PROBABILISTIC SIMPLEX COMPONENT ANALYSIS VIA VARIATIONAL AUTO-ENCODING

Yuening Li[⋆]*, Xiao Fu*[†]*, Wing-Kin Ma*[⋆]

[⋆]Department of EE, The Chinese University of Hong Kong
[†]School of EECS, Oregon State University

ABSTRACT

Simplex component analysis (SCA) aims to estimate the vertices of the convex hull where data samples reside in. SCA finds various applications in signal processing, *e.g.*, hyperspectral unmixing and noisy label learning. Recent works proposed to tackle SCA from a probabilistic viewpoint using variational inference (VI) tools, which fends against noise more effectively relative to the deterministic counterparts. However, the computational efficiency of VI for SCA hinges on the use of the Dirichlet variational posterior. Such variational posterior appears to lack expressiveness—making the SCA performance limited if the true posterior is complex. This work proposes to employ a logistic-normal variational posterior, which exhibits enhanced expressive power. To circumvent the computational bottleneck, a neural representation-based inference algorithm is proposed—which exploits a connection between the logistic-normal distribution and variational auto-encoding. Numerical experiments using simulated and semi-real data are conducted to showcase the effectiveness of our algorithm design.

Index Terms— Simplex component analysis, maximum likelihood, variational auto-encoder, hyperspectral unmixing

1. INTRODUCTION

Simplex component analysis (SCA) aims to identify the latent factors of a structured linear mixture model. There, the data samples reside in a simplex whose vertices are the columns of the basis matrix. SCA arises in a wide spectrum of applications, *e.g.,* hyperspectral unmixing [1, 2], spectrum sensing [3], and noisy label learning [4, 5]; see more in [6, 7].

Theories and methods of SCA have been heavily studied in the literature. A large number of approaches take a deterministic viewpoint—treating SCA as a structured matrix factorization problem; see, *e.g.,* [8–12]. Many of these methods have elegant identifiability guarantees of the sought latent factors [6]. However, these approaches often require nontrivial care to handle noise via regularization design and hyper-parameter tuning. Probabilistic SCA was also considered in the literature; see, *e.g.,* [13–18]. These methods are often more versatile in dealing with uncertainties brought by noise—*e.g.,* via choosing proper priors to avoid agnostic regularization parameter tuning. However, many of the probabilistic methods (*e.g.,* [13, 14]) resort to Bayesian inference using computationally demanding algorithms (*e.g.,* Markov chain Monte Carlo (MCMC) methods) to accommodate the structural constraints of SCA. Identifiability of the target latent factors was also unclear in early developments of probabilistic SCA.

Recently, an alternative probabilistic framework advocated using a maximum marginal likelihood (MML) viewpoint to tackle SCA [16–18]. Unlike the early Bayesian SCA methods, the MML-based approach was shown to admit asymptotic identifiability of the latent factors of the SCA model [17]. More importantly, the MML formulation allows to design a computationally lightweight *variational inference algorithm* (VIA)—by carefully choosing the Dirichlet distribution as the variational posterior. However, as a trade-off, the use of such a Dirichlet variational posterior also appears to be the limiting factor of the approach's modeling power. The reason is that the Dirichlet distribution has limited tunable parameters and thus lacks expressiveness. Hence, the ground-truth posterior may not be approximated well by the variational posterior, which could degrade the VIA's performance.

This work revisits the MML framework for SCA and proposes to employ the *logistic-normal* (LN) distribution as the variational posterior. Notably, the LN and Dirichlet distributions both model random vectors whose supports are the probability simplex, but the former admits more tunable parameters, leading to arguably stronger expressiveness. Nonetheless, using the LN distribution makes the algorithmic structure in [17] no longer valid—which relied on the Dirichlet distribution for efficiency. To overcome this challenge, we propose to exploit a connection between the LN distribution and the *variational auto-encoder* (VAE) [19]. Using the reparameterization trick and neural representations of the key parameters of the LN model, we design an efficient algorithm to approximate the MML. We show the effectiveness of our algorithm using numerical examples in hyperspectral unmixing.

2. BACKGROUND

SCA considers the following linear mixture model (LMM):

$$\boldsymbol{y}_t = \boldsymbol{A}\boldsymbol{s}_t + \boldsymbol{v}_t, \quad t = 1, \ldots, T, \tag{1}$$

where $\boldsymbol{s}_t \in \Delta := \{\boldsymbol{s} \in \mathbb{R}^N_+ | \mathbf{1}^\top \boldsymbol{s} = 1\}$, $\boldsymbol{A} \in \mathbb{R}^{M\times N}$ is a full column-rank basis matrix, and $\boldsymbol{v}_t$ represents the noise. Geometrically, Eq. (1) means that $\boldsymbol{y}_t$ resides in a simplex whose vertices are $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N$, if $\boldsymbol{v}_t$ is absent. SCA aims to identify the unknown latent factors $\boldsymbol{A}$ and $\{\boldsymbol{s}_t\}_{t=1}^T$ from the data samples $\{\boldsymbol{y}_t\}_{t=1}^T$.

