`GURLS_mkl`: A PFBS-based Implementation for Multiple Kernel Learning

(Jeremiah) Zhe Liu
Department of Biostatistics
Harvard University
Boston, MA 02115
`zhl112@mail.harvard.edu`

Contents

1 Hierarchical Dirichlet Process

1.1 Model

Classic view:

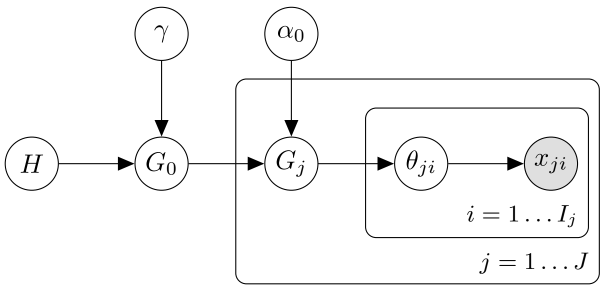
$$
\begin{aligned}
G_0|\gamma, H &\sim DP(\gamma, H) \\
G_j|\alpha_0, G_0 &\sim DP(\alpha_0, G_0) \\
\theta_{ji}|G_j &\sim G_j \\
x_{ji}|\theta_{ji} &\sim F(\theta_{ji})
\end{aligned}
$$

where $P \sim DP(\alpha, G)$ adopts the stick breaking representation w.p. 1:

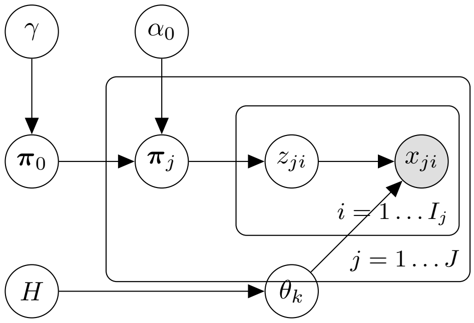
$$
P = \sum_{k=1}^{\infty} \pi_k \delta_{\phi_k} \qquad \text{where:} \qquad \pi_k \sim GEM(\alpha), \quad \phi_k \sim G
$$

Alternatively, one may describe the generative processes of π_k and θ_k separately as:

$$
\begin{aligned}
\boldsymbol{\pi}_0|\gamma &\sim GEM(\gamma) & \qquad \theta_k|H \sim H \\
\boldsymbol{\pi}_j|\alpha_0, \boldsymbol{\pi}_0 &\sim DP(\alpha_0, \boldsymbol{\pi}_0) & \\
& & \\
z_{ji}|\boldsymbol{\pi}_j &\sim \boldsymbol{\pi}_j & \\
x_{ji}|z_{ji}, (\theta_k)_{k=1}^{\infty} &\sim F(\theta_{z_{ji}}) &
\end{aligned}
$$



(a) Hierarchical Dirichlet Process



(b) Hierarchical Dirichlet Process

Figure 1: Hierarchical Dirichlet Process

1.2 Inference

Assuming conjugacy between H and F [1] and holding (γ, α_0) fixed,

we now describe a simplied Gibbs approach to sample parameters $(z_{ji}, m_{jk}, \boldsymbol{\pi}_0)$ from the Chinese Restaurant Franchise (see Appendix A) representation of the posterior, where the parameter z_{ji} are refered to respectively as customer-specific dish assignment, m_{jk} as dish-specific table count, and π_0 as global dish distribution. This

[1] so we can integrate out the mixture component parameters

particular method is referred to as "direct assignment" in Teh et al. [2006] since it circumvented the issue of bookkeeping for every t_{ij} (customer-specific table assignment) and k_{jt} (table-specific dish assignment) variables.

In each Gibbs iteration, denote $f_k^{-x_{ji}}(x_{ji}) = \frac{\int f(\mathbf{x}|\theta_k)h(\theta_k)d_{\theta_k}}{\int f(\mathbf{x}_{-(ji)}|\theta_k)h(\theta_k)d_{\theta_k}}$ the conditional distribution $x_{ji}|\mathbf{x}_{-(ji)}$ under $\theta = \theta_k$, and assume there are currently K dishes and T tables, we sample $(z_{ji}, m_{jk}, \boldsymbol{\pi}_0)$ iteratively as:

