Sparse Additive Matrix Factorization for Robust PCA and Its Generalization

Shinichi Nakajima NAKAJIMA.S@NIKON.CO.JP
Nikon Corporation, Tokyo, 140-8601, Japan

Masashi Sugiyama SUGI@CS.TITECH.AC.JP
Tokyo Institute of Technology, Tokyo 152-8552, Japan

S. Derin Babacan DBABACAN@ILLINOIS.EDU
Beckman Institute, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA

Editor: Steven C.H. Hoi and Wray Buntine

Abstract

Principal component analysis (PCA) can be regarded as approximating a data matrix with a low-rank one by imposing sparsity on its singular values, and its robust variant further captures sparse noise. In this paper, we extend such sparse matrix learning methods, and propose a novel unified framework called *sparse additive matrix factorization* (SAMF). SAMF systematically induces various types of sparsity by the so-called *model-induced regularization* in the Bayesian framework. We propose an iterative algorithm called the *mean update* (MU) for the variational Bayesian approximation to SAMF, which gives the global optimal solution for a large subset of parameters in each step. We demonstrate the usefulness of our method on artificial data and the foreground/background video separation.

Keywords: Variational Bayes, Robust PCA, Matrix Factorization, Sparsity, Model-induced Regulariztion

1. Introduction

Principal component analysis (PCA) (Hotelling, 1933) is a classical method for obtaining low-dimensional expression of data. PCA can be regarded as approximating a data matrix with a low-rank one by imposing sparsity on its singular values. A robust variant of PCA further copes with sparse spiky noise included in observations (Candes et al., 2009; Babacan et al., 2012).

In this paper, we extend the idea of robust PCA, and propose a more general framework called *sparse additive matrix factorization* (SAMF). The proposed SAMF can handle various types of sparse noise such as row-wise and column-wise sparsity, in addition to element-wise sparsity (spiky noise) and low-rank sparsity (low-dimensional expression); furthermore, their arbitrary additive combination is also allowed. In the context of robust PCA, row-wise and column-wise sparsity can capture noise observed when some sensors are broken and their outputs are always unreliable, or some accident disturbs all sensor outputs at a time.

Technically, our approach induces sparsity by the so-called *model-induced regularization* (MIR) (Nakajima and Sugiyama, 2011). MIR is an implicit regularization property of the Bayesian approach, which is based on one-to-many (i.e., redundant) mapping of parameters and outcomes (Watanabe, 2009). In the case of matrix factorization, an observed matrix is

Table 1: Examples of SMF term. See the main text for details.

Factorization	Induced sparsity	K	$(L'^{(k)}, M'^{(k)})$	$\mathcal{X} : (k, l', m') \mapsto (l, m)$
$U = BA^\top$	low-rank	1	(L, M)	$\mathcal{X}(1, l', m') = (l', m')$
$U = \mathit{\Gamma}_E D$	row-wise	L	$(1, M)$	$\mathcal{X}(k, 1, m') = (k, m')$
$U = E\mathit{\Gamma}_D$	column-wise	M	$(L, 1)$	$\mathcal{X}(k, l', 1) = (l', k)$
$U = E * D$	element-wise	$L \times M$	$(1, 1)$	$\mathcal{X}(k, 1, 1) = \mathit{vec\text{-}order}(k)$

decomposed into two redundant matrices, which was shown to induce sparsity in the singular values under the variational Bayesian approximation (Nakajima and Sugiyama, 2011).

We also show that MIR in SAMF can be interpreted as *automatic relevance determination* (ARD) (Neal, 1996), which is a popular Bayesian approach to inducing sparsity. Nevertheless, we argue that the MIR formulation is more preferable since it allows us to derive a practically useful algorithm called the *mean update* (MU) from a recent theoretical result (Nakajima et al., 2011): the MU algorithm is based on the variational Bayesian approximation, and gives the global optimal solution for a large subset of parameters in each step. Through experiments, we show that the MU algorithm compares favorably with a standard iterative algorithm for variational Bayesian inference. We also demonstrate the usefulness of SAMF in foreground/background video separation, where sparsity is induced based on image segmentation.

2. Formulation

In this section, we formulate the sparse additive matrix factorization (SAMF) model.

2.1. Examples of Factorization

In ordinary MF, an observed matrix $V \in \mathbb{R}^{L\times M}$ is modeled by a low rank target matrix $U \in \mathbb{R}^{L\times M}$ contaminated with a random noise matrix $\mathcal{E} \in \mathbb{R}^{L\times M}$.

$$V = U + \mathcal{E}.$$

Then the target matrix U is decomposed into the product of two matrices $A \in \mathbb{R}^{M\times H}$ and $B \in \mathbb{R}^{L\times H}$:

$$U^{\text{low-rank}} = BA^\top = \sum_{h=1}^{H} \boldsymbol{b}_h \boldsymbol{a}_h^\top, \tag{1}$$

where $\top$ denotes the transpose of a matrix or vector. Throughout the paper, we denote a column vector of a matrix by a bold smaller letter, and a row vector by a bold smaller letter with a tilde:

$$A = (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_H) = (\widetilde{\boldsymbol{a}}_1, \ldots, \widetilde{\boldsymbol{a}}_M)^\top,$$
$$B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_H) = (\widetilde{\boldsymbol{b}}_1, \ldots, \widetilde{\boldsymbol{b}}_L)^\top.$$

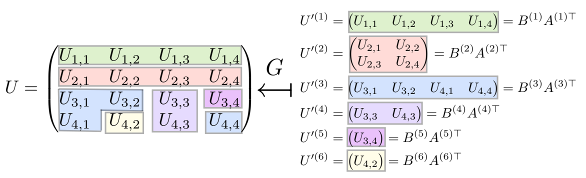

$U = \begin{pmatrix} U_{1,1} & U_{1,2} & U_{1,3} & U_{1,4} \\ U_{2,1} & U_{2,2} & U_{2,3} & U_{2,4} \\ U_{3,1} & U_{3,2} & U_{3,3} & U_{3,4} \\ U_{4,1} & U_{4,2} & U_{4,3} & U_{4,4} \end{pmatrix}$
G
$U'^{(1)} = (U_{1,1} \; U_{1,2} \; U_{1,3} \; U_{1,4}) = B^{(1)}A^{(1)\top}$
$U'^{(2)} = \begin{pmatrix} U_{2,1} & U_{2,2} \\ U_{2,3} & U_{2,4} \end{pmatrix} = B^{(2)}A^{(2)\top}$
$U'^{(3)} = (U_{3,1} \; U_{3,2} \; U_{4,1} \; U_{4,4}) = B^{(3)}A^{(3)\top}$
$U'^{(4)} = (U_{3,3} \; U_{4,3}) = B^{(4)}A^{(4)\top}$
$U'^{(5)} = (U_{3,4}) = B^{(5)}A^{(5)\top}$
$U'^{(6)} = (U_{4,2}) = B^{(6)}A^{(6)\top}$

Figure 1: An example of SMF term construction. $G(\cdot;\mathcal{X})$ with $\mathcal{X} : (k, l', m') \mapsto (l, m)$ maps the set $\{U'^{(k)}\}_{k=1}^{K}$ of the PR matrices to the target matrix U, so that $U'^{(k)}_{l',m'} = U_{\mathcal{X}(k,l',m')} = U_{l,m}$.