SCA naturally arises in many signal processing and machine learning applications. For example, SCA has been widely used in hyperspectral unmixing [1, 2], wherein each pixel of a hyperspectral image (*i.e.,* $\boldsymbol{y}_t$) is modeled as a weighted combination of spectral signatures (*endmembers*) of different materials (*i.e.,* $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N$). The elements of the combination coefficient vector $\boldsymbol{s}_t$ correspond to the *abundances* of the endmembers in pixel $\boldsymbol{y}_t$. In topic model-

The work of Y. Li and W.-K. Ma was supported by a General Research Fund (GRF) of Hong Kong Research Grant Council (RGC) under Project ID CUHK 14203721. The work of X. Fu was supported in part by the National Science Foundation CAREER Award under Project ECCS-2144889.

ing [10, 20], $\boldsymbol{y}_t$, $\boldsymbol{a}_i$ and $\boldsymbol{s}_t$ stand for the term-frequency representation of the tth document, the probability mass function of the ith topic, and the proportions of the topics constituting the tth document, respectively; see more applications in [6, 7].

2.1. Existing SCA Approaches

Many early SCA approaches took a deterministic viewpoint. This viewpoint typically leads to exploitation of the convex geometry in (1) and (implicit or explicit) matrix factorization (MF)-based formulations and algorithms [1, 2]. Some of these methods have algebraically simple algorithmic structures and elegant identifiability analysis, under relatively ideal conditions, *e.g.,* that the so-called pure pixels exist and that the noise is absent [21–24]. When more critical scenarios are considered, a typical formulation is as follows:

$$\min_{\boldsymbol{A},\boldsymbol{s}_t\in\Delta} \|\boldsymbol{Y}-\boldsymbol{A}\boldsymbol{S}\|_F^2+\lambda\cdot r_1(\boldsymbol{A})+\mu\cdot r_2(\boldsymbol{S}), \tag{2}$$

where $\boldsymbol{S}=[\boldsymbol{s}_1,\ldots,\boldsymbol{s}_T]$, and $r_i(\cdot)$ for $i=1,2$ are regularization terms driven by different purposes (*e.g.,* promoting model identifiability and/or enhancing physical interpretability of the estimated $\boldsymbol{A}$ and $\boldsymbol{S}$). The regularization can take various forms; see, *e.g.,* [9, 10, 25, 26]. Such MF methods can be effective, but often involve nontrivial efforts to choose regularizers and to tune the associated parameters.

Probabilistic SCA was also considered in the literature [13–15]. The probabilistic approaches can flexibly incorporate noise. More importantly, these methods can sidestep the agnostic pain of parameter tuning via choosing proper priors for the latent factors and the noise. However, a notable downside of the classical probabilistic SCA approaches lies in their heavy computational load, as they often use resource–demanding Bayesian inference methods (*e.g.,* MCMC) to accommodate the structural constraints of SCA [13, 14]. Moreover, the early Bayesian inference based SCA developments rarely consider identifiability—as the notion of identifiability is not clearly defined in this line of work. The more recent probabilistic SCA approaches [16–18] took an MML perspective. This framework treats $\boldsymbol{A}$ as a deterministic parameter and assigns priors to $\boldsymbol{s}_t$ and $\boldsymbol{v}_t$, *i.e.,*

$$\boldsymbol{y}_t=\boldsymbol{A}\boldsymbol{s}_t+\boldsymbol{v}_t,\quad \boldsymbol{s}_t\sim D(\boldsymbol{1}),\ \boldsymbol{v}_t\sim\mathcal{N}(\boldsymbol{0},\sigma_v^2\boldsymbol{I}). \tag{3}$$

Here, $D(\boldsymbol{1})=(N-1)!\cdot\mathbb{1}_\Delta(\boldsymbol{s})$ denotes the uniform Dirichlet distribution, in which $\mathbb{1}_\Delta(\cdot)$ is the indicator function of the probability simplex. Under this setting, the SCA problem can be tackled via solving the MML estimator:

$$\widehat{\boldsymbol{A}}=\arg\max_{\boldsymbol{A}}\frac{1}{T}\sum_{i=1}^{T}\log p(\boldsymbol{y}_t;\boldsymbol{A}), \tag{4}$$

where $p(\boldsymbol{y}_t;\boldsymbol{A})=\int p(\boldsymbol{y}_t|\boldsymbol{s}_t;\boldsymbol{A})p(\boldsymbol{s}_t)d\boldsymbol{s}_t$ is the marginal likelihood. Like the classical probabilistic SCA methods, the MML approaches spare the efforts of manually picking regularization terms and hand-tuning the regularization parameters. In addition, it was shown in [17] that solving (4) identifies the ground-truth $\boldsymbol{A}$ up to column permutations, under the conditions that infinite observations are available. This identifiability result is much desired in the context of probabilistic SCA—as it attests to the soundness and consistency of the estimator, even in the presence of noise.

