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

Vincent Y. F. Tan
University of Wisconsin-Madison
Madison, WI 53706
vtan@wisc.edu

Cédric Févotte
CNRS LTCI, TELECOM ParisTech,
75014 Paris, France
fevotte@telecom-paristech.fr

Abstract

This paper addresses the problem of estimating the latent dimensionality in nonnegative matrix fatorization (NMF) via automatic relevance determination (ARD). Uncovering the latent dimensionality is necessary for striking the right balance between data fidelity and overfitting. We propose a Bayesian model for NMF and two algorithms known as ℓ_1- and ℓ_2-ARD, each assuming different priors on the basis and the coefficients. The proposed algorithms leverage on the recent algorithmic advances in NMF with the β-divergence using majorization-minimization (MM) methods. We show by using auxiliary functions that the cost function decreases monotonically to a local minimum. We demonstrate the efficacy and robustness of our algorithms by performing experiments on the `swimmer` dataset.

1 Introduction

Given a nonnegative data matrix $\mathbf{V}$ of dimensions $F \times N$ with nonnegative entries, nonnegative matrix factorization (NMF) refers to the problem of finding a factorization $\mathbf{V} \approx \hat{\mathbf{V}} \triangleq \mathbf{WH}$ where $\mathbf{W}$ and $\mathbf{H}$ are nonnegative matrices of dimensions $F \times K$ and $K \times N$, respectively. The dimension K is usually chosen such that $F\,K + K\,N \ll F\,N$. The factorization is usually sought after through the minimization problem of a cost function $D(\mathbf{V}|\mathbf{WH})$ subject to the nonnegativity constraints $\mathbf{W}, \mathbf{H} \geq 0$. The distance (or divergence or distortion) function $D(\mathbf{V}|\mathbf{WH})$ is a separable measure of fit such that $D(\mathbf{V}|\mathbf{WH}) = \sum_f \sum_n d([\mathbf{V}]_{fn} \,|\, [\mathbf{WH}]_{fn})$ where $d(x|y)$ is a nonnegative scalar cost function of $y \in \mathbb{R}_+$ given $x \in \mathbb{R}_+$, with a single minimum when $x = y$. Without loss of generality, we assume that $d(x|y) = 0$ iff $x = y$. We will consider the $d(x|y)$ to be the so-called β-divergence, a family of cost functions parametrized by a single scalar shape parameter $\beta \in \mathbb{R}$ [1].

In many applications, it is crucial that "right" model order K is selected to balance between data fit and overfitting. We propose a Bayesian model for β-NMF based on automatic relevance determination (ARD) [2] to derive *computationally efficient* algorithms with *monotonicity* guarantees to select K. At the same time, we estimate the basis $\mathbf{W}$ and the activation coefficients $\mathbf{H}$. The proposed algorithms are based on surrogate auxiliary functions (a local majorization of the cost function). These auxiliary functions lead to majorization-minimization (MM) algorithms, which then result in efficient multiplicative updates. The monotonicity of the cost function can be proven by leveraging on techniques in [1]. This paper represents a significant extension of our previous work in [3]. Firstly, the cost function in [3] was restricted to be the Kullback-Leibler (KL) divergence. In this paper, we consider a continuum of costs parameterized by a shape parameter β. Secondly, the algorithms described herein are such that the cost function *monotonically decreases* to a local minimum.

2 Model and Inference

We are inspired by the use of ARD in Bayesian PCA [2] where each element of $\mathbf{W}$ is assigned a Gaussian prior. However, our formulation has two main differences vis-à-vis Bayesian PCA. Firstly,

there are no nonnegativity constraints in Bayesian PCA. Secondly, in Bayesian PCA, thanks to the simplicity of the statistical model (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 $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_K)$ is the vector of relevance weights. We have to maintain the nonnegativity of the elements in $\mathbf{W}$ and $\mathbf{H}$ and, in our case, the activation matrix $\mathbf{H}$ cannot be integrated out analytically.

