Sticky Hidden Markov Modeling of Comparative Genomic Hybridization

[1]Lan Du, [1]Minhua Chen, [2]Joseph Lucas and [1]Lawrence Carin
[1]Department of Electrical and Computer Engineering
[2]Center for Applied Genomics and Technology
Duke University
Durham, NC, 27708
{ld53,minhua.chen,lcarin}@ece.duke.edu, joe@stat.duke.edu

Abstract

We develop a sticky hidden Markov model (HMM) with a Dirichlet distribution (DD) prior, motivated by the problem of analyzing comparative genomic hybridization (CGH) data. As formulated the sticky DD-HMM prior is employed to infer the number of states in an HMM, while also imposing state persistence. The form of the proposed hierarchical model allows efficient variational Bayesian (VB) inference, of interest for large-scale CGH problems. We compare alternative formulations of the sticky HMM, while also examining the relative efficacy of VB and Markov chain Monte Carlo (MCMC) inference. To validate the formulation, example results are presented for an illustrative synthesized data set, and for speaker diarization from audio data (the first problem class for which the sticky HMM was developed). Our main application is CGH, for which we consider data for breast cancer. For the latter, we also make comparisons and partially validate the CGH analysis through factor analysis of associated (but distinct) gene-expression data.

Index Terms

Infinite hidden Markov model, Hierarchical Bayesian modeling, Variational Bayesian, Multi-task Learning, DNA copy number

I. Introduction

Comparative genomic hybridization (CGH) yields data consisting of fluorescence intensity ratios of test and reference DNA samples. The intensity ratios provide information about the number of DNA copies in localized regions of a chromosome. Array CGH data analysis has recently attracted increasing interest in both the biology and statistics communities. Specifically, there is a growing need for algorithms that can automatically identify gains and losses in number

of copies, and relate this to disease and illness.

A number of well-known methods strive to fulfill this need. For example, [1], [18], [19], [21], [22] use segmentation to identify chromosomal segments with altered copy number. A variation of a binary segmentation method [21], called circular binary segmentation (CBS), segments the CGH data in each chromosome and computes the within-segment means. An edge filter is applied in [19] to detect segments. Since there is also a clear dependence among the intensity ratios of neighboring clones, [18] performs smoothing using the signs of neighboring data values, inspecting the width and magnitude of the segments to detect regions of copy number change. A disadvantage of such methods is that they cannot directly detect gains or losses. A recent paper by [29] applies penalized matrix decomposition (PMD) for selecting important "clones" in array CGH data. Nevertheless, the sequential information is not explicitly exploited in such a matrix decomposition method, and the biological meaning of the selected "clones" are not assigned via this method.

To make use of the physical dependence of the nearby fragments or "clones", the hidden Markov model (HMM) has been utilized by [10], [13] to analyze array CGH data, of which [10] uses a traditional HMM employing Baum-Welch EM learning; [13] assigns biological meaning to the latent states and implements a Bayesian HMM via an Metropolis-within-Gibbs algorithm. The number of states must be preset in these two papers. However, this can lead to over- or under-fitting if the underlying state structure is not modeled correctly.

Bayesian approaches have been investigated to automatically infer the number of states in an HMM using Markov chain Monte Carlo (MCMC) [12] and reversible jumps [5], as well as a nonparametric, infinite-state model that utilizes the hierarchical Dirichlet process (HDP) [26]. The latter method has proven effective in many applications [20]. To impose state persistence, [9] proposes a *sticky* extension of HDP-HMM, allowing more robust learning of smoothly varying dynamics. However, the lack of conjugacy between the two levels of Dirichlet processes and the delta function in the sticky HDP-HMM [9] prohibit fast variational inference [2], [17], making this approach computationally prohibitive when modeling very large data sets (of interest for our CGH data).

Motivated by previous work, we develop a simplified form of the sticky HDP-HMM [9], called the sticky hidden Markov model with Dirichlet distribution prior (sticky DD-HMM), and extend the new model structure to analyze array CGH data in all chromosomes for multiple samples.

Inference is performed efficiently via a variational Bayesian (VB) analysis [2], [17].

To validate the sticky HMM formulation, we first present example results on synthesized data that is generated by a sticky-HMM, and real data associated with speaker diarization based on audio data. The diarization problem was the first motivating problem for the sticky HMM [9], and therefore it is also briefly considered here for the proposed new sticky-HMM formulation. Another motivation for this example is that it has a clear definition of "truth", aiding model validation (this example also demonstrates that the proposed model is applicable to problems beyond CGH analysis). The third data set we consider constitutes the motivating application of this paper. Specifically, we consider CGH data associated with breast cancer. While there is no explicit "truth" for this problem, the sticky-HMM results based on CGH data are compared to complementary results manifested on associated gene-expression data. For that data we employ a completely distinct modeling paradigm, based on factor-analysis formulations. Specifically, we consider sparse Bayesian factor analysis [7] and a non-Bayesian penalized matrix decomposition [29]. While these separate analyses on gene-expression data do not explicitly validate our CGH analysis, there is a strong suggestion of biological correspondence.

The reminder of the paper is organized as follows. In Section II we review the learning of HMMs with Dirichlet priors. In Section III we introduce the proposed sticky HMM for a single data sequence, and make connections with related models. The proposed model is extended for array CGH data analysis in Section IV, developing a "multi-task" model. The variational Bayesian (VB) inference method is discussed in Section V. We demonstrate model performance in Section VI, and conclude in Section VII.

II. Review of Hidden Markov Model with Dirichlet Priors

A. Hidden Markov Model

The hidden Markov model (HMM) [24] is a generative statistical representation of sequential data, with an underlying discrete Markovian process selecting state-dependent distributions from which observations are drawn. Specifically, for a sequence of length T, an underlying "hidden" state sequence $\boldsymbol{S} = (s_1, s_2, \cdots, s_T)$ is drawn from $p(s_t|s_{t-1}, \cdots, s_1) = p(s_t|s_{t-1})$. The observed sequence $\boldsymbol{X} = (x_1, x_2, \cdots, x_T)$ is drawn as $f(\theta_{s_t})$, where $f(\cdot)$ represents the observation model, and θ_{s_t} is the set of parameters for the model indexed by the state at time t, s_t. Note that, given the underlying states, the observations at each time are conditionally independent.

Traditionally, the number of states associated with an HMM is initialized and fixed [24]. A J-state HMM can be modeled as $\Omega = \{\boldsymbol{W}, \boldsymbol{\theta}, \boldsymbol{w}_0\}$, where $\boldsymbol{W}$ is a $J \times J$ matrix with entry w_{ij} representing the transition probability from state i to j; $\boldsymbol{\theta} = \{\theta_1, \theta_2, \cdots, \theta_J\}$ where θ_j represents the observation-model parameters associated with state j; and $\boldsymbol{w}_0$ is a J-dimensional probability vector defining the probability of being in each of the J states when performing the first observation.

The data-generating process may be represented as

$$\begin{aligned} x_t &\sim f(\theta_{s_t}) \; ; \quad t = 1, \cdots, T \\ s_t &\sim \begin{cases} \boldsymbol{w}_0 = [w_{0,1}, w_{0,2}, \cdots, w_{0,J}] & \text{if } t = 1 \\ \boldsymbol{w}_{s_{t-1}} = [w_{s_{t-1},1}, w_{s_{t-1},2}, \cdots, w_{s_{t-1},J}] & \text{if } t > 1 \end{cases} \end{aligned} \tag{1}$$

For given Ω, the joint probability of the observation and the underlying state sequence is expressed as

$$p(\boldsymbol{X}, \boldsymbol{S} | \boldsymbol{W}, \boldsymbol{\theta}, \boldsymbol{w}_0) = w_{0,s_1} \prod_{t=1}^{T-1} w_{s_t, s_{t+1}} \prod_{t=1}^{T} p(x_t | \theta_{s_t}) \tag{2}$$

The data likelihood $p(\boldsymbol{X} | \boldsymbol{W}, \boldsymbol{\theta}, \boldsymbol{w}_0)$ can be obtained by integrating over the states using the forward algorithm [24].

B. HMM with Dirichlet Distribution prior

The priors associated with $\boldsymbol{w}_0$ and the rows of $\boldsymbol{W}$ are typically Dirichlet distributions, since these are conjugate to the multinomial likelihood. The standard Dirichlet distribution is written as

$$p(w_1, \cdots, w_J | \alpha_1, \cdots, \alpha_J) = \frac{\Gamma(\sum_{j=1}^{J} \alpha_j)}{\prod_{j=1}^{J} \Gamma(\alpha_j)} \prod_{j=1}^{J} w_j^{\alpha_j - 1} \tag{3}$$

with the mean and variance of an element, w_j, represented as

$$\mathbb{E}[w_j] = \frac{\alpha_j}{\sum_{j=1}^{J} \alpha_j} \; , \quad \mathbb{V}[w_j] = \frac{\alpha_j (\sum_{j=1}^{J} \alpha_j - \alpha_j)}{(\sum_{j=1}^{J} \alpha_j)^2 (\sum_{j=1}^{J} \alpha_j + 1)} \tag{4}$$

To understand the properties of a draw from the Dirichlet distribution (DD), recall the "stick-breaking" representation [25] for the draw $\boldsymbol{w} \sim \text{Dir}(\alpha_1, \ldots, \alpha_J)$. We define $\alpha = \sum_{j=1}^{J} \alpha_j$, and $\boldsymbol{g}_0$ is a base probability vector with jth element $g_{0j} = \alpha_j / \alpha$. A draw $\boldsymbol{w} \sim \text{Dir}(\alpha_1, \ldots, \alpha_J)$ may

be constructed as

$$
\begin{aligned}
w_i &= \sum_{k=1}^{\infty} h_k \delta(Z_k = i) \ ; \quad i = 1, \cdots, J \\
h_k &= V_k \prod_{\tau=1}^{k-1} (1 - V_\tau) \ , \quad V_k = \text{Beta}(1, \alpha) \ ; \quad k = 1, \cdots, \infty \\
Z_k &\sim \boldsymbol{g}_0 \ ; \quad k = 1, \cdots, \infty
\end{aligned} \tag{5}
$$

where $\delta(Z_k = i)$ equals to one if $Z_k = i$, and its zero otherwise. Hence $\boldsymbol{w}$ is built up as a sum of probability vectors with all zeros and a single randomly selected one (defined by draws from $\boldsymbol{g}_0$), with these probability vectors multiplied by the stick weights h_k. Note that if α is small then the draws from Beta$(1, \alpha)$ are such that only a relatively small number of sticks h_k will have significant weight, and hence with high probability a draw $\boldsymbol{w}$ will only possess a relatively small number of components with significant mass (for large J). To simplify notation below, the infinite-dimensional probability vector $\boldsymbol{h}$ constructed as above is denoted $\boldsymbol{h} \sim \text{Stick}(\alpha)$.

C. Infinite HMM with HDP prior

The above discussion motivates drawing $\boldsymbol{w}_0$ and the rows of $\boldsymbol{W}$ from Dir$(\alpha/J, \ldots, \alpha/J)$, and setting J large. By setting α we place a prior on the number of anticipated states (via (5)), and setting J large one may uncover the number of states required by the data. This idea has motivated considering the limit $J \to \infty$, yielding the *infinite* hidden Markov model (iHMM) [3]. It was subsequently shown [26] that the iHMM can be recast as a special case of the hierarchical Dirichlet process (HDP).

