Compressive Sensing on Manifolds Using a Nonparametric Mixture of Factor Analyzers: Algorithm and Performance Bounds

Minhua Chen, Jorge Silva, John Paisley, Chunping Wang, David Dunson, and Lawrence Carin, *Fellow, IEEE*

***Abstract*—Nonparametric Bayesian methods are employed to constitute a mixture of low-rank Gaussians, for data $\boldsymbol{x} \in \mathbb{R}^N$ that are of high dimension N but are constrained to reside in a low-dimensional subregion of $\mathbb{R}^N$. The number of mixture components and their rank are inferred automatically from the data. The resulting algorithm can be used for learning manifolds and for reconstructing signals from manifolds, based on compressive sensing (CS) projection measurements. The statistical CS inversion is performed analytically. We derive the required number of CS random measurements needed for successful reconstruction, based on easily-computed quantities, drawing on block-sparsity properties. The proposed methodology is validated on several synthetic and real datasets.**

***Index Terms*—Beta process, compressive sensing, Dirichlet process, low-rank Gaussian, manifold learning, mixture of factor analyzers, nonparametric Bayes.**

I. Introduction

COMPRESSIVE sensing (CS) theory [1] shows that if a signal may be sparsely rendered in some basis, it may be recovered perfectly based on a relatively small set of random-projection measurements. In practice most signals are *compressible* in an appropriate basis (*not* exactly sparsely represented), and in this case highly accurate CS reconstructions are realized based on such measurements. Recent work has extended the notion of sparsity-based CS to the more general framework of manifold-based CS [2]–[4]. In manifold-based CS, the signal is assumed to belong to a manifold, and low information content corresponds to low intrinsic dimension of the manifold. For a simple example (taken from [4]), consider signals generated by taking samples of a truncated and shifted Gaussian pulse, as shown in the left panel of Fig. 1. Although the ambient dimension (number of samples) of the signals is high, the only degree of freedom is the scalar shift a; therefore, the signals belong to a one-dimensional manifold. If we have a collection of signals, corresponding to different shifts, and project them onto, say, a random three-dimensional subspace, then the signals will, with high probability, form a twisting curve that does not self-intersect (right panel of Fig. 1). Thus, very few measurements are required to capture the characteristics of a signal on this manifold. We will return to this illustrative example in our experimental results. There are many more real examples of low-dimensional manifold signals, such as digit and face images (see, e.g., [5]).

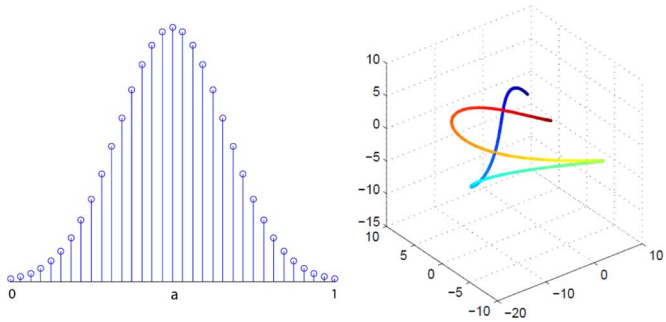


Fig. 1. Left: samples from a truncated and shifted Gaussian, with peak shift a. Right: projections of *multiple* such signals onto a random 3-D subspace, with the different points on the curve corresponding to different shifts a.

Although a theoretical analysis for CS on manifolds has been established in [2] and [3], very few algorithms exist for practical implementation. Moreover, existing performance guarantees depend on quantities that are not easily computable, such as the manifold condition number. In this paper, we propose a statistical framework for CS on manifolds, using a well-studied statistical model: a mixture of factor analyzers (MFA) [6], [7]. We model a manifold as a finite mixture of Gaussians, but we depart from conventional Gaussian mixture models (GMMs) by imposing a very particular structure—the covariances should be approximately low-rank, and the rank should equal the intrinsic dimension of the manifold. We employ nonparametric statistical methods [8], [9] to infer an appropriate number of mixture components for a given data set, as well as the associated rank of the Gaussians.

This model class is rich and can be used for modeling compact manifolds (see, for instance, [10]). To give some intuition why this is the case, note that if a manifold is compact (which among other things implies that it is bounded) then it admits a finite covering by topological disks whose dimensionality equals the intrinsic dimension. We can equate these topological disks to the principal hyperplanes of sufficiently flat ellipsoids corresponding to high-probability mass sets of our Gaussians. If there are a sufficiently high number of ellipsoids, then the hyperplanes

Manuscript received August 31, 2009; accepted August 15, 2010. Date of publication August 30, 2010; date of current version November 17, 2010. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Deniz Erdogmus.

M. Chen, J. Silva, J. Paisley, C. Wang, and L. Carin are with the Electrical and Computer Engineering Department, Duke University, Durham, NC 27708-0291 USA (e-mail: lcarin@ee.duke.edu).

D. Dunson is with the Statistics Department, Duke University, Durham, NC 27708-0291 USA.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2010.2070796

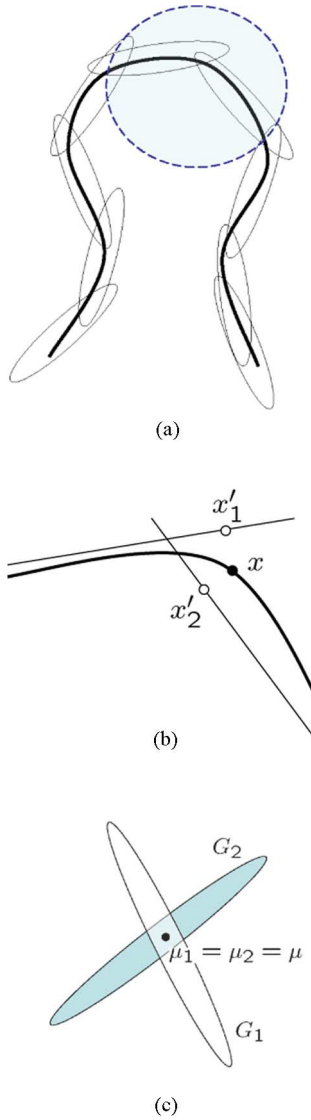


Fig. 2. Modeling a manifold with a mixture of Gaussians: (a) shows a covering of the manifold by high-probability mass ellipsoids; the shaded blue area is zoomed in (b), where we can see that point x on the manifold can be made arbitrarily close to projections x'_1 and x'_2 if the mixture contains enough Gaussians; in (c), Gaussians G_1 and G_2 with common mean μ model a union of hyperplanes which is not a manifold due to self-intersection.

are approximately tangent to the manifold and (by definition of manifold) we can establish locally valid one-to-one mappings between points on the manifold and points on the hyperplanes, with arbitrarily small distance between those points and their mappings. Fig. 2(a) and (b) illustrates this. Moreover, there are certain sets that are *not* manifolds but can still be well modeled by an MFA—a simple example consists of two Gaussians with the same mean but differently oriented principal hyperplanes, which do not constitute a manifold due to self intersection, as shown in Fig. 2(c). Thus, while the proposed model is appropriate for learning the statistics of manifolds, it is more generally applicable to data that reside in a low-dimensional region of a high-dimensional space.

In this paper, we show how to nonparametrically learn the MFA based upon available training data, and how to reconstruct signals based on limited random-projection measurements (while motivated by manifold learning, we emphasize, as discussed above, the MFA may also be applied to other signals not necessarily restricted to a manifold). We obtain guarantees similar to those in CS for sparse signals, using sub-Gaussian random projections that are, with high probability, incoherent with a certain *block*-sparsity dictionary. Unlike typical CS results, the dictionary is not, in general, an orthonormal basis. Namely, we derive a restricted isometry property (RIP) for the composite measurement-dictionary ensemble, drawing on one of two assumptions: i) separability of the Gaussian means and ii) *block*-incoherence of the low-rank hyperplanes spanned by the principal directions of the covariances. Importantly, it is shown that only one of the conditions, i) or ii), needs to hold.

An important issue is that, as Fig. 2 suggests, a manifold can be covered by an infinite number of Gaussian mixtures, i.e., the entire model is not uniquely identifiable. However, the posterior log-probability of different mixture models with similar quality and parsimony should be similar—we believe *learnability* is a more interesting property than identifiability in the context of learning the MFA. Our method favors MFA models that have a small number of components and a small number of factors (enforced by the corresponding priors), while maintaining high reconstruction quality, as measured by posterior log-probability. Additionally, and very importantly, *given the mixture parameters*, our results guarantee that (with high probability) a signal drawn from that mixture *is* uniquely identifiable from random projections.

Our contributions are as follows.

- We develop a hierarchical Bayesian algorithm that learns an MFA for the manifold based on training data. Unlike existing MFA inference algorithms [7], [11], [12], we adopt nonparametric techniques to simultaneously infer the number of clusters and the intrinsic subspace dimensionality.
- We present a method for reconstructing out-of-sample data using compressed random measurements. By using the probability density function learned from the MFA as the prior distribution, the reconstruction can be found analytically by Bayes' rule.
- We derive bounds on the number of required random-projection measurements, in terms of easily-computable quantities, such as the rank of the covariances, number of clusters, separation of the Gaussians, coherence and subcoherence of the dictionary.

The remainder of the paper is organized as follows. In Section II we develop the nonparametric mixture of factor analyzers, and in Section III we discuss how this model may be employed for CS inversion. Section IV provides performance bounds for CS assuming that the signals of interest are drawn from a known low-rank Gaussian mixture model, with example results presented in Section V. Conclusions are provided in Section VI.

II. Nonparametric Mixture of Factor Analyzers

We assume the signals $\boldsymbol{x} \in \mathbb{R}^N$ under study are drawn from a Gaussian mixture model, and that the rank of the covariance of each mixture component is small relative to N. We represent this statistical model as a mixture of factor analyzers (MFA)

[7], [13]–[15], with the MFA parameters learned from training data. As a special case this model may be applied to data drawn from a manifold or manifolds. In the discussion that follows, for conciseness, we will continually refer to data drawn from manifolds, since this is an important and motivating subproblem.

Many existing inference algorithms for learning an MFA [7], [11], [12] require one to *a priori* fix the subspace dimension and the number of mixture components (clusters). Unfortunately, these quantities are usually unknown in advance. We address this issue by placing Dirichlet process (DP) [8], [16] and Beta process (BP) [9], [17] priors on the MFA model, to infer the above mentioned quantities in a data-driven manner. As discussed further below, the DP is a nonparametric tool for mixture modeling, for inferring an appropriate number of mixture components [18]. The BP is a tool that allows one to uncover an appropriate number of factors [9], [17]. By integrating DP and BP into a single algorithm, we address both of the aforementioned problems associated with previous development of mixtures of factor analyzers.

A. Beta Process for Inferring Number of Factors

Assume access to n samples $\boldsymbol{x}_i \in \mathbb{R}^N$, with $i = 1, \ldots, n$ (all vectors are column vectors). The data are said to be drawn from a factor model if for all i

$$\boldsymbol{x}_i \sim \mathcal{N}(\boldsymbol{A}\boldsymbol{w}_i + \boldsymbol{\mu}, \alpha^{-1}\boldsymbol{I}_N), \quad \boldsymbol{w}_i \sim \mathcal{N}(0, \boldsymbol{I}_J) \tag{1}$$

where $\boldsymbol{A} \in \mathbb{R}^{N\times J}$, $\boldsymbol{\mu} \in \mathbb{R}^N$ and $\boldsymbol{I}_N$ is the $N \times N$ dimensional identity matrix (with $\boldsymbol{I}_J$ similarly defined). The precision $\alpha \in \mathbb{R}^+$, and one typically places a gamma prior on this quantity (discussed further below). We initially assume we know the number of factors J, and at this point we do not consider a mixture model.

By first considering (1) we may motivate the model that follows. Specifically, we may re-express (1) as

$$\boldsymbol{x}_i \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \quad \boldsymbol{\Sigma} = \boldsymbol{A}\boldsymbol{A}^T + \alpha^{-1}\boldsymbol{I}_N. \tag{2}$$

If $J \leq N$ and the columns of $\boldsymbol{A}$ are linearly independent, we may express $\boldsymbol{A}\boldsymbol{A}^T = \sum_{j=1}^J \zeta_j \boldsymbol{v}_j \boldsymbol{v}_j^T$, with orthonormal $\boldsymbol{v}_j \in \mathbb{R}^N$ and singular values $\zeta_j \in \mathbb{R}^+$, where ζ_1 through ζ_J are ordered in decreasing amplitude. If $1/\alpha$ is small relative to ζ_J, then $\boldsymbol{x}_i$ is drawn approximately from a Gaussian of covariance rank J, and if $J \ll N$ these Gaussians corresponds to localized, low-dimensional "pancakes" in $\mathbb{R}^N$ (see Fig. 2). As is well known [19], the vectors $\boldsymbol{v}_j$ define the principal coordinates of the low-rank Gaussian, which is centered about mean position $\boldsymbol{\mu}$.

Before generalizing (1) to a *mixture* of factor analyzers, we wish to address the problem of inferring J, which defines the rank of the Gaussians, with this related to the dimensionality of the manifold. Toward this end, we modify (1) as

$$\begin{aligned}
&\boldsymbol{x}_i \sim \mathcal{N}(\boldsymbol{A}\boldsymbol{w}_i + \boldsymbol{\mu}, \alpha^{-1}\boldsymbol{I}_N)\\
&\boldsymbol{w}_i = \hat{\boldsymbol{w}}_i \circ \boldsymbol{z}, \quad \hat{\boldsymbol{w}}_i \sim \mathcal{N}(0, \boldsymbol{I}_K)\\
&\boldsymbol{z} \sim \prod_{k=1}^K \text{Bernoulli}(\pi_k)\\
&\boldsymbol{\pi} \sim \prod_{k=1}^K \text{Beta}(a/K, b(K-1)/K)
\end{aligned} \tag{3}$$

where K is an integer chosen to be large relative to the number of anticipated factors (e.g., we may set $K = N$), π_k is the kth component of $\boldsymbol{\pi}$, and $\circ$ represents the point-wise (Hadamard) vector product. We use Bernoulli$(\cdot)$, Beta$(\cdot)$ and Mult$(\cdot)$ to refer respectively to Bernoulli, Beta and Multinomial densities parameterized by the arguments inside parentheses. This is a hierarchical model, in the sense that the priors for the parameters of the density of $\boldsymbol{x}_i \,|\, \boldsymbol{w}_i, \boldsymbol{A}, \boldsymbol{\mu}$ have their own priors (hyper-priors) with hyper-parameters.