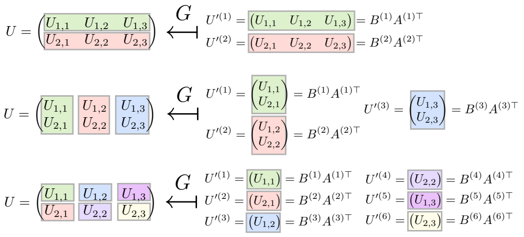

$U = \begin{pmatrix} U_{1,1} & U_{1,2} & U_{1,3} \\ U_{2,1} & U_{2,2} & U_{2,3} \end{pmatrix}$
G
$U'^{(1)} = (U_{1,1} \; U_{1,2} \; U_{1,3}) = B^{(1)}A^{(1)\top}$
$U'^{(2)} = (U_{2,1} \; U_{2,2} \; U_{2,3}) = B^{(2)}A^{(2)\top}$
$U = \begin{pmatrix} U_{1,1} & U_{1,2} & U_{1,3} \\ U_{2,1} & U_{2,2} & U_{2,3} \end{pmatrix}$
G
$U'^{(1)} = \begin{pmatrix} U_{1,1} \\ U_{2,1} \end{pmatrix} = B^{(1)}A^{(1)\top}$
$U'^{(3)} = \begin{pmatrix} U_{1,3} \\ U_{2,3} \end{pmatrix} = B^{(3)}A^{(3)\top}$
$U'^{(2)} = \begin{pmatrix} U_{1,2} \\ U_{2,2} \end{pmatrix} = B^{(2)}A^{(2)\top}$
$U = \begin{pmatrix} U_{1,1} & U_{1,2} & U_{1,3} \\ U_{2,1} & U_{2,2} & U_{2,3} \end{pmatrix}$
G
$U'^{(1)} = (U_{1,1}) = B^{(1)}A^{(1)\top}$
$U'^{(4)} = (U_{2,2}) = B^{(4)}A^{(4)\top}$
$U'^{(2)} = (U_{2,1}) = B^{(2)}A^{(2)\top}$
$U'^{(5)} = (U_{1,3}) = B^{(5)}A^{(5)\top}$
$U'^{(3)} = (U_{1,2}) = B^{(3)}A^{(3)\top}$
$U'^{(6)} = (U_{2,3}) = B^{(6)}A^{(6)\top}$

Figure 2: SMF construction for the row-wise (top), the column-wise (middle), and the element-wise (bottom) sparse terms.

The last equation in Eq.(1) implies that the *plain* matrix product (i.e., $BA^\top$) is the sum of rank-1 components. It was elucidated that this product induces an implicit regularization effect called *model-induced regularization* (MIR), and a low-rank (singular-component-wise sparse) solution is produced under the variational Bayesian approximation (Nakajima and Sugiyama, 2011).

Let us consider other types of factorization:

$$U^{\text{row}} = \mathit{\Gamma}_E D = (\gamma^e_1 \widetilde{\boldsymbol{d}}_1, \ldots, \gamma^e_L \widetilde{\boldsymbol{d}}_L)^\top, \tag{2}$$

$$U^{\text{column}} = E\mathit{\Gamma}_D = (\gamma^d_1 \boldsymbol{e}_1, \ldots, \gamma^d_M \boldsymbol{e}_M), \tag{3}$$

where $\mathit{\Gamma}_D = \mathrm{diag}(\gamma^d_1, \ldots, \gamma^d_M) \in \mathbb{R}^{M\times M}$ and $\mathit{\Gamma}_E = \mathrm{diag}(\gamma^e_1, \ldots, \gamma^e_L) \in \mathbb{R}^{L\times L}$ are diagonal matrices, and $D, E \in \mathbb{R}^{L\times M}$. These examples are also matrix products, but one of the factors is restricted to be diagonal. Because of this diagonal constraint, the l-th diagonal entry γ^e_l in $\mathit{\Gamma}_E$ is shared by all the entries in the l-th row of U^{row} as a common factor. Similarly, the m-th diagonal entry γ^d_m in $\mathit{\Gamma}_D$ is shared by all the entries in the m-th column of U^{column}.

Another example is the Hadamard (or element-wise) product:

$$U^{\text{element}} = E * D, \text{ where } (E * D)_{l,m} = E_{l,m}D_{l,m}. \tag{4}$$

In this factorization form, no entry in E and D is shared by more than one entry in U^{element}.

In fact, the forms (2)–(4) of factorization induce different types of sparsity, through the MIR mechanism. In Section 2.2, they will be derived as a row-wise, a column-wise, and an element-wise sparsity inducing terms, respectively, within a unified framework.

2.2. A General Expression of Factorization

Our general expression consists of partitioning, rearrangement, and factorization. The following is the form of a sparse matrix factorization (SMF) term:

$$U = G(\{U'^{(k)}\}_{k=1}^{K}; \mathcal{X}), \text{ where } U'^{(k)} = B^{(k)}A^{(k)\top}. \tag{5}$$

Figure 1 shows how to construct an SMF term. First, we partition the entries of U into K parts. Then, by rearranging the entries in each part, we form partitioned-and-rearranged (PR) matrices $U'^{(k)} \in \mathbb{R}^{L'^{(k)} \times M'^{(k)}}$ for $k = 1, \ldots, K$. Finally, each of $U'^{(k)}$ is decomposed into the product of $A^{(k)} \in \mathbb{R}^{M'^{(k)} \times H'^{(k)}}$ and $B^{(k)} \in \mathbb{R}^{L'^{(k)} \times H'^{(k)}}$, where $H'^{(k)} \leq \min(L'^{(k)}, M'^{(k)})$.

In Eq.(5), the function $G(\cdot; \mathcal{X})$ is responsible for partitioning and rearrangement: It maps the set $\{U'^{(k)}\}_{k=1}^{K}$ of the PR matrices to the target matrix $U \in \mathbb{R}^{L \times M}$, based on the one-to-one map $\mathcal{X} : (k, l', m') \mapsto (l, m)$ from indices of the entries in $\{U'^{(k)}\}_{k=1}^{K}$ to indices of the entries in U, such that

$$\left(G(\{U'^{(k)}\}_{k=1}^{K}; \mathcal{X})\right)_{l,m} = U_{l,m} = U_{\mathcal{X}(k,l',m')} = U'^{(k)}_{l',m'}. \tag{6}$$

As will be discussed in Section 4.1, the SMF term expression (5) under the variational Bayesian approximation induces low-rank sparsity in each partition. Therefore, partition-wise sparsity is induced, if we design a SMF term so that $\{U'^{(k)}\}$ for all k are rank-1 matrices (i.e., vectors).

Let us, for example, assume that row-wise sparsity is required. We first make the row-wise partition, i.e., separate $U \in \mathbb{R}^{L \times M}$ into L pieces of M-dimensional row vectors $U'^{(l)} = \widetilde{\boldsymbol{u}}_l^{\top} \in \mathbb{R}^{1 \times M}$. Then, we factorize each partition as $U'^{(l)} = B^{(l)} A^{(l)\top}$ (see the top illustration in Figure 2). Thus, we obtain the row-wise sparse term (2). Here, $\mathcal{X}(k, 1, m') = (k, m')$ makes the following connection between Eqs.(2) and (5): $\gamma_l^e = B^{(k)} \in \mathbb{R}, \widetilde{\boldsymbol{d}}_l = A^{(k)} \in \mathbb{R}^{M \times 1}$ for $k = l$. Similarly, requiring column-wise and element-wise sparsity leads to Eqs.(3) and (4), respectively (see the bottom two illustrations in Figure 2). Table 1 summarizes how to design these SMF terms, where $\textit{vec-order}(k) = (1 + ((k-1) \bmod L), \lceil k/L \rceil)$ goes along the columns one after another in the same way as the *vec* operator forms a vector by stacking the columns of a matrix (in other words, $(U'^{(1)}, \ldots, U'^{(K)})^{\top} = \textit{vec}(U)$).

In practice, SMF terms should be designed based on side information. In robust PCA (Candes et al., 2009; Babacan et al., 2012), the element-wise sparse term is added to the low-rank term for the case where the observation is expected to contain spiky noise. Here, we can say that the 'expectation of spiky noise' is used as side information. Using the SMF expression (5), we can similarly add a row-wise term and/or a column-wise term when the corresponding type of sparse noise is expected.

The SMF expression enables us to use side information in a more flexible way. In Section 5.2, we apply our method to a foreground/background video separation problem, where *moving* objects are considered to belong to the foreground. The previous approach (Candes et al., 2009; Babacan et al., 2012) adds an element-wise sparse term for capturing the moving objects. However, we can also use a natural assumption that the pixels in an image segment with similar intensity values tend to belong to the same object and hence share the same label. To use this side information, we adopt a segment-wise sparse term, where the PR matrix is constructed based on a precomputed over-segmented image. We will show in Section 5.2 that the segment-wise sparse term captures the foreground more accurately than the element-wise sparse term.

