

UC San Diego

UC San Diego Previously Published Works

Title

A unified framework for sparse non-negative least squares using multiplicative updates and the non-negative matrix factorization problem

Permalink

<https://escholarship.org/uc/item/66c2c4nj>

Authors

Fedorov, Igor
Nalci, Aican
Giri, Ritwik
et al.

Publication Date

2018-05-01

DOI

10.1016/j.sigpro.2018.01.001

Peer reviewed



Published in final edited form as:

Signal Processing. 2018 May ; 146: 79–91. doi:10.1016/j.sigpro.2018.01.001.

A unified framework for sparse non-negative least squares using multiplicative updates and the non-negative matrix factorization problem

Igor Fedorov^{a,*}, Alican Nalci^a, Ritwik Giri^b, Bhaskar D. Rao^a, Truong Q. Nguyen^a, and Harinath Garudadri^a

^aUniversity of California, San Diego, 9500 Gilman Dr, San Diego, CA 92103, USA

^bStarkey Hearing Technologies, 6700 Washington Avenue S, Eden Prairie, MN 55344, USA

Abstract

We study the sparse non-negative least squares (S-NNLS) problem. S-NNLS occurs naturally in a wide variety of applications where an unknown, non-negative quantity must be recovered from linear measurements. We present a unified framework for S-NNLS based on a rectified power exponential scale mixture prior on the sparse codes. We show that the proposed framework encompasses a large class of S-NNLS algorithms and provide a computationally efficient inference procedure based on multiplicative update rules. Such update rules are convenient for solving large sets of S-NNLS problems simultaneously, which is required in contexts like sparse non-negative matrix factorization (S-NMF). We provide theoretical justification for the proposed approach by showing that the local minima of the objective function being optimized are sparse and the S-NNLS algorithms presented are guaranteed to converge to a set of stationary points of the objective function. We then extend our framework to S-NMF, showing that our framework leads to many well known S-NMF algorithms under specific choices of prior and providing a guarantee that a popular subclass of the proposed algorithms converges to a set of stationary points of the objective function. Finally, we study the performance of the proposed approaches on synthetic and real-world data.

Keywords

Sparsity; Non-negativity; Dictionary learning

1. Introduction

Least squares problems occur naturally in numerous research and application settings. At a high level, given an observation $x \in \mathbb{R}^d$ of $h \in \mathbb{R}^n$ through a linear system $W \in \mathbb{R}^{d \times n}$, the least squares problem refers to

*Corresponding author. ifedorov@eng.ucsd.edu (I. Fedorov).

²Available at <http://cbcl.mit.edu/cbcl/software-datasets/FaceData.html>.

$$\arg \min_h \|x - Wh\|_2^2. \quad (1)$$

Quite often, prior information about h is known. For instance, h may be known to be non-negative. Non-negative data occurs naturally in many applications, including text mining [1], image processing [2], speech enhancement [3], and spectral decomposition [4,5]. In this case, (1) is modified to

$$\operatorname{argmin}_{h \geq 0} \|x - Wh\|_2^2 \quad (2)$$

where $h \geq 0$ refers to the elements of h being constrained to be non-negative and (2) is referred to as the non-negative least squares (NNLS) problem. A solution to (2) can be obtained using the well-known active set Lawson–Hanson algorithm [6] or one of its many variants [7]. In this work, we are interested in a specific flavor of NNLS problems where $n > d$. Under this constraint, the linear system in (2) is underdetermined and admits an infinite number of solutions. To constrain the set of possible solutions, a sparsity constraint on h can be added, leading to a sparse NNLS (S-NNLS) formulation:

$$\operatorname{argmin}_{h \geq 0, \|h\|_0 \leq k} \|x - Wh\|_2^2 \quad (3)$$

where $\|\cdot\|_0$ refers to the ℓ_0 pseudo-norm, which counts the number of non-zero entries. Solving (3) directly is difficult because the ℓ_0 pseudo-norm is non-convex. In fact, solving (3) requires a combinatorial search and has been shown to be NP-hard [8]. Therefore, greedy methods have been adopted to approximate the solution [8,9]. One effective approach, called reverse sparse NNLS (rsNNLS) [10], first finds an h such that $\|x - Wh\|_2^2 \leq \delta$ using the active-set Lawson–Hanson algorithm and then prunes h with a greedy procedure until $\|h\|_0 \leq k$, all while maintaining $h \geq 0$. Other approaches include various relaxations of the ℓ_0 pseudo-norm in (3) using the ℓ_1 norm [11] or a combination of the ℓ_1 and ℓ_2 norms [12], leading to easier optimization problems.

The purpose of this work is to address the S-NNLS problem in a setting often encountered by practitioners, i.e. when several S-NNLS problems must be solved simultaneously. We are primarily motivated by the problem of sparse non-negative matrix factorization (S-NMF). NMF falls under the category of dictionary learning algorithms. Dictionary learning is a common ingredient in many signal processing and machine learning algorithms [13–16]. In NMF, the data, the dictionary, and the encoding of the data under the dictionary are all restricted to be non-negative. Constraining the encoding of the data to be non-negative leads to the intuitive interpretation of the data being decomposed into an additive combination of dictionary atoms [17–19]. More formally, let $X \in \mathbb{R}_+^{d \times m}$ be a matrix representing the given

data, where each column of X , $X_{(:,j)} \in \mathbb{R}_+^d$, $1 \leq j \leq m$, is a data vector. The goal of NMF is to decompose X into two matrices $W \in \mathbb{R}_+^{d \times n}$ and $H \in \mathbb{R}_+^{n \times m}$. When $n < d$, NMF is often stated in terms of the optimization problem

$$\theta^* = \operatorname{argmin}_{\theta \geq 0} \|X - WH\|_F^2 \quad (4)$$

where $\theta = \{W, H\}$, W is called the dictionary, H is the encoding of the data under the dictionary, and $\theta \geq 0$ is short-hand for the elements of W and H being constrained to be non-negative. Optimizing (4) is difficult because it is not convex in θ [20]. Instead of performing joint optimization, a block coordinate descent method [21] is usually adopted where the algorithm alternates between holding W fixed while optimizing H and vice versa [17,19,20,22,23]:

$$\text{Update } W \text{ given } H \quad (5)$$

$$\text{Update } H \text{ given } W. \quad (6)$$

Note that (5) and (6) are a collection of d and m NNLS problems, respectively, which motivates the present work. The block coordinate descent method is advantageous because (5) and (6) are convex optimization problems for the objective function in (4), so that any number of techniques can be employed within each block. One of the most widely used optimization techniques, called the multiplicative update rules (MUR's), performs (5)–(6) using simple element-wise operations on W and H [17,19]:

$$W^{t+1} = W^t \odot \frac{XH^T}{W^t H H^T} \quad (7)$$

$$H^{t+1} = H^t \odot \frac{W^T X}{W^T W H^t} \quad (8)$$

where \odot denotes element-wise multiplication, A/B denotes element-wise division of matrices A and B , and t denotes the iteration index. The MUR's shown in (7)–(8) are guaranteed to not increase the objective function in (4) [17,19] and, due to their simplicity, are widely used in the NMF community [24–26]. The popularity of NMF MUR's persists despite the fact that there is no guarantee that the sequence $\{W^t, H^t\}_{t=0}^\infty$ generated by (7)–(8) will converge to a local minimum [27] or even a stationary point [20,27] of (4).

Unlike traditional NMF methods [17,19], this work considers the scenario where W is overcomplete, i.e. $n \gg d$. Overcomplete dictionaries have much more flexibility to represent diverse signals [28] and, importantly, lead to effective sparse and low dimensional representations of the data [18,28]. As in NNLS, the concept of sparsity has an important role in NMF because when W is overcomplete, (4) is not well-posed without some additional regularization. Sparsity constraints limit the set of possible solutions of (4) and, in some cases, lead to guarantees of uniqueness [29]. The S-NMF problem can be stated as the solution to

$$\theta^* = \underset{\theta \geq 0, \|H\|_0 \leq k}{\operatorname{argmin}} \|X - WH\|_F^2 \quad (9)$$

where $\|H\|_0 \leq k$ is shorthand for $\{\|H(:, j)\|_0 \leq k\}_{j=1}^m$. One classical approach to S-NMF relaxes the ℓ_0 constraint and appends a convex, sparsity promoting ℓ_1 penalty to the objective function [11]:

$$\theta^* = \underset{\theta \geq 0}{\operatorname{argmin}} \|X - WH\|_F^2 + \lambda \|H\|_1 \quad (10)$$

where $\|H\|_1$ is shorthand for $\sum_{j=1}^m \|H(:, j)\|_1$. As shown in [11], (10) can be iteratively minimized through a sequence of multiplicative updates where the update of W is given by (7) and the update of H is given by

$$H^{t+1} = H^t \odot \frac{W^T X}{W^T W H^t + \lambda}. \quad (11)$$

We also consider an extension of S-NMF where a sparsity constraint is placed on W [12]

$$\theta^* = \underset{\theta \geq 0, \|H\|_0 \leq k_H, \|W\|_0 \leq k_W}{\operatorname{argmin}} \|X - WH\|_F^2 \quad (12)$$

which encourages basis vectors that explain localized features of the data [12]. We refer to (12) as S-NMF-W.

The motivation of this work is to develop a maximum a posteriori (MAP) estimation framework to address the S-NNLS and S-NMF problems. We build upon the seminal work in [30] on Sparse Bayesian Learning (SBL). The SBL framework places a sparsity-promoting prior on the data [31] and has been shown to give rise to many models used in the compressed sensing literature [32]. It will be shown that the proposed framework provides a general class of algorithms that can be tailored to the specific needs of the user. Moreover,

inference can be done through a simple MUR for the general model considered and the resulting S-NNLS algorithms admit convergence guarantees.

The key contribution of this work is to detail a unifying framework that encompasses a large number of existing S-NNLS and S-NMF approaches. Therefore, due to the very nature of the framework, many of the algorithms presented in this work are not new. Nevertheless, there is value in the knowledge that many of the algorithms employed by researchers in the S-NNLS and S-NMF fields are actually members of the proposed family of algorithms. In addition, the proposed framework makes the process of formulating novel task-specific algorithms easy. Finally, the theoretical analysis of the proposed framework applies to any member of the family of proposed algorithms. Such an analysis has value to both existing S-NNLS and S-NMF approaches like [33,34], which do not perform such an analysis, as well as to any future approaches which fall under the umbrella of the proposed framework. It should be noted that several authors have proposed novel sets of MUR's with provable convergence guarantees for the NMF problem in (4) [35] and S-NMF problem in (10) [36]. In contrast to Zhao and Tan [36], the proposed framework does not use the ℓ_1 regularization function to solve (9). In addition, since the proposed framework encompasses the update rules used in existing works, the analysis presented here applies to works from existing literature, including [33,34].

1.1. Contributions of the paper

- A general class of rectified sparsity promoting priors is presented and it is shown that the computational burden of the resulting inference procedure is handled by a class of simple, low-complexity MUR's.
- A monotonicity guarantee for the proposed class of MUR's is provided, justifying their use in S-NNLS and S-NMF algorithms.
- A convergence guarantee for the proposed class of S-NNLS and S-NMF-W algorithms is provided.

1.2. Notation

Bold symbols are used to denote random variables and plain font to denote a particular realization of a random variable. MATLAB notation is used to denote the (i, j) th element of the matrix H as $H_{(i, j)}$ and the j th column of H as $H_{(:, j)}$. We use H^s to denote the matrix H at iteration s of a given algorithm and $(H)^z$ to denote the matrix H with each element raised to the power z .

