Nonparametric Estimation of Multi-View Latent Variable Models

Le Song LSONG@CC.GATECH.EDU
Georgia Institute of Technology, Atlanta, GA 30345 USA
Animashree Anandkumar A.ANANDKUMAR@UCI.EDU
University of California, Irvine, CA 92697, USA
Bo Dai, Bo Xie BODAI,BXIE33@GATECH.EDU
Georgia Institute of Technology, Atlanta, GA 30345 USA

Abstract

Spectral methods have greatly advanced the estimation of latent variable models, generating a sequence of novel and efficient algorithms with strong theoretical guarantees. However, current spectral algorithms are largely restricted to mixtures of discrete or Gaussian distributions. In this paper, we propose a kernel method for learning multi-view latent variable models, allowing each mixture component to be nonparametric and learned from data in an *unsupervised* fashion. The key idea of our method is to embed the joint distribution of a multi-view latent variable model into a reproducing kernel Hilbert space, and then the latent parameters are recovered using a robust tensor power method. We establish that the sample complexity for the proposed method is quadratic in the number of latent components and is a low order polynomial in the other relevant parameters. Thus, our nonparametric tensor approach to learning latent variable models enjoys good sample and computational efficiencies. As a special case of our framework, we also obtain a first *unsupervised* conditional density estimator of the kind with provable guarantees. In both synthetic and real world datasets, the nonparametric tensor power method compares favorably to EM algorithm and other spectral algorithms.

1. Introduction

Recently, there is a surge of interest in designing spectral algorithms for estimating the parameters of latent variable models (Hsu et al., 2009; Song et al., 2010; Parikh et al., 2011; Song et al., 2011; Foster et al., 2012; Anandkumar et al., 2012a;b). Compared to the Expectation-Maximization (EM) algorithm (Dempster et al., 1977) traditionally used for this task, spectral algorithms are better in terms of their computational efficiency and provable guarantees. However, current spectral algorithms are largely restricted to mixture of discrete or Gaussian distributions, e.g. (Anandkumar et al., 2012a; Hsu & Kakade, 2013). When the mixture components are distributions other than these standard distributions, the theoretical guarantees for these algorithms are no longer applicable, and their empirical performance can be very poor.

We propose a kernel method for obtaining sufficient statistics of a multi-view latent variable model (for $\ell \geq 3$),

$$\mathbb{P}\left(\{X_t\}_{t\in[\ell]}\right) = \sum\nolimits_{h\in[k]} \mathbb{P}(h) \cdot \prod\nolimits_{t\in[\ell]} \mathbb{P}(X_t|h), \quad (1)$$

given samples only from the observed variables $\{X_t\}_{t\in[\ell]}$, but *not* the hidden variable H. These statistics allow us to answer integral query, $\int_{\mathcal{X}} f(x_t)\, d\mathbb{P}(x_t|h)$, for functions f from a reproducing kernel Hilbert space (RKHS) *without* the need to assume any parametric form for the involved latent component $\mathbb{P}(X_t|h)$ (we call this setting "*unsupervised*"). Note that this is a very challenging problem, since we do not have samples to directly estimate $\mathbb{P}(X_t|h)$. Hence traditional kernel density estimator does not apply. Furthermore, the nonparametric form of $\mathbb{P}(X_t|h)$ renders previous spectral methods inapplicable.

Our solution is to embed the distribution of the observed variables in such a model into a reproducing kernel Hilbert space, and exploit tensor decomposition of the embedded distribution (or covariance operators) to recover the unobserved embedding $\mu_{X_t|h} = \int_{\mathcal{X}} \phi(x)\, d\mathbb{P}(x|h)$ of the mixture components. The key computation of our algorithm involves a kernel singular value decomposition of the two-view covariance operator, followed by a robust tensor power method on the three-view covariance operator. These standard matrix operations makes the algorithm very efficient and easy to deploy.

Although kernel methods have been previously applied to learning latent variable models, none of them can provably recover the exact latent component $\mathbb{P}(X_t|h)$ or its sufficient statitiscs to support integral query on this distribution. For

Proceedings of the 31^{st} International Conference on Machine Learning, Beijing, China, 2014. JMLR: W&CP volume 32.

instance, Song et al. (2010; 2011); Song & Dai (2013) estimated an (unknown) invertible transformation of the sufficient statistics of the latent component $\mathbb{P}(X_t|h)$, and only supported integral query associated with the distribution of the observed variables. Sgouritsa et al. (2013) used kernel independence measure to cluster data points, and treated each cluster as a latent component. Besides computational issues, it is also difficult to provide theoretical guarantees to such an approach since the clustering step only finds a local minimum. Benaglia et al. (2009) designed an EM-like algorithm for learning the conditional densities in latent variable models. This algorithm alternates between the E-step, proportional assignment of data points to components, and the M-step, kernel density estimation based on weighted data points. Similarly, theoretical analysis of such a local search heurstic is difficult.

The kernel algorithm proposed in this paper is also significantly more general than the previous spectral algorithms which work only for distributions with parametric assumptions (Anandkumar et al., 2012a; Hsu & Kakade, 2013). In fact, when we use the delta kernel, our algorithm recovers the previous algorithm of Anandkumar et al. (2012a) for discrete mixture components as a special case. When we use universal kernels, such as the Gaussian RBF kernel, our algorithm can recover Gaussian mixture components as well as mixture components with other distributions. In this sense, our work also provides a unifying framework for previous spectral algorithms. We prove sample complexity bounds for the nonparametric tensor power method and show that it is both computational and sample efficient. As a special case of our framework, we also obtain a first *unsupervised* conditional density estimator of the kind with provable guarantees. Furthermore, our approach can also be generalized to other latent variable learning tasks such as independent component analysis and latent variable models with Dirichlet priors.

Experimentally, we corroborate our theoretical results by comparing our algorithm to the EM algorithm and previous spectral algorithms. We show that when the model assumptions are correct for the EM algorithm and previous spectral algorithms, our algorithm converges in terms of estimation error to these competitors. In the opposite cases when the model assumptions are incorrect, our algorithm is able to adapt to the nonparametric mixture components and beating alternatives by a very large margin.

2. Notation

We denote by X a random variable with domain $\mathcal{X}$, and refer to instantiations of X by the lower case character, x. We endow $\mathcal{X}$ with some σ-algebra $\mathscr{A}$ and denote a distributions (with respect to $\mathscr{A}$) on $\mathcal{X}$ by $\mathbb{P}(X)$. For the multi-view model in equation (1), we also deal with multiple random variables, $X_1, X_2, \ldots, X_\ell$, with joint distribution $\mathbb{P}(X_1, X_2, \ldots, X_\ell)$. For simplicity of notation, we assume that the domains of all $X_t, t \in [\ell]$ are the same, but the methodology applies to the cases where they have different domains. Furthermore, we denote by H a hidden variable with domain $\mathcal{H}$ and distribution $\mathbb{P}(H)$.

A *reproducing kernel Hilbert space (RKHS)* $\mathcal{F}$ on $\mathcal{X}$ with a kernel $\kappa(x, x')$ is a Hilbert space of functions $f(\cdot) : \mathcal{X} \mapsto \mathbb{R}$ with inner product $\langle \cdot, \cdot \rangle_{\mathcal{F}}$. Its element $\kappa(x, \cdot)$ satisfies the reproducing property: $\langle f(\cdot), \kappa(x, \cdot) \rangle_{\mathcal{F}} = f(x)$, and consequently, $\langle \kappa(x, \cdot), \kappa(x', \cdot) \rangle_{\mathcal{F}} = \kappa(x, x')$, meaning that we can view the evaluation of a function f at any point $x \in \mathcal{X}$ as an inner product. Alternatively, $\kappa(x, \cdot)$ can be viewed as an implicit feature map $\phi(x)$ where $\kappa(x, x') = \langle \phi(x), \phi(x') \rangle_{\mathcal{F}}$. In this paper, we will focus on $\mathcal{X} = \mathbb{R}^d$, and the *normalized* Gaussian RBF kernel

$$\kappa(x, x') = \exp(-\|x - x'\|^2 / (2s^2)) / (\sqrt{2\pi} s^d). \quad (2)$$

But kernel functions have also been defined on graphs, time series, dynamical systems, images and other structured objects (Schölkopf et al., 2004). Thus the methodology presented below can be readily generalized to a diverse range of data types as long as kernel functions are defined.

3. Kernel Embedding of Distributions

Kernel embeddings of distributions are *implicit* mappings of distributions into potentially *infinite* dimensional RKHS. The kernel embedding approach represents a distribution by an element in the RKHS associated with a kernel function (Smola et al., 2007),

$$\mu_X := \mathbb{E}_X [\phi(X)] = \int_{\mathcal{X}} \phi(x) \, d\mathbb{P}(x), \quad (3)$$

where the distribution is mapped to its expected feature map, *i.e.*, to a point in a potentially infinite-dimensional and implicit feature space. By the reproducing property of an RKHS, the kernel embedding is a sufficient statistic for integral query $\forall f \in \mathcal{F}$, *i.e.*, $\int_{\mathcal{X}} f(x) \, d\mathbb{P}(x) = \langle \mu_X, f \rangle_{\mathcal{F}}$. Kernel embedding of distributions has rich representational power. The mapping is injective for characteristic kernels (Sriperumbudur et al., 2008). That is, if two distributions, $\mathbb{P}(X)$ and $\mathbb{Q}(X)$, are different, they are mapped to two distinct points in the RKHS. For domain $\mathbb{R}^d$, many commonly used kernels are characteristic, such as the normalized Gaussian RBF kernel.

Kernel embeddings can be readily generalized to joint distributions of two or more variables using tensor product feature maps. We can embed the joint distribution of two variables X_1 and X_2 into a tensor product feature space $\mathcal{F} \times \mathcal{F}$ by $\mathcal{C}_{X_1 X_2} := \int_{\mathcal{X} \times \mathcal{X}} \phi(x_1) \otimes \phi(x_2) \, d\mathbb{P}(x_1, x_2)$, where the reproducing kernel for the tensor product features satisfies $\langle \phi(x_1) \otimes \phi(x_2), \phi(x_1') \otimes \phi(x_2') \rangle_{\mathcal{F} \times \mathcal{F}} = \kappa(x_1, x_1') \, \kappa(x_2, x_2')$. By analogy, we can also define $\mathcal{C}_{X_1 X_2 X_3} := \mathbb{E}_{X_1 X_2 X_3} [\phi(X_1) \otimes \phi(X_2) \otimes \phi(X_3)]$.

Given a sample $\mathcal{D}_X = \{x^1, \ldots, x^m\}$ of size m drawn *i.i.d.* from $\mathbb{P}(X)$, the empirical kernel embedding can be estimated simply as $\widehat{\mu}_X = \frac{1}{m}\sum_{i=1}^m \phi(x^i)$ with an error $\|\widehat{\mu}_X - \mu_X\|_{\mathcal{F}}$ scaling as $O_p(m^{-\frac{1}{2}})$ (Smola et al., 2007). Similarly, $\mathcal{C}_{X_1X_2}$ and $\mathcal{C}_{X_1X_2X_3}$ can be estimated as $\widehat{\mathcal{C}}_{X_1X_2} = \frac{1}{m}\sum_{i=1}^m \phi(x_1^i) \otimes \phi(x_2^i)$, and $\widehat{\mathcal{C}}_{X_{1:3}} = \frac{1}{m}\sum_{i=1}^m \phi(x_1^i) \otimes \phi(x_2^i) \otimes \phi(x_3^i)$ respectively. Note that we never explicitly compute the feature maps $\phi(x)$ for each data point. Instead, most of the computation required for subsequent statistical inference using kernel embeddings can be reduced to the Gram matrix manipulation.

