Automatic Relevance Determination in Nonnegative Matrix Factorization with the β-Divergence

Vincent Y.F. Tan, *Member*, *IEEE*, and Cédric Févotte, *Member*, *IEEE*

Abstract—This paper addresses the estimation of the latent dimensionality in nonnegative matrix factorization (NMF) with the β-divergence. The β-divergence is a family of cost functions that includes the squared euclidean distance, Kullback-Leibler (KL) and Itakura-Saito (IS) divergences as special cases. Learning the model order is important as it is necessary to strike the right balance between data fidelity and overfitting. We propose a Bayesian model based on *automatic relevance determination* (ARD) in which the columns of the dictionary matrix and the rows of the activation matrix are tied together through a common scale parameter in their prior. A family of majorization-minimization (MM) algorithms is proposed for maximum a posteriori (MAP) estimation. A subset of scale parameters is driven to a small lower bound in the course of inference, with the effect of pruning the corresponding spurious components. We demonstrate the efficacy and robustness of our algorithms by performing extensive experiments on synthetic data, the `swimmer` dataset, a music decomposition example, and a stock price prediction task.

Index Terms—Nonnegative matrix factorization, model order selection, majorization-minimization, group-sparsity, automatic relevance determination

1 Introduction

Given a data matrix $\mathbf{V}$ of dimensions $F \times N$ with nonnegative entries, nonnegative matrix factorization (NMF) consists in finding a low-rank factorization

$$\mathbf{V} \approx \hat{\mathbf{V}} \triangleq \mathbf{WH}, \tag{1}$$

where $\mathbf{W}$ and $\mathbf{H}$ are nonnegative matrices of dimensions $F \times K$ and $K \times N$, respectively. The common dimension K is usually chosen such that $F\,K + K\,N \ll F\,N$; hence the overall number of parameters to describe the data (i.e., data dimension) is reduced. Early references on NMF include the work of Paatero and Tapper [1] and a seminal contribution by Lee and Seung [2]. Since then, NMF has become a widely used technique for nonsubtractive, parts-based representation of nonnegative data. There are numerous applications of NMF in diverse fields, such as audio signal processing [3], image classification [4], analysis of financial data [5], and bioinformatics [6]. The factorization (1) is usually sought after through the minimization problem:

$$\underset{\mathbf{W},\mathbf{H}}{\text{minimize}}\ D(\mathbf{V}|\mathbf{WH}) \ \text{ subject to } \ \mathbf{W} \geq 0, \mathbf{H} \geq 0, \tag{2}$$

where $\mathbf{A} \geq 0$ means that all entries of the matrix $\mathbf{A}$ are nonnegative (and not positive semidefiniteness). The function $D(\mathbf{V}|\mathbf{WH})$ is a separable measure of fit, i.e.,

$$D(\mathbf{V}|\mathbf{WH}) = \sum_{f=1}^{F}\sum_{n=1}^{N} d([\mathbf{V}]_{fn} \mid [\mathbf{WH}]_{fn}), \tag{3}$$

where $d(x|y)$ is a scalar cost function of $y \in \mathbb{R}_+$ given $x \in \mathbb{R}_+$, and it equals zero when $x = y$. In this paper, we will consider the $d(x|y)$ to be the β-divergence, a family of cost functions parameterized by a single scalar $\beta \in \mathbb{R}$. The squared euclidean (EUC) distance, the generalized Kullback-Leibler (KL) divergence, and the Itakura-Saito (IS) divergence are special cases of the β-divergence. NMF with the β-divergence (or, in short, β-NMF) was first considered by Cichocki et al. [7], and more detailed treatments have been proposed in [8], [9], and [10].

1.1 Main Contributions

In most applications, it is crucial that the "right" model order K is selected. If K is too small, the data does not fit the model well. Conversely, if K is too large, overfitting occurs. We seek to find an elegant solution for this dichotomy between data fidelity and overfitting. Traditional model selection techniques such as the Bayesian information criterion (BIC) [11] are not applicable in our setting as the number of parameters is $FK + KN$ and this scales linearly with the number of data points N, whereas BIC assumes that the number of parameters stays constant as the number of data points increases.

To ameliorate this problem, we propose a Bayesian model for β-NMF based on automatic relevance determination (ARD) [12], and in particular, we are inspired by Bayesian principal component analysis (PCA) [13]. We derive *computationally efficient* algorithms with *monotonicity*

- *V.Y.F. Tan is with the Institute for Infocomm Research, A*STAR, 1 Fusionopolis Way, #21-01 Connexis (South Tower), Singapore 138632, and the Department of Electrical and Computer Engineering, National University of Singapore, Singapore. E-mail: vtan@nus.edu.sg.*
- *C. Févotte is with Laboratoire Lagrange (CNRS, Observatoire de la Côte d'Azur & Université de Nice Sophia Antipolis), Parc Valrose, 06000 Nice, France. E-mail: cfevotte@unice.fr.*

Manuscript received 23 Apr. 2012; revised 15 Aug. 2012; accepted 1 Oct. 2012; published online 26 Oct. 2012.
Recommended for acceptance by Y.W. Teh.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2012-04-0312.
Digital Object Identifier no. 10.1109/TPAMI.2012.240.

guarantees to estimate the model order K and to estimate the basis $\mathbf{W}$ and the activation coefficients $\mathbf{H}$. The proposed algorithms are based on the use of auxiliary functions (local majorizations of the objective function). The optimization of these auxiliary functions leads directly to majorization-minimization (MM) algorithms, resulting in efficient multiplicative updates. The monotonicity of the objective function can be proven by leveraging on techniques in [9]. We show via simulations in Section 6 on synthetic data and real datasets (such as a music decomposition example) that the proposed algorithms recover the correct model order and produce better decompositions. We also describe a procedure based on the *method of moments* for adaptive and data-dependent selection of some of the hyperparameters.

1.2 Prior Work

To the best of our knowledge, there is fairly limited literature on model order selection in NMF. References [14] and [15] describe Markov chain Monte Carlo (MCMC) strategies for evaluation of the model evidence in EUC-NMF or KL-NMF. The evidence is calculated for each candidate value of K, and the model with highest evidence is selected. The studies in [16] and [17] describe reversible jump MCMC approaches that allow to sample the model order K, along with any other parameter. These sampling-based methods are computationally intensive. Another class of methods, given in [18], [19], [20], and [21], is closer to the principles that underlie this work; in these works, the number of components K is set to a large value and irrelevant components in $\mathbf{W}$ and $\mathbf{H}$ are driven to zero during inference. A detailed but qualitative comparison between our work and these methods is given in Section 5. In Section 6, we compare the empirical performance of our methods to [18] and [21].

This paper is a significant extension of the authors' conference publication in [22]. First, the cost function in [22] was restricted to be the KL-divergence. In this paper, we consider a continuum of costs parameterized by β, underlying different statistical noise models. We show that this flexibility in the cost function allows for better quality of factorization and model selection on various classes of real-world signals such as audio and images. Second, the algorithms described herein are such that the cost function monotonically decreases to a local minimum whereas the algorithm in [22] is heuristic. Convergence is guaranteed by the MM framework.

1.3 Paper Organization

In Section 2, we state our notation and introduce β-NMF and the MM technique. In Section 3, we present our Bayesian model for β-NMF. Section 4 details ℓ_1- and ℓ_2-ARD for model selection in β-NMF. We then compare the proposed algorithms to other related works in Section 5. In Section 6, we present extensive numerical results to demonstrate the efficacy and robustness of ℓ_1- and ℓ_2-ARD. We conclude the discussion in Section 7.

2 Preliminaries

2.1 Notations

We denote by $\mathbf{V}$, $\mathbf{W}$, and $\mathbf{H}$, the data, dictionary and activation matrices, respectively. These nonnegative matrices are of dimensions $F \times N$, $F \times K$, and $K \times N$, respectively. The entries of these matrices are denoted by v_{fn}, w_{fk}, and h_{kn} respectively. The kth *column* of $\mathbf{W}$ is denoted by $\mathbf{w}_k \in \mathbb{R}_+^F$, and $\underline{h}_k \in \mathbb{R}_+^N$ denotes the kth *row* of $\mathbf{H}$. Thus, $\mathbf{W} = [\mathbf{w}_1, \ldots, \mathbf{w}_K]$ and $\mathbf{H} = [\underline{h}_1^T, \ldots, \underline{h}_K^T]^T$.

2.2 NMF with the β-Divergence

This paper considers NMF based on the β-divergence, which we now review. The β-divergence was originally introduced for $\beta \geq 1$ in [23] and [24] and later generalized to $\beta \in \mathbb{R}$ in [7], which is the definition we use here:

$$d_\beta(x|y) \triangleq \begin{cases} \dfrac{x^\beta}{\beta\,(\beta-1)} + \dfrac{y^\beta}{\beta} - \dfrac{x\,y^{\beta-1}}{\beta-1}, & \beta \in \mathbb{R}\backslash\{0,1\}, \\ x\,\log\dfrac{x}{y} - x + y, & \beta = 1, \\ \dfrac{x}{y} - \log\dfrac{x}{y} - 1, & \beta = 0. \end{cases} \quad (4)$$

The limiting cases $\beta = 0$ and $\beta = 1$ correspond to the IS and KL-divergences, respectively. Another case of note is $\beta = 2$, which corresponds to the squared euclidean distance, i.e., $d_{\beta=2}(x|y) = (x-y)^2/2$. The parameter β essentially controls the assumed statistics of the observation noise and can either be fixed or learned from training data by cross-validation. Under certain assumptions, the β-divergence can be mapped to a log-likelihood function for the Tweedie distribution [25], parameterized with respect to its mean. In particular, the values $\beta = 0, 1, 2$ underlie the multiplicative Gamma observation noise, Poisson noise, and Gaussian additive observation noise, respectively. We describe this property in greater detail in Section 3.2. The β-divergence offers a continuum of noise statistics that interpolates between these three specific cases. In the following, we use the notation $D_\beta(\mathbf{V}|\mathbf{WH})$ to denote the separable cost function in (3) with the scalar cost $d = d_\beta$ in (4).

2.3 Majorization-Minimization for β-NMF

We briefly recall some results in [9] on standard β-NMF. In particular, we describe how an MM algorithm [26] that recovers a stationary point of (3) can be derived. The algorithm updates $\mathbf{H}$ given $\mathbf{W}$, and $\mathbf{W}$ given $\mathbf{H}$, and these two steps are essentially the same by the symmetry of $\mathbf{W}$ and $\mathbf{H}$ by transposition ($\mathbf{V} \approx \mathbf{WH}$ is equivalent to $\mathbf{V}^T \approx \mathbf{H}^T\mathbf{W}^T$). Let us thus focus on the optimization of $\mathbf{H}$ given $\mathbf{W}$. The MM framework involves building a (nonnegative) *auxiliary function* $G(\mathbf{H}|\tilde{\mathbf{H}})$ that majorizes the objective $C(\mathbf{H}) = D_\beta(\mathbf{V}|\mathbf{WH})$ everywhere, i.e.,

$$G(\mathbf{H}|\tilde{\mathbf{H}}) \geq C(\mathbf{H}), \quad (5)$$

for all pairs of nonnegative matrices $\mathbf{H}, \tilde{\mathbf{H}} \in \mathbb{R}_+^{K\times N}$. The auxiliary function also matches the cost function whenever its arguments are the same, i.e., for all $\tilde{\mathbf{H}}$,

$$G(\tilde{\mathbf{H}}|\tilde{\mathbf{H}}) = C(\tilde{\mathbf{H}}). \quad (6)$$

If such an auxiliary function exists and the optimization of $G(\mathbf{H}|\tilde{\mathbf{H}})$ over $\mathbf{H}$ for fixed $\tilde{\mathbf{H}}$ is simple, the optimization of $C(\mathbf{H})$ may be replaced by the simpler optimization of $G(\mathbf{H}|\tilde{\mathbf{H}})$ over $\mathbf{H}$. Indeed, any iterate $\mathbf{H}^{(i+1)}$ such that $G(\mathbf{H}^{(i+1)}|\mathbf{H}^{(i)}) \leq G(\mathbf{H}^{(i)}|\mathbf{H}^{(i)})$ reduces the cost since

$$C(\mathbf{H}^{(i+1)}) \leq G(\mathbf{H}^{(i+1)}|\mathbf{H}^{(i)}) \leq G(\mathbf{H}^{(i)}|\mathbf{H}^{(i)}) = C(\mathbf{H}^{(i)}). \quad (7)$$

