Tree-Structured Infinite Sparse Factor Model

XianXing Zhang XIANXING.ZHANG@DUKE.EDU
Electrical & Computer Engineering Department, Duke University, Durham, NC, USA

David B. Dunson DUNSON@STAT.DUKE.EDU
Department of Statistical Sciences, Duke University, Durham, NC, USA

Lawrence Carin LCARIN@DUKE.EDU
Electrical & Computer Engineering Department, Duke University, Durham, NC, USA

Abstract

A tree-structured multiplicative gamma process (TMGP) is developed, for inferring the depth of a tree-based factor-analysis model. This new model is coupled with the nested Chinese restaurant process, to nonparametrically infer the depth and width (structure) of the tree. In addition to developing the model, theoretical properties of the TMGP are addressed, and a novel MCMC sampler is developed. The structure of the inferred tree is used to learn relationships between high-dimensional data, and the model is also applied to compressive sensing and interpolation of incomplete images.

1. Introduction

Factor models are classical tools for analysis of high-dimensional data, widely utilized in the social sciences, statistics and machine learning literature. Such models seek to represent data in $\mathbb{R}^P$, typically for large P, as the superposition of a small number of factor loadings; each factor loading is also in $\mathbb{R}^P$, and the same typically small set of loadings are used to linearly represent each data sample. The sample-dependent weights on the loadings are termed factor scores. Recent developments include sparse PCA in which the loadings are regularized to be sparse, allowing for potentially interpretable loadings (Zou et al., 2004; Archambeau & Bach, 2009). Another direction of research involves nonlinear extensions, for example via a *mixture* of factor analysis models (MFAs) (Tipping & Bishop, 1999). In this setting each mixture component is a linear factor model, and cumulatively all mixture components yield a nonlinear mapping from data to factor scores. MFAs may be understood as a Gaussian mixture model with a low-rank assumption for the covariance matrix of each Gaussian (Roweis & Ghahramani, 1999). There are two model-selection challenges for an MFA: inferring the number of mixture components, and the number of factor loadings per mixture component (the number of loadings need not be the same for each mixture component). To address this problem, Bayesian priors (Griffiths & Ghahramani, 2006; Paisley & Carin, 2009; Knowles & Ghahramani, 2007; Bhattacharya & Dunson, 2010) have been utilized, allowing the number of factor loadings and mixture components to be inferred from the data (Rasmussen, 1999; Teh et al., 2006). For example, in (Chen et al., 2010) beta-Bernoulli priors were utilized to infer the number of factors, and a Dirichlet process was used to perform mixture modeling; this framework simultaneously learns the number of mixture components and the number of factor loadings in each mixture component. Using this approach (Chen et al., 2010) reported state-of-the-art results for a compressive sensing application. However, the method in (Chen et al., 2010) does not share factor loadings between mixture components, missing an opportunity to enhance statistical strength, and improve learning of relationships between the data.

In this paper we extend the mixture model setting to learn a multi-scale tree-structured hierarchy, with each factor loading defined by a node of the tree. Nodes and hence factor loadings may be shared among different mixture components (tree branches) and each tree branch is modeled as a probabilistic sparse PCA. The depth of each branch is inferred from the data, defining the number of factor loadings for a given mixture component. Further, the number of mixture components is also inferred, corresponding to the number of branches in the tree. The multi-scale nature of the learned factor loadings (tree) is of interest for model interpretation, allowing the viewing of data at multiple scales.

Appearing in *Proceedings of the 28^{th} International Conference on Machine Learning*, Bellevue, WA, USA, 2011. Copyright 2011 by the author(s)/owner(s).

Learning a tree-structured hierarchy of observed variables is an appealing but challenging approach for exploring latent structure. In (Jenatton et al., 2010) a set of dictionary elements embedded in a *prespecified* tree-structured hierarchy was developed, and the model was successfully applied to represent both natural images and documents. However, such hierarchical structure is often unobserved, and it is desirable that it be inferred from data. The combinatoric nature of selecting from among possible tree structures makes typical model-selection techniques impractical ($e.g.$, cross validation). In the conclusion to (Jenatton et al., 2010), the authors noted that the next major challenge is to infer dictionary-learning trees in a nonparametric Bayesian setting, to avoid the assumptions that they were required to make with regard to the structure of the tree; this paper seeks to address this research challenge, presenting a new nonparametric Bayesian model for learning tree-based hierarchical factor models.

The nested Chinese restaurant process (nCRP) (Blei et al., 2004) has been proposed as a generative probabilistic model for inferring a latent tree-structured hierarchy with an unbounded width, inferring semantic topics from a document corpus. Further, (Blei et al., 2010) extended this model to let the branch depth also be inferred from the data; this was done through modeling the discrete distribution over topics of each document using a stick-breaking process. In (Wang & Blei, 2009) the authors integrate the nCRP with factor analysis to model both continous data and discrete data, with factor loadings embedded in a tree as well; however, the depth of each branch was fixed in advance and as a result the number of factor loadings per branch cannot be inferred based on data.

A tree structure learned by nCRP has also been applied successfully in the computer-vision community, for example to discover latent hierarchies of images or high-level semantic information (Li et al., 2010; Bart et al., 2008). However, such models operate only on a discrete representation of data, in terms of a pre-defined codebook of features extracted from images. In contrast, the proposed model can learn the codebook (dictionary) at the same time it builds the tree-structured hierarchy for the continuous data. Further, the data are not mapped to a single codebook, but are represented as a linear combination of the factor loadings of the nodes of a tree branch. The stick-breaking process of (Blei et al., 2010) may not be applied readily to this problem. Other priors over an infinite tree have been proposed, but based on a different modeling philosophy; for example, the tree-structured stick-breaking prior (Adams et al., 2010) has been constructed to partition the data to nodes of a tree. The model proposed in (Rai & Daumé, 2008) is similar to our work in spirit, but the tree is restricted to be binary and requires a pseudo-time hazard process to model the depth of the tree.

To address the open problems elucidated above, this paper makes two principal contributions:

- A tree-structured multiplicative gamma process is developed; coupled with the nCRP, it manifests factor loadings embedded in a tree-structured hierarchy with unbounded depth and width. A convergence guarantee is also provided for the proposed model.
- We propose an efficient collapsed Gibbs sampler to explore the combinatorial tree-structured hierarchy space, automatically inferring the appropriate data-adapted depth of each branch.