As discussed in [9], when $K \rightarrow \infty$ the number of non-zero components in $\boldsymbol{z}$ is drawn from Poisson(a/b). For finite K, of interest here, one may show that the number of non-zero components of $\boldsymbol{z}$ is drawn from Binomial$(K, a/(a + b(K-1)))$, and therefore one may set a and b to impose prior belief on the number of factors that will be important per mixture component. The expected number of non-zero components in $\boldsymbol{z}$ is $aK/[a + b(K-1)]$. This construction has been related to a Beta Process, as discussed in [9]. Briefly, we draw $\boldsymbol{z}$ from a Bernoulli process parameterized by a Beta process over a measurable space Ω. Note that a Beta process, like a Dirichlet process, admits a representation as an infinite sum. We use a finite approximation to the Beta process (denoted, say, $H(\omega)$ for $\omega \in \Omega$) of the form $H(\omega) = \sum_{k=1}^K \pi_k \delta_{\omega_k}(\omega)$, with $\boldsymbol{\pi}$ specified in the second line of (3) and where $\omega_1, \ldots, \omega_K$ is a partition of Ω. In our problem Ω represents the space of possible columns of the factor-loading matrix $\boldsymbol{A}$, and the $\{\omega_k\}_{k=1,K}$ represent possible columns of $\boldsymbol{A}$ (factor loadings), drawn from a base measure H_0. The design of these columns and H_0 are discussed further below. The truncated Beta process procedure defines K candidate columns of $\boldsymbol{A}$, with respective probability of usage $\{\pi_k\}_{k=1,K}$, and via the sparseness-inducing properties of this construction (the vector $\boldsymbol{z}$ is sparse), only a subset of the candidate columns are selected, those that best fit the data, as quantified via the likelihood.

When one performs Bayesian inference using a construction like (3), the posterior density function on the binary vector $\boldsymbol{z}$ defines the number of columns of $\boldsymbol{A}$ that contribute to the factor analysis model, and hence this yields the rank of the Gaussian. As discussed further below, Bayesian inference may be performed relatively simply via this construction.

B. Dirichlet Process for Mixture of Factor Analyzers

The hierarchical model in (3) may be used to infer the number of factors in a factor model, with all $\boldsymbol{x}_i$ residing in a single associated subspace. However, to capture the nonlinear shape of a manifold, we are interested in a *mixture* of low-rank Gaussians (see Fig. 2). In general one does not know *a priori* the proper number of mixture components. We consequently employ the Dirichlet process (DP) [8], [16]. A draw G from a DP may be expressed as $G \sim \text{DP}(\eta G_0)$, where $\eta \in \mathbb{R}^+$ and G_0 is a base probability measure. A constructive representation for such a draw may be written as [8]

$$\begin{aligned}
&G = \sum_{t=1}^{\infty} \lambda_t \delta_{\theta_t^*},\\
&\lambda_t = v_t \prod_{l=1}^{t-1}(1 - v_l), \quad v_t \sim \text{Beta}(1, \eta), \quad \theta_t^* \sim G_0
\end{aligned} \tag{4}$$

where we note that, by construction, $\sum_{t=1}^{\infty}\lambda(t)=1$; the expression $\delta_{\theta_t^*}$ is a point measure situated at θ_t^*. The observed data samples $\{\boldsymbol{x}_i\}_{i=1,N}$ may be drawn from a parametric model $f(\theta_i)$ with associated parameter θ_i, with $\theta_i\sim G$. Since G is of the form in (4), for a relatively large number of samples N, many of the $\boldsymbol{x}_i$ will share the same parameters θ_t^*, and therefore the $\{\boldsymbol{x}_i\}_{i=1,N}$ are drawn from a mixture model. In our problem $f(\cdot)$ is a Gaussian, and hence we yield a Gaussian mixture model. While there are an infinite number of components (in principle) within G, via posterior inference we infer an appropriate number of mixture components for the data $\{\boldsymbol{x}_i\}_{i=1,N}$.

The DP favors a small number of mixture components, via the "stick" weights λ_t, which become small as t increases (only a relatively small set of mixture components are probable, with the number of components selected impacted by the data via the likelihood function). The base measure G_0 is here a prior on the components of each mixture-component-dependent factor model. Specifically, the G_0 is a prior on the factor loading $\boldsymbol{A}$, and mean $\boldsymbol{\mu}$. The component of G_0 associated with $\boldsymbol{A}$ is represented as discussed above in terms of a truncated Beta process construction (the number of used columns of $\boldsymbol{A}$ may be different for each of the mixture components). Our hierarchical model may be represented as

$$\boldsymbol{x}_i\,|\,\boldsymbol{w}_i,\boldsymbol{A}_{t(i)},\boldsymbol{\mu}_{t(i)},\alpha_{t(i)}\sim\mathcal{N}\left(\boldsymbol{A}_{t(i)}\boldsymbol{w}_i+\boldsymbol{\mu}_{t(i)},\alpha_{t(i)}^{-1}\boldsymbol{I}_N\right) \tag{5}$$

$$\boldsymbol{w}_i=\hat{\boldsymbol{w}}_i\circ\boldsymbol{z}_{t(i)},\quad \boldsymbol{A}_{t(i)}=\tilde{\boldsymbol{A}}_{t(i)}\boldsymbol{\Delta}_{t(i)} \tag{6}$$

$$\hat{\boldsymbol{w}}_i\sim\mathcal{N}_{t(i)}(0,\boldsymbol{I}_K) \tag{7}$$

$$t(i)\,|\,\lambda_1,\ldots,\lambda_T\sim\text{Mult}(1;\lambda_1,\ldots,\lambda_T) \tag{8}$$

$$\lambda_t=v_t\prod_{l=1}^{t-1}(1-v_l) \tag{9}$$

$$v_t\,|\,\eta\sim\text{Beta}(1,\eta) \tag{10}$$

$$\boldsymbol{z}_t\,|\,\boldsymbol{\pi}_t\sim\prod_{k=1}^{K}\text{Bernoulli}(\pi_{tk}) \tag{11}$$

$$\boldsymbol{\pi}_t\sim\prod_{k=1}^{K}\text{Beta}(a/K,b(K-1)/K) \tag{12}$$

$$\boldsymbol{\mu}_t\,|\,\tau_0\sim\mathcal{N}\left(\boldsymbol{\mu},\tau_0^{-1}\boldsymbol{I}_N\right) \tag{13}$$

$$\tilde{\boldsymbol{A}}_t\sim\prod_{k=1}^{K}\mathcal{N}\left(0,\frac{1}{N}\boldsymbol{I}_N\right) \tag{14}$$

$$\boldsymbol{\Delta}_t\,|\,\tau_{t1},\ldots,\tau_{tK}\sim\prod_{k=1}^{K}\mathcal{N}\left(0,\tau_{tk}^{-1}\right) \tag{15}$$

with $\lambda_T=1-\sum_{t=1}^{T-1}\lambda_t$, where we have truncated the DP sum to T terms (properties of this truncation are discussed in [20]). The notation $\boldsymbol{\Delta}_t\sim\prod_{k=1}^{K}\mathcal{N}(0,\tau_{tk}^{-1})$ is meant to mean that the K *diagonal* elements of $\boldsymbol{\Delta}_t$ are drawn from $\mathcal{N}(0,\tau_{tk}^{-1})$, with $k=1,2,\ldots,K$. The diagonal matrix $\boldsymbol{\Delta}_t$ encodes the importance of each column in $\tilde{\boldsymbol{A}}_t$, playing the same role of singular values in SVD. The expression $\text{Mult}(1;\lambda_1,\ldots,\lambda_T)$ represents drawing one sample from a multinomial distribution defined by $(\lambda_1,\ldots,\lambda_T)$, and $t(i)$ corresponds to the mixture component associated with the ith draw. The expression $\mathcal{N}_{t(i)}(0,\boldsymbol{I}_K)$ is meant to indicate that the factor score associated with a given sample i is explicitly linked to a particular mixture component (this is important for yielding a mixture-component-dependent posterior density function of $\boldsymbol{w}$, and impacts the Gibbs-sampler update equations). The vector $\boldsymbol{\mu}$ represents the mean computed based on all training data used to design the model, i.e., $\boldsymbol{\mu}=1/n\sum_{i=1}^{n}\boldsymbol{x}_i$.

The expression $\tilde{\boldsymbol{A}}_t\sim\prod_{k=1}^{K}\mathcal{N}(0,(1/N)\boldsymbol{I}_N)$ means that each of the K columns of $\tilde{\boldsymbol{A}}_t$ are drawn independently from $\mathcal{N}(0,(1/N)\boldsymbol{I}_N)$ (this corresponds to the base measure H_0 in the Beta process), implying that on average these columns will have unit norm (although any given draw will not exactly have unit norm). The covariance associated with mixture component t, as constituted via the prior, is $\boldsymbol{\Sigma}_t=\tilde{\boldsymbol{A}}_t\tilde{\boldsymbol{\Delta}}_t^2\tilde{\boldsymbol{A}}_t^T+\alpha_t^{-1}\boldsymbol{I}$, where $\tilde{\boldsymbol{\Delta}}_t=\boldsymbol{\Delta}_t\,\text{diag}(z_{t1},\ldots,z_{tK})$. Hence, the number of non-zero components in the binary vector $\boldsymbol{z}_t$ defines the approximate rank of $\boldsymbol{\Sigma}_t$, assuming the smallest diagonal element of $\boldsymbol{\Delta}_t^2$ is large relative to α_t^{-1}. There are other ways one may draw $\boldsymbol{A}_t$ from a hierarchical generative model, but this construction appears to be the most stable among many we have examined.

Note that in (12)–(16), within the prior the T components $\{\boldsymbol{z}_t\}_{t=1,T}$, $\{\boldsymbol{\pi}_t\}_{t=1,T}$, $\{\boldsymbol{\mu}_t\}_{t=1,T}$, $\{\tilde{\boldsymbol{A}}_t\}_{t=1,T}$ and $\{\boldsymbol{\Delta}_t\}_{t=1,T}$ are drawn *once* for all samples $\{\boldsymbol{x}_i\}_{i=1,N}$, and these effectively correspond to the draws from the DP base measure G_0; (10)–(11) are also drawn once, these yielding the T mixture weights in the truncated "stick-breaking" representation of the DP [20]. The $\hat{\boldsymbol{w}}_i$ is drawn separately for each of the N samples. Hence, all data drawn from a given mixture component t share the factor-loading matrix $\boldsymbol{A}_t\in\mathbb{R}^{N\times K}$ and the same set of important columns (factor loadings) defined by $\boldsymbol{z}_t$, but each draw from a given mixture component has unique weights $\boldsymbol{w}_i$.

Concerning the way in which $\boldsymbol{\mu}_t\in\mathbb{R}^N$ and $\boldsymbol{A}_t\in\mathbb{R}^{N\times K}$ are drawn, the use of independent Gaussians allows for convenient inference. It is important to note that (14) and (15) simply constitute convenient *priors*, while the posterior Bayesian analysis will infer the correlations within these terms. The same is true for the factor score $\hat{\boldsymbol{w}}_i$.

C. Inference and Hyper-Parameter Settings

To complete the model, we assume $\eta\sim\text{Gamma}(c,d)$, $\tau_{tk}\sim\text{Gamma}(e,f)$, $\alpha_t\sim\text{Gamma}(g,h)$. Non-informative hyper-parameters are employed throughout, setting $c=d=e=f=g=h=10^{-6}$. For further illustration, we present the corresponding complete graphical model in Fig. 3. The precision $\tau_0=10^{-6}$, implying that the means $\boldsymbol{\mu}_t$ were drawn almost from a uniform prior. Further, we set $a=b=1$ in all examples. While there may appear to be a large number of hyper-parameters, all of these settings are "standard" [9], [21] and there has been no parameter tuning for any of the examples. In all examples below, we set the truncations as $K=T=50$.

The model parameters can be estimated using Gibbs sampling—we provide a detailed explanation in the Appendix. It is important to note that Gibbs sampling can sometimes suffer from slow mixing, becoming trapped in local modes. Other sampling techniques for DP have been developed, such as split-merge MCMC [22], that attempt to overcome this limitation. However, [23] is based on a Chinese Restaurant Process (CRP) DP construction, while here we employ a stick-breaking formulation. An advantage of the CRP is that it truly allows an

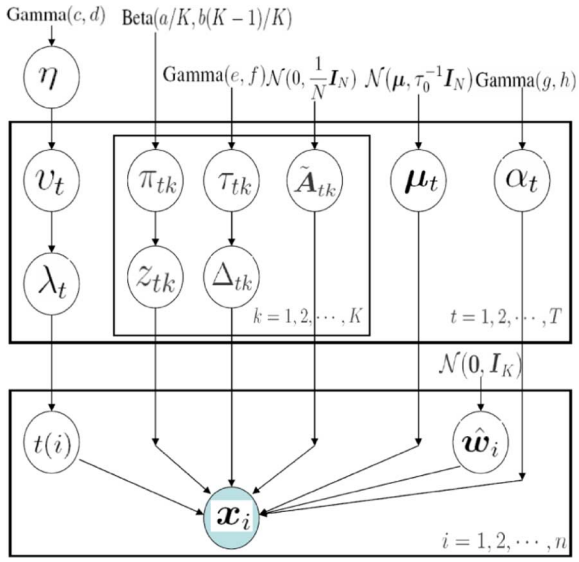


Fig. 3. Graphical model corresponding to (5)–(15).

infinite number of mixture components (in principle), while the stick-breaking construction discussed above employed a truncation. To remove this truncation, we also have considered a retrospective Gibbs sampler [23] for the stick-breaking construction, with this compared below to the truncated stick-breaking representation. The method in [23] also allows an infinite number of mixture components in principle, like [22]. The relative effectiveness of the truncated stick-breaking DP representation, and the associated Gibbs-sampler implementation, is discussed in detail in [24].

