldsymbol{A}\boldsymbol{W} \approx \boldsymbol{B}\boldsymbol{C}\boldsymbol{W}$, where $\boldsymbol{W}$ is a diagonal matrix of weights such that $\mathrm{Tr}(\boldsymbol{W}) = 1$. They present results for such a modification to the KL-Divergence NNMA problem. Our weighted NNMA described in § 4.5 subsumes this approach.

Szatmáry et al. [2002] perform NNMA that has been augmented with sparse code shrinkage and weight sparsification. The latter two techniques were employed to improve the performance of NNMA. For more on SCS the reader is referred to [Hyvärinen et al., 2001]. Heiler and Schnörr [2006b] use NNMA and cone programming to obtain sparse representations.

The NNMA problem has been extended to nonnegative approximations for tensors. Welling and Weber [2001] derive algorithms similar to the iterative Lee/Seung schemes for minimizing squared and KL-Divergence losses (for tensors). Shashua and Hazan [2005] perform nonnegative tensor factorization by repeated rank-1 approximations, while minimizing a squared loss objective function. They include a proof of convergence of their procedure. Heiler and Schnörr [2006a] study sparseness in the context of NTF.

New methods for minimizing Csiszar's divergence are described by Cichocki et al. [2006b,c]. NNMA using quasi-Newton methods is considered by Zdunek and Cichocki [2006], who apply it to Amari's α-disparity [Amari, 1985]. Cichocki et al. [2006a] also derive other iterative methods for minimizing Amari's α-divergence.

Berry et al. [2006] provide a short survey on the algorithms and applications of NNMA though they mainly focus on the Frobenius norm based NNMA problems.

6.2 Applications

We now enlist some of the numerous applications of NNMA that have appeared in the literature. We have roughly categorized them for easier perusal. Some of the applications are divergent from a traditional machine learning setting, but as the original PMF series of algorithms arose in such applications, we have decided to retain references to them for completeness.

6.2.1 Environmetrics and Chemometrics

Paatero et al. applied the ideas of PMF to environmental data as early as 1991. For a list of references that indicate some of these applications the reader is referred to the original PMF paper [Paatero and Tapper, 1994]. Later Paatero [1999] applied his multi-linear engine to analyze atmospheric emission and pollution data. A paper discussing the application of orthogonal projection approach, alternating least squares and PMF to analyze chromatographic spectral data (which is used to analyze mixtures of chemicals) was presented by Frenich et al. [2000]. The results obtained by these three methods are compared by evaluating measures of dissimilarity between real and estimated spectra (matrix $\boldsymbol{C}$). The authors concluded that in general PMF2 and alternating least squares had little differences in the quality of results, and that PMF2 is a good tool for curve resolution analysis of chromatographic data. Qin et al. [2002] used PMF on a large aerosol database measured in Hong Kong incorporating error estimates through the $\boldsymbol{W}$ matrix.

Paatero et al. [2002] discuss the resolution of the problem of rotational indeterminacy in the PMF (PMF2, PMF3, ME) solutions using a specific two factor model as an example. The conclusions and recommendations of the paper are however, largely empirical in nature. Ramadan et al. [2003] compare PMF and the ME on a data matrix of pollutant concentrations in Phoenix, and they conclude that the ME did not yield significant modeling advantages over PMF2. Sajda et al. [2003] applied their constrained version of NNMA to recovering constituent spectra in 3D chemical shift imaging. They compared their results to Bayesian Spectral Decomposition [Ochs et al., 1999] and suggested that NNMA obtains similar results in orders of magnitude lesser time.

6.2.2 Image Processing and Computer Graphics

In their seminal paper Lee and Seung [1999] demonstrated how one could obtain a parts based representation for image data. That is, the sparse basis vectors (columns of $\boldsymbol{B}$) approximating faces roughly corresponded to individual parts of faces such as lips, noses and eyes. Feng et al. [2002] used their local NNMA algorithm for learning a spatially localized, parts-based representation for images. They compare their method to PCA and NNMA to demonstrate the situations where a spatially localized approach has advantages (such as highly occluded faces during face recognition). Guillamet and Vitrià [2002c] suggest using the Earth Movers Distance as a relevant metric for doing face recognition using NNMA. Other work on face and image processing applications of NNMA by these authors includes [Guillamet et al., 2001, Guillamet and Vitrià, 2002a,b, Guillamet et al., 2003]. Cooper and Foote [2002] applied NNMA to summarizing video and audio data.