2. Sparse non-negative least squares framework specification

The S-NNLS signal model is given by

$$X = \mathbf{W}\mathbf{H} + \mathbf{V} \quad (13)$$

where the columns of \mathbf{V} , the noise matrix, follow a $N(0, \sigma^2\mathbf{I})$ distribution. To complete the model, a prior on the columns of \mathbf{H} , which are assumed to be independent and identically

distributed, must be specified. This work considers separable priors of the form $p(\mathbf{H}_{(:,j)}) = \prod_{i=1}^n p(H_{(i,j)})$, where $p(H_{(i,j)})$ has a scale mixture representation [37,38]:

$$p(H_{(i,j)}) = \int_0^\infty p(H_{(i,j)}|\gamma_{(i,j)})p(\gamma_{(i,j)})d\gamma_{(i,j)}. \quad (14)$$

Separable priors are considered because, in the absence of prior knowledge, it is reasonable to assume independence amongst the coefficients of \mathbf{H} . The case where dependencies amongst the coefficients exist is considered in Section 5. The proposed framework extends the work on power exponential scale mixtures [39,40] to rectified priors and uses the Rectified Power Exponential (RPE) distribution for the conditional density of $\mathbf{H}_{(i,j)}$ given $\gamma_{(i,j)}$:

$$p^{\text{RPE}}(H_{(i,j)}|\gamma_{(i,j)}; z) = \frac{e^{-\left(\frac{H_{(i,j)}}{\gamma_{(i,j)}}\right)^z}}{\gamma_{(i,j)}\Gamma\left(\frac{1}{z}\right)}u(H_{(i,j)})$$

where $u(\cdot)$ is the unit-step function, $0 < z \leq 2$, and $\int_0^\infty t^{a-1}e^{-t}dt$. The RPE distribution is chosen for its flexibility. In this context, (14) is referred to as a rectified power exponential scale mixture (RPESM).

The advantage of the scale mixture prior is that it introduces a Markovian structure of the form

$$\gamma_{(:,j)} \rightarrow \mathbf{H}_{(:,j)} \rightarrow \mathbf{X}_{(:,j)} \quad (15)$$

and inference can be done in either the \mathbf{H} or γ domains. This work focuses on doing MAP inference in the \mathbf{H} domain, which is also known as Type 1 inference, whereas inference in the γ domain is referred to as Type 2. The scale mixture representation is flexible enough to represent most heavy-tailed densities [41–45], which are known to be the best sparsity promoting priors [30,46]. One reason for the use of heavy-tailed priors is that they are able to model both the sparsity and large non-zero entries of \mathbf{H} .

The RPE encompasses many rectified distributions of interest. For instance, the RPE reduces to a Rectified Gaussian by setting $z = 2$, which is a popular prior for modeling non-negative data [38,47] and results in a Rectified Gaussian Scale Mixture in (14). Setting $z = 1$ corresponds to an Exponential distribution and leads to an Exponential Scale Mixture in (14) [48]. Table 2 shows that many rectified sparse priors of interest can be represented as a RPESM. Distributions of interest are summarized in Table 1.

3. Unified MAP inference procedure

In the MAP framework, H is directly estimated from X by minimizing

$$L(H) = -\log \left(\prod_{j=1}^m p(H_{(:,j)} | X_{(:,j)}) \right). \quad (16)$$

We have made the dependence of the negative log-likelihood on X and W implicit for brevity. Minimizing (16) in closed form is intractable for most priors, so the proposed framework resorts to an Expectation-Maximization (EM) approach [45]. In the E-step, the expectation of the negative complete data log-likelihood with respect to the distribution of γ , conditioned on the remaining variables, is formed:

$$Q(H, \bar{H}^t) \doteq \left\| X - WH \right\|_F^2 + \lambda \left(\sum_{i=1, j=1}^{i=n, j=m} (H_{(i,j)})^2 \left\langle \frac{1}{(\gamma_{(i,j)})^z} \right\rangle - \log u(H_{(i,j)}) \right) \quad (17)$$

where $\langle \cdot \rangle$ refers to the expectation with respect to the density $p(\gamma_{(i,j)} | \bar{H}_{(i,j)}^t)$, t refers to the iteration index, \bar{H}^t denotes the estimate of H at the t th EM iteration, and \doteq refers to dropping terms that do not influence the M-step and scaling by $\lambda = 2\sigma^2$. The last term in (17) acts as a barrier function against negative values of H . The function $Q(H, \bar{H}^t)$ is separable in the columns of H . In an abuse of notation, we use $Q(H_{(:,j)}, \bar{H}_{(:,j)}^t)$ to refer to the dependency of $Q(H, \bar{H}^t)$ on $H_{(:,j)}$.

In order to compute the expectation in (17), a similar method to the one used in [39,41] is employed, with some minor adjustments due to non-negativity constraints. Let $p(H_{(i,j)}) = p^R(H_{(i,j)})u(H_{(i,j)})$, where $p^R(H_{(i,j)})$ is the portion of $p(H_{(i,j)})$ that does not include the rectification term, and let $p^R(H_{(i,j)})$ be differentiable on $[0, \infty)$. Then,

$$\left\langle \frac{1}{(\gamma_{(i,j)})^z} \right\rangle = -\frac{\partial \log p^R(\bar{H}_{(i,j)}^t)}{\partial \bar{H}_{(i,j)}^t} \frac{1}{z(\bar{H}_{(i,j)}^t)^{z-1}}. \quad (18)$$

Turning to the M-step, the proposed approach employs the Generalized EM (GEM) M-step [45]:

$$\bar{H}^{t+1} \text{ such that } Q(\bar{H}^{t+1}, \bar{H}^t) \leq Q(\bar{H}^t, \bar{H}^t). \quad (\text{GEM M-step})$$

In particular, $Q(H, \bar{H}^t)$ is minimized through an iterative gradient descent procedure. As with any gradient descent approach, selection of the learning rate is critical in order to ensure that the objective function is decreased and the problem constraints are met. Following Lee and

Seung [17,19], the learning rate is selected such that the gradient descent update is guaranteed to generate non-negative updates and can be implemented as a low-complexity

$$H^{s+1} = H^s \odot \frac{W^T X}{W^T W H^s + \lambda \Omega^t \odot (H^s)^{z-1}} \quad (19)$$

$$\Omega^t_{(i,j)} = - \left| \frac{1}{(\bar{H}^t_{(i,j)})^{z-1}} \frac{\partial \log p^R(\bar{H}^t_{(i,j)})}{\partial \bar{H}^t_{(i,j)}} \right|$$

where s denotes the gradient descent iteration index (not to be confused with the EM iteration index t). The resulting S-NNLS algorithm is summarized in Algorithm 1, where ζ denotes the specific MUR used to update H , which is (19) in this case.

3.1. Extension to S-NMF

We now turn to the extension of our framework to the S-NMF problem. As before, the signal model in (13) is used as well as the RPESM prior on H . To estimate W and H , the proposed framework seeks to find

$$\operatorname{argmin}_{W, H} L^{NMF}(W, H), \quad L^{NMF}(W, H) = -\log p(W, H | X). \quad (20)$$

Algorithm 1 S-NNLS algorithm.

Require: $X, W, \bar{H}^0, \lambda, \zeta, S, t^\infty$
Initialize $t = 0, \mathcal{Z} = \{(i, j)\}_{i=1, j=1}^{i=n, j=m}$
while $\mathcal{Z} \neq \emptyset$ **do**
 Form Ω^t and initialize $H^1 = \bar{H}^t, \mathcal{J} = \mathcal{Z}$
 for $s = 1$ to S **do**
 Generate $H_{(i,j)}^{s+1}$ using update rule ζ for $(i, j) \in \mathcal{J}$
 Set $H_{(i,j)}^{s+1} = H_{(i,j)}^s$ for any $(i, j) \notin \mathcal{J}$
 $\mathcal{J} \leftarrow \mathcal{J} \setminus \{(i, j) : H_{(i,j)}^{s+1} = 0 \text{ or } H_{(i,j)}^{s+1} = H_{(i,j)}^s\}$
 end for
 Set $\bar{H}^{t+1} = H^{S+1}$ and $\mathcal{Z} \leftarrow \mathcal{Z} \setminus \{(i, j) : \bar{H}_{(i,j)}^{t+1} = \bar{H}_{(i,j)}^t \text{ or } \bar{H}_{(i,j)}^{t+1} = 0\}$
 $t \leftarrow t + 1$
 if $t = t^\infty$ **then**
 Break
 end if
end while
Return \bar{H}^t

The random variables W and H are assumed independent and a non-informative prior over the positive orthant is placed on W for S-NMF. For S-NMF-W, a separable prior from the RPESM family is assumed for W . In order to solve (20), the block-coordinate descent optimization approach in (5)–(6) is employed. For each one of (5) and (6), the GEM procedure described above is used.

The complete S-NMF/S-NMF-W algorithm is given in Algorithm 2. Due to the symmetry between (5) and (6) and to avoid unnecessary repetition, heavy use of Algorithm 1 in Algorithm 2 is made. Note that $\zeta_h = (19)$, $\zeta_w = (8)$ for S-NMF, and $\zeta_w = (19)$ for S-NMF-W.

Algorithm 2 S-NMF/S-NMF-W algorithm.

Require: $X, \lambda, S, \zeta_w, \zeta_h, t^\infty$
 Initialize $W_{(i,j)}^0 = 1, H_{(i,j)}^0 = 1, t = 0$
while $t \neq t^\infty$ and $(\tilde{H}^{t+1} \neq \tilde{H}^t$ or $\tilde{W}^{t+1} \neq \tilde{W}^t)$ **do**
 $\tilde{W}^{t+1} = (\text{Algorithm1}(X^T, (\tilde{H}^t)^T, (\tilde{W}^t)^T, \lambda, \zeta_w, S, 1))^T$
 $\tilde{H}^{t+1} = \text{Algorithm1}(X, \tilde{W}^{t+1}, \tilde{H}^t, \lambda, \zeta_h, S, 1)$
 $t \leftarrow t + 1$
end while

4. Examples of S-NNLS and S-NMF algorithms

In the following, evidence of the utility of the proposed framework is provided by detailing several specific algorithms which naturally arise from (19) with different choices of prior. It will be shown that the algorithms described in this section are equivalent to well-known S-NNLS and S-NMF algorithms, but derived in a completely novel way using the RPESM prior. The S-NMF-W algorithms described are, to the best of our knowledge, novel. In Section 5, it will be shown that the proposed framework can be easily used to define novel algorithms where block-sparsity is enforced.

4.1. Reweighted l_2

Consider the prior $\mathbf{H}_{(i,j)} \sim p^{\text{RST}}(H_{(i,j)}; \tau)$. Given this prior, (19) becomes

$$H^{s+1} = H^s \odot \frac{W^T X}{W^T W H^s + \frac{2\lambda(\tau+1)H^s}{\tau + (\bar{H}^t)^2}}. \quad (21)$$

Given this choice of prior on $\mathbf{H}_{(i,j)}$ and a non-informative prior on $\mathbf{W}_{(i,j)}$, it can be shown that $L^{\text{NMF}}(W, H)$ reduces to

$$\left\| X - WH \right\|_F^2 + \tilde{\lambda} \sum_{i=1, j=1}^{i=n, j=m} \log\left((H_{(i,j)})^2 + \tau\right) \quad (22)$$

over $H \in \mathbb{R}_+^{n \times m}$ and $W \in \mathbb{R}_+^{d \times n}$ (i.e. $\log u(\cdot)$ term have been omitted for and brevity), where $\tilde{\lambda} = 2\sigma^2(\tau + 1)$. The sparsity-promoting regularization term in (22) was first studied in [49] in the context of vector sparse coding (i.e. without non-negativity constraints). Majorizing the sparsity promoting term in (22), it can be shown that (22) is upper-bounded by

$$\left\| X - WH \right\|_2^2 + \tilde{\lambda} \left\| \frac{H}{Q^t} \right\|_F^2 \quad (23)$$

where $Q_{(i,j)}^t = \bar{H}_{(i,j)}^t + \tau$. Note that this objective function was also used in [50], although it was optimized using a heuristic approach based on the Moore–Penrose pseudoinverse operator. Letting $R = H/Q^t$ and $\tilde{\lambda} \rightarrow 0$, (23) becomes

$$\|X - W(Q^t \odot R)\|_2^2 \quad (24)$$

which is exactly the objective function that is iteratively minimized in the NUIRLS algorithm [34] if we let $\tau \rightarrow 0$. Although Grady and Rickard [34] gives a MUR for minimizing (24), the MUR can only be applied for each column of H individually. It is not clear why the authors of Grady and Rickard [34] did not give a matrix based update rule for minimizing (24), which can be written as

$$R^{s+1} = R^s \odot \frac{W^T X}{W^T W(Q^t \odot R^s)}.$$

This MUR is identical to (21) in the setting $\lambda, \tau \rightarrow 0$. Although Grady and Rickard [34] makes the claim that NUIRLS converges to a local minimum of (24), this claim is not proved. Moreover, nothing is said regarding convergence with respect to the actual objective function being minimized (i.e. (22) as opposed to the majorizing function in (24)). As the analysis in Section 6 will reveal, using the update rule in (21) within Algorithm 1, the iterates are guaranteed to converge to a stationary point of (22). We make no claims regarding convergence with respect to the majorizing function in (23) or (24).

