2nd Symposium on Advances in Approximate Bayesian Inference, 2019 1–7

Sequential Learning for Dirichlet Process Mixtures

Chunlin Ji CHUNLIN.JI@KUANG-CHI.ORG
Kuang-Chi Institute of Advanced Technology
Shenzhen, China

Bin Liu BINS@IEEE.ORG
School of Computer Science, Nanjing University of Posts and Telecommunications
Nanjing, China

Yingkai Jiang YKJIANG15@GMAIL.COM
Ke Deng KDENG@TSINGHUA.EDU.CN
Center for Statistical Science, Tsinghua University
Beijing, China

Abstract

Dirichlet process mixture model provides a flexible nonparametric framework for unsupervised learning. Monte Carlo based sampling methods always involve heavy computation efforts; conventional variational inference requires careful design of the variational distribution and the conditional expectation. In this work, we treat the DP mixture itself as the variational proposal, and view the given data as drawn samples of the unknown target distribution. We propose an evidence upper bound (EUBO) to act as the surrogate loss, and fit a DP mixture to the given data by minimizing the EUBO, which is equivalent to minimizing the KL-divergence between the target distribution and the DP mixture. We provide three advantages of the EUBO based DP mixture fitting and show how to build the black-box style sequential learning algorithm. We use the stochastic gradient descent (SGD) algorithm for optimization that leverages on the automatic differentiation tools. Simulation studies are provided to demonstrate the efficiency of our proposed methods.

1. Introduction

Dirichlet process mixture model provides a flexible nonparametric framework for unsupervised learning (Blei and Jordan, 2004). But fitting DP mixtures for large dataset is nontrivial: Monte Carlo based sampling methods always involve heavy computation efforts and require an extremely long running time (MacEachern, 1994; Escobar and West, 1995); variational inference methods for the DP mixtures require careful design of the variational distribution and the conditional expectation (Blei and Jordan, 2004; Kurihara et al., 2007; Huynh et al., 2015). Inspired by the adaptive MCMC with mixture proposal (Ji and Schmidler, 2013; Cappé et al., 2008) as well as stochastic approximation version of EM (Celeux and Diebolt, 1992; Delyon et al., 1999; Chen et al., 2018), a sequential learning approach for fitting a DP mixture model is proposed in this work. We propose a new perspective on variational inference for DP mixtures. Specifically, we use the DP mixture itself as the variational proposal, and view the given data as drawn samples of the unknown target distribution. We present an evidence upper bound (EUBO) as the optimization surrogate loss. By minimizing this EUBO, we fit a DP mixture for the given data in the sense of

minimizing their KL-divergence. We present three nice properties of the EUBO, which make it an elegant choice for fitting DP mixtures. The SGD algorithm with mini-batch data is utilized for optimizing the variational parameters, which enables our method to deal with large scale dataset. Leveraging on the advantage of truncated DP mixtures, we can obtain a closed form update formula to iteratively update the weights of the truncated DP mixture components; for the mean and covariance of the DP mixtures, we utilize the modern automatic differentiation tools Paszke et al. (2017).

2. Sequential learning for DP Mixtures

Assume a set of data $X = [x_1, ..., x_N]$, which follows an unknown distribution $\pi(x)$. Our goal is to fit a nonparametric DP mixtures $q(x; \phi)$ for the entire data set X to approximate $\pi(x)$ in a sequential fashion: at each iteration we choose X_s, a subset of X selected either randomly or by design, then update the nonparametric distribution $q(x; \phi)$ by learning from X_s.

2.1. Truncated DP Mixtures

DP mixtures have different representation schemes such as Polya urn (Ferguson, 1973) and stick-breaking (Sethuraman, 1994). We choose the truncated DP mixture as the nonparametric variational proposal. The truncated DP mixtures assume a large but finite K for the components number (Ishwaran and James, 2001), that is $q_\phi(x) = \sum_{k=1}^{K} w_k N(\cdot|\mu_k, \Sigma_k)$ where $w_k = V_k \prod_{j=1}^{k-1}(1 - V_j)$ with V_k being i.i.d. samples from Beta distribution $Beta(1, \alpha)$ and μ_k, Σ_k denote the mean and the covariance of the normal distribution. Let ϕ denotes $\{V_k, \mu_k, \Sigma_k\}_{k=1}^{K}$. Leveraging on the representation form of truncated DP mixtures, we can obtain a closed form update formula to iteratively update the parameters of the truncated DP mixtures.