2.2. Challenges

In [17], a VIA was designed to handle the MML in (4). The idea is to find a "tractable" lower bound of $\log p(\boldsymbol{y}_t;\boldsymbol{A})$ by introducing a surrogate posterior $q_t(\boldsymbol{s}_t)$ (which is also called the *variational posterior*). By the Jensen inequality, we have

$$\begin{aligned}\log p(\boldsymbol{y}_t;\boldsymbol{A})&=\log\mathbb{E}_{\boldsymbol{s}_t\sim q_t}[p(\boldsymbol{y}_t|\boldsymbol{s}_t;\boldsymbol{A})p(\boldsymbol{s}_t)/q_t(\boldsymbol{s}_t)]\\&\geq\mathbb{E}_{\boldsymbol{s}_t\sim q_t}[\log p(\boldsymbol{y}_t,\boldsymbol{s}_t;\boldsymbol{A})-\log q_t(\boldsymbol{s}_t)]\\&:=\widehat{\ell}(\boldsymbol{A},q_t;\boldsymbol{y}_t).\end{aligned} \tag{5}$$

Then, a lower-bounding surrogate function of the MML is

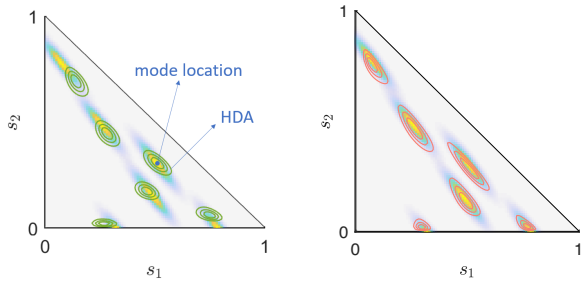
$$\max_{\boldsymbol{A},q_t\in\mathcal{D},\forall t}\mathcal{L}_T(\boldsymbol{A},\{q_t\}):=\frac{1}{T}\sum_{t=1}^{T}\widehat{\ell}(\boldsymbol{A},q_t;\boldsymbol{y}_t), \tag{6}$$

where $\mathcal{D}$ is a chosen family of distributions with support Δ. In [17], the variational posterior family $\mathcal{D}$ was chosen to be the Dirichlet distribution. This way, q_t has analytical moments, thereby making the computation of (6) tractable.

However, the choice of $\mathcal{D}$ also brings challenges. As mentioned, $q_t\in\mathcal{D}$ acts as the *variational posterior*, which is supposed to tightly approximate the true posterior $p(\boldsymbol{s}_t|\boldsymbol{y}_t;\boldsymbol{A})$. This is because the equality in (5) holds if and only if

$$q_t(\boldsymbol{s}_t)=p(\boldsymbol{s}_t|\boldsymbol{y}_t;\boldsymbol{A})\propto\varphi_{\sigma_v}(\boldsymbol{y}_t-\boldsymbol{A}\boldsymbol{s}_t)\mathbb{1}_\Delta(\boldsymbol{s}_t), \tag{7}$$

where φ_{σ_v} denotes a zero-mean Gaussian distribution with variance σ_v^2, and $\mathbb{1}_\Delta(\boldsymbol{s})$ is the indicator function as defined before. Selecting $\mathcal{D}$ as the Dirichlet family faces challenges in attaining/approximating (7). To explain, recall that for the Dirichlet distribution $D(\boldsymbol{\alpha})$, the mean and the covariance are both determined by $\boldsymbol{\alpha}$ [27]. Hence, the shape of the high-density area (HDA) of the distribution (related to the covariance) and the mode location (related to the mean) are dependent—both determined by $\boldsymbol{\alpha}$. As $\boldsymbol{\alpha}$ is the only tunable parameter, the variational posterior lacks flexibility/expressiveness to effectively match the ground-truth posterior in (7), especially when the HDA of $p(\boldsymbol{s}_t|\boldsymbol{y}_t;\boldsymbol{A})$ exhibits complex shapes; see illustrations of the HDAs and mode locations of the variational and ground-truth posteriors in Fig. 1(a). Such lack of expressiveness of $D(\boldsymbol{\alpha})$ turns out to be the major performance-limiting factor of the VIA in [17].



(a) Dirichlet q_t (PRISM) (b) LN q_t (Proposed)

Fig. 1: Posterior matching using Dirichlet q_t (left) and LN q_t (right) in an $N=3$ case for six different $\boldsymbol{y}_t$'s. HDA of the ground-truth posterior: colored contour in solid lines. More settings are in Sec. 4.

3. PROPOSED APPROACH

Our interest lies in enhancing the performance of the MML-based probabilistic SCA. To this end, we propose to replace the Dirichlet variational posterior in [17] by the LN distribution [28, 29]:

$$q_t\in\mathcal{D}:=\{q=\mathcal{LN}(\boldsymbol{\mu},\boldsymbol{\Sigma})\}, \tag{8}$$

where $\mathcal{LN}(\boldsymbol{s};\boldsymbol{\mu},\boldsymbol{\Sigma})=\frac{1}{|2\pi\boldsymbol{\Sigma}|^{1/2}}\frac{1}{\prod_{i=1}^{N}s_i}\exp(-\frac{1}{2}\{\log(\frac{\boldsymbol{s}_{-N}}{s_N})-\boldsymbol{\mu}\}^\top\boldsymbol{\Sigma}^{-1}\{\log(\frac{\boldsymbol{s}_{-N}}{s_N})-\boldsymbol{\mu}\})$ is the PDF of the $(\boldsymbol{\mu},\boldsymbol{\Sigma})$-parameterized LN distribution, with $\boldsymbol{s}_{-N}$ being a subvector containing the first

