Online Inference in Bayesian Non-Parametric Mixture Models under Small Variance Asymptotics

Ajay Kumar Tanwani[1,2], **Sylvain Calinon**[1]
[1]Idiap Research Institute, Switzerland.
[2]Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland.
{ajay.tanwani, sylvain.calinon}@idiap.ch

Abstract

Adapting statistical learning models online with large scale streaming data is a challenging problem. Bayesian non-parametric mixture models provide flexibility in model selection, however, their widespread use is limited by the computational overhead of existing sampling-based and variational techniques for inference. This paper analyses the online inference problem in Bayesian non-parametric mixture models under small variance asymptotics for large scale applications. Direct application of small variance asymptotic limit with isotropic Gaussians does not encode important coordination patterns/variance in the data. We apply the limit to discard only the redundant dimensions in a non-parametric manner and project the new datapoint in a latent subspace by online inference in a Dirichlet process mixture of probabilistic principal component analyzers (DP-MPPCA). We show its application in teaching a new skill to the Baxter robot online by teleoperation, where the number of clusters and the subspace dimension of each cluster is incrementally adapted with the streaming data to efficiently encode the acquired skill.

1 Introduction

We are interested in online clustering of high-dimensional streaming data in a non-parametric manner. Let us denote the streaming observation sequence by $\{\boldsymbol{\xi}_1 \dots \boldsymbol{\xi}_t\}$, where $\boldsymbol{\xi}_t \in \mathbb{R}^D$ is obtained at the current time step t. The corresponding cluster assignment sequence $\{z_1 \dots z_t\}$ where $z_t \in \{1 \dots K\}$ belongs to the discrete set of K cluster indices at time t, and the observation $\boldsymbol{\xi}_t$ is drawn from a multivariate Gaussian with mixture coefficients $\pi_{t,i} \in \mathbb{R}$, mean $\boldsymbol{\mu}_{t,i} \in \mathbb{R}^D$ and covariance $\boldsymbol{\Sigma}_{t,i} \in \mathbb{R}^{D \times D}$ at time t. We seek to update the parameters online upon observation of a new datapoint $\boldsymbol{\xi}_{t+1}$, such that the datapoint can be discarded afterwards. Small variance asymptotic (SVA) analysis implies that the covariance matrix $\boldsymbol{\Sigma}_{t,i}$ of all the Gaussians reduces to the isotropic noise σ^2, i.e., $\boldsymbol{\Sigma}_{t,i} \approx \lim_{\sigma^2 \to 0} \sigma^2 \boldsymbol{I}$ [4, 2, 5]. In this paper, we present online inference algorithms based on applying Bayesian non-parametric treatment to Gaussian mixture model (GMM) and mixture of probabilistic principal component analyzers (MPPCA) under SVA.

2 Online DP-GMM

Consider a Bayesian non-parametric GMM with *Chinese Restaurant Process* (CRP) prior over the cluster assignment, $z_t \sim \text{CRP}(\alpha)$, and non-informative prior over cluster means, $\boldsymbol{\mu}_{t,i} \sim \mathcal{N}(\mathbf{0}, \varrho^2 \boldsymbol{I}_D)$. Applying SVA limit $\boldsymbol{\Sigma}_{t,j} \approx \lim_{\sigma^2 \to 0} \sigma^2 \boldsymbol{I}$ to the Gibbs sampler reduces the model to the DP-means algorithm [4].