2.2. Evidence upper bound

In the previous study of variational inference for DP mixtures (Blei and Jordan, 2004), they assume the parameters of DP mixtures as hidden variable and using the mean-field variational method to find the Bayesian posterior for these parameters. But the complex setting of this method prevents it from working in a black-box fashion, which is desirable in dealing with large dataset. In comparison, here we view x as the hidden variable and the given dataset X as drawn samples from an unknown ground true distribution $\pi(x)$, also denoted as $p(x|\mathcal{M})$ where $\mathcal{M}$ is a specified model. We treat the DP mixture $q_\phi(x)$ itself as the variational distribution. Then we introduce an upper bound on the *evidence*: $p(\mathcal{M}) = \int p(x, \mathcal{M})dx = \int p(x|\mathcal{M})p(\mathcal{M})dx$, by using the Gibbs' inequality that $-\int p(x|\mathcal{M}) \log p(x|\mathcal{M})dx \leq -\int p(x|\mathcal{M}) \log q_\phi(x)dx$ for any distribution $q_\phi(x)$,

$$\begin{aligned} \log p(\mathcal{M}) &= \int p(x|\mathcal{M})[\log p(x, \mathcal{M}) - \log p(x|\mathcal{M})]dx \\ &\leq \int p(x|\mathcal{M})[\log p(x, \mathcal{M}) - \log q_\phi(x)]dx. \end{aligned} \tag{1}$$

We define $\mathcal{U} \equiv \mathbb{E}_{\pi(x)}[\log p(x, \mathcal{M}) - \log q_\phi(x)]$ as the Evidence Upper BOund (EUBO) (Ji and Shen, 2019). It is easy to find that $\mathcal{U} = \log p(\mathcal{M}) + \mathcal{D}_{\mathrm{KL}}(\pi(x)||q_\phi(x))$, where the KL-divergence $\mathcal{D}_{\mathrm{KL}}(\pi(x)||q_\phi(x))$ is just the discrepancy between the EUBO $\mathcal{U}$ and the true $\log p(\mathcal{M})$. Minimizing the EUBO leads to fitting $q_\phi(x)$ to the dataset X in the sense that $\mathcal{D}_{\mathrm{KL}}(\pi(x)||q_\phi(x))$ is minimized. This EUBO has several nice properties: 1) $\mathcal{D}_{\mathrm{KL}}(\pi(x)||q_\phi(x))$ posses mass covering (or say zero-avoiding) property (Minka, 2005), that is when minimizing this $\mathcal{D}_{\mathrm{KL}}$, $q_\phi(x)$ tends to cover all the area where $\pi(x)$ is non-zero; 2) in our setting, we assume X are samples from the unknown $\pi(x)$, so we can apply the Monte Carlo method to deal with the integration in $\mathcal{U}$; 3) we do not require the numerical value of $\pi(x)$, but other bounds like the evidence lower bound(ELBO)(Hoffman et al., 2013), the Rényi bound (Li and Turner, 2016) and the Chi-upper bound (Dieng et al., 2017) may need to evaluate $\pi(x)$, which is unknown in our problem setting. Moreover, the EUBO or $\mathcal{D}_{\mathrm{KL}}(\pi(x)||q_\phi(x))$ is a different loss compared with the marginal likelihood of the observed data used in EM methods, and we do not need the help of hidden variable which is required in the EM algorithm. To our knowledge, using EUBO as the surrogate loss to fit for DP mixtures is quite unique and has not been discussed in the literature.

2.3. SGD optimization

To apply the SGD optimization, we derive the gradient of $\mathcal{U}$ with respect to V_k, μ_k and Σ_k as follows,

$$\nabla_{V_k}\mathcal{U} = \int \pi(x)\frac{1}{q_\phi(x)}\left[-\sum_{l=k+1}^{K} V_l \prod_{j\le l-1, j\neq k}(1-V_j)q(x|\mu_l,\Sigma_l) + \prod_{j=1}^{k-1}(1-V_j)q(x|\mu_k,\Sigma_k)\right]dx, \quad (2)$$

$$\nabla_{\mu_k}\mathcal{U} = \int \pi(x)\frac{w_k q(x;\mu_k,\Sigma_k)}{q_\phi(x)}\nabla_{\mu_k}\log q(x;\mu_k,\Sigma_k)dx, \quad (3)$$

$$\nabla_{\Sigma_k}\mathcal{U} = \int \pi(x)\frac{w_k q(x;\mu_k,\Sigma_k)}{q_\phi(x)}\nabla_{\Sigma_k}\log q(x;\mu_k,\Sigma_k)dx. \quad (4)$$

