Motivation

Big data prefer

large-capacity models like deep latent variable models (LVMs)

scalable inference methods like SG-MCMC

However, to make a deep LVM scalable is challenging, because

gradients of model parameters are difficult to compute

different layers may be suitable for different learning rates

Most existing methods adopt greedy layer-wise training, which

usually uses a shared learning rates across all layers

has no communication between different layers

We develop a scalable SG-MCMC algorithm for deep latent Dirichlet allocation that jointly learns all hidden layers.

Background

Poisson Gamma Belief Network (PGBN)

\[
\begin{align*}
\theta_j^{(L)} &\sim \text{Gam}(r, 1/c_j^{(L+1)}), \\
\theta_j^{(l)} &\sim \text{Gam}\left(\theta_j^{(l+1)} \theta_j^{(L)}, 1/c_j^{(l+1)}\right), \\
x_j^{(l)} &\sim \text{Pois}\left(\Phi(x_j^{(L)} \theta_j^{(l)} \right), \\
\Phi_j^{(l)} &\sim \text{Gamma}\left(\text{Gam}(r, 1/c_j^{(l+1)})\right),
\end{align*}
\]

Stochastic Gradient MCMC

For unknown \( z \) obeying \( p(z \mid X) \propto e^{-H(z)} \), one has a mini-batch update rule as

\[
z_{l+1} = z_l + \varepsilon_l \left( - D(z_l) + Q(z_l) \nabla H(z_l) + \Gamma(z_l) \right)
\]

\[
+ X_l \left( \varepsilon_l \varepsilon_0 D(z_l) - \alpha B_l \right),
\]

where \( D(z) \) is a positive semidefinite matrix, \( Q(z) \) is a skew-symmetric matrix,

\[
\Gamma(z) = \sum_{l=1}^{L} \nabla H(z) + Q(z), \quad \nabla H(z) = - \ln p(z) - \rho \sum_{l=1}^{L} \ln p(z_l),
\]

\( \alpha \) is the SN Gaussian variance estimate.

Contributions

To develop principled SG-MCMC for the PGBN, with \( z = \{ \Phi^{(1)}, \ldots, \Phi^{(L)}, r \} \) and simple settings \( D(z) = G(z) \), \( Q(z) = 0 \), and \( B_l = 0 \), all we need are (a) the Fisher information matrix (FIM) \( G(z) \) and (b) \( \nabla H(z) \).

Directly computing the FIM of PGBN is intractable. But an alternative view of the PGBN makes it straightforward.

Deep Latent Dirichlet Allocation (DLDA)

Exploiting data augmentation and marginalization, one may re-express the hierarchical model of the PGBN as DLDA as

\[
x_j^{(l)} \sim \text{Log}\left(\gamma\right), \quad K_L \sim \text{Pois}\left(\frac{\gamma}{r} \ln(1 - \beta)\right),
\]

\[
x^{(L-1)} = \sum_{k=1}^{K_{L}} \sum_{j=1}^{N_{k}} x_{ij}^{(L)} \theta_{ij}^{(L)}
\]

\[
x_{ij}^{(L)} \sim \text{Mult}\left( \left[ m_{ij}^{(L)}(\theta_{ij}^{(L)}) \right], \left[ \Phi_{ij}^{(L)} \right] \right),
\]

\[
x_{ij}^{(l)} \sim \text{Mult}\left( \left[ m_{ij}^{(l)}(\theta_{ij}^{(l)}) \right], \left[ \Phi_{ij}^{(l)} \right] \right),
\]

\[
x_{ij}^{(1)} = \sum_{k=1}^{K_{1}} x_{ik}^{(1)} \theta_{ik}^{(1)} \sim \text{Mult}\left( \left[ m_{ik}^{(1)}(\theta_{ik}^{(1)}) \right], \left[ \Phi_{ik}^{(1)} \right] \right),
\]

ICML 2017

Analytical and Practical Fisher Information Matrix

Under the alternative DLDA representation, one may readily derive a block-diagonal and thus easily-inverted Fisher information matrix (FIM) as

\[
G(z) = \text{diag}\left(I \left( \phi^{(1)} \right), \ldots, I \left( \phi^{(L)} \right), I(z) \right).
\]

Note the FIM, playing a similar role as the Hessian matrix in optimization, enables principled joint inference of the PGBN (DLDA).

Reduced-Mean Inference on the Probability Simplices

For the interested batch posterior \( \phi^* \sim \text{Dir}(x_1 + \eta, \ldots, x_N + \eta) \) on the probability simplex, one may have the mini-batch-based inference as

\[
\phi^*_j = \left[ \phi_j + \varepsilon_j \frac{1}{M_i} \left( (\mu_j + \eta) - (\mu_j + \eta) \right) \right] + \mathcal{N}\left( 0, \frac{2\varepsilon_j}{M_i} \text{diag}(\phi_j) \right),
\]

which is derived with the reduced-mean parameterization \( \phi_j \) of \( \phi_j \), namely

\[
\phi_j = \left( \phi_j^T + 1 - \sum \phi_j^T \right)^{-1} \phi_j^T, \quad \text{and Theorem 2 of [1].}
\]

Topic-Layer-Adaptive Stochastic Gradient Riemannian MCMC

Algorithm 1 TLASGR MCMC for DLDA (PGBN).

Input:

Data mini-batches;

Output:

Global parameters of DLDA (PGBN).

1: for \( l = 1, \ldots, L \) do

2: \( \Phi^l \) Collect local information

3: Upward-downward Gibbs sampling (Zhu et al., 2016a) on the \( l \)-th mini-batch for \( \Phi^l \), \( \phi^l \), \( \Phi^{L+1} \), and \( \Phi^{L+1} \);

4: \( \Phi^l \) Update global parameters

5: for \( l = 1, \ldots, L \) and \( k = 1, \ldots, K_l \) do

6: Update \( M_{jk} \) with (15); then \( \phi_j^k \) with (15);

7: end for

8: Update \( \lambda^{(L+1)} \) with (19) and then \( \nu \) with (17).

9: end for

Joyal Chong, Bo Chen, Hongwei Liu, Mingyuan Zhou

Department, The University of Texas at Austin, Austin, TX, USA