To ameliorate the abovementioned problems, we propose to *tie* the columns of $\mathbf{W}$ and the rows of $\mathbf{H}$ together through *common* scale parameters and subsequently prune these columns and rows out of the model. This construction is not overconstraining the scales of $\mathbf{W}$ and $\mathbf{H}$, because of the inherent scale indeterminacy between $\mathbf{w}_k$ and h_k. Mørup and Hansen in [4] considered the $\beta = 1, 2$ models and pruned only the rows $\mathbf{H}$ (via an ℓ_1 penalty) but the corresponding columns of $\mathbf{W}$ may take any nonnegative value, which makes the problem ill-posed. In contrast, in our approach $\mathbf{w}_k$ and h_k are *jointly* driven to zero. We choose nonnegative priors for $\mathbf{W}$ and $\mathbf{H}$ to ensure that all elements of the basis and activation matrices are nonnegative. More precisely, we adopt a maximum a-posteriori (MAP) Bayesian approach and assign $\mathbf{W}$ and $\mathbf{H}$ Half-Normal or Exponential priors. When $\mathbf{W}$ and $\mathbf{H}$ have Half-Normal priors,

$$p(w_{fk}|\lambda_k) = \mathcal{HN}(w_{fk}|\lambda_k), \qquad p(h_{kn}|\lambda_k) = \mathcal{HN}(w_{kn}|\lambda_k), \tag{1}$$

where $\mathcal{HN}(\,\cdot\,|\lambda_k)$ is the Half-Normal prior with variance-like parameter λ_k. Note that if x is a Gaussian then $|x|$ is a Half-Normal. When $\mathbf{W}$ and $\mathbf{H}$ have Exponential priors,

$$p(w_{fk}|\lambda_k) = \mathcal{E}(w_{fk}|\lambda_k), \qquad p(h_{kn}|\lambda_k) = \mathcal{E}(w_{kn}|\lambda_k). \tag{2}$$

Note from (1) and (2) that the k^{th} column of $\mathbf{W}$ and the k^{th} row of $\mathbf{H}$ are tied together by a *common* variance-like parameter λ_k. We refer to these λ_k's as the *relevance parameters*. When a particular λ_k is small, that particular column of $\mathbf{W}$ *and* row of $\mathbf{H}$ are not relevant. When a row and a column are not relevant, their norms are close to zero and thus can be removed from the factorization. This removal of *common* rows and columns makes the model more parsimonious.

We now describe the likelihood function. When $\beta = 0, 1, 2$, $D_\beta(\mathbf{V}|\mathbf{WH})$ is proportional to the (negative) log-likelihood of the Itakara-Saito (IS), KL and Euclidean noise models [5] up to a constant. More precisely, the noise models are given as: $\beta = 0 : v_{fn} \sim \mathcal{G}(v_{kn}; s, \hat{v}_{fn}/s)$, $\beta = 1 : v_{fn} \sim \mathcal{P}(v_{kn}; \hat{v}_{fn})$ and $\beta = 2 : v_{fn} \sim \mathcal{N}(v_{kn}; \hat{v}_{fn}, \sigma^2)$ where $\mathcal{G}, \mathcal{P}$ and $\mathcal{N}$ refer to the Gamma, Poisson and Gaussian. Hence, for these three special cases $-\log p(\mathbf{V}|\mathbf{W}, \mathbf{H}) = \vartheta D_\beta(\mathbf{V}|\mathbf{WH}) + \text{cst}$ where the proportionality constant $\vartheta \triangleq s$ for $\beta = 0$, $\vartheta \triangleq 1$ for $\beta = 1$ and $\vartheta \triangleq 1/\sigma^2$ for $\beta = 2$. We extend the use of $D_\beta(\mathbf{V}|\mathbf{WH})$ for all β according to d_β defined in [1].