TABLE 1
The Form of the Auxiliary Function for Various βs [9]

Auxiliary function $G(\mathbf{H}\|\tilde{\mathbf{H}})$	β
$\sum_{kn} q_{kn}h_{kn} - \frac{1}{\beta-1}p_{kn}\tilde{h}_{kn}\left(\frac{h_{kn}}{\tilde{h}_{kn}}\right)^{\beta-1} + \text{cst}$	$\beta < 1$
$\sum_{kn} q_{kn}h_{kn} - p_{kn}\tilde{h}_{kn}\log\left(\frac{h_{kn}}{\tilde{h}_{kn}}\right) + \text{cst}$	$\beta = 1$
$\sum_{kn} \frac{1}{\beta}q_{kn}\tilde{h}_{kn}\left(\frac{h_{kn}}{\tilde{h}_{kn}}\right)^{\beta} - \frac{1}{\beta-1}p_{kn}\tilde{h}_{kn}\left(\frac{h_{kn}}{\tilde{h}_{kn}}\right)^{\beta-1} + \text{cst}$	$\beta \in (1, 2]$
$\sum_{kn} \frac{1}{\beta}q_{kn}\tilde{h}_{kn}\left(\frac{h_{kn}}{\tilde{h}_{kn}}\right)^{\beta} - p_{kn}\tilde{h}_{kn} + \text{cst}$	$\beta > 2$

The first inequality follows from (5) and the second from the optimality of $\mathbf{H}^{(i+1)}$. Thus, the MM update is

$$\mathbf{H}^{(i+1)} = \underset{\mathbf{H}\geq 0}{\arg\min}\ G(\mathbf{H}|\mathbf{H}^{(i)}). \tag{8}$$

Note that if $\mathbf{H}^{(i+1)} = \mathbf{H}^{(i)}$, a local minimum is attained since the inequalities in (7) are equalities. The key of the MM approach is thus to build an auxiliary function G which reasonably approximates the original objective at the current iterate $\tilde{\mathbf{H}}$ and such that the function is easy to minimize (over the first variable $\mathbf{H}$). In our setting, the objective function $C(\mathbf{H})$ can be decomposed into the sum of a convex term and a concave term. As such, the construction proposed in [8] and [9] involves majorizing the convex and concave terms separately, using Jensen's inequality and a first-order Taylor approximation, respectively. Denoting $\tilde{v}_{fn} \triangleq [\mathbf{W}\tilde{\mathbf{H}}]_{fn}$ and

$$p_{kn} \triangleq \sum_f w_{fk}v_{fn}\tilde{v}_{fn}^{\beta-2}, \qquad q_{kn} \triangleq \sum_f w_{fk}\tilde{v}_{fn}^{\beta-1}, \tag{9}$$

the resulting auxiliary function can be expressed as in Table 1, where cst denotes constant terms that do not depend on $\mathbf{H}$. In the sequel, the use of the tilde over a parameter will generally denote its *previous* iterate. Minimization of $G(\mathbf{H}|\tilde{\mathbf{H}})$ with respect to (w.r.t) $\mathbf{H}$ thus leads to the following simple update:

$$h_{kn} = \tilde{h}_{kn}\left(\frac{p_{kn}}{q_{kn}}\right)^{\gamma(\beta)}, \tag{10}$$

where the exponent $\gamma(\beta)$ is defined as

$$\gamma(\beta) \triangleq \begin{cases} 1/(2-\beta), & \beta < 1, \\ 1, & 1 \leq \beta \leq 2, \\ 1/(\beta-1), & \beta > 2. \end{cases} \tag{11}$$

3 The Model for Automatic Relevance Determination in β-NMF

In this section, we describe our probabilistic model for NMF. The model involves tying the kth column of $\mathbf{W}$ to the kth row of $\mathbf{H}$ together through a common scale parameter λ_k. If λ_k is driven to zero (or, as we will see, a positive lower bound) during inference, then all entries in the corresponding column of $\mathbf{W}$ and row of $\mathbf{H}$ will also be driven to zero.

3.1 Priors

We are inspired by Bayesian PCA [13], where each element of $\mathbf{W}$ is assigned a Gaussian prior with column-dependent variance-like parameters λ_k. These λ_ks are known as the *relevance weights*. However, our formulation has two main differences vis-à-vis Bayesian PCA. First, there are no nonnegativity constraints in Bayesian PCA. Second, in Bayesian PCA, thanks to the simplicity of the statistical model (multivariate Gaussian observations with Gaussian parameter priors), $\mathbf{H}$ can be easily integrated out of the likelihood, and the optimization can be done over $p(\mathbf{W}, \boldsymbol{\lambda}|\mathbf{V})$, where $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_K) \in \mathbb{R}_+^K$ is the vector of relevance weights. We have to maintain the nonnegativity of the elements in $\mathbf{W}$ and $\mathbf{H}$ and also, in our setting, the activation matrix $\mathbf{H}$ cannot be integrated out analytically.

To ameliorate the above-mentioned problems, we propose to tie the columns of $\mathbf{W}$ and the rows of $\mathbf{H}$ together through common scale parameters. This construction is not overconstraining the scales of $\mathbf{W}$ and $\mathbf{H}$ because of the inherent scale indeterminacy between $\mathbf{w}_k$ and $\underline{h}_k$. Moreover, we choose nonnegative priors for $\mathbf{W}$ and $\mathbf{H}$ to ensure that all elements of the basis and activation matrices are nonnegative. We adopt a Bayesian approach and assign $\mathbf{W}$ and $\mathbf{H}$ Half-Normal or Exponential priors. When $\mathbf{W}$ and $\mathbf{H}$ have Half-Normal priors:

$$p(w_{fk}|\lambda_k) = \mathcal{HN}(w_{fk}|\lambda_k), \quad p(h_{kn}|\lambda_k) = \mathcal{HN}(h_{kn}|\lambda_k), \tag{12}$$

where for $x \geq 0$, $\mathcal{HN}(x|\lambda) \triangleq (\frac{2}{\pi\lambda})^{1/2}\exp(-\frac{x^2}{2\lambda})$, and $\mathcal{HN}(x \mid \lambda) = 0$ when $x < 0$. Note that if x is a Gaussian (Normal) random variable, then $|x|$ is a Half-Normal. When $\mathbf{W}$ and $\mathbf{H}$ are assigned Exponential priors:

$$p(w_{fk}|\lambda_k) = \mathcal{E}(w_{fk}|\lambda_k), \quad p(h_{kn}|\lambda_k) = \mathcal{E}(h_{kn}|\lambda_k), \tag{13}$$

where for $x \geq 0$, $\mathcal{E}(x|\lambda) \triangleq \frac{1}{\lambda}\exp(-\frac{x}{\lambda})$, and $\mathcal{E}(x|\lambda) = 0$ otherwise. Note from (12) and (13) that the kth column of $\mathbf{W}$ and the kth row of $\mathbf{H}$ are tied together by a *common* variance-like parameter λ_k, also known as the *relevance weight*. When a particular λ_k is small, that particular column of $\mathbf{W}$ *and* row of $\mathbf{H}$ are not relevant and vice versa. When a row and a column are not relevant, their norms are close to zero and thus can be removed from the factorization without compromising too much on data fidelity. This removal of *common* rows and columns makes the model more parsimonious.

Finally, we impose inverse-Gamma priors on each relevance weight λ_k, i.e.,

$$p(\lambda_k; a, b) = \mathcal{IG}(\lambda_k|a, b) = \frac{b^a}{\Gamma(a)}\lambda_k^{-(a+1)}\exp\left(-\frac{b}{\lambda_k}\right), \tag{14}$$

where a and b are the (nonnegative) shape and scale hyperparameters, respectively. We set a and b to be constant

for all k. We will state how to choose these in a principled manner in Section 4.5. Furthermore, each relevance parameter is independent of every other, i.e., $p(\boldsymbol{\lambda}; a, b) = \prod_{k=1}^{K} p(\lambda_k; a, b)$.

3.2 Likelihood

The β-divergence is related to the family of Tweedie distributions [25]. The relation was noted by Cichocki et al. [27] and detailed in [28]. The Tweedie distribution is a special case of the exponential dispersion model [29], itself a generalization of the more familiar natural exponential family. It is characterized by the simple polynomial relation between its mean and variance:

$$\mathrm{var}[x] = \phi \mu^{2-\beta}, \tag{15}$$

where $\mu = \mathbb{E}[x]$ is the mean, β is the *shape parameter* , and ϕ is referred to as the *dispersion parameter*. The Tweedie distribution is only defined for $\beta \leq 1$ and $\beta \geq 2$. For $\beta \neq 0, 1$, its probability density function (pdf) or probability mass function (pmf) can be written in the following form:

$$\mathcal{T}(x|\mu, \phi, \beta) = h(x, \phi) \exp\left[\frac{1}{\phi}\left(\frac{1}{\beta - 1} x \mu^{\beta-1} - \frac{1}{\beta}\mu^{\beta}\right)\right], \tag{16}$$

where $h(x, \phi)$ is referred to as the *base function*. For $\beta \in \{0, 1\}$, the pdf or pmf takes the appropriate limiting form of (16). The support of $\mathcal{T}(x|\mu, \phi, \beta)$ varies with the value of β, but the set of values that μ can take on is generally $\mathbb{R}^+$, except for $\beta = 2$, for which it is $\mathbb{R}$, and the Tweedie distribution coincides with the Gaussian distribution of mean μ and variance ϕ. For $\beta = 1$ (and $\phi = 1$), the Tweedie distribution coincides with the Poisson distribution. For $\beta = 0$, it coincides with the Gamma distribution with shape parameter $\alpha = 1/\phi$ and scale parameter μ/α.[1] The base function admits a closed form only for $\beta \in \{-1, 0, 1, 2\}$.

Finally, the *deviance* of Tweedie distribution, i.e., the log-likelihood ratio of the saturated ($\mu = x$) and general model, is proportional to the β-divergence. In particular,

$$\log \frac{\mathcal{T}(x|\mu = x, \phi, \beta)}{\mathcal{T}(x|\mu, \phi, \beta)} = \frac{1}{\phi} d_{\beta}(x|\mu), \tag{17}$$

where $d_{\beta}(\,\cdot\,|\,\cdot\,)$ is the scalar cost function defined in (4). As such the β-divergence acts as a minus log-likelihood for the Tweedie distribution whenever the latter is defined. Because the data coefficients $\{v_{fn}\}$ are conditionally independent given $(\mathbf{W}, \mathbf{H})$, the negative log-likelihood function is

$$-\log p(\mathbf{V}|\mathbf{W}, \mathbf{H}) = \frac{1}{\phi} D_{\beta}(\mathbf{V}|\mathbf{WH}) + \mathrm{cst}. \tag{18}$$

3.3 Objective Function

We now form the maximum a posteriori (MAP) objective function for the model described in Sections 3.1 and 3.2. Due to (12), (13), (14), and (18):

$$C(\mathbf{W}, \mathbf{H}, \boldsymbol{\lambda}) \stackrel{\triangle}{=} -\log p(\mathbf{W}, \mathbf{H}, \boldsymbol{\lambda}|\mathbf{V}), \tag{19}$$

$$\begin{aligned} &= \frac{1}{\phi} D_{\beta}(\mathbf{V}|\mathbf{WH}) + \sum_{k=1}^{K} \frac{1}{\lambda_k}\left(f(\mathbf{w}_k) + f(\underline{h}_k) + b\right) \\ &\quad + c \log \lambda_k + \mathrm{cst}, \end{aligned} \tag{20}$$

where (20) follows from Bayes' rule and, for the two statistical models,

- Half-Normal model as in (12), $f(\mathbf{x}) = \frac{1}{2}\|\mathbf{x}\|_2^2$ and $c = (F + N)/2 + a + 1$;
- Exponential model as in (13), $f(\mathbf{x}) = \|\mathbf{x}\|_1$ and $c = F + N + a + 1$.

