# MULTIPLICATIVE UPDATES FOR NMF WITH $\beta$-DIVERGENCES UNDER DISJOINT EQUALITY CONSTRAINTS*

VALENTIN LEPLAT†, NICOLAS GILLIS†, AND JÉRÔME IDIER‡

**Abstract.** Nonnegative matrix factorization (NMF) is the problem of approximating an input nonnegative matrix, $V$, as the product of two smaller nonnegative matrices, $W$ and $H$. In this paper, we introduce a general framework to design multiplicative updates (MU) for NMF based on $\beta$-divergences ($\beta$-NMF) with disjoint equality constraints, and with penalty terms in the objective function. By disjoint, we mean that each variable appears in at most one equality constraint. Our MU satisfy the set of constraints after each update of the variables during the optimization process, while guaranteeing that the objective function decreases monotonically. We showcase this framework on three NMF models, and show that it competes favorably with the state of the art: (1) $\beta$-NMF with sum-to-one constraints on the columns of $H$, (2) minimum-volume $\beta$-NMF with sum-to-one constraints on the columns of $W$, and (3) sparse $\beta$-NMF with $\ell_2$-norm constraints on the columns of $W$.

**1. Introduction.** Given a nonnegative matrix $V \in \mathbb{R}_+^{F \times N}$ and a factorization rank $K \ll \min(F, N)$, nonnegative matrix factorization (NMF) aims to compute two nonnegative matrices, $W$ with $K$ columns and $H$ with $K$ rows, such that $V \approx WH$ [23]. Over the last two decades, NMF has shown to be a powerful tool for the analysis of high-dimensional data. The main reason is that NMF automatically extracts sparse and meaningful features from a set of nonnegative data vectors. NMF has been successfully used in many applications such as image processing, text mining, hyperspectral imaging, blind source separation, single-channel audio source separation, clustering, and music analysis; see [15, 7, 5, 28, 13, 16] and the references therein.

To compute $W$ and $H$, the most standard approach is to solve the following optimization problem:

$$(1) \qquad \min_{W \in \mathbb{R}^{F \times K}, H \in \mathbb{R}^{K \times N}} D(V|WH) \quad \text{such that} \quad H \geqslant 0 \quad \text{and} \quad W \geqslant 0,$$

where $D(V|WH) = \sum_{f,n} d(V_{fn}|[WH]_{fn})$ with $d(x|y)$ a measure of distance between two scalars, and $A \geqslant 0$ means that the matrix $A$ is componentwise nonnegative. In

this paper, we focus on $\beta$-NMF for which the measure of fit is the discrete $\beta$-divergence denoted $d_\beta(x|y)$ and defined as

$$
d_\beta\left(x|y\right) = \begin{cases} \frac{1}{\beta(\beta-1)}\left(x^\beta + \left(\beta-1\right)y^\beta - \beta xy^{\beta-1}\right) & \text{for} \quad \beta \in \mathbb{R}\setminus\{0,1\}, \\ x\log\frac{x}{y} - x + y & \text{for} \quad \beta = 1, \\ \frac{x}{y} - \log\frac{x}{y} - 1 & \text{for} \quad \beta = 0. \end{cases}
$$

For $\beta = 2$, $d_2(x|y) = \frac{1}{2}(x-y)^2$, so $D\left(V|WH\right)$ is the halved standard squared Euclidean distance between $V$ and $WH$, that is, the halved squared Frobenius norm $\frac{1}{2}\|V - WH\|_F^2$. For $\beta = 1$ and $\beta = 0$, the $\beta$-divergence corresponds to the Kullback–Leibler (KL) divergence and the Itakura–Saito (IS) divergence, respectively. The error measure should be chosen accordingly with the distribution of the noise assumed on the data. The Frobenius norm assumes i.i.d. Gaussian noise, KL divergence assumes a Poisson distribution, and the IS divergence assumes multiplicative gamma noise; see, for example, [11, 8, 20] and the references therein. In the NMF literature, $\beta$-divergences are the most widely used objective functions.

Most NMF algorithms developed to tackle (1) are based on iterative schemes that alternatively updates the factors $W$ and $H$. At each iteration, the minimization over one factor, $W$ or $H$, is performed with various optimization methods. For $\beta$-divergences, the most popular approach is to use multiplicative updates (MU) which were introduced for NMF in the seminal papers of Lee and Seung [23, 22]. In all applications we are aware of, $\beta$ is always chosen smaller than two. The reason is that, for $\beta > 2$, $\beta$-divergences become more and more sensitive to outliers. Already for $\beta = 2$, it is well known that the squared Frobenius norm is sensitive to outliers. However, the case $\beta = 2$ is particular because the subproblem in $W$ and $H$ are nonnegative least squares problems, that is, convex quadratic problems with Lipschitz continuous gradient. Therefore, highly efficient schemes exist when $\beta = 2$ that outperform the MU; for example, exact block coordinate descent methods [6, 17, 21, 2], or fast gradient methods [18, 19]. In this paper, we focus on the case $\beta < 2$.

In many applications, on top of the nonnegative constraints on the variables, additional constraints are needed to provide a meaningful solution. An instrumental example is the constraint that the entries in each column of $H$ sum to one; this is the so-called sum-to-one constraint that is crucial in blind hyperspectral unmixing; see section 3. Another example is a sum-to-one constraint on the columns of $W$ along with a volume regularizer on $W$. This model leads to identifiability of the factors $W$ and $H$ under mild conditions; see section 4. Most algorithms that deal with such equality constraints do it a posteriori with a projection onto the feasible set, or with a renormalization of the columns of $W$ and the rows of $H$ (that is, replace $W(:,k)$ and $H(k,:)$ by $\alpha_k W(:,k)$ and $H(k,:)/\alpha_k$ for some $\alpha_k > 0$), so that their product $WH$ remains unchanged, and hence $D(V|WH)$ remains unchanged. Such approaches are not ideal:

- Projection requires performing a line-search to ensure the monotonicity of the algorithm, that is, to ensure that the objective does not increase after each iteration, which may be computationally heavy.
- Renormalization of the columns of $W$ and the rows of $H$ is only useful when each constraint applies to a column of $W$ or a row of $H$. It is not applicable, for example, for the sum-to-one constraint on the columns of $H$ mentioned above. Moreover, in the presence of regularization terms in the objective

function, it may destroy the monotonicity of the algorithm.

Another approach is to use parametrization. However, as far as we know, it does not guarantee the monotonicity of the algorithm; see section 3 for more details.

*Outline and contribution.* In this paper, we introduce a general framework to design MU for $\beta$-NMF with disjoint linear equality constraints, and with penalty terms in the objective function. By disjoint, we mean that each variable appears in at most one equality constraint. This framework, presented in section 2, does not resort to projection, renormalization, or parametrization. Our MU satisfy the set of constraints after each update of the variables during the optimization process, while guaranteeing that the objective function decreases monotonically. This framework works as follows:

- First, as for the standard MU for $\beta$-NMF, we majorize the objective function using a separable majorizer, that is, the majorizer is the sum of functions involving a single variable.
- Second, we construct the augmented Lagrangian for the majorizer. Because the majorizer is separable, the problem can be decomposed into independent subproblems involving only variables that occur in the same equality constraint since they are disjoint. For a fixed value of the Lagrange multipliers, we prove that the solution of these subproblems are unique, under mild conditions (Proposition 1). Moreover, it can be written in closed form via MU for specific values of $\beta$ and depending on the regularizer used (this is summarized in Table 1).
- Finally, we prove that, under mild conditions, there is a unique solution for the Lagrange multipliers so that the equality constraints are satisfied (Proposition 2). This allows us to apply the Newton–Raphson method to compute the Lagrange multipliers while guaranteeing quadratic convergence (Proposition 3).

We then showcase this framework on two NMF models, and show that it competes favorably with the state of the art:

1. A $\beta$-NMF model with sum-to-one constraints on the columns of $H$, which we refer to as simplex-structured $\beta$-NMF (section 3) and
2. A minimum-volume $\beta$-NMF model with sum-to-one constraints on the columns of $W$ (section 4).

Finally, section 5 shows that the framework can be extended to the case of quadratic disjoints constraints, which we showcase on sparse $\beta$-NMF with $\ell_2$-norm constraints on the columns of $W$.

**2. General framework to design MU for $\beta$-NMF under disjoint linear equality constraints and penalization.** In this paper, we introduce a general framework to tackle $\beta$-NMF with disjoint linear equality constraints, and with penalty terms in the objective function. Let us first introduce the following notation: given a matrix $A \in \mathbb{R}^{F \times N}$ and a list of indices $\mathcal{K} \subseteq \{(f, n) \mid 1 \leqslant f \leqslant F, 1 \leqslant n \leqslant N\}$, we denote by $A(\mathcal{K})$ the vector of dimension $|\mathcal{K}|$ whose entries are the entries of $A$ corresponding to the indices within $\mathcal{K}$. Let us introduce $\mathcal{K}_i$ $(1 \leqslant i \leqslant I)$ and $\mathcal{B}_j$ $(1 \leqslant j \leqslant J)$ to be disjoint sets of indices for the entries of $W$ and $H$, respectively, that is,

- $\mathcal{K}_i \subseteq \{(f, k) \mid 1 \leqslant f \leqslant F, 1 \leqslant k \leqslant K\}$ for $i = 1, 2, \ldots, I$,
- $\mathcal{B}_j \subseteq \{(k, n) \mid 1 \leqslant k \leqslant K, 1 \leqslant n \leqslant N\}$ for $j = 1, 2, \ldots, J$,
- $\mathcal{K}_u \cap \mathcal{K}_v = \varnothing$ for all $1 \leqslant u, v \leqslant I$ and $u \neq v$,
- $\mathcal{B}_p \cap \mathcal{B}_q = \varnothing$ for all $1 \leqslant p, q \leqslant J$ and $p \neq q$.

We now define penalized $\beta$-NMF with disjoint linear equality constraints as follows:

(2)
$$\min_{W\in\mathbb{R}_+^{F\times K},H\in\mathbb{R}_+^{K\times N}} \quad D_\beta\left(V|WH\right) + \lambda_1\Phi_1(W) + \lambda_2\Phi_2(H)$$

$$\text{such that} \qquad \boldsymbol{\alpha}_i^T W\left(\mathcal{K}_i\right) = b_i \text{ for } 1 \leqslant i \leqslant I,$$

$$\boldsymbol{\gamma}_j^T H\left(\mathcal{B}_j\right) = c_j \text{ for } 1 \leqslant j \leqslant J,$$

where

- the penalty functions $\Phi_1(W)$ and $\Phi_2(H)$ are lower bounded and admit a particular upper approximation; see Assumption 1.
- $\lambda_1$ and $\lambda_2$ are the penalty weights (nonnegative scalars).
- $\boldsymbol{\alpha}_i \in \mathbb{R}_{++}^{|\mathcal{K}_i|}$ $(1 \leqslant i \leqslant I)$ and $\boldsymbol{\gamma}_j \in \mathbb{R}_{++}^{|\mathcal{B}_j|}$ $(1 \leqslant j \leqslant J)$ are vectors with positive entries. Note that if $\boldsymbol{\alpha}_i$ or $\boldsymbol{\gamma}_j$ contains zero entries, the corresponding indices can be removed from $\mathcal{K}_i$ and $\mathcal{B}_j$.
- $b_i$ $(1 \leqslant i \leqslant I)$ and $c_j$ $(1 \leqslant j \leqslant J)$ are positive scalars.