Wild et al. [2003] described an application of NNMA to Airborne Visible/Infrared Imaging Spectrometer data. They describe feature extraction using a random initialization of NNMA as well as via an initialization based on a spherical kmeans clustering. Szatmáry et al. [2003] proposed hierarchical image representation using NNMA augmented with sparse code shrinkage preprocessing and applied their methods to the FERET image database. Other image processing work that uses NNMA includes [Kun et al., 2005, Lawrence et al., 2004a,b, Zhang et al., 2004]. The recent article of Spratling [2006] evaluates the empirical performance of some NNMA algorithms for recognizing elementary image features, especially in the presence of occlusion.

Nonnegative tensor factorization (NTF) was used by Welling and Weber [2001] to the decomposition of color images. Shashua and Hazan [Hazan et al., 2005, Shashua and Hazan, 2005] applied NTF to low-rank representation of images, obtaining good parts based representations.

6.2.3 Text analysis

Lee and Seung [1999] applied NNMA to text documents and highlighted the ability of NNMA to tackle semantic issues such as synonymy. Owing to the low-rank approximations produced NNMA is a natural candidate for a clustering procedure. Xu et al. [2003] described clustering experiments with NNMA, wherein they compared NNMA against spectral methods, suggesting that the former can obtain higher accuracy. Xu

et al. [2003] used NNMA for clustering text data. Other related work on clustering and text analysis using NNMA includes [Badea, 2005, Pauca et al., 2004b, Shahnaz et al., 2006]. An application to email surveillance was discussed in [Berry and Browne, 2005],

6.2.4 Blind Source Separation & ICA

Some authors have considered blind source separation by using either nonnegative PCA [Oja and Plumbley, 2003] or ICA [Plumbley, 2002a,b]. Work that directly applies NNMA to blind source separation and ICA includes Cichocki et al. [2006c], Li and Cichocki [2003]. Pauca et al. [2004a] use NNMA and ICA for unmixing data.

6.2.5 Bioinformatics

Recently various data mining techniques have been applied to problems or data sets from biology forming a significant part of the field of bioinformatics. NNMA has had its share of applications. Brunet et al. [2004] apply NNMA to form *metagenes* to infer biological information from cancer-related microarray data. They use the KL-Divergence based NNMA algorithm and also provide heuristic methods for model selection. Kim and Tidor [2003] apply NNMA for performing dimensionality reduction to aid in the identification of subsystems from gene microarray data. They hinged their arguments on the ability to detect local features from the data using NNMA. Other applications include lung cancer prognosis [Inamura et al., 2005], analysis of lung cancer profiles [Fujiwara et al., 2005], sparse NNMA for cancer class discovery [Gao and Church, 2005], among others. Further references that apply NNMA or sparse variants thereof, to gene data are [Badea and Tilivea, 2005, Pascual-Montano et al., 2003, Rao et al., 2004]. Chen et al. [2006] apply their NNMA algorithms to the analysis of data related to Alzheimer's disease.

6.2.6 Miscellaneous applications

NNMA has been applied to problems of a diverse nature. Though we summarized some of the major applications above, there remain numerous other applications. We cannot hope to be exhaustive in our coverage and must thereby satisfy ourselves by being indicative. Hoyer [2002] added sparsity constraints to NNMA and in a later paper [Hoyer, 2003] modeled the receptive fields of the primary visual cortex in mammals. Hoyer's experiments on natural images revealed the usefulness of an NNMA based approach.

Behnke [2003] proposed a variant of NNMA called convolutional NNMA and applied it to a hierarchical approach for extracting speech features. NNMA was combined with a Neural Abstraction Pyramid architecture [Behnke, 1999] and recursively applied to to obtain a hierarchical decomposition of the features.

A somewhat offbeat application to the transcription of polyphonic music via NNMA was attempted by Smaragdis and Brown [2003], who analyzed polyphonic music passages that comprised of notes that exhibit a harmonically fixed spectral profile.