Observe that for the regularized cost function in (20), the second term is monotonically decreasing in λ_k, while the third term is monotonically increasing in λ_k. Thus, a subset of the λ_ks will be forced to a lower bound, which we specify in Section 4.4, while the others will tend to a larger value. This serves the purpose of pruning irrelevant components out of the model. In fact, the vector of relevance parameters $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_K)$ can be optimized analytically in (20) leading to an objective function that is a function of $\mathbf{W}$ and $\mathbf{H}$ only, i.e.,

$$\begin{aligned} C(\mathbf{W}, \mathbf{H}) &= \frac{1}{\phi} D_{\beta}(\mathbf{V}|\mathbf{WH}) \\ &\quad + c \sum_{k=1}^{K} \log(f(\mathbf{w}_k) + f(\underline{h}_k) + b) + \mathrm{cst}, \end{aligned} \tag{21}$$

where $\mathrm{cst} = Kc(1 - \log c)$.

In our algorithms, instead of optimizing (21), we keep λ_k as an auxiliary variable for optimizing $C(\mathbf{W}, \mathbf{H}, \boldsymbol{\lambda})$ in (20) to ensure that the columns $\mathbf{H}$ and the rows of $\mathbf{W}$ are decoupled. More precisely, $\mathbf{w}_k$ and $\underline{h}_k$ are conditionally independent given λ_k. In fact, (21) shows that the $\boldsymbol{\lambda}$-optimized objective function $C(\mathbf{W}, \mathbf{H})$ induces sparse regularization among groups, where the groups are pairs of columns and rows, i.e., $\{\mathbf{w}_k, \underline{h}_k\}$. In this sense, our work is related to group LASSO [30] and its variants. See, for example, [31]. The function $x \mapsto \log(x + b)$ in (21) is a sparsity-inducing term and is related to reweighted ℓ_1-minimization [32]. We discuss these connections in greater detail in the supplementary material, which can be found in the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2012.240, [33].

4 Inference Algorithms

In this section, we describe two algorithms for optimizing the objective function (20) for $\mathbf{H}$ given fixed $\mathbf{W}$. The updates for $\mathbf{W}$ are symmetric given $\mathbf{H}$. These algorithms will be based on the MM idea for β-NMF and on the two prior distributions of $\mathbf{W}$ and $\mathbf{H}$. In particular, we use the auxiliary function $G(\mathbf{H}|\dot{\mathbf{H}})$ defined in Table 1 as an upper bound of the data fit term $D_{\beta}(\mathbf{V}|\mathbf{WH})$.

4.1 Algorithm for ℓ_2-ARD β-NMF

We now introduce ℓ_2-ARD β-NMF. In this algorithm, we assume that $\mathbf{W}$ and $\mathbf{H}$ have Half-Normal priors as in (12) and thus the regularizer is

1. We employ the following convention for the Gamma distribution $\mathcal{G}(x; a, b) = x^{a-1} e^{-x/b}/(b^a \Gamma(a))$.

$$R_2(\mathbf{H}) \triangleq \sum_k \frac{1}{\lambda_k} f(\underline{h}_k) = \sum_{kn} \frac{1}{2\lambda_k} h_{kn}^2. \tag{22}$$

The main idea behind the algorithms is as follows: Consider the function $F(\mathbf{H}|\tilde{\mathbf{H}}) \triangleq \phi^{-1}\, G(\mathbf{H}|\tilde{\mathbf{H}}) + R_2(\mathbf{H})$, which is the original auxiliary function $G(\mathbf{H}|\tilde{\mathbf{H}})$ times ϕ^{-1} plus the ℓ_2 regularization term. It can, in fact, be easily shown in [9, Section 6] that $F(\mathbf{H}|\tilde{\mathbf{H}})$ is an auxiliary function to the (penalized) objective function in (20). Ideally, we would take the derivative of $F(\mathbf{H}|\tilde{\mathbf{H}})$ w.r.t h_{kn} and set it to zero. Then the updates would proceed in a manner analogous to (10). However, the regularization term $R_2(\mathbf{H})$ does not "fit well" with the form of the auxiliary function $G(\mathbf{H}|\tilde{\mathbf{H}})$ in the sense that $\nabla_{\mathbf{H}} F(\mathbf{H}|\tilde{\mathbf{H}}) = 0$ cannot be solved analytically for all $\beta \in \mathbb{R}$. Thus, our idea for ℓ_2-ARD is to consider the cases $\beta \geq 2$ and $\beta < 2$ separately and to find an upper bound of $F(\mathbf{H}|\tilde{\mathbf{H}})$ by some other auxiliary function $J(\mathbf{H}|\tilde{\mathbf{H}})$ so that the resulting equation $\nabla_{\mathbf{H}} J(\mathbf{H}|\tilde{\mathbf{H}}) = 0$ can be solved in closed-form.

To derive our algorithms, we first note the following.

Lemma 1. *For every $\nu > 0$, the function $g_\nu(t) = \frac{1}{t}(\nu^t - 1)$ is monotonically nondecreasing in $t \in \mathbb{R}$. In fact, $g_\nu(t)$ is monotonically increasing unless $\nu = 1$.*

In the above lemma, $g_\nu(0) \triangleq \log \nu$ by L'Hôpital's rule. The proof of this simple result can be found in [34].

We first derive ℓ_2-ARD for $\beta > 2$. The idea is to upper bound the regularizer $R_2(\mathbf{H})$ in (22) elementwise using Lemma 1, and is equivalent to the *moving-term* technique described by Yang and Oja in [34] and [35]. Indeed, we have

$$\frac{1}{2}\left[\left(\frac{h_{kn}}{\tilde{h}_{kn}}\right)^2 - 1\right] \leq \frac{1}{\beta}\left[\left(\frac{h_{kn}}{\tilde{h}_{kn}}\right)^\beta - 1\right], \tag{23}$$

by taking $\nu = h_{kn}/\tilde{h}_{kn}$ in Lemma 1. Thus, for $\beta > 2$,

$$\frac{1}{2\lambda_k} h_{kn}^2 \leq \frac{1}{\lambda_k \beta} \tilde{h}_{kn}^2 \left(\frac{h_{kn}}{\tilde{h}_{kn}}\right)^\beta + \text{cst}, \tag{24}$$

where cst is a constant w.r.t the optimization variable h_{kn}. We upper bound the regularizer (22) elementwise by (24). The resulting auxiliary function (modified version of $F(\mathbf{H}|\tilde{\mathbf{H}})$) is

$$J(\mathbf{H}|\tilde{\mathbf{H}}) = \frac{1}{\phi}\, G(\mathbf{H}|\tilde{\mathbf{H}}) + \sum_{kn} \frac{1}{\lambda_k \beta} \tilde{h}_{kn}^2 \left(\frac{h_{kn}}{\tilde{h}_{kn}}\right)^\beta. \tag{25}$$

Note that (24) holds with equality iff $\nu = 1$ or, equivalently, $h_{kn} = \tilde{h}_{kn}$ so (6) holds. Thus, $J(\mathbf{H}|\tilde{\mathbf{H}})$ is indeed an auxiliary function to $F(\mathbf{H}|\tilde{\mathbf{H}})$. Recalling the definition of $G(\mathbf{H}|\tilde{\mathbf{H}})$ for $\beta > 2$ in Table 1, differentiating $J(\mathbf{H}|\tilde{\mathbf{H}})$ w.r.t h_{kn} and setting the result to zero yields the update

$$h_{kn} = \tilde{h}_{kn} \left(\frac{p_{kn}}{q_{kn} + (\phi/\lambda_k)\tilde{h}_{kn}}\right)^{1/(\beta-1)}. \tag{26}$$

Note that the exponent $1/(\beta-1)$ corresponds to $\gamma(\beta)$ for the $\beta > 2$ case. Also observe that the update is similar to MM for β-NMF (cf. (10)) except that there is an additional term in the denominator.

Algorithm 1. ℓ_2-ARD for β-NMF

Input: Data matrix $\mathbf{V}$, hyperparameter a, tolerance τ

Output: Nonnegative matrices $\mathbf{W}$ and $\mathbf{H}$, nonnegative relevance vector $\boldsymbol{\lambda}$ and model order K_{eff}

Init: Fix K. Initialize $\mathbf{W} \in \mathbb{R}_+^{F\times K}$ and $\mathbf{H} \in \mathbb{R}_+^{K\times N}$ to nonnegative values and tolerance parameter tol $= \infty$

Calculate: $c = (F+N)/2 + a + 1$ and $\xi(\beta)$ as in (31)

Calculate: Hyperparameter b as in (38)

while (tol $< \tau$) **do**

$$\mathbf{H} \leftarrow \mathbf{H} \cdot \left(\frac{\mathbf{W}^T[(\mathbf{WH})^{\cdot(\beta-2)} \cdot \mathbf{V}]}{\mathbf{W}^T[(\mathbf{WH})]^{\cdot(\beta-1)} + \phi \mathbf{H}/\text{repmat}(\boldsymbol{\lambda},1,N)}\right)^{\cdot \xi(\beta)}$$

$$\mathbf{W} \leftarrow \mathbf{W} \cdot \left(\frac{[(\mathbf{WH})^{\cdot(\beta-2)} \cdot \mathbf{V}]\mathbf{H}^T}{[(\mathbf{WH})^{\cdot(\beta-1)}]\mathbf{H}^T + \phi \mathbf{W}/\text{repmat}(\boldsymbol{\lambda},F,1)}\right)^{\cdot \xi(\beta)}$$

$\lambda_k \leftarrow [(\frac{1}{2}\sum_f w_{fk}^2 + \frac{1}{2}\sum_n h_{kn}^2) + b]/c$ for all k

tol $\leftarrow \max_{k=1,\ldots,K} |(\lambda_k - \tilde{\lambda}_k)/\tilde{\lambda}_k|$

end while

Calculate: K_{eff} as in (34)

For the case $\beta \leq 2$, our strategy is not to majorize the regularization term. Rather, we majorize the auxiliary function $G(\mathbf{H}|\tilde{\mathbf{H}})$ itself. By applying Lemma 1 with $\nu = h_{kn}/\tilde{h}_{kn}$, we have that for all $\beta \leq 2$:

$$\frac{1}{\beta}\left[\left(\frac{h_{kn}}{\tilde{h}_{kn}}\right)^\beta - 1\right] \leq \frac{1}{2}\left[\left(\frac{h_{kn}}{\tilde{h}_{kn}}\right)^2 - 1\right], \tag{27}$$

which means that

$$\frac{1}{\beta} q_{kn} \tilde{h}_{kn} \left(\frac{h_{kn}}{\tilde{h}_{kn}}\right)^\beta \leq \frac{1}{2} q_{kn} \tilde{h}_{kn} \left(\frac{h_{kn}}{\tilde{h}_{kn}}\right)^2 + \text{cst}. \tag{28}$$

By replacing the first term of $G(\mathbf{H}|\tilde{\mathbf{H}})$ in Table 1 (for $\beta \leq 2$) with the upper bound above, we have the following new objective function:

$$J(\mathbf{H}|\tilde{\mathbf{H}}) = \sum_{kn} \frac{q_{kn}\tilde{h}_{kn}}{2\phi}\left(\frac{h_{kn}}{\tilde{h}_{kn}}\right)^2 - \frac{p_{kn}\tilde{h}_{kn}}{\phi(\beta-1)}\left(\frac{h_{kn}}{\tilde{h}_{kn}}\right)^{\beta-1} + \frac{h_{kn}^2}{2\lambda_k}. \tag{29}$$

Differentiating $J(\mathbf{H}|\tilde{\mathbf{H}})$ w.r.t h_{kn} and setting to zero yields the simple update

$$h_{kn} = \tilde{h}_{kn} \left(\frac{p_{kn}}{q_{kn} + (\phi/\lambda_k)\tilde{h}_{kn}}\right)^{1/(3-\beta)}. \tag{30}$$

To summarize the algorithm concisely, we define the exponent used in the updates in (26) and (30) as

$$\xi(\beta) \triangleq \begin{cases} 1/(3-\beta) & \beta \leq 2, \\ 1/(\beta-1) & \beta > 2. \end{cases} \tag{31}$$

Finally, we remark that even though the updates in (26) and (30) are easy to implement, we either majorized the regularizer $R_2(\mathbf{H})$ or the auxiliary function $G(\mathbf{H}|\tilde{\mathbf{H}})$. These bounds may be loose and thus may lead to slow convergence in the resulting algorithm. In fact, we can show that for $\beta = 0, 1, 2$, we do not have to resort to upper bounding the original function $F(\mathbf{H}|\tilde{\mathbf{H}}) = \phi^{-1}\, G(\mathbf{H}|\tilde{\mathbf{H}}) + R_2(\mathbf{H})$. Instead, we can choose to solve a polynomial equation to update h_{kn}. The cases $\beta = 0, 1, 2$ correspond to

solving cubic, quadratic, and linear equations in h_{kn}, respectively. It is also true that for all rational β, we can form a polynomial equation in h_{kn}, but the order of the resulting polynomial depends on the exact value of β. See the online supplementary material [33].