The SMF expression also provides a unified framework where a single theory can be applied to various types of factorization. Based on this framework, we derive a useful algorithm for variational approximation in Section 3.

2.3. Formulation of SAMF

We define a sparse additive matrix factorization (SAMF) model as a sum of SMF terms (5):

$$V = \textstyle\sum_{s=1}^{S} U^{(s)} + \mathcal{E}, \tag{7}$$

$$\text{where } U^{(s)} = G(\{B^{(k,s)} A^{(k,s)\top}\}_{k=1}^{K^{(s)}}; \mathcal{X}^{(s)}). \tag{8}$$

Let us summarize the parameters as follows:

$$\Theta = \{\Theta_{\mathrm{A}}^{(s)}, \Theta_{\mathrm{B}}^{(s)}\}_{s=1}^{S},$$
$$\text{where } \Theta_{\mathrm{A}}^{(s)} = \{A^{(k,s)}\}_{k=1}^{K^{(s)}}, \qquad \Theta_{\mathrm{B}}^{(s)} = \{B^{(k,s)}\}_{k=1}^{K^{(s)}}.$$

As in the probabilistic MF (Salakhutdinov and Mnih, 2008), we assume independent Gaussian noise and priors. Thus, the likelihood and the priors are written as

$$p(V|\Theta) \propto \exp\left(-\frac{1}{2\sigma^2}\left\|V - \textstyle\sum_{s=1}^{S} U^{(s)}\right\|_{\mathrm{Fro}}^2\right), \tag{9}$$

$$p(\{\Theta_{\mathrm{A}}^{(s)}\}_{s=1}^{S}) \propto \exp\Big(-\frac{1}{2}\cdot \textstyle\sum_{s=1}^{S}\sum_{k=1}^{K^{(s)}} \mathrm{tr}\left(A^{(k,s)} C_A^{(k,s)-1} A^{(k,s)\top}\right)\Big), \tag{10}$$

$$p(\{\Theta_{\mathrm{B}}^{(s)}\}_{s=1}^{S}) \propto \exp\Big(-\frac{1}{2}\cdot \textstyle\sum_{s=1}^{S}\sum_{k=1}^{K^{(s)}} \mathrm{tr}\left(B^{(k,s)} C_B^{(k,s)-1} B^{(k,s)\top}\right)\Big), \tag{11}$$

where $\|\cdot\|_{\mathrm{Fro}}$ and $\mathrm{tr}(\cdot)$ denote the Frobenius norm and the trace of a matrix, respectively. We assume that the prior covariances of $A^{(k,s)}$ and $B^{(k,s)}$ are diagonal and positive-definite:

$$C_A^{(k,s)} = \mathrm{diag}(c_{a_1}^{(k,s)2}, \ldots, c_{a_H}^{(k,s)2}),$$
$$C_B^{(k,s)} = \mathrm{diag}(c_{b_1}^{(k,s)2}, \ldots, c_{b_H}^{(k,s)2}).$$

Without loss of generality, we assume that the diagonal entries of $C_A^{(k,s)} C_B^{(k,s)}$ are arranged in the non-increasing order, i.e., $c_{a_h}^{(k,s)} c_{b_h}^{(k,s)} \geq c_{a_{h'}}^{(k,s)} c_{b_{h'}}^{(k,s)}$ for any pair $h < h'$.

2.4. Variational Bayesian Approximation

The Bayes posterior is written as

$$p(\Theta|V) = \frac{p(V|\Theta)p(\Theta)}{p(V)}, \tag{12}$$

where $p(V) = \langle p(V|\Theta)\rangle_{p(\Theta)}$ is the marginal likelihood. Here, $\langle\cdot\rangle_p$ denotes the expectation over the distribution p. Since the Bayes posterior (12) is computationally intractable, the variational Bayesian (VB) approximation was proposed (Bishop, 1999; Lim and Teh, 2007; Ilin and Raiko, 2010; Babacan et al., 2012).

Let $r(\Theta)$, or r for short, be a trial distribution. The following functional with respect to r is called the free energy:

$$F(r|V) = \left\langle \log \frac{r(\Theta)}{p(\Theta|V)} \right\rangle_{r(\Theta)} - \log p(V). \tag{13}$$

The first term is the Kullback-Leibler (KL) distance from the trial distribution to the Bayes posterior, and the second term is a constant. Therefore, minimizing the free energy (13) amounts to finding a distribution closest to the Bayes posterior in the sense of the KL distance. In the VB approximation, the free energy (13) is minimized over some restricted function space.

Following the standard VB procedure (Bishop, 1999; Lim and Teh, 2007; Babacan et al., 2012), we impose the following decomposability constraint on the posterior:

$$r(\Theta) = \prod_{s=1}^{S} r_{\mathrm{A}}^{(s)}(\Theta_{\mathrm{A}}^{(s)}) r_{\mathrm{B}}^{(s)}(\Theta_{\mathrm{B}}^{(s)}). \tag{14}$$

Under this constraint, it is easy to show that the VB posterior minimizing the free energy (13) is written as

$$r(\Theta) = \prod_{s=1}^{S} \prod_{k=1}^{K^{(s)}} \left(\prod_{m'=1}^{M'^{(k,s)}} \mathcal{N}_{H'^{(k,s)}}(\widetilde{\boldsymbol{a}}_{m'}^{(k,s)}; \widetilde{\widehat{\boldsymbol{a}}}_{m'}^{(k,s)}, \Sigma_{A}^{(k,s)}) \cdot \prod_{l'=1}^{L'^{(k,s)}} \mathcal{N}_{H'^{(k,s)}}(\widetilde{\boldsymbol{b}}_{l'}^{(k,s)}; \widetilde{\widehat{\boldsymbol{b}}}_{l'}^{(k,s)}, \Sigma_{B}^{(k,s)}) \right), \tag{15}$$

where $\mathcal{N}_d(\cdot; \boldsymbol{\mu}, \Sigma)$ denotes the d-dimensional Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance Σ.

3. Algorithm for SAMF

In this section, we first give a theorem that reduces a partial SAMF problem to the ordinary MF problem, which can be solved *analytically*. Then we derive an algorithm for the entire SAMF problem.

3.1. Key Theorem

Let us denote the mean of $U^{(s)}$, defined in Eq.(8), over the VB posterior by

$$\begin{aligned} \widehat{U}^{(s)} &= \langle U^{(s)} \rangle_{r_{\mathrm{A}}^{(s)}(\Theta_{A}^{(s)}) r_{\mathrm{B}}^{(s)}(\Theta_{B}^{(s)})} \\ &= G(\{\widehat{B}^{(k,s)} \widehat{A}^{(k,s)\top}\}_{k=1}^{K^{(s)}}; \mathcal{X}^{(s)}). \end{aligned} \tag{16}$$

Then we obtain the following theorem (the proof is omitted because of the space limitation):

Theorem 1 *Given $\{\widehat{U}^{(s')}\}_{s' \neq s}$ and the noise variance σ^2, the VB posterior of $(\Theta_{A}^{(s)}, \Theta_{B}^{(s)}) = \{A^{(k,s)}, B^{(k,s)}\}_{k=1}^{K^{(s)}}$ coincides with the VB posterior of the following MF model:*

$$p(Z'^{(k,s)} | A^{(k,s)}, B^{(k,s)}) \propto \exp\left(-\frac{1}{2\sigma^2} \left\| Z'^{(k,s)} - B^{(k,s)} A^{(k,s)\top} \right\|_{Fro}^{2} \right), \tag{17}$$

$$p(A^{(k,s)}) \propto \exp\left(-\frac{1}{2} tr \left(A^{(k,s)} C_{A}^{(k,s)-1} A^{(k,s)\top} \right) \right), \tag{18}$$

$$p(B^{(k,s)}) \propto \exp\left(-\frac{1}{2} tr \left(B^{(k,s)} C_{B}^{(k,s)-1} B^{(k,s)\top} \right) \right), \tag{19}$$

for each $k = 1, \ldots, K^{(s)}$. Here, $Z'^{(k,s)} \in \mathbb{R}^{L'^{(k,s)} \times M'^{(k,s)}}$ is defined as

$$Z'^{(k,s)}_{l',m'} = Z^{(s)}_{\mathcal{X}^{(s)}(k,l',m')}, \text{ where } Z^{(s)} = V - \sum_{s' \neq s} \widehat{U}^{(s)}. \tag{20}$$

The left formula in Eq.(20) relates the entries of $Z^{(s)} \in \mathbb{R}^{L \times M}$ to the entries of $\{Z'^{(k,s)} \in \mathbb{R}^{L'^{(k,s)} \times M'^{(k,s)}}\}_{k=1}^{K^{(s)}}$ by using the map $\mathcal{X}^{(s)} : (k, l', m') \mapsto (l, m)$ (see Eq.(6) and Figure 1).

