2010 International Conference on Pattern Recognition

Annealed SMC Samplers for Dirichlet Process Mixture Models[1]

Yener Ulker, Bilge Gunsel
Multimedia Signal Proc.& Pattern Recognition Lab.
Dept. of Electronics and Communications Eng.
Istanbul Technical University, 34469 Maslak, Istanbul, Turkey
yenerulker@itu.edu.tr, gunselb@itu.edu.tr

Ali Taylan Cemgil
Dept. of Computer Eng.
Bogazici University, 34342 Bebek
Istanbul, Turkey
taylan.cemgil@boun.edu.tr

***Abstract*— In this work we propose a novel algorithm that approximates sequentially the Dirichlet Process Mixtures (DPM) model posterior. The proposed method takes advantage of the Sequential Monte Carlo (SMC) samplers framework to design an effective annealing procedure that prevents the algorithm to get trapped in a local mode. We evaluate the performance in a Bayesian density estimation problem with unknown number of components. The simulation results suggest that the proposed algorithm represents the target posterior much more accurately and provides significantly smaller Monte Carlo error when compared to particle filtering.**

***Keywords*-Bayesian nonparametrics, Dirichlet process mixture, sequential Monte Carlo**

I. Introduction

In recent years, Dirichlet Process Mixtures (DPM) have been one of the most popular approach to probabilistic modelling [1], [2], [3]. Originally, DPM have been widely used as a building block in hierarchical models for solving density estimation and clustering problems, where the actual form of the underlying density is not constrained to a parametric family. If inference can be carried out effectively in a DPM, at least in principle, any density can be approximated with arbitrary precision. Inference in a DPM model is unfortunately intractable, hence there has been a surge of interest for efficient inference strategies. In this context, variational approximations [4] and Monte Carlo Markov Chain (MCMC) [5] techniques are widely used.

Inference in DPM is closely linked to clustering, and hence inherently a batch operation where the order of data should not matter. However, even if the data generating process is not sequential, it might be nevertheless beneficial to process data online in some prespecified order. Such sequential processing may give computational advantages for large datasets. Moreover this provides a natural tempering effect, i.e., a sequence of inference problems with increasing difficulty in contrast to one hard problem directly to be solved by a batch algorithm. Intuitively, it is more beneficial to "reuse" past inference instead of starting from scratch each time a new observation arrives. Therefore, development of online inference techniques have been proposed to estimate the time evolving DPM posterior [2], [3].

In this work we propose a novel sequential Monte Carlo sampler that estimates the sequentially evolving DPM model posterior. Our algorithm differs from the existing sequential methods [2], because it enables us to update past trajectories of the particles in the light of recent observations. Unlike the conventional approaches [3] that apply Gibbs moves to the weighted set of particles, the proposed method takes advantage of the SMC sampler framework [6] to design an annealing scheme that prevents the algorithm to get stuck in a local mode due to slow mixing property of the Gibbs sampler. In contrast to our previous work [7], here we concentrate on using an annealing scheme that utilizes a single proposal kernel. We also show that the sequential algorithm proposed in [3] is also a particular instance of the SMC framework.

II. Dirichlet Process Mixtures (DPM)

In this section we will construct a mixture model sequentially, where data arrives one by one. To denote the sequential construction, we extend our notation with an explicit 'time' index n. We denote the observation sequence by $y_n = \{y_{n,1} \dots y_{n,n}\}$. Each observation $y_{n,i}$, $i = 1, \dots n$, is assigned to a cluster where $z_{n,i} \in \{1, \dots k_n\}$ is the corresponding cluster label and, $k_n \in \{1 \dots n\}$ represents the number of existing clusters at time n. The vector of cluster variables is defined as $z_n = \{z_{n,1} \dots z_{n,n}\}$ and corresponding cluster parameters are represented with the parameter vector $\theta_n = \{\theta_{n,1} \dots \theta_{n,k_n}\}$.