Efore, we decided to retain the truncated stick-breaking construction.

One may also perform variational Bayesian (VB) [11] analysis for this model. However, learning of the model need only be performed once, and therefore Gibbs sampling has been employed for this purpose. In all examples presented below, we employed 2000 burn-in iterations, and 1000 collection iterations, and we document the convergence behavior by showing log-probability plots. One may clearly use more collection iterations if desired, but we found this unnecessary for the CS application which is the focus of this paper.

D. Probability Density Estimation From the Above Nonparametric MFA Model

The above model can be interpreted as a Bayesian local PCA model, in which the signal manifold is approximated by a mixture of local subspaces. After model inference, the probability density function (pdf) of the signal can be estimated as follows:

$$p(\boldsymbol{x}) = \sum_{t=1}^{T} \lambda_t \int \mathcal{N}\left(\boldsymbol{x}; \tilde{\boldsymbol{A}}_t(\boldsymbol{\Delta}_t \operatorname{diag}(\boldsymbol{z}_t))\hat{\boldsymbol{w}} + \boldsymbol{\mu}_t, \alpha_t^{-1}\boldsymbol{I}_N\right) \times \mathcal{N}(\hat{\boldsymbol{w}}; \boldsymbol{\xi}_t, \boldsymbol{\Lambda}_t)\mathrm{d}\hat{\boldsymbol{w}} = \sum_{t=1}^{T} \lambda_t \mathcal{N}(\boldsymbol{x}; \boldsymbol{\chi}_t, \boldsymbol{\Omega}_t) \quad (16)$$

$$\boldsymbol{\chi}_t = \boldsymbol{\mu}_t + \tilde{\boldsymbol{A}}_t(\boldsymbol{\Delta}_t \operatorname{diag}(\boldsymbol{z}_t))\boldsymbol{\xi}_t;$$

$$\boldsymbol{\Omega}_t = \tilde{\boldsymbol{A}}_t(\boldsymbol{\Delta}_t \operatorname{diag}(\boldsymbol{z}_t))\boldsymbol{\Lambda}_t(\operatorname{diag}(\boldsymbol{z}_t)\boldsymbol{\Delta}_t)\tilde{\boldsymbol{A}}_t^{\top} + \alpha_t^{-1}\boldsymbol{I}_N \quad (17)$$

This is the explicit form of the low-rank Gaussian mixture model density we estimate. If we use the prior distribution for $\hat{\boldsymbol{w}}$, i.e., $\boldsymbol{\xi}_t = \boldsymbol{0}$ and $\boldsymbol{\Lambda}_t = \boldsymbol{I}_K$, then $\boldsymbol{\chi}_t = \boldsymbol{\mu}_t$ and $\boldsymbol{\Omega}_t = \boldsymbol{\Sigma}_t$ with $\boldsymbol{\Sigma}_t$ defined in Section II-B. However, this estimator may not be accurate since $\hat{\boldsymbol{w}}$ has its own posterior distribution in each mixture component. Thus, we use the following posterior mean and covariance estimate for $\hat{\boldsymbol{w}}$ from Gibbs sampling:

$$\boldsymbol{\xi}_t = \frac{1}{M}\sum_{m=1}^{M}\left(\sum_{i:t(i)=t}\hat{\boldsymbol{w}}_i^{(m)} \Big/ \sum_{i:t(i)=t} 1\right);$$

$$\boldsymbol{\Lambda}_t = \frac{1}{M}\sum_{m=1}^{M}\left(\sum_{i:t(i)=t}\hat{\boldsymbol{w}}_i^{(m)}\hat{\boldsymbol{w}}_i^{(m)\top} \Big/ \sum_{i:t(i)=t} 1\right) - \boldsymbol{\xi}_t\boldsymbol{\xi}_t^{\top}$$

where $\hat{\boldsymbol{w}}_i^{(m)}$ is the mth collected Gibbs sample. We use the mean of the Gibbs samples to estimate $\tilde{\boldsymbol{A}}_t$. The estimated pdf in (16) will be used as the prior distribution for new testing signal in the compressive sensing application, which is illustrated in the following section.

III. Compressive Sensing Using a Low-Rank MFA

A. CS Inversion for Data Drawn From a Low-Rank GMM

Using the procedure discussed in the previous section, assume access to an MFA for a low-rank Gaussian mixture model of interest (e.g., for representation of a manifold). Let $\boldsymbol{x} \in \mathbb{R}^N$ be a vector drawn from this distribution. Rather than measuring $\boldsymbol{x}$ directly, we perform a projection measurement $\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{x} + \boldsymbol{\nu}$, where $\boldsymbol{\Phi} \in \mathbb{R}^{m\times N}$ is a measurement projection matrix, and we are interested in the case $m \ll N$. Details on the design of $\boldsymbol{\Phi}$ are provided in Section IV. The vector $\boldsymbol{\nu} \in \mathbb{R}^N$ represents measurement noise. Our goal is to recover $\boldsymbol{x}$ from $\boldsymbol{y}$, with this done effectively with $m \ll N$ measurements because $\boldsymbol{x}$ is known to be drawn from a low-dimensional MFA model. Bounds on requirements for m, based upon the properties of the MFA, are discussed in Section IV. Our objective here is to describe a general-purpose algorithm for recovering $\boldsymbol{x}$ from $\boldsymbol{y}$.

Let $p(\boldsymbol{x})$ represent the MFA learned via the procedure discussed in Section II. Note that the nonparametric learning procedure discussed there infers a full posterior density function on all parameters of the mixture model. Within the CS inversion we utilize the inferred mean of each mixture component, and an approximation to the covariance matrix based on averaging across all collection samples. Specifically, we have $p(\boldsymbol{x}) = \sum_{t=1}^{T}\lambda_t\mathcal{N}(\boldsymbol{x};\boldsymbol{\chi}_t,\boldsymbol{\Omega}_t)$ where $\boldsymbol{\chi}_t$ represents the mean for mixture component t, and $\boldsymbol{\Omega}_t$ is the approximate inferred covariance matrix defined in (16) and (17). The λ_t are the mean mixture weights learned via the DP analysis, noting that in practice many of these will be very near zero (hence, we infer the proper number of *meaningful* mixture components, with T simply a large-valued truncation of the DP stick-breaking representation [20]).

The noise $\boldsymbol{\nu}$ is assumed drawn from a zero-mean Gaussian with precision matrix (inverse covariance) $\boldsymbol{R}$. The condition distribution for $\boldsymbol{y}$ given $\boldsymbol{x}$ may be evaluated analytically as

$$\begin{aligned} p(\boldsymbol{x}\,|\,\boldsymbol{y}) &= \frac{p(\boldsymbol{x})p(\boldsymbol{y}\,|\,\boldsymbol{x})}{\int p(\boldsymbol{x})p(\boldsymbol{y}\,|\,\boldsymbol{x})d\boldsymbol{x}} \\ &= \frac{\sum_{t=1}^{T}\lambda_t\mathcal{N}(\boldsymbol{x};\boldsymbol{\chi}_t,\boldsymbol{\Omega}_t)\times\mathcal{N}(\boldsymbol{y};\boldsymbol{\Phi}\boldsymbol{x},\boldsymbol{R}^{-1})}{\int\sum_{l=1}^{T}\lambda_l\mathcal{N}(\boldsymbol{x};\boldsymbol{\chi}_l,\boldsymbol{\Omega}_l)\times\mathcal{N}(\boldsymbol{y};\boldsymbol{\Phi}\boldsymbol{x},\boldsymbol{R}^{-1})d\boldsymbol{x}} \\ &= \sum_{t=1}^{T}\tilde{\lambda}_t\mathcal{N}(\boldsymbol{x};\tilde{\boldsymbol{\chi}}_t,\tilde{\boldsymbol{\Omega}}_t) \end{aligned} \tag{18}$$

with

$$\begin{aligned} \tilde{\lambda}_t &= \frac{\lambda_t\mathcal{N}(\boldsymbol{y};\boldsymbol{\Phi}\boldsymbol{\chi}_t,\boldsymbol{R}^{-1}+\boldsymbol{\Phi}\boldsymbol{\Omega}_t\boldsymbol{\Phi}^\top)}{\sum_{l=1}^{T}\lambda_l\mathcal{N}(\boldsymbol{y};\boldsymbol{\Phi}\boldsymbol{\chi}_l,\boldsymbol{R}^{-1}+\boldsymbol{\Phi}\boldsymbol{\Omega}_l\boldsymbol{\Phi}^\top)} \\ \tilde{\boldsymbol{\Omega}}_t &= (\boldsymbol{\Phi}^\top\boldsymbol{R}\boldsymbol{\Phi}+\boldsymbol{\Omega}_t^{-1})^{-1} \\ &= \boldsymbol{\Omega}_t-\boldsymbol{\Omega}_t\boldsymbol{\Phi}^\top(\boldsymbol{R}^{-1}+\boldsymbol{\Phi}\boldsymbol{\Omega}_t\boldsymbol{\Phi}^\top)^{-1}\boldsymbol{\Phi}\boldsymbol{\Omega}_t \\ \tilde{\boldsymbol{\chi}}_t &= \tilde{\boldsymbol{\Omega}}_t(\boldsymbol{\Phi}^\top\boldsymbol{R}\boldsymbol{y}+\boldsymbol{\Omega}_t^{-1}\boldsymbol{\chi}_t) \\ &= \boldsymbol{\Omega}_t\boldsymbol{\Phi}^\top(\boldsymbol{R}^{-1}+\boldsymbol{\Phi}\boldsymbol{\Omega}_t\boldsymbol{\Phi}^\top)^{-1}(\boldsymbol{y}-\boldsymbol{\Phi}\boldsymbol{\chi}_t)+\boldsymbol{\chi}_t. \end{aligned}$$

In the above computations, the following identity for normal distributions is used:

$$\begin{aligned} &\mathcal{N}(\boldsymbol{x};\boldsymbol{\chi}_t,\boldsymbol{\Omega}_t)\times\mathcal{N}(\boldsymbol{y};\boldsymbol{\Phi}\boldsymbol{x},\boldsymbol{R}^{-1}) \\ &\quad=\mathcal{N}(\boldsymbol{x};\tilde{\boldsymbol{\chi}}_t,\tilde{\boldsymbol{\Omega}}_t)\times\mathcal{N}(\boldsymbol{y};\boldsymbol{\Phi}\boldsymbol{\chi}_t,\boldsymbol{R}^{-1}+\boldsymbol{\Phi}\boldsymbol{\Omega}_t\boldsymbol{\Phi}^\top) \end{aligned}$$

with the reader referred to [21] for a fuller description of a related derivation. In the results presented below we consider the case for which the components of $\boldsymbol{R}^{-1}$ tend to zero, therefore assuming noise-free measurements. If the measurements are noisy one may infer $\boldsymbol{R}$ within a hierarchical Bayesian analysis, but the inversion for $\boldsymbol{x}$ is no longer analytic (unless the noise covariance $\boldsymbol{R}^{-1}$ is known *a priori*).

It is interesting to note that the MFA mixture model may be relatively computationally expensive to learn, depending on the number of samples one has available for learning the properties of $p(\boldsymbol{x})$ (it is desirable that the number of samples be as large as possible, to improve model quality). However, once $p(\boldsymbol{x})$ is learned "offline," the CS recovery $\boldsymbol{y}\rightarrow\boldsymbol{x}$ is *analytic*, in the sense that we have closed-form expressions for all the parameters of the posterior distribution $p(\boldsymbol{x}\,|\,\boldsymbol{y})$. Moreover, rather than simply yielding a single "point" estimate for $\boldsymbol{x}$, we recover the full distribution $p(\boldsymbol{x}\,|\,\boldsymbol{y})$. When presenting results we plot the mean value of the inferred $\boldsymbol{x}$.

B. Illustration Using Simple Manifold Data

To make the discussion more concrete, we return to the shifted Gaussian manifold example discussed in the Introduction. This simple example is considered to examine learning $p(\boldsymbol{x})$, as well as inference of $p(\boldsymbol{y}\,|\,\boldsymbol{x})$ for CS inversion. The training set consists of $n=900$ shifted Gaussian samples, each with dimension $N=128$.

Fig. 4 examines the number of mixture components inferred, as well as the properties of any particular data point as viewed from the MFA. Specifically, a total of eight important mixture components are inferred, and the vast majority of data samples (896 of 900) are associated with only one mixture component (cluster). This implies that the signals possess *block* sparsity, in that a given signal $\boldsymbol{x}$ only employs factors from one of the mixture components. We exploit this property when deriving bounds for the number of required CS measurements. These results were obtained using a truncated stick-breaking DP representation.