$N-1$ elements of $\boldsymbol{s}$. Here, we use diagonal covariance $\boldsymbol{\Sigma} = \mathrm{Diag}(\boldsymbol{\sigma})$. The support of the LN distribution is the probability simplex [28, 29]—which is the same as that of the Dirichlet distribution. Unlike $D(\boldsymbol{\alpha})$, which is controlled by a single parameter vector $\boldsymbol{\alpha} \in \mathbb{R}^N$, the LN distribution is parameterized by two sets of parameters (*i.e.,* $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$). This helps represent the variational posterior with various HDA shapes and mode locations in a flexible way, as the mean and variance do not affect each other. Consequently, $q_t \in \mathcal{LN}$ exhibits enhanced expressiveness and can better approximate complex ground-truth posteriors; see Fig. 1(b). Such enhanced expressiveness is expected to improve the performance of the VIA in the presence of complex $p(\boldsymbol{s}_t|\boldsymbol{y}_t;\boldsymbol{A})$.

3.1. Computational Bottleneck

Recall that the VIA approach in [17] hinged on the structure of $D(\boldsymbol{\alpha})$ to come up with a lightweight algorithm for solving (6). This is because using $\mathcal{D} = D(\boldsymbol{\alpha})$ allows to express $\widehat{\ell}$ (and thus $\mathcal{L}_T$) in (6) with an analytical form; see [17] for details. The use of $\mathcal{D} = \mathcal{LN}$ breaks down the nice algorithmic structure—and thus a different algorithm for handling the MML is needed.

In [17], a brute-force method was also used to evaluate $\mathcal{L}_T$ in the cases where $\mathcal{D}$ is not specified. There, the *importance sampling approximation* (ISA) approach was employed. ISA approximates $\widehat{\ell}$ using a large amount of independent samples, *i.e.,*

$$\widehat{\ell}(\boldsymbol{A}, q_t; \boldsymbol{y}_t) \approx \frac{1}{R}\sum_{r=1}^{R}\big(\log p(\boldsymbol{y}_t, \boldsymbol{s}_t^r; \boldsymbol{A}) - \log q_t(\boldsymbol{s}_t^r)\big), \quad (9)$$

where $\{\boldsymbol{s}_t^r\}_{r=1}^R$ is drawn from a distribution q_t, typically $q_t \propto \varphi_{\sigma_v}(\boldsymbol{y}_t - \boldsymbol{A}\boldsymbol{s}_t)\mathbb{1}_\Delta(\boldsymbol{s}_t)$. This can be realized using rejection sampling or the MCMC methods [30, 31]. However, generating useful samples under structural constraints of SCA is a highly nontrivial task. For example, rejection sampling tends to reject more than 99.9% of the samples when $N > 10$ [17], making this approach infeasible even for a moderate N. In the next subsection, we will show that using $\mathcal{D} = \mathcal{LN}$ together with a trick from the VAE literature allows us to approximate (9) in a sample-efficient way.

3.2. Probabilistic SCA via VAE

Our approach starts by noticing that the $\mathcal{LN}$ class is parameterized by $(\boldsymbol{\mu}, \boldsymbol{\Sigma})$; see (8). These are the same parameters as those of the multivariate Gaussian distribution. This structure allows us to use the reparameterization trick that originated from the VAE literature [19] to simplify computation. To see this, recall that we aim to approximate $\widehat{\ell}$ by (9). Our idea is to use the following representation of the samples of q_t:

$$\boldsymbol{s}_t^r = g(\boldsymbol{\mu}_t + \boldsymbol{\sigma}_t \odot \boldsymbol{\epsilon}_t^r), \quad \boldsymbol{\epsilon}_t^r \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), \quad (10)$$

where $g(\boldsymbol{z}) = \frac{1}{1+\sum_{i=1}^{N-1} e^{z_i}}[e^{\boldsymbol{z}}; 1] = \mathrm{softmax}([\boldsymbol{z}; 0])$ denotes the additive logistic transformation. Such a transformation of $\boldsymbol{s}$ from an auxiliary variable $\boldsymbol{\epsilon}$ with simple distribution is termed as *reparameterization* [19]. The benefit of doing so is twofold: First, the reparameterized $\boldsymbol{s}_t^r$ is a differentiable function w.r.t. all the key parameters $\boldsymbol{\mu}_t$ and $\boldsymbol{\sigma}_t$, which will be useful for designing simple gradient descent-type algorithms. Second, the reparameterization allows us to sample $\boldsymbol{s}_t^r$ via sampling the Gaussian variable $\boldsymbol{\epsilon}_t^r$—without violating the structural constraints on $\boldsymbol{s}_t^r$. That is, using such logistic transformation g, the sampled $\boldsymbol{s}_t^r$ still lies in the probability simplex and follows the LN distribution [28]. This makes generating legitimate samples within the probability simplex fairly easy.