4.2. Reweighted ℓ_1

Assuming $H_{(i,j)} \sim p^{\text{RGDP}}(H_{(i,j)}; 1, 1, \tau)$, (19) reduces to

$$H^{s+1} = H^s \odot \frac{W^T X}{W^T W H^s + \frac{\lambda(\tau+1)}{\tau + \bar{H}^t}}. \quad (25)$$

Plugging the RGDP prior into (20) and assuming a non-informative prior on $W_{(i,j)}$ leads to the Lagrangian of the objective function considered in [51] for unconstrained vector sparse coding (after omitting the barrier function terms): $\|X - WH\|_F^2 + \tilde{\lambda} \sum_{i=1}^n \sum_{j=1}^m \log(H_{(i,j)} + \tau)$.

Interestingly, this objective function is a special case of the block sparse objective considered in [33] (where the Itakura–Saito reconstruction loss is used instead of the Frobenius norm loss) if each $H_{(i,j)}$ is considered a separate block. Lefevre et al. [33] did not offer a convergence analysis of their algorithm, in contrast with the present work. To the best of our knowledge, the reweighted ℓ_1 formulation has not been considered in the S-NNLS literature.

4.3. Reweighted ℓ_2 and reweighted ℓ_1 for S-NMF-W

Using the reweighted ℓ_2 or reweighted ℓ_1 formulations to promote sparsity in W is straightforward in the proposed framework and involves setting ζ_w to (21) or (25), respectively, in Algorithm 2.

5. Extension to block sparsity

As a natural extension of the proposed framework, we now consider the block sparse S-NNLS problem. This section will focus on the S-NNLS context only because the extension to S-NMF is straightforward. Block sparsity arises naturally in many contexts, including speech processing [24,52], image denoising [53], and system identification [54]. The central idea behind block-sparsity is that W is assumed to be divided into disjoint blocks and each $X_{(:,j)}$ is assumed to be a linear combination of the elements of a *small number of blocks*. This constraint can be easily accommodated by changing the prior on $H_{(:,j)}$ to a block rectified power exponential scale mixture:

$$p(H_{(:,j)}) = \prod_{g_b \in \mathcal{G}} \underbrace{\int_0^\infty \prod_{i \in g_b} p(H_{(i,j)} | \gamma_{(b,j)}) p(\gamma_{(b,j)}) d\gamma_{(b,j)}}_{p(H_{(g_b,j)})} \quad (26)$$

where \mathcal{G} is a disjoint union of $\{g_b\}_{b=1}^B$ and $H_{(g_b,j)}$ is a vector consisting of the elements of

$H_{(:,j)}$ whose indices are in g_b . To find the MAP estimate of H given X , the same GEM procedure as before is employed, with the exception that the computation of the weights in (17) is modified to:

$$\left\langle \frac{1}{(\gamma_{(b,j)})^z} \right\rangle = - \frac{\partial \log p^R(\bar{H}_{(g_b,j)})}{\partial \bar{H}_{(i,j)}} \frac{1}{z(\bar{H}_{(i,j)})^{z-1}}$$

where $i \in g_b$. It can be shown that the MUR for minimizing $Q(H, \bar{H}^t)$ in (19) can be modified to account for the block prior in (26) to

$$H^{s+1} = H^s \odot \frac{W^1 X}{W^T W H^s + \lambda \Phi^t \odot (H^s)^{z-1}} \quad (27)$$

$$\Phi^t_{(g_b,j)} = - \frac{1}{(\bar{H}^t_{(i,j)})^{z-1}} \frac{\partial \log p^R(\bar{H}^t_{(g_b,j)})}{\partial \bar{H}^t_{(i,j)}} \text{ for any } i \in g_b.$$

Next, we show examples of block S-NNLS algorithms that arise from our framework.

5.1. Example: reweighted ℓ_2 block S-NNLS

Consider the block-sparse prior in (26), where $p(H_{(i,j)}|\gamma_{(b,j)})$, $i \in g_b$ is a RPE with $z = 2$ and $\gamma_{(b,j)} \sim p^{\text{Ga}}(\gamma_{(b,j)}; \tau/2, \tau/2)$. The resulting density $p(H_{(:,j)})$ is a block RST (BRST) distribution:

$$p(H_{(:,j)}) = \left(\prod_{g_0 \in \mathcal{G}} \frac{2\Gamma(\frac{\tau+1}{2})}{\sqrt{\pi\tau}\Gamma(\frac{\tau}{2})} \left(1 + \frac{\|H_{(g_b,j)}\|_2^2}{\tau} \right)^{-\frac{(\tau+1)}{2}} \right) \prod_{i=1}^n u(H_{(i,j)}).$$

The MUR for minimizing $Q(H, \bar{H}^t)$ under the BRST prior is given by:

$$H^{s+1} = H^s \odot \frac{W^T X}{W^T W H^s + \frac{2\lambda(\tau+1)H^s}{\tau + S^t}} \quad (28)$$

$$\text{where } S_{(g_b,j)}^t = \left\| \bar{H}_{(g_b,j)}^t \right\|_2^2.$$

5.2. Example: reweighted ℓ_1 block S-NNLS

Consider the block-sparse prior in (26), where $p(H_{(i,j)}|\gamma_{(b,j)})$, $i \in g_b$ is a RPE with $z = 1$ and $\gamma_{(b,j)} \sim p^{\text{Ga}}(\gamma_{(b,j)}; \tau, \tau)$. The resulting density $p(H_{(:,j)})$ is a block rectified generalized double pareto (BRGDP) distribution:

$$p(H_{(:,j)}) = \left(\prod_{g_b \in \mathcal{G}} 2\eta \left(1 + \frac{\|H_{(g_b,j)}\|_1}{\tau} \right)^{-(\tau+1)} \right) \prod_{i=1}^n u(H_{(i,j)}).$$

The MUR for minimizing $Q(H, \bar{H}^t)$ under the BRGDP prior is given by:

$$H^{t+1} = \bar{H}^t \odot \frac{W^T X}{W^T W \bar{H}^t + \frac{\lambda(\tau+1)}{\tau + V^t}} \quad (29)$$

$$\text{where } V_{(g_b,j)}^t = \left\| \bar{H}_{(g_b,j)}^t \right\|_1.$$

5.3. Relation to existing block sparse approaches

Block sparse coding algorithms are generally characterized by their block-sparsity measure. The analog of ℓ_0 sparsity measure for block-sparsity is the $\ell_2 - \ell_0$ measure

$\sum_{g_b \in \mathcal{G}} 1_{\left\|H_{(g_b, j)}\right\|_2 > 0}$, which simply counts the number of blocks with non-zero energy.

This sparsity measure has been studied in the past and block versions of the popular MP and OMP algorithms have been extended to Block-MP (BMP) and Block-OMP (BOMP) [55]. Extending BOMP to non-negative BOMP (NNBOMP) is straightforward, but details are omitted due to space considerations. One commonly used block sparsity measure in the NMF literature is the log $-l_1$ measure [33]: $\sum_{g_h \in \mathcal{G}} \log \left(\left\|H_{(g_h, j)}\right\|_1 + \tau \right)$. This sparsity measure arises naturally in the proposed S-NNLS framework when the BRGDP prior is plugged into (16). We are not aware of any existing algorithms which use the sparsity measure induced by the BRST prior: $\sum_{g_b \in \mathcal{G}} \log \left(\left\|H_{(g_b, j)}\right\|_2^2 + \tau \right)$.

6. Analysis

In this section, important properties of the proposed framework are analyzed. First, the properties of the framework as it applies to S-NNLS are studied. Then, the proposed framework is studied in the context of S-NMF and S-NMF-W.

6.1. Analysis in the S-NNLS setting

We begin by confirming that (GEM M-step) does not have a trivial solution at $H_{(i, j)} = \infty$ for any (i, j) because $\langle (\gamma_{(i, j)})^{-z} \rangle > 0$, since it is an expectation of a non-negative random variable. In the following discussion, it will be useful to work with distributions whose functional dependence on $H_{(i, j)}$ has a power function form:

$$f(H_{(i, j)}, z, \tau, \alpha) = \left(\tau + \left(H_{(i, j)} \right)^z \right)^{-\alpha} \quad (30)$$

where $\tau, \alpha > 0$ and $0 < z \leq 2$. Note that the priors considered in this work have a power function form.

6.1.1. Monotonicity of $Q(H, \bar{H}^l)$ under (19)—The following theorem states one of the main contributions of this work, validating the use of (19) in (GEM M-step).

Theorem 1.: Let $z \in \{1, 2\}$ and the functional dependence of $p^R(H_{(i, j)})$ on $H_{(i, j)}$ have a power function form. Consider using the update rule stated in (19) to update $H_{(i, j)}^S$ for all $(i, j) \in \mathcal{J} = \{(i, j): H_{(i, j)}^S > 0\}$. Then, the update rule in (19) is well defined and $Q(H^{S+1}, \bar{H}^l) \leq Q(H^S, \bar{H}^l)$.

Proof.: Proof provided in Appendix A.

Theorem 1 also applies to the block-sparse MUR in (27).

6.1.2. Local minima of $L(H)$ —Before proceeding to the analysis of the convergence of Algorithm 1, it is important to consider the question as to whether the local minima of $L(H)$ are desirable solutions from the standpoint of being sparse.

Theorem 2.: Let H^* be a local minimum of (16) and let the functional dependence of $p^R(H_{(i,j)})$ on $H_{(i,j)}$ have a power function form. In addition, let one of the following conditions be satisfied: 1) $z = 1$ or 2) $z > 1$ and $\tau \rightarrow 0$. Then, $\|H_{(:,j)}^*\|_0 \leq d$.

Proof.: Proof provided in Appendix B.

6.1.3. Convergence of Algorithm 1—First, an important property of the cost function in (16) can be established.

Theorem 3.: The function $-\log p(H_{(i,j)})$ is coercive for any member of the RPESM family.

Proof.: The proof is provided in Appendix C.

Theorem 3 can then be used to establish the following corollary.

Corollary 1.: Assume the signal model in (13) and let $p(H_{(i,j)})$ be a member of the RPESM family. Then, the cost function $L(H)$ in (16) is coercive.

Proof.: This follows from the fact that $\|X - WH\|_F^2 \geq 0$ and the fact that $-\log p(H_{(i,j)})$ is coercive due to Theorem 3.

The coercive property of the cost function in (16) allows us to establish the following result concerning Algorithm 1.

Corollary 2.: Let $z \in \{1, 2\}$ and the functional dependence of $p^R(H_{(i,j)})$ on $H_{(i,j)}$ have a power function form. Then, the sequence $\{\bar{H}^t\}_{t=1}^\infty$ produced by Algorithm 1 with S , the number of inner loop iterations, to 1 admits at least one limit point.

Proof.: The proof is provided in Appendix D.

We are now in a position to state one of the main contributions of this paper regarding the convergence of Algorithm 1 to the set of stationary points of (16). A stationary point is defined to be any point satisfying the Karush–Kuhn–Tucker (KKT) conditions for a given optimization problem [56].

Theorem 4.: Let $z \in \{1, 2\}$, $\zeta = (19)$, $t^\infty = \infty$, $S = 1$, the functional dependence of $p^R(H_{(i,j)})$ on $H_{(i,j)}$ have a power function form, the columns of W and X have bounded norm, and W be full rank. In addition, let one of the following conditions be satisfied: (1) $z = 1$ and $\tau = \lambda / \max_{i,j} (W^T X)_{(i,j)}$ or (2) $z = 2$ and $\tau \rightarrow 0$. Then the sequence $\{\bar{H}^t\}_{t=1}^\infty$

produced by Algorithm 1 is guaranteed to converge to the set of stationary points of $L(H)$. Moreover, $\{L(\bar{H}^t)\}_{t=1}^{\infty}$ converges monotonically to $L(\bar{H}^*)$, for stationary point \bar{H}^* .

Proof.: The proof is provided in Appendix E.