The DPM model assumes that the cluster parameters are independently drawn from the prior $\pi(\theta_n)$ and the observations are independent of each other conditional on the assignment variable $z_{n,i}$. Hence the DPM posterior density $\pi(x_n)$ can be expressed as,

$$\pi_n(x_n) \propto p(z_n) \prod_{j=1}^{k_n} p(\theta_{n,j}) \prod_{i=1}^{n} p(y_{n,i}|\theta_{n,z_{n,i}}) \qquad (1)$$

where $x_n = \{z_n, \theta_n\}$. The prior on clustering variable vector

[1]This work is partially supported by TUBITAK BIDEP. A. Taylan Cemgil is supported by Bogazici University research fund BAP 09A105P

1051-4651/10 $26.00 © 2010 IEEE
DOI 10.1109/ICPR.2010.688

z_n is formulated by Eq.(2) in a recursive way,

$$p(z_{n,i+1} = j | z_{n,\{1:i\}}) = \begin{cases} \frac{l_j}{i+\kappa}, & j = 1, .., k_i \\ \frac{\kappa}{i+\kappa}, & j = k_i + 1 \end{cases} \quad (2)$$

where k_i is the number of clusters in the assignment $z_{n,\{1:i\}}$. In Eq.(2) l_j is the number of observations that $z_{n,\{1:i\}}$ assigns to cluster j and κ is a 'novelty' parameter [8]. In our work, we assume that conjugate prior is chosen for the parameters to ensure the conjugacy. Typically given z_n, the parameter θ_n can be integrated out and the DPM posterior distribution can be calculated up to a normalizing constant.

III. Sequential Monte Carlo Samplers

In sequential Monte Carlo algorithms such as particle filtering, we sample from a sequence of target densities evolving with a countable index n, $\pi_1(x_1) \dots \pi_n(x_n)$, each defined on a measurable space $(E_n, \mathcal{E}_n)$ where $x_n \in E_n$. In order to derive the importance weights sequentially one needs to define the sequence of proposal distributions as $\eta_1(x_1) \dots \eta_n(x_n)$ of which closed form solution is not available.

To eliminate this limitation, Del Moral et al. [6] proposed an auxiliary variable technique which solves the sequential importance sampling problem in an extended space $E^n = \{E_1 \times \dots \times E_n\}$. SMC sampler performs importance sampling between the joint importance distribution $\eta_n(x_{1:n})$ and the artificial joint target distribution defined by $\widetilde{\pi}_n(x_{1:n}) = \widetilde{\gamma}_n(x_{1:n})/Z_n$ where Z_n denotes the normalizing constant and

$$\widetilde{\gamma}_n(x_{1:n}) = \gamma_n(x_n) \prod_{k=1}^{n-1} L_k(x_{k+1}, x_k). \quad (3)$$

L_n is the backward Markov Kernel from space E_{n+1} to E_n and the joint posterior $\widetilde{\pi}_n(x_{1:n})$ defined on the extended space, E^n, admits $\pi_n(x_n)$ as a marginal. Therefore the resultant weight function ensures convergence to the original target density. The generic SMC algorithm can be presented as follows [6].

Assume that a set of weighted particles $\left\{W^i_{n-1}, X^i_{1:n-1}\right\}_{i=1}^{N_p}$ approximate $\widetilde{\pi}_{n-1}$ at time $n-1$. At time n the path of each particle can be extended using a Markov kernel, $K_n(x_{n-1}, x_n)$. The unnormalized importance weights, $\widetilde{\gamma}_n(x_{1:n})/\eta_n(x_{1:n})$, associated with the extended particles are calculated according to $w_n(x_{1:n}) = w_{n-1}(x_{1:n-1}) v_n(x_{n-1}, x_n)$ where the incremental term of weight equation, $v_n(x_{n-1}, x_n)$, is equal to

$$v_n(x_{n-1}, x_n) = \frac{\gamma_n(x_n) L_{n-1}(x_n, x_{n-1})}{\gamma_{n-1}(x_{n-1}) K_n(x_{n-1}, x_n)}. \quad (4)$$