3.1. Kernel Embedding as Multi-Linear Operator

The joint embeddings can also be viewed as an uncentered covariance operator $\mathcal{C}_{X_1X_2} : \mathcal{F} \mapsto \mathcal{F}$ by the standard equivalence between a tensor product feature and a linear map. That is, given two functions $f_1, f_2 \in \mathcal{F}$, their covariance can be computed by $\mathbb{E}_{X_1X_2}[f_1(X_1)f_2(X_2)] = \langle f_1, \mathcal{C}_{X_1X_2} f_2\rangle_{\mathcal{F}}$, or equivalently $\langle f_1 \otimes f_2, \mathcal{C}_{X_1X_2}\rangle_{\mathcal{F}\times\mathcal{F}}$, where in the former we view $\mathcal{C}_{XY}$ as an operator while in the latter we view it as an element in tensor product feature space. By analogy, $\mathcal{C}_{X_1X_2X_3}$ (with shorthand $\mathcal{C}_{X_{1:3}}$) can be regarded as a multi-linear operator from $\mathcal{F} \times \mathcal{F} \times \mathcal{F}$ to $\mathbb{R}$. It will be clear from the context whether we use $\mathcal{C}_{X_{1:3}}$ as an operator between two spaces or as an element from a tensor product feature space. For generic introduction to tensors, please see (Kolda & Bader, 2009).

In the multi-linear operator view, the application of $\mathcal{C}_{X_{1:3}}$ to a set of elements $\{f_1, f_2, f_3 \in \mathcal{F}\}$ can be defined using the inner product from the tensor product feature space, *i.e.*,

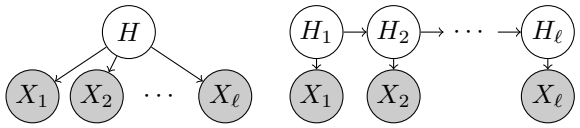
$$\mathcal{C}_{X_{1:3}} \times_1 f_1 \times_2 f_2 \times_3 f_3 := \langle \mathcal{C}_{X_{1:3}},\ f_1 \otimes f_2 \otimes f_3\rangle_{\mathcal{F}^3}$$

which is further equal to $\mathbb{E}_{X_1X_2X_3}[\prod_{t\in[3]} \langle \phi(X_t),\ f_t\rangle_{\mathcal{F}}]$. Furthermore, we can define the Hilbert-Schmidt norm $\|\cdot\|$ as $\|\mathcal{C}_{X_{1:3}}\|^2 = \sum_{i_1,i_2,i_3=1}^{\infty} (\mathcal{C}_{X_{1:3}} \times_1 u_{i_1} \times_2 u_{i_2} \times_3 u_{i_3})^2$ using three collections of orthonormal bases $\{u_{i_1}\}_{i_1=1}^{\infty}$, $\{u_{i_2}\}_{i_2=1}^{\infty}$, and $\{u_{i_3}\}_{i_3=1}^{\infty}$.

The joint embedding, $\mathcal{C}_{X_1X_2}$, can be viewed as infinite dimensional matrices. For instance, we can perform singular value decomposition $\mathcal{C}_{X_1X_2} = \sum_{i=1}^{\infty} \sigma_i \cdot u_{i_1} \otimes u_{i_2}$, where $\sigma_i \in \mathbb{R}$ are singular values ordered in nonincreasing manner, and $\{u_{i_1}\}_{i_1=1}^{\infty} \subset \mathcal{F}, \{u_{i_2}\}_{i_2=1}^{\infty} \subset \mathcal{F}$ are singular vectors and orthonormal bases. The rank of $\mathcal{C}_{X_1X_2}$ is the smallest k such that $\sigma_i = 0$ for $i > k$.

4. Multi-View Latent Variable Models

Multi-view latent variable models studied in this paper are a special class of Bayesian networks in which (*i*) observed variables $X_1, X_2, \ldots, X_\ell$ are conditionally independent given a **discrete** latent variable H, and (*ii*) the conditional distributions, $\mathbb{P}(X_t|H)$, of the $X_t, t \in [\ell]$ given the hidden variable H can be different. The conditional independent structure of a multi-view latent variable model



(a) Naïve Bayes model (b) Hidden Markov model

Figure 1. Examples of multi-view latent variable models.

is illustrated in Figure 1(a), and many complicated graphical models, such as the hidden Markov model in Figure 1(b), can be reduced to a multi-view latent variable model. **For simplicity of exposition, we will explain our method using the model with symmetric view**. That is the conditional distribution are the same for each view, *i.e.*, $\mathbb{P}(X|h) = \mathbb{P}(X_1|h) = \mathbb{P}(X_2|h) = \mathbb{P}(X_3|h)$. In Appendix 9, we will show that multi-view models with different views can be reduced to ones with symmetric view.

4.1. Conditional Embedding Operator

For simplicity of exposition, we focus on a simple model with three observed variables ($\ell = 3$). Suppose $H \in [k]$, then we can embed each conditional distribution $\mathbb{P}(X|h)$ corresponding to a particular value of $H = h$ as

$$\mu_{X|h} = \int_{\mathcal{X}} \phi(x)\, d\mathbb{P}(x|h). \tag{4}$$

If we vary the value of H, we obtain the kernel embedding for different $\mathbb{P}(X|h)$. Conceptually, we can tile these embeddings into a matrix (with infinite number of rows)

$$\mathcal{C}_{X|H} = \left(\mu_{X|h=1}, \mu_{X|h=2}, \ldots, \mu_{X|h=k}\right), \tag{5}$$

which is called the conditional embedding operator. If we use the standard basis e_h in $\mathbb{R}^k$ to represent each value of h, we can retrieve each $\mu_{X|h}$ from $\mathcal{C}_{X|H}$ by

$$\mu_{X|h} = \mathcal{C}_{X|H} e_h \tag{6}$$

Once we have the conditional embedding $\mu_{X|h}$, we can compute the conditional expectation of a function $f \in \mathcal{F}$ as $\int_{\mathcal{X}} f(x)\, d\mathbb{P}(x|h) = \langle f, \mu_{X|h}\rangle_{\mathcal{F}}$.

Remarks. For data from $\mathbb{R}^d$ and the normalized Gaussian RBF kernel in (2), the conditional density $p(x|h)$ exists, and it can be approximated by the embedding as $\widetilde{p}(x|h) := \langle \phi(x), \mu_{X|h}\rangle_{\mathcal{F}} = \mathbb{E}_{X|h}[\kappa(x, X)]$. Essentially, this is the convolution of the conditional density with the kernel function. For continuous density $p(x|h)$ with suitable smoothness conditions, the approximation error is of the order (Wasserman, 2006)

$$|p(x|h) - \widetilde{p}(x|h)| = O(s^2). \tag{7}$$

4.2. Factorized Kernel Embedding

For multi-view latent variable models, $\mathbb{P}(X_1, X_2)$ and $\mathbb{P}(X_1, X_2, X_3)$, can be factorized respectively as

$$\mathbb{P}(x_1, x_2) = \sum\nolimits_{h\in[k]} \mathbb{P}(x_1|h)\, \mathbb{P}(x_2|h)\, \mathbb{P}(h), \text{ and}$$

$$\mathbb{P}(x_1, x_2, x_3) = \sum\nolimits_{h\in[k]} \mathbb{P}(x_1|h)\, \mathbb{P}(x_2|h)\, \mathbb{P}(x_3|h)\, \mathbb{P}(h).$$

Since we assume the hidden variable $H \in [k]$ is discrete, we let $\pi_h := \mathbb{P}(h)$. Furthermore, if we apply Kronecker delta kernel $\delta(h, h')$ with feature map e_h, then the embeddings for $\mathbb{P}(H)$

$$\mathcal{C}_{HH} = \mathbb{E}_H[e_H \otimes e_H] = \left(\pi_h \delta(h, h') \right)_{h,h' \in [k]}, \text{ and}$$

$$\mathcal{C}_{HHH} = \mathbb{E}_H[e_H \otimes e_H \otimes e_H] = \left(\pi_h \, \delta(h, h') \, \delta(h', h'') \right)_{h,h',h'' \in [k]}$$

are diagonal tensors. Making use of $\mathcal{C}_{HH}$ and $\mathcal{C}_{HHH}$, and the factorization of the distributions $\mathbb{P}(X_1, X_2)$ and $\mathbb{P}(X_1, X_2, X_3)$, we obtain the factorization of the embedding of $\mathbb{P}(X_1, X_2)$ (second order embedding)

$$\begin{aligned} \mathcal{C}_{X_1 X_2} &= \sum\nolimits_{h \in [k]} \left(\mu_{X_1|h} \otimes \mu_{X_2|h} \right) \mathbb{P}(h) \\ &= \sum\nolimits_{h \in [k]} \left(\mathcal{C}_{X|H} e_h \right) \otimes \left(\mathcal{C}_{X|H} e_h \right) \mathbb{P}(h) \\ &= \mathcal{C}_{X|H} \left(\sum\nolimits_{h \in [k]} e_h \otimes e_h \, \mathbb{P}(h) \right) \mathcal{C}_{X|H}^\top \\ &= \mathcal{C}_{X|H} \, \mathcal{C}_{HH} \, \mathcal{C}_{X|H}^\top, \end{aligned} \tag{8}$$

and that of $\mathbb{P}(X_1, X_2, X_3)$ (third order embedding)

$$\mathcal{C}_{X_1 X_2 X_3} = \mathcal{C}_{HHH} \times_1 \mathcal{C}_{X|H} \times_2 \mathcal{C}_{X|H} \times_3 \mathcal{C}_{X|H}. \tag{9}$$

4.3. Identifiability of Parameters

We note that $\mathcal{C}_{X|H} = \left(\mu_{X|h=1}, \, \mu_{X|h=2}, \, \ldots, \, \mu_{X|h=k} \right)$, and the kernel embeddings for $\mathcal{C}_{X_1 X_2}$ and $\mathcal{C}_{X_1 X_2 X_3}$ can be alternatively written as

$$\mathcal{C}_{X_1 X_2} = \sum\nolimits_{h \in [k]} \pi_h \cdot \mu_{X|h} \otimes \mu_{X|h}, \tag{10}$$

$$\mathcal{C}_{X_1 X_2 X_3} = \sum\nolimits_{h \in [k]} \pi_h \cdot \mu_{X|h} \otimes \mu_{X|h} \otimes \mu_{X|h}. \tag{11}$$

Allman et al. (2009) showed that, under mild conditions, a finite mixture of nonparametric product distributions is identifiable. The multi-view latent variable model in (10) and (11) has the same form as a finite mixture of nonparametric product distribution, and therefore we can adapt Allman's results to the current setting.