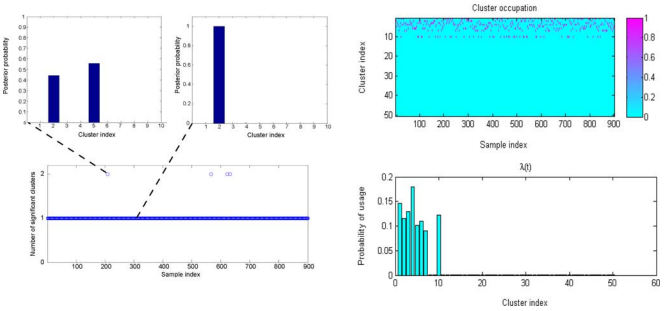


Fig. 4. Learned MFA mixture-component occupancy for the shifted Gaussian data. Bottom left: Number of mixture components (clusters) with significant (>0.1) posterior probability, for each training point. Note how only four out of 900 points on the manifold have more than one significant cluster. Top left: plots of the posterior $p(t_i\,|\,\boldsymbol{x}_i,-)$, for $i=208$ and $i=300$. In most cases, the posterior behaves as shown for $i=300$. Only weights for mixture components 1–10 are shown, with the remaining components having negligible mixture weights for these samples. Top right: Cluster occupation probability for each training point. Bottom right: Learned weights of the clusters in the mixture (in total, only eight dominant mixture components are inferred).

In Fig. 5 are shown the properties of the individual factor models (for a given mixture component) of the MFA. We utilized both a truncated stick-breaking construction and retrospective sampling, as discussed in Section II-C. Note in Fig. 5(b) how the truncated stick-breaking construction results in each mixture component being dominated by a single factor, consistent with the one-dimensional character of the manifold. This is less evident in Fig. 5(c), corresponding to retrospective sampling, where less clusters are used but each cluster requires more factors. This behavior in undesirable, since it dilutes the low-rank structure of the Gaussians and, in our experiments it yields worse reconstruction performance, especially for real data. Hence, all subsequent results are presented using the truncated stick-breaking DP representation. We note that it may be possible to improve the retrospective Gibbs sampler performance by careful parameter tuning, but given the effectiveness of the truncated stick-breaking construction, such tuning was not exhaustively considered.

After the training process using the $n=900$ samples, we have an MFA model for $p(\boldsymbol{x})$, which we may now employ in CS inversion. To test the CS inversion, we generated 100 new (noise-free) test signals with different peak positions a, and manifested associated compressive projection measurements $\boldsymbol{y}$ (again, with the components of $\boldsymbol{\Phi}$ drawn i.i.d. from $\mathcal{N}(0,1)$). The performance is shown in Fig. 6(a) and examples of the reconstruction are given in Fig. 7 for $m=5$ CS measurements. In Fig. 6(a) (and in related results in Section V), the relative reconstruction error is defined as follows. If all n testing signals consist of the matrix $\boldsymbol{X}\in\mathbb{R}^{N\times n}$ and the associated reconstructed signals are represented as $\hat{\boldsymbol{X}}\in\mathbb{R}^{N\times n}$, the relative reconstruction error is de-

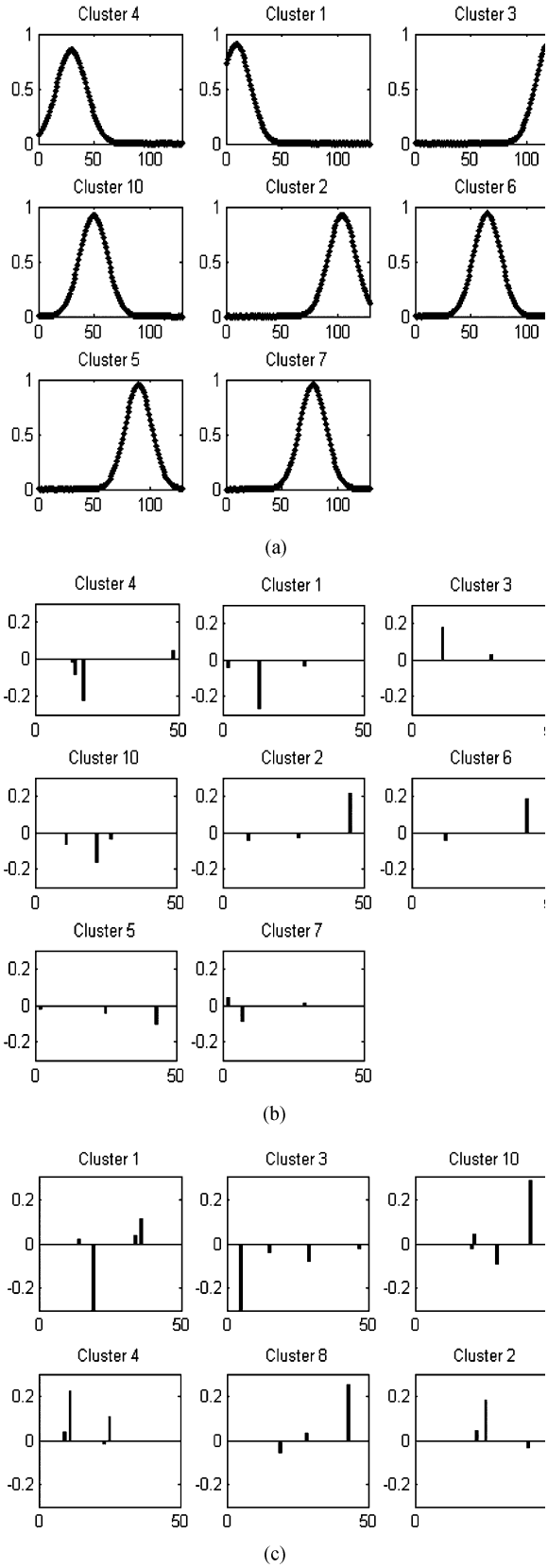


Fig. 5. Properties of the individual (mixture-component-dependent) factor models for the MFA, considering the shifted Gaussian data. (a) Cluster centers ($\boldsymbol{\chi}_t$) using truncated stick-breaking; (b) Factor usage ($\boldsymbol{\Delta}_t \ \mathrm{diag}(z_t)$) of each cluster using truncated stick-breaking; (c) Factor usage of each cluster using retrospective sampling.

fined as $(\|\boldsymbol{X}-\hat{\boldsymbol{X}}\|_F)/(\|\boldsymbol{X}\|_F)$. Using very few measurements ($m \ll N$), we reconstruct $\boldsymbol{x}$ almost perfectly. Also shown is the performance of Block-OMP (BOMP) [25], a CS method for block-sparse signals which, as we explain in Section IV, utilizes some similar ideas to ours.

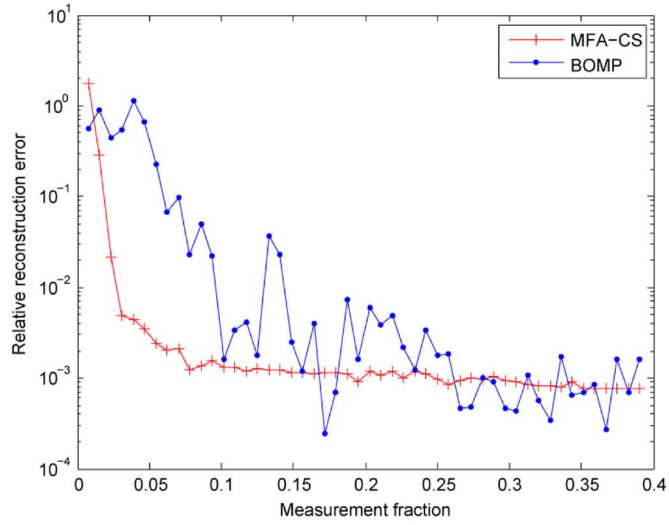
C. Gibbs Sampling and Label Switching

When performing Gibbs sampling to learn the MFA, we infer the number of factors per mixture component, as well as the number of mixtures. Further, we set the truncation levels for the number of factors and mixture components to large values ($K = T = 50$). In principle, the indexes on the factors and mixtures are exchangeable, and therefore within the Gibbs sampler the indexes (between the 50 possible values) of the factors and mixture components may be interchanged between consecutive Gibbs samples (in fact, this would be an indication of good mixing, since the labels *are* exchangeable). Plots like the right figures in Fig. 4 indicate that the labels of the factors and mixtures converge to a local mode, as the associated labels are stable after sufficient number of Gibbs burn-in iterations. We also show, in Fig. 6(b), a plot of the posterior log-probability as a function of the number of Gibbs iterations; while this does *not* assure convergence of the Gibbs sampler to the posterior of all model parameters, the convergence in Fig. 6(b) may explain why our learned MFA works well in practice for the objective of CS recovery. We note that if label switching becomes a problem for particular MFA examples, techniques are available to address this [26], [27], such that one may recover an analytic expression for the mixture model. In all examples considered here label switching was not found to be a problem.

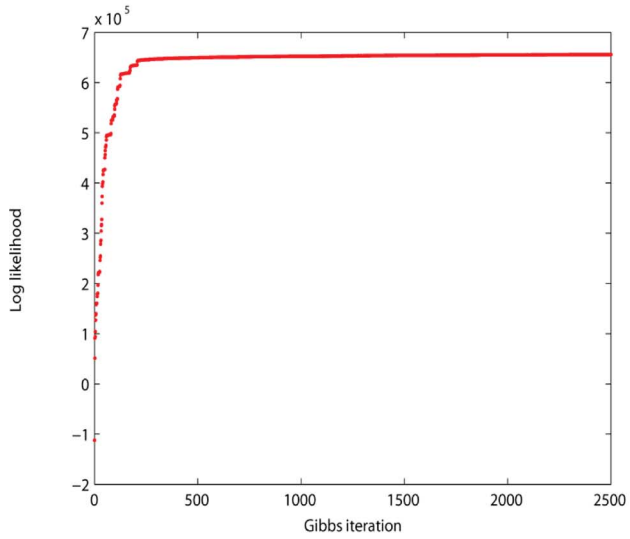
IV. Bounds on the Number of Random Measurements

We derive sample-complexity bounds for CS reconstruction under the MFA model. In practice we assume available training data for images of interest, and using these data we *learn* the MFA, using the procedure discussed in the previous subsections. We now assume access to this learned MFA, and we wish to examine the required number of *compressive* measurements on new data, required to assure accurate recovery of the underlying signal (which is assumed drawn from the learned MFA).

Each observation $\boldsymbol{x}_i$ is assumed drawn from an MFA as in (6), where in the analysis below we consider the case $\alpha_{t(i)} \rightarrow \infty$; hence, we ignore additive measurement noise for simplicity. For notational simplicity, we additionally assume that each mixture component is composed of d factors. We define a matrix $\boldsymbol{\Psi} \in \mathbb{R}^{N \times (d+1)T}$, where consecutive blocks of $d+1$ contiguous columns correspond to the associated columns in the mixture-component-dependent factor matrices $\boldsymbol{A}_t$, for $t = 1, \ldots, T$ (T mixture components). The first column in each block corresponds to the respective normalized (to unit Euclidean distance) mean vector, and the remaining d columns in each block are defined by the columns of $\boldsymbol{A}$ (which are assumed to be linearly independent). Any $\boldsymbol{x}$ satisfies $\boldsymbol{x} = \boldsymbol{\Psi}\boldsymbol{\theta}$, where $\boldsymbol{\theta}$ is assumed to have only $d+1$ non-zero components, corresponding to the respective block. Note that this assumption comes from the way in which we have arranged $\boldsymbol{\Psi}$ to match the MFA model: each contiguous block of $d+1$ columns corresponds to one mixture component; within each block the first column is the normalized



(a)



(b)

Fig. 6. (a) Relative reconstruction error for the shifted Gaussian data, as a function of the number of measurements (fraction of the $N = 128$). We compare MFA-CS (our method) with BOMP. (b) Log-probability plot as a function of the number of Gibbs iterations (for learning the MFA model).

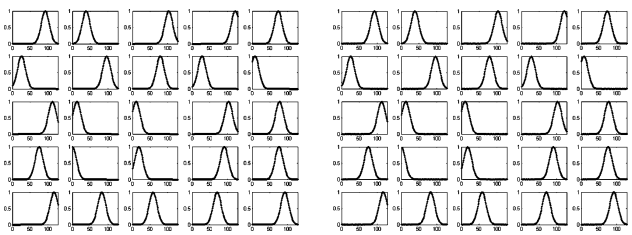


Fig. 7. CS reconstruction result for the shifted Gaussian data. The left figure shows examples of the test signals, and the right figure shows the reconstructed signals with 3.9% measurements (recovery of $\boldsymbol{x} \in \mathbb{R}^{128}$ based on $\boldsymbol{y} \in \mathbb{R}^5$).

mean of that component and the other d columns are the factors. This block structure in $\boldsymbol{\Psi}$ naturally imposes the same structure on $\boldsymbol{\theta}$, which we exploit. Thus, the dictionary $\boldsymbol{\Psi}$ contains a subset of the information present in the MFA parameters, namely the means and covariances of the mixture component s.

Recovering $\boldsymbol{x}$ from random projections amounts to recovering $\boldsymbol{\theta}$, which we assume to follow a particular sparsity pattern: $\boldsymbol{\theta}$ is block-sparse, meaning that the non-zero coordinates of $\boldsymbol{\theta}$ appear in predefined $(d+1)$-sized blocks; in this case, it also happens that the first element is always equal to $\|\boldsymbol{\mu}_t\|$ for the tth block.

The role of block-sparsity has recently been noted in the problem of reconstructing signals that live in a union of linear subspaces [28], [29]. The related notion of *block-coherence*, introduced in the same work, will be of use here, although in our setting we are interested in a union of affine spaces rather than linear subspaces (our hyperplanes will generally not include the origin). An additional difference is that our dictionaries can be, and usually are under-complete, i.e., $(d+1)T < N$. Also of crucial importance is the role of *separability*, which has been explored by Dasgupta [19] in the context of learning mixtures of high dimensional Gaussians (not necessarily low-rank) from random projections. We build on these results and extend them to certain cases where separability does not hold.