Design of efficient sampling schemata hinges on properly choosing the backward kernel L_n. Assuming K_n is an Monte Carlo Markov Chain (MCMC) kernel of invariant distribution π_n, an approximate backward kernel can be formulated as shown in Eq.(5).

$$L_{n-1}(x_n, x_{n-1}) = \frac{\pi_n(x_{n-1}) K_n(x_{n-1}, x_n)}{\pi_n(x_n)} \quad (5)$$

Eq.(5) is accepted as a good approximation for $\pi_{n-1} \approx \pi_n$ and yields to the incremental weight, $v_n(x_{n-1}, x_n) = \gamma_n(x_{n-1})/\gamma_{n-1}(x_{n-1})$.

A well known method in order to increase the efficiency of the sequential importance sampling based approaches is to apply MCMC moves to each particle using a Gibbs sampler [3]. In the following section we will explicitly show that such a method is a special case of the SMC framework which utilizes MCMC kernels.

IV. MCMC kernels for DPM models

In this section, we will define the forward Kernel K_n generating samples from the sequence of distributions built according to Eq.(1). We first partition an assignment vector $z_n = \{z_{n,r}, z_{n,d}, z_{n,n}\}$ where r is a subset of $\{1, \dots, n-1\}$, a set of not necessarily consecutive indicies, and $d = \{1, \dots, n-1\} - r$. Note that throughout the text we will call the set $z_{n,r}$ as the active block. We define $u = r \cup \{n\}$, and denote $-u \equiv d$.

Now, let us define a forward kernel as follows,

$$K_n(z_{n-1}, z_n) = \delta_{z_{n-1,-u}}(z_{n,-u}) K_n(z_{n,n}, z_{n,r} | z_{n-1}) \quad (6)$$

where $K_n(z_{n,n}, z_{n,r} | z_{n-1})$ is a valid MCMC kernel applying a single Gibbs iteration targeting the full conditional distribution $\pi_n(z_{n,n}, z_{n,r} | z_{n,-u})$.

The backward kernel for the MCMC kernel can be obtained by substituting Eq.(6) into Eq.(5) and the resulting incremental weight update equation is,

$$v_n(z_{n-1}, z_n) = \frac{\gamma_n(z_{n-1,r}, z_{n,-u})}{\gamma_{n-1}(z_{n-1})}. \quad (7)$$

Eq.(7) is independent of the MCMC kernel K_n hence valid for any initialization of the kernel. Note that when the active block is selected as the set $r = \{1 \dots n-1\}$, Eq.(7) corresponds to the S4 algorithm utilized by [3].

Intuitively, the algorithm first samples the latest clustering label $z_{n,n}$ using the full conditional distribution $\pi_n(z_{n,n} | z_{n-1})$ and updates the active block $z_{n,r}$ using a Gibbs sampler. In a sequential problem the posterior distribution changes over time and new modes of the posterior distribution may emerge as new observations are received. The algorithm must have good mixing property to explore the modes of the time evolving posterior distribution and achieve a good approximation to the true target posterior. However, conventional sequential and batch algorithms based on Gibbs sampler may fail to represent the modes of the true target posterior due to slow convergence property of the Gibbs samplers and will likely to stuck in local modes particularly when the posterior distribution has a multi modal form where

the modes are isolated. To deal with this problem, in the next subsection we introduce an algorithm that converges to the true DPM posterior as the new observations are received sequentially.