A draw from a Dirichlet *process* (DP) may also be represented in stick-breaking form [14], [25]

$$
G = \sum_{j=1}^{\infty} v_j \delta_{\theta_j} \ , \quad \boldsymbol{v} \sim \text{Stick}(\gamma) \ , \quad \theta_j \sim G_0 \tag{6}
$$

where δ_{θ_j} is a point measure concentrated at θ_j (each θ_j is termed an atom). Such a draw is denoted $G \sim \text{DP}(\gamma, G_0)$.

The DP is commonly used as a prior on the parameters of a mixture model with unknown number of mixture components (see Figure 1(a)). This sampling process is often described via a discrete indicator variable $s_n \sim \boldsymbol{v}$, indicating which atom generates $x_n \sim f(\theta_{s_n})$.

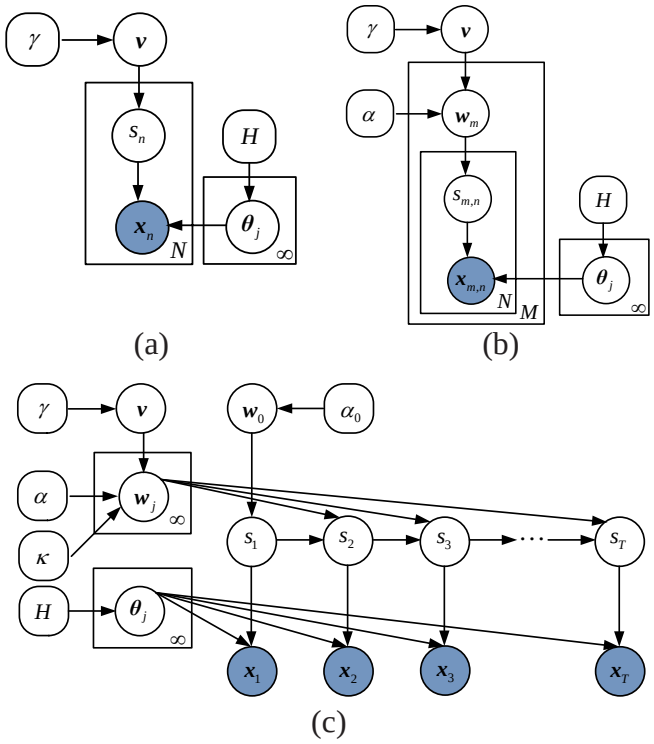


Fig. 1. (a) DPMM in which $\boldsymbol{v} \sim \text{Stick}(\gamma)$, $\theta_j \sim H$, $s_n \sim \boldsymbol{v}$, and $x_n \sim f(\theta_{s_n})$. (b) HDP mixture model with $\boldsymbol{v} \sim \text{Stick}(\gamma)$, $\boldsymbol{w}_m \sim \text{DP}(\alpha, \boldsymbol{v})$, $\theta_j \sim H$, $s_{m,n} \sim \boldsymbol{w}_m$, and $x_{m,n} \sim f(\theta_{s_{m,n}})$. (c) Sticky HDP-HMM where the state evolves as $s_{t+1} \sim \boldsymbol{w}_{s_t}$, $\boldsymbol{w}_j \sim \text{DP}(\alpha + \kappa, (\alpha \boldsymbol{v} + \kappa \delta_j)/(\alpha + \kappa))$, $\boldsymbol{v} \sim \text{Stick}(\gamma)$, and $x_t \sim f(\theta_{s_t})$. The original HDP-HMM has $\kappa = 0$.

The HDP [26] extends the DP to cases in which groups of data are produced by related, yet unique, generative processes. In the HDP structure, the base probability measure, G_0, is itself drawn from a Dirichlet process. The formal notation is as follows,

$$G_m \sim \text{DP}(\alpha, G_0) \ , \quad G_0 \sim \text{DP}(\gamma, H) \tag{7}$$

where G_m represents the prior distribution associated with group m. The HDP is a two-level model, where the distribution on the atoms is shifted from the continuous H to the discrete (but countably infinite) G_0. An alternative representation of the model is

$$G_m = \sum_{j=1}^{\infty} w_{m,j} \delta_{\theta_j} \ , \quad \boldsymbol{w}_m \sim \text{DP}(\alpha, \boldsymbol{v}) \ , \quad G_0 = \sum_{j=1}^{\infty} v_j \delta_{\theta_j} \ , \quad \boldsymbol{v} \sim \text{Stick}(\gamma) \ , \quad \theta_j \sim H \tag{8}$$

Observation $x_{m,n}$ is associated with one of the global set of discrete parameters via an indicator random variable $s_{m,n} \sim \boldsymbol{w}_m$; see Figure 1(b).

The HDP can be used to develop an HMM with an unknown state space [26]. For this HDP-HMM, each HDP group-specific distribution, $\boldsymbol{w}_m$, is a state-specific transition distribution and, due to the infinite state space, there are infinite many groups. Let s_t denote the state of the Markov chain at time t. As is done typically, $s_1 \sim \boldsymbol{w}_0$. For $t > 1$, we have the Markov process $s_t \sim \boldsymbol{w}_{s_{t-1}}$, so that s_{t-1} indexes the group to which x_t is assigned. The current HMM state s_t then indexes the parameter θ_{s_t} used to generate observation x_t (See Figure Figure 1(c)). According to (8), the HDP formulation effectively selects the number of states and their observation parameters via the top-level DP and uses the mixing weights as the prior for a second-level Dirichlet distribution from which the transition probabilities are drawn. Importantly, since G_0 is composed of a discrete set of atoms, the state-dependent probabilities are shared across the different $\boldsymbol{w}_m$. The lack of conjugacy between the two levels in the model, however, means that a truly variational solution [2], [17] does not exist.

D. Sticky HMM with HDP Prior

In the HDP-HMM construction, by sampling $\boldsymbol{w}_j \sim \text{DP}(\alpha, v)$, the HDP encourages states to have a similar transition distribution ($\mathbb{E}[w_{j,i}] = v_i$). However, it does not differentiate self-transitions from moves between states. In many applications one would like to be able to incorporate prior knowledge that slow, smoothly varying dynamics are probable ($i.e.$, that it is likely to stay in the same state for prolonged time periods). When modeling systems with state persistence, traditional HMM design may lead to many redundant (essentially duplicate) states into which transitions occur, with the effect of manifesting a persistence in the observation statistics. However, such models impede our ability to identify a single dynamical model which best explains the observations (it undermines interpretability). Therefore, [9] proposed to instead sample transition distributions $\boldsymbol{w}_j$ as follows:

$$\boldsymbol{w}_j \sim \text{DP}(\alpha + \kappa, \frac{\alpha \boldsymbol{v} + \kappa \delta_j}{\alpha + \kappa}) \tag{9}$$

Here, $(\alpha v + \kappa \delta_j)$ indicates that an amount $\kappa > 0$ is added to the jth component of $\alpha \boldsymbol{v}$. When $\kappa = 0$ the original HDP-HMM is recovered. Because positive κ values increase the prior probability $\mathbb{E}(w_{j,j})$ of self-transitions, [9] referred this extension as the *sticky* HDP-HMM (the model favors "sticking" in the same state for prolong periods).

The inference algorithm is simplified if we introduce the auxiliary random variable z_t as follows:

$$z_t \sim \text{Ber}(\frac{\alpha}{\alpha+\kappa}) \ , \quad s_t \sim \begin{cases} \boldsymbol{w}_0 & \text{if } t=1 \\ \delta_{s_{t-1}} & \text{if } t>1 \text{ and } z_t=0 \\ \boldsymbol{v} & \text{if } t>1 \text{ and } z_t=1 \end{cases} \tag{10}$$

where $\text{Ber}(\cdot)$ represents the Bernoulli distribution. In practice, the gamma prior and beta prior are respectively put on $\alpha+\kappa$ and $\kappa/(\alpha+\kappa)$, which allows the degree of self-transition bias to be strongly influenced by the statistics of observed data, as desired. Due to the delta function and the HDP structure, the proposed sticky model was implemented by a slice sampler in [9]. However, such MCMC [12] inference may be impractical computationally when considering a large dataset of sequential data.

III. Sticky HMM with Dirichlet Distribution Prior

A. Model Construction

We seek a simplified implementation of the sticky HDP-HMM, affording the opportunity to avoid MCMC. Specifically, the proposed hidden Markov model, termed the sticky DD-HMM, is represented as

$$\begin{aligned}
&x_t \sim f(\theta_{s_t}) \ ; \quad t=1,\cdots,T \\
&s_t \sim \begin{cases} \boldsymbol{w}_0 & \text{if } t=1 \\ \boldsymbol{u}_{s_{t-1}} & \text{if } t>1 \text{ and } z_t=0 \\ \boldsymbol{w}_{s_{t-1}} & \text{if } t>1 \text{ and } z_t=1 \end{cases} \ ; \quad t=1,\cdots,T \\
&z_t \sim \text{Ber}(\beta_{s_{t-1}}) \ ; \quad t=2,\cdots,T \\
&\beta_j \sim \text{Beta}(c_0,d_0) \ ; \quad j=1,\cdots,J \\
&\boldsymbol{w}_0 \sim \text{Dir}(\alpha_0/J,\cdots,\alpha_0/J) \\
&\boldsymbol{w}_j \sim \text{Dir}(\alpha/J,\cdots,\alpha/J) \ ; \quad j=1,\cdots,J \\
&\boldsymbol{u}_j \sim \text{Dir}(\gamma_{j,1},\cdots,\gamma_{j,J}) \ , \quad \gamma_{j,j} \gg \gamma_{j,j'} \ , \quad j' \neq j \ ; \quad j=1,\cdots,J \\
&\theta_j \sim H \ ; \quad j=1,\cdots,J
\end{aligned} \tag{11}$$

A graphical representation of this model is depicted in Figure 2. When $c_0/(c_0+d_0) \rightarrow 0$ the original DD-HMM described in Section II-B is approximated. For future notational convenience, below we represent

$$\begin{aligned} s_t &\sim \begin{cases} \boldsymbol{w}_0 & \text{if } t=1 \\ \boldsymbol{u}_{s_{t-1}} & \text{if } t>1 \text{ and } z_t=0 \\ \boldsymbol{w}_{s_{t-1}} & \text{if } t>1 \text{ and } z_t=1 \end{cases} ; \quad t=1,\cdots,T \\ z_t &\sim \text{Ber}(\beta_{s_{t-1}}) \; ; \quad t=2,\cdots,T \end{aligned} \tag{12}$$

as $s_t \sim$ S-HMM$(\{\boldsymbol{w}_j,\boldsymbol{u}_j,\beta_j\}_{j=1}^J,\boldsymbol{w}_0,s_{t-1})$.

Recall from (5) that draws $\boldsymbol{w} \sim \text{Dir}(\alpha/J,\cdots,\alpha/J)$ will have components with an appreciable number of components defined by Stick(α), and hence a relatively small α setting will allow inference on which subset of J possible states are actually needed based on the data (J may be set large). Further, from (5), a draw $\boldsymbol{u}_j \sim \text{Dir}(\gamma_{j,1},\ldots,\gamma_{j,J})$ for $\gamma_{jj} \gg \gamma_{j,j'}$ for $j' \neq j$ will favor $\boldsymbol{u}_j$ having large probability mass at component j and small mass for all $j' \neq j$ (we refer to this as an approximation to a point measure – delta function – at point j).