As for most NMF algorithms, we propose to resort to a block coordinate descent (BCD) framework to solve problem (2): at each iteration we tackle two subproblems separately; one in $W$ and the other in $H$. The subproblems in $W$ and $H$ are essentially the same, by symmetry of the model, since transposing the relation $X \approx WH$ gives $X^T \approx H^T W^T$. Hence, we may focus on solving the subproblem in $H$ only, namely

(3)
$$\min_{H\in\mathbb{R}_+^{K\times N}} D_\beta\left(V|WH\right) + \lambda_2\Phi_2(H) \quad \text{such that} \quad \boldsymbol{\gamma}_j^T H\left(\mathcal{B}_j\right) = c_j \text{ for } 1 \leqslant j \leqslant J.$$

In order to solve (3), we will design MU based on the majorization-minimization (MM) framework [31], which is the standard in the NMF literature; see [12] and the references therein. Let us briefly recall the high-level ideas to obtain MU via MM. Let us consider the general problem

$$\min_{h\in\mathcal{H}} f(h).$$

Given an initial iterate $\widetilde{h} \in \mathcal{H}$, MM generates a new iterate $\hat{h} \in \mathcal{H}$ that is guaranteed to decrease the objective function, that is, $f\big(\hat{h}\big) \leqslant f\big(\widetilde{h}\big)$. To do so, it uses the following two steps:

- Majorization: find a function that is an upper approximation of the objective and is tight at the current iterate, which is referred to as a majorizer. More precisely find a function $g\big(h|\widetilde{h}\big)$ such that

$$\text{(i) } g\big(\widetilde{h}|\widetilde{h}\big) = f\big(\widetilde{h}\big) \quad \text{and} \quad \text{(ii) } g\big(h|\widetilde{h}\big) \geqslant f(h) \text{ for all } h \in \mathcal{H}.$$

- Minimization: minimize the majorizer, that is, solve $\min_{h\in\mathcal{H}} g\big(h|\widetilde{h}\big)$ approximately or exactly, to obtain the next iterate $\hat{h} \in \mathcal{H}$ which is such that (iii) $g\big(\hat{h}|\widetilde{h}\big) \leqslant g\big(\widetilde{h}|\widetilde{h}\big)$. This guarantees the objective function to decrease at each step of this iterative process since

$$f\big(\hat{h}\big) \underset{\text{(ii)}}{\leqslant} g\big(\hat{h}|\widetilde{h}\big) \underset{\text{(iii)}}{\leqslant} g\big(\widetilde{h}|\widetilde{h}\big) \underset{\text{(i)}}{=} f\big(\widetilde{h}\big).$$

The MU for NMF are obtained using MM where the majorizer $g$ is chosen separable, that is, $g\big(h|\widetilde{h}\big) = \sum_{i=1} g_i\big(h_i|\widetilde{h}_i\big)$ for some well-chosen univariate functions $g_i$'s; see (4) in the next section. This choice typically makes the minimization of $g$ admit a closed-form solution which is multiplicative, that is, it has the form $\hat{h} = \widetilde{h} \odot c\big(\widetilde{h}\big)$, where $\odot$

is the componentwise product, and $c(\widetilde{h})$ is a nonnegative vector that depends on $\widetilde{h}$. We will encounter several examples later in this paper.

To summarize, to derive MU for (3), we will follow the MM framework. We first provide a majorizer for the objective of (3) in section 2.1. This majorizer has the property to be separable in each entry of $H$. In order to handle the equality constraints, we introduce Lagrange dual variables in section 2.2, and explain how they can be computed efficiently. This allows us to derive general MU in section 2.3 in the case of non-penalized $\beta$-NMF under disjoint linear equality constraints. This is showcased on simplex-structured $\beta$-NMF in section 3. In section 4, we will illustrate on minimum-volume KL-NMF how to derive MU in the presence of penalty terms.

**2.1. Separable majorizer for the objective function.** Let us derive a majorizer for $\Psi(H) := D_\beta(V|WH) + \lambda\Phi(H)$, that is, a function $G(H|\widetilde{H})$ satisfying (i) $G(H|\widetilde{H}) \geqslant \Psi(H)$ for all $H$, and (ii) $G(\widetilde{H}|\widetilde{H}) = \Psi(\widetilde{H})$. Note that, to simplify the presentation, we denote $\Phi_2(H) = \Phi(H)$ and $\lambda = \lambda_2$. To do so, let us analyze each term of $\Psi(H)$ independently.

*Majorizing $D_\beta(V|WH)$.* The first term $D_\beta(V|WH)$ can be decoupled into $N$ independent terms, one for each column $\boldsymbol{h}_n$ of $H$, that is, $D_\beta(V|WH) = \sum_{n=1}^{N} D_\beta(\boldsymbol{v}_n|W\boldsymbol{h}_n)$, where $\boldsymbol{v}_n$ denotes the $n$th column of matrix $V$. Let us focus on a specific column of $H$, denoted $\boldsymbol{h} \in \mathbb{R}_+^K$, and the corresponding column of $V$, denoted $\boldsymbol{v} \in \mathbb{R}_+^K$. We majorize $D_\beta(\boldsymbol{v}|W\boldsymbol{h}) = \sum_{f=1}^{F} d_\beta(v_f|(W\boldsymbol{h})_f)$ following the methodology introduced in [12], which consists of applying a convex-concave procedure [34] to $d_\beta$, as presented in Appendix A. The resulting upper bound is given by

$$(4) \quad d_\beta(v_f|(W\boldsymbol{h})_f) \leqslant \sum_{k=1}^{K} \frac{w_{fk}\widetilde{h}_k}{\widetilde{v}_f}\breve{d}\left(v_f\Big|\widetilde{v}_f\frac{h_k}{\widetilde{h}_k}\right) + \widehat{d'}(v_f|\widetilde{v}_f)\sum_{k=1}^{K} w_{fk}(h_k - \widetilde{h}_k) + \widehat{d}(v_f|\widetilde{v}_f),$$

where $w_{fk}$ denotes the entry of matrix $W$ at position $(f,k)$, $\widetilde{v}_f := (W\widetilde{\boldsymbol{h}})_f$ denotes the $f$th entry of $\widetilde{\boldsymbol{v}}$, and $\widehat{d}$ and $\breve{d}$ are the concave and convex parts of $d$, respectively.

*Majorizing $\Phi(H)$.* For the second term $\Phi(H)$, we rely on the following assumption for $\Phi$.

ASSUMPTION 1. *The function $\Phi : \mathbb{R}_+^{K\times N} \mapsto \mathbb{R}$ is lower bounded, and for any $\widetilde{H} \in \mathbb{R}_+^{K\times N}$ there exists constants $L_{kn}$ ($1 \leqslant k \leqslant K, 1 \leqslant n \leqslant N$) such that the inequality*

$$(5) \qquad \Phi(H) \leqslant \Phi(\widetilde{H}) + \left\langle \nabla\Phi(\widetilde{H}), H - \widetilde{H} \right\rangle + \sum_{k,n} \frac{L_{kn}}{2}(H_{kn} - \widetilde{H}_{kn})^2$$

*is satisfied for all $H \in \mathbb{R}_+^{K\times N}$. (Note that the constants $L_{kn}$ may depend on $\widetilde{H}$. This will be the case, for example, in section 4.)*

Let us mention two important classes of functions satisfying Assumption 1.
1. Smooth concave functions that are lower bounded on the nonnegative orthant. For such functions, we can take $L_{kn} = 0$ for all $k, n$ since they are upper approximated by their first-order Taylor approximation. Note that, in this case,

$$(6) \qquad\qquad\qquad \nabla\Phi(\widetilde{H}) \geqslant 0.$$

Otherwise we would have $\lim_{y\to\infty} \Phi(H + y\boldsymbol{e}_i\boldsymbol{e}_j^T) = -\infty$, where $\boldsymbol{e}_i$ is the $i$th unit vector, which would contradict the fact that $\Phi$ is bounded from below.

This observation will be useful in the proof of Proposition 1 and is only valid for the special case $L_{kn} = 0$ for all $k, n$.

Examples of such penalty functions include the sparsity-promoting regularizers $\Phi(H) = \|H\|_p^p = \sum_{k,n} H(k,n)^p$ for $0 < p \leqslant 1$ since $H \geqslant 0$.

2. Lower-bounded functions with Lipschitz continuous gradient for which (5) follows from the descent lemma [3].

Examples of such penalty functions include any smooth convex functions; for example any quadratic penalty, such as $\|AH - B\|_2^2$ for some matrices $A$ and $B$ in which case $L_{kn} = \sigma_1(A)^2$ for all $k, n$. We will encounter another example later in the paper, namely $\mathrm{logdet}\left(HH^\top + \delta I\right)$ for $\delta > 0$ which allows one to minimize the volume of the rows of $H$; see section 4 for details (Note that we will use this regularizer for $W$.)

*Majorizing* $\Psi(H)$. Combining (4) and (5), we can construct a majorizer for $\Psi(H)$. Since both (4) and (5) are separable in each entry of $H$, their combination is also separable into a sum of $K \times N$ componentwise majorizers, up to an additive constant:

$$(7) \qquad G\big(H|\widetilde{H}\big) = \sum_{n=1}^{N} \sum_{k=1}^{K} g\big(h_{kn}|\widetilde{H}\big) + C\big(\widetilde{H}\big),$$

where

$$(8) \qquad g\big(h_{kn}|\widetilde{H}\big) = \sum_{f=1}^{F} \frac{w_{fk}\widetilde{h}_{kn}}{\widetilde{v}_{fn}} \breve{d}\left(v_{fn}|\widetilde{v}_{fn}\frac{h_{kn}}{\widetilde{h}_{kn}}\right) + a_{kn}h_{kn}^2 + p_{kn}h_{kn},$$

$$C\big(\widetilde{H}\big) = \sum_{n=1}^{N} \sum_{f=1}^{F} \left(\widehat{d}(v_{fn}|\widetilde{v}_{fn}) - \sum_{k=1}^{K} \widehat{d}'\big(v_{fn}|\widetilde{v}_{fn}\big)w_{fk}\widetilde{h}_{kn}\right) + a_{kn}\widetilde{h}_{kn}^2,$$

with $a_{kn} = \lambda\frac{L_{kn}}{2}$, and

$$p_{kn} = \sum_{f=1}^{F} w_{fk}\widehat{d}'\big(v_{fn}|\widetilde{v}_{fn}\big) + \lambda\left(\frac{\partial\Phi}{\partial h_{kn}}\big(\widetilde{H}\big) - L_{kn}\widetilde{h}_{kn}\right).$$

**2.2. Dealing with equality constraints via Lagrange dual variables.** In the previous section, we derived a majorizer for $\Psi(H)$, $G\big(H|\widetilde{H}\big)$, which is separable in each entry of $H$. Without the equality constraints, we could then compute closed-form solutions to univariate problems to minimize $G\big(H|\tilde{H}\big)$ to obtain the standard MU for NMF as in [12].

However, in problem (3), the entries of $H$ in the subsets $\mathcal{B}_j$ are not independent as they are linked with the equality constraints $\boldsymbol{\gamma}_j^T H(\mathcal{B}_j) = c_j$ for $j = 1, 2, \ldots, J$. In fact, to minimize the majorizer under the equality constraints, we need to solve

$$(9) \qquad \min_{H \in \mathbb{R}_+^{K \times N}} G\big(H|\widetilde{H}\big) \quad \text{such that} \quad \boldsymbol{\gamma}_j^T H(\mathcal{B}_j) = c_j \text{ for } 1 \leqslant j \leqslant J.$$

The variables in different sets $\mathcal{B}_j$ can be optimized independently, as they do not interact in the majorizer nor in the constraints. Note that, for the entries of $H$ that do not appear in any constraints, the standard MU [12] can be used. For simplicity, let us fix $j$ and denote $\mathcal{B} = \mathcal{B}_j$, $Q = |\mathcal{B}|$, $\boldsymbol{y} = H(\mathcal{B}) \in \mathbb{R}_+^Q$, $\boldsymbol{\gamma} = \boldsymbol{\gamma}_j \in \mathbb{R}_{++}^Q$, and $c = c_j > 0$. The problems we need to solve have the form

$$(10) \qquad \min_{\boldsymbol{y} \in \mathcal{Y}} G\big(\boldsymbol{y}|\widetilde{H}\big),$$

where $\mathcal{Y} = \left\{ \boldsymbol{y} \in \mathbb{R}_+^Q \mid \boldsymbol{\gamma}^T \boldsymbol{y} = c \right\}$ and

$$(11) \qquad G\big(\boldsymbol{y}|\widetilde{H}\big) = \sum_{(k,n) \in \mathcal{B}} g\big(h_{kn}|\widetilde{H}\big),$$

where the componentwise majorizers $g\big(h_{kn}|\widetilde{H}\big)$ are defined by (8). Let us introduce a convenient notation: for $q = 1, 2, \ldots, Q$, we denote by $(k(q), n(q))$ the $q$th pair belonging to $\mathcal{B}$. Hence the Lagrangian function of (11) can be written as

$$(12) \qquad G^\mu\big(\boldsymbol{y}|\widetilde{H}\big) = G\big(\boldsymbol{y}|\widetilde{H}\big) - \mu(\boldsymbol{\gamma}^T \boldsymbol{y} - c) = \mu c + C\big(\widetilde{H}\big) + \sum_{q=1}^Q g^\mu\big(y_q|\widetilde{H}\big),$$

where

$$g^\mu\big(y_q|\widetilde{H}\big) = g\big(y_q|\widetilde{H}\big) - \mu\gamma_q\, y_q$$

$$(13) \qquad = \sum_{f=1}^F \frac{w_{fk(q)}\widetilde{y}_q}{\widetilde{v}_{fn(q)}} \breve{d}\left(v_{fn(q)}|\widetilde{v}_{fn(q)}\frac{y_q}{\widetilde{y}_q}\right) + a_q y_q^2 + (p_q - \mu\gamma_q)y_q,$$

$$(14) \qquad p_q = \sum_{f=1}^F w_{fk(q)}\widehat{d'}\big(v_{fn(q)}|\widetilde{v}_{fn(q)}\big) + \lambda\left(\frac{\partial \Phi}{\partial y_q}\big(\widetilde{H}\big) - L_{k(q)n(q)}\widetilde{y}_q\right),$$

and $\mu \in \mathbb{R}$. Note that $G^\mu$ is separable, as is $G$, because the term $\boldsymbol{\gamma}^T \boldsymbol{y}$ is linear.