Proposition 1 (Identifiability) *Let $\mathbb{P}(X_1, X_2, X_3)$ be a multi-view latent variable model, such that the conditional distributions $\{\mathbb{P}(X|h)\}_{h \in [k]}$ are linearly independent. Then, the set of parameters $\left\{\pi_h, \mu_{X|h}\right\}_{h \in [k]}$ are identifiable from $\mathcal{C}_{X_1 X_2 X_3}$, up to label swapping of the hidden variable H.*

Example 1. The probability vector of a discrete variable $X \in [n]$, and the joint probability table of two discrete variables $X_1 \in [n]$ and $X_2 \in [n]$, are both kernel embeddings. To see this, let the kernel be the Kronecker delta kernel $\kappa(x, x') = \delta(x, x')$ whose feature map $\phi(x)$ is the standard basis of e_x in $\mathbb{R}^n$. The x-th dimension of e_x is 1 and 0 otherwise. Then

$$\mu_X = \left(\mathbb{P}(x = 1) \quad \ldots \quad \mathbb{P}(x = n) \right)^\top,$$

$$\mathcal{C}_{X_1 X_2} = \left(\mathbb{P}(x_1 = s, x_2 = t) \right)_{s,t \in [n]}.$$

We require that the conditional probability table $\{P(X|h)\}_{h \in [k]}$ to have full column rank for identifiability in this case.

Example 2. Suppose we have a k-component mixture of one dimensional spherical Gaussian distributions. The Gaussian components have identical covariance σ^2, but their mean values are distinct. Note that this model is not identifiable under the framework of Hsu & Kakade (2013) since the mean values are just scalars and therefore, rank deficient. However, if we embed the density functions using universal kernels such as Gaussian RBF kernel, it can be shown that the mixture model becomes identifiable. This is because we are working with the entire density function which are linearly independent from each other in this case. Thus, the non-parametric framework allows us to incorporate a wider range of latent variable models.

Finally, we remark that the identifiability result in Proposition 1 can be extended to cases where the conditional distributions do not satisfy linear independence, *i.e.*, they are overcomplete, e.g. (Kruskal, 1977; De Lathauwer et al., 2007; Anandkumar et al., 2013b). However, in general, it is not tractable to learn such overcomplete models and we do not consider them here.

5. Kernel Algorithm

We first design a kernel algorithm to recover the parameters, $\left\{\pi_h, \mu_{X|h}\right\}_{h \in [k]}$, of the multi-view latent variable model based on $\mathcal{C}_{X_1 X_2}$ and $\mathcal{C}_{X_1 X_2 X_3}$. This can be easily extended to the sample versions and this is discussed in Section 5.2. Again for simplicity of exposition, the algorithm is explained for symmetric view case. The more general version is presented in Appendix 9.

5.1. Population Case

We first derive the algorithm for the population case as if we could access the true operator $\mathcal{C}_{X_1 X_2}$ and $\mathcal{C}_{X_1 X_2 X_3}$. Its finite sample counterpart will be presented in the next section. The algorithm can be thought of as a kernel generalization of the algorithm in Anandkumar et al. (2013a) using embedding representations.

Step 1. We perform eigen-decomposition of $\mathcal{C}_{X_1 X_2}$,

$$\mathcal{C}_{X_1 X_2} = \sum\nolimits_{i=1}^{\infty} \sigma_i \cdot u_i \otimes u_i$$

where the eigen-values are ordered in non-decreasing manner. According to the factorization in Eq. (8), $\mathcal{C}_{X_1 X_2}$ has rank k. Let the leading eigenvectors corresponding to the largest k eigen-value be $\mathcal{U}_k := (u_1, u_2, \ldots, u_k)$, and the eigen-value matrix be $S_k := \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_k)$. We define the whitening operator $\mathcal{W} := \mathcal{U}_k S_k^{-1/2}$ which satisfies

$$\mathcal{W}^\top \mathcal{C}_{X_1 X_2} \mathcal{W} = (\mathcal{W}^\top \mathcal{C}_{X|H} \mathcal{C}_{HH}^{1/2})(\mathcal{C}_{HH}^{1/2} \mathcal{C}_{X|H}^\top \mathcal{W}) = I,$$

and $M := \mathcal{W}^\top \mathcal{C}_{X|H} \mathcal{C}_{HH}^{1/2}$ is an orthogonal matrix.

Step 2. We apply the whiten operator to the 3rd order kernel embedding $\mathcal{C}_{X_1X_2X_3}$

$$\mathcal{T} := \mathcal{C}_{X_1X_2X_3} \times_1 (\mathcal{W}^\top) \times_2 (\mathcal{W}^\top) \times_3 (\mathcal{W}^\top).$$

According to the factorization in Eq. (9), $\mathcal{T} = \mathcal{C}_{HHH}^{-1/2} \times_1 M \times_2 M \times_3 M$, which is a tensor with orthogonal factors. Essentially, each column v_i of M is an eigenvector of $\mathcal{T}$.

Step 3. We use tensor power method to find the leading k eigenvectors M for $\mathcal{T}$ (Anandkumar et al., 2012a). The corresponding k eigenvalues $\lambda = (\lambda_1, \ldots, \lambda_k)^\top$ will then be equal to $(\mathbb{P}(h=1)^{-1/2}, \ldots, \mathbb{P}(h=k)^{-1/2})$. The tensor power method is provided in the Appendix in Algorithm 2 for completeness.

Step 4. We recover the conditional embedding operator by undoing the whitening step

$$\mathcal{C}_{X|H} = (\mathcal{W}^\top)^\dagger M \operatorname{diag}(\lambda).$$

5.2. Finite Sample Case

Given m observation $\mathcal{D}_{X_1X_2X_3} = \{(x_1^i, x_2^i, x_3^i)\}_{i\in[m]}$ drawn *i.i.d.* from a multi-view latent variable model $\mathbb{P}(X_1, X_2, X_3)$, we now design a kernel algorithm to estimate the latent parameters from data. Although the empirical kernel embeddings can be infinite dimensional, we can carry out the decomposition using just the kernel matrices. We denote the implicit feature matrix by

$$\Phi := (\phi(x_1^1), \ldots, \phi(x_1^m), \phi(x_2^1), \ldots, \phi(x_2^m)),$$
$$\Psi := (\phi(x_2^1), \ldots, \phi(x_2^m), \phi(x_1^1), \ldots, \phi(x_1^m)),$$

and the corresponding kernel matrix by $K = \Phi^\top\Phi$ and $L = \Psi^\top\Psi$ respectively. And we denote $K_{:x} := \Phi^\top\phi(x)$ as a column vector containing the kernel between x and data points in Φ. For three vectors ξ_1, ξ_2 and ξ_3, denote the symmetric tensor obtained from their outer product

$$\otimes[\xi_1, \xi_2, \xi_3] := \xi_1\otimes\xi_2\otimes\xi_3 + \xi_3\otimes\xi_1\otimes\xi_2 + \xi_2\otimes\xi_3\otimes\xi_1.$$

Then the steps in the population case can be mapped one-by-one into kernel operations.

Step 1. We perform a kernel eigenvalue decomposition of the empirical 2nd order embedding

$$\widehat{\mathcal{C}}_{X_1X_2} := \frac{1}{2m}\sum_{i=1}^m \left(\phi(x_1^i)\otimes\phi(x_2^i) + \phi(x_2^i)\otimes\phi(x_1^i)\right),$$

which can be expressed succinctly as $\widehat{\mathcal{C}}_{X_1X_2} = \frac{1}{2m}\Phi\Psi^\top$. Its leading k eigenvectors $\widehat{\mathcal{U}}_k = (\widehat{u}_1, \ldots, \widehat{u}_k)$ lie in the span of the column of Φ, *i.e.*, $\widehat{\mathcal{U}}_k = \Phi(\beta_1, \ldots, \beta_k)$ with $\beta \in \mathbb{R}^{2m}$. Then we can transform the eigen-value decomposition problem for an infinite dimensional matrix to a problem involving finite dimensional kernel matrices,

$$\widehat{\mathcal{C}}_{X_1X_2}\widehat{\mathcal{C}}_{X_1X_2}^\top u = \widehat{\sigma}^2 u \;\Rightarrow\; \frac{1}{4m^2}\Phi\Psi^\top\Psi\Phi^\top\Phi\beta = \widehat{\sigma}^2\Phi\beta$$
$$\Rightarrow\; \frac{1}{4m^2}KLK\beta = \widehat{\sigma}^2 K\beta.$$

Algorithm 1 Kernel Spectral Algorithm

In: Kernel matrices K and L, and desired rank k
Out: A vector $\widehat{\pi} \in \mathbb{R}^k$ and a matrix $A \in \mathbb{R}^{2m\times k}$

1: Cholesky decomposition: $K = R^\top R$
2: Eigen-decomposition: $\frac{1}{4m^2}RLR^\top\widetilde{\beta} = \widehat{\sigma}^2\widetilde{\beta}$
3: Use k leading eigenvalues: $\widehat{S}_k = \operatorname{diag}(\widehat{\sigma}_1, \ldots, \widehat{\sigma}_k)$
4: Use k leading eigenvectors $(\widetilde{\beta}_1, \ldots, \widetilde{\beta}_k)$ to compute: $(\beta_1, \ldots, \beta_k) = R^\dagger(\widetilde{\beta}_1, \ldots, \widetilde{\beta}_k)$
5: Form tensor: $\widehat{\mathcal{T}} = \frac{1}{3m}\sum_{i=1}^m \otimes\left[\xi(x_1^i), \xi(x_2^i), \xi(x_3^i)\right]$ where $\xi(x_1^i) = \widehat{S}_k^{-1/2}(\beta_1, \ldots, \beta_k)^\top K_{:x_1^i}$
6: Power method: eigenvectors $\widehat{M} := (\widehat{v}_1, \ldots, \widehat{v}_k)$, and the eigenvalues $\widehat{\lambda} := (\widehat{\lambda}_1, \ldots, \widehat{\lambda}_k)^\top$ of $\widehat{\mathcal{T}}$
7: $A = (\beta_1, \ldots, \beta_k)\widehat{S}_k^{1/2}\widehat{M}\operatorname{diag}(\widehat{\lambda})$
8: $\widehat{\pi} = (\widehat{\lambda}_1^{-2}, \ldots, \widehat{\lambda}_k^{-2})^\top$

Let the Cholesky decomposition of K be $R^\top R$. Then by redefining $\beta = R\beta$, and solving an eigenvalue problem

$$\frac{1}{4m^2}RLR^\top\widetilde{\beta} = \widehat{\sigma}^2\widetilde{\beta}, \text{ and obtain } \beta = R^\dagger\widetilde{\beta}. \tag{12}$$

The resulting eigenvectors satisfy $u_i^\top u_{i'} = \beta_i^\top\Phi^\top\Phi\beta_{i'} = \beta_i^\top K\beta_{i'} = \widetilde{\beta}_i^\top\widetilde{\beta}_{i'} = \delta_{ii'}$.

Step 2. We whiten the empirical 3rd order embedding

$$\widehat{\mathcal{C}}_{X_1X_2X_3} := \frac{1}{3m}\sum_{i=1}^m \otimes\left[\phi(x_1^i), \phi(x_2^i), \phi(x_3^i)\right]$$

using $\widehat{\mathcal{W}} := \widehat{\mathcal{U}}_k\widehat{S}_k^{-1/2}$, and obtain

$$\widehat{\mathcal{T}} := \frac{1}{3m}\sum_{i=1}^m \otimes\left[\xi(x_1^i), \xi(x_2^i), \xi(x_3^i)\right]$$

where $\xi(x_1^i) := \widehat{S}_k^{-1/2}(\beta_1, \ldots, \beta_k)^\top K_{:x_1^i} \in \mathbb{R}^k$.

Step 3. We run tensor power method (Anandkumar et al., 2012a) on the finite dimension tensor $\widehat{\mathcal{T}}$ to obtain its leading k eigenvectors $\widehat{M} := (\widehat{v}_1, \ldots, \widehat{v}_k)$ and the corresponding eigenvalues $\widehat{\lambda} := (\widehat{\lambda}_1, \ldots, \widehat{\lambda}_k)^\top$.