A. Annealed kernels for DPM mixtures

Conventional approach defined in Section IV applies Gibbs moves to each particle in order to obtain weighted samples from a sequence of target distributions given by $\pi_1(z_1), \ldots, \pi_n(z_n)$. This paper proposes an annealing scheme to improve the efficiency of posterior estimation. In literature annealing schemes have been widely used to handle isolated modes in batch processing. It is adopted to importance sampling to construct good proposal distributions for sampling sequence of distributions [9]. To achieve our goal let us construct an annealed time evolving target posterior as, $\pi'_1(z'_1), \ldots \pi'_n(z'_n)$, where π'_n is also defined on the measurable space $(E_n, \mathcal{E}_n)$ and $z'_n \in E_n$. The notation $'$ is used to denote the annealed target posterior $\pi'_k(z'_k) = \pi_k(z_k|\kappa = \alpha_k)$ where $z'_k = z_k$, $k = \{1 \ldots n\}$, and annealing is achieved by changing the parameter α_k of the underlying Dirichlet process. Note that α_k is the prior parameter on the number of components where higher values yields higher number of mixture components . Hence the idea behind constructing a sequence of annealed target posterior distributions is to obtain intermediate distributions that cover the high probability regions of the time evolving true target posterior. In other words, the annealed distributions are DPM models with relaxed parameters for which the particle filter approximation is hopefully more efficient than the true target. In the following we will explain how we define the annealed target distribution $\pi'_n(z'_n)$ as an intermediate step to estimate the time evolving true target posterior $\pi_n(z_n)$.

In order to sample the sequence of annealed target distribution, let us define a forward kernel as follows,

$$K_n(z'_{n-1}, z'_n) = \delta_{z'_{n-1,-u}}\left(z'_{n,-u}\right) K_n(z'_{n,n}, z'_{n,r}|z'_{n-1}) \quad (8)$$

where $K_n(z'_{n,n}, z'_{n,r}|z'_{n-1})$ is an MCMC kernel which targets the conditional distribution $\pi'_n(z'_{n,n}, z'_{n,r}|z'_{n,-u})$. Using Eq.(5), the backward kernel can be written as in Eq.(9),

$$L_{n-1}(z'_n, z'_{n-1}) = \frac{\pi'_n(z'_{n-1})K_n(z'_{n-1}, z'_n)}{\pi'_n(z'_n)} \quad (9)$$

and the incremental weights for the annealed target posterior can be obtained as follows,

$$v'_n(z'_{n-1}, z'_n) = \frac{\gamma'_n(z'_n)\pi'_n(z'_{n-1,r}|z'_{n-1,-u})}{\gamma'_{n-1}(z'_{n-1})\pi'_n(z'_n, z'_{n,r}|z'_{n,-u})} \quad (10)$$

where the weights associated with the particles can be calculated according to $w'_n(z'_{1:n}) = w'_{n-1}(z'_{1:n-1}) \times v'_n(z'_{n-1}, z'_n)$. Assuming $\left\{W'^{(i)}_n\right\}_{i=1}^{N_p}$ represents the normalized weights approximating to $\pi'_n(z'_n)$, we perform a resampling step if effective sample size, $N_{eff} = \left[\sum_{i=1}^{N_p}(W'^{(i)}_n)^2\right]^{-1}$, is below a predefined threshold.

Finally, in order to approximate to the target distribution $\pi_n(z_n)$, we use a Dirac kernel of the form $K_n(z'_n, z_n) = \delta_{z'_n}(z_n)$ and find the weights according to $w_n(z_{1:n}) = w'_n(z'_{1:n}) \times v_n(z'_n, z_n)$ where $v_n(z'_n, z_n) = \gamma_n(z_n)/\gamma'_n(z'_n)$.

Specification of the active block size r shown in Eq.(10) is an important issue in the design of proposed sampler. In order to limit the computational cost required at each time step we initially determine a constant block size Q and index the block with $r_1 \ldots r_Q$. The indexes of the active block is incremented by Q as each new observation is received. The blocks do not overlap to each other and update scheme is cycled whenever all the clustering labels up to time n are updated. Note that similar block update strategies are also used in [10] under the SMC samplers framework.