We now define block-sparsity, block-coherence and separability more precisely, following [28] and [19], respectively.

a) Block-sparsity: [25] Let $\boldsymbol{x} \in \mathbb{R}^N$ be represented in a dictionary $\boldsymbol{\Psi} \in \mathbb{R}^{N \times (d+1)T}$, so that $\boldsymbol{x} = \boldsymbol{\Psi}\boldsymbol{\theta}$, where $\boldsymbol{\theta} \in \mathbb{R}^{(d+1)T}$ is a parameter vector. We say that $\boldsymbol{x}$ is block L-sparse in $\boldsymbol{\Psi}$, with blocks of size $d+1$, if $\boldsymbol{\theta}$ can be written as

$$\boldsymbol{\theta}^\top = [\underbrace{\theta_1 \dots \theta_{d+1}}_{\boldsymbol{\theta}[1]} \dots \underbrace{\theta_{(d+1)T-d-1} \dots \theta_{(d+1)T}}_{\boldsymbol{\theta}[T]}]^\top \tag{19}$$

with at most L of the $\boldsymbol{\theta}[1] \dots \boldsymbol{\theta}[T]$ blocks having non-zero norm. We will assume $L = 1$, so only one block is active for each $\boldsymbol{x}$.

b) Block-coherence: [25] Let $\rho(\boldsymbol{A}) = \sqrt{\lambda_{\max}(\boldsymbol{A}^\top \boldsymbol{A})}$ denote the spectral radius of some matrix $\boldsymbol{A}$, where $\lambda_{\max}(\boldsymbol{A}^\top \boldsymbol{A})$ is the largest eigenvalue of the positive semidefinite matrix $\boldsymbol{A}^\top \boldsymbol{A}$. Let us also express dictionary $\boldsymbol{\Psi}$ in block form as

$$\boldsymbol{\Psi} = [\underbrace{\boldsymbol{\psi}_1 \dots \boldsymbol{\psi}_{(d+1)}}_{\boldsymbol{\Psi}[1]} \dots \underbrace{\boldsymbol{\psi}_{(d+1)T-d-1} \dots \boldsymbol{\psi}_{(d+1)T}}_{\boldsymbol{\Psi}[T]}] \tag{20}$$

and write $\boldsymbol{M}[l,r] = \boldsymbol{\Psi}[l]^T \boldsymbol{\Psi}[r]$, with $\boldsymbol{M}[l,r] \in \mathbb{R}^{(d+1)\times(d+1)}$. The block coherence of $\boldsymbol{\Psi}$ is defined as $\mu_B = \max_{l,r \neq l}(1)/(d+1)\rho(\boldsymbol{M}[l,r])$.

It should be stressed that the subdictionaries $\boldsymbol{\Psi}[1], \dots, \boldsymbol{\Psi}[T]$ do not need to be orthonormal bases themselves. However, they can be orthogonalized without changing the block-sparsity of $\boldsymbol{\Psi}$, as long as the columns of $\boldsymbol{\Psi}[t]$ are linearly independent, for all t. This can be ensured by the following proposition from [28].

Proposition 1: The representation $\boldsymbol{x} = \boldsymbol{\Psi}\boldsymbol{\theta}$ is unique iff $\boldsymbol{\Psi}\boldsymbol{g} \neq \boldsymbol{0}$ for every $\boldsymbol{g} \neq \boldsymbol{0}$ that is block $2L$-sparse.

In other words, no $2L$ subdictionaries (or less) can be linearly dependent. In our setting, with $L = 1$, this condition excludes pairs of linearly dependent subdictionaries. It also excludes degenerate subdictionaries where, for instance, the mean lies in the span of the vectors of a given factor.

Additionally, we introduce the concept of *subcoherence* as in [25]. Subcoherence is defined as $\nu = \max_\ell \max_{i,j \neq i} |\boldsymbol{\psi}_i^\top \boldsymbol{\psi}_j|$, for $\boldsymbol{\psi}_i, \boldsymbol{\psi}_j \in \boldsymbol{\Psi}[\ell]$. Clearly, if all subdictionaries $\boldsymbol{\Psi}[\ell]$ are orthonormal, then $\nu = 0$.

c) Separation: [19] Two Gaussians, $\mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and $\mathcal{N}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ in $\mathbb{R}^N$, are c-separated if $\|\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2\| \geq c\sqrt{N \max(\lambda_{\max}(\boldsymbol{\Sigma}_1), \lambda_{\max}(\boldsymbol{\Sigma}_2))}$.

A. Separability Result

Our first result stems almost directly from [19]. Namely, we draw from [19, Lemma 2, p. 23] and from [19, Lemmas 18 and 19, p. 57].

Theorem 1: Let $\boldsymbol{x}$ come from Gaussian component t in an MFA where each component has d factors. Choose accuracy and confidence parameters $\epsilon, \delta > 0$. Assume that all the Gaussians are 1-separated, and that we observe a vector $\boldsymbol{y} \in \mathbb{R}^m$ of sub-Gaussian random projections with $m \geq \max(d, (4)/(\epsilon^2)\log(T^2)/(2\delta))$. The original means and covariances will also be projected onto known m-dimensional vectors $\boldsymbol{\mu}_1^*, \ldots, \boldsymbol{\mu}_T^*$ and $m \times m$ matrices $\boldsymbol{\Sigma}_1^*, \ldots, \boldsymbol{\Sigma}_T^*$. Assign to $\boldsymbol{y}$ the label t^* such that $t^* = \min_k \sqrt{(\boldsymbol{y} - \boldsymbol{\mu}_k^*)^\top \boldsymbol{\Sigma}_k^* (\boldsymbol{y} - \boldsymbol{\mu}_k^*)}$. Then, $\Pr(t^* \neq t)$, i.e., the probability that $\boldsymbol{x}$ is misclassified into Gaussian j whose separation (in low dimensions, with probability $>1-\delta$) from t is $c_{tj}^* \geq \sqrt{1-\epsilon}$, is at most

$$e^{-c_{tj}^{*2} m/20} + \frac{6}{c_{tj}^* \sqrt{2\pi m}} e^{-c_{tj}^{*2} m/72}. \tag{21}$$

Consequently, for the T-component mixture, with probability at least

$$1 - K\left(e^{-(1-\epsilon)m/20} + \frac{6}{\sqrt{(1-\epsilon)2\pi m}} e^{-(1-\epsilon)m/72}\right) \tag{22}$$

if we reconstruct $\boldsymbol{x}$ by mapping $\boldsymbol{y}$ back onto $\mathbb{R}^N$, more specifically to the d-dimensional hyperplane associated with Gaussian t^*, and denote the reconstruction as $\hat{\boldsymbol{x}}$, then we have $\|\boldsymbol{x} - \hat{\boldsymbol{x}}\| \sim \mathcal{N}(0, \lambda_d(\boldsymbol{\Sigma}_t))$, where $\lambda_d(\boldsymbol{\Sigma}_t)$ is the dth largest eigenvalue of $\boldsymbol{\Sigma}_t$.

The proofs are essentially those in [19], but we have made a few changes, as follows.

- Lemmas 18 and 19 assume that we observe $\boldsymbol{x}$ and have estimates for the mixture parameters. In contrast, we assume known mixture parameters but unknown $\boldsymbol{x}$—we only observe $\boldsymbol{y}$, therefore we must work with projections in $\mathbb{R}^m$, not with the full data in $\mathbb{R}^N$;
- Lemmas 18 and 19 assume that the Gaussians have the same covariance, which clearly does not suit us (even in reduced space, where the covariances are closer to spherical). We address this by noting that the original proofs rely on assigning labels according to minimum Euclidean distance between $\boldsymbol{x}$ and the $\boldsymbol{\mu}_k$ and assuming the worst case of spherical Gaussians with variance $\lambda_{\max}$. Instead, we assign labels according to the Mahalanobis distances between $\boldsymbol{y}$ and the projected $\boldsymbol{\mu}_k^*$, which obviates the assumption of equal covariances.

Importantly, we also quantify the influence of d in the bound for m, since Dasgupta's result makes no assumption of low-rank covariances. If $d > 4/(\epsilon^2)\log(T^2)/2\delta$, then we can use for discrimination between different Gaussians the same measurements that we would need anyway, in order to obtain a d-dimensional projection of $\boldsymbol{x}$ in the correct hyperplane. The converse is also true when $d < 4/(\epsilon^2)\log(T^2)/2\delta$.

B. Block-Coherence Result

We now turn to our second result. We begin by writing the following concentration inequality, which plays a major role in many CS results for sub-Gaussian random measurements:

$$\Pr(|\|\sqrt{N/m}\Phi\boldsymbol{x}\|^2 - \|\boldsymbol{x}\|^2| \geq \epsilon\|\boldsymbol{x}\|^2) \leq 2e^{-mC_0(\epsilon)}. \tag{23}$$

For Gaussian measurements, we have $C_0(\epsilon) = \epsilon^2/4 - \epsilon^3/6$.

Our second main theorem is a modification of [30, Theorem 3.3], which is itself a modification of [31, Theorem 5.2]:

Theorem 2: Let $\boldsymbol{\Psi} \in \mathbb{R}^{N \times (d+1)T}$ be a dictionary with block-coherence μ_B and subcoherence ν such that $d < (\mu_B^{-1} - 1)/(1 + (\nu)/(\mu_B))$. Let $\boldsymbol{\Phi} \in \mathbb{R}^{m \times N}$ be a measurement matrix whose elements satisfy the concentration inequality (23). Then, for any signal $\boldsymbol{\theta}$ that is block 1-sparse in $\boldsymbol{\Psi}$ with block size $d+1$, and for given $\delta \in (0,1)$ and $\tau > 0$, the linear transformation $\boldsymbol{\theta} \longrightarrow \boldsymbol{U\theta} = \sqrt{(N/m)}\boldsymbol{\Phi\Psi\theta}$, $\boldsymbol{U} \in \mathbb{R}^{m \times (d+1)T}$ verifies the RIP with constant δ, with probability at least $1 - e^{-\tau}$, if

$$m \geq \frac{1}{C_0\left(\frac{\delta}{2}\right)}(\log(2T) + (d+1)\log(12/\delta) + \tau). \tag{24}$$

We thus have

$$(1-\delta)\|\boldsymbol{\theta}\|^2 \leq \|\boldsymbol{U\theta}\|^2 \leq (1+\delta)\|\boldsymbol{\theta}\|^2. \tag{25}$$

C. Proof of Theorem 2

The proof is similar to that in [30], which follows the same method as [31] but, due to block-sparsity, has a drastically reduced number of hyperplanes to search over. For sparsity level S, usual CS bounds depend on $\binom{N}{S}$, which is the number of S-dimensional subspaces in $\mathbb{R}^N$. For the mixture of rank-d covariance Gaussians case, however, there exist only T possible combinations of $(d+1)$-sized dictionary atom blocks—i.e., the sparsity is highly structured. In addition, we use results from [25] that ensure exact recovery is possible for certain sparsity levels that depend on the block coherence of $\boldsymbol{\Psi}$.

Following [31], we invoke the Johnson–Lindenstrauss (JL) lemma and the concentration bound (23). First, we consider subdictionary $\boldsymbol{\Psi}_\Lambda$ with $|\Lambda| = d+1$. We cover the unit sphere $\mathbb{R}^d$ with a finite set Q of points such that $\|q\| = 1$ for all $q \in Q$ and, for all $\theta : \|\theta\| = 1$, we have $\min_{q \in Q} \|\theta - q\| \leq \delta/4$. It is known that there exists such a Q with $|Q| \leq (12/\delta)^{d+1}$. We can apply (23) with $\epsilon = (\delta/2)$ to all points $\boldsymbol{\Psi}_\Lambda q$ and get

$$\begin{aligned}(1-\delta/2)\|\boldsymbol{\Psi}_\Lambda q\| &\leq \left\| \underbrace{\sqrt{\frac{N}{m}}\boldsymbol{\Phi\Psi}_\Lambda}_{\boldsymbol{U}_\Lambda} q \right\| \\ &\leq (1+\delta/2)\|\boldsymbol{\Psi}_\Lambda q\|\end{aligned} \tag{26}$$

with probability at least $1 - 2(12/\delta)^{d+1}e^{-mC_0(\delta/2)}$, by the union bound. Defining γ as the smallest number such that $\|\boldsymbol{U}_\Lambda\boldsymbol{\theta}\|^2 \leq (1+\gamma)\|\boldsymbol{\theta}\|^2$ for all $\boldsymbol{\theta}$ supported on Λ, it is proved in [31] that i) $\gamma < \delta \in (0,1)$ and ii) $(1-\gamma)\|\boldsymbol{\theta}\|^2 \leq \|\boldsymbol{U}_\Lambda\boldsymbol{\theta}\|^2 \leq (1+\gamma)\|\boldsymbol{\theta}\|^2$.

We now apply the union bound over all valid subdictionaries—in our case, there are T of them. Thus, the probability that (25) will fail is less than

$$2T\left(\frac{12}{\delta}\right)^{d+1} e^{-mC_0(\frac{\delta}{2})} = 2Te^{-mC_0(\frac{\delta}{2})+(d+1)\log\left(\frac{12}{\delta}\right)}. \quad (27)$$

Now we require that $2Te^{-mC_0(\delta/2)+(d+1)\log((12)/(\delta))} \leq e^{-\tau}$ and we get the result that if

$$m \geq \frac{1}{C_0\left(\frac{\delta}{2}\right)}(\log 2T + (d+1)\log(12/\delta) + \tau) \quad (28)$$

for given $\delta \in (0,1)$ and $\tau > 0$, then with probability at least $1-e^{-\tau}$ the composite matrix $\boldsymbol{U}$ has the restricted isometry property with the prescribed δ. The constant $C_0(\delta/2)$, for a Gaussian measurement matrix, is equal to $\delta^2/16 - \delta^3/48$.