When the noise variance σ^2 is unknown, the following lemma is useful (the proof is omitted):

Lemma 2 *Given the VB posterior for* $\{\varTheta_A^{(s)}, \varTheta_B^{(s)}\}_{s=1}^{S}$*, the noise variance* σ^2 *minimizing the free energy (13) is given by*

$$\sigma^2 = \frac{1}{LM}\Bigg\{\|V\|_{Fro}^2 - 2\sum_{s=1}^{S} tr\left(\widehat{U}^{(s)\top}\left(V - \sum_{s'=s+1}^{S}\widehat{U}^{(s')}\right)\right)$$

$$+ \textstyle\sum_{s=1}^{S}\sum_{k=1}^{K^{(s)}} tr\Big((\widehat{A}^{(k,s)\top}\widehat{A}^{(k,s)} + M'^{(k,s)}\varSigma_A^{(k,s)}) \cdot (\widehat{B}^{(k,s)\top}\widehat{B}^{(k,s)} + L'^{(k,s)}\varSigma_B^{(k,s)})\Big)\Bigg\}. \quad (21)$$

3.2. Partial Analytic Solution

Theorem 1 allows us to utilize the results given in Nakajima et al. (2011), which give the global analytic solution for VBMF. Combining Theorem 1 above and Corollaries 1–3 in Nakajima et al. (2011), we obtain the following corollaries. Below, we assume that $L'^{(k,s)} \leq M'^{(k,s)}$ for all (k, s). We can always take the mapping $\mathcal{X}^{(s)}$ so, without any practical restriction.

Corollary 1 *Assume that* $\{\widehat{U}^{(s')}\}_{s' \neq s}$ *and the noise variance* σ^2 *are given. Let* $\gamma_h^{(k,s)}$ (≥ 0) *be the* h*-th largest singular value of* $Z'^{(k,s)}$*, and let* $\boldsymbol{\omega}_{a_h}^{(k,s)}$ *and* $\boldsymbol{\omega}_{b_h}^{(k,s)}$ *be the associated right and left singular vectors:*

$$Z'^{(k,s)} = \sum_{h=1}^{L'^{(k,s)}} \gamma_h^{(k,s)} \boldsymbol{\omega}_{b_h}^{(k,s)} \boldsymbol{\omega}_{a_h}^{(k,s)\top}.$$

Let $\widehat{\gamma}_h^{(k,s)}$ *be the* second *largest real solution of the following* quartic *equation with respect to* t*:*

$$f_h(t) := t^4 + \xi_3^{(k,s)} t^3 + \xi_2^{(k,s)} t^2 + \xi_1^{(k,s)} t + \xi_0^{(k,s)} = 0, \quad (22)$$

where the coefficients are defined by

$$\xi_3^{(k,s)} = \frac{(L'^{(k,s)} - M'^{(k,s)})^2 \gamma_h^{(k,s)}}{L'^{(k,s)} M'^{(k,s)}},$$

$$\xi_2^{(k,s)} = -\left(\xi_3 \gamma_h^{(k,s)} + \frac{(L'^{(k,s)2} + M'^{(k,s)2})\eta_h^{(k,s)2}}{L'^{(k,s)} M'^{(k,s)}} + \frac{2\sigma^4}{c_{a_h}^{(k,s)2} c_{b_h}^{(k,s)2}}\right),$$

$$\xi_1^{(k,s)} = \xi_3^{(k,s)} \sqrt{\xi_0^{(k,s)}},$$

$$\xi_0^{(k,s)} = \left(\eta_h^{(k,s)2} - \frac{\sigma^4}{c_{a_h}^{(k,s)2} c_{b_h}^{(k,s)2}}\right)^2,$$

$$\eta_h^{(k,s)2} = \left(1 - \frac{\sigma^2 L'^{(k,s)}}{\gamma_h^{(k,s)2}}\right)\left(1 - \frac{\sigma^2 M'^{(k,s)}}{\gamma_h^{(k,s)2}}\right)\gamma_h^{(k,s)2}.$$

Let

$$\widetilde{\gamma}_h^{(k,s)} = \sqrt{\tau + \sqrt{\tau^2 - L'^{(k,s)}M'^{(k,s)}\sigma^4}}, \tag{23}$$

where

$$\tau = \frac{(L'^{(k,s)} + M'^{(k,s)})\sigma^2}{2} + \frac{\sigma^4}{2c_{a_h}^{(k,s)2}c_{b_h}^{(k,s)2}}.$$

Then, the **global** *VB solution can be expressed as*

$$\widehat{U}'^{(k,s)\mathrm{VB}} = (\widehat{B}^{(k,s)}\widehat{A}^{(k,s)\top})^{\mathrm{VB}} = \sum_{h=1}^{H'^{(k,s)}} \widehat{\gamma}_h^{(k,s)\mathrm{VB}}\boldsymbol{\omega}_{b_h}^{(k,s)}\boldsymbol{\omega}_{a_h}^{(k,s)\top},$$

$$\textit{where} \quad \widehat{\gamma}_h^{(k,s)\mathrm{VB}} = \begin{cases} \widehat{\gamma}_h^{(k,s)} & \textit{if } \gamma_h^{(k,s)} > \widetilde{\gamma}_h^{(k,s)}, \\ 0 & \textit{otherwise.} \end{cases} \tag{24}$$

Corollary 2 *Given* $\{\widehat{U}^{(s')}\}_{s' \neq s}$ *and the noise variance* σ^2*, the* **global empirical** *VB solution is given by*

$$\widehat{U}'^{(k,s)\mathrm{EVB}} = \sum_{h=1}^{H'^{(k,s)}} \widehat{\gamma}_h^{(k,s)\mathrm{EVB}}\boldsymbol{\omega}_{b_h}^{(k,s)}\boldsymbol{\omega}_{a_h}^{(k,s)\top},$$

$$\textit{where} \quad \widehat{\gamma}_h^{(k,s)\mathrm{EVB}} = \begin{cases} \breve{\gamma}_h^{(k,s)\mathrm{VB}} & \textit{if } \gamma_h^{(k,s)} > \underline{\gamma}_h^{(k,s)} \textit{ and } \varDelta_h^{(k,s)} \leq 0, \\ 0 & \textit{otherwise.} \end{cases} \tag{25}$$

Here,

$$\underline{\gamma}_h^{(k,s)} = (\sqrt{L'^{(k,s)}} + \sqrt{M'^{(k,s)}})\sigma, \tag{26}$$

$$\breve{c}_h^{(k,s)2} = \frac{1}{2L'^{(k,s)}M'^{(k,s)}}\left(\gamma_h^{(k,s)2} - (L'^{(k,s)} + M'^{(k,s)})\sigma^2 + \sqrt{\left(\gamma_h^{(k,s)2} - (L'^{(k,s)} + M'^{(k,s)})\sigma^2\right)^2 - 4L'^{(k,s)}M'^{(k,s)}\sigma^4}\right), \tag{27}$$

$$\varDelta_h^{(k,s)} = M'^{(k,s)}\log\left(\frac{\gamma_h^{(k,s)}}{M'^{(k,s)}\sigma^2}\breve{\gamma}_h^{(k,s)\mathrm{VB}} + 1\right) + L'^{(k,s)}\log\left(\frac{\gamma_h^{(k,s)}}{L'^{(k,s)}\sigma^2}\breve{\gamma}_h^{(k,s)\mathrm{VB}} + 1\right) + \frac{1}{\sigma^2}\left(-2\gamma_h^{(k,s)}\breve{\gamma}_h^{(k,s)\mathrm{VB}} + L'^{(k,s)}M'^{(k,s)}\breve{c}_h^{(k,s)2}\right), \tag{28}$$

and $\breve{\gamma}_h^{(k,s)\mathrm{VB}}$ *is the VB solution for* $c_{a_h}^{(k,s)}c_{b_h}^{(k,s)} = \breve{c}_h^{(k,s)}$.