Assume for now that the Lagrangian multiplier $\mu$ is known, and let us minimize $G^\mu\big(\boldsymbol{y}|\widetilde{H}\big)$ on $(0, \infty)^Q$. Such a problem is separable under the form of $Q$ subproblems, consisting in minimizing univariate functions $g^\mu\big(\cdot|\widetilde{H}\big)$ separately over $(0, \infty)$. We now show in Proposition 1 that, under mild conditions, each subproblem admits a unique solution over $(0, \infty)$.

PROPOSITION 1. *Let $q \in \{1, 2, \ldots, Q\}$. Assume that $\beta < 2$ and $\widetilde{y}_q, v_{fn(q)}, w_{fk(q)} > 0$ for all $f$. Moreover, when $\beta \leqslant 1$, assume that $\mu < \frac{p_q}{\gamma_q}$ for all $q$ such that $a_q = 0$. Then there exists a unique minimizer $y_q^\star(\mu)$ of $g^\mu\big(y_q|\widetilde{H}\big)$ in $(0, \infty)$.*

*Proof.* According to Proposition 4 (see Appendix A), each $g^\mu$ is $C^\infty$ and strictly convex on $(0, \infty)$, so its infimum is uniquely attained in the closure of $(0, \infty)$. We have to prove that it is neither reached at 0 nor at $\infty$. On the one hand, from (13), we have

$$(15) \qquad (g^\mu)'\big(y_q|\widetilde{H}\big) = \sum_{f=1}^F w_{fk(q)}\breve{d'}\left(v_{fn(q)}|\widetilde{v}_{fn(q)}\frac{y_q}{\widetilde{y}_q}\right) + 2a_q y_q + p_q - \gamma_q\mu$$

and, for any $\beta < 2$ and any $x > 0$,

$$\lim_{y \to 0^+} \breve{d'}(x|y) = -\infty,$$

so $\lim_{y_q \to 0^+} (g^\mu)'\big(y_q|\widetilde{H}\big) = -\infty$, which ensures that the infimum is not reached at 0. On the other hand,

$$(16) \qquad \lim_{y \to \infty} \breve{d'}(x|y) = \begin{cases} 0 & \text{if } \beta \leqslant 1, \\ \infty & \text{otherwise.} \end{cases}$$

According to (15) and (16), the distinction must be made between two cases:

- If $a_q > 0$ or $\beta \in (1,2)$: $\lim_{y_q \to \infty} (g^\mu)'(y_q|\widetilde{H}) = \infty$, so the infimum is reached for a finite $y_q$.
- If $a_q = 0$ and $\beta \leqslant 1$: $\lim_{y_q \to \infty} (g^\mu)'(y_q|\widetilde{H}) = p_q - \gamma_q \mu$, so the same conclusion holds if $\mu < \frac{p_q}{\gamma_q}$. $\qquad\qquad\square$

We just proved that, under mild conditions, each $g^\mu$ has a unique minimizer over $(0, \infty)$. However, we assumed that the value of $\mu$ is fixed. Now given $\boldsymbol{y}^\star(\mu) = \left[y_1^\star(\mu), \ldots, y_Q^\star(\mu)\right]^T$, let us show that the solution to $\boldsymbol{\gamma}^T \boldsymbol{y}^\star(\mu) = c$ is unique. The corresponding value of $\mu$, which we denote $\mu^\star$, provides the minimizer $\boldsymbol{y}^\star(\mu^\star)$ of $G^\mu(\boldsymbol{y}|\widetilde{H})$ that satisfies the linear constraint $\boldsymbol{\gamma}^T \boldsymbol{y}^\star(\mu^\star) = c$. Moreover, $\mu^\star$ naturally fulfills $\mu^\star < \frac{p_q}{\gamma_q}$ for all $q$ when $\beta \leqslant 1$ and $a_q = 0$, as required in Proposition 1.

PROPOSITION 2. *Assume that $\beta < 2$ and $\widetilde{y}_q, v_{fn(q)}, w_{fk(q)} > 0$ for all $q, f$. Then the scalar equation $\boldsymbol{\gamma}^T \boldsymbol{y}^\star(\mu) = c$ in the variable $\mu$ admits a unique solution $\mu^\star$ in $(-\infty, t)$, where*

$$(17) \qquad t = \min_{1 \leqslant q \leqslant Q} t_q, \quad \text{where } t_q = \begin{cases} \frac{p_q}{\gamma_q} & \text{if } \beta \leqslant 1 \text{ and } a_q = 0, \\ \infty & \text{otherwise}, \end{cases}$$

*so that $\boldsymbol{y}^\star(\mu^\star) \in (0, \infty)^Q$ is the unique solution to problem (10).*

*Proof.* Under the conditions of Proposition 1, $g^\mu(y_q|\widetilde{H})$ has a unique minimizer $y_q^\star(\mu)$ for each $j$. By the first-order optimality condition, $y_q^\star(\mu)$ is a solution of $(g^\mu)'(y_q|\widetilde{H}) = 0$ or equivalently, by (15), a solution of $\gamma_q^{-1} g'(y_q|\widetilde{H}) = \mu$ over $(0, \infty)$ where

$$(18) \qquad \gamma_q^{-1} g'(y_q|\widetilde{H}) = \gamma_q^{-1} \sum_{f=1}^{F} w_{fk(q)} \breve{d}'\left(v_{fn(q)}|\widetilde{v}_{fn(q)} \frac{y_q}{\widetilde{y}_q}\right) + 2\frac{a_q}{\gamma_q} y_q + \frac{p_q}{\gamma_q}$$

is strictly increasing on $(0, \infty)$ (since $g$ is strictly convex) and one-to-one from $(0, \infty)$ to an open interval $T_q = (t_q^-, t_q^+)$, where

$$(19) \qquad\qquad t_q^- = \lim_{y_q \to 0} g'(y_q|\widetilde{H}) = -\infty,$$

$$(20) \qquad\qquad t_q^+ = \lim_{y_q \to \infty} g'(y_q|\widetilde{H}) = t_q.$$

Moreover, $p_q \geqslant 0$ if $a_q = 0$ (then $L = 0$) and $\beta \leqslant 1$ according to (6) and (14). As a consequence, $\gamma_q^{-1} g'(y_q^\star|\widetilde{y}_q) = \mu$ is equivalent to

$$(21) \qquad\qquad y_q^\star(\mu) = (g')^{-1}(\gamma_q \mu),$$

where $\mu \in T_q$, and $(g')^{-1}$ denotes the inverse function of $g'$.

Coming back to the multivariate problem (10), we must find a value $\mu^\star$ of the Lagrangian multiplier such that the constraint $\boldsymbol{\gamma}^T \boldsymbol{y}^\star(\mu) = c$ is satisfied. Given (21), $\mu^\star$ is a solution of

$$(22) \qquad\qquad \sum_{q=1}^{Q} \gamma_q (g')^{-1}(\gamma_q \mu) = c.$$

Each $g'(y_q|\widetilde{H})$ being strictly increasing on $(0, \infty)$, $(g')^{-1}(\gamma_q \mu)$ is also strictly increasing (from $T_q$ to $(0, \infty)$), which is a direct consequence of $(f^{-1})' = \frac{1}{f' \circ f^{-1}}$, where $f$ is any

strictly increasing function on some interval. Finally, $\sum_{q=1}^{Q} \gamma_q (g')^{-1}(\gamma_q \mu)$ is strictly increasing from $\cap_{j=1}^{J} T_q = (-\infty, t)$ to $(0, \infty)$, with $t \geqslant 0$. Therefore, the solution $\mu^\star$ is unique.                                                                                    □

Proposition 2 shows that the optimal Lagrangian multiplier is the unique solution of (22). Finding the solution of (22) is equivalent to finding the root of a function $r(\mu)$. We propose hereafter to use a Newton–Raphson method to compute $\mu^\star$, and show that this method generates a sequence of iterates $\mu_n$ that converges towards $\mu^\star$ at a quadratic speed.

PROPOSITION 3. *Assume that $\beta < 2$ and $\widetilde{y}_q, v_{fn(q)}, w_{fk(q)} > 0$ for all $q, f$. Let*

$$r(\mu) = \sum_{q=1}^{Q} \gamma_q (g')^{-1}\big(\gamma_q \mu\big) - c$$

*for $\mu \in (-\infty, t)$, where $t$ is defined in (17), and denote $\mu^\star$ to be the unique solution of $r(\mu) = 0$. From any initial point $\mu_0 \in (\mu^\star, t)$, Newton–Raphson's iterates*

$$\mu_{n+1} = \mu_n - \frac{r(\mu_n)}{r'(\mu_n)}$$

*decrease towards $\mu^\star$ at a quadratic speed.*

*Proof.* We already know that $r$ is strictly increasing from $(-\infty, t)$ to $(0, \infty)$. Let us show that $r$ is also strictly convex. According to the third item of Proposition 4 in Appendix A, $\breve{d}''(x|y)$ is completely monotonic, so it is strictly decreasing in $y$. Equivalently, $\breve{d}'(x|y)$ is strictly concave in $y$, and each $g'$ is also strictly concave according to (18). Since the inverse of a strictly increasing, strictly concave function $f$ is strictly increasing and strictly convex, which is a direct consequence of $(f^{-1})'' = -\frac{f'' \circ f^{-1}}{(f' \circ f^{-1})^3}$, then each $(g')^{-1}$ is strictly convex, and finally, $r$ is strictly convex.

For any $\mu_0 \in (\mu^\star, t)$, we have $r(\mu_0) > 0$, so $\mu_1 = \mu_0 - \frac{r(\mu_0)}{r'(\mu_0)} < \mu_0$. We also have $\mu_1 > \mu^\star$ as a consequence of the strict convexity of $r$. By immediate recurrence, we obtain that $\mu_n$ is a decreasing series that converges towards $\mu^\star$. According to [29], it converges at a quadratic speed since $|r'|$ and $|r''|$ are bounded away from 0 in $[\mu^\star, \mu_0]$.                                                                                  □

*Discussion.* At this point, we have derived an optimization framework to tackle problem (10). The optimal Lagrangian multiplier value is determined before each majorization-minimization update using a Newton–Raphson algorithm. However, such a formal solution is implementable if and only if each $y_q^\star(\mu)$ can be actually computed as the minimizer of $g^\mu\big(y_q|\widetilde{H}\big)$ in $(0, \infty)$. In some cases, computing $y_q^\star(\mu)$ is equivalent to extracting the roots of a polynomial of a degree lesser than or equal to four, which is possible in closed form. In other cases, we have to solve a polynomial equation of degree larger than four, or even an equation that is not polynomial. Table 1 indicates the cases where a closed-form solution is available, and hence when our framework can be efficiently implemented.

We observe that, without penalization, the polynomial equation is of degree one, and hence always admit a closed form. This particular case is discussed in the next section, which we will exemplify in section 3 with $\beta$-NMF with sum-to-one constraints on the columns of $H$. In section 4, we will present an important example with $L_{kn} > 0$ for all $k, n$ and $\beta = 1$, namely minimum-volume KL-NMF.