Step 4. The estimates of the conditional embeddings are

$$\widehat{\mathcal{C}}_{X|H} = \Phi(\beta_1, \ldots, \beta_k)\widehat{S}_k^{1/2}\widehat{M}\operatorname{diag}(\widehat{\lambda}).$$

The overall kernel algorithm is summarized in Algorithm 1.

6. Sample Complexity

Let $\rho := \sup_{x\in\mathcal{X}}\kappa(x,x)$, $\|\cdot\|$ be the Hilbert-Schmidt norm, $\pi_{\min} := \min_{i\in[k]}\pi_i$ and $\sigma_k(\mathcal{C}_{X_1X_2})$ be the k-th largest singular value of $\mathcal{C}_{X_1X_2}$. In the following, we provide sample complexity bounds for the estimated conditional embedding $\mu_{X|h}$ and the corresponding prior distribution π (the proof is in Appendix 11).

Theorem 2 *Pick any $\delta \in (0,1)$. When the number of samples m satisfies*

$$m > \frac{\theta\rho^2\log\frac{2}{\delta}}{\sigma_k^2(\mathcal{C}_{X_1X_2})}, \quad \theta := \max\left(\frac{C_3k^2\rho}{\sigma_k(\mathcal{C}_{X_1X_2})}, \frac{C_4k^{2/3}}{\pi_{\min}^{1/3}}\right),$$

for some constants $C_3, C_4 > 0$, and the number of iterations N and the number of random initialization vectors L (drawn uniformly on the sphere $\mathcal{S}^{k-1}$) satisfy

$$N \geq C_2 \cdot \left(\log(k) + \log\log\Big(\frac{1}{\sqrt{\pi_{\min}}\epsilon_{\mathcal{T}}}\Big)\right),$$

for constant $C_2 > 0$ and $L = \mathrm{poly}(k)\log(1/\delta)$, the robust power method in (Anandkumar et al., 2012a) yields eigenpairs $(\widehat{\lambda}_i, \widehat{v}_i)$ such that there exists a permutation η, with probability $1 - 4\delta$, we have

$$\|\pi_j^{-1/2}\mu_{X|h=j} - (\beta_1, \ldots, \beta_k)\widehat{S}_k^{1/2}\widehat{v}_{\eta(j)}\|_{\mathcal{F}} \leq 8\epsilon_{\mathcal{T}} \cdot \pi_j^{-1/2},$$

$$|\pi_j^{-1/2} - \widehat{\lambda}_{\eta(j)}| \leq 5\epsilon_{\mathcal{T}}, \quad \forall j \in [k],$$

and $\|\mathcal{T} - \sum_{j=1}^k \hat{\lambda}_j \hat{\phi}_j^{\otimes 3}\| \leq 55\epsilon_{\mathcal{T}}$ where $\epsilon_{\mathcal{T}} := \|\widehat{\mathcal{T}} - \mathcal{T}\|$ is the tensor perturbation bound

$$\epsilon_{\mathcal{T}} \leq \frac{8\rho^{1.5}\sqrt{\log\frac{2}{\delta}}}{\sqrt{m}\,\sigma_k^{1.5}(\mathcal{C}_{X_1X_2})} + \frac{512\sqrt{2}\rho^3\left(\log\frac{2}{\delta}\right)^{1.5}}{m^{1.5}\,\sigma_k^3(\mathcal{C}_{X_1X_2})\sqrt{\pi_{\min}}}$$

Proof Sketch: Our proof is different from those in Anandkumar et al. (2012a) which only analyze the perturbation of the tensor decomposition. Our proof further takes into account the error introduced by the approximate whitening step, and its effects to the tensor decomposition. ■

Remark 1: We note that the sample complexity is $\mathrm{poly}(k, \rho, 1/\pi_{\min}, 1/\sigma_k(\mathcal{C}_{X_1X_2}))$ of a low order, and in particular, it is $O(k^2)$, when the other parameters are fixed. For the special case of discrete measurements, where the kernel $\kappa(x, x') = \delta(x, x')$, we have $\rho = 1$. Note that the sample complexity depends in this case only on the number of components k and not on the dimensionality of the observed state space.

Remark 2: Theorem 2 also gives us an error bound for estimating the integral of a function $f \in \mathcal{F}$ with respect to a mixture component in unsupervised fashion. Under the conditions specified in the theorem, we have

$$\left|\int_{\mathcal{X}} f(x)\, d\mathbb{P}(x|h) - \left\langle f, \widehat{\mu}_{X|h}\right\rangle_{\mathcal{F}}\right| \leq \|f\|_{\mathcal{F}}\left\|\mu_{X|h} - \widehat{\mu}_{X|h}\right\|_{\mathcal{F}} = O\left(\tfrac{1}{\sqrt{m}}\right)$$

assuming $\|f\|_{\mathcal{F}}$ is bounded and $\rho/\sigma_k = O(1)$. We are not aware of any other result of the similar type in this unsupervised setting.

Remark 3: For $x \in \mathbb{R}^d$ and the normalized Gaussian RBF kernel in (2), the recovered conditional embedding $\widehat{\mu}_{X|h}$ can be used to estimate the conditional density, $p(x|h) \approx \widehat{p}(x|h) := \left\langle \phi(x), \widehat{\mu}_{X|h}\right\rangle_{\mathcal{F}}$. In this case, the error can be decomposed into two terms

$$|p(x|h) - \widehat{p}(x|h)| \leq \underbrace{|p(x|h) - \widetilde{p}(x|h)|}_{O(s^2)\text{ bias as in (7)}} + \underbrace{|\widetilde{p}(x|h) - \widehat{p}(x|h)|}_{\text{estimation error}}$$

where s is kernel bandwidth and $\widetilde{p}$ is the density convolved with the kernel function. The estimation error is bounded by $|\widetilde{p}(x|h) - \widehat{p}(x|h)| \leq \|\phi(x)\|_{\mathcal{F}}\|\mu_{X|h} - \widehat{\mu}_{X|h}\|_{\mathcal{F}} = O(\rho^{1/2} \cdot m^{-1/2}) = O(s^{-d/2}m^{-1/2})$ assuming $\rho/\sigma_k = O(1)$ and using $\rho = O(s^{-d})$. Under the conditions specified in Theorem 2, we combine the analysis for the two sources of errors, and obtain the bound

$$|p(x|h) - \widehat{p}(x|h)| = O(s^2 + s^{-d/2}m^{-1/2})$$

Then we have $|p(x|h) - \widehat{p}(x|h)| = O(m^{-2/(4+d)})$ if we balance the two terms by setting $s = O(m^{-1/(4+d)})$. We are not aware of any other result of the similar type in this unsupervised setting.

7. Discussion

Our algorithm and theoretical results can also be generalized to the settings of latent variable models with Dirichlet priors and nonparametric independent component analysis (ICA) as in Anandkumar et al. (2012a). In the first setting, a Dirichlet prior is placed on the mixing weights π of the multi-view latent variables, $\mathbb{P}(\pi) = \frac{\Gamma(\theta_0)}{\prod_{i\in[k]}\Gamma(\theta_i)}\prod_{i\in[k]}\pi_i^{\theta_i - 1}$ where $\theta_0 = \sum_{i\in[k]}\theta_i$ with $\theta_i > 0$, and $\Gamma(\cdot)$ is the Gamma function. In this case, we only need to modify the second and third order kernel embedding $\mathcal{C}_{X_1X_2}$ and $\mathcal{T}$ respectively, and then Algorithm 1 applies. In the nonparametric ICA setting, the feature map $\phi(X)$ of an observed variable X is assumed to be generated from a latent vector $H \in \mathbb{R}^k$ with independent coordinates via an operator $\mathcal{A} : \mathbb{R}^k \mapsto \mathcal{F}$, $\phi(X) := \mathcal{A}H + Z$, where Z is a zero mean random vector independent of H. In this case, we need to start with a modified 4-th order kernel embedding, and then reduce to a multi-view problem and estimate $\mathcal{A}$ via Algorithm 1.

8. Experiments

Methods. We compared our kernel spectral algorithm with four alternatives

1. The EM algorithm for mixture of Gaussians. The EM algorithm is not guaranteed to find the global solution in each trial. Thus we randomly initialize it 10 times.
2. The EM-like algorithm for mixture of nonparametric densities (Benaglia et al., 2009). We initialize the algorithm with k-means as Benaglia et al. (2009).
3. The spectral algorithm for mixture of spherical Gaussians (Hsu & Kakade, 2013). Their assumption is restrictive: the centers of the Gaussian need to span a k-dimension subspace, thus it is not applicable for rank deficiency case where $k \geq l$.
4. A discretization based spectral algorithm (Kasahara & Shimotsu, 2010). This algorithm approximates the joint distribution of the observed variables with histogram and then applies the spectral algorithm to recover the discretized conditional density.

Both our method and the (Benaglia et al., 2009) have a hyper-parameter, kernel bandwidth, which we selected for each view separately using cross-validation.

8.1. Synthetic Data

We generated three-dimensional synthetic data from various mixture models. The variables corresponding to the dimensions are independent given the latent component indicator. More specifically, we explored two settings (1) Gaussian conditional densities with different variances; (2) Mixture of Gaussian and shifted Gamma conditional densities. The shifted Gamma distribution has density

$$p(x|h) = \frac{(x-\mu)^{(d-1)}e^{-x/\theta}}{\theta^d \Gamma(d)},\ x \geq \mu$$

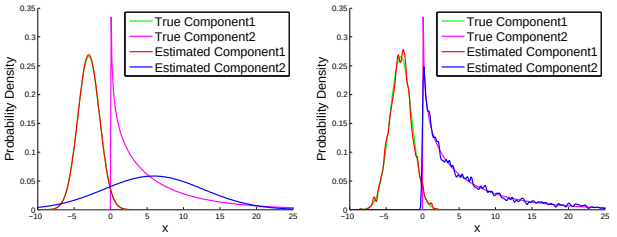
where we chose the shape parameter $d \leq 1$ such that density is very skewed. Furthermore, we chose the mean and variance parameters of the Gaussian/Gamma density such that component pair-wise overlap is relatively small according to the Fisher ratio $\frac{(\mu_1-\mu_2)^2}{\sigma_1^2+\sigma_2^2}$.

We also varied the number of samples m for the observed variables X_1, X_2 and X_3 from 50 to $10,000$, and experimented with $k = 2, 3, 4$ or 8 mixture components. The mixture proportion for the h-th component is set to be $\pi_h = \frac{2h}{\kappa(k+1)},\ \forall h \in [k]$ (unbalanced). It is worth noting that as k becomes larger, it is more difficult to recover parameters. This is because only a small number of data will be generated for the first several clusters. For every n, k in each setting, we randomly generated 10 sets of samples and reported the average results. **We note that the values for the latent variables are not given to the algorithms, and hence this is an unsupervised setting to recover the conditional density $p(x|h)$ and the ratio $p(h)$.**

Error measure. We measured the performance of algorithms by the following weighted ℓ_2 norm difference $MSE := \sum_{h=1}^{k} \pi_h \sqrt{\sum_{j=1}^{m'} (p(x^j|h) - \widehat{p}(x^j|h))^2}$, where $\{x^j\}_{j\in[m]}$ is a set of uniformly-spaced test points.

Results. We first illustrated the actual recovered conditional densities of our method and EM-GMM in Figure 2 as a concrete example. The kernel spectral algorithm recovers nicely both the Gaussian and Gamma components, while the EM-GMM fails to fit the Gamma component.