V. Test Results and Conclusions

Our goal in this section is to illustrate the effectiveness of the SMC samplers framework for online inference in DPM models. For this purpose, we compare performance of three samplers namely; the SMC-G which utilizes conventional Gibbs moves on the DPM space [7], the proposed SMC sampler (SMC-A), and the Particle filter (PF). Results are reported in terms of mean estimates and respective standard errors. Mixture density estimates obtained by the SMC-G and SMC-A samplers are also provided for visual comparison.

Performance of the algorithms are evaluated for the standard Gaussian mixture density estimation problem with unknown number of components. It is assumed that observations are drawn from a univariate Gaussian with unknown mean μ and variance σ^2, hence $\theta = \{\mu, \sigma^2\}$. The distribution of the parameters μ and σ^2 are respectively chosen as normal and inverse-gamma, to ensure the conjugacy condition.

To alleviate the degeneracy, a systematic resampling scheme is applied for sequential algorithms when $N_{eff} < 4/5N_p$. For a fair comparison the number of particles is selected as $N_p = 1000$ for particle filter and $N_p = 100$ for SMC algorithms. Results are reported for 100 independent Monte Carlo runs for each model. The active block size Q is set to 9.

The initial annealing parameter for annealed target distribution is set to $\alpha_1 = 1$ and it is geometrically updated according to $\alpha_n = \alpha_{n-1} + c_\alpha(\kappa - \alpha_{n-1})$, $n \in \mathbf{N}$, at each time step where the common parameter, c_α, is set to $1/100$. Note that as $n \to \infty$, α_n will converge to κ that ensures convergence to the true target posterior with the increasing time.

Two test sets (D-1 and D-2) are generated from a Gaussian mixture model comprising three mixture components. Parameters of the generated data are given in Table.I where μ_i, σ_i, and p_i, for $i \in \{1 \ldots 3\}$, denote the mean, standard

Table I
TRUE MODEL PARAMETERS

	p_1, p_2, p_3	μ_1, μ_2, μ_3	$\sigma_1, \sigma_2, \sigma_3$
Data-1 (D-1)	1/3,1/3,1/3	0,1.5,3	0.5,0.5,0.5
Data-2 (D-2)	1/2,1/6,1/3	0,2,4	0.5,0.5,2.5

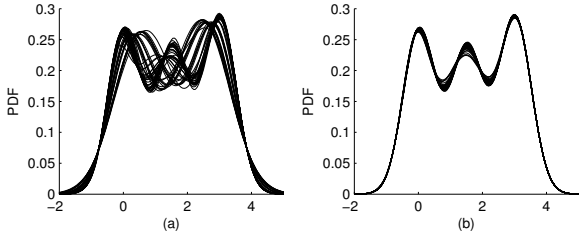

Figure 1. Estimated mixture densities by the a) SMC-G and b) SMC-A algorithm for 100 Monte Carlo runs. SMC-A represent all tree components of the mixture density in all runs.

deviation and the mixture weight for each component, respectively. Each data set has 1000 points, that we run the test sequentially for $200, 500$, and 1000 samples.

In order to illustrate the estimation quality of the proposed algorithm we set the novelty parameter to a very low value of $\kappa = 0.05$. Note that a low κ value will cause the posterior to have isolated modes and leads to a hard inference problem. This test aims to assess the ability of the algorithms to escape from local modes. We perform the test by generating a total of 1000 observations from the model D-1 which comprise three overlapping mixture components. In Fig. 1 (a) and (b), the mixture densities plotted for each run of the SMC-G and SMC-A algorithms, respectively. We observe that SMC-A can represent all tree components of the mixture density in all runs of the algorithm whereas SMC-G commonly gets trapped at a local mode and fits 2 mixture components to the data for several runs (nearly the half) of the algorithm. We also report the mean estimate and the standard errors (in parenthesis) of the number of components in Table.II for SMC-G and SMC-A. The results illustrate that SMC-A is able to converge to the true number of components (3) for a small number of observations, however the SMC-G algorithm does not converge even when the observation size is 1000.