TABLE 1

*Cases where* (21) *can be computed in closed form are indicated by the degree of the corresponding polynomial equation. Otherwise, the symbol* ⅄ *is used. The constants* $L_{kn}$ *are the ones needed in Assumption 1 for the penalization functions* $\Phi_1(H)$ *and* $\Phi_2(W)$*; see* (5)*.*

| | $\beta \in (-\infty, 1)\backslash\{0\}$ | $\beta = 0$ | $\beta = 1$ | $\frac{5}{4}$ | $\frac{4}{3}$ | $\frac{3}{2}$ | Other |
|---|---|---|---|---|---|---|---|
| | | | | | $\beta \in (1,2)$ | | |
| No penalization | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| $L_{kn} = 0$ for all $k, n$ | 1 | 1 | 1 | 3 | 4 | 2 | ⅄ |
| $L_{kn} > 0$ for some $k, n$ | ⅄ | 3 | 2 | ⅄ | ⅄ | 3 | ⅄ |

**2.3. MU for $\beta$-NMF with disjoint linear equality constraints without penalization.** In this section, we derive an algorithm based on the general framework presented in the previous section to tackle the $\beta$-NMF problem under disjoint linear equality constraints without penalization, that is, problem (2) with $\lambda_1 = \lambda_2 = 0$. We consider this simplified case here as it allows one to provide explicit MU for any value of $\beta < 2$; see the row "No penalization" of Table 1. These updates satisfy the constraints after each update of $W$ or $H$, and monotonically decrease the objective function $D_\beta(V|WH)$.

Let us then consider the subproblem of (2) over $H$ when $W$ is fixed and with $\lambda_2 = 0$, that is,

$$(23) \qquad \min_{H \in \mathbb{R}_+^{K \times N}} D_\beta(V|WH) \quad \text{such that} \quad \boldsymbol{\gamma}_j^T H(\mathcal{B}_j) = c_j \text{ for } 1 \leqslant j \leqslant J.$$

Let us follow the framework presented above. First, an auxiliary function, which we denote $G(H|\widetilde{H})$, is constructed at the current iterate $\widetilde{H}$ so that it majorizes the objective for all $H$ and is defined as follows:

$$(24) \qquad \begin{aligned} G(H|\widetilde{H}) &= \sum_{f,n}\left[\sum_k \frac{w_{fk}\widetilde{h}_{kn}}{\widetilde{v}_{fn}}\breve{d}\left(v_{fn}\Big|\widetilde{v}_{fn}\frac{h_{kn}}{\widetilde{h}_{kn}}\right)\right] \\ &+ \left[\widehat{d}'(v_{fn}|\widetilde{v}_{fn})\sum_k w_{fk}(h_{kn}-\widetilde{h}_{kn}) + \widehat{d}(v_{fn}|\widetilde{v}_{fn})\right], \end{aligned}$$

where $\breve{d}(.|.)$ and $\widehat{d}(.|.)$ are given in Appendix A. Second, we need to minimize $G(H|\widetilde{H})$ while imposing the set of linear constraints $\boldsymbol{\gamma}_j^T H(\mathcal{B}_j) = c_j$. The Lagrangian function of $G$ is given by

$$(25) \qquad G^\mu(H|\widetilde{H}) = G(H|\widetilde{H}) - \sum_j^J \left[\mu_j\left(\boldsymbol{\gamma}_j^T H(\mathcal{B}_j) - c_j\right)\right],$$

where $\mu_j$ are the Lagrange multipliers associated to each linear constraint $\boldsymbol{\gamma}_j^T H(\mathcal{B}_j) = c_j$. We observe that $G^\mu$ in (25) is a separable majorizer in the variables $H$ of the Lagrangian function $D_\beta(V|WH) - \sum_j^J \left[\mu_j\left(\boldsymbol{\gamma}_j^T H(\mathcal{B}_j) - c_j\right)\right]$. Due to the disjointness of each subset of variables $\mathcal{B}_j$ (25), we only consider the optimization over one specific subset $\mathcal{B}_j$. The minimizer (21) of $G^\mu(H(\mathcal{B}_j)|\widetilde{H}(\mathcal{B}_j))$ has the following componentwise expression:

$$(26) \qquad H^\star(\mathcal{B}_j) = \widetilde{H}(\mathcal{B}_j) \odot \left(\frac{[C(\mathcal{B}_j)]}{[D(\mathcal{B}_j) - \mu_j\boldsymbol{\gamma}_j]}\right)^{.\eta(\beta)},$$

where $C = W^T\left((WH)^{\cdot(\beta-2)} \odot V\right)$, $D = W^T(WH)^{\cdot(\beta-1)}$, $\eta(\beta) = \frac{1}{2-\beta}$ for $\beta < 1$, $\eta(\beta) = 1$ for $\beta \in [1,2]$ and $\eta(\beta) = \frac{1}{\beta-1}$ for $\beta > 2$ [12, Table 2], $A \odot B$ (resp., $\frac{[A]}{[B]}$) is the Hadamard product (resp., division) between $A$ and $B$, $A^{\cdot\alpha}$ is the elementwise $\alpha$ exponent of $A$. According to Proposition 2, (26) is a well-defined update from $(0,\infty)^Q$ to itself, provided that $\mu_j$ is tuned to $\mu_j^\star$. This brings us a structural guarantee that $D(\mathcal{B}_j) - \mu_j^\star\gamma_j$ cannot cancel.

Finally, we need to determine $\mu_j^\star$. This amounts to solving $\gamma_j^T H^\star(\mathcal{B}_j) = c_j$, or equivalently, to find the root of the function

$$(27) \qquad r_j(\mu_j) = \sum_{q=1}^{Q} \gamma_{j,q}\left[\tilde{H}(\mathcal{B}_j) \odot \left(\frac{[C(\mathcal{B}_j)]}{[D(\mathcal{B}_j) - \mu_j\gamma_j]}\right)^{\cdot\eta(\beta)}\right]_q - c_j,$$

where $[A]_q$ denotes the $q$th entry of expression $A$. Proposition 2 shows that $\mu_j^\star$ is unique on some interval $(-\infty, t)$. Indeed, $r_j(\mu_j)$ is a finite sum of elementary rational functions of $\mu_j$ and each of them is an increasing, convex function in $\mu_j$ over $(-\infty, t_q)$ with $t_q = \frac{D_q(\mathcal{B}_j)}{\gamma_{j,q}}$ for each $q$. It is even completely monotone for all $\mu$ in $(-\infty, t_q)$ because $\eta(\beta) > 0$ [27]. As a consequence $r_j(\mu_j)$ is also a completely monotone, convex increasing function of $\mu_j$ in $(-\infty, t)$, where $t = \min(t_q)$. Finally, we can easily show that the function $r_j(\mu_j)$ changes of sign on the interval $(-\infty, t)$ by computing two limits at the closure of the interval. As $\mu^\star \in (-\infty, t)$, the update (26) is nonnegative. To evaluate $\mu^\star$, we use a Newton–Raphson method, with any initial point $\mu_0 \in (\mu^\star, t)$, with a quadratic rate of convergence as demonstrated in Proposition 3. Algorithm 1 summarizes our method to tackle (2) for all the $\beta$-divergences which we refer to as the disjoint-constrained $\beta$-NMF algorithm. The update for matrix $W$ can be derived in the same way, by symmetry of the problem.

*Remark* 1. As noted above, the denominators of (26) and (27) will be different from zero. This follows notably from our assumption that $(W, H) > 0$; see Propositions 1, 2, and 3. This is a standard assumption in the NMF literature: the entries of $(W, H)$ are initialized with positive entries which ensures all iterates to remain positive. This is important because the MU cannot change an entry equal to zero [25]; this is the so-called zero-locking phenomenon. This implies $C$ and $D$ in (26) and (27) are positive matrices (as long as $V$ has at least one nonzero entry per row and column). In practice, one should, however, be careful because some entries of $W$ and $H$ can numerically be set to zero (because of finite precision). Hence, in our implementation, we use the machine precision as a lower bound for the entries of $W$ and $H$, as recommended in [17].

*Computational cost.* The computational cost of Algorithm 1 is asymptotically equivalent to the standard MU for $\beta$-NMF, that is, it requires $\mathcal{O}(FNK)$ operations per iteration. Indeed, the complexity is mainly driven by matrix products required to compute $C$ and $D$; see (26). To compute the roots of (27) corresponding to $H$ using Newton–Raphson, each iteration requires one to compute $r_j(\mu_j)/r'_j(\mu_j)$ for all $j$ which requires $\mathcal{O}(KN)$ operations (when every entry of $H$ appears in a constraint). Finding the roots therefore requires $\mathcal{O}(KN)$ operations times the number of Newton–Raphson iterations. By symmetry, it requires $\mathcal{O}(KF)$ operations to compute the roots corresponding to $W$. Because of the quadratic convergence, the number of iterations required for the convergence of the Newton–Raphson method is typically small, namely between 10 to 100 in our experiments using the stopping criterion $|r(\mu_j)| \leq 10^{-6}$ for all $j$. Therefore, in practice, the overall complexity of Algorithm 1

is dominated by the matrix products that require $\mathcal{O}(FNK)$ operations. The same conclusions apply to the algorithms presented in sections 3, 4, and 5, and this will be confirmed by our numerical experiments.

---

**Algorithm 1** $\beta$-NMF with disjoint linear constraints.

---

**Input:** A matrix $V \in \mathbb{R}^{F \times N}$, initial matrices $H \in \mathbb{R}_+^{K \times N}$ and $W \in \mathbb{R}^{F \times K}$, a factorization rank $K$, a maximum number of iterations, maxiter, a value for $\beta$, and the linear constraints defined by $\mathcal{K}_i$, $\boldsymbol{\alpha}_j$ and $b_i$ for $i = 1, 2, \ldots, I$, and $\mathcal{B}_j$, $\boldsymbol{\gamma}_j$, and $c_j$ for $j = 1, 2, \ldots, J$.

**Output:** A rank-$K$ NMF $(W, H)$ of $V$ satisfying constraints in (2).

---

1: **for** $it = 1$ : maxiter **do**
2:      *% Update of matrix H*
3:      $C \leftarrow W^T \left( (WH)^{\cdot(\beta-2)} \odot V \right)$
4:      $D \leftarrow W^T (WH)^{\cdot(\beta-1)}$
5:      **for** $j = 1 : J$ **do**
6:          $\mu_j \leftarrow \text{root}(r_j(\mu_j))$      *% see* (27)
7:          $H(\mathcal{B}_j) \leftarrow H(\mathcal{B}_j) \odot \left( \frac{[C(\mathcal{B}_j)]}{[D(\mathcal{B}_j) - \mu_j \gamma_j]} \right)^{\cdot(\eta(\beta))}$
8:      **end for**
9:      $\mathcal{B}_c = \{(k,n) \mid 1 \leqslant k \leqslant K, 1 \leqslant n \leqslant N\} \setminus \left( \cup_j^J \mathcal{B}_j \right)$.     *% $\mathcal{B}_c$ is the complement of* $\cup_j^J \mathcal{B}_j$
10:     $H(\mathcal{B}_c) \leftarrow H(\mathcal{B}_c) \odot \left( \frac{[C(\mathcal{B}_c)]}{[D(\mathcal{B}_c)]} \right)^{\cdot \eta(\beta)}$
11:     *% Update of matrix W*
12:     $W$ is updated in the same way as $H$, by symmetry of the problem.
13: **end for**

---

**3. Showcase 1: Simplex-structured $\beta$-NMF.** In this section, we showcase a particularly important example of $\beta$-NMF with linear disjoint constraints and no penalization, namely, the simplex-structured matrix factorization (SSMF) problem. It is defined as follows: given a data matrix $V \in \mathbb{R}^{F \times N}$ and a factorization rank $K$, SSMF refers to the problem of computing $W$ and $H$ such that $V \approx WH$ and the columns of $H$ lie on the unit simplex, that is, the entries of each column of $H$ are nonnegative and sum to one. SSMF is a powerful tool in many applications such as hyperspectral unmixing in geoscience and remote sensing [4, 26, 1], document analysis [5], and self-modeling curve resolution [28]. We refer the reader to the recent survey [13] for more applications and details about SSMF.