Given the subset of the observation $X_s = \{x_s^{(i)}\}_{i=1}^{N_s}$ from $\pi(x)$, or say a mini-batch, the Monte Carlo approximation of these gradients is

$$\widetilde{\nabla}_{V_k}\mathcal{U} = \frac{1}{N_s}\sum_{i=1}^{N_s}\frac{1}{q_\phi(x_s^{(i)})}\left[-\sum_{l=k+1}^{K} V_l \prod_{j\le l-1, j\neq k}(1-V_j)q(x_s^{(i)}|\mu_l,\Sigma_l) + \prod_{j=1}^{k-1}(1-V_j)q(x_s^{(i)}|\mu_k,\Sigma_k)\right], \quad (5)$$

$$\widetilde{\nabla}_{\mu_k}\mathcal{U} = \frac{1}{N_s}\sum_{i=1}^{N_s}\frac{w_k q(x_s^{(i)};\mu_k,\Sigma_k)}{q_\phi(x_s^{(i)})}\nabla_{\mu_k}\log q(x_s^{(i)};\mu_k,\Sigma_k), \quad (6)$$

$$\widetilde{\nabla}_{\Sigma_k}\mathcal{U} = \frac{1}{N_s}\sum_{i=1}^{N_s}\frac{w_k q(x_s^{(i)};\mu_k,\Sigma_k)}{q_\phi(x_s^{(i)})}\nabla_{\Sigma_k}\log q(x_s^{(i)};\mu_k,\Sigma_k). \quad (7)$$

Leveraging on the automatic differentiation tools in Python, such as Pytorch (Paszke et al., 2017), we can obtain the numerical gradient of $\widetilde{\nabla}_{\mu_k}\mathcal{U}$ (or $\widetilde{\nabla}_{\Sigma_k}\mathcal{U}$) without further exploration of $\nabla_{\mu_k}\log q(x_s^{(i)};\mu_k,\Sigma_k)$ (or $\nabla_{\Sigma_k}\log q(x_s^{(i)};\mu_k,\Sigma_k)$), which significantly benefits the implementation. Advanced SGD type algorithms, such as Adam (Kingma and Ba, 2014), help in accelerating the convergence. The entire algorithm is shown as follows,

Algorithm 1: Sequential Learning for DP Mixtures

- Initialization: Choose $\psi_0 = (V_0, \mu_0, \Sigma_0)$ and set $t = 1$.
- For $t = 1 : T$, set the learning rate $\{r_{V,t+1}, r_{\mu,t+1}, r_{\Sigma,t+1}\}$, update the $V_{k,t}$, $w_{k,t}$, $\mu_{k,t}$ and $\Sigma_{k,t}$ (for $k = 1, ..., K$) as follows,

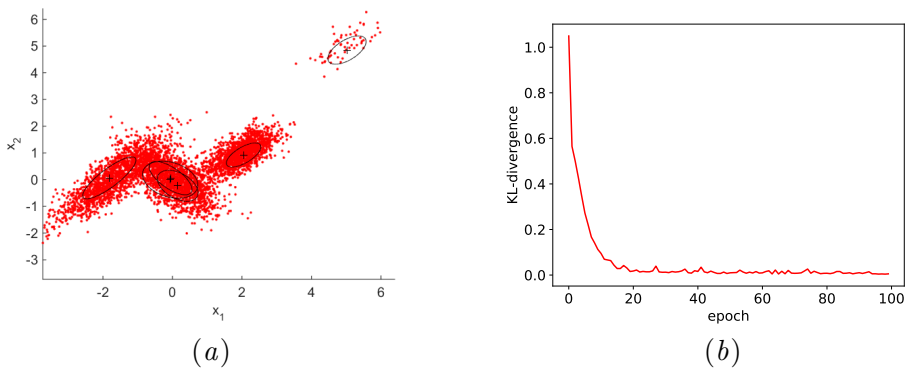
$$\begin{aligned} V_{k,t+1} &= V_{k,t} - r_{V,t+1}\widetilde{\nabla}_{V_k}\mathcal{U}, \\ w_{k,t+1} &= V_{k,t+1}\prod_{j=1}^{k-1}(1 - V_{j,t+1}), \\ \mu_{k,t+1} &= \mu_{k,t} - r_{\mu,t+1}\widetilde{\nabla}_{\mu_k}\mathcal{U}, \\ \Sigma_{k,t+1} &= \Sigma_{k,t} - r_{\Sigma,s+1}\widetilde{\nabla}_{\Sigma_k}\mathcal{U}. \end{aligned}$$