2. Background

2.1. Nested Chinese restaurant process

The nested Chinese restaurant process (nCRP) (Blei et al., 2004; 2010) is a generative probabilistic model that defines a prior distribution over a tree-structured hierarchy with infinite many branches. We denote the infinite set of branches as $\mathcal{T} = \{\boldsymbol{b}^k\}_{k=1}^{\infty}$, with the superscript defining the kth branch; each branch $\boldsymbol{b}^k = \{b_l^k\}_{l=1}^{\infty}$ is a set of an infinite number of nodes, and the subscript means the lth layer of the branch. We use $|\boldsymbol{b}^k|$ to denote the size of set $\boldsymbol{b}^k$, defining the number of associated nodes. For observed variable $\{\boldsymbol{y}_i\}_{i=1}^{N}$, where N is the total number of data, a branch $\boldsymbol{b}^k \in \mathcal{T}$ is assigned to it according to a distribution specified below. Here we use $\boldsymbol{b}_i = \{b_{l,i}\}_{l=1}^{\infty}$ to represent the set of nodes chosen by sample i at each layer l of the branch. Finally, we use $l(n)$ to denote the layer that node n lives in, and $c(n)$ denotes the set of children nodes of node n.

Now assume that data sample i is at a particular parent node n; integer index $b_{c(n),i}$ defines the *child* of node n that sample i transitions to. In the nCRP the probability of which child node sample i transitions to is dictated by the behavior of the previous $i-1$ samples (the data are distributed within the tree sequentially). Specifically, the probability that sample i transits to child $b_{c(n),i} = k$ is $p(b_{c(n),i} = k|b_{c(n),1:i-1}) = \frac{\alpha}{m_n+\alpha}$ if k is a newly visited node, and $p(b_{c(n),i} = k|b_{c(n),1:i-1}) = \frac{m_{n,k}}{m_n+\alpha}$ otherwise. Integer m_n denotes the number of the previous $i-1$ samples that employ node n, $m_{n,k}$ denotes the number of these that employ child k, and α is a parameter controlling the probability of spawning a new node/branch; a different α may be used for each layer of the tree. Note that the nCRP statistical relationship is defined recursively for an infinite number of nodes $\{b_{l,i}\}_{l=1}^{\infty}$, which we simply denote as $\boldsymbol{b}_i \sim \text{nCRP}(\alpha)$.

2.2. Multiplicative gamma process

Consider a factor model of form $\boldsymbol{y}_i = \mathbf{D}\boldsymbol{z}_i + \boldsymbol{\epsilon}_i$, $\boldsymbol{\epsilon}_i \sim \mathcal{N}\left(\boldsymbol{\epsilon}_i|0, \boldsymbol{\Lambda}^{-1}\right)$, where $\mathbf{D} = \{d_{pn}, 1 \leq p \leq P, 1 \leq n \leq$

$K\}$ and $\boldsymbol{y}_i \in \mathbb{R}^P$, $\boldsymbol{z}_i \in \mathbb{R}^K$. The multiplicative gamma process (MGP) (Bhattacharya & Dunson, 2010) is defined on each d_{pn} as

$$d_{pn} \sim \mathcal{N}\left(d_{pn} \middle| 0, \phi_{pn}^{-1}\tau_n^{-1}\right), \ \phi_{pn} \sim \text{Ga}(\phi_{pn}|3/2, 3/2)$$

$$\tau_n = \prod_{l=1}^{n} \delta_l \ , \delta_1 \sim \text{Ga}(\delta_1|a_1, 1), \ \delta_l \sim \text{Ga}(\delta_l|a_2, 1), \ l \geq 2$$

where $\delta_l, l = 1, \ldots, \infty$ are independent. The τ_n is a globally shared shrinkage parameter for factor loadings $\boldsymbol{d}_n$, and ϕ_{pn} is a local shrinkage parameter for d_{pn}. The $\prod_{l \in p(n)} \delta_l$ are stochastically increasing under the restriction $a_2 > 1$, which favors more shrinkage as n increases.

Although each draw of δ_l from a gamma distribution is not guaranteed to be greater than one, in practice for normalized data δ_l is inferred to be great than one when $a_2 > 1$, for moderately large l ($l \geq 3$ in all our experiments). However, an MGP based on a left-truncated gamma distribution may be easily derived: $\tau_n = \prod_{l=1}^{n} \delta_l$, $(\delta_l - 1) \sim \text{Ga}(\delta_l - 1|a, 1)$, where both the conjugacy and theoretical properties are retained (Bhattacharya & Dunson, 2010). In the following we only focus on the non-truncated version of MGP.

3. Proposed Model

3.1. Model and prior specification

To learn an infinite tree-structured hierarchical model means to infer both the number of tree branches and depth of each branch. To address the first problem we adopt the nCRP prior. As opposed to other priors on infinite trees (Mauldin et al., 1992; Rai & Daumé, 2008), the nCRP has the flexibility of allowing an unbounded number of children nodes for each parent node, rather than only allowing two children; this enhances model flexibility, removing redundant inner nodes (Adams et al., 2010). Let $\boldsymbol{b}_i$ represent the branch that data $\boldsymbol{y}_i$ chooses, according to the nested Chinese restaurant process (nCRP): $\boldsymbol{b}_i \sim \text{nCRP}(\boldsymbol{\alpha})$ where $\boldsymbol{\alpha} = \{\alpha_1, \alpha_2, \ldots, \alpha_\infty\}$, allowing different α for each layer of the tree.