To understand the underlying significance of SSMF, it is necessary to give more insights on a research topic for which important SSMF techniques were initially developed which is the blind Hyperspectral Unmixing (HU), a main research topic in remote sensing. The task of blind HU is to decompose a remotely sensed hyperspectral image into endmember spectral signatures and the corresponding abundance maps with limited prior information, usually the only known information being the number of endmembers. In this context, the columns of $W$ correspond to the endmembers spectral signatures and the columns of $H$ contain the proportion of the endmembers in each column of $V$, so the column-stochastic assumption for $H$ naturally holds. The nonnegativity of $W$ follows from the nonnegativity of the spectral signatures. We refer to the corresponding problem as simplex-structured NMF with the $\beta$-divergence

($\beta$-SSNMF), and is formulated as follows:

$$(28) \quad \min_{W \in \mathbb{R}_+^{F \times K}, H \in \mathbb{R}_+^{K \times N}} D_\beta(V|WH) \quad \text{such that} \quad \boldsymbol{e}^T \boldsymbol{h}_j = 1 \text{ for } 1 \leqslant j \leqslant N,$$

where $\boldsymbol{e}$ is the vector of all ones of appropriate dimension. This is a particular case of (2) where
- the subsets $\mathcal{B}_j$ correspond to the columns of $H$, and there is no subset $\mathcal{K}_i$ (no constraint on $W$),
- $\boldsymbol{\gamma}_j^T = \boldsymbol{e}$ and $c_j = 1$ for $j = 1, 2, \dots, N$.

Hence Algorithm 1 can be directly applied to (28).

*Numerical experiments.* Let us perform numerical experiments to evaluate the effectiveness of Algorithm 1 on the simplex-structure $\beta$-NMF problem against existing methods. To the best of our knowledge, the so-called group robust NMF (GR-NMF) algorithm[1] from [14] is the most recent algorithm that is able to tackle problem (28) for the full range of $\beta$-divergences. The approach is not based on Lagrangian multipliers but introduces a change of variables for matrix $H$. This approach, initially used for NMF in [10], does not provide an auxiliary function for the subproblem in $H$ and resort to a heuristic commonly used in NMF; see, for example, [32, 11]. Therefore there is no guarantee that the objective function is decreasing at each update of the abundance matrix, unlike Algorithm 1.

We apply Algorithm 1 and GR-NMF on three widely used real hyperspectral data sets[2] [35]:
- Samson: 156 spectral bands with $95 \times 95$ pixels, containing mostly three materials ($K = 3$), namely "Soil," "Tree," and "Water."
- Jasper Ridge: 198 spectral bands with $100 \times 100$ pixels, containing mostly four materials ($K = 4$), namely "Road," "Soil," "Water," and "Tree."
- Cuprite: 188 spectral bands with $250 \times 190$ pixels, containing mostly 12 types of minerals ($K = 12$).

$\beta$-SSNMF has shown itself as a powerful one to tackle blind HU, hence this comparative study between Algorithm 1 and GR-NMF [14] focuses on the convergence aspects including the evolution of the objective function and the runtime. The algorithms are compared for $\beta \in \left\{0, \frac{1}{2}, 1, \frac{3}{2}, 2\right\}$. To report the results, we use the relative objective function, denoted $\bar{F}(W, H)$ and defined as[3]

$$\bar{F}(W, H) = \frac{D_\beta(V|WH)}{D_\beta(V|v\boldsymbol{e}\boldsymbol{e}^T)},$$

where $v = \frac{\boldsymbol{e}^T V \boldsymbol{e}}{FN}$ is the average of the entries of $V$. The relative error $\bar{F}$ should be between 0 and 1: it is equal to 0 for an exact decomposition with $V = WH$, and is equal to 1 for a trivial rank-one approximation where all entries are equal to the average of the entries of $V$. This allows one to meaningfully interpret the results, especially since we consider in this comparative study multiple values for $\beta$. In fact, the degree of homogeneity of the $\beta$-divergence is a function of $\beta$. For example, if all the entries of the input matrix are multiplied by 10 and keeping the same NMF

---

[1]https://www.irit.fr/~Cedric.Fevotte/extras/tip2015/code.zip

[2]http://lesun.weebly.com/hyperspectral-data-set.html

[3]For the Frobenius norm, that is, $\beta = 2$, the relative error is typically defined as $\frac{D_\beta(V|WH)}{D_\beta(V|0)}$ meaning that the trivial solution used is the all-zero matrix. However, for other $\beta$-divergences, the value of $D_\beta(V|0)$ might not be defined; in particular, for $\beta \leqslant 1$ and $v_{fn} > 0$ for some $f, n$.

solution properly scaled, the squared Frobenius error ($\beta = 2$) is multiplied by 100 while the IS-divergence ($\beta = 0$) is not affected.

As for all tests performed in this paper, the algorithms are tested on a desktop computer with Intel Core i7-8700 @ 3.2GHz CPU and 32GB memory. The codes are written in MATLAB R2018a, and available from https://sites.google.com/site/nicolasgillis/. For all simulations, the algorithms are run for 20 random initializations of $W$ and $H$ (each entry sampled from the uniform distribution in $[0, 1]$). Table 2 reports the average and standard deviation of the runtime (in seconds) as the final value for the relative objective function over these 20 runs for a maximum of 300 iterations.

TABLE 2
*Runtime performance in seconds and final value of relative objective function $\bar{F}_{end}(W, H)$ for Algorithm 1 and the GR-NMF reported for $\beta \in \{0, \frac{1}{2}, 1, \frac{3}{2}, 2\}$. The table reports the average and standard deviation over 20 random initializations with a maximum of 300 iterations for three hyperspectral data sets. A bold entry indicates the best value for each experiment.*

| Algorithms | Samson | | Jasper Ridge | | Cuprite | |
|---|---|---|---|---|---|---|
| | runtime (s.) | $\bar{F}_{end}(W, H)$ | runtime (s.) | $\bar{F}_{end}(W, H)$ | runtime (s.) | $\bar{F}_{end}(W, H)$ |
| | | | $\beta = 2$ | | | |
| Algorithm 1 | **16.62$\pm$0.15** | $(\mathbf{1.89\pm0.04})10^{-3}$ | **22.86$\pm$0.08** | $(\mathbf{4.68 \pm 0.39})10^{-3}$ | 121.04 $\pm$ 0.62 | $(\mathbf{0.98 \pm 0.06})10^{-3}$ |
| GR-NMF | 18.23$\pm$0.29 | $(1.91\pm0.05)10^{-3}$ | 25.32$\pm$0.16 | $(5.87 \pm 1.22)10^{-3}$ | **114.27 $\pm$ 0.20** | $(1.29 \pm 0.07)10^{-3}$ |
| | | | $\beta = 3/2$ | | | |
| Algorithm 1 | **63.69$\pm$0.40** | $(\mathbf{2.52 \pm 0.78})10^{-3}$ | **89.23 $\pm$0.30** | $(\mathbf{4.92 \pm 0.29})10^{-3}$ | 421.49 $\pm$ 2.79 | $(\mathbf{1.54 \pm 0.07})10^{-3}$ |
| GR-NMF | 80.09$\pm$0.60 | $(2.60 \pm0.63)10^{-3}$ | 112.72 $\pm$0.67 | $(6.32 \pm 1.37)10^{-3}$ | 508.57 $\pm$ 3.50 | $(2.01 \pm 0.09)10^{-3}$ |
| | | | $\beta = 1$ | | | |
| Algorithm 1 | **18.33 $\pm$ 0.08** | $(\mathbf{3.54 \pm 0.27})10^{-3}$ | **24.82 $\pm$ 0.35** | $(\mathbf{6.07 \pm 0.21})10^{-3}$ | **182.98 $\pm$ 14.14** | $(\mathbf{2.07 \pm 0.09})10^{-3}$ |
| GR-NMF | 44.78 $\pm$ 0.18 | $(3.77 \pm 0.38\ )10^{-3}$ | 62.83 $\pm$ 0.76 | $(7.26 \pm 1.50)10^{-3}$ | 370.25 $\pm$ 21.33 | $(2.67 \pm 0.10)10^{-3}$ |
| | | | $\beta = 1/2$ | | | |
| Algorithm 1 | **89.80 $\pm$ 0.65** | $(7.21 \pm 0.75)10^{-3}$ | **126.43 $\pm$ 0.61** | $(\mathbf{1.08 \pm 0.10})10^{-2}$ | 682.80 $\pm$ 3.32 | $(\mathbf{3.13 \pm 0.15})10^{-3}$ |
| GR-NMF | 102.21 $\pm$ 0.72 | $(\mathbf{6.93 \pm 0.88})10^{-3}$ | 141.75 $\pm$ 0.69 | $(1.12 \pm 0.13)10^{-2}$ | **642.49 $\pm$ 1.22** | $(3.14 \pm 0.14)10^{-3}$ |
| | | | $\beta = 0$ | | | |
| Algorithm 1 | **52.89$\pm$0.54** | $(4.60 \pm 0.66)10^{-2}$ | **69.59$\pm$0.44** | $(\mathbf{3.76 \pm 0.11})10^{-2}$ | 479.84 $\pm$ 16.02 | $(4.39 \pm 0.31)10^{-3}$ |
| GR-NMF | 55.61$\pm$0.47 | $(\mathbf{4.22 \pm 0.79})10^{-2}$ | 77.87$\pm$0.63 | $(3.76 \pm 0.44)10^{-2}$ | **354.65 $\pm$6.01** | $(\mathbf{3.35 \pm 0.10})10^{-3}$ |

We observe that Algorithm 1 outperforms the GR-NMF in terms of runtime and final values for the relative objective function for all test cases except when $\beta = 0$ for the Samson and Cuprite data sets. In particular, for $\beta = 1$, Algorithm 1 is up to 2.5 times faster than the GR-NMF. For the Cuprite data set with $\beta = 1/2$, Algorithm 1 and GR-NMF perform similarly. We also observe that the standard deviations obtained with Algorithm 1 are, in general, significantly smaller for all $\beta$, except for $\beta = 0$ for the Samson and Cuprite data sets.

In section SM1 of the supplementary material file Supplementary.pdf [local/web 2.75MB], we provide figures that show the evolution of the relative objective function values with respect to iterations, and that confirm the observations above.

**4. Showcase 2: Minimum-volume KL-NMF.** In this section, we showcase another important example of $\beta$-NMF with linear disjoint constraints, namely, the minimum volume NMF with the $\beta$-divergences (min-vol $\beta$-NMF) model. This model is based on the minimization of $\beta$-divergences including a penalty term promoting solutions with minimum volume spanned by the columns of the matrix $W$. It is defined as follows:

$$
(29) \qquad \min_{W \in \mathbb{R}_+^{F \times K}, H \in \mathbb{R}_+^{K \times N}} D_\beta(V|WH) + \lambda \text{vol}(W) \quad \text{such that} \quad W^T e = e.
$$

where $\lambda$ is a penalty parameter, and $\text{vol}(W)$ is a function measuring the volume spanned by the columns of $W$. In [24], the authors use $\text{vol}(W) = \text{logdet}(W^T W + \delta I)$,

where $\delta$ is a small positive constant that prevents $\log\det(W^TW)$ from going to $-\infty$ when $W$ tends to a rank-deficient matrix (that is, when $\operatorname{rank}(W) < K$). This model is particularly powerful as it leads to identifiability which is crucial in many applications such as in hyperspectral imaging or audio source separation [13]. Indeed, under some mild assumptions and in the exact case, authors of [24] prove that (29) is able to identify the groundtruth factors $(W^\#, H^\#)$ that generated the input data $V$, in the absence of noise. In [24], (29) is used for blind audio source separation. In a nutshell, blind audio source separation consists of isolating and extracting unknown sources based on an observation of their mix recorded with a single microphone.[4] We have to mention that model (29) is also well suited for hyperspectral imaging as discussed in [16].

In the next subsections, we show that we can tackle the min-vol $\beta$-NMF optimization problem defined in (29) with the general framework presented in section 2 in the case $\beta = 1$.

**4.1. Problem formulation and algorithm.** As the minimum-volume penalty of model (29) concerns matrix $W$ only, the main challenge concerns the update of $W$. Indeed, the update of $H$ is simply the one from [22]. Let us therefore consider the subproblem in $W$ for $H$ fixed:

(30)
$$\min_{W \in \mathbb{R}_+^{F \times K}} D_\beta\left(V | WH\right) + \lambda \log\det(W^TW + \delta I) \quad \text{such that} \quad \boldsymbol{e}^T \boldsymbol{w}_i = 1 \text{ for } 1 \leqslant i \leqslant K.$$

Compared to the general model (2), we have the following:
- the subsets $\mathcal{K}_i$ correspond to the columns of $W$, and there is no subset $\mathcal{B}_j$,
- $\boldsymbol{\alpha}_i^T = \boldsymbol{e}$ and $b_i = 1$ for $1 \leqslant i \leqslant K$.