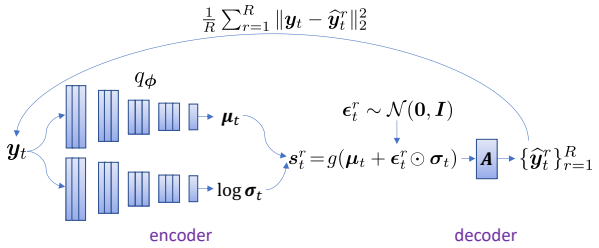


Fig. 2: Neural network-based implementation of VASCA.

Algorithm 1 VASCA

Input: $\boldsymbol{Y}, \sigma_v^2, N$
initialize $\boldsymbol{A}$ and $\boldsymbol{\phi}$
for $epoch = 1, \ldots,$ num_epoch **do**
  for $i = 1, \ldots,$ num_batch **do**
    $\boldsymbol{Y}^i \leftarrow i$-th minibatch of $\boldsymbol{Y}$
    draw R samples $\boldsymbol{\epsilon}_t^r$ from $\mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), \forall t$
    calculate loss $\mathcal{L}_{T_i}(\boldsymbol{A}, q_{\boldsymbol{\phi}}; \boldsymbol{Y}^i)$
    update $(\boldsymbol{A}, \boldsymbol{\phi})$ using *Adam* optimizer
  end for
end for
Output: $\boldsymbol{A}, \boldsymbol{\phi}$

To attain an efficient algorithm, we use a neural representation for q_t, *i.e.,*

$$q_t = q_{\boldsymbol{\phi}}(\boldsymbol{y}_t) = \mathcal{LN}\big(\boldsymbol{\mu}_{\boldsymbol{\phi}}(\boldsymbol{y}_t), \boldsymbol{\Sigma}_{\boldsymbol{\phi}}(\boldsymbol{y}_t)\big), \quad (11)$$

where $\boldsymbol{\phi}$ collects all the network parameters, and $\boldsymbol{\mu}_{\boldsymbol{\phi}}(\boldsymbol{y}_t), \boldsymbol{\Sigma}_{\boldsymbol{\phi}}(\boldsymbol{y}_t)$ are two neural networks mapping $\boldsymbol{y}_t$ to its "generating parameters". This mapping is often called the *encoder* in the literature. Using neural networks to represent the encoder is considered reasonable, as the neural networks are universal function representers. The encoder enables network parameter-sharing across all $\boldsymbol{y}_t$'s, which reduces the computational load and sample complexity.

With the reparameterization and neural representation, the expression of $\widehat{\ell}$ in (9) is as follows:

$$\widehat{\ell}(\boldsymbol{A}, q_{\boldsymbol{\phi}}; \boldsymbol{y}_t) = \frac{1}{R}\sum_{r=1}^{R}\left(-\frac{1}{2\sigma_v^2}\|\boldsymbol{y}_t - \widehat{\boldsymbol{y}}_t^r\|_2^2 + H_{\boldsymbol{\phi}}(\boldsymbol{s}_t^r)\right) + C, \quad (12)$$

where $\widehat{\boldsymbol{y}}_t^r = \boldsymbol{A}\boldsymbol{s}_t^r$, $H_{\boldsymbol{\phi}}(\boldsymbol{s}_t^r) = \big((\tilde{\boldsymbol{s}}_t^r)^\top \boldsymbol{\Sigma}_t^{-1}\tilde{\boldsymbol{s}}_t^r + \boldsymbol{1}^\top \log \boldsymbol{\sigma}_t\big)/2 + \boldsymbol{1}^\top \log \boldsymbol{s}_t^r$; $\boldsymbol{\Sigma}_t^{-1} = \mathrm{Diag}(\frac{1}{\boldsymbol{\sigma}_t})$, $\tilde{\boldsymbol{s}}_t^r = \log(\frac{[\boldsymbol{s}_t^r]_{-N}}{[\boldsymbol{s}_t^r]_N}) - \boldsymbol{\mu}_t$, and $C = \frac{N-1}{2}\log(2\pi) - \frac{M}{2}\log(2\pi\sigma_v^2) + \log\Gamma(N)$ is a constant. This loss function gives rise to a special neural network structure as shown in Fig. 2. The structure is similar to the celebrated VAE in neural representation learning [19]. Like VAE, our neural network architecture has an encoding part that maps $\boldsymbol{y}_t$ to a latent distribution and a decoding part that reconstructs $\widehat{\boldsymbol{y}}_t$ from the latent domain. Unlike the classical VAE, our latent embeddings (*i.e.,* $\boldsymbol{s}_t^r$) are generated using the LN distribution other than the Gaussian distribution—due to the probability simplex constraint. More importantly, the classical VAE's encoder and decoder are both represented by neural networks, but our decoder only has a linear operator $\boldsymbol{A}$. This is because in probabilistic SCA the decoder part corresponds to $\widehat{\boldsymbol{y}}_t^r = \boldsymbol{A}\boldsymbol{s}_t^r$, and is induced by the $\log p(\boldsymbol{y}_t, \boldsymbol{s}_t^r; \boldsymbol{A})$ term in (9).