J-H. Ahn and Choi [2004], Lee et al. [2001] apply NNMA to the analysis of matrices obtained via dynamic Positron Emission Tomography (PET). The ability to use a Poisson statistics based noise model for NNMA for PET images is suggested to be one of the benefits of NNMA over traditional Gaussian based methods since PET data comes from a process where the Poisson distribution makes more sense. This motivation also lies behind using an appropriate Bregman divergence for an NNMA problem depending on the assumed underlying nature of the noise distribution.

Other applications include object characterization [Piper et al., 2005], spectral data analysis [Pauca et al., 2005], learning sound dictionaries [Asari, 2005], mining ratio-rules [Hu et al., 2004], and multiway clustering [Badea, 2005, Shashua et al., 2006].

7 Nonnegative Matrix Factorization

For completeness (and to ratify our selection of the name NNMA) we digress briefly to describe the nonnegative matrix factorization problem, i.e., an NNMA problem where an exact factorization of the form $\boldsymbol{A} = \boldsymbol{BC}$ exists. We provide only a smattering of references to this problem, hopefully pointing the interested reader in the correct direction.

Markham [1972] derived necessary and sufficient conditions for a nonnegative matrix $\boldsymbol{A}$ to have a factorization of the form $\boldsymbol{LU}$, where $\boldsymbol{L}$ is nonnegative lower triangular and $\boldsymbol{U}$ is a nonnegative unit upper triangular matrix. He restricted $\boldsymbol{A}$ to the class of matrices that have nonzero principal subminors. This somewhat artificial restriction was lifted in a subsequent paper [Lau and Markham, 1978]. Related work discussing "correct" decomposition into parts may be found in a more recent paper [Donoho and Stodden, 2003]. Markham has also discussed factorizations of completely positive matrices, i.e., matrices all of whose minors are positive [Markham, 1971]. Later Cryer [1973] proved that a matrix $\boldsymbol{A}$ is strictly totally positive iff $\boldsymbol{A} = \boldsymbol{LU}$, where $\boldsymbol{L}$ and $\boldsymbol{U}$ are triangular matrices all of whose non-trivial minors are strictly positive. Other relevant references include [Hannah and Laffey, 1983, Kaykobad, 1987, Li et al., 2004].

Gray and Wilson [1980] provided geometric proofs of the fact that for $n \leq 4$, $n \times n$ nonnegative positive-definite matrices can be factored into $n \times n$ nonnegative factors. They also show that their conditions are not sufficient to guarantee the existence of such factorizations for $n \geq 5$.

Suppose $\boldsymbol{A}$ is an $m \times n$ matrix of rank $r \leq \min(m, n)$. Then, $\boldsymbol{BC}$ is called a *rank factorization* of $\boldsymbol{A}$ if $\boldsymbol{B}$ and $\boldsymbol{C}$ are $m \times r$, $n \times r$ full-rank matrices, and $\boldsymbol{A} = \boldsymbol{BC}$. Of course, for a nonnegative rank factorization (NRF) both $\boldsymbol{B}$ and $\boldsymbol{C}$ are nonnegative. Campbell and Poole [1981] discuss the existence of generalized matrix inverses in terms of NRFs. They also present an algorithm that can compute a NRF of a nonnegative matrix when a nonnegative 1-inverse exists[5]. Thomas [1974] gave a simple characterization when a NRF exists for a given matrix. Wall [1979] discusses rank factorizations of positive operators. Jeter and Pye [1981] prove that if $\boldsymbol{A}$ is weakly monotone [Berman and Plemmons, 1976] then it has a NRF if and only if it possesses an $r \times r$ monomial submatrix. Chen [1984] describes when $\boldsymbol{A}$ has "trivial" or "non-trivial" NRFs.

References

S. Amari. *Differential-Geometric methods in Statistics*. Springer, 1985.

H. Asari. Nonnegative matrix factorization: a possible way to learn sound dictionaries. Preprint, 2005. URL `http://zadorlab.cshl.edu/asari/pdf/nmf.pdf`.