Assuming a Gaussian noise/residual model, observed data $\boldsymbol{y}_i \in \mathbb{R}^P$ are assumed drawn

$$\boldsymbol{y}_i \sim \mathcal{N}\left(\boldsymbol{y}_i \middle| \sum_{n \in \boldsymbol{b}_i} \boldsymbol{d}_n x_{ni} + \boldsymbol{m}_{\boldsymbol{b}_i}, \boldsymbol{\Lambda}^{-1}\right) \qquad (1)$$

where $\boldsymbol{\Lambda} = \text{diag}\{\lambda_1, \ldots, \lambda_P\}$ is a diagonal precision matrix. The set of N data samples are denoted $\mathbf{Y} = \{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N\}$. Factor loading $\boldsymbol{d}_n$ is associated with node n in the tree, and $x_{ni} \in \mathbb{R}$ is the associated weight (factor score) on this factor loading for data $\boldsymbol{y}_i$; vector $\boldsymbol{m}_{\boldsymbol{b}_i} \in \mathbb{R}^P$ is the mean on branch $\boldsymbol{b}_i$. The diagonal $\boldsymbol{\Lambda} \in \mathbb{R}^{P \times P}$ allows the residual precision to vary across the P components of the data, and we place a gamma prior $\text{Ga}(a_0, b_0)$ on each diagonal element. Note that we could make $\boldsymbol{\Lambda}$ be branch specific. We impose the following priors on the factor loadings, scores and means: $d_{pn} \sim \mathcal{N}\left(d_{pn} \middle| 0, \gamma_{pn}^{-1}\right)$, $m_{p\boldsymbol{b}_i} \sim \mathcal{N}\left(m_{p\boldsymbol{b}_i} \middle| 0, \xi_{p\boldsymbol{b}_i}^{-1}\right)$, $x_{ni} \sim \mathcal{N}\left(x_{ni} \middle| 0, 1\right)$. We restrict x_{ni} to be drawn from a unit-variance standard Gaussian because of the arbitrary sharing of scale between x_{ni} and γ_{pn}^{-1}, as discussed in (Roweis & Ghahramani, 1999). Upon marginalizing out the factor scores, we have

$$\boldsymbol{y}_i \sim \mathcal{N}\left(\boldsymbol{y}_i \middle| \boldsymbol{m}_{\boldsymbol{b}_i}, \boldsymbol{\Omega}_{\boldsymbol{b}_i}\right) \qquad (2)$$

with $\boldsymbol{\Omega}_{\boldsymbol{b}_i} = \sum_{n \in \boldsymbol{b}_i} \boldsymbol{d}_n \boldsymbol{d}_n' + \boldsymbol{\Lambda}^{-1}$ where $\boldsymbol{d}_n'$ denotes the transpose of column vector $\boldsymbol{d}_n$. Note that for any two tree branches (mixture components) $\boldsymbol{b}^i$ and $\boldsymbol{b}^j$, the covariance matrices $\boldsymbol{\Omega}_{\boldsymbol{b}^i}$ and $\boldsymbol{\Omega}_{\boldsymbol{b}^j}$ are partly shared (via the shared nodes).

Notice that data associated with each branch $\boldsymbol{b}^k \in \mathcal{T}$ is modeled via a factor model, and the rank of each factor model is $|\boldsymbol{b}^k|$. However, $|\boldsymbol{b}^k|$ is unbounded, as each branch is drawn from nCRP. Thus an extra condition is needed for (2) to be well defined. Toward this end, we extend the multiplicative gamma process to a tree-structured multiplicative gamma process (TMGP): denote $p(n)$ as the set of ancestors of node n (those nodes above node n) and for each node n in the infinite tree, we define the TMGP for d_{pn}'s precision parameter

$$\gamma_{pn} = \phi_{pn} \prod_{l \in p(n)} \zeta_l, \ \phi_{pn} \sim \text{Ga}\left(\phi_{pn}|3/2, 3/2\right)$$

$$\zeta_1 \sim \text{Ga}\left(\zeta_1|c_1, 1\right), \ \zeta_l \sim \text{Ga}\left(\zeta_l|c_2, 1\right), \ l \geq 2 \qquad (3)$$

denoted simply as $\boldsymbol{\gamma}_p \sim \text{TMGP}(c_1, c_2)$. As for MGP, the TMGP is also *conjugate* to the precision parameter in a normal density function, allowing an efficient sampling scheme, as discussed below. Note that for indices n corresponding to nodes that are deeper in the tree, the parameter γ_{pn} increases. Thus with TMGP each tree branch is modeled as a probabilistic sparse PCA, or sparse FA if diagonal covariance matrix is employed. Note that for usual shrinkage priors on the loadings they exhibit the phenomenon of *factor splitting*, in which none of the columns $\boldsymbol{d}_n, n \in \boldsymbol{b}$ have all loading elements d_{pn} close to zero even when $l(n)$ is large. The TMGP avoids this problem by shrinking increasingly in columns $\boldsymbol{d}_n$ for which $l(n)$ is large. More specifically, this choice of shrinkage prior on the infinite number of factor loadings and means allows $\boldsymbol{\Omega}_{\boldsymbol{b}^k}$ to converge almost surely for every infinite branch $\boldsymbol{b}^k \in \mathcal{T}$, as stated in the following theorem; a sketch of the proof can be found in the supplemental material.

Theorem 1. *For all* $\boldsymbol{b}^k \in \mathcal{T}$, *the covariance matrix* $\boldsymbol{\Omega}_{\boldsymbol{b}^k} = \sum_{n \in \boldsymbol{b}^k} \boldsymbol{d}_n \boldsymbol{d}_n' + \boldsymbol{\Lambda}^{-1}$ *converges almost surely.*

Our model can be thought as an innovative tree-structured extension of infinite Gaussian mixture model (iGMM)

(Rasmussen, 1999), with means and low-rank covariance matrices shared by mixture components via a tree topology. Specifically, we can rewrite the model as

$$\boldsymbol{y} \sim \sum_{\boldsymbol{b}_k \in \mathcal{T}} w_{\boldsymbol{b}_k} \mathcal{N}\left(\boldsymbol{y} \middle| \boldsymbol{m}_{\boldsymbol{b}_k}, \boldsymbol{\Omega}_{\boldsymbol{b}_k}\right) \tag{4}$$

where $\boldsymbol{\Omega}_{\boldsymbol{b}_k}$ is defined in (2), $w_{\boldsymbol{b}_k}$ is the mixture weight of this tree-structured iGMM drawn from the tree-structured stick-breaking process introduced in (Wang & Blei, 2009). We will discuss the usage of this specific formulation in Section 5, where *analytic* compressive sensing(CS) inversion is performed.