The reason that $S = 1$ is specified in Theorem 4 is that it allows for providing convergence guarantees for Algorithm 1 without needing any convergence properties of the sequence generated by (19). Theorem 4 also applies to Algorithm 1 when the block-sparse MUR in (27) is used. To see the intuition behind the proof of Theorem 4 (given in Appendix E), consider the visualization of Algorithm 1 shown in Fig. 1. The proposed framework seeks a minimum of $-\log p(H_{(:,j)}|X_{(:,j)})$, for all j , through an iterative optimization procedure. At each iteration, $-\log p(H_{(:,j)}|X_{(:,j)})$ is bounded by the auxiliary function $Q(H_{(:,j)}, \bar{H}_{(:,j)}^t)$ [45,56]. This auxiliary function is then bounded by another auxiliary function, $G(H_{(:,j)}, H_{(:,j)}^S)$, defined in (A.1). Therefore, the proof proceeds by giving conditions under which (GEM M-step) is guaranteed to reach a stationary point of $-\log p(H_{(:,j)}|X_{(:,j)})$ by repeated minimization of $Q(H_{(:,j)}, \bar{H}_{(:,j)}^t)$ and then finding conditions under which $Q(H_{(:,j)}, \bar{H}_{(:,j)}^t)$ can be minimized by minimization of $G(H_{(:,j)}, H_{(:,j)}^S)$ through the use of (19).

6.2. Analysis in S-NMF and S-NMF-W settings

We now extend the results of Section 6.1 to the case where W is unknown and is estimated using Algorithm 2. For clarity, let (z_w, τ_w) and (z_h, τ_h) refer to the distributional parameters of the priors over W and H , respectively. As before, $\tau_w, \tau_h > 0$ and $0 < z_w, z_h \leq 2$. First, it is confirmed that Algorithm 2 exhibits the same desirable optimization properties as the NMF MUR's (7)–(8).

Corollary 3.—Let $z_w, z_h \in \{1, 2\}$ and the functional dependence of $p^R(H_{(i,j)})$ on $H_{(i,j)}$ have a power function form. If performing S-NMF-W, let the functional dependence of $p^R(W_{(i,j)})$ on $W_{(i,j)}$ have a power function form. Consider using Algorithm 2 to generate $\{\bar{W}^t, \bar{H}^t\}_{t=0}^{\infty}$. Then, the update rules used in Algorithm 2 are well defined and $L^{NMF}(\bar{W}^{t+1}, \bar{H}^{t+1}) \leq L^{NMF}(\bar{W}^t, \bar{H}^t)$.

Proof.—The proof is shown in Appendix F.

Therefore, the proposed S-NMF framework maintains the monotonicity property of the original NMF MUR's, with the added benefit of promoting sparsity in H (and W , in the case of S-NMF-W). Unfortunately, it is not clear how to obtain a result like Theorem 4 for Algorithm 2 in the S-NMF setting. The reason that such a result cannot be shown is because it is not clear that if a limit point, $(\bar{W}^{\infty}, \bar{H}^{\infty})$, of Algorithm 2 exists, that this point is a stationary point of $L^{NMF}(\cdot, \cdot)$. Specifically, if there exists (i, j) such that $\bar{W}_{(i,j)}^{\infty} = 0$, the KKT condition $-(X - \bar{W}^{\infty} \bar{H}^{\infty})(\bar{H}^{\infty})^T \geq 0$ cannot be readily verified. This deficiency is unrelated to

the size of W and H and is, in fact, the reason that convergence guarantees for the original update rules in (7)–(8) do not exist. Interestingly, if Algorithm 2 is considered in S-NMF-W mode, this difficulty is alleviated.

Corollary 4.—Let $z_w, z_h \in \{1, 2\}$, $S = 1$, and the functional dependence of $p^R(H_{(i,j)})$ on $H_{(i,j)}$ and of $p^R(W_{(i,j)})$ on $W_{(i,j)}$ have power function forms. Then, the sequence $\{\bar{H}^t, \bar{W}^t\}_{t=1}^\infty$ produced by Algorithm 2 admits at least one limit point.

Proof.—The objective function is now coercive with respect to W and H as a result of the application of Theorem 3 to $-\log p^R(H_{(i,j)})$ and $-\log p^R(W_{(i,j)})$. Since $\{L^{NMF}(\bar{W}^t, \bar{H}^t)\}_{t=1}^\infty$ is non-increasing sequence, the proof for Corollary 2 in Appendix D can be applied to obtain the stated result.

Corollary 5.—Let $\{\bar{W}^t, \bar{H}^t\}_{t=1}^\infty$ be a sequence generated by Algorithm 2 with $\zeta_w = (19)$. Let $z_h, z_w \in \{1, 2\}$, the functional dependence of $p^R(H_{(i,j)})$ on $H_{(i,j)}$ have a power function form, the functional dependence of $p^R(W_{(i,j)})$ on $W_{(i,j)}$ have a power function form, the columns and rows of X have bounded norm, the columns of \bar{W}^∞ have bounded norm, the rows of \bar{H}^∞ have bounded norm, and \bar{W}^∞ and \bar{H}^∞ be full rank. Let one of the following conditions be satisfied: (1h) $z_h = 1$ and $\tau_h \leq \lambda / \max_{i,j} \left((\bar{W}^\infty)^T X \right)_{(i,j)}$ or (2h) $z_h = 2$ and $\tau_h \rightarrow 0$. In addition, let one of the following conditions be satisfied: (1w) $z_w = 1$ and $\tau_w \leq \lambda / \max_{i,j} \left(H^\infty X^T \right)_{(i,j)}$ or (2w) $z_w = 2$ and $\tau_w \rightarrow 0$. Then, $\{\bar{W}^t, \bar{H}^t\}_{t=1}^\infty$ is guaranteed to converge to set of stationary points of $L^{NMF}(\cdot, \cdot)$.

Proof.—The proof is provided in Appendix G.

7. Experimental results

In the following, experimental results for the class of proposed algorithms are presented. The experiments performed were designed to highlight the main properties of the proposed approaches. First, the accuracy of the proposed S-NNLS algorithms on synthetic data is studied. Then, experimental validation for claims made in Section 6 regarding the properties of the proposed approaches is provided. Finally, the proposed framework is shown in action on real-world data by learning a basis for a database of face images.

7.1. S-NNLS results on synthetic data

In order to compare the described methods, a sparse recovery experiment was undertaken. First, a dictionary $W \in \mathbb{R}_+^{100 \times n}$ is generated, where each element of W is drawn from the RG(0,1) distribution. The columns of W are then normalized to have unit ℓ_2 norm. The matrix $H \in \mathbb{R}_+^{n \times 100}$ is then generated by randomly selecting k coefficients of $H_{(:,j)}$ to be non-zero and drawing the non-zero values from a RG(0,1) distribution. The columns of H are normalized to have unit ℓ_2 norm. We then feed $X = WH$ and W to the S-NNLS algorithm and

approximate $H_{(:,j)}$ with $\hat{H}_{(:,j)}$. Note that this is a noiseless experiment. The distortion of the approximation is measured using the relative Frobenius norm error, $\|H - \hat{H}\|_F / \|H\|_F$. A total of 50 trials are run and averaged results are reported.

We use Algorithm 1 to generate recovery results for the proposed framework, with the number of inner-loop iterations, \mathcal{S} , of Algorithm 1 set to 2000 and the outer EM loop modified to run a maximum of 50 iterations. For reweighted ℓ_2 S-NNLS, the same annealing strategy for τ as reported in [49] is employed, where τ is initialized to 1 and decreased by a factor of 10 (up to a prespecified number of times) when the relative ℓ_2 difference between $\bar{H}_{(:,j)}^{t+1}$ and $\bar{H}_{(:,j)}^t$ is below $\sqrt{\tau}/100$ for each j . Note that this strategy does not influence the convergence properties described in Section 6 for the reweighted ℓ_2 approach since τ can be viewed as fixed after a certain number of iterations. For reweighted ℓ_1 S-NNLS, we use $\tau = 0.1$. The regularization parameter λ is selected using cross-validation by running the S-NNLS algorithms on data generated using the same procedure as the test data.

We compare our results with rsNNLS [10], the SUnSAL algorithm for solving (10) [57], the non-negative ISTA (NN-ISTA) algorithm¹ for solving (10) [58], NUIRLS, and ℓ_1 S-NNLS [12] (i.e. (11)). Since rsNNLS requires k as an input, we incorporate knowledge of k into the tested algorithms in order to have a fair comparison. This is done by first thresholding $\hat{H}_{(:,j)}$ by zeroing out all of the elements except the largest k and then executing (8) until convergence.

The S-NNLS results are shown in Fig. 2. Fig. 2a shows the recovery results for $n = 400$ as a function of the sparsity level k . All of the tested algorithms perform almost equally well up to $k = 30$, but the reweighted approaches dramatically outperform the competing methods for $k = 40$ and $k = 50$. Fig. 2b shows the recovery results for $k = 50$ as a function of n . All of the tested algorithms perform relatively well for $n = 200$, but the reweighted approaches separate themselves for $n = 400$ and $n = 800$. Fig. 2c and d show the average computational time for the algorithms tested as a function of sparsity level and dictionary size, respectively.

Two additional observations from the results in Fig. 2a can be made. First, the reweighted approaches perform slightly worse for sparsity levels $k \leq 20$. We believe that this is a result of suboptimal parameter selection for the reweighted algorithms and using a finer grid during cross-validation would improve the result. This claim is supported by the observation that NUIRLS performs at least as well or better than the reweighted approaches for $k \leq 20$ and, as argued in Section 4.1, NUIRLS is equivalent to reweighted ℓ_2 S-NNLS in the limit $\lambda, \tau \rightarrow 0$. The second observation is that the reweighted ℓ_2 approach consistently outperforms NUIRLS at high values of k . This suggests that the strategy of allowing $\lambda > 0$ and annealing τ , instead of setting it to 0 as in NUIRLS [34], is much more robust.

In addition to displaying superior S-NNLS performance, the proposed class of MUR's also exhibits fast convergence. Fig. 3 compares the evolution of the objective function $L(H)$ under the RGDP signal prior (i.e. the reweighted ℓ_1 formulation of Section 4.2) for

¹We modify the soft-thresholding operator to $S\beta(h) = \max(0, |h| - \beta)$.

Algorithm 1, with $S = 1$, with a baseline approach. The baseline employs the NN-ISTA algorithm to solve the reweighted ℓ_1 optimization problem which results from bounding the regularization term by a linear function of $H_{(i,j)}$ (similar to (23), but with $\|H/Q^t\|_F^2$ replaced by $\|H/Q^t\|_1$). The experimental results show that the MUR in (25) achieves much faster convergence as well as a lower objective function value compared to the baseline.

7.2. Block S-NNLS results on synthetic data

In this experiment, we first generate $W \in \mathbb{R}_+^{80 \times 160}$ by drawing its elements from a RG (0,1) distribution. We generate the columns of $H \in \mathbb{R}_+^{160 \times 100}$ by partitioning each column into blocks of size 8 and randomly selecting k blocks to be non-zero. The non-zero blocks are filled with elements drawn from a RG (0,1) distribution. We then attempt to recover H from $X = WH$. The relative Frobenius norm error is used as the distortion metric and results averaged over 50 trials are reported.

The results are shown in Fig. 4. We compare the greedy NN-BOMP algorithm with the reweighted approaches. The reweighted approaches consistently outperform the ℓ_1 based method, showing good recovery performance even when the number of non-zero elements of each column of H is equal to the dimensionality of the column.

7.3. A numerical study of the properties of the proposed methods

In this section, we seek to provide experimental verification for the claims made in the Section 6. First, the sparsity of the solutions obtained for the synthetic data experiments described in Section 7.1 is studied. Fig. 5 shows the magnitude of the n_0 th largest coefficient in $\hat{H}_{(:,j)}$ for various sizes of W , averaged over all 50 trials, all j , and all sparsity levels tested. The statement in Theorem 2 claims that the local minima of the objective function being optimized are sparse (i.e. that the number of nonzero entries is at most $d = 100$). In general, the proposed methods cannot be guaranteed to converge to a local minimum as opposed to a saddle point, so it cannot be expected that every solution produced by Algorithm 1 is sparse. Nevertheless, Fig. 5 shows that for $n = 200$ and $n = 400$, both reweighted approaches consistently find solutions with sparsity levels much smaller than 100. For $n = 800$, the reweighted ℓ_2 approach still finds solutions with sparsity smaller than 100, but the reweighted ℓ_1 method deviates slightly from the general trend.