L. Badea. Clustering and Metaclustering with Nonnegative Matrix Decompositions. In J. Gama, R. Camacho, P. B. Brazdil, A. M. Jorge, and L. Torgo, editors, *16th European Conference on Machine Learning*, Porto, Portugal, October 2005. Springer.

L. Badea and D. Tilivea. Sparse factorizations of gene expressions guided by binding data. In *Pacific Symposium on Biocomputing*, volume 10, pages 447–458, 2005.

A. Banerjee, I. S. Dhillon, J. Ghosh, S. Merugu, and D. S. Modha. A Generalized Maximum Entropy Approach to Bregman Co-Clustering and Matrix Approximations. In *ACM SIGKDD International Conference on Knowledge Discovery and Datamining (KDD-2004)*, 2004a.

A. Banerjee, S. Merugu, I. S. Dhillon, and J. Ghosh. Clustering with Bregman Divergences. In *SIAM International Conf. on Data Mining*, Lake Buena Vista, Florida, April 2004b. SIAM.

[5] A matrix $\boldsymbol{X}$ is called a 1-inverse of $\boldsymbol{A}$ if $\boldsymbol{AXA} = \boldsymbol{A}$ and a 2-inverse if $\boldsymbol{XAX} = \boldsymbol{X}$.

A. Banerjee, S. Merugu, I. S. Dhillon, and J. Ghosh. Clustering with Bregman Divergences. *JMLR*, 6(6): 1705–1749, October 2005.

H. H. Bauschke and J. M. Borwein. Legendre functions and the method of random Bregman projections. *Journal of Convex Analysis*, 4:27–67, 1997.

S. Behnke. Hebbian learning and competition in the Neural Association Pyramid. In *IJCNN*, Washington, DC, 1999.

S. Behnke. Discovering hierarchical speech features using convolutional nonnegative matrix factorization. In *International Joint Conference on Neural Networks*, volume 4, pages 2758–2763, Portland, OR, 2003.

A. Berman and R. J. Plemmons. Eight types of matrix monotonicity. *Linear Algebra and its Applications*, 13:115–123, 1976.

M. Berry, M. Browne, A. Langville, P. Pauca, and R. J. Plemmons. Algorithms and applications for approximation nonnegative matrix factorization. *Computational Statistics and Data Analysis*, 2006. Preprint.

M.W. Berry and M. Browne. Email Surveillance Using Nonnegative Matrix Factorization. In *Workshop on Link Analysis, Counterterrorism and Security, SIAM Data Mining*, pages 45–54, April 2005.

S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, March 2004. ISBN 0521833787.

L. M. Bregman. The relaxation method of finding the common point of convex sets and its applications to the solution of problems in convex programming. *U.S.S.R. Computational Mathematics and Mathematical Physics*, 7(3):200–217, 1967.

R. Bro and S. de Jong. A fast non-negativity-constrained least squares algorithm. *Journal of Chemometrics*, 11:393–401, 1997.

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

S. L. Campbell and G. D. Poole. Computing nonnegative rank factorizations. *Linear Algebra and its Applications*, 35:175–182, 1981. ISSN 0024-3795.

Y. Censor and S. A. Zenios. *Parallel Optimization: Theory, Algorithms, and Applications*. Oxford University Press, 1997.

J-C. Chen. The nonnegative rank factorizations of nonnegative matrices. *Linear Algebra and its Applications*, 62:207–217, 1984. ISSN 0024-3795.

Z. Chen, A. Cichocki, and T. M. Rutkowski. Constrained Non-Negative Matrix Factorization Method for EEG Analysis in Early Detection of Alzheimer's Disease. In *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP2006*, Toulouse, France, 2006.

H. Cho, I. S. Dhillon, Y. Guan, and S. Sra. Minimum Sum Squared Residue based Co-clustering of Gene Expression data. In *Proc. 4th SIAM International Conference on Data Mining (SDM)*, pages 114–125, Florida, 2004. SIAM.

A. Cichocki, S. Amari, R. Zdunek, Z. He, and R. Kompass. Extended SMART Algorithms for Non-Negative Matrix Factorization. In *Eighth International Conference on Artificial Intelligence and Soft Computing, ICAISC*, Zakopane, Poland, 2006a.