4.2 Algorithm for ℓ_1-ARD β-NMF

The derivation of ℓ_1-ARD β-NMF is similar to its ℓ_2 counterpart. We find majorizers for either the likelihood or the regularizer. We omit the derivations and refer the reader to the online supplementary material [33]. In sum:

$$h_{kn} = \tilde{h}_{kn}\left(\frac{p_{kn}}{q_{kn}+\phi/\lambda_k}\right)^{\gamma(\beta)}, \tag{32}$$

where $\gamma(\beta)$ is defined in (11).

Algorithm 2. ℓ_1-ARD for β-NMF

Input: Data matrix $\mathbf{V}$, hyperparameter a, tolerance τ

Output: Nonnegative matrices $\mathbf{W}$ and $\mathbf{H}$, nonnegative relevance vector $\boldsymbol{\lambda}$ and model order K_{eff}

Init: Fix K. Initialize $\mathbf{W} \in \mathbb{R}_+^{F\times K}$ and $\mathbf{H} \in \mathbb{R}_+^{K\times N}$ to nonnegative values and tolerance parameter $\text{tol} = \infty$

Calculate: $c = F + N + a + 1$ and $\gamma(\beta)$ as in (11)

Calculate: Hyperparameter b as in (38)

while (tol $< \tau$) **do**

$$\mathbf{H} \leftarrow \mathbf{H} \cdot \left(\frac{\mathbf{W}^T[(\mathbf{WH})^{\cdot(\beta-2)}\cdot\mathbf{V}]}{\mathbf{W}^T[(\mathbf{WH})^{\cdot(\beta-1)}]+\phi/\text{repmat}(\boldsymbol{\lambda},1,N)}\right)^{\cdot\gamma(\beta)}$$

$$\mathbf{W} \leftarrow \mathbf{W} \cdot \left(\frac{[(\mathbf{WH})^{\cdot(\beta-2)}\cdot\mathbf{V}]\mathbf{H}^T}{[(\mathbf{WH})^{\cdot(\beta-1)}]\mathbf{H}^T+\phi/\text{repmat}(\boldsymbol{\lambda},F,1)}\right)^{\cdot\gamma(\beta)}$$

$\lambda_k \leftarrow (\sum_f w_{fk} + \sum_n h_{kn} + b)/c$ for all k

$\text{tol} \leftarrow \max_{k=1,\ldots,K}|(\lambda_k - \tilde{\lambda}_k)/\tilde{\lambda}_k|$

end while

Calculate: K_{eff} as in (34)

4.3 Update of λ_k

We have described how to update $\mathbf{H}$ using either ℓ_1-ARD or ℓ_2-ARD. Since $\mathbf{H}$ and $\mathbf{W}$ are related in a symmetric manner, we have also effectively described how to update $\mathbf{W}$. We now describe a simple update rule for the λ_ks. This update is the same for both ℓ_1- and ℓ_2-ARD. We first find the partial derivative of $C(\mathbf{W},\mathbf{H},\boldsymbol{\lambda})$ w.r.t λ_k and set it to zero. This gives the update:

$$\lambda_k = \frac{f(\mathbf{w}_k) + f(\underline{h}_k) + b}{c}, \tag{33}$$

where $f(\cdot)$ and c are defined after (20).

4.4 Stopping Criterion and Determination of K_{eff}

In this section, we describe the stopping criterion and the determination of the effective number of components K_{eff}. Let $\boldsymbol{\lambda} = (\lambda_1,\ldots,\lambda_K)$ and $\tilde{\boldsymbol{\lambda}} = (\tilde{\lambda}_1,\ldots,\tilde{\lambda}_K)$ be the vector of relevance weights at the current (updated) and previous iterations, respectively. The algorithm is terminated whenever $\text{tol} \triangleq \max_{k=1,\ldots,K}|(\lambda_k - \tilde{\lambda}_k)/\tilde{\lambda}_k|$ falls below some threshold $\tau > 0$. Note from (33) that iterates of each λ_k are bounded from below as $\lambda_k \geq B \triangleq b/c$ and this bound is attained when $\mathbf{w}_k$ and $\underline{h}_k$ are zero vectors, i.e., the kth column of $\mathbf{W}$ and the kth row of $\mathbf{H}$ are pruned out of the model. After convergence, we set K_{eff} to be the number of components of such that the ratio $(\lambda_k - B)/B$ is strictly larger than τ, i.e.,

$$K_{\text{eff}} \triangleq \left|\left\{k \in \{1,\ldots,K\} : \frac{\lambda_k - B}{B} > \tau\right\}\right|, \tag{34}$$

where $\tau > 0$ is some threshold. We choose this threshold to be the same as that for the tolerance level tol.

The algorithms ℓ_2-ARD and ℓ_1-ARD are detailed in Algorithms 1 and 2, respectively. In the algorithms, we use the notation $\mathbf{A}\cdot\mathbf{B}$ to mean entrywise multiplication of $\mathbf{A}$ and $\mathbf{B}$, $\frac{\mathbf{A}}{\mathbf{B}}$ to mean entrywise division, and $\mathbf{A}^{\cdot\gamma}$ to mean entrywise raising to the γth power. In addition, $\text{repmat}(\boldsymbol{\lambda},1,N)$ denotes the $K\times N$ matrix with each column being the λ vector.

4.5 Choosing the Hyperparameters

4.5.1 Choice of Dispersion Parameter ϕ

The dispersion parameter ϕ represents the tradeoff between the data fidelity and the regularization terms in (20). It needs to be fixed, based on prior knowledge about the noise distribution, or learned from the data using either cross-validation or MAP estimation. In the latter case, ϕ is assigned a prior $p(\phi)$ and the objective $C(\mathbf{W},\mathbf{H},\lambda,\phi)$ can be optimized over ϕ. This is a standard feature in penalized likelihood approaches and has been widely discussed in the literature. In this work, we will not address the estimation of ϕ, but only study the influence of the regularization term on the factorization *given* ϕ. In many cases, it is reasonable to fix ϕ based on prior knowledge. In particular, under the Gaussian noise assumption, $v_{fn} \sim \mathcal{N}(v_{fn}|\hat{v}_{fn},\sigma^2)$, and $\beta = 2$ and $\phi = \sigma^2$. Under the Poisson noise assumption, $v_{fn} \sim \mathcal{P}(v_{fn}|\hat{v}_{fn})$, and $\beta = 1$ and $\phi = 1$. Under multiplicative Gamma noise assumption, $v_{fn} = \hat{v}_{fn}\cdot\epsilon_{fn}$ and ϵ_{fn} is a Gamma noise of mean 1, or equivalently, $v_{fn} \sim \mathcal{G}(v_{kn}|\alpha, \hat{v}_{fn}/\alpha)$, and $\beta = 0$ and $\phi = 1/\alpha$. In audio applications where the power spectrogram is to be factorized, as in Section 6.3, the multiplicative exponential noise model (with $\alpha = 1$) is a generally agreed upon assumption [3] and thus $\phi = 1$.

4.5.2 Choice of Hyperparameters a and b

We now discuss how to choose the hyperparameters a and b in (14) in a data-dependent and principled way. Our method is related to the *method of moments*. We first focus on the selection of b using the sample mean of data, given a. Then the selection of a based on the sample variance of the data is discussed at the end of the section.

Consider the approximation in (1), which can be written element-wise as

$$v_{fn} \approx \hat{v}_{fn} = \sum_k w_{fk}h_{kn}. \tag{35}$$

The statistical models corresponding to shape parameter $\beta \notin (1,2)$ imply that $\mathbb{E}[v_{fn}|\hat{v}_{fn}] = \hat{v}_{fn}$. We extrapolate this property to derive a rule for selecting the hyperparameter b for all $\beta \in \mathbb{R}$ (and for nonnegative real-valued data in general), even though there is no known statistical model governing the noise when $\beta \in (1,2)$. When FN is large, the law of large numbers implies that the sample mean of the elements in $\mathbf{V}$ is close to the population mean (with high probability), i.e.,

$$\hat{\mu}_{\mathbf{V}} \triangleq \frac{1}{FN}\sum_{fn} v_{fn} \approx \mathbb{E}[v_{fn}] = \mathbb{E}[\hat{v}_{fn}] = \sum_k \mathbb{E}[w_{fk}h_{kn}]. \quad (36)$$

We can compute $\mathbb{E}[\hat{v}_{fn}]$ for the Half-Normal and Exponential models using the moments of these distributions and those of the inverse-Gamma for λ_k. These yield

$$\mathbb{E}[\hat{v}_{fn}] = \begin{cases} \dfrac{2Kb}{\pi(a-1)} & \text{Half-Normal}, \\ \dfrac{Kb^2}{(a-1)(a-2)} & \text{Exponential}. \end{cases} \quad (37)$$

By equating these expressions to the empirical mean $\hat{\mu}_{\mathbf{V}}$, we conclude that we can choose b according to

$$\hat{b} = \begin{cases} \dfrac{\pi(a-1)\hat{\mu}_{\mathbf{V}}}{2K} & \ell_2\text{-ARD}, \\ \sqrt{\dfrac{(a-1)(a-2)\hat{\mu}_{\mathbf{V}}}{K}} & \ell_1\text{-ARD}. \end{cases} \quad (38)$$

In summary, $\hat{b} \propto \hat{\mu}_{\mathbf{V}}/K$ and $\hat{b} \propto (\hat{\mu}_{\mathbf{V}}/K)^{1/2}$ for ℓ_2- and ℓ_1-ARD, respectively.

By using the empirical variance of $\mathbf{V}$ and the relation between the mean and variance of the Tweedie distribution in (15), we may also estimate a from the data. The resulting relations are more involved and these calculations are deferred to the online supplementary material [33] for $\beta \in \{0, 1, 2\}$. However, experiments showed that the resulting learning rules for a did not consistently give satisfactory results, especially when FN is not sufficiently large. In particular, the estimates sometimes fall out of the parameter space, which is a known feature of the method of moments. Observe that a appears in the objective function (21) only through $c = (F+N)/2 + a + 1$ (ℓ_2-ARD) or $c = F + N + a + 1$ (ℓ_1-ARD). As such, the influence of a is moderated by $F+N$. Hence, if we want to choose a prior on a that is not too informative, then we should choose a to be small compared to $F+N$. Experiments in Section 6 confirm that smaller values of a (relative to $F+N$) typically produce better results. As discussed in the conclusion, a more robust estimation of a (as well as b and ϕ) would involve a fully Bayesian treatment of our problem, which is left for future work.