Cluster Assignment z_{t+1}: In the online setting, the cluster assignment z_{t+1} for new datapoint $\boldsymbol{\xi}_{t+1}$ is based on the distance of the datapoint to the existing cluster means. If the minimum distance

30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

is greater than a certain threshold λ, a new cluster is initialized with that datapoint; otherwise the assigned cluster prior, mean and the corresponding number of datapoints $w_{t+1,z_{t+1}}$ are incrementally updated. We can thus write,

$$z_{t+1} = \underset{j=1:K+1}{\arg\min} \begin{cases} \|\boldsymbol{\xi}_{t+1} - \boldsymbol{\mu}_{t,j}\|_2^2, & \text{if } j \leq K \\ \lambda, & \text{otherwise.} \end{cases} \tag{1}$$

Parameters Update $\{\pi_{t+1,i}, \boldsymbol{\mu}_{t+1,i}\}$: Given the cluster assignment $z_{t+1} = i$, the parameters are updated as follows with the covariance matrix set to $\boldsymbol{\Sigma}_{t,i} = \sigma^2 \boldsymbol{I}$,

$$\pi_{t+1,i} = \frac{1}{t+1}\Big(t\pi_{t,i} + 1\Big), \qquad \boldsymbol{\mu}_{t+1,i} = \frac{1}{w_{t,i}+1}\Big(w_{t,i}\boldsymbol{\mu}_{t,i} + \boldsymbol{\xi}_{t+1}\Big), \quad w_{t+1,i} = w_{t,i} + 1. \tag{2}$$

Loss function $\mathcal{L}(z_{t+1}, \boldsymbol{\mu}_{t+1,z_{t+1}})$: The loss function optimized at time step $t+1$ is given as,

$$\mathcal{L}(z_{t+1}, \boldsymbol{\mu}_{t+1,z_{t+1}}) = \lambda K + \|\boldsymbol{\xi}_{t+1} - \boldsymbol{\mu}_{t+1,z_{t+1}}\|_2^2 \leq \ \mathcal{L}(z_{t+1}, \boldsymbol{\mu}_{t,z_{t+1}}). \tag{3}$$

Although attractive for scalability and parsimonious structure, restricting the covariance matrix parameters to a constant isotropic/spherical noise under small variance asymptotics severely limits the model from encoding important coordination patterns/variance in the streaming data.

3 Online DP-MPPCA

Consequently, we further assume that the i-th Gaussian groups the observation $\boldsymbol{\xi}_t$ in its intrinsic low-dimensional affine subspace of dimension $d_{t,i}$ with projection matrix $\boldsymbol{\Lambda}_{t,i}^{d_{t,i}} \in \mathbb{R}^{D \times d_{t,i}}$, such that $d_{t,i} < D$ and $\boldsymbol{\Sigma}_{t,i} = \boldsymbol{\Lambda}_{t,i}^{d_{t,i}} \boldsymbol{\Lambda}_{t,i}^{d_{t,i}^\top} + \sigma^2 \boldsymbol{I}$. Under this assumption, we apply the small variance asymptotic limit on the remaining $(D - d_{t,i})$ dimensions to encode the most important coordination patterns while being parsimonious in the number of parameters. Bayesian non-parametric treatment is used to alleviate the problem of model selection by placing a CRP prior over the cluster assignment z_t as before, and a hierarchical prior over the projection matrix $\boldsymbol{\Lambda}_{t,i}^{d_{t,i}}$ with an exponential prior on the subspace rank $d_{t,i} \sim r^{d_{t,i}}$ where $r \in (0, 1)$. Applying SVA limit on the resulting partially collapsed Gibbs sampler leads to an efficient deterministic algorithm for subspace clustering with an infinite MPPCA [9].

Cluster Assignment z_{t+1}: The cluster assignment z_{t+1} of $\boldsymbol{\xi}_{t+1}$ in the online case follows the same principle as in Eq. (1) except the distance is now computed from the subspace of a cluster $\text{dist}(\boldsymbol{\xi}_{t+1}, \boldsymbol{\mu}_{t,i}, \boldsymbol{U}_{t,i}^{d_{t,i}})^2 = \Big\|(\boldsymbol{\xi}_{t+1} - \boldsymbol{\mu}_{t,j}) - \rho_j \boldsymbol{U}_{t,j}^{d_{t,j}} \boldsymbol{U}_{t,j}^{d_{t,j}^\top} (\boldsymbol{\xi}_{t+1} - \boldsymbol{\mu}_{t,j})\Big\|_2^2$, defined using the difference between the mean-centered datapoint and the mean-centered datapoint projected upon the subspace $\boldsymbol{U}_{t,i}^{d_{t,i}} \in \mathbb{R}^{D \times d_{t,i}}$ spanned by the $d_{t,i}$ unit eigenvectors of the covariance matrix, i.e.,