A. Cichocki, R. Zdunek, and S. Amari. Csiszar's Divergences for Non-Negative Matrix Factorization: Family of New Algorithms. In *6th International Conference on Independent Component Analysis and Blind Signal Separation*, volume Springer LNCS 3889, pages 32–39, Charleston SC, USA, 2006b.

A. Cichocki, R. Zdunek, and S. Amari. New Algorithms for Non-Negative Matrix Factorization in Applications to Blind Source Separation. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Toulouse, France, 2006c.

M. Collins, S. Dasgupta, and R. E. Schapire. A Generalization of Principal Components Analysis to the Exponential Family. In *NIPS 2001*, 2001.

M. Collins, R. Schapire, and Y. Singer. Logistic regression, adaBoost, and Bregman distances. In *Thirteenth annual conference on COLT*, 2000.

M. Cooper and J. Foote. Summarizing video using nonnegative similarity matrix factorization. In *IEEE Multimedia Signal Processing Workshop*, St. Thomas, USVI, December 2002.

C. Cryer. The LU-factorization of totally positive matrices. *Linear Algebra and its Applications*, 7:83–92, 1973.

I. S. Dhillon and S. Sra. Generalized Nonnegative Matrix Approximations with Bregman Divergences. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *NIPS 18*, Cambridge, MA, 2006. MIT Press.

D. Donoho and V. Stodden. When does nonnegative matrix factorization give a correct decomposition into parts? In *Neural Information Processing Systems*, 2003.

T. Feng, S. Z. Li, H-Y. Shum, and H. Zhang. Local nonnegative matrix factorization as a visual representation. In *Proceedings of the 2nd International Conference on Development and Learning*, pages 178–193, Cambridge, MA, June 2002.

A. G. Frenich, M. M. Galera, J. L. M. Vidal, D. L. Massart, J.R. Torres-Lapasió, K. De Braekeleer, J-H. Wang, and P. K. Hopke. Resolution of multicomponent peaks by orthogonal projection approach, positive matrix factorization and alternating least squares. *Analytica Chimica Acta*, 411:145–155, 2000.

T. Fujiwara, S. Ishikawa, Y. Hoshida, K. Inamura, T. Isagawa, M. Shimane, H. Aburatani, Y. Ishikawa, and H. Nomura. Non-Negative Matrix Factorization of Lung Adenocarcinoma Expression Profiles. In *16th International Conference on Genome Informatics*, Yokohama Pacifico, Japan, December 2005.

Y. Gao and G. Church. Improving molecular cancer class discovery through sparse non-negative matrix factorization. *Bioinformatics*, 21(21):3970–3975, 2005.

E. Gaussier and C. Goutte. Relation between PLSA and NMF and implications. In *ACM SIGIR conference on Research and development in information retrieval*, pages 601–602, 2005.

Geoffrey J. Gordon. Generalized^2 linear^2 models. In S. Thrun S. Becker and K. Obermayer, editors, *Advances in Neural Information Processing Systems 15*, pages 577–584, Cambridge, MA, 2003. MIT Press.

L. J. Gray and D. G. Wilson. Nonnegative factorization of positive semidefinite nonnegative matrices. *Linear Algebra and its Applications*, 31:119–127, 1980.

D. Guillamet, M. Bressan, and J. Vitrià. A weighted nonnegative matrix factorization for local representations. In *CVPR*. IEEE, 2001.

D. Guillamet and J. Vitrià. Analyzing non-negative matrix factorization for image classification. In *IEEE International Conference on Pattern Recognition*, volume 2, pages 116–119, 2002a.

D. Guillamet and J. Vitrià. Classifying faces with nonnegative matrix faces. In *CCIA*, Castelló de la Plana, Spain, 2002b.

D. Guillamet and J. Vitrià. Determining a suitable metric when using nonnegative matrix factorization. In *16th International Conference on Pattern Recognition*. IEEE Computer Society, 2002c.

D. Guillamet, J. Vitrià, and B. Schiele. Introducing a weighted nonnegative matrix factorization for image classification. *Pattern Recognition Letters*, 24(14):2447–2454, October 2003. ISSN 0167-8655.

J. Hannah and T. J. Laffey. Nonnegative factorization of completely positive matrices. *Linear Algebra and its Applications*, 55:1–9, 1983. ISSN 0024-3795.