All that is left to do is to relate d with the block-coherence μ_B and subcoherence ν, in order to guarantee stable and efficient recovery in the same sense as [32]. We take a different route than [33], who reaches the result $d \leq 1/16\mu$ using standard coherence. Instead, we can get much less strict limitations on d by using the result in [28] for block-coherence, where we need the condition $L(d+1) < (1/2)(\mu_B^{-1} + d + 1 - d(\nu)/\mu_B)$. With $L = 1$, it is straightforward to obtain

$$d < \frac{\mu_B^{-1} - 1}{1 + \frac{\nu}{\mu_B}} \quad (29)$$

(which for $\nu = 0$ reduces to $d + 1 < \mu_B^{-1}$). This completes the proof.

We point out that condition (29) can be checked for the MFA model learned through the procedure described in Section II, by explicitly computing the quantities ν and μ_B. Moreover, it is also possible to incorporate (29) directly into the learning process, without necessitating *a posteriori* verification. For instance, one might employ a rejection sampling step in tandem with the Gibbs sampler to preclude any solutions that would violate (29).

D. Discussion on the Significance of the Bounds

Theorems 1 and 2 allow us to establish expectations regarding the comparative performance of manifold-based CS, and more generally of MFA-based CS, versus traditional sparsity-based CS. For comparison, and assuming sparsity level S (i.e., $\|\boldsymbol{\theta}\|_0 \leq S$), the current best sparsity-based CS bounds on the number of measurements are $O(S\log(N/S))$ for the case of a subgaussian measurement matrix $\boldsymbol{\Phi}$, and $O(S\log^4(N/S))$ when $\boldsymbol{\Phi}$ is drawn from an orthobasis ensemble, assuming maximum incoherence with the sparsity dictionary [34].

In contrast, our separability and block-sparsity bounds are $\max(d, O(\log T^2))$ and $O(\log(2T)) + O(d+1)$ respectively, and they have no dependence on the ambient dimension N, while the dependence on d (which play a role analogous to the sparsity S) remains no worse than linear. While the dependence on $\log N$ is, in both our results, essentially replaced by a dependence on $\log T$, this works to our considerable advantage because we expect $T \ll N$ by many orders of magnitude. This is consistent with the experimental results we show in the following section, where our reconstruction quality is comparable to that of the best sparsity-based CS algorithms, but using a much smaller fraction of the measurements.

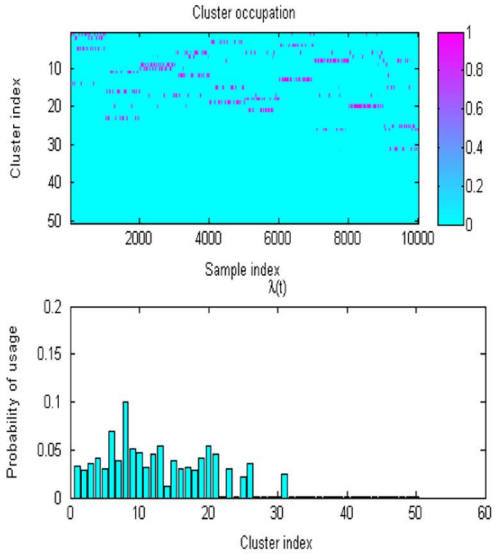
Furthermore, our bounds relate to recent results in [4] for manifold CS, where the sample complexity bound is also linear in d and logarithmic in the manifold volume and condition number. If one notes the fact that these parameters tend to be larger for more geometrically complex manifolds and that, likewise, such manifolds typically require more Gaussians in our MFA model, i.e., a larger value for the parameter T, we can see that T therefore plays an interesting and perhaps unsuspected geometric role.

V. Experimental Results With Real Data

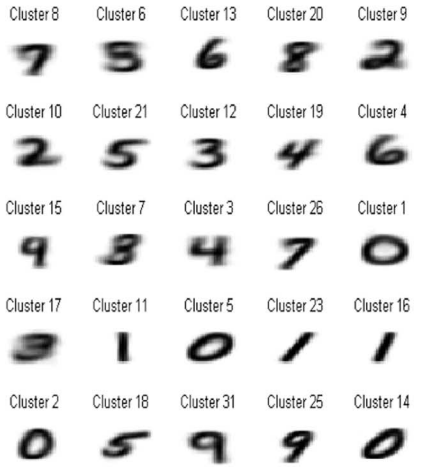
A. Digit Data

The MNIST digit dataset is commonly used for manifold learning. In this experiment, the training set contains 10 000 images (1000 images for each digit "0" through "9", each image of size 28×28). We train the nonparametric MFA model without using the digit labels. The training results are depicted in Fig. 8. The model uses 25 clusters [Fig. 8(a)] and for each cluster the subspace dimension is around 10 [Fig. 8(b)–(c)]. We then use 100 testing images for compressive sensing and reconstruction. The performance is shown in Fig. 9(a) and examples of the reconstruction are given in Fig. 10. With very few measurements we can obtain a reasonable reconstruction. In these and all subsequent examples the components of $\boldsymbol{\Phi}$ are drawn i.i.d. from $\mathcal{N}(0,1)$. As seen in Fig. 9(a), our method slightly outperforms the BOMP algorithm [25] which, like ours, exploits block-sparsity structure but [25] is not probabilistic. We have chosen the block-sparsity level of BOMP (the number of active blocks) to be the same as that inferred by our method (this is a difficult parameter to set in the absence of our MFA model). Also, since BOMP does not include a procedure for learning the dictionary in the first place, we supply it with our learned MFA factor loadings and means, concatenated as explained in Section IV, which constitute the dictionary.

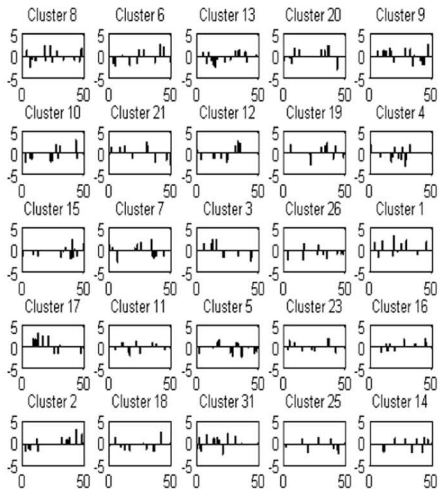
In Figs. 9(a) and 10, CS reconstruction results are also made with the tree-based wavelet CS inversion algorithm developed in [35]. While the method in [35] does not explicitly exploit the properties of the manifold, it does exploit structure in the wavelet coefficients of typical images and has demonstrated state-of-the-art performance relative to almost all other existing CS algorithms (see [35] for details on those comparisons). The comparative results in Fig. 9(a) and 10 show the benefits of leveraging the manifold information, particularly for a small number of measurements, as predicted by the theory in Section IV. As a trade-off, the wavelet-based method outperforms both our algorithm and BOMP for the high-measurement regime. This is due to approximation error, since the wavelet basis is a complete basis for the observation space, while our learned dictionary is under-complete and much smaller.



(a)

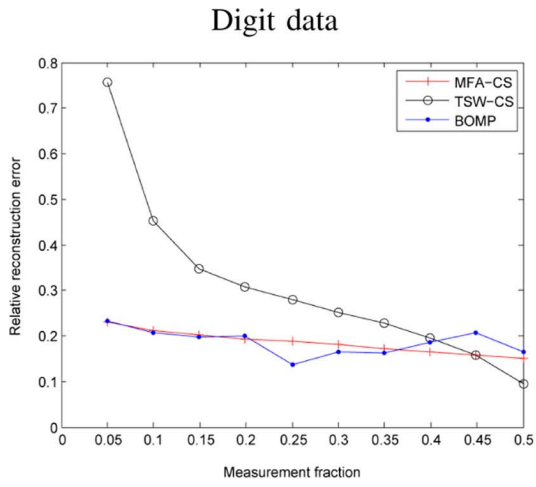


(b)

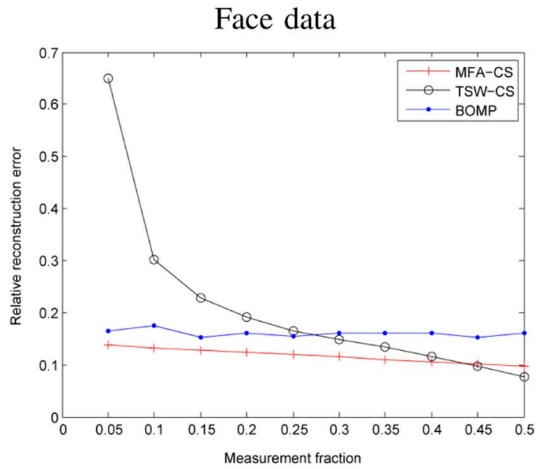


(c)

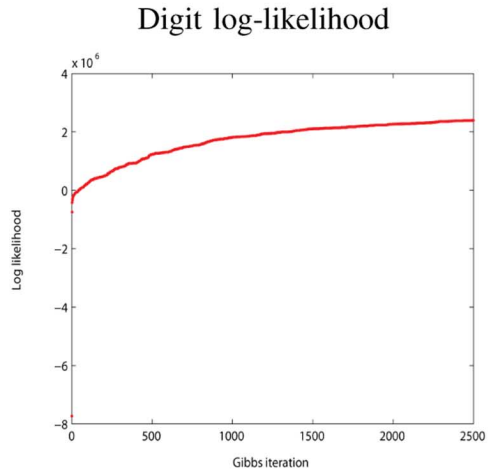
Fig. 8. Training results for the MNIST digit data. (a) The top figure is the cluster occupation probability for each sample, and the bottom figure is the probability of using the clusters. (b) Center for each cluster ($\boldsymbol{\chi}_t$). (c) Factor usage of each cluster ($\boldsymbol{\Delta}_t\ \mathrm{diag}(z_t)$).



(a)



(b)



(c)

Fig. 9. (a) CS reconstruction error for the MNIST digit data. The vertical axis denotes relative error as a function of the number of measurements, where the latter is in fraction of the size of the original image. The MFA-CS results correspond to the proposed method, BOMP to Block-OMP [25] and TSW-CS corresponds to the tree-structured wavelet-based CS inversion developed in [35], using a Haar wavelet. (b) Similar plot for the face data. (c) MNIST log-probability plot as a function of the number of Gibbs iterations (for learning the underlying MFA, with this done "offline" with the training data, prior to CS analysis); the corresponding plot for the face data looks very similar, and we omit it for brevity.

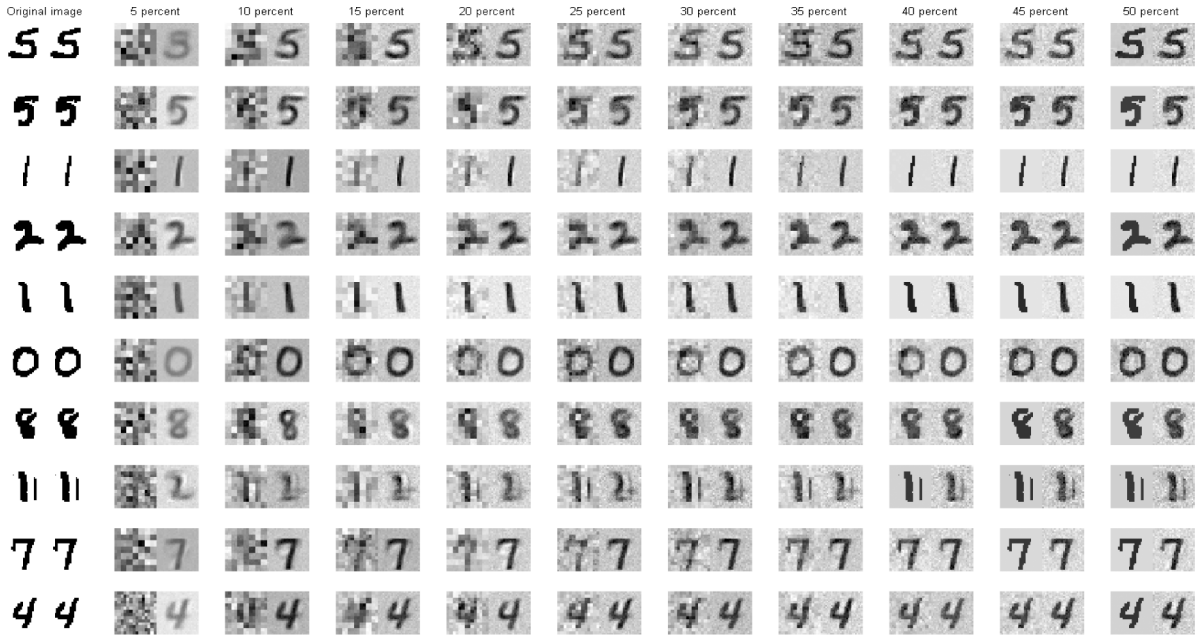


Fig. 10. CS reconstruction results for the MNIST digit data. The left-most column shows the original image (different examples for each row), and the subsequent columns show CS reconstruction as a function of number of measurements, quantified in terms of percent relative to the size of the original image. In each column, the left figure corresponds to the CS inversion method in [35] using a Haar wavelet, and the right subfigure corresponds to the proposed method. The second through eleventh columns correspond to 5% to 50% measurements, in increments of 5%.

For many CS applications, we are most interested in the low-measurement-number regime.