In (11) the parameter β_j controls the degree of "stickiness" for state j ($i.e.$, the probability of staying in state j), and this is inferred by the data (with a beta prior imposed). The state-dependent β_j therefore plays the role of $\alpha/(\alpha+\kappa)$ in (10), and can be different for different states in this model. Further, the $\boldsymbol{u}_j$ in (11) play the role of the delta function in (10). Therefore, the model in (11) has many of the characteristics of the original sticky iHMM, but it yields simplified inference, as discussed further below.

B. Relationship between sticky DD-HMM and sticky HDP-HMM

Comparing the sticky HDP-HMM in [9] to the proposed sticky DD-HMM, there are two main modifications: *i*) using the DD prior to replace the HDP prior; *ii*) using a special DD construction to approximate the delta function in (10).

1) DD vs HDP: In HDP structure as shown in (8), the draw $\boldsymbol{w}_j \sim \text{DP}(\alpha,\boldsymbol{v})$ may be presented in stick-breaking form, with the ith element of $\boldsymbol{w}_j$ construct as $w_{j,i}=\sum_{k=1}^{\infty} h_{j,k}\delta(Z_{j,k}=i)$, with $\boldsymbol{h}_j \sim \text{Stick}(\alpha)$ and $Z_{j,k} \sim \boldsymbol{v}$. We may also truncate the draw $\boldsymbol{v} \sim \text{Stick}(\gamma)$ to J sticks (denoted $\boldsymbol{v}_J \sim \text{Stick}_J(\gamma)$), for large J [14]. Note that for the HMM transition matrix $\boldsymbol{W}$ is square, and therefore we consider J rows. Using these representations, the truncated HDP construction in

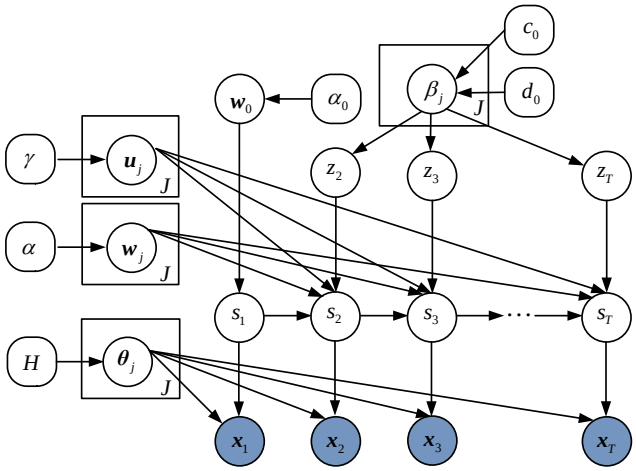


Fig. 2. Graph of the sticky DD-HMM. The detailed generative process is described in (11).

HDP-HMM may be represented as

$$
\begin{aligned}
& w_{j,i} = \sum_{k=1}^{\infty} h_{j,k} \delta(Z_{j,k} = i) \ ; \quad i = 1, \cdots, J \ ; \quad j = 1, \cdots, J \\
& \boldsymbol{h}_j \sim \text{Stick}(\alpha) \ ; \quad j = 1, \cdots, J \\
& Z_{j,k} \sim \boldsymbol{v} \ ; \quad k = 1, \cdots, \infty \ ; \quad j = 1, \cdots, J \\
& \boldsymbol{v}_J \sim \text{Stick}_J(\gamma)
\end{aligned} \tag{13}
$$

Note that we truncate Stick(γ) to J sticks [14], but do *not* truncate Stick(α). In other words, although the transition matrix $\boldsymbol{W}$ is truncated, the $\boldsymbol{h}$ vector is still an infinite vector.

By comparing (13) to (5) we observe that the difference between the truncated HDP-HMM (with truncation only for Stick$_J(\gamma)$) is that in the HDP-HMM the atoms are drawn from $\boldsymbol{v}_J$, which is drawn from Stick(γ); in the DD-HMM, $\boldsymbol{v}_J$ is essentially fixed as $(1/J, \ldots, 1/J)$ (Here $\boldsymbol{g}_0 = [1/J, \ldots, 1/J]^{\mathrm{T}}$). It is felt that this is a very modest difference between the models, with the DD-HMM construction having the advantage of simplified inference (particularly, variational Bayesian inference [2] is tractable, of interest for large-scale problems).

The proposed DD-HMM is non-parametric, in that setting a large J allows the model to infer the proper number of states from the data, analogous to studies of the truncated stick-breaking representation [14]. Setting a large J does not imply that we believe that there are actually J states, since from (5) only a relatively small set of components in $\boldsymbol{w}_j$ will have appreciable

amplitude (the same type of motivation for the stick-breaking view of DP and HDP). We also emphasize that the stick-breaking representation of a draw from a Dirichlet distribution has been introduced above to make the connection between the proposed model and a truncated representation of HDP-HMM. However, when actually performing inference, it is often simpler to just draw directly from $\text{Dir}(\alpha/J, \cdots, \alpha/J)$.

2) Special DD construction and relationship to delta function: To impose state persistence, [9] introduced a binary switch variable $z_t \sim \text{Ber}(\frac{\alpha}{\alpha+\kappa})$ to control the state indicator s_t. If $z_t = 0$, $s_t = \delta_{s_{t-1}}$; otherwise, s_t is drawn from $\boldsymbol{w}_{s_{t-1}}$. However, such a prior cannot be directly implemented by variational Bayesian (VB) inference [2], [17].

In (11), we use a discrete distribution vector $\boldsymbol{u}_{s_{t-1}} \sim \text{Dir}(\gamma_{s_{t-1},1}, \cdots, \gamma_{s_{t-1},J})$ with $\gamma_{s_{t-1},s_{t-1}} \gg \gamma_{s_{t-1},j}$ and $j \neq s_{t-1}$ to sample the state indicator s_t if $z_t = 0$. According to (4), with such special hyper-parameters, $\mathbb{E}[u_{s_{t-1},s_{t-1}}] \to 1$ and $\mathbb{V}[u_{s_{t-1},s_{t-1}}] \to 0$, therefore, a draw $s_t \sim \boldsymbol{u}_{s_{t-1}}$ will likely be s_{t-1}. In this manner we approximate the point measure $\delta_{s_{t-1}}$.

IV. Multi-Task Analysis for CGH Data

For the motivating CGH problem of interest here, one typically has access to data from all 23 chromosomes. We wish to learn a sticky HMM for each of these chromosomes, and recognize that there is likely statistical inter-relationships between the chromosomes that may be exploited. We therefore wish to learn sticky HMMs for data sets $m = 1, \ldots, M$, for $M = 23$, and the data from chromosome m is termed learning task m. The learning of sticky HMMs for all $M = 23$ tasks jointly is referred to as multi-task learning [6]. We here extend the discussion in the previous sections to sticky HMM learning in a multi-task setting. Similar MTL techniques have been successfully applied to information retrieval [4] and computer vision [27], as well as music (sequential data) analysis [20]. For the CGH problem of interest, since the CGH data in one chromosome are limited, rather than building HMMs for each task (chromosome) separately, it is desirable to appropriately share information ("strength") across the $M = 23$.

In the proposed MTL model, each of the $M = 23$ chromosomes is assumed to have unique state-transition statistics, but the state-dependent observation statistics are shared across the M tasks. This implies that the different chromosomes share the same underlying states, but the state-dependent transition probabilities are chromosome independent. The state-dependent observations for the CGH data are assumed to be drawn from a Gaussian model [13], and the overall model

is summarized as

$$
\begin{aligned}
&x_t \sim \text{Norm}(\mu_{s_t^{(m,l)}}, \sigma^{-1}_{s_t^{(m,l)}}) \ ; \quad t = 1, \cdots, T_m \ ; \quad m = 1, \cdots, 23 \ ; \quad l = 1, \cdots, L \\
&s_t^{(m,l)} \sim \text{S-HMM}(\{\boldsymbol{w}_j^{(m)}, \boldsymbol{u}_j^{(m)}, \beta_j\}_{j=1}^{J}, \boldsymbol{w}_0^{(m)}, s_{t-1}^{(m,l)}) \ ; \\
&\qquad\qquad t = 1, \cdots, T_m \ ; \quad m = 1, \cdots, 23 \ ; \quad l = 1, \cdots, L \\
&\boldsymbol{w}_0^{(m)} \sim \text{Dir}(\alpha_0/J, \cdots, \alpha_0/J) \ ; \quad m = 1, \cdots, 23 \\
&\boldsymbol{w}_j^{(m)} \sim \text{Dir}(\alpha/J, \cdots, \alpha/J) \ ; \quad j = 1, \cdots, J \ ; \quad m = 1, \cdots, 23 \\
&\boldsymbol{u}_j^{(m)} \sim \text{Dir}(\gamma_{j,1}, \cdots, \gamma_{j,J}) \ , \quad \gamma_{j,j} \gg \gamma_{j,j'} \ , \quad j' \neq j \ ; \quad j = 1, \cdots, J \ ; \quad m = 1, \cdots, 23 \\
&\beta_j \sim \text{Beta}(c_0, d_0) \ ; \quad j = 1, \cdots, J \\
&(\mu_j, \sigma_j) \sim \begin{cases} \delta_0 \times \text{Ga}(b^{(0)}, \lambda^{(0)}) & \text{if } j = 1 \\ \text{Norm}(r^{(1)}, 1/(t^{(1)}\sigma_j)) - \text{Ga}(b^{(1)}, \lambda^{(1)}) & \text{if } j > 1 \end{cases} \ ; \quad j = 1, \cdots, J
\end{aligned} \tag{14}
$$

As discussed above, each chromosome has its own state-transition statistics, defined by $\boldsymbol{w}_0^{(m)}$, $\{\boldsymbol{w}_j^{(m)}\}_{j=1,J}$ and $\{\boldsymbol{u}_j^{(m)}\}_{j=1,J}$ with chromosome $m \in \{1, \ldots, 23\}$. The state-dependent "stickiness", defined by β_j for state j, is shared across the 23 chromosomes, as are the observation statistics defined by (μ_j, σ_j). Note that state $j = 1$ has an imposed mean of zero, and this corresponds to the no/low copy number state. In this model, we learn DD-HMMs with independent state transition matrixes for each of the tasks (chromosomes) as well as share the same state set across all tasks.

V. Variational Bayesian Inference

Bayesian inference seeks to estimate the posterior distribution of the latent variables $\boldsymbol{\Psi}$, given the observed data $\boldsymbol{X}$ and hyper-parameters $\boldsymbol{\Upsilon}$:

$$
p(\boldsymbol{\Psi}|\boldsymbol{X}, \boldsymbol{\Upsilon}) = \frac{p(\boldsymbol{X}|\boldsymbol{\Psi}, \boldsymbol{\Upsilon}) p(\boldsymbol{\Psi}|\boldsymbol{\Upsilon})}{\int p(\boldsymbol{X}|\boldsymbol{\Psi}, \boldsymbol{\Upsilon}) p(\boldsymbol{\Psi}|\boldsymbol{\Upsilon}) \mathrm{d}\boldsymbol{\Psi}} \tag{15}
$$

where the denominator $\int p(\boldsymbol{X}|\boldsymbol{\Psi}, \boldsymbol{\Upsilon}) p(\boldsymbol{\Psi}|\boldsymbol{\Upsilon}) \mathrm{d}\boldsymbol{\Psi}$ is the model evidence (marginal likelihood). Here we employ variational Bayesian (VB) [2], [17] inference as a compromise between accuracy and efficiency (the detailed reason for selecting VB will be discussed further in Section VI). VB inference seeks a variational distribution $q(\boldsymbol{\Psi})$ to approximate the true posterior distribution of

the latent variables $p(\boldsymbol{\Psi})$. The expression

$$\log p(\boldsymbol{X}|\boldsymbol{\Upsilon}) = \mathcal{L}(q(\boldsymbol{\Psi})) + \mathcal{KL}(q(\boldsymbol{\Psi}) \parallel p(\boldsymbol{\Psi}|\boldsymbol{X}, \boldsymbol{\Upsilon})) \tag{16}$$

with

$$\mathcal{L}(q(\boldsymbol{\Psi})) = \int q(\boldsymbol{\Psi}) \log \frac{p(\boldsymbol{X}|\boldsymbol{\Psi}, \boldsymbol{\Upsilon}) p(\boldsymbol{\Psi}|\boldsymbol{\Upsilon})}{q(\boldsymbol{\Psi})} \mathrm{d}\boldsymbol{\Psi} \tag{17}$$

forms a lower bound for $\log p(\boldsymbol{X}|\boldsymbol{\Upsilon})$. Accordingly, the goal of minimizing the KL divergence between the variational distribution and the true posterior reduces to adjusting $\boldsymbol{\Psi}$ to maximize (17).