R. A. Harshman and M. E. Lundy. The PARAFAC model for three-way factor analysis and multidimensional scaling. In Law et al., editor, *Research Methods for Multimode Data Analysis*, pages 122–215. Praeger, New York, 1984.

T. Hazan, S. Polak, and A. Shashua. Sparse image coding using non-negative tensor factorization. In *IEEE Conference on Computer Vision*, 2005.

M. Heiler and C. Schnörr. Controlling sparseness in nonnegative tensor factorization. In *ECCV*, 2006a.

M. Heiler and C. Schnörr. Learning Sparse Representations by Non-Negative Matrix Factorization and Sequential Cone Programming. *JMLR*, 2006b. To Appear.

T. Hofmann. Probabilistic latent semantic indexing. In *Proc. ACM SIGIR*. ACM Press, August 1999.

P. O. Hoyer. Non-negative sparse coding. In *Proc. IEEE Workshop on Neural Networks for Signal Processing*, pages 557–565, 2002.

P. O. Hoyer. Modeling receptive fields with nonnegative sparse coding. *Neurocomputing*, 52–54:547–552, 2003.

P. O. Hoyer. Nonnegative matrix factorization with sparseness constraints. *Journal of Machine Learning Research*, 5:1457–1469, 2004.

C. Hu, B. Zhang, S. Yan, Q. Yang, J. Yan, Z. Chen, and W.-Y. Ma. Mining Ratio Rules Via Principal Sparse Non-Negative Matrix Factorization. In *Fourth IEEE International Conference on Data Mining (ICDM'04)*, pages 407–410, 2004.

A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis*. John Wiley, 2001.

K. Inamura, T. Fujiwara, Y. Hoshida, T. Isagawa, M. H. Jones, C. Virtanen, M. Shimane, Y. Satoh, S. Okumura, K. Nakagawa, E. Tsuchiya, S. Ishikawa, H. Aburatani, H. Nomura, and Y. Ishikawa. Two subclasses of lung squamous cell carcinoma with different gene expression profiles and prognosis identified by hierarchical clustering and non-negative matrix factorization. *Oncogene*, 24:7105–7113, June 2005.

J-H. Oh J-H. Ahn, S. Kim and S. Choi. Multiple nonnegative-matrix factorization of dynamic pet images. In *ACCV*, 2004.

M. W. Jeter and W. C. Pye. A note on nonnegative rank factorizations. *Linear Algebra and its Applications*, 38:171–173, 1981. ISSN 0024-3795.

M. Kaykobad. On nonnegative factorization of matrices. *Linear Algebra and its Applications*, 96:27–33, 1987. ISSN 0024-3795.

P. M. Kim and B. Tidor. Subsystem Identification Through Dimensionality Reduction of Large-Scale Gene Expression Data. *Genome Research*, 13:1706–1718, 2003.

B. Kulis, M. Sustik, and I. S. Dhillon. Learning low-rank kernel matrices. In *Proc. 23rd ICML*. 2006. To appear.

W. Kun, Z. Nanning, and L. Weixiang. Natural image matting with nonnegative matrix factorization. In *IEEE International Conference on Image Processing*, volume 2, pages 1186–1189, September 2005.

A. N. Langville and C. D. Meyer. Text mining using the nonnegative matrix factorization. SIAM Southeastern Section Annual Meeting, 2005. Talk.

C. M. Lau and T. L. Markham. Factorization of Nonnegative Matrices–II. *Linear Algebra and its Applications*, 20:51–56, 1978.

J. Lawrence, A. B-Artzi, C. DeCoro, W. Matusik, H. Pfister, R. Ramamoorthi, and S. Rusinkiewicz. Inverse shade trees for non-parametric material representation and editing. In *SIGGRAPH*, 2004a.

J. Lawrence, S. Rusinkiewicz, and R. Ramamoorthi. Efficient BRDF Importance Using a Factored Representation. In *SIGGRAPH*, 2004b.

C. Lawson and R Hanson. *Solving least squares problems*. Prentice-Hall, 1974.

D. D. Lee and H. S. Seung. Unsupervised learning by convex and conic coding. In *NIPS*, pages 515–521. MIT Press, 1997.

D. D. Lee and H. S. Seung. Learning the parts of objects by nonnegative matrix factorization. *Nature*, 401: 788–791, October 1999.