5 Connections with Other Works

Our work draws parallels with a few other works on model order selection in NMF. The closest work is [18], which also proposes automatic component pruning via a MAP approach. It was developed during the same period as and independently of our earlier work [22]. An extension to multi-array analysis is also proposed in [19]. In [18], Mørup and Hansen consider NMF with the euclidean and KL costs. They constrained the columns of $\mathbf{W}$ to have unit norm (i.e., $\|\mathbf{w}_k\|_2 = 1$) and assumed that the coefficients of $\mathbf{H}$ are assigned exponential priors $\mathcal{E}(h_{kn}|\lambda_k)$. A noninformative Jeffrey's prior is further assumed on λ_k. Put together, they consider the following optimization over $(\mathbf{W}, \mathbf{H})$:

$$\begin{aligned} \underset{\mathbf{W},\mathbf{H},\boldsymbol{\lambda}}{\text{minimize}} \quad & D(\mathbf{V}|\mathbf{WH}) + \sum_k \frac{1}{\lambda_k}\|\underline{h}_k\|_1 + N\log\lambda_k \\ \text{subject to} \quad & \mathbf{W} \geq 0, \ \ \mathbf{H} \geq 0, \ \ \|\mathbf{w}_k\|_2 = 1, \ \ \forall\, k, \end{aligned} \quad (39)$$

where $D(\cdot|\cdot)$ is either the squared euclidean distance or the KL-divergence. A major difference compared to our objective function in (20) is that this method involves optimizing $\mathbf{W}$ under the constraint $\|\mathbf{w}_k\|_2 = 1$, which is nontrivial. As such, to solve (39), Mørup and Hansen [18] used a change of variables $\mathbf{w}'_k \leftarrow \mathbf{w}_k/\|\mathbf{w}_k\|_2$ and derived a heuristic multiplicative algorithm based on the ratio of negative and positive parts of the new objective function, along the lines of [36]. In contrast, our approach treats $\mathbf{w}_k$ and $\underline{h}_k$ symmetrically and the updates are simple. Furthermore, the pruning approach in [18] only occurs in the rows $\mathbf{H}$ and the corresponding columns of $\mathbf{W}$ may take any nonnegative value (subject to the norm constraint), which makes the estimation of these columns of $\mathbf{W}$ ill-posed (i.e., the parameterization is such that a part of the model is not observable). In contrast, in our approach $\mathbf{w}_k$ and $\underline{h}_k$ are tied together so they converge to zero jointly when λ_k reaches its lower bound.

Our work is also related to the automatic rank determination method in Projective NMF proposed by Yang et al. [20]. Following the principle of PCA, Projective NMF seeks a nonnegative matrix $\mathbf{W}$ such that the projection of $\mathbf{V}$ on the subspace spanned by $\mathbf{W}$ best fits $\mathbf{V}$. In other words, it is assumed that $\mathbf{H} = \mathbf{W}^T\mathbf{V}$. Following ARD in Bayesian PCA as originally described by Bishop [13], Yang et al. consider the additive Gaussian noise model and propose placing half-normal priors with relevance parameters λ_k on the columns of $\mathbf{W}$. They describe how to adapt EM to achieve MAP estimation of $\mathbf{W}$ and its relevance parameters.

Estimation of the model order in the Itakura-Saito NMF (multiplicative exponential noise) was addressed by Hoffman et al. [21]. They employ a nonparametric Bayesian setting in which K is assigned a large value (in principle, infinite), but the model is such that only a finite subset of components is retained. In their model, the coefficients of $\mathbf{W}$ and $\mathbf{H}$ have Gamma priors with fixed hyperparameters and a weight parameter θ_k is placed before each component in the factor model, i.e., $\hat{v}_{fn} = \sum_k \theta_k w_{fk} h_{kn}$. The weight, akin to the relevance parameter in our setting, is assigned a Gamma prior with a sparsity-enforcing shape parameter. A difference with our model is the a priori independence of the factors and the weights. Variational inference is used in [21].

In contrast with the above-mentioned works, the work herein presents a unified framework for model selection in β-NMF. The proposed algorithms have low complexity per iteration and are simple to implement while decreasing the objective function at every iteration. We compare the performance of our algorithms to those in [18] and [21] in Sections 6.3 (music decomposition) and 6.4 (stock price prediction).

6 Experiments

In this section, we present extensive numerical experiments demonstrating the robustness and efficiency of the proposed algorithms for 1) uncovering the correct model order and 2) learning better decompositions for modeling nonnegative data.

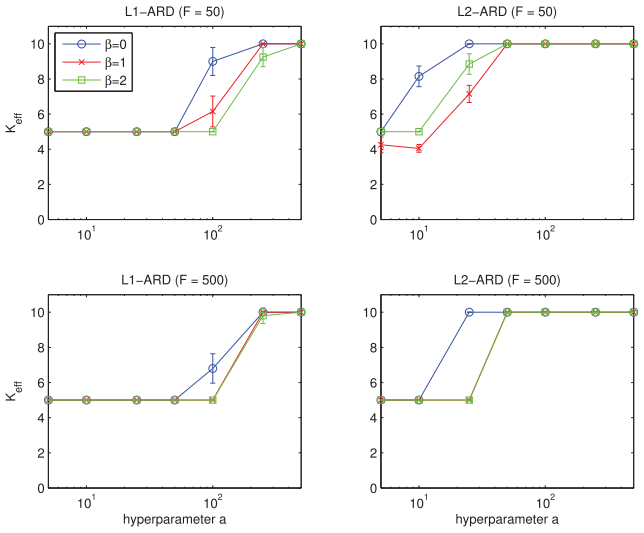

Fig. 1. Estimated number of components as a function of the hyperparameter a (log-linear plot). The true model order is $K_{\text{true}} = 5$. The solid line is the mean across 10 runs and the error bars display $\pm$ the standard deviation.

6.1 Simulations with Synthetic Data

In this section, we describe experiments on synthetic data generated according to the model. In particular, we fixed a pair of hyperparameters $(a_{\text{true}}, b_{\text{true}})$ and sampled $K_{\text{true}} = 5$ relevance weights λ_k according to the inverse-Gamma prior in (14). Conditioned on these relevance weights, we sampled the elements of $\mathbf{W}$ and $\mathbf{H}$ from the Half-Normal or Exponential models depending on whether we chose to use ℓ_2- or ℓ_1-ARD. These models are defined in (12) and (13), respectively. We set $a_{\text{true}} = 50$ and $b_{\text{true}} = 70$ for reasons that will be made clear in the following. We defined the noiseless matrix $\hat{\mathbf{V}}$ as $\mathbf{WH}$. We then generated a noisy matrix $\mathbf{V}$ given $\hat{\mathbf{V}}$ according to the three statistical models $\beta = 0, 1, 2$ corresponding to IS-, KL- and EUC-NMF, respectively. More precisely, the parameters of the noise models are chosen so that the signal-to-noise ratio SNR in dB, defined as $\text{SNR} = 20\log_{10}(\|\hat{\mathbf{V}}\|_F / \|\mathbf{V} - \hat{\mathbf{V}}\|_F)$, is approximately 10 dB for each $\beta \in \{0, 1, 2\}$. For $\beta = 0$, this corresponds to an α, the shape parameter, of approximately 10. For $\beta = 1$, the parameterless Poisson noise model results in an *integer-valued* noisy matrix $\mathbf{V}$. Since there is no noise parameter to select Poisson noise model, we chose b_{true} so that the elements of the data matrix $\mathbf{V}$ are large enough, resulting in an SNR ≈ 10 dB. For the Gaussian observation model ($\beta = 2$), we can analytically solve for the noise variance σ^2 so that the SNR is approximately 10 dB. In addition, we set the number of columns $N = 100$, the initial number of components $K = 2\,K_{\text{true}} = 10$, and chose two different values for F, namely, 50 and 500. The threshold value τ is set to 10^{-7} (refer to Section 4.4). It was observed using this value of the threshold that the iterates of λ_k converged to their limiting values. We ran ℓ_1- and ℓ_2-ARD for $a \in \{5, 10, 25, 50, 100, 250, 500\}$ and using b computed as in Section 4.5.2. The dispersion parameter ϕ is assumed known and set as in the discussion after (18).

To make fair comparisons, the data and the initializations are the same for ℓ_2- and ℓ_1-ARD as well as for every (β, a). We averaged the inferred model order K_{eff} over 10 different runs. The results are displayed in Fig. 1.

First, we observe that ℓ_1-ARD recovers the model order $K_{\text{true}} = 5$ correctly when $a \leq 100$ and $\beta \in \{0, 1, 2\}$. This range includes $a_{\text{true}} = 50$, which is the true hyperparameter we generated the data from. Thus, if we use the correct range of values of a and if the SNR is of the order 10 dB (which is reasonable in most applications), we are able to recover the true model order from the data. On the other hand, from the top right and bottom right plots, we see that ℓ_2-ARD is not as robust in recovering the right latent dimensionality.

Second, note that the quality of estimation is relatively consistent across various βs. The success of the proposed algorithms hinges more on the amount of noise added (i.e., the SNR) compared to which specific β is assumed. However, as discussed in Section 3.2, the shape parameter β should be chosen to reflect our belief in the underlying generative model and the noise statistics.

Third, observe that when more data are available ($F = 500$), the estimation quality improves significantly. This is evidenced by the fact that even ℓ_2-ARD (bottom right plot) performs much better—it selects the right model order for all $a \leq 25$ and $\beta \in \{1, 2\}$. The estimates are also much more consistent across various initializations. Indeed the standard deviations for most sets of experiments is zero, demonstrating that there is little or no variability due to random initializations.

Fig. 2. Sample images of the noisy `swimmer` data. The colormap is adjusted such that black corresponds to the smallest data coefficient value ($v_{fn} = 0$) and white the largest ($v_{fn} = 24$).

6.2 Simulations with the `swimmer` Dataset

In this section, we report experiments on the `swimmer` dataset introduced in [37]. This is a synthetic dataset of $N = 256$ images each of size $F = 32 \times 32 = 1{,}024$. Each image represents a swimmer composed of an invariant torso and four limbs, where each limb can take one of four positions. We set background pixel values to 1 and body pixel values to 10, and generated noisy data with Poisson noise. Sample images of the resulting noisy data are shown in Fig. 2. The "ground truth" number of components for this dataset is $K_{\text{true}} = 16$, which corresponds to all the different limb positions. The torso and background form an invariant component that can be associated with any of the four limbs, or equally split among limbs. The data images are vectorized and arranged in the columns of $\mathbf{V}$.

We applied ℓ_1- and ℓ_2-ARD with $\beta = 1$ (KL-divergence, matching the Poisson noise assumption, and thus $\phi = 1$), $K = 32 = 2\,K_{\text{true}}$ and $\tau = 10^{-6}$. We tried several values for the hyperparameter a, namely, $a \in \{5, 10, 25, 50, 75, 100, 250, 500, 750, 1{,}000\}$, and set b according to (38). For every value of a we ran the algorithms from 10 common positive random initializations. The regularization paths returned by the two algorithms are displayed in Fig. 3. ℓ_1-ARD consistently estimates the correct number of components ($K_{\text{true}} = 16$) up to $a = 500$. Fig. 4 displays the learned basis, objective function, and relevance parameters along iterations

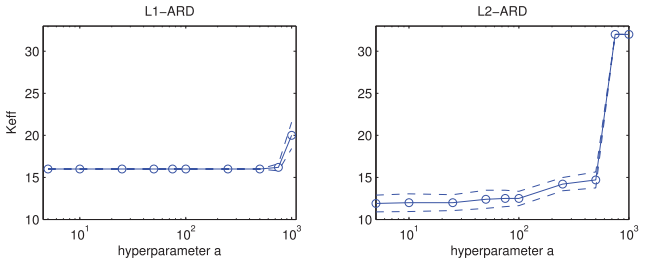

Fig. 3. Estimated number of components K_{eff} as a function of a for ℓ_1- and ℓ_2-ARD. The plain line is the average value of K_{eff} over the 10 runs and dashed lines display $\pm$ the standard deviation.

in one run of ℓ_1-ARD when $a = 100$. It can be seen that the ground truth is perfectly recovered.