Next, we test the claim made in Theorem 4 that the proposed approaches reach a stationary point of the objective function by monitoring the KKT residual norm of the scaled objective function. Note that, as in Appendix E, the $-\log u(H_{(i,j)})$ terms are omitted from $L(H)$ and the minimization of $L(H)$ is treated as a constrained optimization problem when deriving KKT conditions. For instance, for reweighted ℓ_1 S-NNLS, the KKT conditions can be stated as

$$\min \left(H, W^T W H - W^T X + \lambda \frac{\tau + 1}{\tau + H} \right) = 0 \quad (31)$$

and the norm of the left-hand side, averaged over all of the elements of H , can be viewed as a measure of how close a given H is to being stationary [27]. Table 3 shows the average KKT residual norm of the scaled objective function for the reweighted approaches for various problem sizes. The reported values are very small and provide experimental support for Theorem 4.

7.4. Learning a basis for face images

In this experiment, we use the proposed S-NMF and S-NMF-W frameworks to learn a basis for the CBCL face image dataset² [12,35]. Each dataset image is a 19×19 grayscale image. We used $n = 3d$ and learned W by running S-NMF with reweighted- ℓ_1 regularization on H and S-NMF-W with reweighted- ℓ_1 regularization on W and H . We used $\tau_w, \tau_h = 0.1$ and ran all algorithms to convergence. Due to a scaling indeterminacy, W is normalized to have unit ℓ_2 column norm at each iteration. A random subset of the learned basis vectors for each method with various levels of regularization is shown in Fig. 6. The results show the flexibility offered by the proposed framework. Fig. 6 a–c show that decreasing λ encourages S-NMF to learn high level features, whereas high values of λ force basis vectors to resemble images from the dataset. Fig. 6 d–f show a similar trend for S-NMF-W, although introducing a sparsity promoting prior on W tends to discourage basis vectors from resembling dataset images. It is difficult to verify Theorem 5 experimentally because W must be normalized at each iteration to prevent scaling instabilities and there is no guarantee that a given stationary point W^* has unit column norms. Nevertheless, the normalized KKT residual for the tested S-NMF-W algorithms with W normalization at each iteration on the CSBL face dataset is reported in Table 4.

7.5. Computational issues

One of the advantages of using the proposed MUR's is that inference can be performed on the entire matrix simultaneously in each block of the block-coordinate descent procedure with relatively simple matrix operations. In fact, the computational complexity of the MUR's in (21), (25), (28), and (29) is equivalent to that of the original NMF MUR given in (8) (which is $\mathcal{O}(nmr)$ where $r = \min(m, n)$ [35]). In other words, the proposed framework allows for performing S-NNLS and S-NMF without introducing computational complexity issues. Another benefit of this framework is that the operations required are simple matrix-based computations which lend themselves to a graphics processing unit (GPU) implementation. For example, a 9-fold speed-up is achieved in computing 500 iterations of (19) on a GPU compared to a CPU.

8. Conclusion

We presented a unified framework for S-NNLS and S-NMF algorithms. We introduced the RPESM as a sparsity promoting prior for non-negative data and provided details for a general class of S-NNLS algorithms arising from this prior. We showed that low-complexity MUR's can be used to carry out the inference, which are validated by a monotonicity guarantee. In addition, it was shown that the class of algorithms presented is guaranteed to converge to a set of stationary points, and that the local minima of the objective function are *sparse*. This framework was then extended to a block coordinate descent technique for S-

NMF and S-NMF-W. It was shown that the proposed class of S-NMF-W algorithms is guaranteed to converge to a set of stationary points.

Appendix A.: Proof of Theorem 1

Due to the assumption on the form of $p^R(H_{(i,j)})$, the functional dependence of $\langle (\gamma_{(i,j)}^{-z}) \rangle$, and hence $\Omega_{(i,j)}^t$, on $\bar{H}_{(i,j)}^t$ has the form $(\tau + (\bar{H}_{(i,j)}^t)^z)^{-1}$ up to a scaling constant, which is well-defined for all $\tau > 0$ and $\bar{H}_{(i,j)}^t \in [0, \infty)$. As a result, (19) is well defined for all (i, j) such that $H_{(i,j)}^S > 0$.

To show that $Q(H, \bar{H}^t)$ is non-increasing under MUR (19), a proof which follows closely to Hoyer and co-workers [11,17] is presented. We omit the $-\log u(H_{(i,j)})$ term in $Q(H, \bar{H}^t)$ in our analysis because it has no contribution to $Q(H, \bar{H}^t)$ if $H = 0$ and the update rules are guaranteed to keep $H_{(i,j)}$ non-negative.

First, note that $Q(H, \bar{H}^t)$ is separable in the columns of H , $H_{(:,j)}$, so we focus on minimizing $Q(H, \bar{H}^t)$ for each $H_{(:,j)}$ separately. For the purposes of this proof, let h and x represent columns of H and X , respectively, and let $Q(h)$ denote the dependence of $Q(H, \bar{H}^t)$ on one of the columns of H , with the dependency on \bar{H}^t being implicit. Then,

$$Q(h) = \left\| x - Wh \right\|_2^2 + \lambda \sum_i q_i (h_i)^z$$

where q represents the non-negative weights in (17). Let $G(h, h^s)$ be

$$G(h, h^s) = Q(h^s) + (h - h^s)^T \nabla Q(h^s) + \frac{(h - h^s)^T K(h^s)(h - h^s)}{2} \quad (\text{A.1})$$

where $K(h^s) = \text{diag}((W^T W h^s + \lambda z q \odot (h^s)^{z-1})/h^s)$. For reference,

$$\nabla Q(h^s) = W^T W h^s - W^T x + \lambda z q \odot (h^s)^{z-1} \quad (\text{A.2})$$

$$\nabla^2 Q(h^s) = W^T W + \lambda z(z-1) \text{diag}(q \odot (h^s)^{z-2}). \quad (\text{A.3})$$

It will now be shown that $G(h, h^s)$ is an auxiliary function for $Q(h)$. Trivially, $G(h, h) = Q(h)$. To show that $G(h, h^s)$ is an upper-bound for $Q(h)$, we begin by using the fact that $Q(h)$ is a polynomial of order 2 to rewrite $Q(h)$ as $Q(h) = Q(h^s) + (h - h^s)^T \nabla Q(h^s) + 0.5(h - h^s)^T \nabla^2 Q(h^s)(h - h^s)$. It then follows that $G(h, h^s)$ is an auxiliary function for $Q(h)$ if and

only if the matrix $M = K(h^s) - \nabla^2 Q(h^s)$ is positive semi-definite (PSD). The matrix M can be decomposed as $M = M_1 + M_2$, where $M_1 = \text{diag}((W^T W h^s)/h^s) - W^T W$ and $M_2 = \lambda z(2 - z) \text{diag}(q \odot (h^s)^{z-2})$. The matrix M_1 was shown to be PSD in [17]. The matrix M_2 is a diagonal matrix with the (i, i) th entry being $\lambda z(2 - z)q_i(h_i^s)^{z-2}$. Since $q_i(h_i^s)^{z-2} \geq 0$ and $z \geq 2$, M_2 has non-negative entries on its diagonal and, consequently, is PSD. Since the sum of PSD matrices is PSD, it follows that M is PSD and $G(h, h^s)$ is an auxiliary function for $Q(h)$. Since $G(h, h^s)$ as an auxiliary function for $Q(h)$, $Q(h)$ is non-increasing under the update rule [17]

$$h^{s+1} = \arg \min_h G(h, h^s). \quad (\text{A.4})$$

The optimization problem in (A.4) can be solved in closed form, leading to the MUR shown in (19). The multiplicative nature of the update rule in (19) guarantees that the sequence $\{H^s\}_{s=1}^\infty$ is non-negative.

Appendix B.: Proof of Theorem 2

This proof is an extension of (Theorem 1 [59] and Theorem 8 [60]). Since $L(H)$ is separable in the columns of H , consider the dependence of $L(H)$ on a single column of H , denoted by $L(h)$. The function $L(h)$ can be written as

$$\|x - Wh\|_2^2 - \sum_{i=1}^n 2\sigma^2 \log p(h_i). \quad (\text{B.1})$$

Let h^* be a local minimum of $L(h)$. We observe that h^* must be non-negative. Note that $-\log p(h_i) \rightarrow \infty$ when $h_i < 0$ since $p(h_i) = 0$ over the negative orthant. As such, if one of the elements of h^* is negative, h^* must be a global maximum of $L(h)$. Using the assumption on the form of $p^R(h_i)$, (B.1) becomes

$$\|x - Wh\|_2^2 + \sum_{i=1}^n 2\sigma^2 (\alpha \log(\tau + (h_i)^z) - \log u(h_i)) + c \quad (\text{B.2})$$

where constants which do not depend on h are denoted by c . By the preceding argument, $\log u(h_i^*) = 0$, so the $\log u(h_i^*)$ term makes no contribution to $L(h^*)$. The vector h^* must be a local minimum of the constrained optimization problem

$$\min_{x = Wh + v^*} \underbrace{\sum_{i=1}^n \log(\tau + (h_i)^z)}_{\phi(h)} \quad (\text{B.3})$$

where $v^* = x - Wh^*$ and $\phi(\cdot)$ is the diversity measure induced by the prior on H . It can be shown that $\phi(\cdot)$ is concave under the conditions of Theorem 2. Therefore, under the conditions of Theorem 2, the optimization problem (B.3) satisfies the conditions of (Theorem 8 [60]). It then follows that the local minima of (B.3) are basic feasible solutions, i.e. they satisfy $x = Wh + v^*$ and $\|h\|_0 = d$. Since h^* is one of the local minima of (B.3), $\|h^*\|_0 = d$.

Appendix C.: Proof of Theorem 3

It is sufficient to show that $\lim_{H_{(i,j)} \rightarrow \infty} p(H_{(i,j)}) = 0$. Consider the form of $p(H_{(i,j)})$ when it is a member of the RPESM family:

$$p(H_{(i,j)}) = \int_0^\infty p(H_{(i,j)} | \gamma_{(i,j)}) p(\gamma_{(i,j)}) d\gamma_{(i,j)} \quad \text{C.1}$$

where $H_{(i,j)} | \gamma_{(i,j)} \sim p^{RPE}(H_{(i,j)} | \gamma_{(i,j)}; z)$. Note that

$$\left| p^{RPE}(H_{(i,j)} | \gamma_{(i,j)}) p(\gamma_{(i,j)}) \right| \leq \left| p^{RPE}(0 | \gamma_{(i,j)}; z) p(\gamma_{(i,j)}) \right|.$$

Coupled with the fact that $p(H_{(i,j)} | \gamma_{(i,j)})$ is continuous over the positive orthant, the dominated convergence theorem can be applied to switch the limit with the integral in ((C.1):

$$\lim_{H_{(i,j)} \rightarrow \infty} \int_0^\infty p(H_{(i,j)} | \gamma_{(i,j)}) p(\gamma_{(i,j)}) d\gamma_{(i,j)} = \int_0^\infty \lim_{H_{(i,j)} \rightarrow \infty} p(H_{(i,j)} | \gamma_{(i,j)}) p(\gamma_{(i,j)}) d\gamma_{(i,j)} = 0.$$

Appendix D.: Proof of Corollary 2

This proof follows closely to the first part of the proof of (Theorem 1, [36]). Let $\mathcal{S}_0 = \{H \in \mathbb{R}_+^{n \times m} | L(H) \leq L(\bar{H}^0)\}$ Lemma 1 established that $L(H)$ is coercive. In addition, $L(H)$ is a continuous function of H over the positive orthant. Therefore, \mathcal{S}_0 is a compact set (Theorem 1.2, [61]). The sequence $\{L(\bar{H}^t)\}_{t=1}^\infty$ is non-increasing as a result of Theorem 1, such that $\{\bar{H}^t\}_{t=1}^\infty \in \mathcal{S}_0$. Since \mathcal{S}_0 is compact, $\{\bar{H}^t\}_{t=1}^\infty$ admits at least one limit point.