VB inference [2], [17] assumes a factorized $q(\boldsymbol{\Psi})$, *i.e.* $q(\boldsymbol{\Psi}) = \prod_k q_k(\Psi_k)$, typically with the same form as employed in $p(\boldsymbol{\Psi}|\boldsymbol{X}, \boldsymbol{\Upsilon})$. The mean-field variational distribution for the model described in (14) is,

$$\begin{aligned} q(\boldsymbol{\Psi}) = & [\prod_{m=1}^{M} q(\boldsymbol{w}_0^{(m)})][\prod_{m=1}^{M}\prod_{j=1}^{J} q(\boldsymbol{u}_j^{(m)})][\prod_{m=1}^{M}\prod_{j=1}^{J} q(\boldsymbol{w}_j^{(m)})][\prod_{j=1}^{J} q(\beta_j)] \\ & \cdot[\prod_{m=1}^{M}\prod_{l=1}^{L} q(s_1^{(m,l)}) \prod_{t=2}^{T_m} q(s_t^{(m,l)})][\prod_{m=1}^{M}\prod_{l=1}^{L}\prod_{t=2}^{T_m} q(z_t^{(m,l)})][q(\sigma_1)\prod_{j=2}^{J} q(\mu_j, \sigma_j)] \end{aligned} \tag{18}$$

A general method for performing variational inference for conjugate-exponential Bayesian networks outlined in [28] is as follows: For a given node in a graph, write out the posterior as though everything were known, take the logarithm, the expectation with respect to all unknown parameters and exponentiate the result. Since it requires computational resources comparable to the expectation-maximization (EM) algorithm, variational inference is fast relative to MCMC methods [12].

The update equations for the variational posteriors are listed as follows:

- $q(\boldsymbol{w}_0^{(m)}) = \text{Dir}(\tilde{\boldsymbol{\alpha}}_0)$; where $\tilde{\alpha}_{0,i} = \alpha_0/J + < s_{1,i}^{(m,\cdot)} >$, with $< s_{1,i}^{(m,\cdot)} >$ denoting the expected number of state indicators $\{s_1^{(m,l)}\}_{l=1}^{L}$ with outcome i, for $i = 1, \cdots, J$.
- $q(\boldsymbol{u}_j^{(m)}) = \text{Dir}(\tilde{\boldsymbol{\gamma}}_j)$; where $\tilde{\gamma}_{j,i} = \sum_{l=1}^{L}\sum_{t=1}^{T_m-1} < s_{t,j}^{(m,l)} >< s_{t+1,i}^{(m,l)} >< z_{t,0}^{(m,l)} >$, with $< z_{t,0}^{(m,l)} >$ denoting the expected number of binary switch indicator $z_t^{(m,l)}$ with outcome 0.
- $q(\boldsymbol{w}_j^{(m)}) = \text{Dir}(\tilde{\boldsymbol{\alpha}}_j)$; where $\tilde{\alpha}_{j,i} = \alpha/J + \sum_{l=1}^{L}\sum_{t=1}^{T_m-1} < s_{t,j}^{(m,l)} >< s_{t+1,i}^{(m,l)} >< z_{t,1}^{(m,l)} >$.
- $q(\beta_j) = \text{Beta}(\tilde{c}_j, \tilde{d}_j)$; where $\tilde{c}_j = c_0 + \sum_{m=1}^{M}\sum_{l=1}^{L}\sum_{t=2}^{T_m} < s_{t-1,j}^{(m,l)} >< z_{t,1}^{(m,l)} >$ and $\tilde{d}_j = d_0 + \sum_{m=1}^{M}\sum_{l=1}^{L}\sum_{t=2}^{T_m} < s_{t-1,j}^{(m,l)} >< z_{t,0}^{(m,l)} >$.

- $q(\boldsymbol{s}^{(m,l)}) \propto \exp\Big[< \log w^{(m)}_{0,s_1^{(m,l)}} > + [\sum_{t=1}^{T_m-1} < z^{(m,l)}_{t+1,0} >< \log u^{(m)}_{s_t^{(m,l)},s_{t+1}^{(m,l)}} >] + [\sum_{t=1}^{T_m-1} < z^{(m,l)}_{t+1,1} >< \log w^{(m)}_{s_t^{(m,l)},s_{t+1}^{(m,l)}} >][\sum_{t=1}^{T_m} < \log p(x_t^{(m,l)}|\mu_{s_t^{(m,l)}}, \sigma_{s_t^{(m,l)}}) >]\Big]$; where the detailed expressions for $< \log w^{(m)}_{i',j} >$, $< \log u^{(m)}_{i,j} >$ and $< \log p(x_t^{(m,l)}|\mu_j, \sigma_j) >$ with $i' = 0, \cdots, J$ and $i, j = 1, \cdots, J$ can be referred to [16], [23].
- $q(z_t^{(m,l)} = 0) \propto \exp\Big[< \log(1 - \beta_{s_t^{(m,l)}}) > + < \log u^{(m)}_{s_{t-1}^{(m,l)},s_t^{(m,l)}} > \Big]$, and $q(z_t^{(m,l)} = 1) \propto \exp\Big[< \log \beta_{s_t^{(m,l)}} > + < \log w^{(m)}_{s_{t-1}^{(m,l)},s_t^{(m,l)}} > \Big]$, for $t = 2, \cdots, T_m$; where $< \log(1 - \beta_j) >= \psi(\tilde{d}_j) - \psi(\tilde{c}_j + \tilde{d}_j)$ and $< \log \beta_j >= \psi(\tilde{c}_j) - \psi(\tilde{c}_j + \tilde{d}_j)$ with $j = 1, \cdots, J$.
- $q(\sigma_1) = \text{Ga}(\tilde{b}_1, \tilde{\lambda}_1)$, and $q(\mu_j, \sigma_j) = \text{Normal}(\tilde{r}_j, 1/(\tilde{t}_j\sigma_j)) - \text{Ga}(\tilde{b}_j, \tilde{\lambda}_j)$ for $j = 2, \cdots, J$; where the detailed expressions for $\tilde{r}_i$,$\tilde{t}_i$,$\tilde{b}_j$ and $\tilde{\lambda}_j$ with $i = 2, \cdots, J$ and $j = 1, \cdots, J$ can be referred to [16].

VI. Experimental Results

We present experimental results on three problems: illustrative synthetic data; audio diarization, motivated by the first application of the sticky HMM [9]; and analysis of comparative genomic hybridization (CGH) data for breast cancer. The second example allows examination of model performance on a real problem for which there is "truth". The CGH problem is the principal motivating application of this paper, and comparisons are made to use of more traditional HMMs for this problem [13]. We also partially validate the sticky-HMM CGH results through a Bayesian factor analysis [7] of accompanying gene-expression data.

A. Synthetic Data

We synthesized data from the following sticky HMM:

$$\boldsymbol{W} = \begin{bmatrix} 0 & 1/2 & 1/2 \\ 1/2 & 0 & 1/2 \\ 1/2 & 1/2 & 0 \end{bmatrix}, \quad \boldsymbol{w}_0 = \begin{bmatrix} 1/3 \\ 1/3 \\ 1/3 \end{bmatrix}, \quad \boldsymbol{\beta} = \begin{bmatrix} 0.05 \\ 0.03 \\ 0.03 \end{bmatrix}$$
$$\mu_1 = 0\ , \quad \mu_2 = 60\ , \quad \mu_3 = -60\ , \quad \sigma_1 = \sigma_2 = \sigma_3 = 0.01 \tag{19}$$

From this model we generated a sequence of length $T = 400$. The generated observation sequence is shown in Figure 3(a). Note that this example is very similar to the simulation example shown in [9].

To apply the DD-HMM and sticky DD-HMM models on this data, we set the truncation level as $J = 10$. We place Beta$(0.1, 0.9)$ priors on each β_j, Dir$(\boldsymbol{\gamma}_j)$ with $\gamma_{j,j} = 1$ and $\gamma_{j,j'} = 10^{-3}$ for $j' \neq j$ priors on each $\boldsymbol{u}_j$, and Dir$([10^{-3}/J, \cdots, 10^{-3}/J])$ priors on $\boldsymbol{w}_0$ and on each $\boldsymbol{w}_j$. All the above hyper-parameters have not been optimized or tuned.

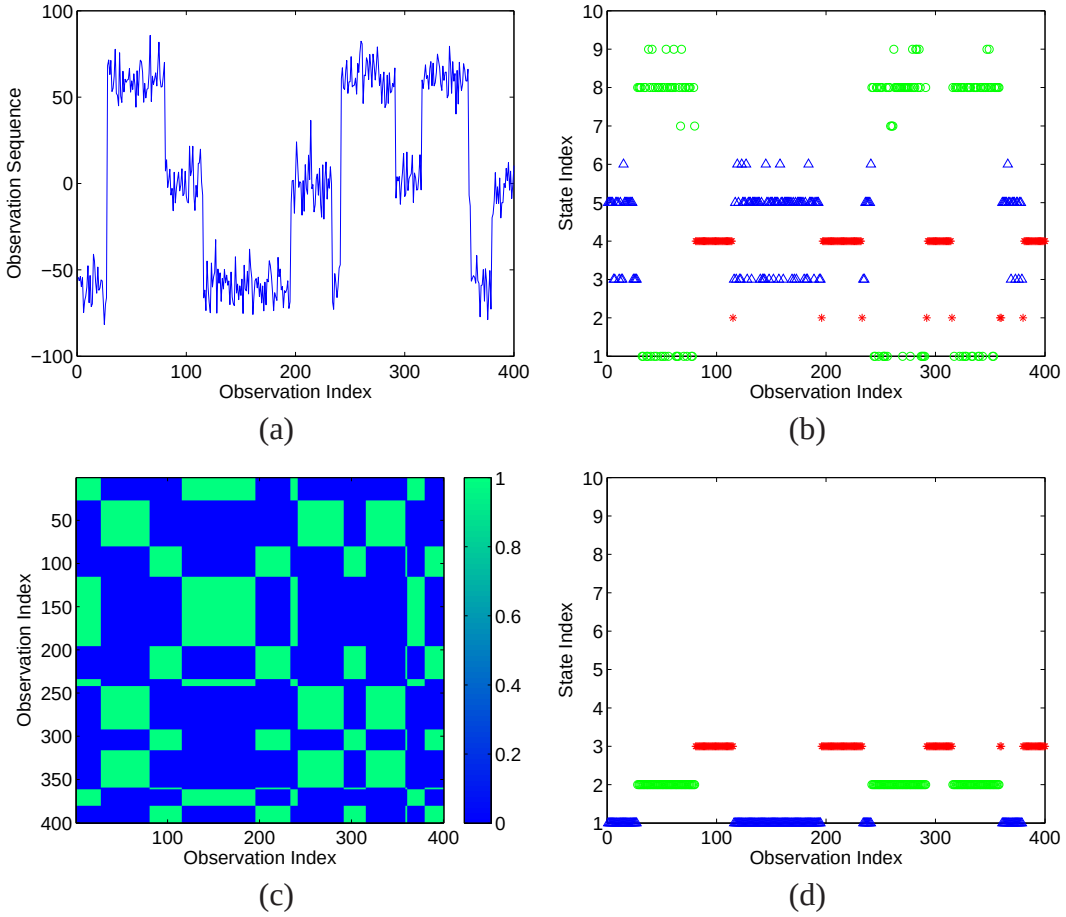


Fig. 3. Synthetic example. (a) Generated observation sequence, (b) DD-HMM using VB inference, (c) sticky HMM using MCMC inference, (d) sticky DD-HMM using VB inference. The colored symbols denote ground truth: red $*$ represents the state with $\mu_1 = 0$, green $\circ$ represents the state with $\mu_2 = 60$, blue $\triangle$ represents the state with $\mu_3 = -60$. Each observation is assigned to a state index ($J = 10$), with the index shown along the vertical axis.