$$z_{t+1} = \underset{j=1:K+1}{\arg\min} \begin{cases} \Big\|(\boldsymbol{\xi}_{t+1} - \boldsymbol{\mu}_{t,j}) - \rho_j \boldsymbol{U}_{t,j}^{d_{t,j}} \boldsymbol{U}_{t,j}^{d_{t,j}^\top} (\boldsymbol{\xi}_{t+1} - \boldsymbol{\mu}_{t,j})\Big\|_2^2, & \text{if } j \leq K \\ \lambda, & \text{otherwise,} \end{cases} \tag{4}$$

where, $\rho_j = \exp\Big(-\frac{\|\boldsymbol{\xi}_{t+1} - \boldsymbol{\mu}_{t,j}\|_2^2}{b_m}\Big)$ weighs the projected mean-centered datapoint according to the distance of the datapoint from the cluster center $(0 < \rho_j \leq 1)$. Its effect is controlled by the bandwidth parameter b_m. If b_m is large, then the far away clusters have a greater influence; otherwise nearby clusters are favoured. Note that ρ_j assigns more weight to the projected mean-centered datapoint for the nearby clusters than the distant clusters to limit the size of the cluster/subspace.

Parameters Update $\{\pi_{t+1,z_{t+1}}, \boldsymbol{\mu}_{t+1,z_{t+1}}, d_{t+1,z_{t+1}}, \boldsymbol{\Lambda}_{t+1,z_{t+1}}^{d_{t+1,z_{t+1}}}\}$: Given the cluster assignment $z_{t+1} = i$, the prior and mean of the assigned cluster are updated according to Eq. (2). The covariance matrix could then be updated online as

$$\bar{\boldsymbol{\Sigma}}_{t+1,i} = \frac{w_{t,i}}{w_{t,i}+1}\, \boldsymbol{\Sigma}_{t,i} + \frac{w_{t,i}}{(w_{t,i}+1)^2}\, (\boldsymbol{\xi}_{t+1} - \boldsymbol{\mu}_{t+1,i})(\boldsymbol{\xi}_{t+1} - \boldsymbol{\mu}_{t+1,i})^\top. \tag{5}$$

However, updating the covariance matrix online in D-dimensional space can be prohibitively expensive for even moderate size problems. To update the covariance matrix in its intrinsic lower

dimension, similarly to [1], we compute $\boldsymbol{g}_{t+1,i} \in \mathbb{R}^{d_i}$ as the projection of datapoint $\boldsymbol{\xi}_{t+1}$ onto the existing set of basis vectors of $\boldsymbol{U}_{t,i}^{d_{t,i}}$. Note that the cardinality of basis vectors is different for each covariance matrix. If the datapoint belongs to the subspace of $\boldsymbol{U}_{t,i}^{d_{t,i}}$, the retro-projection of the datapoint in its original space, as given by the residual vector $\boldsymbol{p}_{t+1,i} \in \mathbb{R}^D$, would be a zero vector; otherwise the residual vector belongs to the null space of $\boldsymbol{U}_{t,i}^{d_{t,i}}$, and its unit vector $\tilde{\boldsymbol{p}}_{t+1,i}$ needs to be added to the existing set of basis vectors, i.e.,