The loss function (12) can be tackled using any off-the-shelf neural network optimizers, *e.g.,* the *Adam* optimizer [32], since the first-order derivative (or sub-derivative, depending on the neural network architecture) of the objective function exists under the reparameterization and neural representation; see Algorithm 1, which is referred to as the VAE-based SCA (VASCA) algorithm.

Table 1: Runtime (sec.) of the schemes in synthetic experiments.

SNR (dB)	10	15	20	25	30	35
PRISM	62.3	69.1	77.6	84.1	86.3	86.8
VASCA	23.4	23.2	22.2	19.3	17.2	16.1

4. NUMERICAL RESULTS

Synthetic Data Experiment. We first consider a simulation of hyperspectral unmixing. The $\boldsymbol{y}_t$'s (*i.e.,* pixels) are generated following (3), with $(M, T) = (50, 10000)$ and $N = 3$. The endmember matrix $\boldsymbol{A}$ is randomly selected from the USGS library [33]. We generate $\boldsymbol{s}_t$ from the uniform Dirichlet distribution. We remove the columns with elements larger than 0.8 to make the unmixing problem more challenging. Zero-mean white Gaussian noise is added.

We use SVMAX [34] and PRISM [17] as baselines. For PRISM, we employ *Adam* optimizer with a learning rate 0.05. For VASCA, we use five equal-size minibatches and set the learning rate of $\boldsymbol{A}, \boldsymbol{\phi}$ as 0.001 and 0.01, respectively. We use two separate neural networks to act as $\boldsymbol{\mu}_{\boldsymbol{\phi}}(\boldsymbol{y}_t)$ and $\boldsymbol{\Sigma}_{\boldsymbol{\phi}}(\boldsymbol{y}_t)$. The two networks both have four fully connected (FC) layers. The layers are all followed by batch normalization. We use the ReLU activation function at the output of each layer. The numbers of activation functions in the four encoder layers are $32, 32, 16$ and 8, respectively. The output of the last layer is then reduced to $N-1$ dimensions (*i.e.,* the dimensions of $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$) using a linear combination. To implement VASCA, we use $R = 1$, *i.e.,* the most sample-efficient setting. PRISM and VASCA are stopped if the relative change of $\boldsymbol{A}$ is less than 10^{-4} or the maximum numbers of epochs exceed 300 and 500, respectively.[1] We initialize $\boldsymbol{A}$ of all algorithms by SVMAX. To measure the accuracy of estimating $\boldsymbol{A}$, we adopt two metrics, namely, the spectral angle mapper (SAM) and the mean square error (MSE); see [18].

Fig. 1 on Page 2 visualizes posterior approximation by PRISM and VASCA, respectively. Here, SNR=15dB. Six $\boldsymbol{y}_t$'s posterior distributions are chosen for illustration. One can see that LN offers a visually much more accurate estimation of the ground-truth posterior, especially for the covariance (*i.e.,* the ellipsoidal shapes in the figures). This is consistent with our postulate that the LN distribution is more expressive than the Dirichlet variational posterior.

Fig. 3 shows the average MSE and SAM of the estimated $\boldsymbol{A}$ over 100 Monte Carlo trials. The proposed algorithm exhibits the most favorable performance in terms of estimating $\boldsymbol{A}$. Specifically, VASCA outperforms PRISM by visible margins in both metrics under all SNRs under test. This shows the benefit of enhanced expressing power in our model.

Table 1 shows the runtime performance of PRISM and VASCA in various SNR cases. One can see that VASCA is about 3∼5 times faster than PRISM, showing the efficiency brought by using the reparameterization and neural representation.

Semi-Real Data Experiment. Next, we consider a semi-real data experiment. Our setting follows that in [35]. Specifically, we take the five endmembers and abundance map extracted from a real hyperspectral image, Urban data, as the ground truth $\boldsymbol{A}$ and $\boldsymbol{S}$. As in the simulations, we truncate the elements of $\boldsymbol{S}$ so that every entry is smaller than or equal to 0.8. Then, we convolve each abundance map using a 9×9 Gaussian blurring kernel with variance 2^2 to enhance visual realism. The SNR is set to be 25dB via adding zero-mean Gaussian noise. The data size is $(M, T) = (162, 307^2)$. The algorithm settings are mostly identical to those in the simulations. The learning rates are set as $(0.01, 0.005)$ for $(\boldsymbol{A}, \boldsymbol{\phi})$ in VASCA. The numbers of activation functions of the encoder of VASCA are set to be $128, 64, 32, 16$ for the four layers, respectively. Moreover, we combine FCLS [36] with SVMAX to estimate $\boldsymbol{S}$. For the probabilistic methods PRISM and VASCA, we use the variational posterior mean as the estimation $\widehat{\boldsymbol{s}}_t$.

[1] Note that running 300 epochs of PRISM costs much more computational resource than that of running 500 epochs of VASCA, as PRISM performs at least 30 inner-updates for its variational parameters using a subroutine.

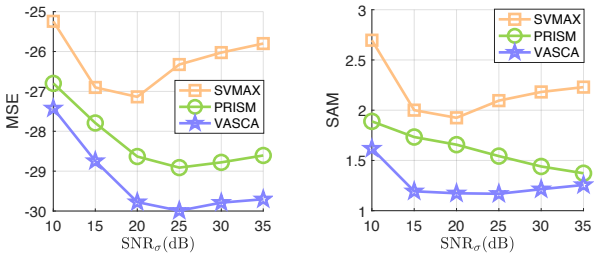


Fig. 3: Synthetic data experiment. The average MSEs (left) and SAMs (right) of the estimated $\boldsymbol{A}$ by the algorithms.