1. Sample $z_{ji} = k|\mathbf{z}_{-(ji)}, \mathbf{m}, \boldsymbol{\pi}_0$ from the distribution:

$$z_{ji} = k|\mathbf{z}_{-(ji)}, \mathbf{m}, \boldsymbol{\pi}_0 \propto \begin{cases} f_k^{-x_{ji}}(x_{ji}) * (n_{jk}^{-(ji)} + \alpha_0\pi_{0,k}) & k \leq K \\ f_{K+1}^{-x_{ji}}(x_{ji}) * \alpha_0\pi_{0,u} & k = K+1 \end{cases}$$

2. Sample $m_{jk} = m|\mathbf{z}, \mathbf{m}_{-(jk)}, \boldsymbol{\pi}_0$, by setting $m_{jk} = \sum_i I(t_{ji} = t_{new}|k_{jt_{new}} = k)$, we can sample t_{ji} from:

$$t_{ji} = t|k_{jt} = k, \mathbf{t}_{-(ji)}, \boldsymbol{\pi}_0 \propto \begin{cases} n_{jt}^{-(ji)} & t \leq T \\ \alpha_0\pi_{0,k} & t = T+1 \end{cases}$$

and as in Fox [2009], sample $I(t_{ji} = t_{new}|k_{jt_{new}} = k)$ directly from:

$$Bern\Big(\frac{\alpha_0\pi_{0,k}}{n_{jk} + \alpha_0\pi_{0,k}}\Big)$$

3. Sample $\boldsymbol{\pi}_0$ from distribution:

$$\boldsymbol{\pi}_0 \sim Dir(m_1, \ldots, m_K, \gamma)$$

Algorithm 1 HDP, Gibbs Sampler through Direct Assignment

procedure `hdp_gibbs_ds`($\mathbf{K}$, $\mathbf{y}$, (τ, μ, σ))
  $\boldsymbol{\alpha}^0 = \mathbf{0}$
  for $p = 1$ to `MAX_ITER` **do**
    $\boldsymbol{\alpha}_0^p = (1 - \frac{\mu}{\sigma})\boldsymbol{\alpha}^{p-1} - \frac{1}{\sigma n}(\mathbf{K}\boldsymbol{\alpha}^{p-1} - \mathbf{y})$
    $\boldsymbol{\alpha}^p = \mathbf{S}_{\frac{\tau}{\sigma}}(K, \boldsymbol{\alpha}_0^p)$
  end for
  return $f^{\texttt{MAX\_ITER}} = (\boldsymbol{\alpha}^{\texttt{MAX\_ITER}})^T\mathbf{k}$
end procedure

1.3 Application: Clustering Hierarchical Gaussian Data

Consider mixture of Gaussian data $\mathbf{x} = \{\mathbf{x}_1, \ldots, \mathbf{x}_K\}$ with $\mathbf{x}_k \overset{iid}{\sim} MVN(\boldsymbol{\theta}_{k,2\times1}, \mathbf{I}_{2\times2})$ with unknown mean $\boldsymbol{\theta}$. Assuming diffused Gaussian prior $\boldsymbol{\theta} \sim N(\mathbf{0}, \sigma^2\mathbf{I})$, the form of likelihood F and base measure H are:

$$f(x_{ji}|\boldsymbol{\theta}_k) \propto exp(-\frac{1}{2\sigma^2}(x_{ji} - \boldsymbol{\theta}_k)^T(x_{ji} - \boldsymbol{\theta}_k))$$
$$h(\boldsymbol{\theta}_k) \propto exp(-\frac{1}{2\sigma_0^2}\boldsymbol{\theta}_k^T\boldsymbol{\theta}_k)$$

Then $f_k^{-x_{ji}}(x_{ji})$ should be:

$$f_k^{-x_{ji}}(x_{ji}) \sim N(\frac{n_k^{-(ji)}\sigma_0^2}{n_k^{-(ji)}\sigma_0^2 + \sigma^2}\bar{\mathbf{x}}_k^{-(ji)}, (1 + \frac{\sigma_0^2}{n_k^{-(ji)}\sigma_0^2 + \sigma^2})\mathbf{I})$$

2 HDP for Hidden Markov Model

2.1 Hidden Markov Model

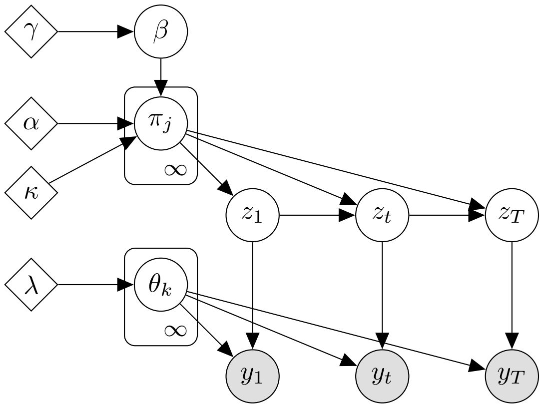


Figure 2: Hidden Markov Model

$$
\begin{aligned}
\beta|\gamma &\sim GEM(\gamma) \\
\pi_j|\beta, \alpha &\sim DP(\alpha, \beta) \\
\theta_k|H, \lambda &\sim H(\lambda) \\
\\
z_t|z_{t-1}, \boldsymbol{\pi} &\sim \pi_{z_{t-1}} \\
y_t|z_t, \boldsymbol{\theta} &\sim F(\theta_{z_t})
\end{aligned}
$$