D. D. Lee and H. S. Seung. Algorithms for nonnegative matrix factorization. In *NIPS*, pages 556–562, 2000.

J. S. Lee, D. D. Lee, S. Choi, and D. S. Lee. Application of nonnegative matrix factorization to dynamic positron emission tomography. In *3rd International Conference on Independent Component Analysis and Blind Signal Separation*, San Diego, CA, December 2001.

Y. Li and A. Cichocki. Non-negative matrix factorization and its application in blind sparse source separation with less sensors than sources. In *Proceedings of XII International Symposium on Theoretical Electrical Engineering*, pages 285–288, Warsaw, Poland, 2003.

Y. Li, A. Kummert, and A. Frommer. A linear programming based analysis of the CP-rank of completely positive matrices. *Int. J. Applied Math. Comput. Sci.*, 14(1):25–31, 2004.

T. L. Markham. Factorizations of completely positive matrices. *Proceedings of the Cambridge Philosophical Society*, 69:53–58, 1971.

T. L. Markham. Factorizations of nonnegative matrices. *Proceedings of the American Mathematical Society*, 32(1):45–47, March 1972.

M. F. Ochs, R. S. Stoyanova, F. Arias-Mendoza, and T. R. Brown. A new method for spectral decomposition using a bilinear bayesian approach. *Journal of Magnetic Resonance*, 137:161–176, 1999.

E. Oja and M. Plumbley. Blind separation of positive sources using nonnegative PCA. In *4th International Symposium on Independent Component Analysis and Blind Signal Separation*, Nara, Japan, April 2003.

P. Paatero. A weighted nonnegative least squares algorithm for three-way 'PARAFAC' factor analysis. *Chemometrics and Intelligent Laboratory Systems*, 38:223–242, 1997a.

P. Paatero. Least-squares formulation of robust nonnegative factor analysis. *Chemometrics and Intelligent Laboratory Systems*, 37:23–35, 1997b.

P. Paatero. The multilinear engine—a table-driven least squares program for solving multilinear problems, including the n-way parallel factor analysis model. *Journal of Computational and Graphical Statistics*, 8 (4):854–888, December 1999.

P. Paatero, P. K. Hopke, X-H. Song, and Z. Ramadan. Understanding and controlling rotations in factory analytic models. *Chemometrics and Intelligent Laboratory Systems*, 60:253–264, 2002.

P. Paatero and U. Tapper. Analysis of different modes of factor analysis as least squares fit problems. *Chemometrics and Intelligent Laboratory Systems*, 18(2):183–194, 1993.

P. Paatero and U. Tapper. Positive matrix factorization: A nonnegative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 5(111–126), 1994.

P. Paatero, U. Tapper, P. Aalto, and M. Kulmala. Matrix factorization methods for analysing diffusion battery data. *Journal of Aerosol Science*, 22(Supplement 1):S273–S276, 1991.

A. D. Pascual-Montano, P. Carmona-Saez, M. Chagoyen, and J.M. Carazo. Non-negative matrix factorization for gene expression and scientific texts analysis. In *ISMB*, 2003.

P. Pauca, R. Plemmons, M. Giffin, and K. Hamada. Unmixing spectral data for space objects using independent component analysis and nonnegative matrix factorization. In *Proceedings Amos Technical Conference*, 2004a.

P. Pauca, F. Shahnaz, M. Berry, and R. Plemmons. Text mining using nonnegative matrix factorizations. In *SIAM Data Mining*, 2004b.

V. P. Pauca, J. Piper, and R. J. Plemmons. Nonnegative matrix factorization for spectral data analysis. Preprint, May 2005.

J. Piper, V. P. Pauca, R. J. Plemmons, and M. Giffin. Object Characterization from Spectral Data using Nonnegative Factorization and Information Theory. Preprint, 2005.

M. D. Plumbley. Algorithms for nonnegative independent component analysis. In *Unpublished*, 2002a.

M. D. Plumbley. Conditions for nonnegative independent component analysis. *IEEE Signal Processing letters*, 9(6):177–180, June 2002b.