More quantitative results are plotted in Figure 3. It is clear that the kernel spectral method converges rapidly with the data increment in all experiment settings. In the mixture of Gaussians setting, the EM algorithm is best since the model is correctly specified. The spectral algorithm for spherical Gaussians does not perform well since the assumption of the method is too restricted. The performance of our kernel method converges to that of the EM algorithm. In the mixture of Gaussian and Gamma setting, our kernel spectral algorithm achieves superior results compared to other



(a) EM Gaussians Mixture (b) Kernel Spectral

Figure 2. Kernel spectral algorithm is able to adapt to the shape of the mixture components, while EM algorithm for mixture of Gaussians misfits the Gamma distribution.

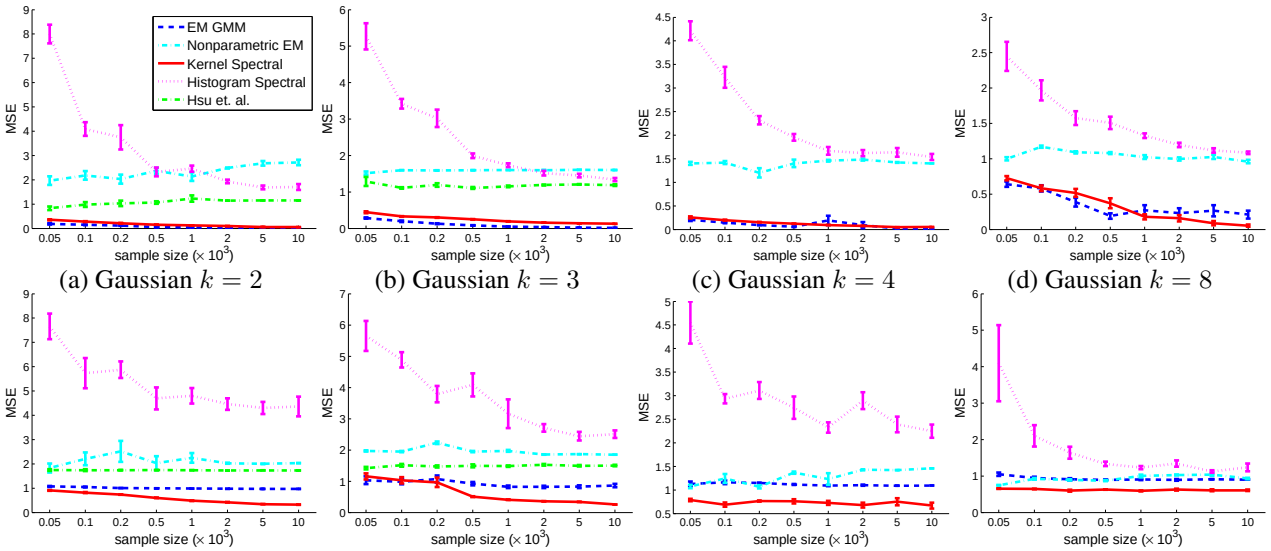
algorithms. These results demonstrate that our algorithm is able to automatically adapt to the shape of the density.

It is worth noting that both the discretized spectral algorithm and nonparametric EM-like algorithm did not perform as well. In the discretized spectral method, the joint distribution is estimated by histogram. It is well-known that the histogram estimation suffers from poor performance even for 3 dimensional data. In the nonparametric EM-like algorithm, besides the issue of local minima, its performance also highly depends on the initialization. And the flexibility of nonparametric densities without regularization makes the issue of overfitting quite severe, often leading to a single component in the algorithm.

We also note that the our method outperforms the EM-GMM more as the number of components increases. This is the key advantage of our method in that it has favorable performance in higher dimensions, which agrees with the theoretical result in Theorem 2 that the sample complexity depends only quadratically in the number of components, when other parameters are held fixed.

8.2. Flow Cytometry Data

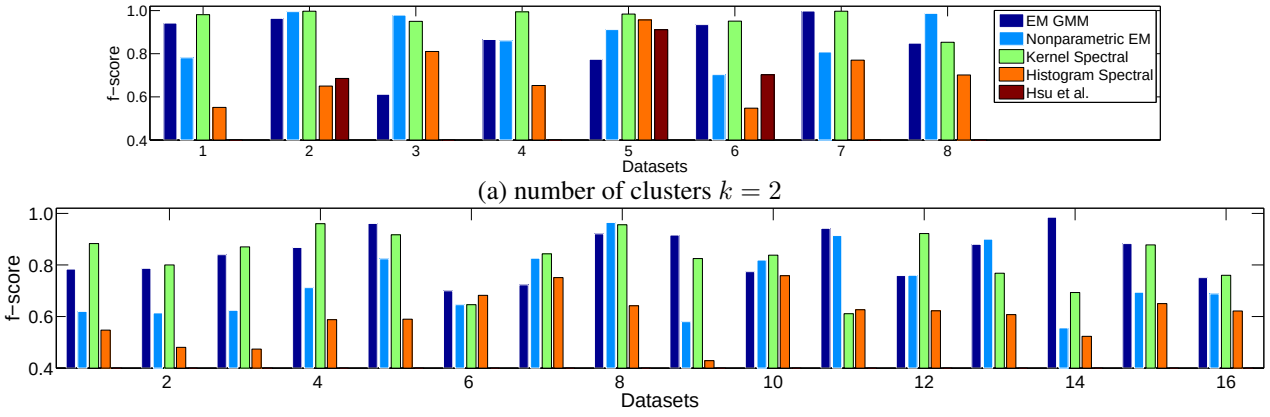
Flow cytometry (FCM) data are multivariate measurements from flow cytometers that record light scatter and fluorescence emission properties of hundreds of thousands of individual cells. They are important to the studying of the cell structures of normal and abnormal cells and the diagnosis of human diseases. Aghaeepour et al. (2013) introduced the FlowCAP-challenge whose main task is grouping the flow cytometry data automatically. Clustering on the FCM data is a difficult task because the distribution of the data is non-Gaussian and heavily skewed. We use the DLBCL Lymphoma dataset collection from (Aghaeepour et al., 2013) to compare our kernel algorithm with the four alternatives. This collection contains 24 datasets with two or three clusters, and each dataset consists of tens of thousands of cell measurements in 5 dimensions. Each dataset is a separate clustering task, so we fit a multi-view model to each dataset separately and use the maximum-a-posteriori assignment to obtain the cluster labels. All the cell measurements have been manually labeled, therefore we can evaluate the clustering performance using f-score (Aghaeepour et al., 2013).



(a) Gaussian $k = 2$ (b) Gaussian $k = 3$ (c) Gaussian $k = 4$ (d) Gaussian $k = 8$

(e) Gaussian/Gamma $k = 2$ (f) Gaussian/Gamma $k = 3$ (g) Gaussian/Gamma $k = 4$ (h) Gaussian/Gamma $k = 8$

Figure 3. (a)-(d) Mixture of Gaussian distributions with $k = 2, 3, 4, 8$ components. (e)-(h) Mixture of Gaussian/Gamma distribution with $k = 2, 3, 4, 8$. For the former case, the performances of kernel spectral algorithm converge to those of EM algorithm for mixture of Gaussian model. For the latter case, the performances of kernel spectral algorithm are consistently much better than EM algorithm for mixture of Gaussian model. Spherical Gaussian spectral algorithm does not work for $k = 4, 8$ since $k > l(= 3)$ causes rank deficiency.



(a) number of clusters $k = 2$

(b) number of clusters $k = 3$

Figure 4. Clustering results on the datasets from the DLBCL flow cytometry data. The results for spherical Gaussian spectral algorithm (Hsu et al.) are not plotted for datasets on which it has rank deficiency problem. The datasets are ordered by increasing sample size.

We split the 5 dimensions into three views: dimension 1 and 2 as the first view, 3 and 4 the second, and 5 the third view based on correlation between views, since we would like the views to satisfy the conditional independence assumptions to ensure good performance for the kernel spectral method. For each dataset, we select the best kernel bandwidth by 5-fold cross validation using log-likelihood. Figure 4 presents the results sorted by the number of clusters. Since the data are collapsed in most cases, the centers cannot span a subspace with enough rank. Thus, the method in (Hsu & Kakade, 2013) is not applicable. However, our method (kernel spectral) outperforms EM-GMM as well as the other algorithms in a majority of datasets. There are also datasets where kernel spectral algorithm has a large gap in performance compared to GMM. These are the datasets where the multi-view assumptions are heavily violated. For example, in some datasets, the correlation coefficient between dimensions 3 and 5 is as high as 0.927 given a particular cluster label, suggesting strong correlation between the two views. Obtaining improved and robust performance in these datasets will be a subject of our future study where we plan to develop even more robust kernel spectral algorithms.

Acknowledgement

L.S. is supported in part by NSF IIS1116886, NSF/NIH BIGDATA 1R01GM108341, NSF CAREER IIS1350983 and Raytheon Faculty Fellowship. A.A. is supported in part by Microsoft Faculty Fellowship, NSF CAREER CCF1254106, NSF CCF1219234, NSF BIGDATA IIS1251267 and ARO YIP Award W911NF-13-1-0084.

References

Aghaeepour, Nima, Finak, Greg, Consortium, The FlowCAP, Consortium, The DREAM, Hoos, Holger, Mosmann, Tim R, Brinkman, Ryan, Gottardo, Raphael, and Scheuermann, Richard H. Critical assessment of automated flow cytometry data analysis techniques. *Nature Methods*, 10(3):228–238, 2013.

Allman, Elizabeth, Matias, Catherine, and Rhodes, John. Identifiability of parameters in latent structure models with many observed variables. *The Annals of Statistics*, 37(6A):3099–3132, 2009.

Anandkumar, A., Ge, R., Hsu, D., Kakade, S. M., and Telgarsky, M. Tensor Methods for Learning Latent Variable Models. *Available at arXiv:1210.7559*, Oct. 2012a.

Anandkumar, A., Ge, R., Hsu, D., and Kakade, S. M. A Tensor Spectral Approach to Learning Mixed Membership Community Models. *ArXiv 1302.2684*, Feb. 2013a.

Anandkumar, A., Hsu, D., Janzamin, M., and Kakade, S. M. When are Overcomplete Topic Models Identifiable? Uniqueness of Tensor Tucker Decompositions with Structured Sparsity. *ArXiv 1308.2853*, Aug. 2013b.

Anandkumar, Animashree, Foster, Dean P., Hsu, Daniel, Kakade, Sham M., and Liu, Yi-Kai. A spectral algorithm for latent dirichlet allocation. *Available at arXiv:1204.6703*, 2012b.

Benaglia, Tatiana, Chauveau, Didier, and Hunter, David R. An em-like algorithm for semi-and nonparametric estimation in multivariate mixtures. *Journal of Computational and Graphical Statistics*, 18(2):505–526, 2009.

De Lathauwer, L., Castaing, J., and Cardoso, J.-F. Fourth-order cumulant-based blind identification of underdetermined mixtures. *IEEE Tran. on Signal Processing*, 55: 2965–2973, June 2007.

Dempster, A. P., Laird, N. M., and Rubin, D. B. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society B*, 39(1):1–22, 1977.

Foster, D.P., Rodu, J., and Ungar, L.H. Spectral dimensionality reduction for hmms. *Arxiv preprint arXiv:1203.6130*, 2012.