4. Posterior inference

4.1. Truncate tree branch depth to finite

For computational purposes, we would like to approximate the infinite set of nodes on each branch $\boldsymbol{b}^k \in \mathcal{T}$ of the tree (which correspond to an infinite set of factor loadings) to a finite set $\boldsymbol{b}^k(L_k) = \{b_1^k, b_2^k, \ldots, b_{L_k}^k\}$. Denote the truncated tree $\widehat{\mathcal{T}}(\mathbf{L}) = \{\boldsymbol{b}^k(L_k)\}_{k=1}^{\infty}$ and $|\widehat{\mathcal{T}}(\mathbf{L})|$ as the number of truncated branches in the tree. As justification, we show theoretical bounds on the depth truncation approximation error between $\boldsymbol{b}^k$ and $\boldsymbol{b}^k(L_k)$. In the following discussion we discard the branch-specific superscript k for notational simplicity. Let $\boldsymbol{\Omega}_{\boldsymbol{b}(L)} = \sum_{n \in \boldsymbol{b}(L)} \boldsymbol{d}_n \boldsymbol{d}_n' + \boldsymbol{\Lambda}^{-1}$ represent the truncated version of $\boldsymbol{\Omega}_{\boldsymbol{b}}$; the following theorem states that the prior probability of $\boldsymbol{\Omega}_{\boldsymbol{b}(L)}$ being arbitrarily close to $\boldsymbol{\Omega}_{\boldsymbol{b}}$ increases *exponentially* fast to one as L tends to infinity, generalizing Theorem 2.4 in (Bhattacharya & Dunson, 2010) to a tree-structured hierarchal setting and the proof can be found therein.

Theorem 2. *If* $c_2 > 1$, *then* $\forall \epsilon > 0$, $\forall \boldsymbol{b} \in \mathcal{T}$,

$$\exists \hat{L} = \frac{\log(4Pb/\epsilon(1-a))}{\log(1/a)}, \text{ s.t. when } L > \hat{L}$$

$$p(d_\infty(\boldsymbol{\Omega}_{\boldsymbol{b}}, \boldsymbol{\Omega}_{\boldsymbol{b}(L)}) > \epsilon) < \frac{6Pb}{\epsilon(1-a)} a^L$$

where $a = E(\delta_1^{-1})$ *and* $a_2 = E(\delta_2^{-1})$, *and* $d_\infty(A, B) = \max_{1 \le r,s \le p} |a_{r,s} - b_{r,s}|$ *is the sup-norm metric for* $P \times P$ *matrices* $A = (a_{rs}), B = (b_{rs})$.

4.2. Collapsed Gibbs sampler with fixed truncation

We propose an efficient collapsed Gibbs sampler with fixed truncation level for simultaneously exploring the parameter space and the large latent tree-structured hierarchy. The Gibbs sampler can be divided into two parts:

4.2.1. GIVEN $\{\boldsymbol{b}_i\}_{i=1}^N$ SAMPLE OTHER PARAMETERS

With known branch assignments, the inference reduces to a conventional sampler for factor models. Factor loading $\boldsymbol{d}_n$, factor score x_{ni}, branch mean $\boldsymbol{m}_{\boldsymbol{b}}$ and precision matrix Λ defined in equation (1) can be sampled from their corresponding conditional distribution which we do not reproduce here. Due to the conjugacy of the TMGP parameters defined in (3), they can be sampled directly from their conditional distribution $p(\cdot|-)$ given all other parameters (Bhattacharya & Dunson, 2010). Denote C_n as the set of children nodes of n and $|C_n|$ as the size of that set we have:

$$p(\phi_{pn}|-) = \text{Ga}\left(\frac{\nu+1}{2}, \frac{\nu + \prod_{l \in p(n)} \zeta_p d_{pl}^2}{2}\right) \tag{5}$$

$$p(\zeta_n|-) = \text{Ga}\left(\hat{c}_n + \frac{P|C_n|}{2}, 1 + \frac{\sum_c \tau_c^{(n)} \sum_{p=1}^{P} \phi_{pc} d_{pc}^2}{2}\right)$$

where $\nu = 3$ as parameterized in our previous setting, and $\hat{c}_1 = c_1, \hat{c}_n = c_2$ for $n > 1$, and $\tau_c^{(n)} = \prod_{t \in p_c, t \neq n} \zeta_t$, for all children nodes $c \in C_n$. Finally, nCRP hyperparameter $\boldsymbol{\alpha}$ and hyperparameters a_0, b_0 on diagonal precision matrix Λ are updated using standard Metropolis-Hastings steps within the Gibbs sampler (Blei et al., 2010).

4.2.2. SAMPLE $\{\boldsymbol{b}_i\}_{i=1}^N$ GIVEN OTHER PARAMETERS

Denote all the hyperparameters as $\boldsymbol{\theta}$, for sample i the conditional distribution of choosing $\boldsymbol{b}_L \in \widehat{\mathcal{T}}(\mathbf{L})$ is:

$$\begin{aligned} & p(\boldsymbol{b}_i = \boldsymbol{b}_L | \{\boldsymbol{d}_l, \boldsymbol{m}_l\}_{l \in \boldsymbol{b}}, \boldsymbol{b}_{-i}, \boldsymbol{y}_i, \boldsymbol{\theta}) \\ \propto & p(\boldsymbol{y}_i | \{\boldsymbol{d}_l, \boldsymbol{m}_l\}_{l \in \boldsymbol{b}}, \boldsymbol{\theta}) p(\boldsymbol{b}_i = \boldsymbol{b}_L | \boldsymbol{b}_{-i}) \end{aligned} \tag{6}$$

where $\boldsymbol{b}_{-i}$ denotes the tree branch assignments for all data other than sample i. This expression is an outcome of Bayes' rule, where $p(\boldsymbol{b}_i = \boldsymbol{b}_L | \boldsymbol{b}_{-i})$ is the prior of choosing $\boldsymbol{b}_i$ given the choices of all other data, $p(\boldsymbol{y}_i | \{\boldsymbol{d}_l, \boldsymbol{m}_l\}_{l \in \boldsymbol{b}_i}, \boldsymbol{b}_i)$ is the data likelihood of the data $\boldsymbol{y}_i$ given a particular tree branch assignment $\boldsymbol{b}_i$ as formulated in (2), where the latent factors in (1) are integrated out for faster mixing of the sampler.