When employing MCMC inference, atoms vary across the collected samples from the posterior [15], therefore, we cannot get an overall state label decision based on all collected samples; using VB inference, atoms are fixed in the posterior computation, and we obtain a posterior distribution on s_t, *i.e.* $< s_{t,j} >$ for $j = 1, \cdots, J$, and approximate the membership for each measurement by assigning it to the state with largest probability. Therefore, besides the relatively fast computation, another advantage of VB inference is the avoidance of the "label-switching" problem associated

with MCMC [15]. Figures 3(b) and (d) present the "hard" (most likely) decisions employed to provide state labels (the Bayesian analysis can also yield a "soft" decision in terms of a full posterior distribution) for the generated sequential data via DD-HMM and sticky DD-HMM using VB inference. To indicate the ground truth, different symbols and colors are used to represent different states in the generated observation. In Figure 3(c) is shown the fraction of times within the collection samples that a given portion of the signal share the same underlying state. As demonstrated in Figure 3(b), without an extra self-transition bias, the original DD-HMM using VB inference rapidly transitions among redundant states. Although not shown here for brevity, the DD-HMM using MCMC has similar behavior. By contrast, from the results in Figures 3(c) and (d), both MCMC and VB results via sticky HMM are in agreement with ground truth, which infer a proper number of states and correct membership of each observation.

All experiments presented here have been performed in non-optimized software written in Matlab, on a Pentium PC with 1.73 GHz CPU and 4G RAM. The above MCMC results were computed using 5000 burn-in iterations and 5000 collection iterations, which took about 2 hours; VB results converged after about 50 VB iterations, which required less than 10 minutes.

B. Audio Data

Provided with a spoken document consisting of multiple speakers, speaker diarization is the process of segmenting the audio signal into contiguous temporal regions, and within a given region a particular individual is speaking [9]. Further, one also wishes to group all temporal regions in which a specific individual is speaking. The total number of speakers is unknown in advance, and should be inferred from the data.

Here we consider identification of different speakers from a recording of broadcast news, which may be downloaded with its ground truth[1]. The spoken document has a length of 122.05 seconds, and consists of three speakers. Figure 4(a) presents the audio waveform with a sampling rate of 16000 Hz. The ground truth indicates that Speaker 1 talked within the first 13.77 seconds, followed by Speaker 2 until the 59.66 second, then Speaker 1 began to talk again until 74.15 seconds, and Speaker 3 followed and speaks until the end.

[1]http://www.itl.nist.gov/iad/mig/tests/rt/2002/index.html

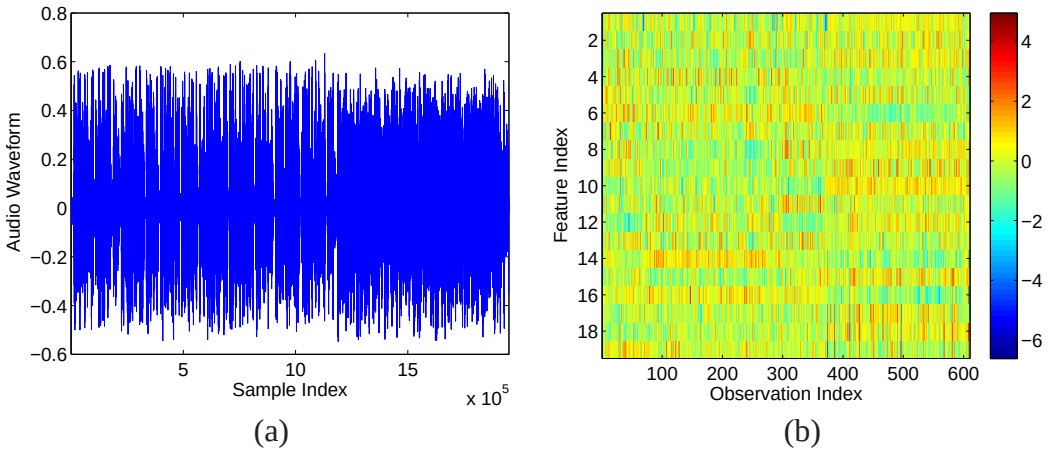


Fig. 4. Audio data under consideration. (a) Original audio waveform, (b) representation in terms of MFCC features.

For the feature vector, we computed the first 19 Mel Frequency Cepstral Coefficients (MFCCs) [11] over a 30 ms window every 20 ms, and defined the observations as averages over every 200 ms block, without overlap. We used the first 19 MFCCs because the high frequency content of these features contained little discriminative information. The software that we used to extract the MFCCs feature can be downloaded online[2]. There are 610 feature vectors in total, shown in Figure 4(b); the features are normalized to zero mean and the standard deviation is made equal to one. In the following experiment, we used multivariate Gaussian (not univariate Gaussian) to characterize the feature vectors, and put the corresponding multivariate Gaussian and Wishart priors on the mean and precision parameters.

The hyperparameters are set as in Section VI-A. The results in Figure 5 are shown in the same style as that of Figure 3, where different symbols and colors in Figure 5(a) and (c) represent different speakers; and Figure 5(b) shows the similarity matrix across all observations. Figure 5 demonstrates that the sticky HMM implemented using MCMC and VB inference yield comparably good segmentation performance, with results in close agreement with ground truth. The experiments were performed on the same PC mentioned in Section VI-A. The MCMC results with 5000 burn-in iterations and 5000 collection iterations took nearly 3 hours; while VB results with 50 VB iterations just took about 10 minutes. Note that the form of the sticky HMM developed here is motivated by but distinct from that in [9], for which a distinct form of

[2]http://www.ee.columbia.edu/∼dpwe/resources/matlab/rastamat/

MCMC inference was employed (VB analysis was not performed in [9]).

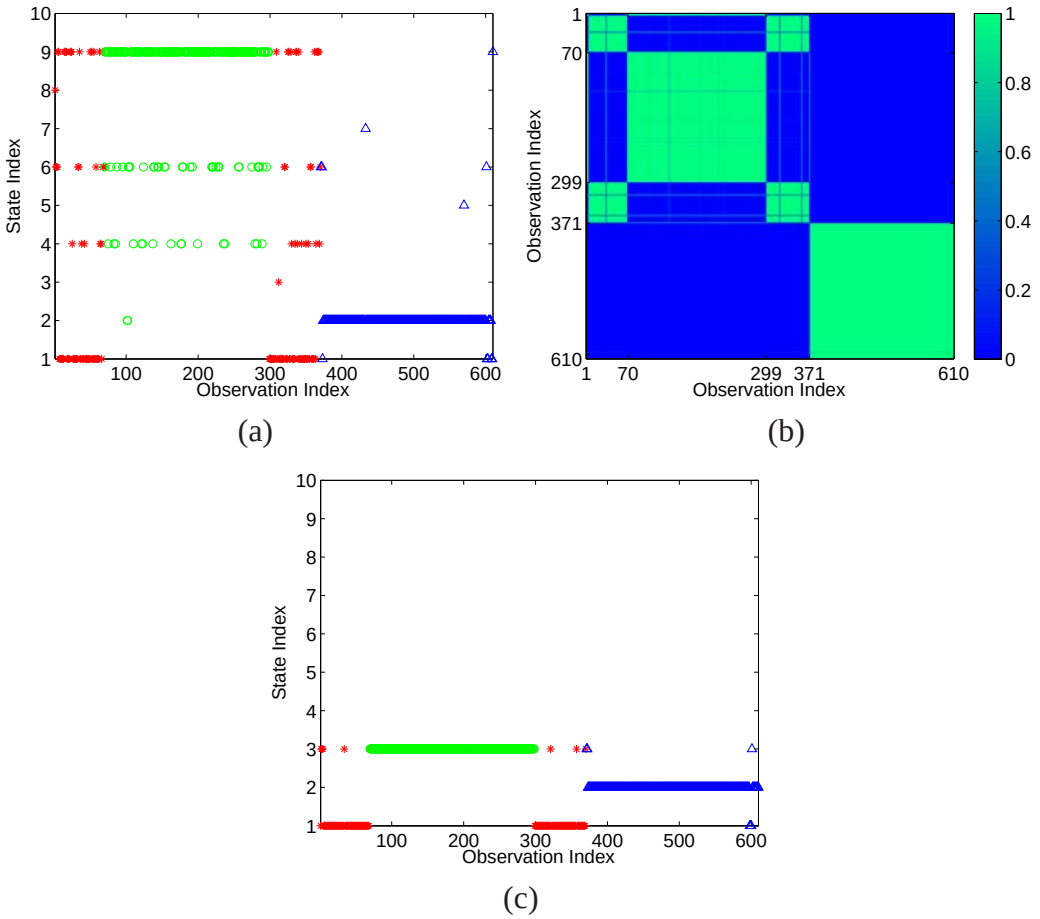


Fig. 5. Segmentation results for the audio recording. The colored symbols in (a) and (c) denote ground truth: red $*$ represents Speaker 1, green $\circ$ represents Speaker 2, blue $\triangle$ represents Speaker 3. Each MFCC feature vector is assigned to a state index ($J = 10$), with the index shown along the vertical axis. (b) denotes the fraction of times within the collection samples that a given portion of the observation sequence shares the same underlying state. (a) DD-HMM using VB inference, (b) Sticky HMM using MCMC inference, (c) Sticky DD-HMM using VB inference.

C. CGH Data

We examine the performance of the sticky DD-HMM on a breast cancer data set described in [8] and available online[3]. The breast cancer data is composed of $L = 89$ tissue samples for which both array CGH and gene expression measurements are available. There are in total 22215 gene-expression measurements and 2149 CGH measurements. We apply the proposed sticky DD-HMM to analyze the CGH data, and make comparisons to the original DD-HMM and the

[3]http://icbp.lbl.gov/breastcancer/

Bayesian HMM proposed in [13], for which the codes are included in the Bioinformatics toolbox 3.3 of Matlab software[4]. As a comparison/validation of the CGH data, the corresponding gene expression data are analyzed via two factor-analysis formulations: sparse Bayesian factor analysis (SBFA) [7] and the related (non-Bayesian) Penalized Matrix Decomposition (PMD) method [29]. All Bayesian results are presented using VB inference; for the size of this problem (with data from all chromosomes analyzed jointly) MCMC inference is very expensive computationally.