$$\begin{aligned} \boldsymbol{g}_{t+1,i} &= \boldsymbol{U}_{t,i}^{d_{t,i}{}^\top}(\boldsymbol{\xi}_{t+1} - \boldsymbol{\mu}_{t,i}), & \boldsymbol{p}_{t+1,i} &= (\boldsymbol{\xi}_{t+1} - \boldsymbol{\mu}_{t,i}) - \boldsymbol{U}_{t,i}^{d_{t,i}}\, \boldsymbol{g}_{t+1,i}, \\ \boldsymbol{U}_{t+1,i}^{d_{t,i}} &= [\boldsymbol{U}_{t,i}^{d_{t,i}}\,,\, \tilde{\boldsymbol{p}}_{t+1,i}]\, \boldsymbol{R}_{t+1,i}, & \tilde{\boldsymbol{p}}_{t+1,i} &= \begin{cases} \frac{\boldsymbol{p}_{t+1,i}}{\|\boldsymbol{p}_{t+1,i}\|_2}, & \text{if } \|\boldsymbol{p}_{t+1,i}\|_2 > 0 \\ \boldsymbol{0}_D, & \text{otherwise.} \end{cases} \end{aligned} \tag{6}$$

where $\boldsymbol{U}_{t+1,i}^{d_{t,i}} \in \mathbb{R}^{D\times(d_{t,i}+1)}$ represents the new set of basis vectors augmented with the residual unit vector $\tilde{\boldsymbol{p}}_{t+1,i}$, and $\boldsymbol{R}_{t+1,i} \in \mathbb{R}^{(d_{t,i}+1)\times(d_{t,i}+1)}$ is the rotation matrix used to incrementally update the augmented basis vectors. $\boldsymbol{R}_{t+1,i}$ is obtained by substituting the value of $\bar{\boldsymbol{\Sigma}}_{t+1,i}$ from Eq. (5) and $\boldsymbol{U}_{t+1,i}^{d_{t,i}}$ from Eq. (6) in $\bar{\boldsymbol{\Sigma}}_{t+1,i} = \boldsymbol{U}_{t+1,i}^{d_{t,i}}\, \boldsymbol{\Sigma}_{t+1,i}^{(\text{diag})}\, \boldsymbol{U}_{t+1,i}^{d_{t,i}}{}^\top$ with $\boldsymbol{\Sigma}_{t+1,i}^{(\text{diag})} \in \mathbb{R}^{(d_{t,i}+1)\times(d_{t,i}+1)}$ and solving the reduced eigendecomposition problem of size $(d_{t,i}+1)\times(d_{t,i}+1)$,

$$\frac{w_{t,i}}{w_{t,i}+1}\begin{bmatrix}\boldsymbol{\Sigma}_{t,i}^{(\text{diag})} & \boldsymbol{0}_{d_{t,i}} \\ \boldsymbol{0}_{d_{t,i}}^\top & 0\end{bmatrix} + \frac{w_{t,i}}{(w_{t,i}+1)^2}\begin{bmatrix}\boldsymbol{g}_{t+1,i}\,\boldsymbol{g}_{t+1,i}^\top & \nu_i \boldsymbol{g}_{t+1,i} \\ \nu_i \boldsymbol{g}_{t+1,i}^\top & \nu_i^2\end{bmatrix} = \boldsymbol{R}_{t+1,i}\, \boldsymbol{\Sigma}_{t+1,i}^{(\text{diag})}\, \boldsymbol{R}_{t+1,i}^\top, \tag{7}$$

where $\nu_i = \tilde{\boldsymbol{p}}_{t+1,i}^\top(\boldsymbol{\xi}_{t+1} - \boldsymbol{\mu}_{t+1,i})$. Solving for $\boldsymbol{R}_{t+1,i}$ and substituting it in Eq. (6) gives the required update of the basis vectors $\boldsymbol{U}_{t+1,i}^{d_{t+1,i}}$ in a computationally and memory efficient manner. The subspace dimension of the i-th mixture component is updated by keeping an estimate of the average distance vector $\bar{\boldsymbol{e}}_{t,i} \in \mathbb{R}^D$ whose k-th element represents the mean distance of the datapoints to the $(k-1)$ subspace basis vectors of $\boldsymbol{U}_{t,i}^k$ for the i-th cluster. Let us denote by $\boldsymbol{\delta}_i$ as the vector measuring the distance of the datapoint $\boldsymbol{\xi}_{t+1}$ to each of the subspaces of $\boldsymbol{U}_{t,i}^k$ for the i-th cluster where $k = \{0 \ldots (d_{t,i}+1)\}$, i.e.,