Note that to evaluate (2) we need the precision matrix and the determinant for every branch $\boldsymbol{b}_L \in \widehat{\mathcal{T}}(\mathbf{L})$, and for a tree with $|\widehat{\mathcal{T}}(\mathbf{L})|$ branches the computational cost is approximately $\mathcal{O}(P|\widehat{\mathcal{T}}(\mathbf{L})|^2)$ if advanced techniques are employed (Roweis & Ghahramani, 1999) but still quadratic in $|\widehat{\mathcal{T}}(\mathbf{L})|$, which is computationally prohibitive as $|\widehat{\mathcal{T}}(\mathbf{L})|$ grows *exponentially* fast to the branch truncation level $\mathbf{L}$. Note that since the branch depth is modeled as the intrinsic latent dimension of observed variables, this issue will be critical when handling complex data, *e.g.,* when P is large. In the following we propose an efficient Gibbs sampler by exploring the tree structure that scales as $\mathcal{O}(P|\widehat{\mathcal{T}}(\mathbf{L})|)$.

Writing $\boldsymbol{\Omega}_{\boldsymbol{b}(0)} = \boldsymbol{\Lambda}$ and $\boldsymbol{\Omega}_{\boldsymbol{b}(l)} = \sum_{n \in \boldsymbol{b}(l)} \boldsymbol{d}_n \boldsymbol{d}_n' + \boldsymbol{\Lambda}$ which is interpreted as the covariance for branch $\boldsymbol{b}(l)$ with truncation level l, then for $1 \le l \le L$ we have the recursive representation of covariance matrix: $\boldsymbol{\Omega}_{\boldsymbol{b}(l)} = \boldsymbol{d}_{b_l} \boldsymbol{d}_{b_l}' + \boldsymbol{\Omega}_{\boldsymbol{b}(l-1)}$.

Denote $\mathbf{\Gamma}_{\boldsymbol{b}}^{l} = (\mathbf{\Omega}_{\boldsymbol{b}(l)})^{-1}$ as the precision matrix of branch $\boldsymbol{b}(l)$ with truncation level l, and writing $\mathbf{\Gamma}_{\boldsymbol{b}}^{0} = \mathbf{\Lambda}^{-1}$, then based on the recursive representation and Sherman-Morrison-Woodbury matrix identities we can calculate the matrix inversion and determinant recursively by operating on matrix $\boldsymbol{d}'_{b_l}\mathbf{\Gamma}_{\boldsymbol{b}(l)}\boldsymbol{d}_{b_l} + 1$ of dimension 1 for $L+1$ times from $l = L$ to 0 as explained below:

$$\mathbf{\Gamma}_{\boldsymbol{b}}^{l} = \mathbf{\Gamma}_{\boldsymbol{b}}^{l-1} - \mathbf{\Gamma}_{\boldsymbol{b}}^{l-1}\boldsymbol{d}_{b_l}\left(\boldsymbol{d}'_{b_l}\mathbf{\Gamma}_{\boldsymbol{b}}^{l-1}\boldsymbol{d}_{b_l} + 1\right)^{-1}\boldsymbol{d}'_{b_l}\mathbf{\Gamma}_{\boldsymbol{b}}^{l-1} \quad (7)$$

$$\frac{1}{|\mathbf{\Gamma}_{\boldsymbol{b}}^{l}|} = \frac{|\boldsymbol{d}'_{b_l}\mathbf{\Gamma}_{\boldsymbol{b}}^{l-1}\boldsymbol{d}_{b_l} + 1|}{|\mathbf{\Gamma}_{\boldsymbol{b}}^{l-1}|} \quad (8)$$

An important observation from (7) and (8) is that, since matrix $\mathbf{\Gamma}_{\boldsymbol{b}}^{l}$ corresponds to the precision matrix on branch $\boldsymbol{b}(l)$ with truncation level l, its result can be reused when computing matrices $\mathbf{\Gamma}_{\boldsymbol{b}}^{l'}$ by *all* branches $\boldsymbol{b}(l')$ with $\boldsymbol{b}(l) \subset \boldsymbol{b}(l')$. Thus we can make use of *breadth first search* (BFS) of the tree to transform the heavy computations of branch specific precision matrices and determinants into operations on each node within one sweep of the tree, where the computational cost on each node is simply $\mathcal{O}(P)$. Since the number of nodes $\mathcal{N} = \mathcal{O}(|\widehat{\mathcal{T}}(\mathbf{L})|)$, the computation cost is reduced from $\mathcal{O}(P|\widehat{\mathcal{T}}(\mathbf{L})|^2)$ to $\mathcal{O}(P|\widehat{\mathcal{T}}(\mathbf{L})|)$.

4.3. Truncating branches using adaptive Gibbs sampler

The above Gibbs sampler needs a predefined depth truncation level. However, its desirable to have a computational strategy for choosing an appropriate level of truncation $L_{\boldsymbol{b}}$ automatically for each $\boldsymbol{b} \in \mathcal{T}$. Here we extend the adaptive Gibbs sampler proposed in (Bhattacharya & Dunson, 2010) to our setting.