In the following experiments we set the state truncation level to $J = 15$ (similar results were found for larger truncations). The hyper-parameters for the sticky DD-HMM are: $\alpha_0 = \alpha = 10^{-3}$; $\gamma_{j,j} = 1$ and $\gamma_{j,j'} = 10^{-3}$ for $j' \neq j$; $c_0 = 0.1$ and $d_0 = 0.9$; $b^{(0)} = 1.5$ and $\lambda^{(0)} = 0.015$; $r^{(1)}$ is set to be the mean of all CGH observations, $t^{(1)} = 0.01$, $b^{(1)} = 1.5$ and $\lambda^{(1)} = 1.5t^{(1)}$. All these hyper-parameters have not been optimized or tuned, and the results are relatively invariant to "reasonable" perturbation of these parameters.

The proposed model learns the posterior state-transition matrices of each chromosome $\boldsymbol{w}^{(m)}$ for $m = 1, \cdots, 23$ simultaneously, and in so doing infers an estimate of the proper number of states (using the multi-task framework of Section IV). As shown in Figure 6, although we initialized the truncation level to $J = 15$, for the CGH data only four states are inferred. As indicated in the sticky DD-HMM described in (14), for the first state $\mu_1 = 0$ (corresponding to no or a low level of copying); the inferred posterior means of μ_i for $i = 2, 3, 4$ are $\tilde{r}_2 = -0.3361$, $\tilde{r}_3 = 0.2937$ and $\tilde{r}_4 = 0.6024$. The second state corresponds to a copy-number reduction state, the third state corresponds to a small copy-number increase, and the fourth state corresponds to marked copy-number amplification. By contrast, the original DD-HMM, without a constraint on the first state and without stickiness, inferred 7 states, of which the posterior means are $\tilde{r}_1 = 0.0063$, $\tilde{r}_2 = -0.1714$, $\tilde{r}_3 = 0.2116$, $\tilde{r}_4 = 0.3712$, $\tilde{r}_5 = -0.2532$, $\tilde{r}_6 = 0.5534$ and $\tilde{r}_7 = -0.3651$. Thus three states corresponds to copy number loss state, and two corresponds to small copy number gain state, as computed via DD-HMM (constituting the state redundancy required to manifest stickiness in the observation statistics). The *inferred* four states of the sticky-HMM is consistent with the biologically motivated and *imposed* four states employed in [13].

In CGH data analysis, we desire the assignment of state labels to each CGH fragment, and wish to detect the copy loss or gain based on the state labels. Here we employed the Bayesian

[4]http://www.mathworks.com/products/bioinfo/demos.html

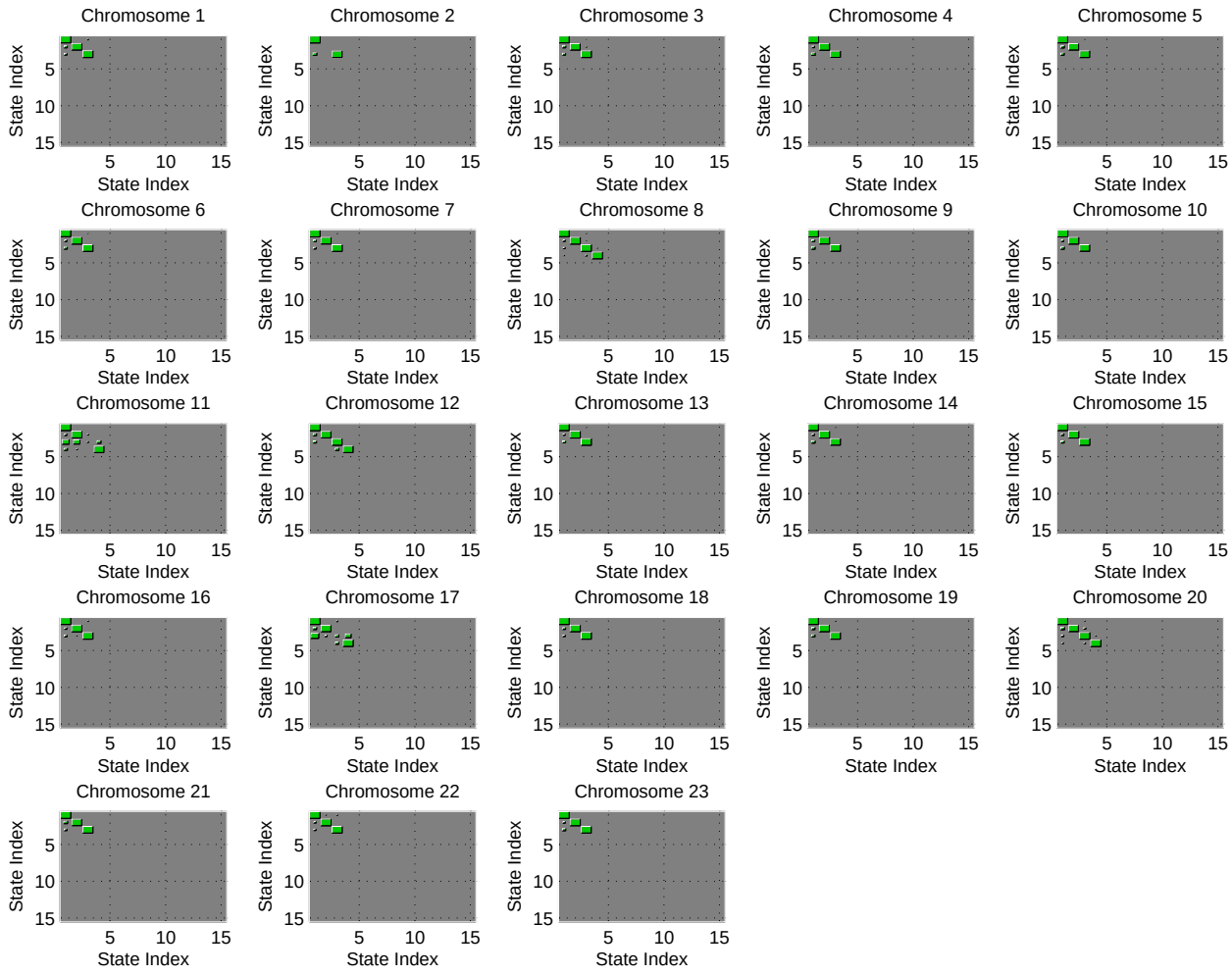


Fig. 6. Mean posterior state-transition matrices of each chromosome, $\boldsymbol{w}^{(m)}$ with $m = 1, \cdots, 23$, inferred via the sticky DD-HMM. The plots are meant to visualize matrices, where the green box reflects the relative probability strength (where there is no green square, the probability is zero). Four meaningful states are inferred, and in the model a state truncation level of $J = 15$ was employed.

HMM [13], DD-HMM and sticky DD-HMM to compare state labeling performance. In [13] the authors explicitly imposed four states, for which the mean parameter μ_i for $i = 1, 2, 3, 4$ has the constraint $\mu_1 < \mu_2 < \mu_3 < \mu_4$. The priors for the means are: $\mu_1 \sim \text{Norm}(-1, \tau_1^2) \cdot \delta(\mu_1 < -\epsilon)$, $\mu_2 \sim \text{Norm}(0, \tau_2^2) \cdot \delta(-\epsilon < \mu_2 < \epsilon)$, $\mu_3 \sim \text{Norm}(0.58, \tau_3^2) \cdot \delta(\epsilon < \mu_3 < 0.58)$, and $[\mu_4 | \mu_3, \sigma_3^{-1}] \sim \text{Norm}(1, \tau_4^2) \cdot \delta(< \mu_4 > \mu_3 + 3\sigma_3)$, where the parameter ϵ needs to be set in [13], and the results may be sensitive to how ϵ is set. Such a prior structure also introduces some difficulties for inference, and the computationally expensive Metropolis-Hastings (MH) method [12] is used within Gibbs sampling [13]. In addition, [13] learns independent HMMs for the CGH data in

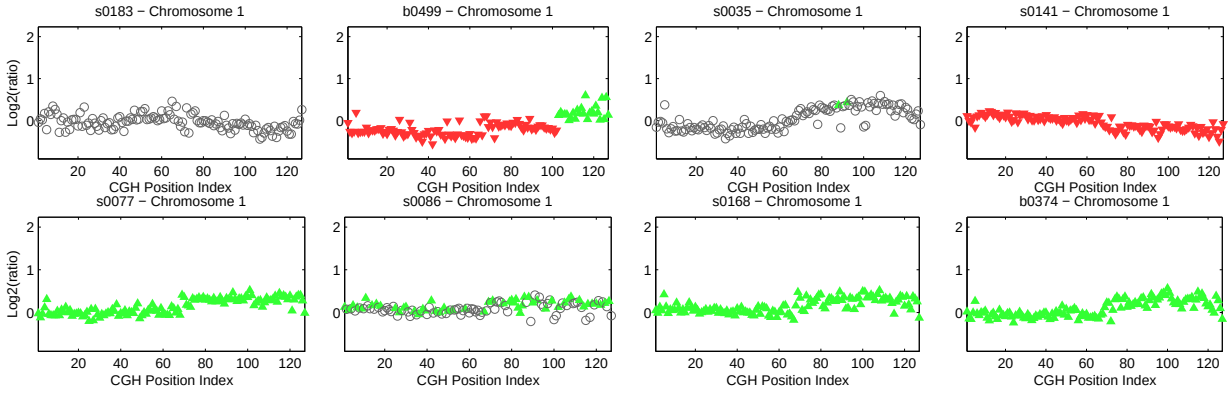
each chromosome separately; the MTL approach developed in Section IV was not considered in [13]. Since the four states are fixed, MCMC results via the Bayesian HMM [13] yield state labels based on collected samples; for the other two models considered (DD-HMM and sticky DD-HMM), for which VB inference is employed, a "hard" (most likely) decision is employed to label the hidden states, as discussed in Section VI-A. Figure 7 presents the state labels for example array-CGH profiles of samples from chromosome 1, as inferred by the three models. The horizontal axis in these plots denotes the index of the DNA fragments or "clones".

From Figure 7 we make the following *subjective* observations, with more quantitative results presented below. The model developed in [13], corresponding to (a) in Figure 7, *a priori* imposes four states. It appears that this model is not particularly discriminating in the underlying states across the chromosome. For example, consider sample s0035 (third from left, top row). Although the CGH value appears to noticeably increase with CGH position index (across the horizontal axis), almost all CGH values associated with this chromosome and sample are assigned to the same state. Similar phenomenon is exhibited on other samples from this chromosome (shown in the figure), and from other chromosomes (not shown here, for brevity).