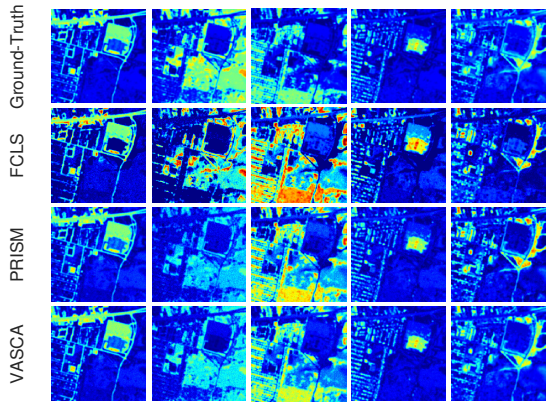


Fig. 4: The ground-truth and estimated abundance maps of the semi-real *Urban* image.

Table 2: The performance of the schemes on semi-real data.

Algorithm	VCA+FCLS	PRISM	VASCA
MSE$_{\boldsymbol{A}}$ ↓	-22.48	-26.32	**-32.54**
RMSE$_{\boldsymbol{s}}$ ↓	0.127	0.078	**0.053**
Time (sec.) ↓	**2.3**	115.5	24.0

Fig. 4 shows the ground-truth and estimated abundance maps of the five materials. Table 2 shows the quantitative evaluation using MSE and RMSE (root MSE, defined as $\frac{1}{T}\sum_{t=1}^{T}\sqrt{\|\boldsymbol{s}_t - \widehat{\boldsymbol{s}}_t\|_2^2/N}$ after fixing permutation) of the estimated $\boldsymbol{A}$ and $\boldsymbol{S}$, respectively. We see that VASCA gives better estimations of both $\boldsymbol{A}$ and $\boldsymbol{S}$ compared with PRISM. In terms of runtime, VASCA is again about 5 times faster than PRISM.

5. CONCLUSION

In this work, we proposed a new method to enhance the expressiveness of the MML-based SCA approach. To be specific, we advocated using the LN distribution (instead of the Dirichlet distribution) based variational posterior to attain more degrees of freedom. Leveraging a connection between the LN distribution and VAE, we proposed a reparameterization and neural representation based implementation to carry out the computation in a sample-efficient way. Numerical results suggested that the proposed scheme is promising for tackling the SCA problem in terms of both accuracy and runtime.

6. REFERENCES

[1] J. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, and J. Chanussot, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE J. Sel. Topics Appl. Earth Observ.*, vol. 5, no. 2, pp. 354–379, 2012.

[2] W. K. Ma, J. M. Bioucas-Dias, T. H. Chan, N. Gillis, P. Gader, A. J. Plaza, A. Ambikapathi, and C. Y. Chi, "A signal processing perspective on hyperspectral unmixing," *IEEE Signal Process. Mag.*, vol. 31, no. 1, pp. 67–81, Jan 2014.

[3] X. Fu, N. D. Sidiropoulos, and W.-K. Ma, "Power spectra separation via structured matrix factorization," *IEEE Trans. Signal Process.*, vol. 64, no. 17, pp. 4592–4605, 2016.

[4] T. Nguyen, S. Ibrahim, and X. Fu, "Deep clustering with incomplete noisy pairwise annotations: A geometric regularization approach," *arXiv preprint arXiv:2305.19391*, 2023.

[5] S. Ibrahim, T. Nguyen, and X. Fu, "Deep learning from crowdsourced labels: Coupled cross-entropy minimization, identifiability, and regularization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2022.

[6] X. Fu, K. Huang, N. D. Sidiropoulos, and W.-K. Ma, "Nonnegative matrix factorization for signal and data analytics: Identifiability, algorithms, and applications," *IEEE Signal Process. Mag.*, vol. 36, no. 2, pp. 59–80, 2019.

[7] N. Gillis, *Nonnegative Matrix Factorization*. SIAM, 2020.

[8] N. Gillis and S. A. Vavasis, "Fast and robust recursive algorithms for separable nonnegative matrix factorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 4, pp. 698–714, 2014.

[9] L. Miao and H. Qi, "Endmember extraction from highly mixed data using minimum volume constrained nonnegative matrix factorization," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 3, pp. 765–777, 2007.

[10] X. Fu, K. Huang, B. Yang, W.-K. Ma, and N. D. Sidiropoulos, "Robust volume minimization-based matrix factorization for remote sensing and document clustering," *IEEE Trans. Signal Process.*, vol. 64, no. 23, pp. 6254–6268, 2016.

[11] N. Gillis and R. Luce, "Robust near-separable nonnegative matrix factorization using linear optimization," *J. Mach. Learn. Res*, vol. 15, no. 1, pp. 1249–1280, 2014.