Hsu, D., Kakade, S., and Zhang, T. A spectral algorithm for learning hidden markov models. In *Proc. Annual Conf. Computational Learning Theory*, 2009.

Hsu, Daniel and Kakade, Sham M. Learning mixtures of spherical gaussians: moment methods and spectral decompositions. In *Proceedings of the 4th conference on Innovations in Theoretical Computer Science*, ITCS '13, pp. 11–20.

Kasahara, Hiroyuki and Shimotsu, Katsumi. Nonparametric identification of multivariate mixtures. *Journal of the Royal Statistical Society - Series B*, 2010.

Kolda, Tamara G. and Bader, Brett W. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009.

Kruskal, J.B. Three-way arrays: Rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics. *Linear algebra and its applications*, 18(2):95–138, 1977.

Parikh, A., Song, L., and Xing, E. P. A spectral algorithm for latent tree graphical models. In *Proceedings of the International Conference on Machine Learning*, 2011.

Rosasco, L., Belkin, M., and Vito, E.D. On learning with integral operators. *Journal of Machine Learning Research*, 11:905–934, 2010.

Schölkopf, B., Tsuda, K., and Vert, J.-P. *Kernel Methods in Computational Biology*. MIT Press, Cambridge, MA, 2004.

Sgouritsa, Eleni, Janzing, Dominik, Peters, Jonas, and Schölkopf, Bernhard. Identifying finite mixtures of nonparametric product distributions and causal inference of confounders. In *Conference on Uncertainty on Artificial Intelligence (UAI)*, 2013.

Smola, A. J., Gretton, A., Song, L., and Schölkopf, B. A Hilbert space embedding for distributions. In *Proceedings of the International Conference on Algorithmic Learning Theory*, volume 4754, pp. 13–31. Springer, 2007.

Song, L. and Dai, B. Robust low rank kernel embedding of multivariate distributions. In *Neural Information Processing Systems (NIPS)*, 2013.

Song, L., Boots, B., Siddiqi, S., Gordon, G., and Smola, A. J. Hilbert space embeddings of hidden markov models. In *International Conference on Machine Learning*, 2010.

Song, L., Parikh, A., and Xing, E.P. Kernel embeddings of latent tree graphical models. In *Advances in Neural Information Processing Systems*, volume 25, 2011.

Sriperumbudur, B., Gretton, A., Fukumizu, K., Lanckriet, G., and Schölkopf, B. Injective Hilbert space embeddings of probability measures. In *Proc. Annual Conf. Computational Learning Theory*, pp. 111–122, 2008.

Wasserman, L. *All of Nonparametric Statistics*. Springer, 2006.

Appendix

9. Symmetrization

We presented the kernel algorithm for learning the multi-view latent variable model where the views have identical conditional distributions. In this section, we will extend it to the general case where the views are different. Without loss of generality, we will consider recover the operator $\mu_{X_3|h}$ for conditional distribution $\mathbb{P}(X_3|h)$. The same strategy applies to other views. The idea is to reduce the multi-view case to the identical-view case based on a method by (Anandkumar et al., 2012b).

Given the observations $\mathcal{D}_{X_1X_2X_3} = \{(x_1^i, x_2^i, x_3^i)\}_{i\in[m]}$ drawn *i.i.d.* from a multi-view latent variable model $\mathbb{P}(X_1, X_2, X_3)$, let the kernel matrix associated with X_1, X_2 and X_3 be K, L and G respectively and the corresponding feature map be ϕ, ψ and υ respectively. Furthermore, let the corresponding feature matrix be $\widetilde{\Phi} = (\phi(x_1^1), \ldots, \phi(x_1^m))$, $\widetilde{\Psi} = (\phi(x_2^1), \ldots, \phi(x_2^m))$ and $\widetilde{\Upsilon} = (\phi(x_3^1), \ldots, \phi(x_3^m))$. Then, we have the empirical estimation of the second/third-order embedding as

$$\widehat{\mathcal{C}}_{X_1X_2} = \frac{1}{m}\widetilde{\Phi}\widetilde{\Psi}^\top,\ \widehat{\mathcal{C}}_{X_3X_1} = \frac{1}{m}\widetilde{\Upsilon}\widetilde{\Phi}^\top,\ \widehat{\mathcal{C}}_{X_2X_3} = \frac{1}{m}\widetilde{\Psi}\widetilde{\Upsilon}^\top$$
$$\widehat{\mathcal{C}}_{X_1X_2X_3} := \frac{1}{m}\boldsymbol{I}_n \times_1 \widetilde{\Phi} \times_2 \widetilde{\Psi} \times_3 \widetilde{\Upsilon}$$

Find two arbitrary matrices $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{k\times\infty}$, so that $\boldsymbol{A}\widehat{\mathcal{C}}_{X_1X_2}\boldsymbol{B}^\top$ is invertible. Theoretically, we could randomly select k columns from Φ and Ψ and set $\boldsymbol{A} = \Phi_k^\top, \boldsymbol{B} = \Psi_k^\top$. In practial, the first k leading eigenvector directions of respect *RKHS* works better. Then, we have

$$\begin{aligned}
\widetilde{\mathcal{C}}_{X_1X_2} &= \frac{1}{m}\widetilde{\Phi}_k^\top\widetilde{\Phi}\widetilde{\Psi}^\top\widetilde{\Psi}_k = \frac{1}{m}K_{nk}^\top L_{nk} \\
\widetilde{\mathcal{C}}_{X_3X_1} &= \widehat{\mathcal{C}}_{X_3X_1}\widetilde{\Phi}_k = \frac{1}{m}\widetilde{\Upsilon}K_{nk} \\
\widetilde{\mathcal{C}}_{X_3X_2} &= \widehat{\mathcal{C}}_{X_3X_2}\widetilde{\Psi}_k = \frac{1}{m}\widetilde{\Upsilon}L_{nk} \\
\widetilde{\mathcal{C}}_{X_1X_2X_3} &= \widehat{\mathcal{C}}_{X_1X_2X_3} \times_1 \widetilde{\Phi}_k^\top \times_2 \widetilde{\Psi}_k^\top = \frac{1}{m}\boldsymbol{I}_n \times_1 K_{nk}^\top \times_2 L_{nk}^\top \times_3 \widetilde{\Upsilon}
\end{aligned}$$

Based on these matrices, we could reduce to a single view

$$\begin{aligned}
Pair_3 &= \widetilde{\mathcal{C}}_{X_3X_1}(\widetilde{\mathcal{C}}_{X_1X_2}^\top)^{-1}\widetilde{\mathcal{C}}_{X_3X_2} \\
&= \frac{1}{m}\widetilde{\Upsilon}K_{nk}(L_{nk}^\top K_{nk})^{-1}L_{nk}^\top\widetilde{\Upsilon}^\top = \frac{1}{m}\widetilde{\Upsilon}H\widetilde{\Upsilon}^\top
\end{aligned}$$

where $H = K_{nk}(\mathcal{L}_{nk}^\top K_{nk})^{-1}L_{nk}^\top$.

Assume the leading k eigenvectors ν_k lie in the span of the column of Υ, i.e., $\nu_k = \Upsilon\beta_k$ where $\beta_k \in \mathbb{R}^{m\times 1}$

$$\begin{aligned}
Pair_3\nu = \lambda\nu &\Rightarrow (Pair_3)^\top Pair_3\nu = \lambda^2\nu \\
&\Rightarrow \frac{1}{m^2}\widetilde{\Upsilon}H^\top\widetilde{\Upsilon}^\top\widetilde{\Upsilon}H\widetilde{\Upsilon}^\top\nu = \lambda^2\nu \\
&\Rightarrow \frac{1}{m^2}\widetilde{\Upsilon}H^\top GHG\boldsymbol{\beta} = \lambda^2\widetilde{\Upsilon}\boldsymbol{\beta} \\
&\Rightarrow \frac{1}{m^2}GH^\top GHG\beta = \lambda^2 G\beta
\end{aligned}$$

Then, we symmetrize and whiten the third-order embedding

$$Triple_3 = \frac{1}{m}\widetilde{\mathcal{C}}_{X_1X_2X_3} \times_1 [\widetilde{\mathcal{C}}_{X_3X_2}\widetilde{\mathcal{C}}_{X_1X_2}^{-1}] \times_2 [\widetilde{\mathcal{C}}_{X_3X_1}\widetilde{\mathcal{C}}_{X_2X_1}^{-1}] \tag{13}$$

Plug $\widetilde{\mathcal{C}}_{X_3X_2}\widetilde{\mathcal{C}}_{X_1X_2}^{-1} = \widetilde{\Upsilon}L_{nk}(K_{nk}^\top L_{nk})^{-1}$ and $\widetilde{\mathcal{C}}_{X_3X_1}\widetilde{\mathcal{C}}_{X_2X_1}^{-1} = \widetilde{\Upsilon}K_{nk}(L_{nk}^\top K_{nk})^{-1}$, we have

$$Triple_3 = \frac{1}{m}\boldsymbol{I}_n \times_1 \widetilde{\Upsilon} L_{nk}(K_{nk}^\top L_{nk})^{-1} K_{nk}^\top \\ \times_2 \widetilde{\Upsilon} K_{nk}(L_{nk}^\top K_{nk})^{-1} L_{nk}^\top \times_3 \Upsilon$$

We multiply each mode with $\Upsilon\beta\widehat{S}_k^{-\frac{1}{2}}$ to whitening the data and apply power method to decompose it

$$\begin{aligned}\widehat{\mathcal{T}} &= Triple_3 \times_1 \widehat{S}_k^{-\frac{1}{2}}\beta^\top\widetilde{\Upsilon}^\top \times_2 \widehat{S}_k^{-\frac{1}{2}}\beta^\top\widetilde{\Upsilon}^\top \times_3 \widehat{S}_k^{-\frac{1}{2}}\beta^\top\widetilde{\Upsilon}^\top \\ &= \frac{1}{m}\boldsymbol{I}_n \times_1 \widehat{S}_k^{-\frac{1}{2}}\beta^\top G\mathcal{L}_{nk}(K_{nk}^\top L_{nk})^{-1}K_{nk}^\top \times_2 \\ &\quad \widehat{S}_k^{-\frac{1}{2}}\beta^\top G K_{nk}(L_{nk}^\top K_{nk})^{-1}L_{nk}^\top \times_3 \widehat{S}_k^{-\frac{1}{2}}\beta^\top G\end{aligned}$$

10. Robust Tensor Power Method

We recap the robust tensor power method for finding the tensor eigen-pairs in Algorithm 2, analyzed in detail in (Anandkumar et al., 2013a) and (Anandkumar et al., 2012a). The method computes the eigenvectors of a tensor through deflation, using a set of initialization vectors. Here, we employ random initialization vectors. This can be replaced with better initialization vectors, in certain settings, e.g. in the community model, the neighborhood vectors provide better initialization and lead to stronger guarantees (Anandkumar et al., 2013a). Given the initialization vector, the method then runs a tensor power update, and runs for N iterations to obtain an eigenvector. The successive eigenvectors are obtained via deflation.

Algorithm 2 $\{\lambda, M\} \leftarrow$TensorEigen$(\mathcal{T}, \{v_i\}_{i\in[k]}, N)$

Input: Tensor $\mathcal{T} \in \mathbb{R}^{k\times k\times k}$, set of k initialization vectors $\{v_i\}_{i\in[k]}$, number of iterations N.