$$\boldsymbol{\delta}_i = \Big[\text{dist}(\boldsymbol{\xi}_{t+1}, \boldsymbol{\mu}_{t+1,i}, \boldsymbol{U}_{t+1,i}^0)^2 \ldots \text{dist}(\boldsymbol{\xi}_{t+1}, \boldsymbol{\mu}_{t+1,i}, \boldsymbol{U}_{t+1,i}^{d_{t,i}+1})^2\Big]^\top, \tag{8}$$

where $\text{dist}(\boldsymbol{\xi}_{t+1}, \boldsymbol{\mu}_{t+1,i}, \boldsymbol{U}_{t+1,i}^0)^2$ is the distance to the cluster subspace with 0 dimensions (the cluster mean), $\text{dist}(\boldsymbol{\xi}_{t+1}, \boldsymbol{\mu}_{t+1,i}, \boldsymbol{U}_{t+1,i}^1)^2$ is the distance to the cluster subspace with 1 dimension (the line), and so on. The average distance vector $\bar{\boldsymbol{e}}_{t+1,i}$, the subspace dimension $d_{t+1,i}$, the projection matrix $\boldsymbol{\Lambda}_{t+1,i}^{d_{t+1,i}}$, and the covariance matrix $\boldsymbol{\Sigma}_{t+1,i}$ are updated as,

$$\bar{\boldsymbol{e}}_{t+1,i} = \frac{1}{w_{t,i}+1}\Big(w_{t,i}\bar{\boldsymbol{e}}_{t,i} + \boldsymbol{\delta}_i\Big), \qquad d_{t+1,i} = \underset{d=0:D-1}{\arg\min}\Big\{\lambda_1 d + \bar{\boldsymbol{e}}_{t+1,i}\Big\}, \tag{9}$$

$$\boldsymbol{\Lambda}_{t+1,i}^{d_{t+1,i}} = \boldsymbol{U}_{t+1,i}^{d_{t+1,i}}\, \sqrt{\boldsymbol{\Sigma}_{t+1,i}^{(\text{diag})}}, \qquad \boldsymbol{\Sigma}_{t+1,i} = \boldsymbol{\Lambda}_{t+1,i}^{d_{t+1,i}}\, \boldsymbol{\Lambda}_{t+1,i}^{d_{t+1,i}}{}^\top + \sigma^2 \boldsymbol{I}. \tag{10}$$

Loss function $\mathcal{L}(z_{t+1}, d_{t+1,z_{t+1}}, \boldsymbol{\mu}_{t+1,z_{t+1}}, \boldsymbol{U}_{t+1,z_{t+1}}^{d_{t+1,z_{t+1}}})$: The loss function at time step $t+1$ is,

$$\begin{aligned} \mathcal{L}(z_{t+1}, d_{t+1,z_{t+1}}, \boldsymbol{\mu}_{t+1,z_{t+1}}, \boldsymbol{U}_{t+1,z_{t+1}}^{d_{t+1,z_{t+1}}}) &= \lambda K + \lambda_1 d_{t+1,z_{t+1}} + \text{dist}(\boldsymbol{\xi}_{t+1}, \boldsymbol{\mu}_{t+1,z_{t+1}}, \boldsymbol{U}_{t+1,z_{t+1}}^{d_{t+1,z_{t+1}}})^2 \\ &\leq \mathcal{L}(z_{t+1}, d_{t,z_{t+1}}, \boldsymbol{\mu}_{t,z_{t+1}}, \boldsymbol{U}_{t,z_{t+1}}^{d_{t,z_{t+1}}}). \end{aligned}$$