We modify the sampler described above, tuning the number of loadings on each branch $\boldsymbol{b}(L_{\boldsymbol{b}})$ as the sampler progresses. To be specific, we trigger the adaptation procedure with probability $p(t) = \exp(z_0 + z_1 t)$ at the tth iteration, with z_0, z_1 chosen so that adaptation occurs around every 10 iterations at the beginning of the chain but decreases in frequency exponentially fast. Denote $L_{\boldsymbol{b}}^{\star}$ as the underlying true number of loadings on branch $\boldsymbol{b}$, and the adaptive sampler starts with a conservative guess $L_{\boldsymbol{b}}$ of $L_{\boldsymbol{b}}^{\star}$. If the adaptation is triggered at iteration t, let $q_\delta(t) = \{n | c(n) = \emptyset, ||\boldsymbol{d}_n||_p < \delta\}$ denotes the set of tree leaves with corresponding loading's ℓ_p norm less than some pre-specified threshold δ. Intuitively for each branches $\boldsymbol{b}(L_{\boldsymbol{b}})$ if its leaf $b_{L_{\boldsymbol{b}}} \in q_\delta(t)$ then its loading has a negligible contribution at the tth iteration to the covariance, and thus removed. On the other hand, if leaf node $b_{L_{\boldsymbol{b}}} \notin q_\delta(t)$ then it suggests that branch $\boldsymbol{b}_{L_{\boldsymbol{b}}}$ may need more parameters to model the data that live in it, and as a result $\boldsymbol{b}_{L_{\boldsymbol{b}}}$ is replaced by $\boldsymbol{b}_{L_{\boldsymbol{b}}+1}$ with a new leaf node $b_{L_{\boldsymbol{b}}+1}$ introduced with parameters draw from prior distribution.

An important aspect of the adaptive Gibbs sampler is that the convergence of the chain is *guaranteed*, as the adaptations are designed to satisfy the diminishing adaptation condition in Theorem 5 of (Roberts & Rosenthal, 2007), which we do not reproduce here for brevity.

5. Experiments

In all experiments the hyperparameters of TMGP were set as $c_1 = 1$, $c_2 = 3$, to ensure Theorems 1 and 2 hold. In the adaptive sampler we adopted the ℓ_2 norm with $z_1 = -0.5$, and $z_2 = -0.001$. An important thresholding parameter δ is introduced by TMGP to discard the factor loadings that has ℓ_2 norm less than δ, and the learned size of the tree is sensitive to δ. Relative large δ will lead to better predictive performance while introduce more factor loadings, thus the choice of δ is a trade-off between performance and scalability. However, it's not required to fix the value of δ in advance and we can vary it based on the model output (e.g., the number of nodes) during the early stage of MCMC chain. This is because the adaptive Gibbs sampler introduced in Section 4.3 has the flexibility of changing the value of δ , where the convergence guarantee would still be met as long as the diminishing adaptation condition meets.

All quantitative results below were obtained based on multiple posterior samples, and for the tree structure we will show only a single (representative) sample from the posterior distribution for illustration, as discussed in (Blei et al., 2010). Unless stated otherwise, we discarded the first 5000 burn-in samples and collected 500 samples from every 10 iterations after burn-in. All the experiments were conducted on a cluster of blade-based servers with 2.5 GHz clock frequency, eight CPU-cores and 16 Gb shared RAM. As an example, for the faces data considered next the MCMC sampler required around 35 seconds per sample.

5.1. Face data

We first consider the face dataset studied in (Tenenbaum et al., 2000). It contains a total of $N = 698$ faces, each with $P = 4906$ pixels; the images are from the *same* subject, but with *different* pose and illumination. In Figure 1 we present the inferred hierarchical tree. Each image is assigned to a branch of the tree, and modeled by a FA model on that branch (one factor loading at each node). In Figure 1, the image at tree node n is the average of all data $\frac{1}{N_n}\sum_{i:n\in\boldsymbol{b}_i}\boldsymbol{y}_i$ that live in that node, where N_n is the total number of such data. Note that a parent may have a single child; if a parent has a single child, this is equivalent to multiple factor loadings contributing to the same node (since there is no branch splitting).

The results are presented in a manner such that disagreements in the pose/illumination of data on the same node manifests blurriness of the average image at that node. The

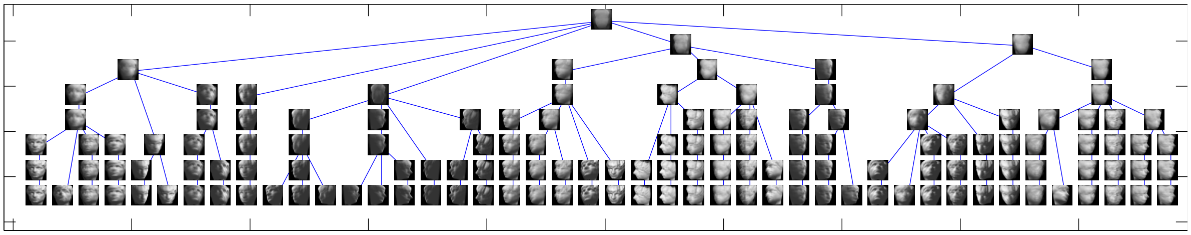

Figure 1. The full tree structure inferred from faces data where each node is plotted as the average of all images that were assigned to that node. Leaves at branches with different depth are placed on same horizontal level for purpose of interpretation.

model captures common structure (nodes on the top layers) and idiosyncrasies (bottom nodes and leaves) characteristic of the whole dataset. The degree of similarity between two clusters (branches) is manifested by the number of nodes they share. By contrast, conventional mixture model based clustering methods (Chen et al., 2010; Rasmussen, 1999) cannot capture the intrinsic relation among observed variables above, because they are modeled to be conditionally independent given cluster assignments. We further studied the proposed model in the context of compressive sensing (CS), with comparison to factor analysis (FA) and traditional mixture of factor analyzer (MFA) models; the latter has achieved state-of-the-art performance in a recent study (Chen et al., 2010). We randomly divide the faces data into a training subset of 598 images, with a testing subset of 100 images, and the relative CS reconstruction error is defined as $\frac{||\mathbf{X}-\hat{\mathbf{X}}||_F}{||\mathbf{X}||_F}$ where $\mathbf{X} \in \mathbb{R}^{4906\times 100}$ is the testing data set and $\hat{\mathbf{X}}$ is the reconstruction. Note that because the underlying model is a low-rank GMM, as shown in (4), we may use the same *analytic* CS inversion as developed in (Chen et al., 2010). In order to perform a fair comparison, we also use the adaptive Gibbs sampler and MGP to infer the number loadings used in the comparison MFA, and for the single (non-mixture) FA. For the MFA model, the same Dirichlet process model as considered in (Chen et al., 2010) is used to infer the number of FA mixture components. We ran the CS analysis 10 times, for different partitions of the training and test data, and the average reconstruction performance of the models is summarized in Figure 2. We observe that the proposed model is better on average, and has tighter variance, than both the MFA (Chen et al., 2010) and FA alternatives. These are believed to be state-of-the-art CS recovery results for data that live on a low-dimensional subspace of $\mathbb{R}^P$.