We further impose an inverse-Gamma prior on each relevance parameter $\lambda_k \sim \mathcal{IG}(\lambda_k|a, b)$, where a and b are the (nonnegative) shape and scale hyperparameters respectively. We set a and b to be constant for all $k = 1, \ldots, K$. It can be shown that the cost function $C(\mathbf{W}, \mathbf{H}, \boldsymbol{\lambda})$ which is equal to the negative log-posterior $-\log p(\mathbf{W}, \mathbf{H}, \boldsymbol{\lambda}|\mathbf{V})$ and is to be minimized, can be written as

$$C(\mathbf{W}, \mathbf{H}, \boldsymbol{\lambda}) = \vartheta\, D_\beta(\mathbf{V}|\mathbf{WH}) + \sum_k \lambda_k^{-1} \left(f(\mathbf{w}_k) + f(h_k) + b\right) + c \log \lambda_k + \text{cst}, \tag{3}$$

and for the two models (i) Half Normal $f(\mathbf{x}) \triangleq \|\mathbf{x}\|_2^2/2$ and $c \triangleq (F + N)/2 + a + 1$, (ii) Exponential $f(\mathbf{x}) \triangleq \|\mathbf{x}\|_1$ and $c \triangleq F + N + a + 1$. The function $f(\,\cdot\,)$ is termed the *regularizer*. Note that in the regularized cost function in (3), 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_k's will be forced to a lower bound while others tend to a larger value. This serves the purpose of pruning irrelevant components out of the model. This is also related to group LASSO [6] and reweighted ℓ_1-minimization [7].

We now sketch our algorithms termed ℓ_2- and ℓ_1-ARD for optimizing $\mathbf{H}$ given $\mathbf{W}$. These algorithms correspond to assuming Half Normal and Exponential priors respectively. Updating $\mathbf{W}$ given $\mathbf{H}$ proceeds analogously. We use an iterative algorithm that sequentially updates one factor w.r.t. the other.

Our algorithms are based on the optimization of an *auxiliary function* $G(\mathbf{H}|\tilde{\mathbf{H}})$ which is a function that is parametrized by the previous iterate $\tilde{\mathbf{H}}$ and majorizes the objective $C(\mathbf{H})$ in (3). Furthermore, $G(\mathbf{H}|\mathbf{H}) = C(\mathbf{H})$. This auxiliary function is derived by upper bounding the convex part of C by using Jensen's inequality and the concave part of C by its tangent. It can be shown by using standard MM techniques that the iterates results in the objective function being non-increasing (just as in the

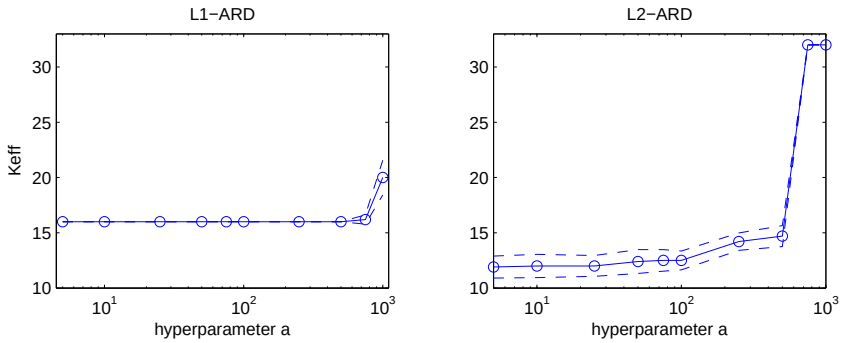

Figure 1: Estimated number of components K_{eff} as a function of a for ℓ_1- and ℓ_2-ARD. The plain line is the average value of K_{eff} over the 10 runs and dashed-lines display $\pm$ the standard deviation.

Expectation-Maximization algorithm). We use such an MM procedure as well as the *moving-term* technique described by Yang and Oja in [8] to derive both ℓ_2- and ℓ_1-ARD. Eventually, we obtain the following updates for h_{kn} given the previous iterate $\tilde{h}_{kn}$: For ℓ_2-ARD:

$$h_{kn} = \tilde{h}_{kn} \left(\frac{p_{kn}}{q_{kn} + \tilde{h}_{kn}/(\vartheta\, \lambda_k)} \right)^{\xi(\beta)}, \quad p_{kn} \triangleq \sum_f w_{fk} v_{fn} \tilde{v}_{fn}^{\beta-2}, \quad q_{kn} \triangleq \sum_f w_{fk} \tilde{v}_{fn}^{\beta-1}, \quad (4)$$