In contrast to ℓ_1-ARD, Fig. 3 shows that the value of K_{eff} returned by ℓ_2-ARD is more variable across runs and values of a. Manual inspection reveals that some runs return the correct decomposition when $a = 500$ (and those are the runs with the lowest end value of the objective function, indicating the presence of apparent local minima), but far less consistently than ℓ_1-ARD. Then it might appear that the decomposition strongly overfits the noise for $a \in \{750, 1{,}000\}$. However, visual inspection of learned dictionaries with these values shows that the solutions still make sense. As such, Fig. 5 displays the dictionary learned by ℓ_2-ARD with $a = 1{,}000$. The figure shows that the hierarchy of the decomposition is preserved, despite the fact that the last

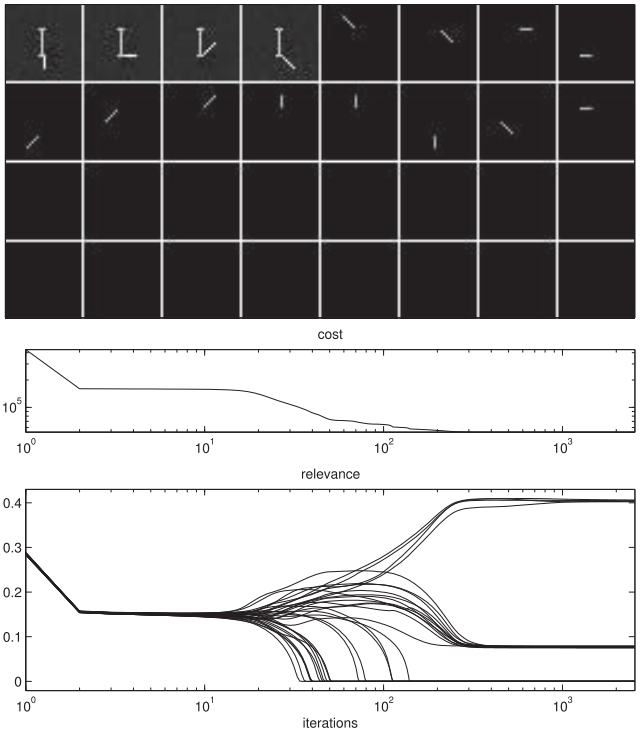

Fig. 4. Top: Dictionary learned in one run of ℓ_1-ARD with $a = 100$. The dictionary elements are presented left to right, top to bottom, by descending order of their relevance λ_k. For improved visualization and fair comparison of the relative importance of the dictionary elements, we display $\mathbf{w}_k$ rescaled by the expectation of h_{kn}, i.e., for ℓ_1-ARD, $\lambda_k \mathbf{w}_k$. The figure colormap is then adjusted to fit the full range of values taken by $\mathbf{W}\,\mathrm{diag}\,\boldsymbol{\lambda}$. Middle: Values of the objective function (21) along iterations (log-log scale). Bottom: Values of $\lambda_k - B$ along iterations (log-linear scale).

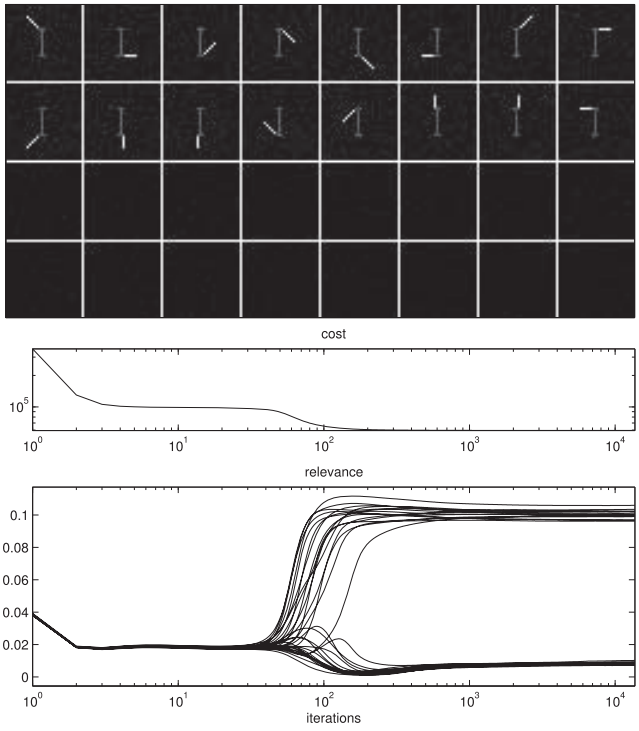

Fig. 5. Top: Dictionary learned by ℓ_2-ARD with $a = 1{,}000$. The dictionary is displayed using the same convention as in Fig. 4, except that the vectors $\mathbf{w}_k$ are now rescaled by the expectation of h_{kn} under the Half-Normal prior, i.e., $(2\lambda_k/\pi)^{1/2}$. Middle: Values of the cost function (21) along iterations (log-log scale). Bottom: Values of $\lambda_k - B$ along iterations (log-linear scale).

16 components capture some residual noise, as a closer inspection would reveal. Thus, despite that fact that pruning is not fully achieved in the 16 extra components, the relevance parameters still give a valid interpretation of what the most significant components are. Fig. 5 shows the evolution of relevance parameters along iterations and it can be seen that the 16 "spurious" components approach the lower bound in the early iterations before they start to fit noise. Note that ℓ_2-ARD returns a solution where the torso is equally shared by the four limbs. This is because the ℓ_2 penalization favors this particular solution over the one returned by ℓ_1-ARD, which favors sparsity of the individual dictionary elements.

With $\tau = 10^{-6}$, the average number of iterations for convergence is approximately $4{,}000 \pm 2{,}000$ for ℓ_1-ARD for all a. The average number of iterations for ℓ_2-ARD is of the same order for $a \leq 500$, and increases to more than 10,000 iterations for $a \geq 750$ because all components are active for these as.

6.3 Music Decomposition

We now consider a music signal decomposition example and illustrate the benefits of ARD in NMF with the IS divergence ($\beta = 0$). Févotte et al. [3] showed that IS-NMF of the power spectrogram underlies a generative statistical model of superimposed Gaussian components, which is relevant to the representation of audio signals. As explained in Sections 3.2 and 4.5, this model is also equivalent to assuming that the power spectrogram is observed in multiplicative exponential noise, i.e., setting $\phi = 1/\alpha = 1$. We investigate the decomposition of the short piano sequence used in [3], a monophonic 15 seconds-long signal x_t recorded in real conditions. The

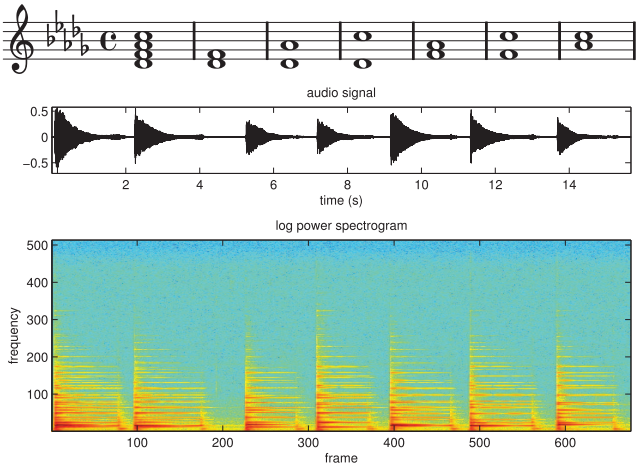

Fig. 6. Three representations of data: Top: original score, middle: time-domain recorded signal, bottom: log-power spectrogram.

sequence is composed of four piano notes, played all at once in the first measure and then played by pairs in all possible combinations in the subsequent measures. The STFT x_{fn} of the temporal data x_t was computed using a sinebell analysis window of length $L = 1{,}024$ (46 ms) with 50 percent overlap between two adjacent frames, leading to $N = 674$ frames and $F = 513$ frequency bins. The musical score, temporal signal, and log-power spectrogram are shown in Fig. 6. In [3], it was shown that IS-NMF of the power spectrogram $v_{fn} = |x_{fn}|^2$ can correctly separate the spectra of the different notes and other constituents of the signal (sound of hammer on the strings, sound of sustain pedal, etc.).

We set $K = 18$ (three times the "ground truth" number of components) and ran ℓ_2-ARD with $\beta = 0$, $a = 5$, and b computed according to (38). We ran the algorithm from 10 random initializations and selected the solution returned with the lowest final cost. For comparison, we ran standard nonpenalized Itakura-Saito NMF using the multiplicative algorithm described in [3], equivalent to ℓ_2-ARD with $\lambda_k \to \infty$ and $\gamma(\beta) = 1$. We ran IS-NMF 10 times with the same random initializations we used for ARD IS-NMF, and selected the solution with minimum fit. Additionally, we ran the methods by Mørup and Hansen (with KL-divergence) [18] and Hoffman et al. [21]. We used Matlab implementations either publicly available [21] or provided to us by Mørup and Hansen [18]. The best among 10 runs of these methods was selected.

Given an approximate factorization $\mathbf{WH}$ of the data spectrogram $\mathbf{V}$ returned by any of the four algorithms, we proceeded to reconstruct time-domain components by Wiener filtering, following [3]. The STFT estimate $\hat{c}_{k,fn}$ of component k is reconstructed by

$$\hat{c}_{k,fn} = \frac{w_{fk}h_{kn}}{\sum_j w_{fj}h_{jn}} x_{fn}, \tag{40}$$

and the STFT is inverted to produce the temporal component $\hat{c}_{k,t}$.[2] By linearity of the reconstruction and inversion, the decomposition is conservative, i.e., $x_t = \sum_k \hat{c}_{k,t}$.

2. With the approach of Hoffman et al. [21], the columns of $\mathbf{W}$ have to be multiplied by their corresponding weight parameter θ_k prior to reconstruction.

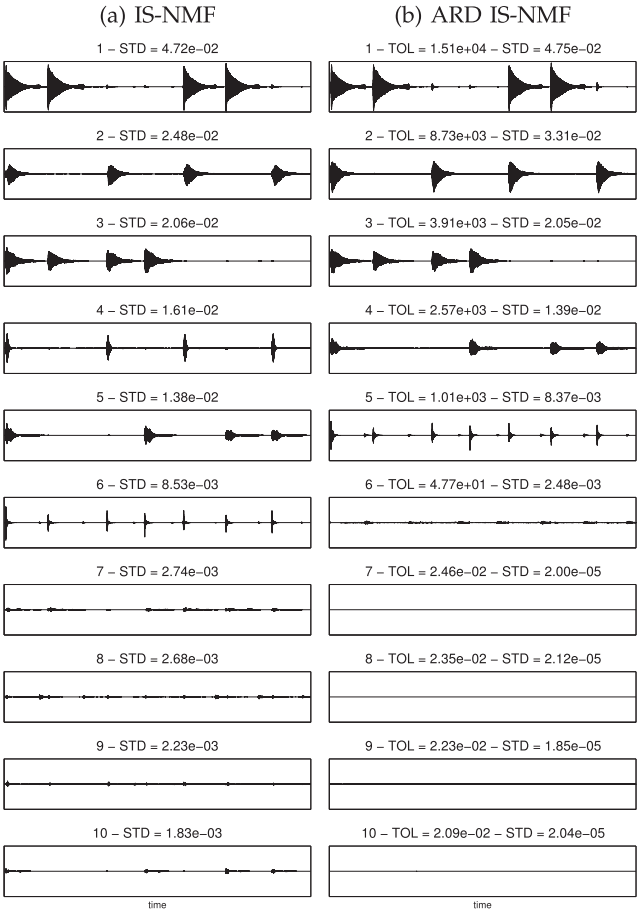

Fig. 7. The first 10 components produced by IS-NMF and ARD IS-NMF. STD denotes the standard deviation of the time samples. TOL is the relevance relative to the bound, i.e., $(\lambda_k - B)/B$. With IS-NMF, the second note of the piece is split into two components ($k = 2$ and $k = 4$).

The components produced by IS-NMF were ordered *by decreasing value of their standard deviations* (computed from the time samples). The components produced by ARD IS-NMF, Mørup and Hansen [18], and Hoffman et al. [21] were ordered *by decreasing value of their relevance weights* ($\{\lambda_k\}$ or $\{\theta_k\}$). Fig. 7 displays the 10 first components produced by IS-NMF and ARD IS-NMF. The y-axes of the two figures are identical so that the component amplitudes are directly comparable. Fig. 8 displays the histograms of the standard deviation values of all 18 components for IS-NMF, ARD IS-NMF, Mørup and Hansen [18], and Hoffman et al. [21].[3]

The histogram in the top right of Fig. 8 indicates that ARD IS-NMF retains six components. This is also confirmed by the value of relative relevance $(\lambda_k - B)/B$ (upon convergence of the relevance weights), displayed with the components in Fig. 7, which drops by a factor of about 2,000 from components 6 to 7. The six components correspond to expected semantic units of the musical sequence: The first four components extract the individual notes and the next two components extract the sound of a hammer hitting the strings and the sound produced by the sustain pedal when it is released. In contrast, IS-NMF has a tendency to overfit; in particular the second note of the piece is split into two

3. The sound files produced by all the approaches are available in the supplementary material, available online. See [33].