Appendix E.: Proof of Theorem 4

From Lemma 2, the sequence $\{\bar{H}^t\}_{t=1}^\infty$ admits at least is limit one point. What remains is to show that every limit point is a stationary point of (16). The sufficient conditions for the limit points to be are tationary are (Theorem 1, [62])

1. $Q(H, \bar{H}^t)$ is continuous in both H and \bar{H}^t ,
2. At each iteration t , one of the following is true

$$Q(H_{(:,j)}^{t+1}, \bar{H}_{(:,j)}^t) < Q(\bar{H}_{(:,j)}^t, \bar{H}_{(:,j)}^t) \quad (\text{E.1})$$

$$\bar{H}_{(:,j)}^{t+1} = \arg \min_{H_{(:,j)} \geq 0} Q(H_{(:,j)}, \bar{H}_{(:,j)}^t). \quad (\text{E.2})$$

The function³ $Q(H, \bar{H}^t)$ is continuous in H , trivially, and in \bar{H}^t if the functional dependence of $p^R(\bar{H}_{(i,j)}^t)$ on $\bar{H}_{(i,j)}^t$ has the form (30).

In order to show that the descent condition is satisfied, we begin by noting that $Q(H_{(:,j)}, \bar{H}_{(:,j)}^t)$ is strictly convex with respect to $H_{(:,j)}$ if the conditions of Theorem 4 are satisfied. This can be seen by examining the expression for the Hessian of $Q(H_{(:,j)}, \bar{H}_{(:,j)}^t)$ in (A.3). If W is full rank, then $W^T W$ is positive definite. In addition, $\lambda z(z-1) \text{diag}(\Omega_{(:,j)}^t \odot (H_{(:,j)}^S)^{z-2})$ is PSD because $z \geq 1$. Therefore, the Hessian of $Q(H_{(:,j)}, \bar{H}_{(:,j)}^t)$ is positive definite if the conditions of Theorem 4 are satisfied.

Since $S = 1 - \bar{H}_{(:,j)}^{t+1}$ is generated by (19) with H^F replaced by \bar{H}^t . This update has two possibilities: (1) $\bar{H}_{(:,j)}^{t+1} \neq \bar{H}_{(:,j)}^t$ or (2) $\bar{H}_{(:,j)}^{t+1} = \bar{H}_{(:,j)}^t$. If condition (1) is true, then (E.1) is satisfied because of the strict convexity of $Q(H_{(:,j)}, \bar{H}_{(:,j)}^t)$ and the monotonicity guarantee of Theorem 1.

It will now be shown that if condition (2) is true, then $\bar{H}_{(:,j)}^{t+1}$ must satisfy (E.2). Since $Q(H_{(:,j)}, \bar{H}_{(:,j)}^t)$ is convex, any $\bar{H}_{(:,j)}^{t+1}$ which satisfies the Karush-Kuhn-Tucker (KKT) conditions associated with (E.2) must be a solution to (E.2) [56]. The KKT conditions associated with (E.2) are given by [35]:

$$H_{(i,j)} \geq 0 \quad (\text{E.3})$$

³As in the proof of Theorem (1), we omit the $-\log u(H(i,j))$ term from $Q(H_{(:,j)}, \bar{H}_{(:,j)}^t)$ and explicitly enforce the non-negativity constraint on $H_{(:,j)}$.

$$\left(\nabla Q(H_{(:,j)}, \bar{H}_{(:,j)}^t) \right)_i \geq 0 \quad (\text{E.4})$$

$$H_{(i,j)} \left(\nabla Q(H_{(:,j)}, \bar{H}_{(:,j)}^t) \right)_i = 0 \quad (\text{E.5})$$

for all i . The expression for $\nabla Q(H_{(:,j)}, \bar{H}_{(:,j)}^t)$ is given in (A.2). For any i such that $\bar{H}_{(i,j)}^{t+1} > 0$, $(W^T X)_{(i,j)} = (W^T W \bar{H}^{t+1})_{(i,j)} + \lambda \Omega_{(i,j)}^t (\bar{H}_{(i,j)}^{t+1})^{z-1}$ because H^{t+1} was generated by (19).

This implies that

$$\left(\nabla Q(H_{(:,j)}, \bar{H}_{(:,j)}^t) \right)_{H_{(:,j)} = \bar{H}_{(:,j)}^{t+1}} = 0$$

for all i such that $\bar{H}_{(i,j)}^{t+1} > 0$. Therefore, all of the KKT conditions are satisfied.

For any i such that $\bar{H}_{(i,j)}^{t+1} > 0$ (E.3) and (E.5) are trivially satisfied. To see that (E.4) is satisfied, first consider the scenario where $z = 1$. In this case,

$$\begin{aligned} \lim_{\bar{H}_{(i,j)}^{t+1} \rightarrow 0} \left(\nabla Q(H_{(:,j)}, \bar{H}_{(:,j)}^t) \right)_{H_{(:,j)} = \bar{H}_{(:,j)}^{t+1}} &= \lim_{\bar{H}_{(i,j)}^{t+1} \rightarrow 0} (W^T W \bar{H}^{t+1})_{(i,j)} + \frac{\lambda (\bar{H}_{(i,j)}^{t+1})^0}{\tau + (\bar{H}_{(i,j)}^{t+1})^1} \\ - (W^T X)_{(i,j)} &= c + \frac{\lambda}{\tau} - (W^T X)_{(i,j)} \geq 2 \cdot 0 \end{aligned}$$

where $c > 0$, (1) follows from the assumption on $p^R(H_{(i,j)})$ having a power exponential form, and (2) follows from the assumptions that the elements of $W^T X$ are bounded and $\tau = \lambda / \max_{i,j} (W^T X)_{(i,j)}$. When $z = 2$,

$$\begin{aligned} \lim_{\bar{H}_{(i,j)}^{t+1} \rightarrow 0} \lim_{\tau \rightarrow 0} \left(\nabla Q(H_{(:,j)}, \bar{H}_{(:,j)}^t) \right)_{H_{(:,j)} = \bar{H}_{(:,j)}^{t+1}} &= \lim_{\bar{H}_{(i,j)}^{t+1} \rightarrow 0} (W^T W \bar{H}^{t+1})_{(i,j)} + \frac{2\lambda}{\bar{H}_{(i,j)}^{t+1}} \\ - (W^T X)_{(i,j)} &\geq 2 \cdot 0 \end{aligned}$$

where (1) follows from the assumption on $p^R(H_{(i,j)})$ having a power exponential form and (2) follows from the assumption that the elements of $W^T X$ are bounded. Therefore, (E.4) is satisfied for all i such that $\bar{H}_{(i,j)}^{t+1} = 0$. To conclude, if $\bar{H}_{(:,j)}^{t+1}$ satisfies $\bar{H}_{(:,j)}^{t+1} = \bar{H}_{(:,j)}^t$, then it satisfies the KKT conditions and must be solution of (E.2).

Appendix F.: Proof of Corollary 3

In the S-NMF setting ($\zeta_w = (8)$, $\zeta_h = (19)$), this result follows from the application of (Theorem 1 [17]) to the W update stage of Algorithm 2 and the application of Theorem 1 to the H update stage of Algorithm 2. In the S-NMF-W setting ($\zeta_w = (19)$, $\zeta_h = (19)$), the result follows from the application of Theorem 1 to each step of Algorithm 2. In both cases,

$$L^{NMF}(\bar{w}^t, \bar{h}^t) \geq L^{NMF}(\bar{w}^{t+1}, \bar{h}^t) \geq L^{NMF}(\bar{w}^{t+1}, \bar{h}^{t+1}).$$

Appendix G.: Proof of Corollary 5

The existence of a limit point $(\bar{w}^\infty, \bar{h}^\infty)$ is guaranteed by Corollary 4. It is sufficient to show that $L^{NMF}(\cdot, \cdot)$ is stationary with respect to \bar{w}^∞ and \bar{h}^∞ individually. The result follows by application of Theorem 4 to \bar{w}^∞ and \bar{h}^∞ .

References

- [1]. Pauca VP, Shahnaz F, Berry MW, Plemmons RJ, Text mining using non-negative matrix factorizations, in: SDM, 4, 2004, pp. 452–456.
- [2]. Monga V, Mihçak MK, Robust and secure image hashing via non-negative matrix factorizations, Inf. Forensics Secur. IEEE Trans. 2 (3) (2007) 376–390.
- [3]. Loizou PC, Speech enhancement based on perceptually motivated bayesian estimators of the magnitude spectrum, Speech Audio Process. IEEE Trans. 13 (5) (2005) 857–869.
- [4]. Févotte C, Bertin N, Durrieu J-L, Nonnegative matrix factorization with the Itakura–Saito divergence: with application to music analysis, Neural Comput. 21 (3) (2009) 793–830. [PubMed: 18785855]
- [5]. Sajda P, Du S, Brown TR, Stoyanova R, Shungu DC, Mao X, Parra LC, Non-negative matrix factorization for rapid recovery of constituent spectra in magnetic resonance chemical shift imaging of the brain, Med. Imaging IEEE Trans. 23 (12) (2004) 1453–1465.
- [6]. Lawson CL, Hanson RJ, Solving Least Squares Problems, 161, SIAM, 1974.
- [7]. Bro R, De Jong S, A fast non-negativity-constrained least squares algorithm, J. Chemom 11 (5) (1997) 393–401.
- [8]. Elad M, Sparse and Redundant Representations, Springer New York, 2010.
- [9]. Eldar YC, Kutyniok G, Compressed Sensing: Theory and Applications, Cambridge University Press, 2012.
- [10]. Peharz R, Pernkopf F, Sparse nonnegative matrix factorization with l0 -constraints, Neurocomputing 80 (2012) 38–46. [PubMed: 22505792]
- [11]. Hoyer PO, Non-negative sparse coding, in: Neural Networks for Signal Processing, 2002. Proceedings of the 2002 12th IEEE Workshop on, IEEE, 2002, pp. 557–565.
- [12]. Hoyer PO, Non-negative matrix factorization with sparseness constraints, J. Mach. Learn. Res 5 (2004) 1457–1469.
- [13]. Mairal J, Ponce J, Sapiro G, Zisserman A, Bach FR, Supervised dictionary learning, in: Advances in Neural Information Processing Systems, 2009, pp. 1033–1040.
- [14]. Tošić I, Frossard P, Dictionary learning, Signal Process. Mag. IEEE 28 (2) (2011) 27–38.
- [15]. Gangeh MJ, Farahat AK, Ghodsi A, Kamel MS, Supervised dictionary learning and sparse representation-a review, arXiv: 1502.05928 (2015).
- [16]. Kreutz-Delgado K, Murray JF, Rao BD, Engan K, Lee T-W, Sejnowski TJ, Dictionary learning algorithms for sparse representation, Neural Comput. 15 (2) (2003) 349–396. [PubMed: 12590811]