Output: the estimated eigenvalue/eigenvector pairs $\{\lambda, M\}$, where $\lambda = (\lambda_1, \ldots, \lambda_k)^\top$ is the vector of eigenvalues and $M = (v_1, \ldots, v_k)$ is the matrix of eigenvectors.

for $i = 1$ to k **do**
  for $\tau = 1$ to k **do**
    $\theta_0 \leftarrow v_\tau$.
    for $t = 1$ to N **do**
      $\widetilde{\mathcal{T}} \leftarrow \mathcal{T}$.
      for $j = 1$ to $i-1$ (when $i > 1$) **do**
        if $|\lambda_j \left\langle \theta_t^{(\tau)}, v_j \right\rangle| > \xi$ **then**
          $\widetilde{\mathcal{T}} \leftarrow \widetilde{\mathcal{T}} - \lambda_j \phi_j^{\otimes 3}$.
        end if
      end for
      Compute power iteration update $\theta_t^{(\tau)} := \frac{\widetilde{\mathcal{T}}(I, \theta_{t-1}^{(\tau)}, \theta_{t-1}^{(\tau)})}{\|\widetilde{\mathcal{T}}(I, \theta_{t-1}^{(\tau)}, \theta_{t-1}^{(\tau)})\|}$
    end for
  end for
  Let $\tau^* := \arg\max_{\tau\in L}\{\widetilde{\mathcal{T}}(\theta_N^{(\tau)}, \theta_N^{(\tau)}, \theta_N^{(\tau)})\}$.
  Do N power iteration updates starting from $\theta_N^{(\tau^*)}$ to obtain eigenvector estimate v_i, and set $\lambda_i := \widetilde{\mathcal{T}}(v_i, v_i, v_i)$.
end for
return the estimated eigenvalue/eigenvectors (λ, M).

11. Proof of Theorem 2

11.1. Recap of Perturbation Bounds for the Tensor Power Method

We now recap the result of Anandkumar et al. (2013a, Thm. 13) that establishes bounds on the eigen-estimates under good initialization vectors for the above procedure. Let $\mathcal{T} = \sum_{i\in[k]} \lambda_i v_i$, where v_i are orthonormal vectors and $\lambda_1 \geq \lambda_2 \geq \ldots \lambda_k$. Let $\widehat{\mathcal{T}} = \mathcal{T} + E$ be the perturbed tensor with $\|E\| \leq \epsilon_T$. Recall that N denotes the number of iterations of the tensor power method. We call an initialization vector u to be (γ, R_0)-good if there exists v_i such that $\langle u, v_i\rangle > R_0$ and

$|\langle u, v_i\rangle| - \max_{j<i} |\langle u, v_j\rangle| > \gamma |\langle u, v_i\rangle|$. Choose $\gamma = 1/100$.

Theorem 3 *There exists universal constants $C_1, C_2 > 0$ such that the following holds.*

$$\epsilon_T \leq C_1 \cdot \lambda_{\min} R_0^2, \qquad N \geq C_2 \cdot \left(\log(k) + \log\log\left(\frac{\lambda_{\max}}{\epsilon_T}\right)\right), \tag{14}$$

Assume there is at least one good initialization vector corresponding to each v_i, $i \in [k]$. The parameter ξ for choosing deflation vectors in each iteration of the tensor power method in Procedure 2 is chosen as $\xi \geq 25\epsilon_T$. We obtain eigenvalue-eigenvector pairs $(\hat{\lambda}_1, \hat{v}_1), (\hat{\lambda}_2, \hat{v}_2), \ldots, (\hat{\lambda}_k, \hat{v}_k)$ such that there exists a permutation η on $[k]$ with

$$\|v_{\eta(j)} - \hat{v}_j\| \leq 8\epsilon_T/\lambda_{\eta(j)}, \qquad |\lambda_{\eta(j)} - \hat{\lambda}_j| \leq 5\epsilon_T, \quad \forall j \in [k],$$

and

$$\left\| \mathcal{T} - \sum_{j=1}^{k} \hat{\lambda}_j \hat{v}_j^{\otimes 3} \right\| \leq 55\epsilon_T.$$

In the sequel, we establish concentration bounds that allows us to translate the above condition on tensor perturbation (14) to sample complexity bounds.

11.2. Concentration Bounds

11.2.1. Analysis of Whitening

Recall that we use the covariance operator $\mathcal{C}_{X_1X_2}$ for whitening the 3rd order embedding $\mathcal{C}_{X_1,X_2,X_3}$. We first analyze the perturbation in whitening when sample estimates are employed.

Let $\widehat{\mathcal{C}}_{X_1X_2}$ denote the sample covariance operator between variables X_1 and X_2, and let

$$B := 0.5(\widehat{\mathcal{C}}_{X_1X_2} + \widehat{\mathcal{C}}_{X_1X_2}^\top) = \widehat{\mathcal{U}}\widehat{S}\widehat{\mathcal{U}}^\top$$

denote the SVD. Let $\widehat{\mathcal{U}}_k$ and $\widehat{S}_k$ denote the restriction to top-k eigen-pairs, and let $B_k := \widehat{\mathcal{U}}_k\widehat{S}_k\widehat{\mathcal{U}}_k^\top$. Recall that the whitening matrix is given by $\widehat{\mathcal{W}} := \widehat{\mathcal{U}}_k\widehat{S}_k^{-1/2}$. Now $\widehat{\mathcal{W}}$ whitens B_k, i.e. $\widehat{\mathcal{W}}^\top B_k \widehat{\mathcal{W}} = I$.

Now consider the SVD of

$$\widehat{\mathcal{W}}^\top \mathcal{C}_{X_1X_2}\widehat{\mathcal{W}} = ADA^\top,$$

and define

$$\mathcal{W} := \widehat{\mathcal{W}}AD^{-1/2}A^\top,$$

and $\mathcal{W}$ whitens $\mathcal{C}_{X_1X_2}$ since $\mathcal{W}^\top \mathcal{C}_{X_1X_2} W = I$. Recall that by exchangeability assumption,

$$\mathcal{C}_{X_1,X_2} = \sum_{j=1}^{k} \pi_j \cdot \mu_{X|j} \otimes \mu_{X|j} = M \operatorname{Diag}(\pi) M^\top, \tag{15}$$

where the j^{th} column of M, $M_j = \mu_{X|j}$.

We now establish the following perturbation bound on the whitening procedure. Recall from (25), $\epsilon_{pairs} := \left\| \mathcal{C}_{X_1,X_2} - \widehat{\mathcal{C}}_{X_1,X_2} \right\|$. Let $\sigma_1(\cdot) \geq \sigma_2(\cdot) \ldots$ denote the singular values of an operator.

Lemma 4 (Whitening perturbation) *Assuming that $\epsilon_{pairs} < 0.5\sigma_k(\mathcal{C}_{X_1X_2})$,*

$$\epsilon_W := \|\operatorname{Diag}(\pi)^{1/2} M^\top (\widehat{\mathcal{W}} - \mathcal{W})\| \leq \frac{4\epsilon_{pairs}}{\sigma_k(\mathcal{C}_{X_1X_2})} \tag{16}$$

Remark: Note that $\sigma_k(\mathcal{C}_{X_1X_2}) = \sigma_k^2(M)$.

Proof: The proof is along the lines of Lemma 16 of (Anandkumar et al., 2013a), but adapted to whitening using the covariance operator here.

$$\begin{aligned} \|\operatorname{Diag}(\pi)^{1/2} M^\top (\widehat{\mathcal{W}} - \mathcal{W})\| &= \|\operatorname{Diag}(\pi)^{1/2} M^\top W (AD^{1/2}A^\top - I)\| \\ &\leq \|\operatorname{Diag}(\pi)^{1/2} M^\top \mathcal{W}\| \|D^{1/2} - I\|. \end{aligned}$$

Since $\mathcal{W}$ whitens $\mathcal{C}_{X_1X_2} = M\,\mathrm{Diag}(\pi)M^\top$, we have that $\|\mathrm{Diag}(\pi)^{1/2}M^\top\mathcal{W}\| = 1$. Now we control $\|D^{1/2} - I\|$. Let $\widetilde{E} := \mathcal{C}_{X_1,X_2} - B_k$, where recall that $B = 0.5(\widehat{\mathcal{C}}_{X_1,X_2} + \widehat{\mathcal{C}}_{X_1X_2}^\top)$ and B_k is its restriction to top-k singular values. Thus, we have $\|\widetilde{E}\| \leq \epsilon_{pairs} + \sigma_{k+1}(B) \leq 2\epsilon_{pairs}$. We now have

$$\begin{aligned}\|D^{1/2} - I\| &\leq \|(D^{1/2} - I)(D^{1/2} + I)\| \leq \|D - I\| \\ &= \|ADA^\top - I\| = \|\widehat{\mathcal{W}}^\top\mathcal{C}_{X_1X_2}\widehat{\mathcal{W}} - I\| \\ &= \|\widehat{\mathcal{W}}^\top\widetilde{E}\widehat{\mathcal{W}}\| \leq \|\widehat{\mathcal{W}}\|^2(2\epsilon_{pairs}).\end{aligned}$$

Now

$$\|\widehat{\mathcal{W}}^2\| \leq \frac{1}{\sigma_k(\widehat{\mathcal{C}}_{X_1X_2})} \leq \frac{2}{\sigma_k(\mathcal{C}_{X_1X_2})},$$

when $\epsilon_{pairs} < 0.5\sigma_k(\mathcal{C}_{X_1X_2})$. □

11.2.2. Tensor Concentration Bounds

Recall that the whitened tensor from samples is given by

$$\widehat{\mathcal{T}} := \widehat{\mathcal{C}}_{X_1X_2X_3} \times_1 (\widehat{\mathcal{W}}^\top) \times_2 (\widehat{\mathcal{W}}^\top) \times_3 (\widehat{\mathcal{W}}^\top).$$

We want to establish its perturbation from the whitened tensor using exact statistics

$$\mathcal{T} := \mathcal{C}_{X_1X_2X_3} \times_1 (\mathcal{W}^\top) \times_2 (\mathcal{W}^\top) \times_3 (\mathcal{W}^\top).$$

Further, we have

$$\mathcal{C}_{X_1X_2X_3} = \sum_{h\in[k]} \pi_h \cdot \mu_{X|h} \otimes \mu_{X|h} \otimes \mu_{X|h} \tag{17}$$

Let $\epsilon_{triples} := \|\widehat{\mathcal{C}}_{X_1X_2X_3} - \mathcal{C}_{X_1X_2X_3}\|$. Let $\pi_{\min} := \min_{h\in[k]} \pi_h$.