Corollary 3 *Given* $\{\widehat{U}^{(s')}\}_{s' \neq s}$ *and the noise variance* σ^2*, the VB posteriors are given by*

$$r_{\mathrm{A}^{(k,s)}}^{\mathrm{VB}}(A^{(k,s)}) = \prod_{h=1}^{H'^{(k,s)}} \mathcal{N}_{M'^{(k,s)}}(\boldsymbol{a}_h^{(k,s)}; \widehat{\boldsymbol{a}}_h^{(k,s)}, \sigma_{a_h}^{(k,s)2}I_{M'^{(k,s)}}),$$

$$r_{\mathrm{B}^{(k,s)}}^{\mathrm{VB}}(B^{(k,s)}) = \prod_{h=1}^{H'^{(k,s)}} \mathcal{N}_{L'^{(k,s)}}(\boldsymbol{b}_h^{(k,s)}; \widehat{\boldsymbol{b}}_h^{(k,s)}, \sigma_{b_h}^{(k,s)2}I_{L'^{(k,s)}}),$$

where, for $\widehat{\gamma}_h^{(k,s)\mathrm{VB}}$ *being the solution given by Corollary 1,*

$$\begin{aligned}
\widehat{\boldsymbol{a}}_h^{(k,s)} &= \pm\sqrt{\widehat{\gamma}_h^{(k,s)\mathrm{VB}}\widehat{\delta}_h^{(k,s)}} \cdot \boldsymbol{\omega}_{a_h}^{(k,s)}, \quad \widehat{\boldsymbol{b}}_h^{(k,s)} = \pm\sqrt{\widehat{\gamma}_h^{(k,s)\mathrm{VB}}\widehat{\delta}_h^{(k,s)-1}} \cdot \boldsymbol{\omega}_{b_h}^{(k,s)}, \\
\sigma_{a_h}^{(k,s)2} &= \frac{1}{2M'^{(k,s)}(\widehat{\gamma}_h^{(k,s)\mathrm{VB}}\widehat{\delta}_h^{(k,s)-1} + \sigma^2 c_{a_h}^{(k,s)-2})}\Bigg\{ - \left(\widehat{\eta}_h^{(k,s)2} - \sigma^2(M'^{(k,s)} - L'^{(k,s)})\right) \\
&\qquad + \sqrt{(\widehat{\eta}_h^{(k,s)2} - \sigma^2(M'^{(k,s)} - L'^{(k,s)}))^2 + 4M'^{(k,s)}\sigma^2\widehat{\eta}_h^{(k,s)2}}\Bigg\}, \\
\sigma_{b_h}^{(k,s)2} &= \frac{1}{2L'^{(k,s)}(\widehat{\gamma}_h^{(k,s)\mathrm{VB}}\widehat{\delta}_h^{(k,s)} + \sigma^2 c_{b_h}^{(k,s)-2})}\Bigg\{ - \left(\widehat{\eta}_h^{(k,s)2} + \sigma^2(M'^{(k,s)} - L'^{(k,s)})\right) \\
&\qquad + \sqrt{(\widehat{\eta}_h^{(k,s)2} + \sigma^2(M'^{(k,s)} - L'^{(k,s)}))^2 + 4L'^{(k,s)}\sigma^2\widehat{\eta}_h^{(k,s)2}}\Bigg\}, \\
\widehat{\delta}_h^{(k,s)} &= \frac{1}{2\sigma^2 M'^{(k,s)} c_{a_h}^{(k,s)-2}}\Bigg\{(M'^{(k,s)} - L'^{(k,s)})(\gamma_h^{(k,s)} - \widehat{\gamma}_h^{(k,s)\mathrm{VB}}) \\
&\qquad + \sqrt{(M'^{(k,s)} - L'^{(k,s)})^2(\gamma_h^{(k,s)} - \widehat{\gamma}_h^{(k,s)\mathrm{VB}})^2 + \frac{4\sigma^4 L'^{(k,s)} M'^{(k,s)}}{c_{a_h}^{(k,s)2} c_{b_h}^{(k,s)2}}}\Bigg\}, \\
\widehat{\eta}_h^{(k,s)2} &= \begin{cases} \eta_h^{(k,s)2} & \textit{if } \gamma_h^{(k,s)} > \widetilde{\gamma}_h^{(k,s)}, \\ \frac{\sigma^4}{c_{a_h}^{(k,s)2} c_{b_h}^{(k,s)2}} & \textit{otherwise.} \end{cases}
\end{aligned}$$

When σ^2 is known, Corollary 1 and Corollary 2 provide the global analytic solution of the *partial* problem, where the variables on which $\{\widehat{U}^{(s')}\}_{s' \neq s}$ depends are fixed. Note that they give the global analytic solution for single-term ($S = 1$) SAMF.

3.3. Mean Update Algorithm

Using Corollaries 1–3 and Lemma 2, we propose an algorithm for SAMF, called the *mean update* (MU). We describe its pseudo-code in Algorithm 1, where $0_{(d_1,d_2)}$ denotes the $d_1 \times d_2$ matrix with all entries equal to zero.

Although each of the corollaries and the lemma above guarantee the global optimality for each step, the MU algorithm does not generally guarantee the simultaneous global optimality over the entire parameter space. Nevertheless, experimental results in Section 5 show that the MU algorithm performs very well in practice.

4. Discussion

In this section, we first discuss the relation between MIR and ARD. Then, we introduce the standard VB iteration for SAMF, which is used as a baseline in the experiments.

4.1. Relation between MIR and ARD

The MIR effect (Nakajima and Sugiyama, 2011) induced by *factorization* actually has a close connection to the *automatic relevance determination* (ARD) effect (Neal, 1996). As-

Algorithm 1 Mean update (MU) algorithm for (empirical) VB SAMF.

```
1: Initialization: $\widehat{U}^{(s)} \leftarrow 0_{(L,M)}$ for $s = 1, \ldots, S$, $\sigma^2 \leftarrow \|V\|^2_{\mathrm{Fro}}/(LM)$.
2: for $s = 1$ to $S$ do
3:     The (empirical) VB solution of $U'^{(k,s)} = B^{(k,s)} A^{(k,s)\top}$ for each $k = 1, \ldots, K^{(s)}$, given $\{\widehat{U}^{(s')}\}_{s' \neq s}$, is computed by Corollary 1 (Corollary 2).
4:     $\widehat{U}^{(s)} \leftarrow G(\{\widehat{B}^{(k,s)} \widehat{A}^{(k,s)\top}\}_{k=1}^{K^{(s)}}; \mathcal{X}^{(s)})$.
5: end for
6: $\sigma^2$ is estimated by Lemma 2, given the VB posterior on $\{\Theta_A^{(s)}, \Theta_B^{(s)}\}_{s=1}^{S}$ (computed by Corollary 3).
7: Repeat 2 to 6 until convergence.
```

sume that $C_A = I_H$, where I_d denotes the d-dimensional identity matrix, in the *plain* MF model (17)–(19) (here we omit the suffixes k and s for brevity), and consider the following transformation: $BA^\top \mapsto U \in \mathbb{R}^{L \times M}$. Then, the likelihood (17) and the prior (18) on A are rewritten as

$$p(Z'|U) \propto \exp\left(-\frac{1}{2\sigma^2}\|Z' - U\|^2_{\mathrm{Fro}}\right), \tag{29}$$

$$p(U|B) \propto \exp\left(-\frac{1}{2}\mathrm{tr}\left(U^\top (BB^\top)^\dagger U\right)\right), \tag{30}$$

where † denotes the Moore-Penrose generalized inverse of a matrix. The prior (19) on B is kept unchanged. $p(U|B)$ in Eq.(30) is so-called the ARD prior with the covariance hyperparameter $BB^\top \in \mathbb{R}^{L \times L}$. It is known that this induces the ARD effect, i.e., the *empirical* Bayesian procedure where the hyperparameter $BB^\top$ is also estimated from observations induces strong regularization and sparsity (Neal, 1996) (see also Efron and Morris (1973) for a simple Gaussian case).