[12] T. Nguyen, X. Fu, and R. Wu, "Memory-efficient convex optimization for self-dictionary separable nonnegative matrix factorization: A frank–wolfe approach," *IEEE Trans. Signal Process.*, vol. 70, pp. 3221–3236, 2022.

[13] N. Dobigeon, S. Moussaoui, M. Coulon, J.-Y. Tourneret, and A. O. Hero, "Joint Bayesian endmember extraction and linear unmixing for hyperspectral imagery," *IEEE Trans. Signal Process.*, vol. 57, no. 11, pp. 4355–4368, 2009.

[14] N. Dobigeon, S. Moussaoui, J.-Y. Tourneret, and C. Carteret, "Bayesian separation of spectral sources under non-negativity and full additivity constraints," *Signal Process.*, vol. 89, no. 12, pp. 2657–2669, 2009.

[15] J. Nascimento and J. Bioucas-Dias, "Hyperspectral unmixing based on mixtures of Dirichlet components," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 3, pp. 863–878, 2012.

[16] R. Wu, Q. Li, and W.-K. Ma, "Stochastic ML simplex-structured matrix factorization under the Dirichlet mixture model," in *2019 IEEE Int. Conf. Acoustics Speech Signal Process. (ICASSP)*, 2019, pp. 5561–5565.

[17] R. Wu, W.-K. Ma, Y. Li, A. M.-C. So, and N. D. Sidiropoulos, "Probabilistic simplex component analysis," *IEEE Trans. Signal Process.*, 2021.

[18] Y. Li, W.-K. Ma, and N. D. Sidiropoulos, "Robust probabilistic simplex component analysis," in *2021 IEEE Workshop Stat. Signal Process. (SSP)*, 2021, pp. 456–460.

[19] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, vol. 1050, 2014, p. 1.

[20] K. Huang, X. Fu, and N. D. Sidiropoulos, "Anchor-free correlated topic modeling: Identifiability and algorithm," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2016, pp. 1786–1794.

[21] T.-H. Chan, W.-K. Ma, A. Ambikapathi, and C.-Y. Chi, "A simplex volume maximization framework for hyperspectral endmember extraction," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 11, pp. 4177–4193, 2011.

[22] J. M. Nascimento and J. M. Dias, "Vertex component analysis: A fast algorithm to unmix hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 4, pp. 898–910, 2005.

[23] T.-H. Chan, C.-Y. Chi, Y.-M. Huang, and W.-K. Ma, "A convex analysis based minimum-volume enclosing simplex algorithm for hyperspectral unmixing," *IEEE Trans. Signal Process.*, vol. 57, no. 11, pp. 4418–4432, 2009.

[24] J. Bioucas-Dias, "A variable splitting augmented lagrangian approach to linear spectral unmixing," in *Proc. 2009 Workshop Hyperspectral Image Signal Process.: Evol. Remote Sens. (WHISPERS)*, 2009, pp. 1–4.

[25] Y. Qian, S. Jia, J. Zhou, and A. Robles-Kelly, "Hyperspectral unmixing via $L_{1/2}$ sparsity-constrained nonnegative matrix factorization," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 11, pp. 4282–4297, 2011.

[26] A. Zare and P. Gader, "Sparsity promoting iterated constrained endmember detection in hyperspectral imagery," *IEEE Geosci. Remote Sens. Lett.*, vol. 4, no. 3, pp. 446–450, 2007.

[27] K. W. Ng, G.-L. Tian, and M.-L. Tang, *Dirichlet and Related Distributions: Theory, Methods and Applications*. John Wiley & Sons, 2011.

[28] J. Aitchison and S. M. Shen, "Logistic-normal distributions: Some properties and uses," *Biometrika*, vol. 67, no. 2, pp. 261–272, 1980.

[29] J. Aitchison, "The statistical analysis of compositional data," *J. R. Stat. Soc., B: Stat. Methodol.*, vol. 44, no. 2, pp. 139–160, 1982.

[30] Y. Altmann, S. McLaughlin, and N. Dobigeon, "Sampling from a multivariate Gaussian distribution truncated on a simplex: a review," in *2014 IEEE Workshop Stat. Signal Process. (SSP)*, 2014, pp. 113–116.

[31] Y. Cong, B. Chen, and M. Zhou, "Fast simulation of hyperplane-truncated multivariate normal distributions," *Bayesian Analysis*, vol. 12, no. 4, pp. 1017–1037, 2017.

[32] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015.

[33] R. N. Clark, G. A. Swayze, R. Wise, K. E. Livo, T. Hoefen, R. F. Kokaly, and S. J. Sutley, "USGS digital spectral library splib06a," *U.S. Geological Survey, Digital Data Series 231*, 2007.

[34] T.-H. Chan, W.-K. Ma, A. Ambikapathi, and C.-Y. Chi, "A simplex volume maximization framework for hyperspectral endmember extraction," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 11, pp. 4177–4193, 2011.

[35] M. Ding, X. Fu, and X.-L. Zhao, "Fast and structured block-term tensor decomposition for hyperspectral unmixing," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, 2023.

[36] D. C. Heinz and C.-I. Chang, "Fully constrained least squares linear spectral mixture analysis method for material quantification in hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 3, pp. 529–545, 2001.