B. Face Data

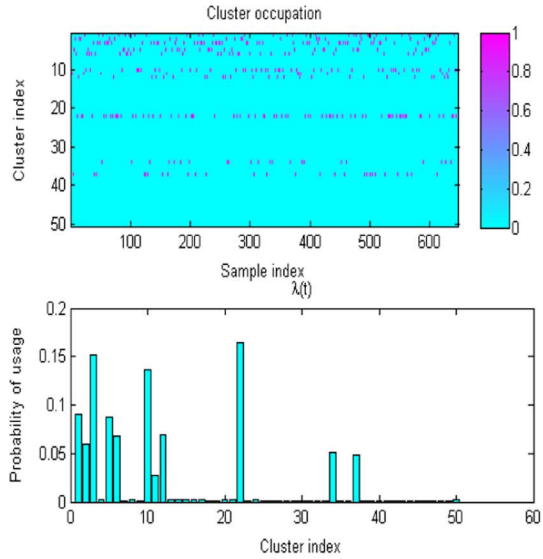
As our last example, we show results on the face dataset used in the Isomap paper [5]. The faces are from the same subject but with different pose and illumination. We use 648 training images, each with dimension 64×64. The training results are depicted in Fig. 11. The model uses 11 clusters [Fig. 11(a)] and for each cluster the subspace dimension is around 13 [Fig. 11(a)–(b)]. We then use 20 (distinct) testing images for compressive sensing and reconstruction. The performance is shown in Fig. 9(b) and examples of the reconstruction are given in Fig. 12. Note again the excellent relative performance, particularly for a small number of measurements, as compared to the state-of-the-art CS algorithm developed in [35] (that exploits wavelet structure, but not manifold information) and to the BOMP algorithm [25], which is not probabilistic. In all examples, note that the wavelet-based approach in [35] eventually provides slightly better performance as the fractional number of measurements becomes large. The MFA-based CS approach is exploiting more prior knowledge than the wavelet-based approach, and therefore it is most advantageous for a relatively small number of CS measurements. The additional prior assumptions about the support of the signals of interest, associated with using the MFA, may introduce small errors with a large number of CS measurements (due to errors in the MFA model). The wavelet-based approach, which makes fewer assumptions about underlying model structure, yields slightly better results in this large-sample regime, but much worse results based on a relatively small number of CS measurements.

VI. Conclusion

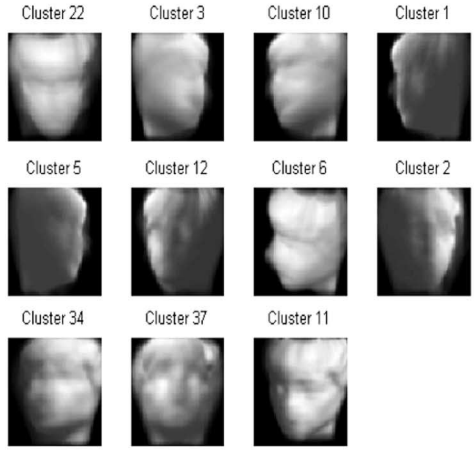
In this paper, we have proposed a nonparametric Bayesian framework to learn MFA models for manifolds, with the advantage of inferring the number of clusters and the subspace dimensions simultaneously from the data. Furthermore, we have shown how this nonparametric MFA can be used to construct dictionaries for compressive sensing of signals from the learned manifold, taking advantage of the block-sparsity of such signals. The CS reconstruction estimate can be efficiently obtained in closed form, using Bayes' rule.

In addition, we have derived theoretical bounds on the number of necessary random measurements in terms of easily computable quantities. It should be noted that Theorem 2 applies to a broader class of signals then the MFA considered here. Any signal living on a union of hyperplanes can be reconstructed with the same guarantees. In fact, the bound is likely loose, since we do not use the fact that the first nonzero coordinate of $\boldsymbol{\theta}$, corresponding to the mean in subdictionary block $\boldsymbol{\Psi}[t]$, is always $||\boldsymbol{\mu}_t||$. We conjecture that the dependence on $d+1$ can be reduced to d. Thus, Theorem 2 is actually true for a superset of the union of hyperplanes considered. Nevertheless, care should be exercised in verifying low dictionary block-coherence and subcoherence, so that the required sparsity level is high enough to be useful.

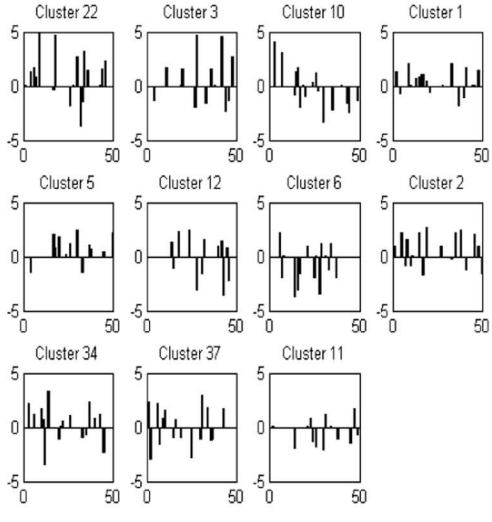
Interestingly, if we interpret the mixture as a manifold model, the covariance rank d is the intrinsic dimension, which plays the same role in the bounds as the sparsity S, as noted by [3] and [4]. In the same vein, the number T of components is related to the manifold condition number introduced by Nyogi *et al.* [36] and used by Wakin and Baraniuk—it models the complexity and curvature of the manifold. The present work thus complements



(a)



(b)



(c)

Fig. 11. Training result for the face data. (a) The top figure is the cluster occupation probability for each sample, and the bottom figure is the probability of using the clusters. (b) Cluster centers ($\boldsymbol{\chi}_t$). (c) Factor usage ($\boldsymbol{\Delta}_t\ \mathrm{diag}(\boldsymbol{z}_t)$) of each cluster.

prior results on CS for manifolds by introducing a general-purpose reconstruction algorithm, filling an existing gap.

APPENDIX

As illustrated in Section II, the model likelihood can be expressed as

$$
\begin{aligned}
&p\left(\{\boldsymbol{x}_i,\hat{\boldsymbol{w}}_i\}_{i=1}^n,\{\tilde{\boldsymbol{A}}_t,\boldsymbol{\Delta}_t,\boldsymbol{\tau}_t,\boldsymbol{z}_t,\boldsymbol{\pi}_t,\boldsymbol{\mu}_t,\alpha_t,v_t\}_{t=1}^T,\eta\right)\\
&=\prod_{i=1}^{n}\Bigg(\mathcal{N}\left(\boldsymbol{x}_i;\tilde{\boldsymbol{A}}_{t(i)}\left(\boldsymbol{\Delta}_{t(i)}\,\mathrm{diag}(\boldsymbol{z}_{t(i)})\right)\hat{\boldsymbol{w}}_i\right.\\
&\quad\left.+\boldsymbol{\mu}_{t(i)},\alpha_{t(i)}^{-1}\boldsymbol{I}_N\right)\mathcal{N}(\hat{\boldsymbol{w}}_i;\boldsymbol{0},\boldsymbol{I}_K)\\
&\quad\times\left(v_{t(i)}\prod_{l=1}^{t(i)-1}(1-v(l))\right)\Bigg)\\
&\quad\times\prod_{t=1}^{T-1}\mathrm{Beta}(v_t;1,\eta)\times\mathrm{Gamma}(\eta;c,d)\\
&\quad\times\prod_{t=1}^{T}\Bigg(\prod_{k=1}^{K}\mathcal{N}(\tilde{\boldsymbol{A}}_{tk};\boldsymbol{0},N^{-1}\boldsymbol{I}_N)\\
&\quad\times\mathrm{Bernoulli}(z_{tk};\pi_{tk})\,\mathrm{Beta}(\pi_{tk};a/K,b(K-1)/K)\\
&\quad\times\mathcal{N}\left(\Delta_{tk};0,\tau_{tk}^{-1}\right)\mathrm{Gamma}(\tau_{tk};e,f)\Bigg)\\
&\quad\times\mathcal{N}(\boldsymbol{\mu}_t;\boldsymbol{\mu},\tau_0^{-1}\boldsymbol{I}_N)\mathrm{Gamma}(\alpha_t;g,h)
\end{aligned}
$$

where $\tilde{\boldsymbol{A}}_{tk}$ denotes the kth column of matrix $\tilde{\boldsymbol{A}}_t$ and Δ_{tk} denotes the kth diagonal element in the diagonal matrix Δ_t. The Gibbs sampling inference algorithm can be derived as follows.

1) Sample the cluster index t_i from

$$
\begin{aligned}
&p(t(i)=t\,|\,-)\\
&\quad\propto\lambda_t\int\mathcal{N}\left(\boldsymbol{x}_i;\tilde{\boldsymbol{A}}_t(\boldsymbol{\Delta}_t\,\mathrm{diag}(\boldsymbol{z}_t))\hat{\boldsymbol{w}}_i+\boldsymbol{\mu}_t,\alpha_t^{-1}\boldsymbol{I}_N\right)\\
&\qquad\times\mathcal{N}(\hat{\boldsymbol{w}}_i;\boldsymbol{0},\boldsymbol{I}_K)\mathrm{d}\hat{\boldsymbol{w}}_i\\
&\quad\propto\lambda_t\mathcal{N}(\boldsymbol{x}_i;\boldsymbol{\mu}_t,\boldsymbol{\Sigma}_t)
\end{aligned}
$$

with $\boldsymbol{\Sigma}_t$ defined in Section II-B and $\lambda_t=v_t\prod_{l=1}^{t-1}(1-v(l))$. After normalization across t, $p(t(i)=t\,|\,-)$ becomes a Multinomial distribution. Due to the low-dimensional subspace structure of $\boldsymbol{\Sigma}_t$, the computation of $\mathcal{N}(\boldsymbol{x}_i;\boldsymbol{\mu}_t,\boldsymbol{\Sigma}_t)$ can be made more efficient by using the Matrix inversion lemma.

2) Sample the DP concentration parameter η from $\mathrm{Gamma}(\eta;c+T-1,d-\sum_{t=1}^{T-1}\log(1-v_t))$.

3) Sample the DP stick weight v_t from $p(v_t\,|\,-)=\mathrm{Beta}(v_t;1+\sum_{i:t(i)=t}1,\eta+\sum_{i:t(i)>t}1)$ $(t=1,2,\ldots,T-1)$ and $v_T=1$.

4) Sample the factor score $\hat{\boldsymbol{w}}_i$ from $p(\hat{\boldsymbol{w}}_i\,|\,t(i),-)=\mathcal{N}(\hat{\boldsymbol{w}}_i;\boldsymbol{\xi}_{t(i)},\boldsymbol{\Lambda}_{t(i)})$ with

$$
\begin{aligned}
\boldsymbol{\Lambda}_{t(i)}&=\Big(\alpha_{t(i)}\,\mathrm{diag}\left(\boldsymbol{z}_{t(i)}\right)\boldsymbol{\Delta}_{t(i)}\tilde{\boldsymbol{A}}_{t(i)}^{\top}\tilde{\boldsymbol{A}}_{t(i)}\boldsymbol{\Delta}_{t(i)}\\
&\qquad\times\mathrm{diag}\left(\boldsymbol{z}_{t(i)}\right)+\boldsymbol{I}_K\Big)^{-1}\\
\boldsymbol{\xi}_{t(i)}&=\boldsymbol{\Lambda}_{t(i)}\left(\alpha_{t(i)}\,\mathrm{diag}\left(\boldsymbol{z}_{t(i)}\right)\boldsymbol{\Delta}_{t(i)}\tilde{\boldsymbol{A}}_{t(i)}^{\top}\left(\boldsymbol{x}_i-\boldsymbol{\mu}_{t(i)}\right)\right)
\end{aligned}
$$

Fig. 12. CS reconstruction results for the face data. The left-most column shows the original image (different examples for each row), and the subsequent columns show CS reconstruction as a function of number of measurements, quantified in terms of percent relative to the size of the original image. In each column, the left figure corresponds to the CS inversion method in [35] using a Haar wavelet, and the right subfigure corresponds to the proposed method. The second through eleventh columns correspond to 5% to 50% measurements, in increments of 5%.

5) $p(z_{tk}, \Delta_{tk} \,|\, -) \propto \text{Bernoulli}(z_{tk}; \pi_{tk}) \mathcal{N}(\Delta_{tk}; 0, \tau_{tk}^{-1}) \times \prod_{i:t(i)=t} \mathcal{N}(\boldsymbol{x}_{ti}^{-k}; \tilde{\boldsymbol{A}}_{tk} \Delta_{tk} z_{tk} \hat{w}_{ki}, \alpha_t^{-1} \boldsymbol{I})$ with $\boldsymbol{x}_{ti}^{-k} \triangleq \boldsymbol{x}_i - \sum_{m \neq k} \tilde{\boldsymbol{A}}_{tm} \Delta_{tm} z_{tm} \hat{w}_{mi} - \boldsymbol{\mu}_t$. Thus, we can sample $\boldsymbol{z}_t$ and $\boldsymbol{\Delta}_t$ from

$$\begin{aligned} p(z_{tk} \,|\, -) &= \int p(z_{tk}, \Delta_{tk} \,|\, -) \mathrm{d}\Delta_{tk} \\ &= \text{Bernoulli}(z_{tk}; \tilde{\pi}_{tk}) \\ p(\Delta_{tk} \,|\, z_{tk}, -) &= z_{tk} \mathcal{N}(\Delta_{tk}; \delta_{tk}, \gamma_{tk}) \\ &\quad + (1 - z_{tk}) \mathcal{N}\left(\Delta_{tk}; 0, \tau_{tk}^{-1}\right) \end{aligned}$$

where

$$\begin{aligned} \log \frac{\tilde{\pi}_{tk}}{1 - \tilde{\pi}_{tk}} &= \log \frac{\pi_{tk}}{1 - \pi_{tk}} + \frac{1}{2} \log \gamma_{tk} \\ &\quad + \frac{\delta_{tk}^2}{2\gamma_{tk}} - \frac{1}{2} \log \tau_{tk}^{-1} \\ \gamma_{tk} &= \left(\tau_{tk} + \alpha_t \tilde{\boldsymbol{A}}_{tk}^\top \tilde{\boldsymbol{A}}_{tk} \sum_{i:t(i)=t} \hat{w}_{ki}^2 \right)^{-1} \\ \delta_{tk} &= \gamma_{tk} \left(\alpha_t \tilde{\boldsymbol{A}}_{tk}^\top \sum_{i:t(i)=t} \boldsymbol{x}_{ti}^{-k} \hat{w}_{ki} \right). \end{aligned}$$