The loss function provides an intuitive trade-off between the fitness term $\text{dist}(\boldsymbol{\xi}_{t+1}, \boldsymbol{\mu}_{t+1,z_{t+1}}, \boldsymbol{U}_{t+1,z_{t+1}}^{d_{t+1,z_{t+1}}})^2$ and the model selection parameters K and d_k. Increasing the number of clusters or the subspace dimension of the assigned cluster decreases the distance of the datapoint to the assigned subspace at the cost of penalty terms λ and λ_1. Parameters of the assigned cluster are updated in a greedy manner such that the loss function is guaranteed to decrease at the current time step. In case a new cluster is assigned to the datapoint, the loss function at time t is evaluated with the cluster having the lowest cost among the existing set of clusters. Note that setting $d_{t,i} = 0$ by choosing $\lambda_1 \gg 0$ gives the same loss function and objective function as the online DP-GMM algorithm with isotropic Gaussians.

4 Results, Discussions and Conclusions

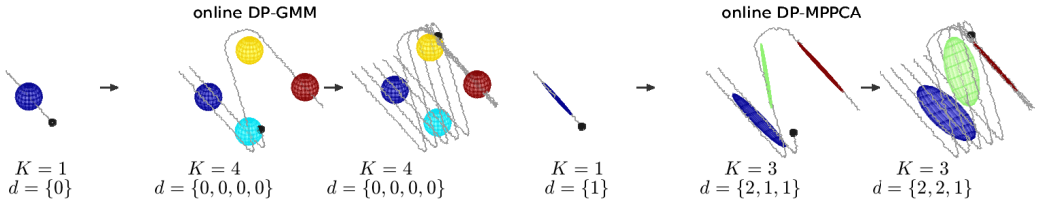

Figure 1: Non-parametric online clustering of Z-shaped streaming data under small variance asymptotics with: *(left)* online DP-GMM, *(right)* online DP-MPPCA.

We first evaluate the performance of the algorithms on a Z-shaped 3-dimensional stream of datapoints with penalty parameters $\{\lambda = 35, \sigma^2 = 100\}$ for online DP-GMM, and $\{\lambda = 14, \lambda_1 = 2, \sigma^2 = 1, b_m = 1 \times 10^4\}$ for online DP-MPPCA. Fig. 1 shows that online DP-GMM under small variance asymptotics fails to represent the variance in the demonstrations with $d = 0$, whereas the number of clusters and the subspace dimension adequately evolves for online DP-MPPCA to model the underlying distribution. We then consider a robotic application of performing remote manipulation tasks by teleoperation. We use the Baxter robot as a mock-up for teleoperation where the left arm is used as the input device of the teleoperator and the right arm is used to perform the task of tracking a movable screwdriver target by teleoperation. Here, we learn a task-parameterized generative model [6] online to assist the teleoperator in performing the task based on the variance observed in the teleoperator demonstrations. We use the model to adjust the robot movement towards low variance segments of the demonstrations as observed from the frame of reference of the target. After 6 demonstrations of reaching different target poses from different initial configurations, the learned model contains 3 clusters of subspace dimensions $\{4, 3, 4\}$ with $D = 14$ using penalty parameters $\{\lambda = 0.65, \lambda_1 = 0.05, \sigma^2 = 2.5 \times 10^{-4}, b_m = 100\}$ (see Fig. 2 for segmented clusters and evolution of the model parameters during learning). Note that if the cluster evolves such that it is closer to another cluster than threshold λ, the two clusters are merged into one and the subspace of the dominant cluster is retained.