In the current context, Eq.(30) induces low-rank sparsity on U if no restriction on $BB^\top$ is imposed. Similarly, we can show that $(\gamma_l^e)^2$ in Eq.(2) plays a role of the prior variance shared by the entries in $\widetilde{\boldsymbol{u}}_l \in \mathbb{R}^M$, $(\gamma_m^d)^2$ in Eq.(3) plays a role of the prior variance shared by the entries in $\boldsymbol{u}_m \in \mathbb{R}^L$, and $E_{l,m}^2$ in Eq.(4) plays a role of the prior variance on $U_{l,m} \in \mathbb{R}$, respectively. This explains the mechanism how the factorization forms in Eqs.(2)–(4) induce row-wise, column-wise, and element-wise sparsity, respectively.

When we employ the SMF term expression (5), MIR occurs in each partition. Therefore, low-rank sparsity in each partition is observed. Corollary 1 and Corollary 2 theoretically support this fact: Small singular values are discarded by thresholding in Eqs.(24) and (25).

4.2. Standard VB Iteration

Following the standard procedure for the VB approximation (Bishop, 1999; Lim and Teh, 2007; Babacan et al., 2012), we can derive the following algorithm, which we call the *stan-*

dard VB iteration:

$$\widehat{A}^{(k,s)} = \sigma^{-2} Z'^{(k,s)\top} \widehat{B}^{(k,s)} \Sigma_A^{(k,s)}, \tag{31}$$

$$\Sigma_A^{(k,s)} = \sigma^2 \left(\widehat{B}^{(k,s)\top} \widehat{B}^{(k,s)} + L'^{(k,s)} \Sigma_B^{(k,s)} + \sigma^2 C_A^{(k,s)-1} \right)^{-1}, \tag{32}$$

$$\widehat{B}^{(k,s)} = \sigma^{-2} Z'^{(k,s)} \widehat{A}^{(k,s)} \Sigma_B^{(k,s)}, \tag{33}$$

$$\Sigma_B^{(k,s)} = \sigma^2 \left(\widehat{A}^{(k,s)\top} \widehat{A}^{(k,s)} + M'^{(k,s)} \Sigma_A^{(k,s)} + \sigma^2 C_B^{(k,s)-1} \right)^{-1}. \tag{34}$$

Iterating Eqs.(31)–(34) for each (k, s) in turn until convergence gives a local minimum of the free energy (13).

In the empirical Bayesian scenario, the hyperparameters $\{C_A^{(k,s)}, C_B^{(k,s)}\}_{k=1,s=1}^{K^{(s)}\ S}$ are also estimated from observations. The following update rules give a local minimum of the free energy:

$$c_{a_h}^{(k,s)2} = \|\widehat{\boldsymbol{a}}_h^{(k,s)}\|^2 / M'^{(k,s)} + (\Sigma_A^{(k,s)})_{hh}, \tag{35}$$

$$c_{b_h}^{(k,s)2} = \|\widehat{\boldsymbol{b}}_h^{(k,s)}\|^2 / L'^{(k,s)} + (\Sigma_B^{(k,s)})_{hh}. \tag{36}$$

When the noise variance σ^2 is unknown, it is estimated by Eq.(21) in each iteration.

The standard VB iteration is computationally efficient since only a single parameter in $\{\widehat{A}^{(k,s)}, \Sigma_A^{(k,s)}, \widehat{B}^{(k,s)}, \Sigma_B^{(k,s)}, c_{a_h}^{(k,s)2}, c_{b_h}^{(k,s)2}\}_{k=1,s=1}^{K^{(s)}\ S}$ is updated in each step. However, it is known that the standard VB iteration is prone to suffer from the local minima problem (Nakajima et al., 2011). On the other hand, although the MU algorithm also does not guarantee the global optimality as a whole, it simultaneously gives the global optimal solution for the set $\{\widehat{A}^{(k,s)}, \Sigma_A^{(k,s)}, \widehat{B}^{(k,s)}, \Sigma_B^{(k,s)}, c_{a_h}^{(k,s)2}, c_{b_h}^{(k,s)2}\}_{k=1,}^{K^{(s)}}$ for each s in each step. In Section 5, we will experimentally show that the MU algorithm gives a better solution (i.e., with a smaller free energy) than the standard VB iteration.

5. Experimental Results

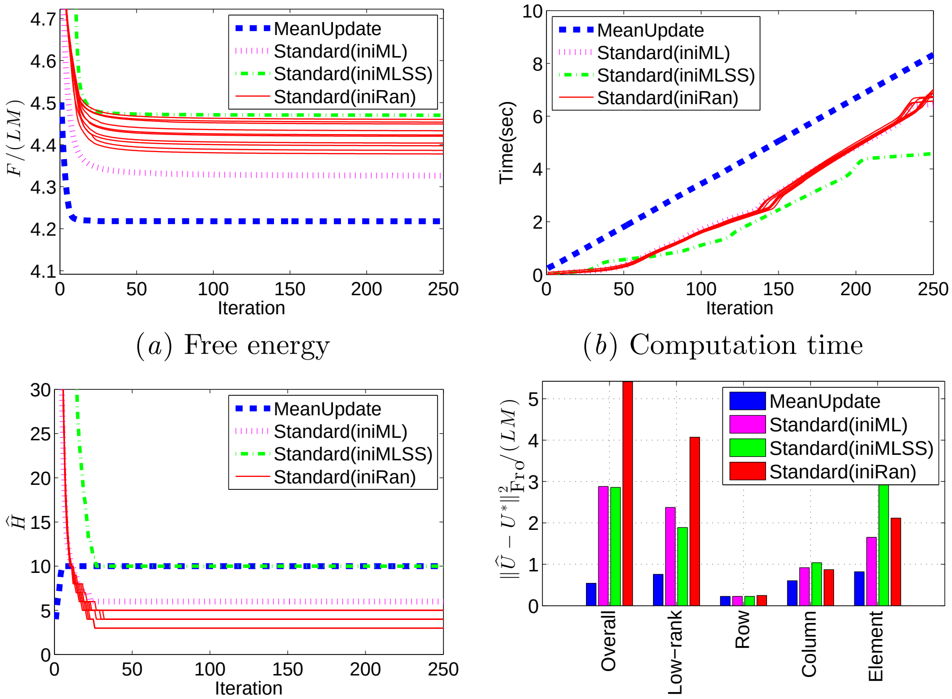
In this section, we first experimentally compare the performance of the MU algorithm and the standard VB iteration. Then, we demonstrate the usefulness of SAMF in a real-world application.