Y. Qin, K. Oduyemi, and L. Y. Chan. Comparative testing of PMF and CFA models. *Chemometrics and Intelligent Laboratory Systems*, 61:75–87, 2002.

Z. Ramadan, B. Eickhout, X-H. Song, L. M. C. Buydents, and P. K. Hopke. Comparison of positive matrix factorization and multilinear engine for the source apportionment of particulate pollutants. *Chemometrics and Intelligent Laboratory Systems*, 66:15–28, 2003.

N. Rao, S. J. Shepherd, and D. Yao. Extracting characteristic patterns form genome-wide expression data by non-negative matrix factorization. In *IEEE Computational Systems Bioinformatics Conference*, 2004.

R. T. Rockafellar. *Convex Analysis*. Princeton Univ. Press, 1970.

P. Sajda, S. Du, T. Brown, L. Parra, and R. Stoyanova. Recovery of Constituent Spectra in 3D Chemical Shift Imaging using Nonnegative Matrix Factorization. In *4th International Symposium on Independent Component Analysis and Blind Signal Separation*, pages 71–76, Nara, Japan, April 2003.

F. Shahnaz, M. Berry, P. Pauca, and R. Plemmons. Document clustering using nonnegative matrix factorization. *J. on Information Processing and Management*, 42:373–386, 2006.

A. Shashua and T. Hazan. Non-Negative Tensor Factorization with Applications to Statistics and Computer Vision. In *ICML*, 2005.

A. Shashua, R. Zass, and T. Hazan. Multiway Clustering using Super-symmetric Nonnegative Tensor Factorization. In A. Leonardis, H. Bischof, and A. Prinz, editors, *ECCV*, pages 595–608. Springer, 2006.

P. Smaragdis and J. C. Brown. Nonnegative matrix factorization for polyphonic music transcription. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 177–180, New Paltz, NY, October 2003.

M. W. Spratling. Learning image components for object recognition. *Journal of Machine Learning Research*, 7:793–815, 2006.

B. Szatmáry, B. Póczos, J. Eggert, E. Körner, and A. Lőrincz. Nonnegative matrix factorization extended by sparse code shrinkage and weight sparsification algorithms. In *ECAI 2002, Proceedings of the 15th European Conference on Artificial Intelligence*, pages 503–507, Amsterdam, 2002. IOS Press.

B. Szatmáry, G. Szirtes, A. Lőrincz, J. Eggert, and E. Körner. Robust hierarchical image representation using nonnegative matrix factorization with sparse code shrinkage preprocessing. *Pattern Analysis and Applications*, 2003. Accepted.

L. B. Thomas. Rank factorizations of nonnegative matrices. *SIAM Review*, 16(4):393–394, 1974. Problem 73-14.

J. A. Tropp. *Topics in Sparse Approximation*. PhD thesis, The University of Texas at Austin, 2004.

K. Tsuda, G. Rätsch, and M. Warmuth. Matrix Exponential Gradient Updates for On-line Learning and Bregman Projection. In Lawrence K. Saul, Yair Weiss, and Léon Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 1425–1432. MIT Press, Cambridge, MA, 2005.

J. R. Wall. Rank factorizations of positive operators. *Linear and Multilinear Algebra*, 8:137–144, 1979.

M. Welling and M. Weber. Positive tensor factorization. *Pattern Recognition Letters*, 22:1255–1261, 2001.

S. Wild, J. Curry, and A. Dougherty. Motivating nonnegative matrix factorizations. SIAM Linear Algebra Meeting, July 2003.

W. Xu, X. Liu, and Y. Gong. Document clustering based on nonnegative matrix factorization. In *SIGIR'03*, pages 267–273, Toronto, 2003. ACM.

R. Zdunek and A. Cichocki. Non-Negative Matrix Factorization with Quasi-Newton Optimization. In *Eighth International Conference on Artificial Intelligence and Soft Computing, ICAISC*, Zakopane, Poland, 2006.

J. Zhang, L. Wei, Q. Miao, and Y. Wang. Image fusion based on nonnegative matrix factorization. In *International Conference on Image Processing*, volume 2, pages 973–976, 2004.