To upper bound $\log\det(W^TW + \delta I)$ as required by (5) in Assumption 1, we majorize it using a convex quadratic separable auxiliary function provided in [24, eq. (3.6)], which is derived as follows. First, the concave function $\log\det(Q)$ for $Q > 0$ can be upper bounded using the first-order Taylor approximation: for any $\widetilde{Q} > 0$,

$$\log\det(Q) \leqslant \log\det(\widetilde{Q}) + \langle \widetilde{Q}^{-1}, Q - \widetilde{Q} \rangle = \langle \widetilde{Q}^{-1}, Q \rangle + \text{cst},$$

where cst is some constant independent of $Q$. For any $W, \widetilde{W}$, and denoting $\widetilde{Q} = \widetilde{W}^T \widetilde{W} + \delta I > 0$, we obtain

$$\log\det(W^TW + \delta I) \leqslant \left\langle \widetilde{Q}^{-1}, W^TW \right\rangle + \text{cst} = \operatorname{trace}(W\widetilde{Q}^{-1}W^T) + \text{cst},$$

which is a convex quadratic and Lipschitz-smooth function in $W$. In fact, letting $\widetilde{Q}^{-1} = DD^T$ be a decomposition (such as Cholesky) of $\widetilde{Q}^{-1} > 0$, we have $\operatorname{trace}(W\widetilde{Q}^{-1}W) = \|WD\|_F^2$, from which (5) can be derived easily; see [24] for the details. With this and following our framework from section 2, we obtain the Lagrangian function

(31) $$G^\mu\left(W | \widetilde{W}\right) = \sum_f G\left(\boldsymbol{w}_f | \widetilde{\boldsymbol{w}}_f\right) + \lambda \left( \sum_f \bar{l}\left(\boldsymbol{w}_f | \widetilde{\boldsymbol{w}}_f\right) + c \right) + \boldsymbol{\mu}^T \sum_f \left( \boldsymbol{w}_f - \frac{1}{F}\boldsymbol{e} \right),$$

where $\boldsymbol{w}_f \in$ denotes the $f$th row of $W$, $G$ is given by (24), $\bar{l}$ by [24, eq. (3.6)] and derived as explained above, and $c$ is a constant. Let $\boldsymbol{\mu}$ be the vector Lagrange multipliers of dimension $K$ associated to each linear constraint $\boldsymbol{e}^T \boldsymbol{w}_i = 1$. Exactly as before

---

[4]The interested reader may watch the video https://www.youtube.com/watch?v=1BrpxvpghKQ to see the application of min-vol KL-NMF on the decomposition of a famous song from the city of Mons.

(hence we omit the details here), $G^\mu$ is separable and, given $\boldsymbol{\mu}$, one can compute the closed-form solution

$$(32) \qquad W^\star(\mu) = \widetilde{W} \odot \frac{\left[\left[C + \boldsymbol{e}\boldsymbol{\mu}^T\right]^{\cdot 2} + S\right]^{\cdot\frac{1}{2}} - \left(C + \boldsymbol{e}\boldsymbol{\mu}^T\right)}{[D]},$$

where $C = e_{F,N}H^T - 4\lambda\left(\widetilde{W}Y^-\right)$, $D = 4\lambda\widetilde{W}\left(Y^+ + Y^-\right)$, and $S = 8\lambda\widetilde{W}\left(Y^+ + Y^-\right) \odot \left(\frac{[V]}{[\widetilde{W}H]}H^T\right)$ with $Y = Y^+ - Y^- = \left(\widetilde{W}^T\widetilde{W} + \delta I\right)^{-1}$, $Y^+ = \max(Y,0) \geqslant 0$, and $Y^- = \max(-Y,0) \geqslant 0$, and $e_{F,N}$ is the $F$-by-$N$ matrix of all ones. As proved in Proposition 2, the constraint $W^\star(\mu)^T\boldsymbol{e} = \boldsymbol{e}$ is satisfied for a unique $\boldsymbol{\mu}$ in $(-\infty, t)$ where $t = \infty$ in this case. We can therefore use a Newton–Raphson method to find the $\mu_i$ with quadratic rate of convergence; see Proposition 3. Algorithm 2 summarizes our method to tackle (29).

---

**Algorithm 2** Min-vol KL-NMF.

---

**Input:** A matrix $V \in \mathbb{R}^{F \times N}$, initial matrices $H \in \mathbb{R}_+^{K \times N}$, and $W \in \mathbb{R}^{F \times K}$, a factorization rank $K$, a maximum number of iterations, maxiter, and the parameters $\delta > 0$ and $\lambda > 0$.

**Output:** A min-vol rank-$K$ NMF $(W, H)$ of $V$ satisfying constraints in (29).

1: **for** $it = 1 :$ maxiter **do**
2:      *% Update of matrix H*
3:      $H \leftarrow H \odot \frac{\left[W^T\left(\frac{[V]}{[WH]}\right)\right]}{[W^T e_{F,N}]}$
4:      *% Update of matrix W*
5:      $Y \leftarrow \left(W^TW + \delta I\right)^{-1}$
6:      $Y^+ \leftarrow \max(Y, 0)$
7:      $Y^- \leftarrow \max(-Y, 0)$
8:      $C \leftarrow e_{F,N}H^T - 4\lambda\left(WY^-\right)$
9:      $S \leftarrow 8\lambda W\left(Y^+ + Y^-\right) \odot \left(\frac{[V]}{[WH]}H^T\right)$
10:     $D \leftarrow 4\lambda W\left(Y^+ + Y^-\right)$
11:     $\boldsymbol{\mu} \leftarrow \text{root}\left(W^\star(\mu)^T\boldsymbol{e} = \boldsymbol{e}\right)$ over $\mathbb{R}^K$      *% see (32) for the expression of $W^\star(\mu)$*
12:     $W \leftarrow W \odot \frac{\left[\left[C + e\boldsymbol{\mu}^T\right]^{\cdot 2} + S\right]^{\cdot\frac{1}{2}} - \left(C + e\boldsymbol{\mu}^T\right)}{[D]}$
13: **end for**

---

**4.2. Numerical experiments.** In this section we compare baseline KL-NMF (that is, the standard MU), the min-vol KL-NMF from [24, Algorithm 1] that solves (29) using MU combined with line search (min-vol KL-NMF LS), and Algorithm 2 applied to the spectrogram of two monophonic piano sequences considered in [24]. The first audio sample is the first measure of "Mary had a little lamb," a popular English song. The second audio sample corresponds to the first 30 seconds of "Prelude and Fugue No. 1 in C major" from de Jean-Sebastien Bach played by Glenn Gould.[5] We use the following three setups:
- Setup ♯1: sample "Mary had a little lamb" with $K = 3$; 200 iterations.

---

[5]https://www.youtube.com/watch?v=ZlbK5r5mBH4

- Setup ♯2: sample "Mary had a little lamb" with $K = 7$; 200 iterations.
- Setup ♯3: "Prelude and Fugue No. 1 in C major" with $K = 16$; 300 iterations.

For each setup, the algorithms are run for the same 20 random initializations of $W$ and $H$. Table 3 reports the average and standard deviation of the runtime (in seconds) over these 20 runs. Table 4 reports the average and standard deviation of the final values for $\beta$-divergences (data fitting term) and the objective function of (29) over these 20 runs for min-vol KL-NMF LS and Algorithm 2. For this last comparison, the value for the penalty weight $\lambda$ has been chosen so that KL-NMF leads to reasonable solutions for $W$ and $H$. More precisely, the values for $\lambda$ are chosen so that the initial value of $\frac{\lambda \left| \text{logdet}(W^{(0)^T} W^{(0)} + \delta I) \right|}{D_\beta(V|WH)}$ is equal to 0.1, 0.1, and 0.022 for setup ♯1, setup ♯2, and setup ♯3, respectively.

TABLE 3
*Runtime performance in seconds of baseline KL-NMF, min-vol KL-NMF LS, and Algorithm 2. The table reports the average and standard deviation over 20 random initializations.*

| Algorithms | Runtime in seconds | | |
|---|---|---|---|
| | setup ♯1 | setup ♯2 | setup ♯3 |
| Baseline KL-NMF | $0.53 \pm 0.03$ | $0.45 \pm 0.02$ | $4.32 \pm 0.30$ |
| min-vol KL-NMF LS [24] | $3.79 \pm 0.13$ | $2.39 \pm 0.30$ | $10.19 \pm 1.28$ |
| Algorithm 2 | $0.58 \pm 0.03$ | $0.66 \pm 0.03$ | $4.80 \pm 0.38$ |

TABLE 4
*Final values for $D_\beta$ and the penalized objective $\Psi$ from (29) obtained with min-vol KL-NMF LS and Algorithm 2. The table reports the average and standard deviation over 20 random initializations for three experimental setups. A bold entry indicates the best value for each experiment.*

| | | min-vol KL-NMF LS [24] | Algorithm 2 |
|---|---|---|---|
| setup ♯1 | $D_{\beta,\text{end}}$ | $(3.52 \pm 0.03)10^3$ | $\mathbf{(2.31 \pm 0.01)}10^3$ |
| | $\Psi_{\text{end}}$ | $(4.17 \pm 0.03)10^3$ | $\mathbf{(3.08 \pm 0.01)}10^3$ |
| setup ♯2 | $D_{\beta,\text{end}}$ | $(3.54 \pm 0.03)10^3$ | $\mathbf{(1.77 \pm 0.02)}10^3$ |
| | $\Psi_{\text{end}}$ | $(4.42 \pm 0.04)10^3$ | $\mathbf{(2.87 \pm 0.02)}10^3$ |
| setup ♯3 | $D_{\beta,\text{end}}$ | $(7.77 \pm 0.23)10^3$ | $\mathbf{(4.67 \pm 0.08)}10^3$ |
| | $\Psi_{\text{end}}$ | $(9.14 \pm 0.20)10^3$ | $\mathbf{(6.50 \pm 0.06)}10^3$ |

We observe that the runtime of Algorithm 2 is close to the baseline KL-NMF algorithm which confirms the negligible cost of the Newton–Raphson steps to compute $\boldsymbol{\mu}^\star$ as discussed in section 2.3. On the other hand, since no line search is needed, we have a drastic acceleration from 2x to 7x compared to the backtracking line-search procedure integrated in min-vol KL-NMF LS [24]. Moreover, we observe in Table 4 that Algorithm 2 outperforms min-vol KL-NMF LS in terms of final values for the data fitting term and objective function values, with lower standard deviations.

**5. Extension to quadratic disjoints constraints.** Our general framework presented in section 2 applies to $\beta$-NMF under disjoint linear equality constraints with penalty terms satisfying Assumption 1; see problem (2). We have showcased our approach on $\beta$-SSNMF in section 3 and on min-vol KL-NMF under sum-to-one constraints on the columns of $W$ in section 4. In this section, we show that the same framework can be extended to other simple constraints, namely disjoint quadratic constraints.

We consider sparse $\beta$-NMF for $\beta = 1$, where the rows of $H$ are penalized with the $\ell_1$ norm and each column of $W$ have a fixed $\ell_2$ norm. We show that MU satisfying the set of constraints can be derived which we apply on blind HU.

**5.1. Problem formulation and algorithm.** In this section we consider the following model involving quadratic disjoints constraints, which we refer to as hyperspheric-structured sparse $\beta$-NMF:

(33)
$$\min_{W \in \mathbb{R}_+^{F \times K}, H \in \mathbb{R}_+^{K \times N}} D_\beta(V|WH) + \sum_{k=1}^{K} \lambda_k \|H(k,:)\|_1 \text{ such that } \boldsymbol{e}^T \boldsymbol{w}_j^{\cdot(2)} = \rho \text{ for } 1 \leqslant i \leqslant K,$$

where $\lambda_k$ is a penalty weight to control the sparsity of the $k$th row of $H$, and the quadratic constraints require the columns of $W$ to lie on the surface of a hypersphere centered at the origin with radius $\sqrt{\rho} > 0$. Without this normalization, the $\ell_1$-norm regularization would make $H$ tend to zero and $W$ grow to infinity.