5.1. Mean Update vs. Standard VB

We compare the algorithms under the following model:

$$V = U^{\mathrm{LRCE}} + \mathcal{E},$$
$$\text{where } U^{\mathrm{LRCE}} = \textstyle\sum_{s=1}^{4} U^{(s)} = U^{\text{low-rank}} + U^{\text{row}} + U^{\text{column}} + U^{\text{element}}. \tag{37}$$

Here, 'LRCE' stands for the sum of the Low-rank, Row-wise, Column-wise, and Element-wise terms, each of which is defined in Eqs.(1)–(4). We call this model 'LRCE'-SAMF. We also evaluate 'LCE'-SAMF, 'LRE'-SAMF, and 'LE'-SAMF models. These models can be regarded as generalizations of robust PCA (Candes et al., 2009; Babacan et al., 2012), of which 'LE'-SAMF corresponds to a SAMF counterpart.



(a) Free energy

(b) Computation time

(c) Estimated rank

(d) Reconstruction error

Figure 3: Experimental results with 'LRCE'-SAMF for an artificial dataset ($L = 40, M = 100, H^* = 10, \rho = 0.05$).

We conducted an experiment with artificial data. We assume the empirical VB scenario with unknown noise variance, i.e., the hyperparameters $\{C_A^{(k,s)}, C_B^{(k,s)}\}_{k=1,s=1}^{K^{(s)}\,S}$ and the noise variance σ^2 are also estimated from observations. We use the full-rank model ($H = \min(L, M)$) for the low-rank term $U^{\text{low-rank}}$, and expect the MIR effect to find the true rank of $U^{\text{low-rank}}$, as well as the non-zero entries in $U^{\text{row}}, U^{\text{column}}$, and U^{element}.

We created an artificial dataset with the data matrix size $L = 40$ and $M = 100$, and the rank $H^* = 10$ of the *true* low-rank matrix $U^{\text{low-rank}*} = B^*A^{*\top}$. Each entry in $A^* \in \mathbb{R}^{M\times H^*}$ and $B^* \in \mathbb{R}^{L\times H^*}$ follows $\mathcal{N}_1(0, 1)$. The *true* row-wise (column-wise) part $U^{\text{row}*}$ ($U^{\text{column}*}$) was created by first randomly selecting ρL rows (ρM columns) for $\rho = 0.05$, and then adding a noise subject to $\mathcal{N}_M(\mathbf{0}, 100 \cdot I_M)$ ($\mathcal{N}_L(\mathbf{0}, 100 \cdot I_L)$) to each of the selected rows (columns). The *true* element-wise part $U^{\text{element}*}$ was similarly created by first selecting ρLM entries, and then adding a noise subject to $\mathcal{N}_1(0, 100)$ to each of the selected entries. Finally, an observed matrix V was created by adding a noise subject to $\mathcal{N}_1(0, 1)$ to each entry of the sum $U^{\text{LRCE}*}$ of the four *true* matrices.

It is known that the standard VB iteration (given in Section 4.2) is sensitive to initialization (Nakajima et al., 2011). We set the initial values in the following way: the mean parameters $\{\widehat{A}^{(k,s)}, \widehat{B}^{(k,s)}\}_{k=1,s=1}^{K^{(s)}\,S}$ were randomly created so that each entry follows $\mathcal{N}_1(0, 1)$. The covariances $\{\Sigma_A^{(k,s)}, \Sigma_B^{(k,s)}\}_{k=1,s=1}^{K^{(s)}\,S}$ and the hyperparameters $\{C_A^{(k,s)}, C_B^{(k,s)}\}_{k=1,s=1}^{K^{(s)}\,S}$ were set to the identity matrix. The initial noise variance was set to $\sigma^2 = 1$. Note that we rescaled V so that $\|V\|^2_{\mathrm{Fro}}/(LM) = 1$, before starting iteration. We ran the standard VB algorithm 10 times, starting from different initial points, and each trial is plotted by a solid line (labeled as 'Standard(iniRan)') in Figure 3.

Initialization for the MU algorithm (described in Algorithm 1) is simple. We just set initial values as follows: $\widehat{U}^{(s)} = 0_{L,M}$ for $s = 1, \ldots, S$, and $\sigma^2 = 1$. Initialization of all other variables is not needed. Furthermore, we empirically observed that the initial value for σ^2 does not affect the result much, unless it is too small. Note that, in the MU algorithm, initializing σ^2 to a large value is not harmful, because it is set to an adequate value after the first iteration with the mean parameters kept $\widehat{U}^{(s)} = 0_{L,M}$. The result with the MU algorithm is plotted by the dashed line in Figure 3.

Figures 3(a)–3(c) show the free energy, the computation time, and the estimated rank, respectively, over iterations, and Figure 3(d) shows the reconstruction errors after 250 iterations. The reconstruction errors consist of the *overall* error $\|\widehat{U}^{\mathrm{LRCE}} - U^{\mathrm{LRCE}*}\|_{\mathrm{Fro}}/(LM)$, and the four component-wise errors $\|\widehat{U}^{(s)} - U^{(s)*}\|_{\mathrm{Fro}}/(LM)$. The graphs show that the MU algorithm, whose iteration is computationally slightly more expensive, immediately converges to a local minimum with the free energy substantially lower than the standard VB iteration. The estimated rank agrees with the true rank $\widehat{H} = H^* = 10$, while all 10 trials of the standard VB iteration failed to estimate the true rank. It is also observed that the MU algorithm well reconstructs each of the four terms.

We can slightly improve the performance of the standard VB iteration by adopting different initialization schemes. The line labeled as 'Standard(iniML)' in Figure 3 indicates the maximum likelihood (ML) initialization, i.e, $(\widehat{\boldsymbol{a}}_h^{(k,s)}, \widehat{\boldsymbol{b}}_h^{(k,s)}) = (\gamma_h^{(k,s)1/2}\boldsymbol{\omega}_{a_h}^{(k,s)}, \gamma_h^{(k,s)1/2}\boldsymbol{\omega}_{b_h}^{(k,s)})$. Here, $\gamma_h^{(k,s)}$ is the h-th largest singular value of the (k,s)-th PR matrix $V'^{(k,s)}$ of V (such that $V'^{(k,s)}_{l',m'} = V_{\mathcal{X}^{(s)}(k,l',m')}$), and $\boldsymbol{\omega}_{a_h}^{(k,s)}$ and $\boldsymbol{\omega}_{b_h}^{(k,s)}$ are the associated right and left singular vectors. Also, we empirically found that starting from a small σ^2 alleviates the local minima problem. The line labeled as 'Standard(iniMLSS)' indicates the ML initialization with $\sigma^2 = 0.0001$. We can see that this scheme tends to successfully recover the true rank. However, the free energy and the reconstruction error are still substantially worse than the MU algorithm.

We tested the algorithms with other SAMF models, including 'LCE'-SAMF, 'LRE'-SAMF, and 'LE'-SAMF, under different settings for L, M, H^*, and ρ. We empirically found that the MU algorithm generally gives a better solution with lower free energy and smaller reconstruction errors than the standard VB iteration.

We also conducted experiments with benchmark datasets available from *UCI repository* (Asuncion and Newman, 2007), and found that, in most of the cases, the MU algorithm gives a better solution (with lower free energy) than the standard VB iteration.

5.2. Real-world Application

Finally, we demonstrate the usefulness of the flexibility of SAMF in a foreground (FG)/background (BG) video separation problem. Candes et al. (2009) formed the observed matrix V by stacking all pixels in each frame into each column, and applied robust PCA (with 'LE'-terms)—the low-rank term captures the *static* BG and the element-wise (or pixel-wise) term captures the *moving* FG, e.g., people walking through. Babacan et al.

(2012) proposed a VB variant of robust PCA, and performed an extensive comparison that showed advantages of the VB robust PCA over other Bayesian and non-Bayesian robust PCA methods (Ding et al., 2011; Lin et al., 2010), as well as the Gibbs sampling inference method with the same probabilistic model. Since their state-of-the-art method is conceptually the same as our VB inference method with 'LE'-SAMF (although the prior design is slightly different), we use 'LE'-SAMF as a baseline method for comparison.