3. Simulation study

Example 1 Toy mixtures: We demonstrate the behavior of the proposed sequential learning algorithm by applying it to a synthetic data set: 5000 data points generated from a mixture of four bivariate normals with weights: $[0.3, 0.4, 0.29, 0.01]$, means: $[-1.75, 0]$, $[0, 0]$, $[2, 1]$, $[5, 5]$ and covariances: $[0.6, 0.5; 0.5, 0.6]$, $[0.4, -0.25; -0.25, 0.4]$, $[0.25, 0.15; 0.15, 2]$, $[0.3, 0.2; 0.2, 0.25]$. In our sequential learning context, we assume that at each iteration the observation is a subset of 50 data points uniformly selected. The TDP mixture is initialized as follows: the maximum number of components is set as $K = 10$, means of normal components μ_k (for $k = 1, ..., K$) are randomly initialized in range $[-5, 5] \times [-5, 5]$, the covariance are set as $\Sigma_k = 2\mathbf{I}$ (for $k = 1, ..., K$), and set $V_k = 1/(K - k + 1)$. The algorithm runs 100 epoches. The output of the sequential learning algorithm is w_k, μ_k, Σ_k for $(k = 1, ..., K')$, where K' is the number of components with weight larger than a threshold (1e-4). We show the fitted mixture model in the final iteration in Figure 1(a) and the estimated $\mathcal{D}_{\mathrm{KL}}(\pi(x)||q_\phi(x))$ over each epoch in Figure 1(b).

Example 2 7-dimension mixtures: We test the proposed approach on another target function $\pi(\cdot)$, which is a outer product of seven univariate distributions, with the marginal likelihood exactly equal to 1. These seven distributions are: 1) $\frac{3}{5}Ga(10+x|2,3)+\frac{2}{5}Ga(10-x|2,5)$; 2) $\frac{3}{4}skN(x|3,1,5)+\frac{1}{4}skN(x|-3,3,-6)$; 3)$T(x|0,9,4)$; 4) $\frac{1}{2}Be(x+3|3,3)+\frac{1}{2}N(x|0,1)$; 5) $\frac{1}{2}\epsilon(x|1)+\frac{1}{2}\epsilon(-x|1)$; 6) $skN(x|0,8,-3)$; 7) $\frac{1}{8}N(x|-10,0.1)+\frac{1}{4}N(x|0,0.15)+\frac{5}{8}N(x|7,0.2)$. where $Ga(\cdot|\alpha,\beta)$ denotes the gamma distribution, $N(\cdot|\mu,\sigma)$ denotes the normal distribution, $skN(\cdot|\mu,\sigma,\alpha)$ denotes the skew-normal distribution, $T(\cdot|\mu,\sigma,df)$ denotes the student-T distribution, $Be(\cdot|\alpha,\beta)$ denotes the beta distribution, and $\epsilon(\cdot|\lambda)$ denotes the exponential distribution. This target distribution is complex: dimension 2 has two modes bracketing a deep ravine; dimension 4 has one low, broad mode that overlaps a second sharper mode; dimension 7 has three distinct well-separated modes.

We syntheses 5000 data points from the target distribution. We use a mini-batch of 50 data points selected uniformly in each iteration. We compare the true target distribution, the kernel density estimation of samples drawn by the MCMC algorithm and the fitted TDP normal mixtures in univariate style in Figure 2. The fitted TDP mixtures matches the true target distribution well.



(a) (b)

Figure 1: (a) Synthetic data points are shown in red dots. The fitted TDP mixture is presented with + representing the mean of normal component and ellipse representing one standard deviation. (b)The KL-divergence between the true target distribution and the estimated TDP mixture per epoch.

These simulation studies show that the proposed sequential learning algorithm can iteratively learn a TDP mixture model to fit the data. The proposed algorithm affords at least three advantages: 1) it utilizes the TDP mixture model, which can provide more flexibility in modelling than a mixture model with a fixed component number; 2) instead of learning from the whole data set at each iteration, it requires only a subset of the data, reducing the computation cost; 3) it takes the advantage of the SGD type algorithm which makes the algorithm be suitable for complex models.