As done before, we update $W$ and $H$ alternatively. We tackle the subproblem in $H$ with $W$ fixed based on the MU developed in [11] and guaranteed to decrease the objective function:

(34)
$$H = \widetilde{H} \odot \frac{\left[W^T \left(V \odot [W\widetilde{H}]^{\cdot(\beta-2)}\right)\right]}{\left[W^T[W\widetilde{H}]^{\cdot(\beta-1)} + \boldsymbol{\lambda} e^T\right]},$$

where $\boldsymbol{\lambda} \in \mathbb{R}_+^K$ is the vector of penalty weights. It remains to compute an update for $W$. To do so, we use the convex separable auxiliary function $G$ from [12] constructed at the current iterate $\widetilde{W}$, from which we obtain, as before, the Lagrangian function

(35)
$$G^\mu\left(W|\widetilde{W}\right) = \sum_f G\left(w_f|\widetilde{w}_f\right) + \sum_k \lambda_k \|H(k,:)\|_1 + \boldsymbol{\mu}^T \sum_f \left(w_f^{\cdot(2)} - \frac{1}{F}\rho\boldsymbol{e}\right),$$

where $\boldsymbol{\mu} \in \mathbb{R}^K$ is the vector of Lagrange multipliers associated to the constraint $\boldsymbol{e}^T \boldsymbol{W}^{\cdot(2)} = \rho \boldsymbol{e}^T$. Exactly as before (hence we omit the details here), given $\boldsymbol{\mu}$, one can obtain a closed-form solution:

(36)
$$W^\star(\boldsymbol{\mu}) = \frac{\left[\left[[C]^{\cdot 2} + 8\left(\boldsymbol{e}\boldsymbol{\mu}^T\right) \odot S\right]^{\cdot\frac{1}{2}} - C\right]}{[4\boldsymbol{e}\boldsymbol{\mu}^T]},$$

where $C = e_{F,N}H^T$ and $S = \widetilde{W} \odot \left(\frac{[V]}{[\widetilde{W}H]}H^T\right)$. Let us now write the expression of the quadratic constraint $\sum_f (W^\star(\boldsymbol{\mu})_{f,i})^2 - \rho = 0$ for one specific column of $W$, say the $i$th:

(37)
$$r_i(\mu_i) := \sum_f (W_{f,i}^\star(\mu_i))^2 - \rho = \sum_f \left(\frac{\sqrt{(C_{f,i})^2 + 8\mu_i S_{f,i}} - C_{f,i}}{4\mu_i}\right)^2 - \rho = 0.$$

Computing the Lagrangian multiplier $\mu_i$ to satisfy the constraint requires computing the roots of the functions $r_i(\mu_i)$. We can show that each $W_{f,i}^\star(\mu_i)$ (36) is a monotone decreasing, nonnegative convex function over $(0, +\infty)$. Therefore, $\sum_f (W_{f,i}^\star(\mu_i))^2$ is also monotone decreasing and convex in $\mu_i$ over $(0, +\infty)$. Indeed, let $g : \mathbb{R}_+ \to \mathbb{R}_+$ be a monotone decreasing, nonnegative convex function. If $g$ is twice-differentiable, then $\left(g^2\right)'' = 2\left(g'\right)^2 + 2gg'' \geqslant 0$ since $g, g'' \geqslant 0$ and $\left(g^2\right)' = 2g'g \leqslant 0$ since $g \geqslant 0$, $g' \leqslant 0$ by hypothesis. Now we can conclude that $r_i(\mu_i)$ is a monotone decreasing convex

function over $(0, +\infty)$. Moreover, using Hospital's rule, we have

$$\lim_{\mu_i \to 0^+} \sum_f (W_{f,i}^\star(\mu_i))^2 - \rho = +\infty \quad \text{and} \quad \lim_{\mu_i \to +\infty} \sum_f (W_{f,i}^\star(\mu_i))^2 - \rho = -\rho < 0,$$

since $\rho > 0$. Therefore, the root of $r_i(\mu_i)$ is unique over $(0, +\infty)$. We use a Newton–Raphson method to solve the problem. Algorithm 3 summarizes our method.

---

**Algorithm 3** Hyperspheric-structured sparse KL-NMF.

---

**Input:** A matrix $V \in \mathbb{R}^{F \times N}$, an initialization $H \in \mathbb{R}_+^{K \times N}$, an initialization $W \in \mathbb{R}^{F \times K}$, a factorization rank $K$, a maximum number of iterations, maxiter, and a weight vector $\lambda > 0$.

**Output:** A sparse rank-$K$ NMF $(W, H)$ of $V$ satisfying constraints in (33).

1: **for** $it = 1 : $ maxiter **do**
2:     % *Update of matrix $H$*
3:     $H \leftarrow H \odot \dfrac{\left[W^T\left(V \odot [WH]^{\cdot(\beta-2)}\right)\right]}{\left[W^T[WH]^{\cdot(\beta-1)} + \lambda e^T\right]}$
4:     % *Update of matrix $W$*
5:     $C \leftarrow e_{F,N} H^T$
6:     $S \leftarrow W \odot \left(\dfrac{[V]}{[WH]} H^T\right)$
7:     **for** $j = 1 : $ K **do**
8:         $\mu_i \leftarrow \text{root}(r_i(\mu_i))$ over $(0, +\infty)$ % *See* (37)
9:     **end for**
10:    $W \leftarrow \dfrac{\left[[[C]^{\cdot 2} + 8(e\mu^T) \odot S]^{\cdot \frac{1}{2}} - C\right]}{[4\mu e^T]}$
11: **end for**

---

**5.2. Numerical experiments.** In this section, we perform numerical experiments to evaluate the effectiveness of Algorithm 3 on the HU problem. To the best of our knowledge, sparse $\beta$-NMF[6] from [30] is the most recent algorithm that is able to tackle problem (33) for the KL-divergence by integrating the $\ell_2$-normalization for each update of matrix $W$. This approach is similar to that of [14] for $\beta$-SSNMF, that is, it uses parametrization, and resorts to a heuristic with no guarantee on the decrease of the objective function. We refer to this algorithm as $\beta$-SNMF.

We apply Algorithm 3 and $\beta$-SNMF [30] to the three real hyperspectral datasets detailed in section 3. This comparative study focuses on the convergence aspects including the evolution of the objective function and the runtime; we refer the interested reader to section SM2 of the supplementary material file Supplementary.pdf [local/web 2.75MB] for a qualitative result on the ability of sparse $\beta$-NMF to decompose such images. For all simulations, the algorithms are ran for 20 random initializations of $W$ and $H$, the entries of the penalty weight $\lambda$ has been set to 0.1, 0.05, and 0.05 for Samson, Jasper Ridge and Cuprite data sets, respectively. In order to fairly compare both algorithms, $\rho$ has been set to 1 as $\beta$-SNMF considers a $\ell_2$-normalization for the columns of $W$, and the entries of the weight vector $\lambda$ in Algorithm 3 have the same values as $\beta$-SNMF requires using the same values for all rows of $H$. Table 5 reports the average and standard deviation of the runtime (in seconds) as the final value for

---

[6]http://www.jonathanleroux.org/software/sparseNMF.zip

the objective function over these 20 runs for a maximum of 300 iterations. Figure 1 displays the objective function values.

TABLE 5
*Runtime performance in seconds and final value of objective function $\Phi_{end}(W, H)$ for Algorithm 3 and $\beta$-SNMF. The table reports the average and standard deviation over 20 random initializations with a maximum of 300 iterations for three hyperspectral data sets. A bold entry indicates the best value for each experiment.*

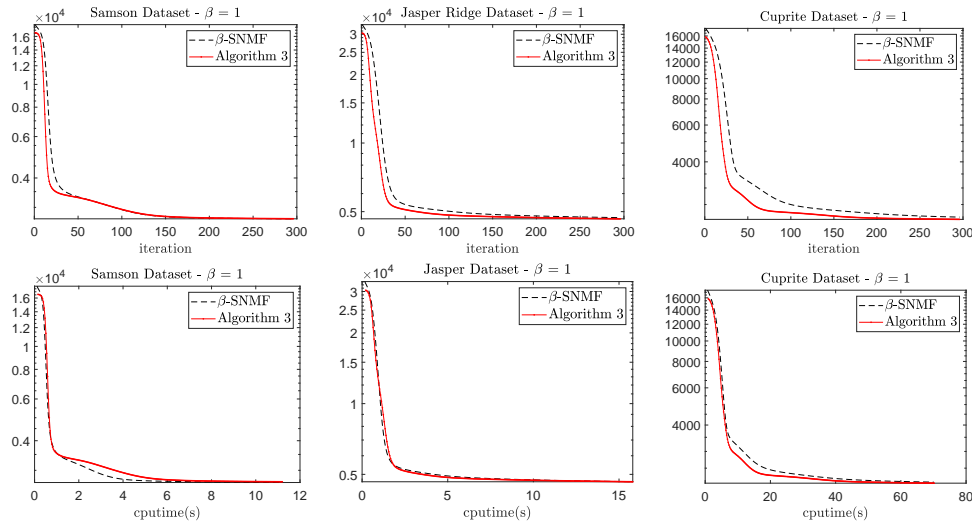| Algorithms | Samson data set | | Jasper Ridge data set | | Cuprite data set | |
|---|---|---|---|---|---|---|
| | runtime (sec) | $\Phi_{end}(W, H)$ | runtime (sec) | $\Phi_{end}(W, H)$ | runtime (sec) | $\Phi_{end}(W, H)$ |
| Algorithm 3 | $11.07{\pm}0.19$ | $(2.68{\pm}0.00)10^3$ | $15.67{\pm}0.17$ | $\mathbf{(4.65 \pm 0.00)}10^3$ | $70.16 \pm 0.85$ | $\mathbf{(2.12 \pm 0.00)}10^3$ |
| $\beta$-SNMF [30] | $\mathbf{7.63{\pm}0.13}$ | $(2.68{\pm}0.00)10^3$ | $\mathbf{10.98{\pm}0.18}$ | $(4.71 \pm 0.00)10^3$ | $\mathbf{51.86 \pm 0.74}$ | $(2.18 \pm 0.00)10^3$ |



FIG. 1. *Averaged objective functions over 20 random initializations obtained for Algorithm 3 with 300 iterations (red line with circle markers), and the heuristic $\beta$-SNMF from [30] (black dashed line). (Figure in color online.)*

According to Table 5 (top row), we observe that Algorithm 3 outperforms the heuristic from [30] in terms of final value for the objective functions while $\beta$-SNMF shows lower runtimes. Additionally, based on Figure 1, we observe that Algorithm 3 converges on average faster than $\beta$-SNMF for all the data sets, in terms of iterations. However, $\beta$-SNMF has a lower computational cost per iteration. Thus, we complete the comparison between both algorithms by imposing the same computational time: we run Algorithm 3 for 300 iterations, record the computational time, and run $\beta$-SNMF for the same amount of time.

Table 6 reports the average and standard deviation of the final value for the objective function over 20 runs in this setting. Figure 1 (bottom row) displays the objective function w.r.t. time for the three data sets. On this comparison, Algorithm 3 and the heuristic from [30] perform similarly although Algorithm 3 has slightly better final objective function values. However, keep in mind that only Algorithm 3 is theoretically guaranteed to decrease the objective function.

**6. Conclusion.** In this paper we have presented a general framework to solve penalized $\beta$-NMF problems that integrates a set of disjoint constraints on the variables; see the general formulation (2). Using this framework, we showed that we can

TABLE 6

*Final value of objective function values $\Phi_{end}(W, H)$ for Algorithm 3 and the heuristic from [30]. The table reports the average and standard deviation over 20 random initializations for an equal computational time that corresponds to 300 iterations of Algorithm 3. A bold entry indicates the best value for each experiment.*

| Algorithms | Samson data set $\Phi_{\text{end}}(W, H)$ | Jasper Ridge data set $\Phi_{\text{end}}(W, H)$ | Cuprite data set $\Phi_{\text{end}}(W, H)$ |
|---|---|---|---|
| Algorithm 3 | $(2.68\pm0.00)10^3$ | $(\mathbf{4.65 \pm 0.00})10^3$ | $(\mathbf{2.12 \pm 0.00})10^3$ |
| $\beta$-SNMF [30] | $(2.68\pm0.00)10^3$ | $(4.66 \pm 0.00)10^3$ | $(2.15 \pm 0.00)10^3$ |

derive algorithms that compete favorably with the state of the art for a wide variety of $\beta$-NMF problems, such as the simplex-structured NMF and the minimum-volume $\beta$-NMF with sum-to-one constraints on the columns of $W$. We have also shown how to extend the framework to nonlinear disjoint constraints, with application to a sparse $\beta$-NMF model for $\beta = 1$, where each column of $W$ lie on a hypersphere.