$$f_k(y_t) = p(y_t|\boldsymbol{\theta}_{z_t})p(z_t|z_{t-1})$$

2.2 Sticky HDP

Though flexible, the fact that HDP-HMM is deploying $\pi_k \sim DP(\alpha, \beta)$ leads to:

1. large posterior probability for unrealistically transition dynamics
2. once instantiated, the unrealistically transition dynamics will be reinforced by CRF

Sticky HDP address above issues by encouraging self-transition. More specifically, the base measure for π_k is augmented *a priori* from β to:

$$\pi_j \sim DP(\alpha + \kappa, \frac{\alpha\beta + \kappa\delta_j}{\alpha + \kappa})$$

2.3 Inference

Inference for HMM with Sticky HDP prior follows the sticky extension of CRF. For a observation y_t at time t, "restaurant" corresponds to the state z_t that y_t is at, and dishes at restaurant z_t indicates the potential states that y_{t+1} can transit to. To improve mixing rate of state sequence $\mathbf{z}$, we deploy the blocked sampler which uses a weak limit approximation of the infinite-dimension DP prior. More specifically, we assume there are L states, and β and π follows:

$$\beta|\gamma \sim Dir(\frac{\gamma}{L}, \ldots, \frac{\gamma}{L})$$
$$\pi_j|\alpha, \beta, \kappa \sim Dir(\alpha\beta_1, \ldots, \alpha\beta_j + \kappa, \ldots, \alpha\beta_L)$$

Define $\boldsymbol{\theta}_k$ as emission parameter for state k, we sample $(\mathbf{z}, \mathbf{m}, \boldsymbol{\pi}_0, \boldsymbol{\theta})$ as follows:

1. Sample $z_{ji} = k|\mathbf{z}_{-(ji)}, \mathbf{m}, \boldsymbol{\pi}_0$ from the distribution:

$$z_{ji} = k|\mathbf{z}_{-(ji)}, \mathbf{m}, \boldsymbol{\pi}_0 \propto \begin{cases} f_k^{-x_{ji}}(x_{ji}) * (n_{jk}^{-(ji)} + \alpha_0\pi_{0,k}) & k \leq K \\ f_{K+1}^{-x_{ji}}(x_{ji}) * \alpha_0\pi_{0,u} & k = K+1 \end{cases}$$

2. Sample $m_{jk} = m|\mathbf{z}, \mathbf{m}_{-(jk)}, \boldsymbol{\pi}_0$, by setting $m_{jk} = \sum_i I(t_{ji} = t_{new}|k_{jt_{new}} = k)$, we can sample t_{ji} from:

$$t_{ji} = t|k_{jt} = k, \mathbf{t}_{-(ji)}, \boldsymbol{\pi}_0 \propto \begin{cases} n_{jt}^{-(ji)} & t \leq T \\ \alpha_0\pi_{0,k} & t = T+1 \end{cases}$$

and as in Fox [2009], sample $I(t_{ji} = t_{new}|k_{jt_{new}} = k)$ directly from:

$$Bern\Big(\frac{\alpha_0\pi_{0,k}}{n_{jk} + \alpha_0\pi_{0,k}}\Big)$$

3. Sample $\boldsymbol{\pi}_0$ from distribution:

$$\boldsymbol{\pi}_0 \sim Dir(m_1, \ldots, m_K, \gamma)$$

2.3.1 Forward-backward Message Passing

In a general graphic model belief propagation algorithm, *message* is defined as the amount of information passed from neighborhood nodes toward the target node though connected edges. In mathematics, a message passed from node i to j is:

$$m_{ij}(z_j) = \int_{\mathcal{Z}_i} \phi_i(z_i)\psi_{ij}(z_i, z_j)m_i(z_i)d_{z_i}$$