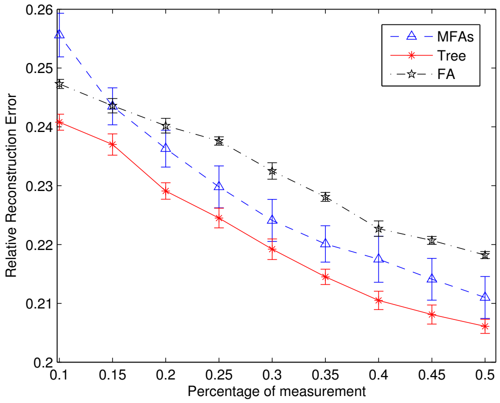

Figure 2. CS reconstruction error for faces data. The curves represent the average over 10 partitions of the data, and the error bars denote standard deviation.

5.2. Cell Line Panel

The HGDP-CEPH Human Genome Diversity Cell Line Panel (Rosenberg et al., 2002) is a dataset comprising genotypes at $P = 377$ autosomal microsatellite loci, sampled from $N = 1056$ individuals in 52 populations across the major geographic regions of the world. It is useful for inferring human evolutionary history and migration. Each data sample has a label that indicates which area it comes from, and there are 26 areas corresponding to 22 countries.

In this experiment we study the hierarchical clustering of our model through analyzing the relationship between the tree-structured hierarchy learned from the data; we relate the results to the geographical locations of the data (geography is not used in the analysis itself, only for presentation). In Figure 3, the top picture plots the inferred tree structure learned from the data, and the middle and bottom two maps illustrate the node-clustering results of the countries on the second and third layer of the tree. We assign each area into one node if most of its data are mapped to it in the learned tree structure. If two areas/countries share the same color, this indicates that they belong to the same node. Consider the middle of Figure 3, which corresponds to layer two in the tree. If a country at that layer is uniquely associated with one node (e.g., Russian and China), this will also be true at layer 3 (bottom), as they will have a unique set of children nodes. If two or more countries share a node at layer 2 (e.g., Mexico, Brazil and Columbia), they may be distinguished at the third layer (note that Brazil separates from these three at layer 3, the bottom in Figure 3).

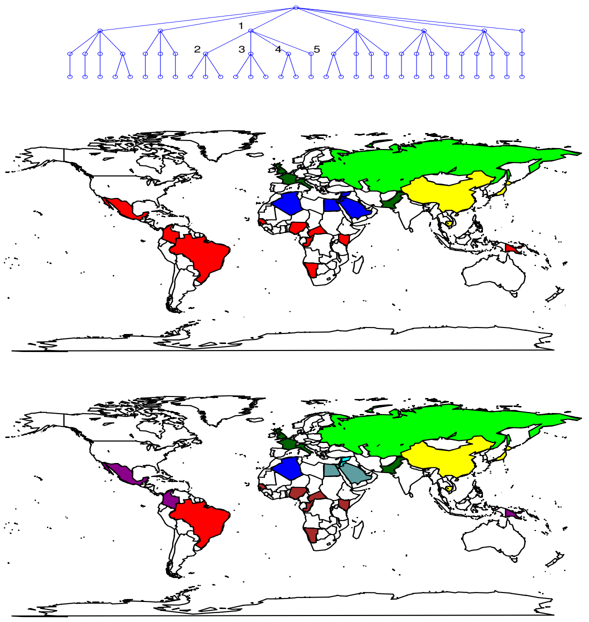

Figure 3. Summary of cell-line results. Top: inferred tree structure. Middle: layer-2 association of countries with nodes (denoted by colors). Bottom: layer-3 association of countries with nodes (denoted by colors). As examples, consider the nodes numbered on the tree (top). On layer-2, node 1 is represented as red in the middle map (Central South America and Central South Africa). On layer-3, node 2 is represented as purple (Mexico and Columbia); node 3 is represented as brown (Central South Africa); node 4 has no corresponding color because none of the countries in the map has majority of data clustered to it; and node 5 is represented as red (Brazil).

Note that for both the second and third tree layers, western countries UK, France and Italy are clustered together with Pakistan, consistent with a previous analysis of these data (Rosenberg et al., 2002). We found that the samples from a given country were generally strongly associated with a particular node, at each scale. For example, 73% of the China samples were associated with one node at layer 2. As another example, for Italy 90% were in the same node at layer 2.

5.3. Natural Image Patches

In the last experiment we test our model on interpolating ("inpainting") missing pixels from images, as also considered in (Jenatton et al., 2010) with a *specified* tree (here the tree structure is learned). In (Jenatton et al., 2010) the authors studied the same problem, and made comparisons to a "flat" model, which here is a conventional FA. We also make comparisons to a "flat" FA model, and also to the same class of MFA models studied above in the context of compressive sensing.

Table 1. Quantitative results of the reconstruction tasks on natural image patches. First row: percentage of missing pixels. Second and third row: mean square error multiplied by 100

	50%	60%	70%	80%
FA	17.6 ± 0.2	22.3 ± 0.1	30.1 ± 0.0	47.7 ± 0.0
MFA	16.1 ± 0.3	22.6 ± 0.2	31.6 ± 0.3	50.2 ± 0.4
Tree	16.6 ± 0.3	21.1 ± 0.2	29.8 ± 0.3	41.3 ± 0.1

We extracted $125,000$ non-overlapping patches of $P = 64$ pixels (8×8 patches), from the Berkeley segmentation database of natural images. We divided them into a training set $\mathbf{X}_{tr}$ of size $100,000$ and a testing set $\mathbf{X}_{te}$ of size $25,000$. The tree learning based on $\mathbf{X}_{tr}$ uses the complete data, and $\mathbf{X}_{te}$ is analyzed in the presence of missing pixels (selected uniformly at random); this is the same task as (Jenatton et al., 2010) considered. When learning the model, we ran the adaptive Gibbs sampler 10,000 iterations on $\mathbf{X}_{tr}$, and retained the maximum-likelihood sample (defining the tree structure and associated multi-scale dictionary). This model was then fixed, and 5000 Gibbs iterations were then employed when analyzing $\mathbf{X}_{te}$.