- [17]. Lee DD, Seung HS, Algorithms for non-negative matrix factorization, in: *Advances in Neural Information Processing Systems*, 2001, pp. 556–562.
- [18]. Aharon M, Elad M, Bruckstein AM, K-svd and its non-negative variant for dictionary design, *Optics & Photonics 2005*, International Society for Optics and Photonics, 2005 591411–591411
- [19]. Lee DD, Seung HS, Learning the parts of objects by non-negative matrix factorization, *Nature* 401 (6755) (1999) 788–791. [PubMed: 10548103]
- [20]. Lin C.-b., Projected gradient methods for nonnegative matrix factorization, *Neural Comput.* 19 (10) (2007) 2756–2779. [PubMed: 17716011]
- [21]. Bertsekas DP, *Nonlinear programming*, 1999.
- [22]. Zhou G, Cichocki A, Xie S, Fast nonnegative matrix/tensor factorization based on low-rank approximation, *IEEE Trans. Signal Process.* 60 (6) (2012) 2928–2940.
- [23]. Zhou G, Cichocki A, Zhao Q, Xie S, Nonnegative matrix and tensor factorizations: an algorithmic perspective, *IEEE Signal Process. Mag* 31 (3) (2014) 54–65.
- [24]. Kim M, Smaragdis P, Mixtures of local dictionaries for unsupervised speech enhancement, *Signal Process. Lett. IEEE* 22 (3) (2015) 293–297.
- [25]. Joder C, Weninger F, Eyben F, Virette D, Schuller B, Real-time speech separation by semi-supervised nonnegative matrix factorization, in: *Latent Variable Analysis and Signal Separation*, Springer, 2012, pp. 322–329.
- [26]. Raj B, Virtanen T, Chaudhuri S, Singh R, Non-negative matrix factorization based compensation of music for automatic speech recognition., in: *Inter-speech*, Citeseer, 2010, pp. 717–720.
- [27]. Gonzalez EF, Zhang Y, Accelerating the lee-seung algorithm for non-negative matrix factorization, Tech. Rep. TR-05–02, Dept. Comput. & Appl. Math, Rice Univ., Houston, TX, 2005.
- [28]. Donoho DL, Elad M, Temlyakov VN, Stable recovery of sparse overcomplete representations in the presence of noise, *Inf. Theory IEEE Trans.* 52 (1) (2006) 6–18.
- [29]. Gillis N, Sparse and unique nonnegative matrix factorization through data pre-processing, *J. Mach. Learn. Res* 13 (1) (2012) 3349–3386.
- [30]. Tipping ME, Sparse bayesian learning and the relevance vector machine, *J. Mach. Learn. Res* 1 (2001) 211–244.
- [31]. Wipf DP, Rao BD, Sparse bayesian learning for basis selection, *Signal Process. IEEE Trans.* 52 (8) (2004) 2153–2164.
- [32]. Wipf D, Nagarajan S, Iterative reweighted and methods for finding sparse solutions, *IEEE J. Sel. Top. Signal Process.* 4 (2) (2010) 317–329.
- [33]. Lefevre A, Bach F, Févotte C, Itakura-saito nonnegative matrix factorization with group sparsity, in: *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, IEEE, 2011, pp. 21–24.
- [34]. Grady PD, Rickard ST, Compressive sampling of non-negative signals, in: *Machine Learning for Signal Processing, 2008. MLSP 2008. IEEE Workshop on*, IEEE, 2008, pp. 133–138.
- [35]. Lin C-J, On the convergence of multiplicative update algorithms for nonnegative matrix factorization, *IEEE Trans. Neural Netw.* 18 (6) (2007) 1589–1596.
- [36]. Zhao R, Tan VY, A unified convergence analysis of the multiplicative update algorithm for nonnegative matrix factorization, arXiv: 1609.00951 (2016).
- [37]. Andrews DF, Mallows CL, Scale mixtures of normal distributions, *J. R. Stat. Soc. Ser. B (Methodological)* (1974) 99–102.
- [38]. Nalci A, Fedorov I, Rao BD, Rectified gaussian scale mixtures and the sparse non-negative least squares problem, arXiv: 1601.06207 (2016).
- [39]. Giri R, Rao B, Type I and type II bayesian methods for sparse signal recovery using scale mixtures, *IEEE Trans. Signal Process.* 64 (13) (2016) 3418–3428, doi: 10.1109/TSP.2016.2546231.
- [40]. Giri R, Rao B, Learning distributional parameters for adaptive bayesian sparse signal recovery, *IEEE Comput. Intell. Mag. Spec. Issue Model Complexity Regularization Sparsity* (2016).

- [41]. Palmer JA, Variational and scale mixture representations of non-gaussian densities for estimation in the bayesian linear model: sparse coding, independent component analysis, and minimum entropy segmentation, 2006 PhD Thesis.
- [42]. Palmer J, Kreutz-Delgado K, Rao BD, Wipf DP, Variational em algorithms for non-gaussian latent variable models, in: Advances in Neural Information Processing Systems, 2005, pp. 1059–1066.
- [43]. Lange K, Sinsheimer JS, Normal/independent distributions and their applications in robust regression, *J. Comput. Graphical Stat.* 2 (2) (1993) 175–198.
- [44]. Dempster AP, Laird NM, Rubin DB, Iteratively reweighted least squares for linear regression when errors are normal/independent distributed, *Multivariate Anal.* V (1980).
- [45]. Dempster AP, Laird NM, Rubin DB, Maximum likelihood from incomplete data via the em algorithm, *J. R. Stat. Soc. Ser. B (methodological)* (1977) 1–38.
- [46]. Wipf DP, Rao BD, An empirical bayesian strategy for solving the simultaneous sparse approximation problem, *Signal Process. IEEE Trans.* 55 (7) (2007) 3704–3716.
- [47]. Schachtner R, Poeppel G, Tomé A, Lang E, A bayesian approach to the lee–seung update rules for nmf, *Pattern Recognit. Lett* 45 (2014) 251–256.
- [48]. Themelis KE, Rontogiannis AA, Koutroumbas KD, A novel hierarchical bayesian approach for sparse semisupervised hyperspectral unmixing, *IEEE Trans. Signal Process.* 60 (2) (2012) 585–599.
- [49]. Chartrand R, Yin W, Iteratively reweighted algorithms for compressive sensing, in: Acoustics, Speech and Signal Processing, 2008 ICASSP 2008. IEEE International Conference on, IEEE, 2008, pp. 3869–3872.
- [50]. Chen F, Zhang Y, Sparse hyperspectral unmixing based on constrained lp-1 2 optimization, *IEEE Geosci. Remote Sens. Lett* 10 (5) (2013) 1142–1146.
- [51]. Candes EJ, Wakin MB, Boyd SP, Enhancing sparsity by reweighted l_1 minimization, *J. Fourier Anal. Appl.* 14 (5–6) (2008) 877–905.
- [52]. Sun DL, Mysore GJ, Universal speech models for speaker independent single channel source separation, in: Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on, IEEE, 2013, pp. 141–145.
- [53]. Dong W, Li X, Zhang L, Shi G, Sparsity-based image denoising via dictionary learning and structural clustering, in: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, IEEE, 2011, pp. 457–464.
- [54]. Jiang S, Gu Y, Block-sparsity-induced adaptive filter for multi-clustering system identification, arXiv: 1410.5024 (2014).
- [55]. Eldar YC, Kuppinger P, Bölcskei H, Block-sparse signals: uncertainty relations and efficient recovery, *Signal Process. IEEE Trans.* 58 (6) (2010) 3042–3054.
- [56]. Boyd S, Vandenberghe L, *Convex Optimization*, Cambridge university press, 2004.
- [57]. Bioucas-Dias JM, Figueiredo MA, Alternating direction algorithms for constrained sparse regression: application to hyperspectral unmixing, in: Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), 2010 2nd Workshop on, IEEE, 2010, pp. 1–4.
- [58]. Daubechies I, Defrise M, De Mol C, An iterative thresholding algorithm for linear inverse problems with a sparsity constraint, *Commun. Pure Appl. Math* 57 (11) (2004) 1413–1457.
- [59]. Rao BD, Egan K, Cotter SF, Palmer J, Kreutz-Delgado K, Subset selection in noise based on diversity measure minimization, *IEEE Trans. Signal Process.* 51 (3) (2003) 760–770, doi: 10.1109/TSP.2002.808076.
- [60]. Kreutz-Delgado K, Rao BD, A general approach to sparse basis selection: majorization, concavity, and affine scaling, Tech. Rep. UCSD-CIE-97-7-1, University of California, San Diego, 1997.
- [61]. Burke JV, Undergraduate nonlinear continuous optimization.
- [62]. Wu CJ, On the convergence properties of the em algorithm, *Ann. Stat* (1983) 95–103.

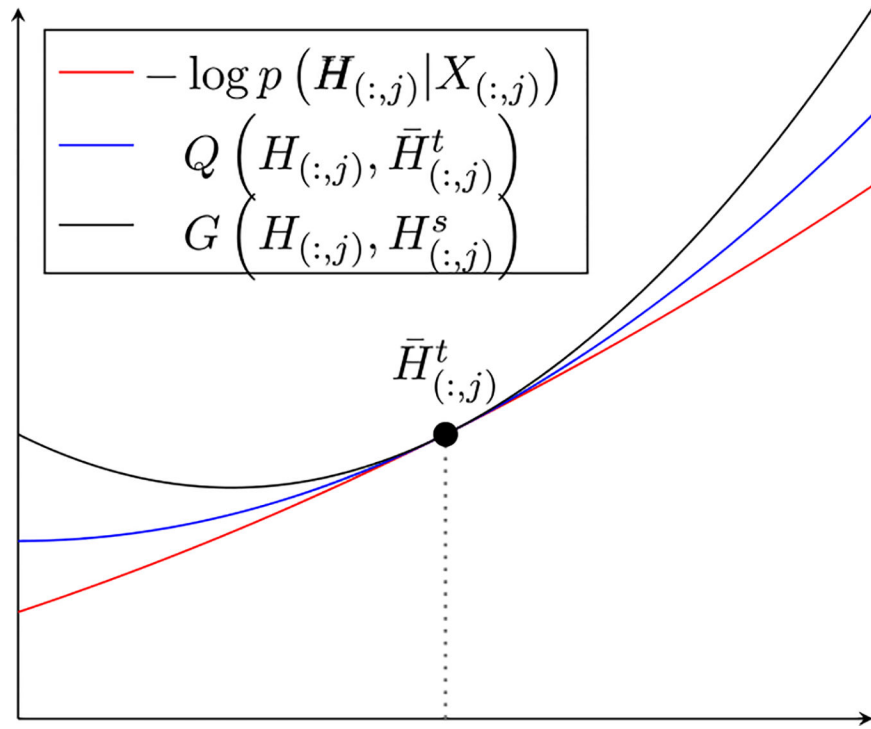
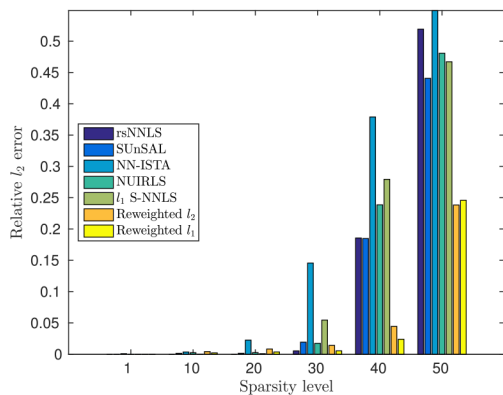
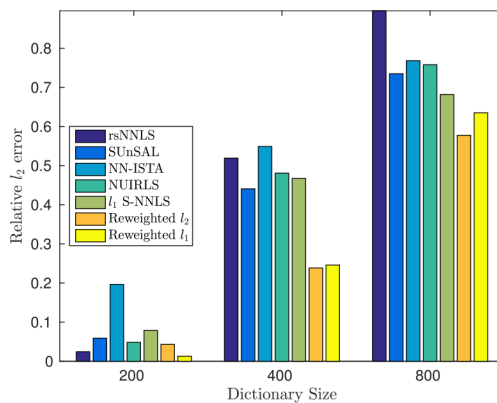


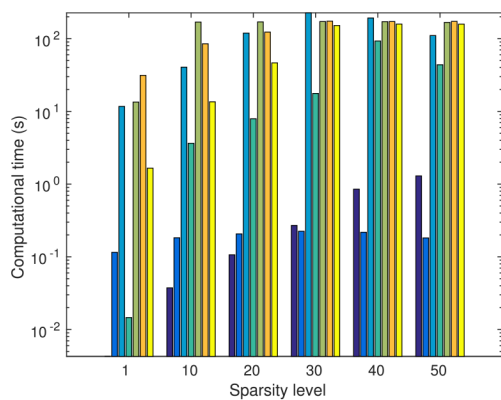
Fig. 1.
Visualization of Algorithm 1.



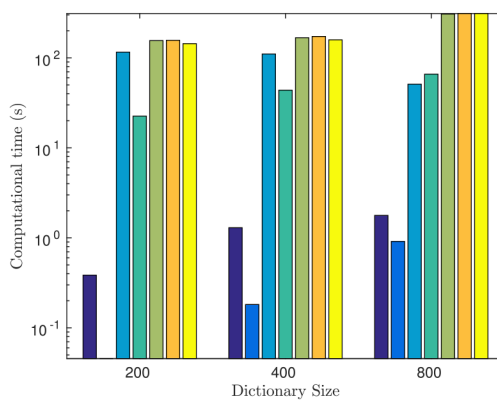
(a) Average relative ℓ_2 error as a function of sparsity level for $n = 400$.



(b) Average relative ℓ_2 error as a function of n for sparsity level 50.



(c) Average computational time as a function of sparsity level for $n = 400$.