Lemma 5 (Tensor perturbation bound) *Assuming that $\epsilon_{pairs} < 0.5\sigma_k(\mathcal{C}_{X_1X_2})$, we have*

$$\epsilon_T := \|\widehat{\mathcal{T}} - \mathcal{T}\| \leq \frac{2\sqrt{2}\epsilon_{triples}}{\sigma_k(\mathcal{C}_{X_1X_2})^{1.5}} + \frac{\epsilon_W^3}{\sqrt{\pi_{\min}}}. \tag{18}$$

Proof: Define intermediate tensor

$$\widetilde{\mathcal{T}} := \mathcal{C}_{X_1X_2X_3} \times_1 (\widehat{\mathcal{W}}^\top) \times_2 (\widehat{\mathcal{W}}^\top) \times_3 (\widehat{\mathcal{W}}^\top).$$

We will bound $\|\widehat{\mathcal{T}} - \widetilde{\mathcal{T}}\|$ and $\|\widehat{\mathcal{T}} - \mathcal{T}\|$ separately.

$$\|\widehat{\mathcal{T}} - \widetilde{\mathcal{T}}\| \leq \|\widehat{\mathcal{C}}_{X_1,X_2,X_2} - \mathcal{C}_{X_1,X_2,X_3}\|\|\widehat{\mathcal{W}}\|^3 \leq \frac{2\sqrt{2}\epsilon_{triples}}{\sigma_k(\mathcal{C}_{X_1X_2})^{1.5}},$$

using the bound on $\|\widehat{\mathcal{W}}\|$ in Lemma 4. For the other term, first note that

$$\mathcal{C}_{X_1,X_2,X_3} = \sum_{h\in[k]} \pi_h \cdot M_h \otimes M_h \otimes M_h,$$

$$\begin{aligned}\|\widehat{\mathcal{T}} - \mathcal{T}\| &= \|\mathcal{C}_{X_1X_2X_3} \times_1 (\widehat{\mathcal{W}} - \mathcal{W})^\top \times_2 (\widehat{\mathcal{W}} - \mathcal{W})^\top \times_3 (\widehat{\mathcal{W}} - \mathcal{W})^\top\| \\ &\leq \frac{\|\mathrm{Diag}(\pi)^{1/2}M^\top(\widehat{\mathcal{W}} - \mathcal{W})\|^3}{\sqrt{\pi_{\min}}} \\ &= \frac{\epsilon_W^3}{\sqrt{\pi_{\min}}}\end{aligned}$$

□

Proof of Theorem 2: We obtain a condition on the above perturbation ϵ_T in (18) by applying Theorem 3 as $\epsilon_T \leq C_1\lambda_{\min}R_0^2$. Here, we have $\lambda_i = 1/\sqrt{\pi_i} \geq 1$. For random initialization, we have that $R_0 \sim 1/\sqrt{k}$, with probability $1-\delta$ using $\mathrm{poly}(k)\,\mathrm{poly}(1/\delta)$ trials, see Thm. 5.1 in (Anandkumar et al., 2012a). Thus, we require that $\epsilon_T \leq \frac{C_1}{k}$. Summarizing,

we require for the following conditions to hold

$$\epsilon_{pairs} \leq 0.5\sigma_k(\mathcal{C}_{X_1X_2}), \quad \epsilon_T \leq \frac{C_1}{k}. \tag{19}$$

We now substitute for ϵ_{pairs} and $\epsilon_{triples}$ in (18) using Lemma 6 and Lemma 7.

From Lemma 6, we have that

$$\epsilon_{pairs} \leqslant \frac{2\sqrt{2}\rho\sqrt{\log\frac{2}{\delta}}}{\sqrt{m}},$$

with probability $1-\delta$. It is required that $\epsilon_{pairs} < 0.5\sigma_k(\mathcal{C}_{X_1,X_2})$, which yields that

$$m > \frac{32\rho^2\log\frac{2}{\delta}}{\sigma_k^2(\mathcal{C}_{X_1,X_2})}. \tag{20}$$

Further we require that $\epsilon_T \leq C_1/k$, which implies that each of the terms in (18) is less than C/k, for some constant C. Thus, we have

$$\frac{2\sqrt{2}\epsilon_{triples}}{\sigma_k^{1.5}(\mathcal{C}_{X_1,X_2})} < \frac{C}{k} \quad \Rightarrow \quad m > \frac{C_3k^2\rho^3\log\frac{2}{\delta}}{\sigma_k^3(\mathcal{C}_{X_1,X_2})},$$

for some constant C_3 with probability $1-\delta$ from Lemma 7. Similarly for the second term in (18), we have

$$\frac{\epsilon_W^3}{\sqrt{\pi_{\min}}} < \frac{C}{k},$$

and from Lemma 4, this implies that

$$\epsilon_{pairs} \leq \frac{C'\pi_{\min}^{1/6}\sigma_k(\mathcal{C}_{X_1,X_2})}{k^{1/3}},$$

Thus, we have

$$m > \frac{C_4k^{\frac{2}{3}}\rho^2\log\frac{2}{\delta}}{\pi_{\min}^{\frac{1}{3}}\sigma_k^2(\mathcal{C}_{X_1,X_2})},$$

for some other constant C_4 with probability $1-\delta$. Thus, we have the result in Theorem 2.

□

11.2.3. Concentration bounds for Empirical Operators

Concentration results for the singular value decomposition of empirical operators.

Lemma 6 (Concentration bounds for pairs) *Let $\rho := \sup_{x\in\Omega} k(x,x)$, and $\|\cdot\|$ be the Hilbert-Schmidt norm, we have for*

$$\epsilon_{pairs} := \left\|\mathcal{C}_{X_1X_2} - \widehat{\mathcal{C}}_{X_1X_2}\right\|, \tag{21}$$

$$\Pr\left\{\epsilon_{pairs} \leqslant \frac{2\sqrt{2}\rho\sqrt{\log\frac{2}{\delta}}}{\sqrt{m}}\right\} \geqslant 1-\delta. \tag{22}$$

Proof We will use similar arguments as in (Rosasco et al., 2010) which deals with symmetric operator. Let ξ_i be defined as

$$\xi_i \;=\; \phi(x_1^i)\otimes\phi(x_2^i) - \mathcal{C}_{X_1,X_2}. \tag{23}$$

It is easy to see that $\mathbb{E}[\xi_i] = 0$. Further, we have

$$\sup_{x_1,x_2}\|\phi(x_1)\otimes\phi(x_2)\|^2 = \sup_{x_1,x_2} k(x_1,x_1)k(x_2,x_2) \leqslant \rho^2, \tag{24}$$

which implies that $\|\mathcal{C}_{X_1X_2}\| \leqslant \rho$, and $\|\xi_i\| \leqslant 2\rho$. The result then follows from the Hoeffding's inequality in Hilbert space. ■

Similarly, we have the concentration bound for 3rd order embedding.

Lemma 7 (Concentration bounds for triples) *Let $\rho := \sup_{x\in\Omega} k(x,x)$, and $\|\cdot\|$ be the Hilbert-Schmidt norm, we have for*

$$\epsilon_{triples} := \left\| \mathcal{C}_{X_1X_2X_3} - \widehat{\mathcal{C}}_{X_1X_2X_3} \right\|, \tag{25}$$

$$\Pr\left\{ \epsilon_{triples} \leqslant \frac{2\sqrt{2}\rho^{3/2}\sqrt{\log\frac{2}{\delta}}}{\sqrt{m}} \right\} \geqslant 1-\delta. \tag{26}$$

Proof We will use similar arguments as in (Rosasco et al., 2010) which deals with symmetric operator. Let ξ_i be defined as

$$\xi_i \;=\; \phi(x_1^i)\otimes\phi(x_2^i)\otimes\phi(x_3^i) - \mathcal{C}_{X_1X_2X_3}. \tag{27}$$

It is easy to see that $\mathbb{E}[\xi_i] = 0$. Further, we have

$$\sup_{x_1,x_2,x_3} \|\phi(x_1)\otimes\phi(x_2)\otimes\phi(x_3)\|^2 = \sup_{x_1,x_2,x_3} k(x_1,x_1)k(x_2,x_2)k(x_3,x_3) \leqslant \rho^3, \tag{28}$$

which implies that $\|\mathcal{C}_{X_1X_2X_3}\| \leqslant \rho^{3/2}$, and $\|\xi_i\| \leqslant 2\rho^{3/2}$. The result then follows from the Hoeffding's inequality in Hilbert space. ■

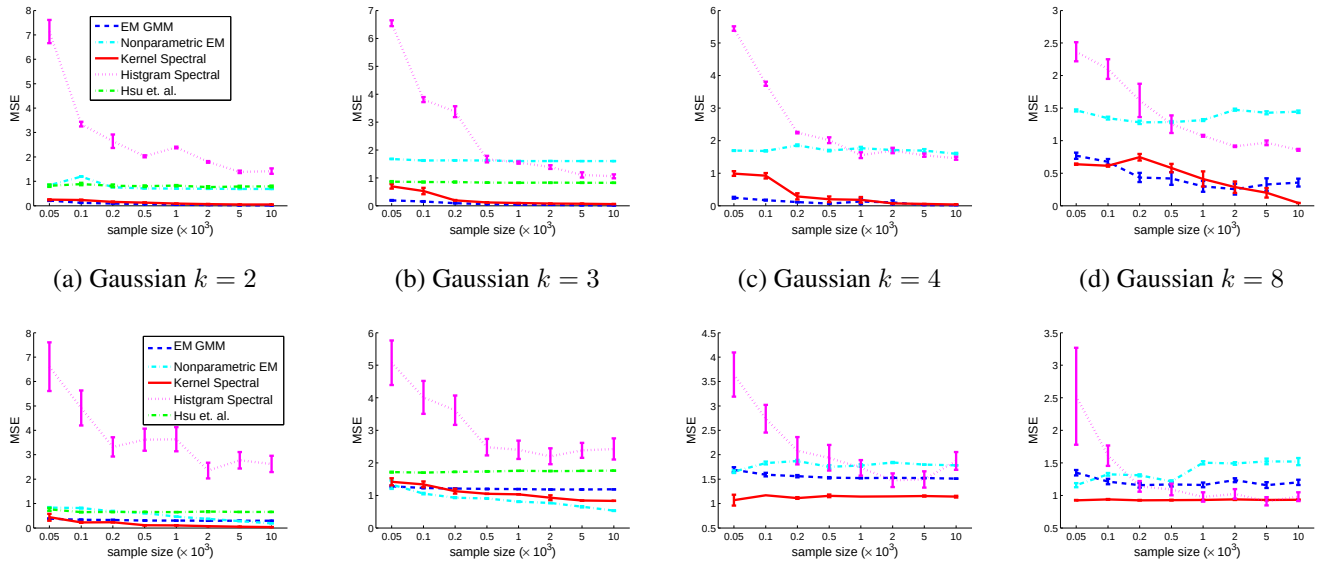
12. Experiment on Single Conditional Distribution

We also did some experiments for three-dimensional synthetic data that each view has the same conditional distribution. We generated the data from two settings:

1. Mixture of Gaussian conditional density;
2. Mixture of Gaussian and shifted Gamma conditional density.

The mixture proportion and other experiment settings are exact same as the experiment in the main text. The only difference is that the conditional densities for each view here are the identical. We use the same measure to evaluate the performance. The empirical results are plotted in Figure 5.

As we expected, the behavior of the proposed method is similar to the results in different conditional densities case. In mixture of Gaussians, our algorithm converges to the EM GMM resuls. And in the mixture of Gaussian/shift Gamma, our algorithm consistently better to other alternatives in most cases, except $k = 3$ where our method achieve comparable to nonparametric EM algorithm.



(a) Gaussian $k = 2$ (b) Gaussian $k = 3$ (c) Gaussian $k = 4$ (d) Gaussian $k = 8$

(e) Gaussian/Gamma $k = 2$ (f) Gaussian/Gamma $k = 3$ (g) Gaussian/Gamma $k = 4$ (h) Gaussian/Gamma $k = 8$

Figure 5. (a)-(d) Mixture of Gaussian distributions with $k = 2, 3, 4, 8$ components. (e)-(h) Mixture of Gaussian/Gamma distribution with $k = 2, 3, 4, 8$. For the former case, the performance of kernel spectral algorithm converge to those of EM algorithm for mixture of Gaussian model. For both cases, the performance of kernel spectral algorithm are consistently the best or comparable. Spherical Gaussian spectral algorithm does not work for $k = 4, 8$, and hence not plotted.