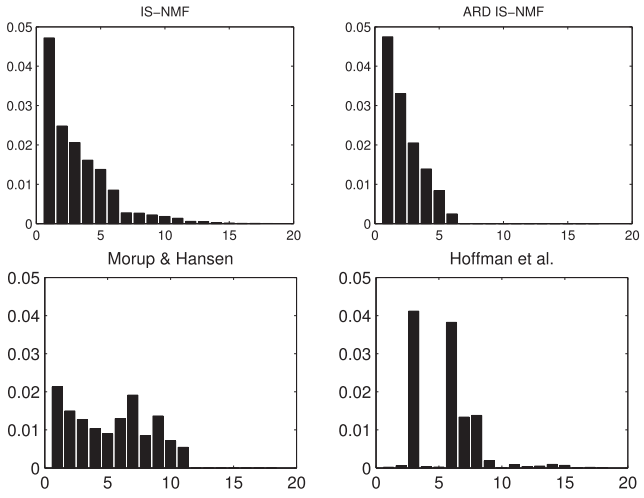

Fig. 8. Histograms of standard deviation values of all 18 components produced by IS-NMF, ARD IS-NMF, Mørup and Hansen [18], and Hoffman et al. [21]. ARD IS-NMF only retains 6 components, which correspond to the expected decomposition, displayed in Fig. 7. On this dataset, the methods proposed in [18] and [21] fail to produce the desired decomposition.

components ($k = 2$ and $k = 4$). The histogram in the bottom left of Fig. 8 shows that the approach of Mørup and Hansen [18] (with the KL-divergence) retains 11 components. Visual inspection of the reconstructed components reveals inaccuracies in the decomposition and significant overfit (some notes are split in subcomponents). The poorness of the results is in part explained by the inadequacy of the KL-divergence (or euclidean distance) for factorization of spectrograms, as discussed in [3]. In contrast, our approach offers flexibility for ARD NMF where the fit-to-data term can be chosen according to the application by setting β to the desired value.

The histogram in the bottom right of Fig. 8 shows that the method by Hoffman et al. [21] retains approximately five components. The decomposition resembles the expected decomposition more closely than [18], except that the hammer attacks are merged with one of the notes. However, it is interesting to note that the distribution of standard deviations does not follow the order of relevance values. This is because the weight parameter θ_k is independent of $\mathbf{W}$ and $\mathbf{H}$ in the prior. As such, the factors are allowed to take very small values while the weight values are not necessarily small.

Finally, we remark that on this data ℓ_1-ARD IS-NMF performed similarly to ℓ_2-ARD IS-NMF and in both cases the retrieved decompositions were fairly robust to the choice of a. We experimented with the same values of a as in previous section and the decompositions and their hierarchies were always found correct. We point out that, as with IS-NMF, initialization is an issue, as other runs did not produce the desired decomposition into notes. However, in our experience the best out of 10 runs always outputs the correct decomposition.

6.4 Prediction of Stock Prices

NMF (with the euclidean and KL costs) has previously been applied on stock data [5] to learn "basis functions" and to cluster companies. In this section, we perform a prediction task on the stock prices of the Dow 30

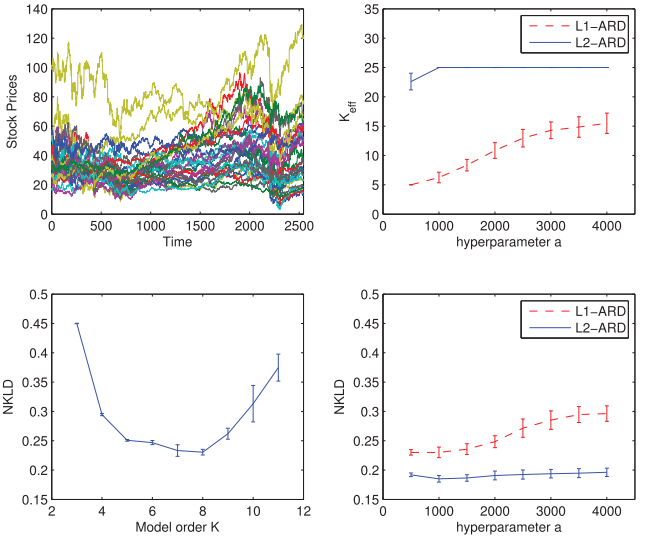

Fig. 9. Top left: The stock data. Top right: Effective model order K_{eff} as a function of a. Bottom: Normalized KL-divergence for KL-NMF (left), ℓ_1- and ℓ_2-ARD KL-NMF (right). Note that the y-axes on both plots are the same. Mørup and Hansen's method [18] yielded an NKLD of 0.37 ± 0.03 (averaged over 10 runs), which is inferior to ℓ_2-ARD, as seen in the bottom right.

companies (comprising the Dow Jones Industrial Average). These are major American companies from various sectors of the economy such as services (e.g., Walmart), consumer goods (e.g., General Motors), and healthcare (e.g., Pfizer). The dataset consists of the stock prices of these $F = 30$ companies from 3 January 2000 to 27 July 2011, a total of $N = 2{,}543$ trading days.[4] The data are displayed in the top left plot of Fig. 9.

In order to test the prediction capabilities of our algorithm, we organized the data into an $F \times N$ matrix $\mathbf{V}$ and removed 50 percent of the entries at random. For the first set of experiments, we performed standard β-NMF with $\beta = 1$, for different values of K, using the observed entries only.[5] We report results for different noninteger values of β in the following. Having performed KL-NMF on the incomplete data, we then estimated the missing entries by multiplying the inferred basis $\mathbf{W}$ and the activation coefficients $\mathbf{H}$ to obtain the estimate $\hat{\mathbf{V}}$. The normalized KL-divergence (NKLD) between the true (missing) stock data and their estimates is then computed as

$$\text{NKLD} \triangleq \frac{1}{|\mathcal{E}|} \sum_{(f,n)\in\mathcal{E}} d_{\text{KL}}(v_{fn}|\hat{v}_{fn}), \tag{41}$$

where $\mathcal{E} \subset \{1,\dots,F\} \times \{1,\dots,N\}$ is the set of missing entries and $d_{\text{KL}}(\,\cdot\,|\,\cdot\,)$ is the KL-divergence ($\beta = 1$). The smaller the NKLD, the better the prediction of the missing stock prices and hence the better the decomposition of $\mathbf{V}$ into $\mathbf{W}$ and $\mathbf{H}$. We then did the same for ℓ_1- and ℓ_2-ARD KL-NMF, for different values of a and using $K = 25$. For

4. Stock prices of the Dow 30 companies are provided at the following link: http://www.optiontradingtips.com/resources/historical-data/dow-jones30.html. The raw data consists of four stock prices per company per day. The mean of the four data points is taken to be the representative of the stock price of that company for that day.

5. Accounting for the missing data involves applying a binary mask to $\mathbf{V}$ and $\mathbf{WH}$, where 0 indicates missing entries [38].

KL-NMF, the criterion for termination is chosen so that it mimics that in Section 4.4. Namely, as is commonly done in the NMF literature, we ensured that the columns of $\mathbf{W}$ are normalized to unity. Then, we computed the *NMF relevance weights* $\lambda_k^{\text{NMF}} \triangleq \frac{1}{2}\|\underline{h}_k\|_2^2$. We terminated the algorithm whenever $\text{tol}^{\text{NMF}} \triangleq \max_k |(\lambda_k^{\text{NMF}} - \tilde{\lambda}_k^{\text{NMF}})/\tilde{\lambda}_k^{\text{NMF}}|$ falls below $\tau = 5 \times 10^{-7}$. We averaged the results over 20 random initializations. The NKLDs and the inferred model orders K_{eff} are displayed in Fig. 9.

In the top right plot of Fig. 9, we observe that there is a general increasing trend; as a increases, the inferred model order K_{eff} also increases. In addition, for the same value of a, ℓ_1-ARD prunes more components than ℓ_2-ARD due to its sparsifying effect. This was also observed for synthetic data and the `swimmer` dataset. However, even though ℓ_2-ARD retains almost all the components, the basis and activation coefficients learned model the underlying data better. This is because ℓ_2 penalization methods result in coefficients that are more dense and are known to be better for prediction (rather than sparsification) tasks.

From the bottom left plot of Fig. 9, we observe that when K is too small, the model is not "rich" enough to model the data and hence the NKLD is large. Conversely, when K is too large, the model overfits the data, resulting in a large NKLD. We also observe that ℓ_2-ARD performs spectacularly across a range of values of the hyperparameter a, uniformly better than standard KL-NMF. The NKLD for estimating the missing stock prices hovers around 0.2, whereas KL-NMF results in an NKLD of more than 0.23 for all K. This shows that ℓ_2-ARD produces a decomposition that is more relevant for modeling missing data. Thus, if one does not know the true model order a priori and chooses to use ℓ_2-ARD with some hyperparameter a, the resulting NKLD would be much better than doing KL-NMF even though many components will be retained. In contrast, ℓ_1-ARD does not perform as spectacularly across all values of a but even when a small number of components is retained (at $a = 500$, $K_{\text{eff}} = 5$, NKLD for ℓ_1-ARD ≈ 0.23, NKLD for KL-NMF ≈ 0.25), it performs significantly better than KL-NMF. It is plausible that the stock data fits the assumptions of the Half-Normal model better than the Exponential model and hence ℓ_2-ARD performs better.

For comparison, we also implemented a version of the method by Mørup and Hansen [18] that handles missing data. The mean NKLD value returned over 10 runs is 0.37 ± 0.03, and thus it is clearly inferior to the methods in this paper. The data does not fit the model well.

Finally, in Fig. 10, we demonstrate the effect of varying the shape parameter β and the dispersion parameter ϕ. The distance between the predicted stock prices and the true ones is measured using the NKLD in (41) and the NEUC (the euclidean analogue of the NKLD). We also computed the NIS (the IS analogue of the NKLD), and noted that the results across all three performance metrics are similar so we omit the NIS. We used l_2-ARD, set $a = 1{,}000$, and calculated b using (38). We also chose integer and noninteger values of β to demonstrate the flexibility of l_2-ARD. It is observed that $\beta = 0.5, \phi = 10$ gives the best NKLD and NEUC and that $1 \leq \beta \leq 1.5$ performs well across a wide range of values of ϕ.

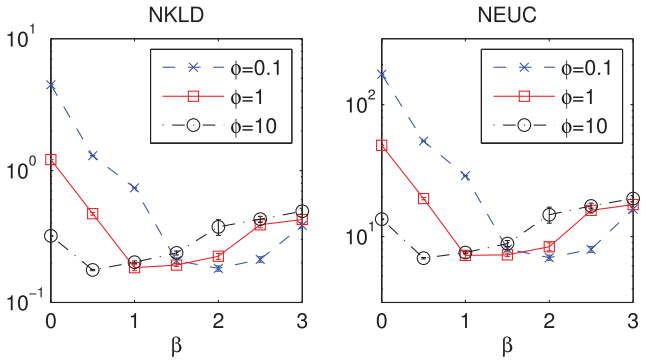

Fig. 10. Effect of varying shape β and dispersion ϕ on prediction performance. Average results over 10 runs.

7 Conclusion

In this paper, we proposed a novel statistical model for β-NMF where the columns of $\mathbf{W}$ and rows $\mathbf{H}$ are tied together through a common scale parameter in their prior, exploiting (and solving) the scale ambiguity between $\mathbf{W}$ and $\mathbf{H}$. MAP estimation reduces to a penalized NMF problem with a group-sparsity inducing regularizing term. A set of MM algorithms accounting for all values of β and either ℓ_1- or ℓ_2-norm group-regularization was presented. They ensure the monotonic decrease of the objective function at each iteration and result in multiplicative update rules of linear complexity in F, K, and N. The updates automatically preserve nonnegativity, given positive initializations, and are easily implemented. The efficiency of our approach was validated on several synthetic and real-world datasets, with competitive performance w.r.t. the state of the art. At the same time, our proposed methods offer improved flexibility over existing approaches (our approach can deal with various types of observation noise and prior structure in a unified framework). Using the method of moments, an effective strategy for the selection of hyperparemeter b given a was proposed and, as a general rule of thumb, we recommend setting a to a small value w.r.t. $F + N$.