and $\tilde{v}_{fn} \triangleq [\mathbf{W}\tilde{\mathbf{H}}]_{fn}$ and the exponent $\xi(\beta) = 1/(3-\beta)$ for $\beta \leq 2$ and $\xi(\beta) = 1/(\beta-1)$ for $\beta > 2$. For ℓ_1-ARD:

$$h_{kn} = \tilde{h}_{kn} \left(\frac{p_{kn}}{q_{kn} + 1/(\vartheta\, \lambda_k)} \right)^{\gamma(\beta)}, \quad (5)$$

where the exponent $\gamma(\beta) = 1/(2-\beta)$ for $\beta < 1$, $\gamma(\beta) = 1$ for $\beta \in [1, 2]$ and $\gamma(\beta) = 1/(\beta-1)$ for $\beta \geq 2$. It should be mentioned that the updates in (4) and (5) are similar to MM in the usual β-NMF up to an additional term in the denominator; $\tilde{h}_{kn}/\lambda_k$ for ℓ_2-ARD and $1/\lambda_k$ for ℓ_1-ARD.

To update the λ_k's, we find the partial derivative of $C(\mathbf{W}, \mathbf{H}, \boldsymbol{\lambda})$ w.r.t. λ_k, then set it to zero. This gives the update $\lambda_k \leftarrow [f(\mathbf{w}_k) + f(h_k) + b]/c$. Notice that $\lambda_k \geq b/c = B$. This allows us to estimate the effective number of components $K_{\text{eff}} = |\{k \in [K] : (\lambda_k - B)/B \geq \tau\}|$. The algorithm is terminated when $\max_k |(\lambda_k - \tilde{\lambda}_k)/\tilde{\lambda}_k|$ falls below a pre-specified threshold τ.

Despite being fully Bayesian, the algorithms are not completely parameter-free. One has to choose the hyperparameters a and b carefully. Roughly, our idea is to equate $\hat{v}_{fn} = \sum_k w_{fk} h_{kn}$ with the empirical mean of elements in the data matrix $\hat{\mu}_{\mathbf{V}}$, then using the method of moments to find the optimal b. In the case of ℓ_2-ARD, it can be shown that $\hat{b} = \pi(a-1)\hat{\mu}_{\mathbf{V}}/(2K)$. In the case of ℓ_1-ARD, $\hat{b} = ((a-1)(a-2)\hat{\mu}_{\mathbf{V}}/K)^{1/2}$. This is the rule we use for choosing b given a fixed a.

3 Numerical Simulations

In this section we report experiments conducted with the popular `swimmer` dataset. It is a synthetic dataset of $N = 256$ images each of size $F = 32 \times 32 = 1024$. 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. 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. We applied ℓ_1- and ℓ_2-ARD with $\beta = 1$ (which matches the Poisson noise assumption), $K = 32 = 2\,K_{\text{true}}$. We tried several values for the hyperparameter $a \in \{5, 10, 25, 50, 75, 100, 250, 500, 750, 1000\}$. 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. 1. ℓ_1-ARD consistently estimates the correct number of components ($K_{\text{true}} = 16$) up to $a = 500$. Fig. 2 displays the basis learnt in one run of ℓ_1- and ℓ_2-ARD when $a = 100$ and it can be seen that the ground truth is perfectly recovered. Values of the cost function and of the relevance parameters along iterations are shown in Fig. 3.

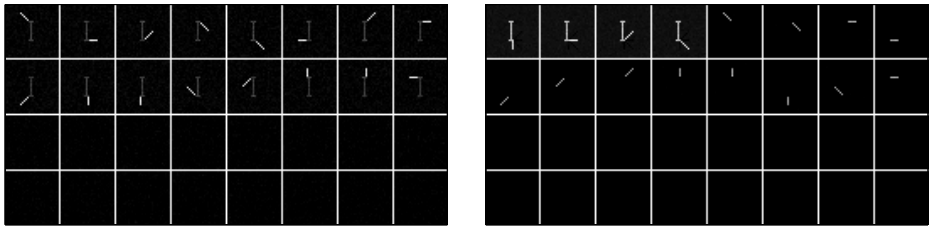

Figure 2: Left: Dictionary learnt in one run of ℓ_2-ARD (Left) and ℓ_1-ARD (Right) with $a = 10^3$.