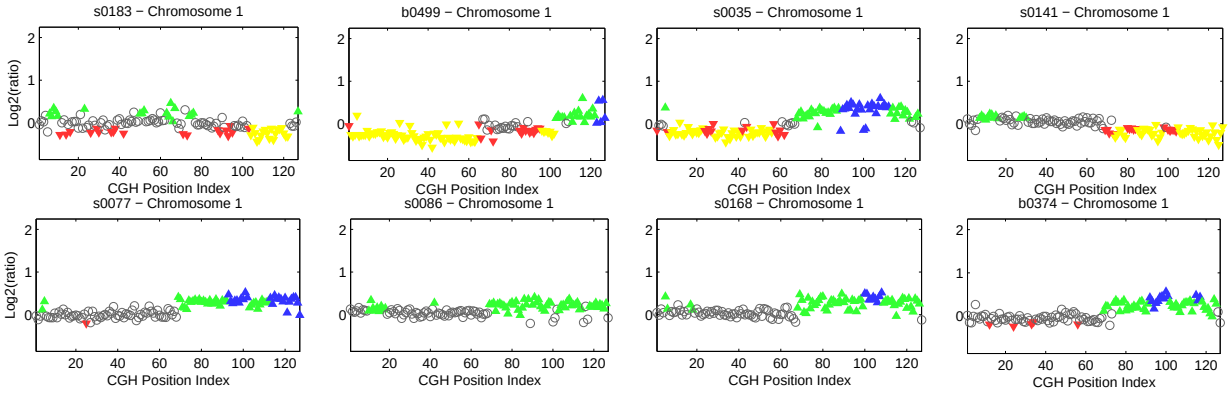
By contrast, the DD-HMM model, corresponding to (b) in Figure 7, is far more discriminating in the underlying states (more state diversity, in general, as a function of position index). However, because this model is not sticky, there is quick changing of states even where the CGH values are not changing significantly in strength ($e.g.$, larger position indices for sample s0077, the left-most sample in the bottom row). This phenomenon appears to manifest redundant (superfluous) states to mitigate the non-stickiness, thereby undermining model interpretability. This has motivated the proposed sticky HMM, and the large scale of this problem has motivated a formulation that admits a VB solution.

The proposed sticky HMM results are shown in (c) of Figure 7. Note that this model yields a more diverse state usage than the model considered in (a), but the inferred states appear to manifest the desired stickiness. For example, sample b0499 (second from left, top row) manifests clear regions defined by three different states, contiguously partitioned as a function of position index. The states appear to capture well the relative CGH intensity. Returning to sample s0077, note that all the CGH values beyond index 70 along the horizontal are assigned to a single state (unlike the results in Figure 7 (b)).

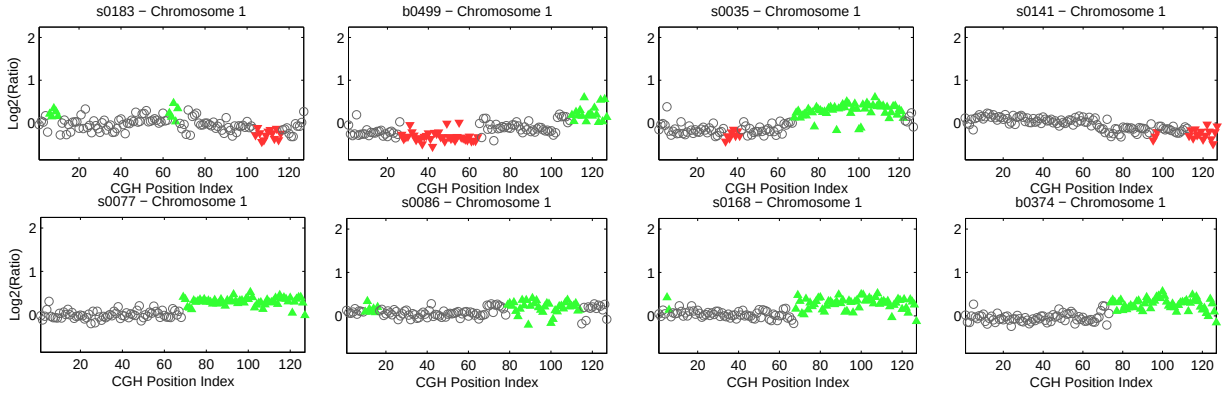
There is no explicit "truth" available for the CGH data, and therefore we use an associated but



(a)



(b)



(c)

Fig. 7. State labels for example array-CGH profiles of chromosome 1. (a) Bayesian HMM [13], (b) DD-HMM, (c) Sticky DD-HMM. The colored symbols denote different states: the normal state is represented via black ○ for the three models; one copy loss state inferred via Bayesian HMM and sticky DD-HMM, represented by red ▼, and two inferred via DD-HMM, represented by red and yellow ▼; one small copy gain state inferred via Bayesian HMM and sticky DD-HMM, represented by green ▲, and two inferred via DD-HMM, represented by green and blue ▲.

independent data set based on gene-expression values on the same samples. Further, to analyze these gene-expression data we employ two established methods: a Bayesian sparse factor analysis (FA) model [7] and the related (but non-Bayesian) PMD method developed in [29].

The gene-expression data $\boldsymbol{Y} \in \mathbb{R}^{N\times L}$ is normalized to zero mean in each row, where each row represents a separate gene, and the columns correspond to different samples. The FA model seeks to factorize the data matrix into the form $\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{S} + \boldsymbol{E}$, where $\boldsymbol{A} \in \mathbb{R}^{N\times K}$ is the factor loading matrix, $\boldsymbol{S} \in \mathbb{R}^{K\times L}$ is the factor matrix, each row being a factor. Here N is the number of genes and K is the number of factors. Typically one simply sets K [7], with $K \ll N$; $\boldsymbol{E} \in \mathbb{R}^{N\times L}$ is the error/noise matrix, addressing those aspects of $\boldsymbol{Y}$ not captured in the factors. For the data considered $N = 22215$ and $L = 89$.

For gene-expression data analysis, one usually imposes sparseness priors on the factor loading matrix, shrinking most of the elements to be near zero. A "spike-slab" sparseness construction is used in [7]. As discussed in [7], since factor loading $\boldsymbol{A}$ is "sparse", which means many of the elements of $\boldsymbol{A}$ are close to zero, each column ideally will represent a particular biological "pathway", composed of a relatively small number of relevant genes related to a given latent factor, which correspond to those having factor loadings not close to zero. In contrast, PMD employs a non-Bayesian method to achieve the same goal. In the PMD method, the matrix $\boldsymbol{Y}$ is approximated as $\hat{\boldsymbol{Y}} = \sum_{k=1}^{K} \xi_k \boldsymbol{U}_k \boldsymbol{\Lambda}_k$ by minimizing $\|\boldsymbol{Y} - \hat{\boldsymbol{Y}}\|_{\mathrm{F}}^2$ subject to sparseness penalties on $\boldsymbol{U}_k$ and $\boldsymbol{\Lambda}_k$. When the PMD is applied using an ℓ_1 penalty on $\boldsymbol{U}_k$ but not on $\boldsymbol{\Lambda}_k$, a method for sparse principle components results. In this way, $\xi_k \boldsymbol{U}_k$ corresponds to factor loading of factor k, and $\boldsymbol{\Lambda}_k$ corresponds to factor score.

In the two factor models, the (sparse) factor-loading matrix $\boldsymbol{A}$ yields the (typically relatively small) subset of genes associated with a given "pathway", and these genes may be linked to regions on the 23 chromosomes. The objective is to examine whether the genes responsible for the inferred pathways reside at portions of the chromosomes at which "interesting" activity (raised or lowered level of copying) is revealed via the aforementioned CGH analysis.

Figure 8 presents results in which the gene-expression analysis was performed via FA [7] (very similar gene-expression results were found using PMD [29] instead of FA, and are therefore omitted for brevity). The top two figures display posterior probabilities of each CGH position selecting the normal state (State 1). Specifically, the average probabilities across all samples

are $\frac{1}{L}\sum_{l=1}^{L} < s_{t,1}^{(m,l)} >$, and the sigmoid transformation is $\frac{1}{1+\exp[-2(\sum_{l=1}^{L} < s_{t,1}^{(m,l)} > - \frac{1}{L}\sum_{l=1}^{L} < s_{t,1}^{(m,l)} >)]}$; $< s_{t,1}^{(m,l)} >$ represents the posterior expectation of state indicator $s_t^{(m,l)}$ with outcome 1, and parameter $L = 89$ denotes the total number of samples. The lower the probability of a CGH being in State 1, presumably the more important the CGH measurements are, since the CGH fragments with copy number losses or gains might be relevant to the breast cancer associated with these data. We observe that the positions of informative (non-State 1) CGH data and inferred important genes are consistent. Specifically, those regions of the chromosome that have a low probability of being in State 1 are also regions (generally) for which the associated genes contribute to the important factors (in Figure 8 we plot the factor loadings with associated significant factor scores, these providing the principal description of the gene-expression data).

These results appear to suggest that CGH copy numbers that are inconsistent with typical behavior (*not* in State 1) are indicative of a portion of the chromosome that are linked to the illness under study, here breast cancer. Note that the ability to explicitly localize State 1 is therefore important, with this a challenge for the *non*-sticky HMM, since redundant states may be manifested, undermining interpretability of the results (see Figure 7(b)).

The experiments have been performed on the same computational platform mentioned in Section VI-A. One VB run of the sticky DD-HMM, for 60 VB iterations, required about 4 hours for the whole CGH dataset (processing all chromosomes simultaneously). Typically 50 VB iterations are required to achieve convergence. All results are based on a single VB run, with random initialization. It only required about 20 seconds for each Bayesian HMM [13] MCMC run when considering a single CGH profile of one chromosome; however, all runs for all 23 chromosomes required more than 11 hours. In addition to the computational challenges associated with Bayesian HMM [13], each of the chromosomes is analyzed in isolation, and therefore multi-task learning is not implemented (the state statistics are not explicitly shared across chromosomes). Further, the results in Figure 7 suggest inferior underlying state inference (which may be attributable to the lack of multi-task analysis).

VII. Conclusion

We have developed a hidden Markov model (HMM) with state persistence, termed the sticky HMM with Dirichlet distribution prior (sticky DD-HMM). The new model is motivated by [9], but the proposed construction allows convenient VB inference, of interest for the large-

scale motivating CGH problem. For array CGH data analysis, we further extended traditional single-task HMM to multi-task learning (MTL), where here the multiple tasks are linked to specific chromosomes. The proposed multi-task model allows simple VB inference, yielding fast computation times and efficient detection of copy losses and gains. The algorithm has been demonstrated on synthetic data, real audio signal and real CGH data. The CGH results are partially validated by a corresponding gene analysis using factor models [7], [29]. The sticky DD-HMM extends the Bayesian HMM for application to CGH [13], in that stickiness is explicitly imposed, and the number of underlying states is inferred from the data.

REFERENCES

[1] A. J. Aguirre, C. Brennan, G. Bailey, R. Sinha, B. Feng, C. Leo, Y. Zhang, J. Zhang, J. D. Gans, N. Bardeesy, C. Cauwels, C. Cordon-Cardo, M. S. Redston, R. A. DePinho, and L. Chin, "High-resolution characterization of the pancreatic adenocarcinoma genome," *Proceedings of the National Academy of Sciences USA*, vol. 101, pp. 9067–9072, 2004.

[2] M. J. Beal, "Variational algorithms for approximate Bayesian inference," Ph.D. dissertation, Gatsby Computational Neuroscience Unit, University College London, 2003.

[3] M. J. Beal, Z. Ghahramani, and C. Rasmussen, "The infinite hidden Markov model," in *Advances in Neural Information Processing Systems (NIPS)*, 2002.