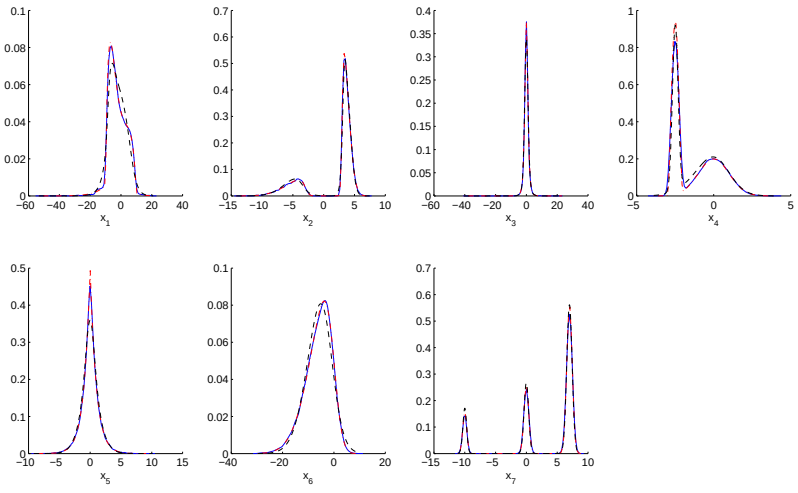


Figure 2: Plots of density functions for comparison: the true univariate density is shown by the red dashed dot curve; kernel density estimation of samples drawn by MCMC is shown by the solid blue curve; the fitted TDP mixtures is shown by black dashed curve.

References

D. Blei and M. Jordan. Variational inference for Dirichlet process mixtures. *Bayesian Analysis*, 1(1):121–144, 2004.

O. Cappé, R. Douc, A. Guillin, J.M. Marin, and C.P. Robert. Adaptive importance sampling in general mixture classes. *Statistics and Computing*, 18(4):447–459, 2008.

G. Celeux and J. Diebolt. A stochastic approximation type EM algorithm for the mixture problem. *Stochastics and Stochastics Reports*, 41:127–146, 1992.

J. Chen, J. Zhu, Y. Teh, and T. Zhang. Stochastic expectation maximization with variance reduction. In *Proceedings of the Neural Information Processing Systems*, 2018.

B. Delyon, M. Lavielle, and E. Moulines. Convergence of a stochastic approximation version of the em algorithm. *The Annals of Statistics*, 27(1):94–128, 1999.

A. Dieng, D. Tran, R. Ranganath, J. Paisley, and D. Blei. Variational inference via χ-upper bound minimization. In *Proceedings of the Neural Information Processing Systems*, 2017.

M. Escobar and M. West. Bayesian density estimation and inference using mixtures. *Journal of the American Statistical Association*, 90:577–588, 1995.

T. S. Ferguson. A Bayesian analysis of some nonparametric problems. *Annals of Statistics*, 1:209–230, 1973.

M. Hoffman, D. Blei, C. Wang, and J. Paisley. Stochastic variational inference. *Journal of Machine Learning Research*, 14:1303–1347, 2013.

V. Huynh, D. Phung, and S. Venkatesh. Streaming variational inference for dirichlet process mixtures. In *ACML*, 2015.

H. Ishwaran and L. F. James. Gibbs sampling methods for stick-breaking priors. *Journal of the American Statistical Association*, 96:161–173, 2001.

C. Ji and S. C. Schmidler. Adaptive markov chain monte carlo for bayesian variable selection. *Journal of Computational and Graphical Statistics*, 22:708–728, 2013.

C. Ji and H. Shen. Stochastic variational inference via upper bound. arXiv preprint, arXiv:1912.00650, 2019.

D. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

K. Kurihara, M. Welling, and Y. Teh. Collapsed variational dirichlet process mixture models. In *Proceedings of International Joint Conference on Artificial Intelligence*, 2007.

Y. Li and R. E. Turner. Variational inference with Rényi divergence. In *Proceedings of the Neural Information Processing Systems*, 2016.

S. MacEachern. Estimating normal means with a conjugate style Dirichlet process prior. *Communications in Statistics: Simulation and Computation*, 23:727–741, 1994.

T. Minka. Divergence measures and message passing. Technical report, Microsoft Research, 2005.

A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. 2017.

J. Sethuraman. A constructive definition of Dirichlet priors. *Statistica Sinica*, 4:639–650, 1994.