An example of a learned dictionary embedded in the tree structure learned from $\mathbf{X}_{tr}$ is shown in Figure 4, and the quantitative reconstruction results are reported on Table 1. Note that we only plot the top six layers of the tree because of the limited space. As observed from Figure 4, the dictionary elements embedded at the bottom of the tree structure corresponds to detail information of the data set, whose ℓ_2 norms are small (around 0.2) and thus contribute less to the model. This is consistent with the assumption imposed by TMGP. From Table 1 we observe that the improvement of tree-structured model is most significant when there are most missing values in $\mathbf{X}_{te}$, similar results were also reported in (Jenatton et al., 2010).

6. Conclusions

A new model has been developed for inferring the structure of a latent tree, used to encode relationships between loadings in a factor model. In addition to developing model properties, an efficient MCMC inference engine has been developed. Several encouraging experimental results have been presented, significantly generalizing related and motivating models that assumed that the tree structure was known (Jenatton et al., 2010).

Acknowledgements

We would like to thank the anonymous reviewers for their insightful comments. The research reported here was supported by AFOSR, ARO, DARPA, DOE, NGA and ONR.

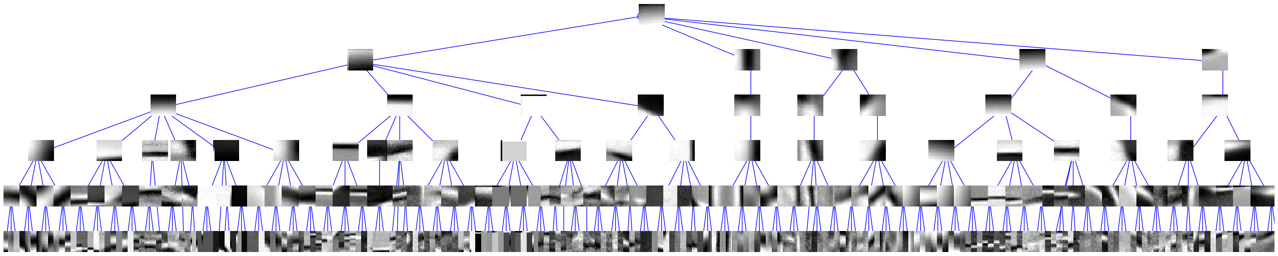

Figure 4. Tree-structured hierarchy(with top six layers) embedded with dictionaries learned from 100,000 patches of size 16×16 pixels.

References

Adams, R., Ghahramani, Z., and Jordan, M. Tree-structured stick breaking for hierarchical data. In *NIPS 23*. 2010.

Archambeau, C. and Bach, F. Sparse probabilistic projections. In *NIPS 21*. 2009.

Bart, E., Porteous, I., Perona, P., and Welling, M. Unsupervised learning of visual taxonomies. In *CVPR*, 2008.

Bhattacharya, A. and Dunson, D. B. Sparse bayesian infinite factor models. *Biometrika*, 2010.

Blei, D., Griffiths, T. L., Jordan, M. I., and Tenenbaum, J. B. Hierarchical topic models and the nested chinese restaurant process. In *NIPS 16*. 2004.

Blei, D. M., Griffiths, T. L., and Jordan, M. I. The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies. *Journal of the ACM*, 57 (2):1–30, 2010.

Chen, M., Silva, J., Paisley, J., Wang, C., Dunson, D., and Carin, L. Compressive sensing on manifolds using a nonparametric mixture of factor analyzers: Algorithm and performance bounds. *IEEE Transactions on Signal Processing*, 58(12):6140–6156, 2010.

Griffiths, T. and Ghahramani, Z. Infinite latent feature models and the indian buffet process. In *NIPS 18*. 2006.

Jenatton, R., Mairal, J., Obozinski, G., and Bach, F. Proximal methods for sparse hierarchical dictionary learning. In *Proceedings of the 27th International Conference on Machine Learning*, pp. 487–494, 2010.

Knowles, D. and Ghahramani, Z. Infinite sparse factor analysis and infinite independent components analysis. In *ICA*, pp. 381–388, 2007.

Li, L., Wang, C., Lim, Y., Blei, D., and Fei-Fei, L. Building and using a semantivisual image hierarchy. In *CVPR*, 2010.

Mauldin, R. D., Sudderth, W. D., and Williams, S. C. Polya trees and random distributions. *The Annals of Statistics*, 20(3):pp. 1203–1221, 1992.

Paisley, J. and Carin, L. Nonparametric factor analysis with beta process priors. In *Proceedings of the 26th International Conference on Machine Learning*, pp. 777–784, 2009.

Rai, P. and Daumé, H. The infinite hierarchical factor regression model. In *NIPS*, 2008.

Rasmussen, C. E. The infinite gaussian mixture model. In *NIPS 12*, 1999.

Roberts, G. O. and Rosenthal, J. S. Coupling and ergodicity of adaptive mcmc. *Journal of Applied Probability*, 44: 458–475, 2007.

Rosenberg, N. A., Pritchard, J. K., Weber, James L., Cann, Howard M., Kidd, Kenneth K., Zhivotovsky, L. A., and Feldman, M. W. Genetic Structure of Human Populations. *Science*, 298:2381–2385, 2002.

Roweis, S. and Ghahramani, Z. A unifying review of linear gaussian models. *Neural Comput.*, 11:305–345, February 1999. ISSN 0899-7667.

Teh, Y. W., Jordan, M. I., Beal, M. J., and Blei, D. M. Hierarchical dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581, 2006.

Tenenbaum, J. B., Silva, V., and Langford, J. C. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290:2319–2323, 2000.

Tipping, M. E. and Bishop, C. M. Mixtures of probabilistic principal component analysers. *Neural Computation*, 11 (2):443–482, 1999.

Wang, C. and Blei, D. Variational inference for the nested chinese restaurant process. In *NIPS*. 2009.

Zou, H., Hastie, T., and Tibshirani, R. Sparse Principal Component Analysis. *Journal of Computational and Graphical Statistics*, 15, 2004.