[4] D. M. Blei, T. L. Griffiths, and J. B. Tenenbaum, "Hierarchical topic models and the nested Chinese restaurant process," in *Advances in Neural Information Processing Systems (NIPS)*, 2004.

[5] R. J. Boys and D. A. Henderson, "A comparison of reversible jump mcmc algorithm for dna sequence segmentation using hidden markov models," *Computing Science and Statistics*, vol. 33, p. 3549, 2001.

[6] R. Caruana, "Multitask learning," *Machine Learning*, vol. 28, pp. 41–75, 1997.

[7] C. Carvalho, J. Chang, J. Lucas, J. Nevins, Q. Wang, and M. West., "High-dimensional sparse factor modeling: applications in gene expression genomics," *Journal of the American Statistical Association*, vol. 103(484), pp. 1438–1456, 2008.

[8] K. Chin, S. DeVries, J. Fridlyand, P. Spellman, R. Roydasgupta, W.-L. Kuo, A. Lapuk, R. Neve, Z. Qian, T. Ryder, F. Chen, H. Feiler, T. Tokuyasu, C. Kingsley, S. Dairkee, Z. Meng, K. Chew, D. Pinkel, A. Jian, B. Ljung, L. Esserman, D. Albertson, F. Waldman, and J. Gray, "Genomic and transcriptional aberrations linked to breast cancer pathophysiologies," *Cancer Cell*, vol. 10, pp. 529–541, 2006.

[9] E. B. Fox, E. B. Sudderth, M. I. Jordan, and A. S. Willsky, "An HDP-HMM systems with state persistence," in *International conference on machine learning (ICML)*, 2008.

[10] J. Fridlyand, A. M. Snijders, D. Pinkel, D. G. Albertson, and A. N. Jain, "Application of Hidden Markov Models to the analysis of the array CGH data," *Journal of Multivariate Analysis*, vol. 90, pp. 132–153, 2004.

[11] T. Ganchev, N. Fakotakis, and G. Kokkinakis, "Comparative evaluation of various MFCC implementations on the speaker verification task," in *Proceedings of the International Conference on Speech and Computer*, 2005.

[12] W. R. Gilks and D. J. Spiegelhalter, *Markov Chain Monte Carlo in practice*. London: Chapman Hall, 1996.

[13] S. Guha, Y. Li, and D. Neuberg, "Bayesian hidden Markov modeling of array CGH data," *Journal of American Statistical Association*, vol. 103, pp. 485–497, 2008.

[14] H. Ishwaran and L. F. James, "Gibbs sampling methods for stick-breaking priors," *Journal of the American Statistical Association*, vol. 96(453), pp. 161–173, 2001.

[15] A. Jasra, C. C. Holmes, and D. A. Stephens, "Markov chain Monte Carlo methods and the label switching problem in Bayesian mixture modeling," *Statistical Science*, vol. 20(1), pp. 50–67, 2005.

[16] S. Ji, B. Krishnapuram, and L. Carin, "Variational Bayes for continuous hidden Markov models and its application to active learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28(4), pp. 522–532, 2006.

[17] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. Saul, "An introduction to variational methods for graphical models," *Maching Learning*, vol. 37(2), pp. 183–233, 1999.

[18] O. C. Lingjaerde, L. O. Baumbusch, K. Liestol, I. K. Glad, and A. L. Borresen-Dale, "CGH-Explorer: a program for analysis of array-CGH data," *Bioinfromatics*, vol. 21, pp. 8218–8222, 2005.

[19] C. L. Myers, M. J. Dunham, S. Y. Kung, and O. G. Troyanskaya, "Accurate detection of aneuplodies in array CGH and gene expression microarray data," *Bioinfromatics*, vol. 20, pp. 3533–3543, 2004.

[20] K. Ni, J. Paisley, L. Carin, and D. Dunson, "Multi-task learning for analyzing and sorting large databases of sequential data," *IEEE Transactions on Signal Processing*, vol. 56(8), pp. 3918–3931, 2008.

[21] A. B. Olshen, E. S. Venkatraman, R. Lucito, and M. Wigler, "Circular binary segmentation for the analysis of array-based DNA copy number data," *Biostatistics*, vol. 4, pp. 557–572, 2004.

[22] J. Pollack, T. Sorlie, C. Perou, C. Rees, S. Jeffrey, P. Lonning, R. Tibshirani, D. Botstein, A. Borresen-Dale, and P. Brown, "Microarray analysis reveals a major direct role of DNA copy number alteration in the transcriptional program of human breast tumors," *Proceedings of the National Academy of Sciences USA*, vol. 99, pp. 12 963–12 968, 2002.

[23] Y. Qi, J. W. Paisley, and L. Carin, "Music analysis using hidden Markov models," *IEEE Transactions on Signal Processing*, vol. 55(11), pp. 5209–5224, 2007.

[24] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77(2), pp. 257–286, 1989.

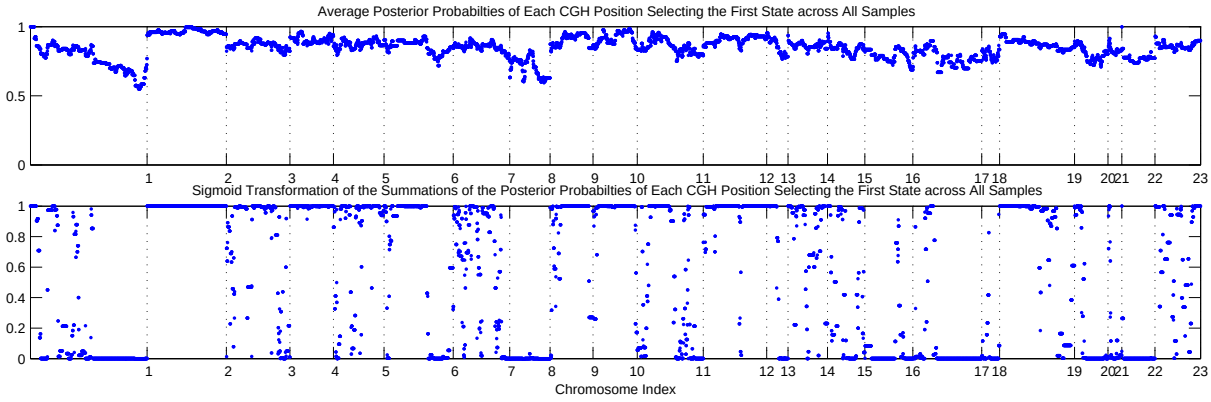
[25] J. Sethuraman, "A constructive definition of Dirichlet priors," *Statistica Sinica*, vol. 4, pp. 639–650, 1994.

[26] Y. Teh, M. Jordan, M. Beal, and D. Blei, "Hierarchical Dirichlet processes," *Journal of the American Statistical Association*, vol. 101, pp. 1566–1582, 2005.

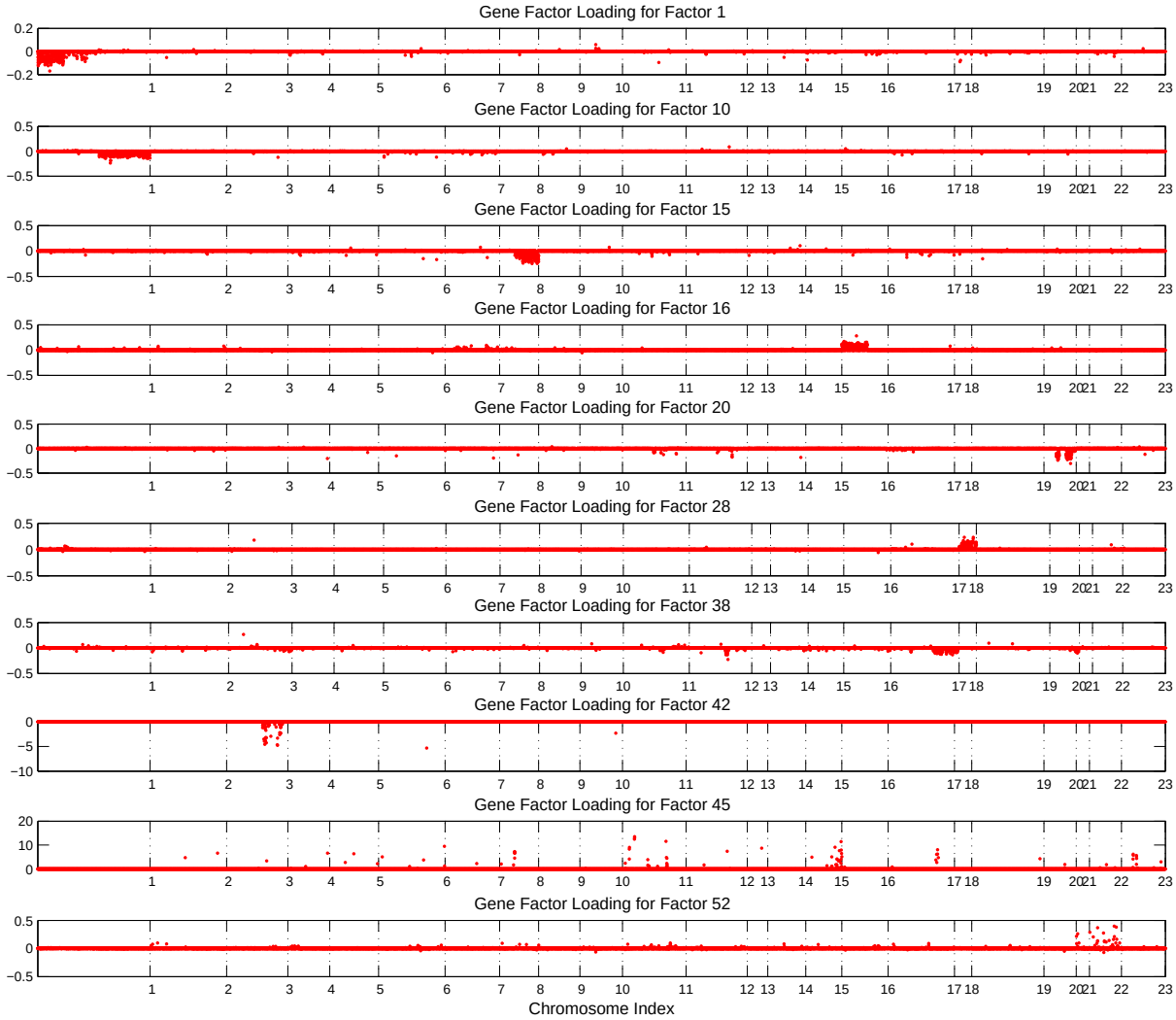
[27] S. Thrun and J. O'Sullivan, "Discovering structure in multiple learning tasks: The TC algorithm," in *International conference on machine learning (ICML)*, 1996.

[28] J. Winn and C. M. Bishop, "Variational message passing," *Journal of Machine Learning Research*, vol. 6, pp. 661–694, 2005.

[29] D. M. Witten, R. Tibshitani, and T. Hastie, "A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis," *Biostatistics*, vol. 10(3), pp. 515–534, 2009.



(a)



(b)

Fig. 8. (a) Posterior probabilities of each CGH position selecting the normal state. The top figure shows the average probabilities across all sample; the bottom figure shows the sigmoid transformation of the summations of the probabilities across all sample. (b) Gene factor loadings of some useful factors via SBFA [7] with a total of $K = 55$ factors. The CGH and gene measurements are aligned according to the index of the DNA fragments or "clones".