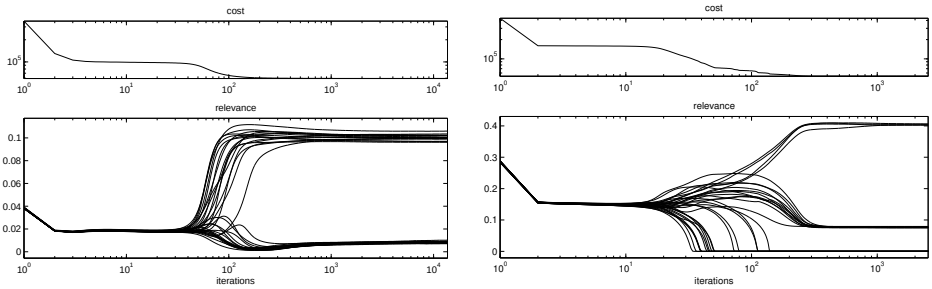

Figure 3: Cost function (3) along iterations (log-log scale). Bottom: Values of $\lambda_k - B$. ℓ_2-ARD and ℓ_1-ARD are plotted on the left and right respectively.

In contrast to ℓ_1-ARD, 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 lowest cost, indicating the presence of local minima), but far less consistently than ℓ_1-ARD. Then it might appear like the decomposition overfits the noise for $a \in \{750, 1000\}$. However, visual inspection of learnt dictionaries with these values show that the solutions are not useless. Fig. 2 displays the dictionary learnt by ℓ_2-ARD with $a = 1000$. The figure shows that the hierarchy of the decomposition is preserved, despite that the last 16 components capture some noise. Thus, despite that pruning is not fully achieved in the 16 extra components, the relevance parameters still give a valid interpretation of what are the most significant components. Fig. 3 shows the evolution of relevance parameters along iterations and it can be seen that the 16 "spurious" components do approach the lower bound in the early iterations before they start to fit the noise. A variant of our approach could consist in stopping to update a λ_k when its relative difference with previous iterate falls under a certain threshold. Note that ℓ_2-ARD returns a solution where the torso is equally shared by the four limbs. This is because ℓ_2 penalization favors this particular solution over the one returned by ℓ_1-ARD, which favors sparsity of the individual dictionary elements. Further results including those on audio datasets using IS-NMF will be presented at the workshop.

References

[1] C. Févotte and J. Idier, "Algorithms for nonnegative matrix factorization with the beta-divergence," *Neural Computation*, vol. 23, Sep. 2011.

[2] C. M. Bishop, "Bayesian PCA," in *Advances in NIPS*, pp. 382–388, 1999.

[3] V. Y. F. Tan and C. Févotte, "Automatic relevance determination in nonnegative matrix factorization," in *SPARS*, (St-Malo, France), Apr. 2009.

[4] M. Mørup and L. K. Hansen, "Tuning pruning in sparse non-negative matrix factorization," in *EUSIPCO*, (Glasgow, Scotland), Aug. 2009.

[5] C. Févotte and A. T. Cemgil, "Nonnegative matrix factorisations as probabilistic inference in composite models," in *EUSIPCO*, pp. 1913–1917, Aug. 2009.

[6] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *Journal of the Royal Statistical Society, Series B*, vol. 68, no. 1, pp. 49–67, 2007.

[7] E. J. Candès, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *Journal of Fourier Analysis and Applications*, vol. 14, pp. 877–905, Dec 2008.

[8] Z. Yang and E. Oja, "Linear and nonlinear projective nonnegative matrix factorization," *IEEE Transactions on Neural Networks*, vol. 21, pp. 734–749, May 2010.