(d) Average computational time as a function of n for sparsity level 50.

Fig. 2. S-NNLS results on synthetic data. The legends for (c) and (d) have been omitted, but are identical to the legends in (a) and (b).

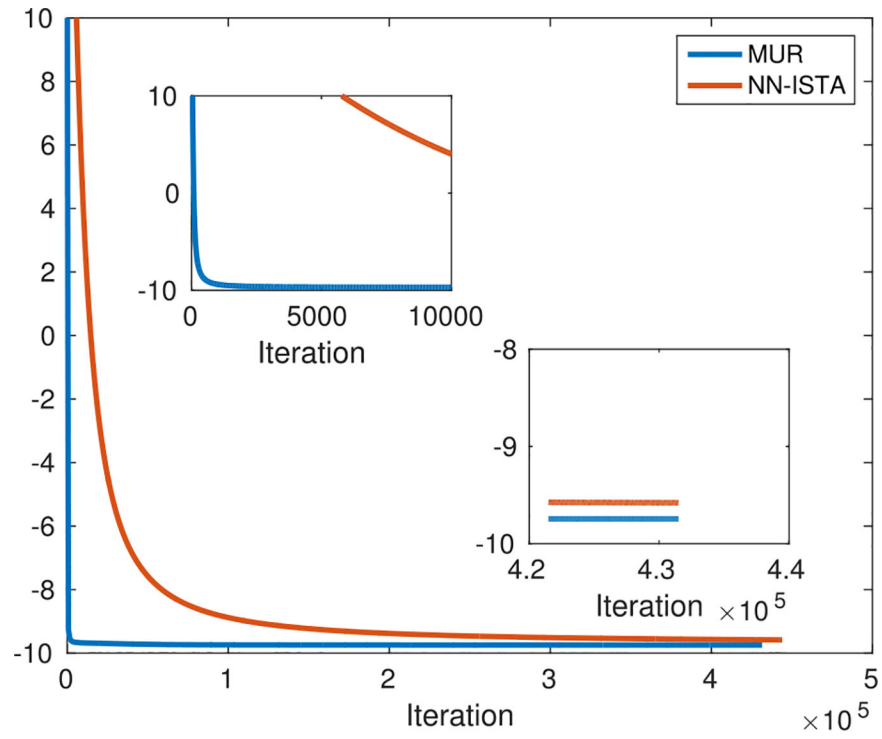


Fig. 3. Evolution of $L(H)$ for the reweighted l_1 formulation in Section 4.2 using Algorithm 1 and a baseline approach employing the NN-ISTA algorithm.

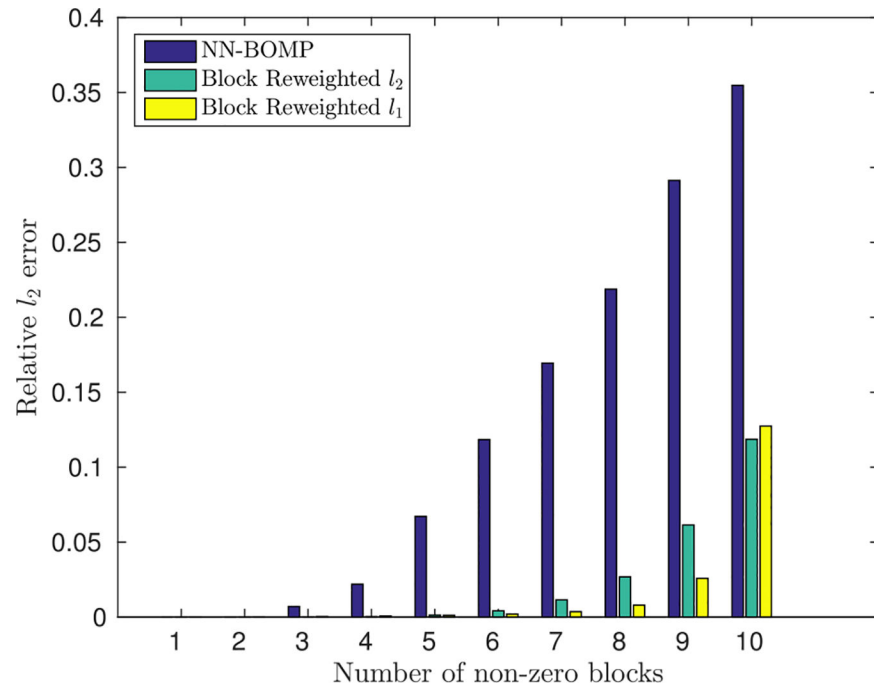
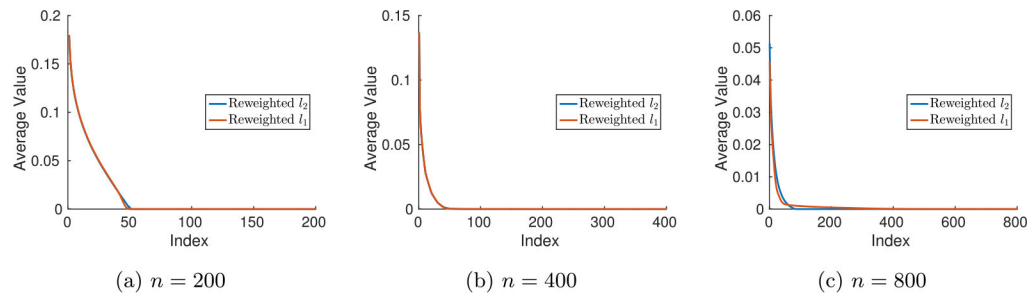


Fig. 4.
Block sparse recovery results.

**Fig. 5.**

Average sorted coefficient value for S-NNLS with $W \in \mathbb{R}_+^{100 \times n}$. The value at index n_0 represents the average value of the n_0 th largest coefficient the estimated \hat{H} .

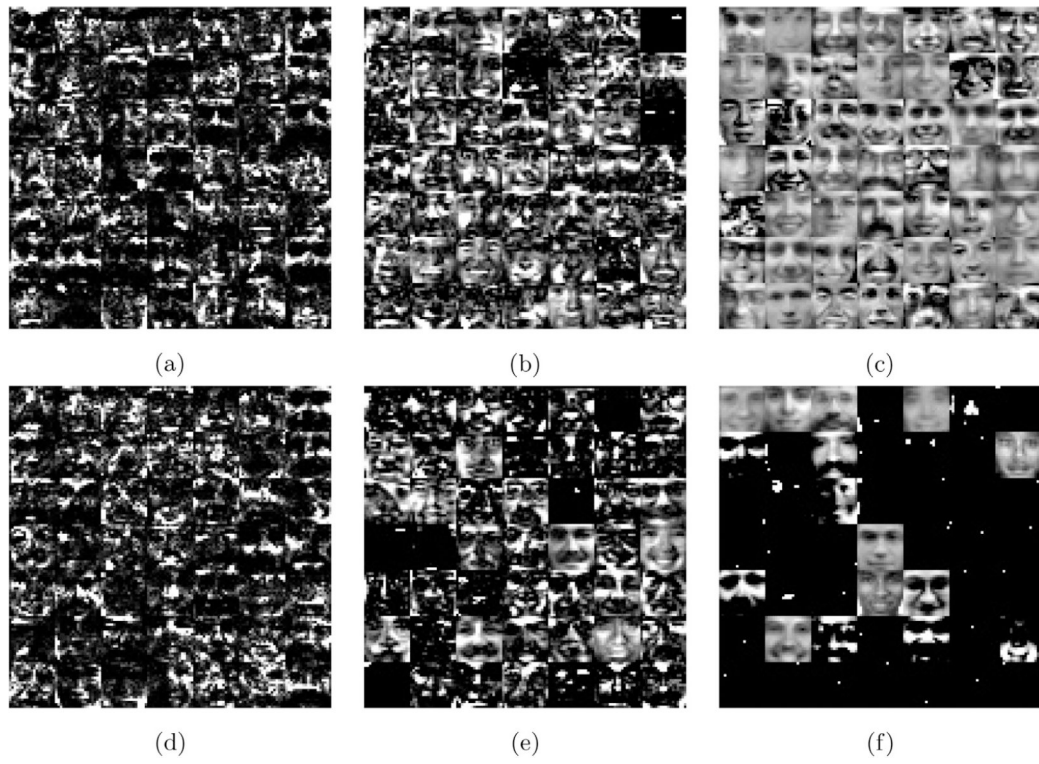


Fig. 6. Visualization of random subset of learned atoms of W for CBCL dataset. 6 a–c: S-NMF with reweighted ℓ_1 regularization on H , $\lambda = 1e-3, 1e-2, 1e-1$, respectively. 6 d–6 f: S-NMF-W with reweighted ℓ_1 regularization on H and W , $\lambda = 1e-3, 1e-2, 1e-1$, respectively.

Table 1

Distributions used throughout this work, where $\exp(a) = e^a$.

| Distribution | pdf |
|-------------------------------------|---|
| Rectified Gaussian | $p^{\text{RG}}(h; \gamma) = \sqrt{\frac{2}{\pi\gamma}} \exp\left(-\frac{h^2}{2\gamma}\right) u(h)$ |
| Exponential | $p^{\text{Exp}}(h; \gamma) = \gamma \exp(-\gamma h) u(h)$ |
| Inverse Gamma | $p^{\text{IGa}}(h; a, b) = \frac{b^a}{\Gamma(a)} h^{-a-1} \exp\left(-\frac{b}{h}\right) u(h)$ |
| Gamma | $p^{\text{Ga}}(h; a, b) = \frac{1}{\Gamma(a)b^a} h^{a-1} \exp(-hb) u(h)$ |
| Rectified Student's-t | $p^{\text{RST}}(h; \tau) = \frac{2\Gamma\left(\frac{\tau+1}{2}\right)}{\sqrt{\pi\tau}\Gamma\left(\frac{\tau}{2}\right)} \left(1 + \frac{h^2}{\tau}\right)^{-\frac{(\tau+1)}{2}} u(h)$ |
| Rectified Generalized Double Pareto | $p^{\text{RGDP}}(h; a, b, \tau) = 2\eta \left(1 + \frac{h^b}{\tau d}\right)^{-\left(\tau + \frac{1}{b}\right)} u(h)$ |

Table 2

RPESM representation of rectified sparse priors.

| z | $P(\boldsymbol{\gamma}_{(i,j)})$ | $P(\mathbf{H}_{(i,j)})$ |
|-----|--|--|
| 2 | $p^{\text{EXP}}(\boldsymbol{\gamma}_{(i,j)}; \boldsymbol{\tau}^2 / 2)$ | $p^{\text{EXP}}(\mathbf{H}_{(i,j)}; \boldsymbol{\tau})$ |
| 2 | $p^{\text{EXP}}(\boldsymbol{\gamma}_{(i,j)}; \boldsymbol{\tau}^2 / 2)$ | $p^{\text{RST}}(\mathbf{H}_{(i,j)}; \boldsymbol{\tau})$ |
| 1 | $p^{\text{Ga}}(\boldsymbol{\gamma}_{(i,j)}; \boldsymbol{\tau}, \boldsymbol{\tau})$ | $p^{\text{RGDP}}(\mathbf{H}_{(i,j)}; 1, 1, \boldsymbol{\tau})$ |

Author Manuscript

Author Manuscript

Author Manuscript

Author Manuscript

Table 3

Normalized KKT residual for S-NNLS algorithms on synthetic data. For all experiments, $d = 100$ and $k = 10$.

| n | 200 | 400 | 800 |
|---------------------|-------------|--------------|--------------|
| Reweighted ℓ_2 | $10^{-9.3}$ | $10^{-9.4}$ | $10^{-9.6}$ |
| Reweighted ℓ_1 | $10^{-9.9}$ | $10^{-10.1}$ | $10^{-10.4}$ |

Author Manuscript

Author Manuscript

Author Manuscript

Author Manuscript

Table 4

Normalized KKT residual for S-NMF-W algorithms on CSBL face dataset.

| | W | H |
|---------------------|-------------|-------------|
| Reweighted ℓ_2 | $10^{-3.9}$ | $10^{-5.3}$ |
| Reweighted ℓ_1 | 10^{-5} | $10^{-7.3}$ |

Author Manuscript

Author Manuscript

Author Manuscript

Author Manuscript