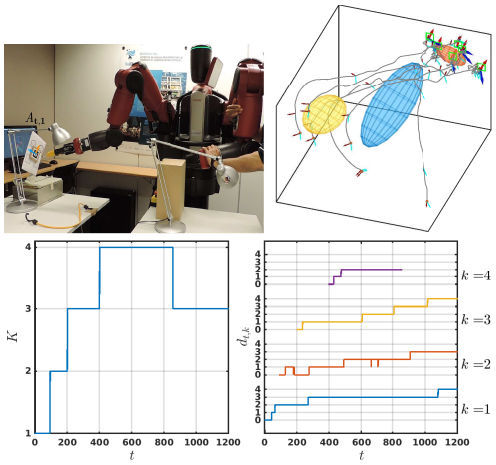

Figure 2: Teleoperation with learned clusters on top and evolution of K and $d_{t,k}$ on bottom.

Online learning with DP-MPPCA under SVA does not require computation of numerically unstable gradients at each iteration and scales well to high-dimensional spaces with its simple deterministic updates [8]. Non-parametric treatment aids the user to build the model online without specifying the number of clusters and the subspace dimension of each cluster, as the parameter set grows with the size of data during learning. The penalty parameters introduced are more intuitive to specify and act as regularization terms for model selection based on the structure of the data. Note that the order of streaming data plays an important role during learning, and multiple starts from different initial configurations lead to different solutions. Alternate strategies to avoid different solutions include initializing the parameters with a batch algorithm, or updating the parameters sequentially in a mini-batch manner [3]. The temporal information in the data is incorporated in the model by online estimation of the state transition and the state duration information in a hidden semi-Markov model based on hierarchical Dirichlet process (see [7] for more details).

In this paper, we have presented a non-parametric clustering algorithm by online inference in DP-GMM and DP-MPPCA under small variance asymptotics. The algorithm incrementally clusters the streaming data with non-parametric locally linear principal component analysis whose redundant dimensions are discarded autonomously by small variance asymptotics. We showed that the model efficiently encodes the demonstrations to teach new skills to robots in an online non-parametric manner.

Acknowledgement

This work is in part supported by the DexROV project through the EC Horizon 2020 programme (Grant #635491).

References

[1] Anastasios Bellas, Charles Bouveyron, Marie Cottrell, and Jérôme Lacaille. Model-based clustering of high-dimensional data streams with online mixture of probabilistic pca. *Advances in Data Analysis and Classification*, 7(3):281–300, 2013.

[2] Tamara Broderick, Brian Kulis, and Michael I. Jordan. Mad-bayes: Map-based asymptotic derivations from bayes. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, pages 226–234, 2013.

[3] Trevor Campbell, Miao Liu, Brian Kulis, Jonathan P. How, and Lawrence Carin. Dynamic clustering via asymptotics of the dependent dirichlet process mixture. In Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger, editors, *NIPS*, pages 449–457, 2013.

[4] Brian Kulis and Michael I. Jordan. Revisiting k-means: New algorithms via bayesian nonparametrics. In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, pages 513–520, New York, NY, USA, 2012. ACM.

[5] Anirban Roychowdhury, Ke Jiang, and Brian Kulis. Small-variance asymptotics for hidden markov models. In *Advances in Neural Information Processing Systems 26*, pages 2103–2111. Curran Associates, Inc., 2013.

[6] Ajay Kumar Tanwani and Sylvain Calinon. Learning robot manipulation tasks with task-parameterized semitied hidden semi-markov model. *Robotics and Automation Letters, IEEE*, 1(1):235–242, 2016.

[7] Ajay Kumar Tanwani and Sylvain Calinon. Small variance asymptotics for non-parametric online robot learning. *CoRR*, abs/1610.02468, 2016.

[8] S. Vijayakumar, A. D'souza, and S. Schaal. Incremental online learning in high dimensions. *Neural Computation*, 17(12):2602–2634, 2005.

[9] Yining Wang and Jun Zhu. DP-space: Bayesian nonparametric subspace clustering with small-variance asymptotics. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pages 862–870, 2015.