In the context of HMM, the $m'_{ij}s$ can be expressed in exact term, where:

$$m_{12}z_2 = \int_{\mathcal{Z}_1} \phi_1(z_1)\psi_{ij}(z_1, z_2)d_{z_1} = \int_{\mathcal{Z}_1} p(z_1|y_1)p(z_2|z_1)d_{z_1} =$$

The forward-backward algorithm provide an efficient method for computing node marginals $p(y_t)$. Define:

$$\begin{aligned} \text{Backward Message}: &\quad \beta_t(z_t) = p(\mathbf{y}_{T>t}|z_t) \\ \text{Forward Message}: &\quad \alpha_t(z_t) = p(\mathbf{y}_{T\leq t}, z_t) \\ \text{Joint Message}: &\quad \alpha_t(z_t)\beta_t(z_t) = p(\mathbf{y}, z_t) \end{aligned}$$

which can be alternatively defined using message m_{t_1,t_2}

$$\begin{aligned} \text{Backward Message}: &\quad \beta_t(z_t) = p(\mathbf{y}_{T>t}|z_t) = m_{t+1,t}(z_t) \\ \text{Forward Message}: &\quad \alpha_t(z_t) = p(y_t|z_t)p(\mathbf{y}_{T<t}, z_t) = p(y_t|z_t)m_{t-1,t}(z_t) \end{aligned}$$

.

These two types of messages can be computed β_t backward and α_t forward in time as:

$$\begin{aligned} \beta_{t-1} &= \sum_{z_t} p(y_t|z_t) \qquad p(z_t|z_{t-1})\beta_t(z_t) \qquad \text{with} \quad \beta_T(z_T) = 1 \\ \alpha_{t+1} &= \sum_{z_t} p(y_{t+1}|z_{t+1})p(z_{t+1}|z_t)\alpha_t(z_t) \qquad \text{with} \quad \alpha_1(z_1) = p(y_1, z_1) = p(y_1|z_1)\pi^0(z_1) \end{aligned}$$

Using Forward and Backward messages, we can compute state assignment posterior as:

$$p(z_t|\mathbf{y}) = \frac{p(z_t, \mathbf{y})}{\sum_{z_t} p(z_t, \mathbf{y})} = \frac{\alpha_t(z_t)\beta_t(z_t) = p(\mathbf{y}, z_t)}{\sum_{z_t} \alpha_t(z_t)\beta_t(z_t) = p(\mathbf{y}, z_t)}$$

A Chinese Restaurant Franchise

A hierarchical analogy of Chinese Restaurant Process, the Chinese Restaurant Franchise offers a convenient scheme to sample from the posterior of cluster-specific θ's in HDP. This process draw below analogy:

- H as the dish distribution for all possible dishes in the world, with the types of possible dishes being $(\theta_k)_{k=1}^{\infty}$.
- $G_0 \sim DP(\gamma, H)$ as the dish distribution for the franchise
- $G_j \sim DP(\alpha_0, G_0)$ as the dish distribution for restaurant j in the franchise
- $\psi_{jt} \sim G_0$ as the dish served at table t in restaurant j.
  $k_{jt} \sim \pi_0$ as the index of dish choice for this table.
- $\theta_{ji} \sim G_j$ as the dish will be enjoyed by customer i in restaurant j.
  $t_{ji} \sim \pi_j$ as the index of table choice for this customer.

Integrating over G_j, the sampling scheme for subject-specific dish $\theta_{ji} \sim G_j$ is:

$$\theta_{ji}|\boldsymbol{\theta}_{j(-i)}, \alpha_0, G_0 \sim \sum_{k=1}^{K} \frac{n_{jt.}}{\alpha_0 + n_{j..}} \delta_{\psi_{jt}} + \frac{\gamma}{n_{j..} + \gamma} G_0$$

Integrating over G_0, the sampling scheme for table-specific dish $\psi_{jt} \sim G_0$ is:

$$\psi_{jk}|\Psi_{j(-k)}, \gamma, H \sim \sum_{k=1}^{K} \frac{m_{.k}}{\gamma + m_{..}} \delta_{\theta_k} + \frac{\gamma}{m_{..} + \gamma} H$$

B References

References

Yee Whye Teh, Michael I. Jordan, Matthew J. Beal, and David M. Blei. Hierarchical Dirichlet Processes. *Journal of the American Statistical Association*, 101(476):1566–1581, 2006. ISSN 0162-1459. URL `http://www.jstor.org/stable/27639773`.

Emily Beth Fox. *Bayesian nonparametric learning of complex dynamical phenomena*. Thesis, Massachusetts Institute of Technology, 2009. URL `http://dspace.mit.edu/handle/1721.1/55111`.