The SAMF framework enables a fine-tuned design for the FG term. Assuming that the pixels in an image segment with similar intensity values tend to share the same label (i.e., FG or BG), we formed a segment-wise sparse SMF term: $U'^{(k)}$ for each k is a column vector consisting of all pixels within each segment. We produced an over-segmented image of each frame by using the efficient graph-based segmentation (EGS) algorithm (Felzenszwalb and Huttenlocher, 2004), and substituted the segment-wise sparse term for the FG term. We call this method a *segmentation-based SAMF* (sSAMF). Note that EGS is very efficient: it takes less than 0.05 sec on a laptop to segment a 192 × 144 grey image. EGS has several tuning parameters, to some of which the obtained segmentation is sensitive. However, we confirmed that sSAMF performs similarly with visually different segmentations obtained over a wide range of tuning parameters. Therefore, careful parameter tuning of EGS is not necessary for our purpose.

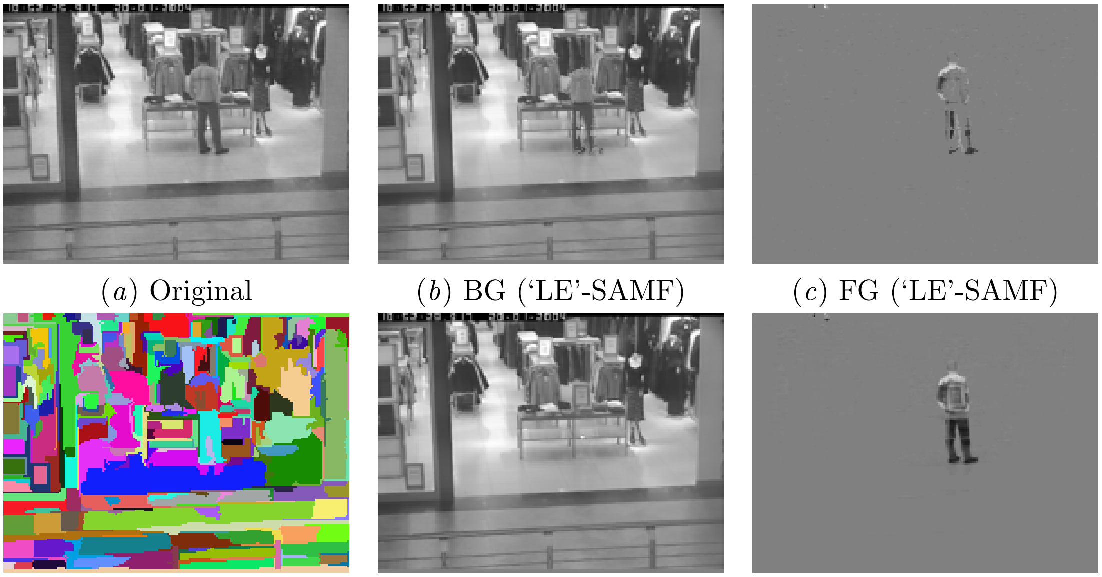
We compared sSAMF with 'LE'-SAMF on the 'WalkByShop1front' video from the *Caviar dataset*.[1] Thanks to the Bayesian framework, all unknown parameters (except the ones for segmentation) are estimated automatically with no manual parameter tuning. For both models ('LE'-SAMF and sSAMF), we used the MU algorithm, which has been shown in Section 5.1 to be practically more reliable than the standard VB iteration. The original video consists of 2360 frames, each of which is an image with 384 × 288 pixels. We resized each image into 192 × 144 pixels, and sub-sampled every 15 frames. Thus, V is of the size of 27684 (pixels) × 158 (frames). We evaluated 'LE'-SAMF and sSAMF on this video, and found that both models perform well (although 'LE'-SAMF failed in a few frames).

To contrast the methods more clearly, we created a more *difficult* video by sub-sampling every 5 frames from 1501 to 2000 (100 frames). Since more people walked through in this period, BG estimation is more unstable. The result is shown in Figure 4.

Figure 4(a) shows an original frame. This is a difficult snap shot, because the person stayed at the same position for a moment, which confuses separation. Figures 4(b) and 4(c) show the BG and the FG terms obtained by 'LE'-SAMF, respectively. We can see that 'LE'-SAMF failed to separate (the person is partly captured in the BG term). On the other hand, Figures 4(e) and 4(f) show the BG and the FG terms obtained by sSAMF based on the segmented image shown in Figure 4(d). We can see that sSAMF successfully separated the person from BG in this difficult frame. A careful look at the legs of the person makes us understand how segmentation helps separation—the legs form a single segment (light blue colored) in Figure 4(d), and the segment-wise sparse term (4(f)) captured all pixels on the legs, while the pixel-wise sparse term (4(c)) captured only a part of those pixels.

We observed that, in all frames of the *difficult* video, as well as the *easier* one, sSAMF gave good separation, while 'LE'-SAMF failed in several frames.

[1] http://groups.inf.ed.ac.uk/vision/CAVIAR/CAVIARDATA1/



(a) Original (b) BG (‘LE’-SAMF) (c) FG (‘LE’-SAMF)

(d) Segmented (e) BG (sSAMF) (f) FG (sSAMF)

Figure 4: ‘LE’-SAMF vs segmentation-based SAMF.

6. Conclusion

In this paper, we formulated a sparse additive matrix factorization (SAMF) model, which allows us to design various forms of factorization that induce various types of sparsity. We then proposed a variational Bayesian (VB) algorithm called the mean update (MU), based on a theory built upon the unified SAMF framework. The MU algorithm gives the global optimal solution for a large subset of parameters in each step. Through experiments, we showed that the MU algorithm compares favorably with the standard VB iteration. We also demonstrated the usefulness of the flexibility of SAMF in a real-world foreground/background video separation experiment, where image segmentation is used for automatically designing a SMF term.

Acknowledgments

The authors thank anonymous reviewers for their suggestions, which improved the paper, and will improve its journal version. Shinichi Nakajima and Masashi Sugiyama thank the support from Grant-in-Aid for Scientific Research on Innovative Areas: Prediction and Decision Making, 23120004. S. Derin Babacan was supported by a Beckman Postdoctoral Fellowship.

References

A. Asuncion and D.J. Newman. UCI machine learning repository, 2007. URL http://www.ics.uci.edu/~mlearn/MLRepository.html.

S. D. Babacan, M. Luessi, R. Molina, and A. K. Katsaggelos. Sparse Bayesian methods for low-rank matrix estimation. *IEEE Trans. on Signal Processing*, 60(8):3964–3977, 2012.

C. M. Bishop. Variational principal components. In *Proc. of ICANN*, volume 1, pages 514–509, 1999.

E. J. Candes, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *CoRR*, abs/0912.3599, 2009.

X. Ding, L. He, and L. Carin. Bayesian robust principal component analysis. *IEEE Transactions on Image Processing*, 20(12):3419–3430, 2011.

B. Efron and C. Morris. Stein's estimation rule and its competitors—an empirical Bayes approach. *Journal of the American Statistical Association*, 68:117–130, 1973.

P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.

H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24:417–441, 1933.

A. Ilin and T. Raiko. Practical approaches to principal component analysis in the presence of missing values. *JMLR*, 11:1957–2000, 2010.

Y. J. Lim and T. W. Teh. Variational Bayesian approach to movie rating prediction. In *Proceedings of KDD Cup and Workshop*, 2007.

Z. Lin, M. Chen, L. Wu, and Y. Ma. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. *University of Illinois at Urbana-Champaign, Tech. Rep.*, 2010.

S. Nakajima and M. Sugiyama. Theoretical analysis of Bayesian matrix factorization. *Journal of Machine Learning Research*, 12:2579–2644, 2011.

S. Nakajima, M. Sugiyama, and S. D. Babacan. Global solution of fully-observed variational Bayesian matrix factorization is column-wise independent. In *Advances in Neural Information Processing Systems 24*, 2011.

R. M. Neal. *Bayesian Learning for Neural Networks.* Springer, 1996.

R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. In J. C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 1257–1264, Cambridge, MA, 2008. MIT Press.

S. Watanabe. *Algebraic Geometry and Statistical Learning.* Cambridge University Press, Cambridge, UK, 2009.