There are several avenues for further research: here, we derived a MAP approach that works efficiently, but more sophisticated inference techniques can be envisaged, such as fully Bayesian inference in the model we proposed in Section 3. Following similar treatments in sparse regression [39], [40] or with other forms of matrix factorization [41], one could seek the maximization of $\log p(\mathbf{V}|a, b, \phi)$ using variational or Markov chain Monte-Carlo inference, and in particular handle hyperparameter estimation in a (more) principled way. Other more direct extensions of this work concern the factorization of tensors and online-based methods akin to [42], [43].

Acknowledgments

The authors would like to acknowledge Francis Bach for discussions related to this work, Y. Kenan Yilmaz and A. Taylan Cemgil for discussions on Tweedie distributions, as well as Morten Mørup and Matt Hoffman for sharing their code. They would also like to thank the reviewers whose comments helped to greatly improve the paper. The work of V.Y.F. Tan is supported by A*STAR, Singapore. The

work of C. Févotte is supported by project ANR-09-JCJC-0073-01 TANGERINE (theory and applications of nonnegative matrix factorization).

REFERENCES

[1] P. Paatero and U. Tapper, "Positive Matrix Factorization: A Non-Negative Factor Model with Optimal Utilization of Error Estimates of Data Values," *Environmetrics,* vol. 5, pp. 111-126, 1994.

[2] D.D. Lee and H.S. Seung, "Learning the Parts of Objects with Nonnegative Matrix Factorization," *Nature,* vol. 401, pp. 788-791, 1999.

[3] C. Févotte, N. Bertin, and J.-L. Durrieu, "Nonnegative Matrix Factorization with the Itakura-Saito Divergence: With Application to Music Analysis," *Neural Computation,* vol. 21, pp. 793-830, Mar. 2009.

[4] D. Guillamet, B. Schiele, and J. Vitri, "Analyzing Non-Negative Matrix Factorization for Image Classification," *Proc. Int'l Conf. Pattern Recognition,* 2002.

[5] K. Drakakis, S. Rickard, R. de Frein, and A. Cichocki, "Analysis of Financial Data Using Non-Negative Matrix Factorization," *Int'l J. Math. Sciences,* vol. 6, June 2007.

[6] Y. Gao and G. Church, "Improving Molecular Cancer Class Discovery through Sparse Non-Negative Matrix Factorization," *Bioinformatics,* vol. 21, pp. 3970-3975, 2005.

[7] A. Cichocki, R. Zdunek, and S. Amari, "Csiszar's Divergences for Non-Negative Matrix Factorization: Family of New Algorithms," *Proc. Sixth Int'l Conf. Independent Component Analysis and Blind Signal Separation,* pp. 32-39, Mar. 2006.

[8] M. Nakano, H. Kameoka, J. Le Roux, Y. Kitano, N. Ono, and S. Sagayama, "Convergence-Guaranteed Multiplicative Algorithms for Non-Negative Matrix Factorization with Beta-Divergence," *Proc. IEEE Int'l Workshop Machine Learning for Signal Processing,* Sept. 2010.

[9] C. Févotte and J. Idier, "Algorithms for Nonnegative Matrix Factorization with the Beta-Divergence," *Neural Computation,* vol. 23, pp. 2421-2456, Sept. 2011.

[10] A. Cichocki, S. Cruces, and S. Amari, "Generalized Alpha-Beta Divergences and Their Application to Robust Nonnegative Matrix Factorization," *Entropy,* vol. 13, pp. 134-170, 2011.

[11] G. Schwarz, "Estimating the Dimension of a Model," *Annals of Statistics,* vol. 6, pp. 461-464, 1978.

[12] D.J.C. Mackay, "Probable Networks and Plausible Predictions—A Review of Practical Bayesian Models for Supervised Neural Networks," *Network: Computation in Neural Systems,* vol. 6, no. 3, pp. 469-505, 1995.

[13] C.M. Bishop, "Bayesian PCA," *Advances in Neural Information Processing Systems,* pp. 382-388, 1999.

[14] A.T. Cemgil, "Bayesian Inference for Nonnegative Matrix Factorisation Models," *Computational Intelligence and Neuroscience,* vol. 2009, Article ID 785152, p. 17, 2009, doi:10.1155/2009/785152.

[15] M.N. Schmidt, O. Winther, and L.K. Hansen, "Bayesian Non-Negative Matrix Factorization," *Proc. Eighth Int'l Conf. Independent Component Analysis and Signal Separation,* Mar. 2009.

[16] M. Zhong and M. Girolami, "Reversible Jump MCMC for Non-Negative Matrix Factorization," *Proc. Int'l Conf. Artificial Intelligence and Statistics,* p. 8, 2009.

[17] M.N. Schmidt and M. Mørup, "Infinite Non-Negative Matrix Factorizations," *Proc. European Signal Processing Conf.,* 2010.

[18] M. Mørup and L.K. Hansen, "Tuning Pruning in Sparse Non-Negative Matrix Factorization," *Proc. 17th European Signal Processing Conf.,* Aug. 2009.

[19] M. Mørup and L.K. Hansen, "Automatic Relevance Determination for Multiway Models," *J. Chemometrics,* vol. 23, nos. 7/8, pp. 352-363, 2009.

[20] Z. Yang, Z. Zhu, and E. Oja, "Automatic Rank Determination in Projective Nonnegative Matrix Factorization," *Proc. Ninth Int'l Conf. Latent Variable Analysis and Signal Separation,* pp. 514-521, 2010.

[21] M.D. Hoffman, D.M. Blei, and P.R. Cook, "Bayesian Nonparametric Matrix Factorization for Recorded Music," *Proc. Int'l Conf. Machine Learning,* 2010.

[22] V.Y.F. Tan and C. Févotte, "Automatic Relevance Determination in Nonnegative Matrix Factorization," *Proc. Workshop Signal Processing with Adaptative Sparse Structured Representations,* Apr. 2009.

[23] A. Basu, I.R. Harris, N.L. Hjort, and M.C. Jones, "Robust and Efficient Estimation by Minimising a Density Power Divergence," *Biometrika,* vol. 85, pp. 549-559, Sept. 1998.

[24] S. Eguchi and Y. Kano, "Robustifying Maximum Likelihood Estimation," technical report, Inst. of Statistical Math., Research Memo. 802, June 2001.

[25] M. Tweedie, "An Index which Distinguishes between Some Important Exponential Families," *Proc. Indian Statistical Inst. of Golden Jubilee Int'l Conf.,* pp. 579-604, 1984.

[26] D.R. Hunter and K. Lange, "A Tutorial on MM Algorithms," *The Am. Statistician,* vol. 58, pp. 30-37, 2004.

[27] 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.

[28] Y.K. Yilmaz, "Generalized Tensor Factorization," PhD thesis, Boğaziçi Univ., 2012.

[29] B. Jørgensen, "Exponential Dispersion Models," *J. Royal Statistical Soc. Series B (Methodological),* vol. 49, no. 2, p. 127162, 1987.

[30] M. Yuan and Y. Lin, "Model Selection and Estimation in Regression with Grouped Variables," *J. Royal Statistical Soc., Series B,* vol. 68, no. 1, pp. 49-67, 2007.

[31] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski, "Optimization with Sparsity-Inducing Penalties," *Foundations and Trends in Machine Learning,* vol. 4, no. 1, pp. 1-106, 2012.

[32] E.J. Candès, M.B. Wakin, and S.P. Boyd, "Enhancing Sparsity by Reweighted ℓ_1 Minimization," *J. Fourier Analysis and Applications,* vol. 14, pp. 877-905, Dec. 2008.

[33] V.Y.F. Tan and C. Févotte, "Supplementary Material for 'Automatic Relevance Determination in Nonnegative Matrix Factorization with the β-Divergence'," http://doi.ieeecomputersociety.org/10.1109/TPAMI.2012.240, 2012.

[34] Z. Yang and E. Oja, "Unified Development of Multiplicative Algorithms for Linear and Quadratic Nonnegative Matrix Factorization," *IEEE Trans. Neural Networks,* vol. 22, no. 12, pp. 1878-1891, Dec. 2011.

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

[36] J. Eggert and E. Körner, "Sparse Coding and NMF," *Proc. IEEE Int'l Joint Conf. Neural Networks,* pp. 2529-2533, 2004.

[37] D. Donoho and V. Stodden, "When Does Non-Negative Matrix Factorization Give a Correct Decomposition into Parts?" *Proc. Advances in Neural Information Processing Systems Conf.,* 2004.

[38] N.-D. Ho, "Nonnegative Matrix Factorization Algorithms and Applications," PhD thesis, Universiteit Katholique de Louvain, 2008.

[39] M.E. Tipping, "Sparse Bayesian Learning and the Relevance Vector Machine," *J. Machine Learning Research,* vol. 1, pp. 211-244, 2001.

[40] D.P. Wipf, B.D. Rao, and S. Nagarajan, "Latent Variable Bayesian Models for Promoting Sparsity," *IEEE Trans. Information Theory,* vol. 57, no. 9, pp. 6236-55, Sept. 2011.

[41] R. Salakhutdinov and A. Mnih, "Probabilistic Matrix Factorization," *Proc. Advances in Neural Information Processing Systems Conf.,* vol. 19, 2007.

[42] A. Lefèvre, F. Bach, and C. Févotte, "Online Algorithms for Nonnegative Matrix Factorization with the Itakura-Saito Divergence," *Proc. IEEE Workshop Applications of Signal Processing to Audio and Acoustics,* Oct. 2011.

[43] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online Learning for Matrix Factorization and Sparse Coding," *J. Machine Learning Research,* vol. 11, pp. 10-60, 2010.

Vincent Y.F. Tan received the BA and MEng degrees in electrical and information sciences tripos (EIST) from the University of Cambridge in 2005 and the PhD degree in electrical engineering and computer science from MIT in 2011, after which he was a postdoctoral researcher at the University of Wisconsin-Madison. He is now a scientist at the Institute for Infocomm Research (I^2R), Singapore, and an adjunct assistant professor in the Department of Electrical and Computer Engineering at the National University of Singapore. During his PhD, he held two summer research internships at Microsoft Research. His research interests include learning and inference in graphical models, statistical signal processing, and network information theory. He received the Charles Lamb prize, a Cambridge University Engineering Department prize awarded to the student who demonstrates the greatest proficiency in the EIST. He also received the MIT EECS Jin-Au Kong outstanding doctoral thesis prize and the A*STAR Philip Yeo prize for outstanding achievements in research. He is a member of the IEEE and of the IEEE "Machine Learning for Signal Processing" technical committee.

Cédric Févotte received the state engineering and PhD degrees in control and computer science from the École Centrale de Nantes, France, in 2000 and 2003, respectively. During his PhD, he was with the Signal Processing Group at the Institut de Recherche en Communication et Cybernétique de Nantes (IRCCyN). From 2003 to 2006, he was a research associate with the Signal Processing Laboratory at the University of Cambridge (Engineering Department). He was then a research engineer with the music editing technology start-up company Mist-Technologies (now Audionamix) in Paris. In 2007, he became a CNRS tenured researcher. He was affiliated with LTCI (CNRS & Télécom ParisTech) from 2007 to 2012. Since 2013, he has been with Laboratoire Lagrangre (CNRS, Observatoire de la Côte d'Azur & Université de Nice Sophia Antipolis). His research interests generally concern statistical signal processing and unsupervised machine learning and, in particular, applications to blind source separation and audio signal processing. He is the scientific leader of project TANGERINE (Theory and applications of nonnegative matrix factorization) funded by the French research funding agency ANR and a member of the IEEE and of the IEEE "Machine Learning for Signal Processing" technical committee.