Table II
ESTIMATED MEAN VALUES AND RESPECTIVE MONTE CARLO ERRORS FOR SMC-G AND SMC-A

	Observation intervals			
	Algo.	200	500	1000
D-1	SMC-G	2.05 (0.002)	2.54 (0.249)	2.71 (0.255)
D-1	SMC-A	2.13 (0.003)	3.05 (0.012)	3.07 (0.003)

In order to examine the performance of the algorithms under different parameter regimes, we set the novelty parameter to a larger value of $\kappa = 0.5$. This tends to lead to a 'smoother' posterior density where inference is arguably simpler. The estimated mean values of the number of components and the standard errors for the test sets D-1 and D-2 are reported in Table.III. As it is shown in Table.III, both PF and SMC algorithms provide almost identical mean estimates for both datasets. However, SMC-G and SMC-A can achieve significantly lower standard error compared to PF at $n = 1000$. This results show that SMC sampler approach provides more reliable estimates than the particle filter for the harder inference problem at the same computational cost. For the simpler problem, both algorithms (SMC-A and SMC-G) show comparable performance.

Table III
ESTIMATED MEAN VALUES AND RESPECTIVE MONTE CARLO ERRORS FOR SMC-G, SMC-A AND PF

		Observation intervals		
	Algo.	200	500	1000
D-1	SMC-G	2.99 (0.037)	3.58 (0.041)	3.69 (0.035)
	SMC-A	3.00 (0.033)	3.57 (0.025)	3.65 (0.024)
	PF	2.95 (0.032)	3.55 (0.226)	3.69 (0.285)
D-2	SMC-G	4.17 (0.039)	4.57 (0.075)	4.66 (0.126)
	SMC-A	4.16 (0.026)	4.54 (0.081)	4.65 (0.115)
	PF	4.16 (0.037)	4.55 (0.145)	4.76 (0.357)

As future work it is appealing to extend the proposed algorithm to a non conjugate setting where the parameters of the model need to be also sampled.

REFERENCES

[1] D. Blei and M. Jordan., "Variational inference for Dirichlet process mixtures.," *Journal of Bayesian Analysis*, vol. 1, pp. 121–144, 2006.

[2] P. Fearnhead, "Particle filters for mixture models with an unknown number of components," *J. Stat. Comput.*, vol. 14, pp. 11–21, 2004.

[3] S. N. MacEachern, M. Clyde, and J. Liu, "Sequential importance sampling for nonparametric Bayes models: the next generation," *Can. J. Stat.*, vol. 27, pp. 251–267, 1999.

[4] D. M. Blei and M. I. Jordan, "Variational methods for the Dirichlet process,," in *Proceedings of the 21st International Conference on Machine Learning*, 2004.

[5] M. Escobar and M. West, "Computing Bayesian nonparametric hierarchical models," tech. rep., Duke University, Durham, USA, 1992.

[6] Del Moral, A. Doucet, and A. Jasra, "Sequential Monte Carlo samplers," *J. Roy. Stat. Soc. B Stat. Meth.*, vol. 63, pp. 11–436, 2006.

[7] Y. Ulker, B. Gunsel, and A. T. Cemgil, "SMC samplers for Dirichlet process mixtures," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS)*, vol 9, pp 876-883, 2010.

[8] J. Sethuraman, "A constructive definition of Dirichlet priors," *Statist. Sin.*, vol. 4, pp. 639–650, 1994.

[9] R. Neal, "Annealed importance sampling," *Statist. Comput*, vol. 11, pp. 125–139, 2001.

[10] A. Doucet, M. Briers, and S. Senecal, "Efficient block sampling strategies for sequential Monte Carlo methods," *J. Comput. Graph. Stat.*, vol. 15, pp. 693–711, 2006.