6) Sample the Bernoulli parameter $\boldsymbol{\pi}_t$ from $p(\pi_{tk} \,|\, -) = \text{Beta}(\pi_{tk}; a/K + z_{tk}, b(K-1)/K + 1 - z_{tk})$.
7) Sample the precision parameter τ_{tk} from $p(\tau_{tk} \,|\, -) = \text{Gamma}(\tau_{tk}; e + (1/2), f + (1/2)\Delta_{tk}^2)$ if $z_{tk} = 1$; otherwise set $\tau_{tk} = 1$. The algorithm also works if τ_{tk} is always fixed to 1 without update.
8) Sample the mean vector for each cluster $\boldsymbol{\mu}_t$ from $p(\boldsymbol{\mu}_t \,|\, -) = \mathcal{N}(\boldsymbol{\mu}_t; \boldsymbol{\zeta}_t, \boldsymbol{\Gamma}_t)$ where

$$\begin{aligned} \boldsymbol{\Gamma}_t &= \left(\tau_0 \boldsymbol{I}_N + \alpha_t \sum_{i:t(i)=t} 1 \right)^{-1}; \\ \boldsymbol{\zeta}_t &= \boldsymbol{\Gamma}_t \left(\tau_0 \boldsymbol{\mu} + \alpha_t \sum_{i:t(i)=t} (\boldsymbol{x}_i - \tilde{\boldsymbol{A}}_t (\boldsymbol{\Delta}_t \,\mathrm{diag}(\boldsymbol{z}_t)) \hat{\boldsymbol{w}}_i) \right). \end{aligned}$$

9) Sample the factor loading matrix $\tilde{\boldsymbol{A}}_t$ from

$$\begin{aligned} &p(\tilde{\boldsymbol{A}}_{jt} \,|\, -) \\ &\quad \propto \mathcal{N}(\tilde{\boldsymbol{A}}_{jt}; \boldsymbol{0}, N^{-1} \boldsymbol{I}_K) \\ &\qquad \times \prod_{i:t(i)=t} \mathcal{N}\left(x_{ji}; \tilde{\boldsymbol{A}}_{jt} (\boldsymbol{\Delta}_t \,\mathrm{diag}(\boldsymbol{z}_t)) \hat{\boldsymbol{w}}_i + \mu_{jt}, \alpha_t^{-1} \right) \\ &\quad = \mathcal{N}(\tilde{\boldsymbol{A}}_{jt}; \boldsymbol{\varrho}_{jt}, \boldsymbol{\Xi}_{jt}) \end{aligned}$$

where $\tilde{\boldsymbol{A}}_{jt}$ denotes the jth row of matrix $\tilde{\boldsymbol{A}}_t$ and

$$\boldsymbol{\Xi}_{jt} = \left(N\boldsymbol{I}_K + \alpha_t \operatorname{diag}(\boldsymbol{z}_t)\boldsymbol{\Delta}_t \times \sum_{i:t(i)=t} \hat{\boldsymbol{w}}_i \hat{\boldsymbol{w}}_i^\top \boldsymbol{\Delta}_t \operatorname{diag}(\boldsymbol{z}_t) \right)$$

$$\boldsymbol{\varrho}_{jt} = \boldsymbol{\Xi}_{jt} \left(\alpha_t \operatorname{diag}(\boldsymbol{z}_t)\boldsymbol{\Delta}_t \sum_{i:t(i)=t} \hat{\boldsymbol{w}}_i (x_{ji} - \mu_{jt}) \right).$$

10) Sample the precision of the additive noise α_t from $p(\alpha_t \,|\, -) = \text{Gamma}(\alpha_t; g_t, h_t)$ with

$$g_t = g + \frac{N}{2} \sum_{i:t(i)=t} 1;$$

$$h_t = h + \frac{1}{2} \sum_{j=1}^{N} \sum_{i:t(i)=t} |x_{ji} - \tilde{\boldsymbol{A}}_{jt}(\boldsymbol{\Delta}_t \operatorname{diag}(\boldsymbol{z}_t))\hat{\boldsymbol{w}}_i - \mu_{jt}|^2.$$

REFERENCES

[1] E. Candès and M. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, 2008.

[2] R. Baraniuk, V. Cevher, M. Duarte, and C. Hegde, "Model-based compressive sensing," 2008, Preprint.

[3] R. Baraniuk and M. Wakin, "Random projections of smooth manifolds," *Found. Comput. Math.*, vol. 9, no. 1, pp. 51–77, 2009.

[4] M. Wakin, "Manifold-based signal recovery and parameter estimation from compressive measurements," 2008, Preprint.

[5] J. Tenenbaum, V. Silva, and J. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[6] M. Tipping and C. Bishop, "Mixtures of probabilistic principal component analyzers," *Neural Comput.*, vol. 11, no. 2, pp. 443–482, 1999.

[7] Z. Ghahramani and G. Hinton, 'The em algorithm for mixtures of factor analyzers," Univ. of Toronto, Toronto, ON, Canada, Tech. Rep. CRG-TR-96-1, 1997.

[8] J. Sethuraman, "A constructive definition of the Dirichlet prior," *Statistica Sinica*, vol. 2, pp. 639–650, 2001.

[9] J. Paisley and L. Carin, "Nonparametric factor analysis with beta process priors," in *Proc. ACM 26th Annu. Int. Conf. Machine Learning*, New York, NY, 2009.

[10] M. Brand, "Charting a manifold," *Adv. Neural Inf. Process. Syst.*, pp. 985–992, 2003.

[11] Z. Ghahramani and M. Beal, "Variational inference for Bayesian mixtures of factor analysers," *Adv. Neural Inf. Process.Systems*, vol. 12, pp. 449–455, 2000.

[12] A. Utsugi and T. Kumagai, "Bayesian analysis of mixtures of factor analyzers," *Neural Comput.*, vol. 13, no. 5, pp. 993–1002, 2001.

[13] G. Hinton, P. Dayan, and M. Revow, "Modeling the manifolds of images of handwritten digits," *IEEE Trans. Neural Netw.*, vol. 8, no. 1, pp. 65–74, 1997.

[14] A. Kannan, N. Jojic, and B. Frey, "Fast transformation-invariant component analysis," *Int. J. Comput. Vis.*, vol. 77, no. 1, pp. 87–101, 2008.

[15] J. Verbeek, "Learning nonlinear image manifolds by global alignment of locally linear models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 8, pp. 1236–1250, 2006.

[16] T. S. Ferguson, "A Bayesian analysis of some nonparametric problems," *Ann. Stat.*, vol. 1, no. 2, pp. 209–230, 1973.

[17] R. Thibaux and M. Jordan, "Hierarchical beta processes and the Indian buffet process," in *Proc. Int. Conf. Artificial Intelligence Statistics*, 2007.

[18] C. Antoniak, "Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems," *Ann. Stat.*, pp. 1152–1174, 1974.

[19] S. Dasgupta, "Learning probability distributions," Ph.D. dissertation, Univ. of California, Berekely, 2000.

[20] H. Ishwaran and L. F. James, "Gibbs sampling methods for stick-breaking priors," *J. Amer. Stat. Assoc.*, vol. 96, pp. 161–173, 2001.

[21] M. Tipping, "Sparse Bayesian learning and the relevance vector machine," *J. Mach. Learn. Res.*, vol. 1, pp. 211–244, 2001.

[22] S. Jain and R. Neal, "A split-merge Markov chain Monte Carlo procedure for the Dirichlet process mixture model," *J. Comput. Graphic. Stat.*, 2000.

[23] O. Papaspiliopoulos and G. Roberts, "Retrospective Markov chain Monte Carlo methods for Dirichlet process hierarchical models," *Biometrika*, vol. 95, no. 1, pp. 169–186, 2008.

[24] D. Blei and M. Jordan, "Variational inference for Dirichlet process mixtures," *Bayesian Anal.*, vol. 1, no. 1, pp. 121–144, 2006.

[25] Y. Eldar, P. Kuppinger, and H. Bölcskei, "Compressed sensing of block-sparse signals: Uncertainty relations and efficient recovery," 2009, Arxiv preprint arXiv:0906.3173.

[26] A. Jasra, C. Holmes, and D. Stephens, "Markov chain Monte Carlo methods and the label switching problem in Bayesian mixture modeling," *Stat. Sci.*, vol. 20, pp. 50–67, 2005.

[27] M. Stephens, "Dealing with label switching in mixture models," *J. Roy. Stat. Soc. Series B*, vol. 62, pp. 795–809, 2000.

[28] Y. Eldar and H. Bolcskei, "Block-sparsity: Coherence and efficient recovery," 2008, Arxiv preprint arXiv:0812.0329.

[29] Y. Eldar and M. Mishali, "Robust recovery of signals from a union of subspaces," 2008, preprint.

[30] T. Blumensath and M. Davies, "Sampling theorems for signals from the union of linear subspaces," *IEEE Trans. Inf. Theory*, vol. 55, no. 4, pp. 1872–1882, 2009.

[31] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin, "A simple proof of the restricted isometry property for random matrices," *Construct. Approxim.*, vol. 28, no. 3, pp. 253–263, 2008.

[32] J. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Trans. Inf. Theory*, vol. 50, no. 10, pp. 2231–2242, 2004.

[33] H. Rauhut, K. Schnass, and P. Vandergheynst, "Compressed sensing and redundant dictionaries," *IEEE Trans. Inf. Theory*, vol. 54, no. 5, pp. 2210–2219, 2008.

[34] M. Rudelson and R. Vershynin, "On sparse reconstruction from Fourier and Gaussian measurements," *Commun. Pure Appl. Math.*, vol. 61, no. 8, 2008.

[35] L. He and L. Carin, "Exploiting structure in wavelet-based Bayesian compressive sensing," *IEEE Trans. Signal Process.*, vol. 57, no. 9, pp. 3488–3497, Sep. 2009.

[36] P. Niyogi, S. Smale, and S. Weinberger, "Finding the homology of submanifolds with high confidence from random samples," *Discrete Comput. Geomet.*, vol. 39, no. 1, pp. 419–441, 2008.

Minhua Chen was born in Wuhan, China, on October 25, 1982. He received the B.S. and M.S. degree in electrical engineering at Tsinghua University, Beijing, China, in 2004 and 2006 respectively. He received the Ph.D. degree at Duke University, Durham, NC, in 2009.

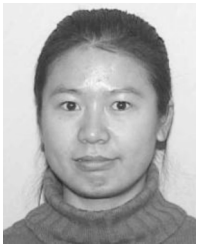
His research interest is in machine learning and signal processing.

Jorge Silva received the E.E., M.Sc., and Ph.D. degrees in electrical and computer engineering from the Instituto Superior Técnico (IST), Lisbon, Portugal, in 1993, 1999, and 2007, respectively.

He was a Researcher at the Instituto de Engenharia de Sistemase Computadores (INESC) from 1993 to 1996, under a PRAXIS XXI award and at the Instituto de Sistemas e Robótica (ISR), Lisbon, Portugal, from 2003 to 2007. He was a Teaching Assistant and later Adjunct Professor at the Instituto Superior de Engenharia de Lisboa (ISEL) from 1996 to 2007. In the same period, he did consulting and research and development work for major Portuguese utility and transportation companies. He is currently a Research Scientist at Duke University, Durham, NC, where he is working on statistical models for very high-dimensional data. His research interests include manifold learning, kernel methods, nonlinear prediction, and filtering and computer vision.

John Paisley received the B.S., M.S., and Ph.D. degrees in electrical and computer engineering from Duke University, Durham, NC, in 2004, 2007, and 2010, respectively.

He is currently a Postdoctoral Research Assistant in the Computer Science Department at Princeton University, Princeton, NJ. His research interests include Bayesian models and machine learning.



Chunping Wang received the B.S. and M.S. degrees in electrical engineering from Tsinghua University, Beijing, China, in 2001 and 2003, respectively.

She is currently a graduate student in the Department of Electrical and Computer Engineering at Duke University, Durham, NC, focusing on machine learning and data mining. Her research interests include learning with incomplete data, multitask learning, and collaborative filtering.

David Dunson is a Professor of Statistical Science at Duke University, Durham, NC. His research focuses on the development and application of novel Bayesian statistical methods motivated by high-dimensional and complex data sets. A particular emphasis is on nonparametric Bayesian methods that avoid assumptions, such as normality and linearity, and on latent factor models that allow dimensionality reduction in massive-dimensional settings. Recent projects have developed sparse latent factor models that scale to massive dimensions and improve performance in predicting disease and other phenotypes based on high-dimensional and longitudinal biomarkers. Related methods can be used for combining high-dimensional data from different sources and for massive-dimensional variable selection.

Dr. Dunson is a Fellow of the American Statistical Association and of the Institute of Mathematical Statistics. He received the 2007 Mortimer Spiegelman Award for the top public health statistician, the 2010 Myrto Lefkopoulou Distinguished Lectureship at Harvard University, and the 2010 COPSS Presidents' Award for the top statistician under 41.

Lawrence Carin (SM'96–F'01) was born in Washington DC on March 25, 1963. He received the B.S., M.S., and Ph.D. degrees in electrical engineering at the University of Maryland, College Park, in 1985, 1986, and 1989, respectively.

In 1989, he joined the Electrical Engineering Department at Polytechnic University, Brooklyn, NY, as an Assistant Professor, and became an Associate Professor there in 1994. In September 1995, he joined the Electrical Engineering Department at Duke University, Durham, NC, where he is currently the William H. Younger Professor of Engineering. He is a co-founder of Signal Innovations Group, Inc. (SIG), a small business, where he serves as the Director of Technology. His current research interests include signal processing, sensing, and machine learning. He has published over 200 peer-reviewed papers.

Dr. Carin is a member of the Tau Beta Pi and Eta Kappa Nu honor societies.