Further works will focus on the possible extension of the methods to nondisjoint constraints. The nondisjoint constraints will lead to roots finding problems of polynomial equations in the Lagrangian multipliers for which we hope to find conditions that ensure the uniqueness of the solution.

Another interesting direction of research would be to apply our framework to other NMF models. For example, in probabilistic latent semantic analysis/indexing (PLSA/PLSI), the model is the following: given a nonnegative matrix $V$ such that $\boldsymbol{e}^T V \boldsymbol{e} = 1$ (this can be assumed w.l.o.g. by dividing the input matrix by $\boldsymbol{e}^T V \boldsymbol{e}$), solve

$$\max_{W \geqslant 0, H \geqslant 0, s \geqslant 0} \sum_{f,n} v_{fn} \log(W \operatorname{Diag}(s)H)_{fn} \quad \text{such that} \quad W^T \boldsymbol{e} = \boldsymbol{e}, H\boldsymbol{e} = \boldsymbol{e}, s^T \boldsymbol{e} = 1.$$

This model is equivalent to KL-NMF [9], with the additional constraint that $\boldsymbol{e}^T W H \boldsymbol{e} = \boldsymbol{e}^T X \boldsymbol{e}$, and hence our framework is applicable to PLSA/PLSI. Such constraints also have applications in soft clustering contexts; see [33].

**Appendix A. Convexity, concavity, and complete monotonicity for a convex-concave decomposition of the discrete $\boldsymbol{\beta}$-divergence.**
The discrete $\beta$-divergence can always be expressed as the sum of convex, concave, and constant terms. In Table 7 we introduce a convex-concave decomposition of the $\beta$-divergence which slightly differ from the one given in [12, Table 1] (by the fact that ours contains no constant term $\bar{d}$) as given in Table 7.

TABLE 7

*Proposed concave-convex decomposition of the discrete $\beta$-divergence.*

| Decomposition $d_\beta = \check{d} + \hat{d}$ | $\beta \in (-\infty, 1)\backslash\{0\}$ | $\beta = 0$ | $\beta = 1$ | $\beta \in (1, 2)$ | $\beta \in [2, +\infty)$ |
|---|---|---|---|---|---|
| $\check{d}(x|y)$ | $\frac{1}{1-\beta}x\,y^{\beta-1}$ | $\frac{x}{y}$ | $-x \log y$ | $\frac{1}{\beta}y^\beta - \frac{1}{\beta-1}xy^{\beta-1}$ | $\frac{1}{\beta}y^\beta$ |
| $\hat{d}(x|y)$ | $\frac{1}{\beta}y^\beta - \frac{1}{\beta(1-\beta)}x^\beta$ | $\log\frac{y}{x} - 1$ | $y + x \log x - x$ | $\frac{1}{\beta(\beta-1)}x^\beta$ | $-\frac{1}{\beta-1}x\,y^{\beta-1} + \frac{1}{\beta(\beta-1)}x^\beta$ |

In Table 7, $y \in (0, \infty)$, $\beta$ is real valued, and $x \in (0, \infty)$. Further, $\beta$ and $x$ are considered as parameters, $d_\beta$, $\hat{d}$ and $\check{d}$ being handled as univariate functions of $y$.

Let us now recall the definition of a complete monotonic function $f$.

DEFINITION 1. *A function $f$ is said to be completely monotonic (c.m.) on an interval $I$ if $f$ has derivatives of all orders on $I$ and $(-1)^n f^{(n)}(x) \geqslant 0$ for $x \in I$ and $n \geqslant 0$.*

We can now introduce the properties of concavity, convexity, and monotonicity for our convex-concave formulation of the discrete $\beta$-divergence.

PROPOSITION 4. *Given $\breve{d}(\cdot|\cdot)$ and $\widehat{d}(\cdot|\cdot)$ as defined above, we have the following:*
1. *$\breve{d}(x|y)$ is $C^\infty$ and strictly convex on $(0, \infty)$ for $x > 0$ and $\beta \in \mathbb{R}$;*
2. *$\widehat{d}(x|y)$ is concave for $x > 0$ and $\beta \in \mathbb{R}$; and*
3. *for all $\beta < 2$, $\breve{d}''(x|y)$ and $\widehat{d}''(x|y)$ are c.m.*

*Proof.* The proof is straightforward, given that $\breve{d}(x|y)$ and $\widehat{d}(x|y)$ linearly combine $C^\infty$ functions on $(0, \infty)$, and that in the same interval, we have the following:
- $\log y$ is strictly concave;
- $y^\nu$ is strictly convex for all $\nu \in (-\infty, 0) \cup (1, \infty)$, and strictly concave for all $\nu \in (0, 1)$; and
- $y^\nu$ is c.m. for all $\nu < 0$.                                                    □

According to the first two items of Proposition 4, $\breve{d}$ and $\widehat{d}$ indeed yield a convex-concave decomposition of the $\beta$-divergence, which is a variant of [12, Table 1]. Let us remark that the successive minimization of an upper approximation of this convex-concave decomposition following the methodology presented in [12] yields to the usual multiplicative update scheme.

## REFERENCES

[1] M. ABDOLALI AND N. GILLIS, *Simplex-structured matrix factorization: Sparsity-based identifiability and provably correct algorithms*, SIAM J. Math. Data Sci., 3 (2021), pp. 593–623, https://doi.org/10.1137/20M1354982.

[2] A. ANG AND N. GILLIS, *Accelerating nonnegative matrix factorization algorithms using extrapolation*, Neural Comput., 31 (2019), pp. 417–439.

[3] D. P. BERTSEKAS, *Nonlinear Programming*, 2nd ed., Athena Scientific, Belmont, MA, 1999.

[4] J. M. BIOUCAS-DIAS, A. PLAZA, N. DOBIGEON, M. PARENTE, Q. DU, P. GADER, AND J. CHANUSSOT, *Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches*, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens., 5 (2012), pp. 354–379.

[5] E. C. CHI AND T. G. KOLDA, *On tensors, sparsity, and nonnegative factorizations*, SIAM J. Matrix Anal. Appl., 33 (2012), pp. 1272–1299, https://doi.org/10.1137/110859063.

[6] A. CICHOCKI, R. ZDUNEK, AND S.-I. AMARI, *Hierarchical ALS algorithms for nonnegative matrix and 3D tensor factorization*, in Lecture Notes in Comput. Sci. 4666, Springer, Berlin, Heidelberg, 2007, pp. 169–176.

[7] A. CICHOCKI, R. ZDUNEK, A. H. PHAN, AND S.-I. AMARI, *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-Way Data Analysis and Blind Source Separation*, John Wiley & Sons, 2009.

[8] O. DIKMEN, Z. YANG, AND E. OJA, *Learning the information divergence*, IEEE Trans. Pattern Anal. Mach. Intell., 37 (2015), pp. 1442–1454.

[9] C. DING, T. LI, AND W. PENG, *On the equivalence between non-negative matrix factorization and probabilistic latent semantic indexing*, Comput. Statist. Data Anal., 52 (2008), pp. 3913–3927.

[10] J. EGGERT AND E. KORNER, *Sparse coding and NMF*, in IEEE International Joint Conference on Neural Networks, Vol. 4, IEEE, 2004, pp. 2529–2533.

[11] C. FÉVOTTE, N. BERTIN, AND J.-L. DURRIEU, *Nonnegative matrix factorization with the Itakura-Saito divergence: With application to music analysis*, Neural Comput., 21 (2009), pp. 793–830.

[12] C. FÉVOTTE AND J. IDIER, *Algorithms for nonnegative matrix factorization with the $\beta$-divergence*, Neural Comput., 23 (2011), pp. 2421–2456.

[13] X. Fu, K. Huang, N. D. Sidiropoulos, and W.-K. Ma, *Nonnegative matrix factorization for signal and data analytics: Identifiability, algorithms, and applications*, IEEE Signal Process. Mag., 36 (2019), pp. 59–80.

[14] C. Févotte and N. Dobigeon, *Nonlinear hyperspectral unmixing with robust nonnegative matrix factorization*, IEEE Trans. Image Process., 24 (2015), pp. 4810–4819.

[15] N. Gillis, *The Why and How of Nonnegative Matrix Factorization*, in Regularization, Optimization, Kernels, and Support Vector Machines, J. Suykens, M. Signoretto, and A. Argyriou, eds., Machine Learning and Pattern Recognition, Chapman & Hall/CRC, Boca Raton, FL, 2014, pp. 257–291.

[16] N. Gillis, *Nonnegative Matrix Factorization*, SIAM, Philadelphia, 2020, https://doi.org/10.1137/1.9781611976410.

[17] N. Gillis and F. Glineur, *Accelerated multiplicative updates and hierarchical ALS algorithms for nonnegative matrix factorization*, Neural Comput., 24 (2012), pp. 1085–1105.

[18] N. Guan, D. Tao, Z. Luo, and B. Yuan, *NeNMF: An optimal gradient method for nonnegative matrix factorization*, IEEE Trans. Signal Process., 60 (2012), pp. 2882–2898.

[19] L. T. K. Hien, N. Gillis, and P. Patrinos, *Inertial block proximal methods for non-convex non-smooth optimization*, in Proceedings of the 37th International Conference on Machine Learning, Proceedings of Machine Learning Research 119, PMLR, 2020, pp. 5671–5681.

[20] D. Hong, T. G. Kolda, and J. A. Duersch, *Generalized canonical polyadic tensor decomposition*, SIAM Rev., 62 (2020), pp. 133–163, https://doi.org/10.1137/18M1203626.

[21] J. Kim, Y. He, and H. Park, *Algorithms for nonnegative matrix and tensor factorizations: A unified view based on block coordinate descent framework*, J. Global Optim., 58 (2014), pp. 285–319.

[22] D. D. Lee and H. Seung, *Algorithms for non-negative matrix factorization*, in Proceedings of the 13th International Conference on Neural Information Processing Systems, NIPS, MIT Press, Cambridge, 2000, pp. 535–541.

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

[24] V. Leplat, N. Gillis, and A. M. S. Ang, *Blind audio source separation with minimum-volume beta-divergence NMF*, IEEE Trans. Signal Process., 68 (2020), pp. 3400–3410.

[25] C.-J. Lin, *Projected gradient methods for nonnegative matrix factorization*, Neural Comput., 19 (2007), pp. 2756–2779.

[26] W.-K. Ma, J. M. Bioucas-Dias, T. Chan, N. Gillis, P. Gader, A. J. Plaza, A. Ambikapathi, and C. Chi, *A signal processing perspective on hyperspectral unmixing: Insights from remote sensing*, IEEE Signal Process. Mag., 31 (2014), pp. 67–81.

[27] K. Miller and G. Samko, *Completely monotonic functions*, Integral Transform. Spec. Funct., 12 (2001), pp. 389–402.

[28] K. Neymeyr and M. Sawall, *On the set of solutions of the nonnegative matrix factorization problem*, SIAM J. Matrix Anal. Appl., 39 (2018), pp. 1049–1069, https://doi.org/10.1137/17M1118439.

[29] J. Ortega and W. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970.

[30] J. L. Roux, F. J. Weninger, and J. R. Hershey, *Sparse NMF – Half-baked or Well Done?*, Tech. Report, Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA, 2015.

[31] Y. Sun, P. Babu, and D. Palomar, *Majorization-minimization algorithms in signal processing, communications, and machine learning*, IEEE Trans. Signal Process., 65 (2017), pp. 794–816.

[32] T. Virtanen, *Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria*, IEEE Trans. Audio Speech Lang. Process., 15 (2007), pp. 1066–1074.

[33] Z. Yang, J. Corander, and E. Oja, *Low-rank doubly stochastic matrix decomposition for cluster analysis*, J. Mach. Learn. Res., 17 (2016), pp. 6454–6478.

[34] A. L. Yuille and A. Rangarajan, *The concave-convex procedure*, Neural Comput., 15 (2003), pp. 915–936, https://doi.org/10.1162/08997660360581958.

[35] F. Zhu, *Hyperspectral Unmixing: Ground Truth Labeling, Datasets, Benchmark Performances and Survey*, preprint, https://arxiv.org/abs/1708.05125, 2017.