Statistical Models & Computing Methods

Lecture 14: Stochastic Variational Inference

Cheng Zhang

School of Mathematical Sciences, Peking University

November 18, 2024

- Mean-field VI can be slow when the data size is large.
- Moreover, the conditional conjugacy required by mean-field VI greatly reduces the general applicability of the method.
- Fortunately, as an optimization approach, VI allows us to easily combine it with various scalable optimization methods.
- In this lecture, we will introduce some of the recent advancements on scalable variational inference, both for mean-field VI and more general VI.

Mean-field VI Could Be Data-inefficient

- A generic class of models

$$p(\beta, z, x) = p(\beta) \prod_{i=1}^{n} p(z_i, x_i | \beta)$$

- The mean-field approximation

$$q(\beta, z) = q(\beta|\lambda) \prod_{i=1}^{n} q(z_i|\phi_i)$$

- Coordinate ascent could be data-inefficient

$$\lambda^* = \mathbb{E}_{q(z)}(\eta_g(x, z)), \quad \phi_i^* = \mathbb{E}_{q(\beta)}(\eta_\ell(x_i, \beta))$$

  - Requires local computation for each data points.
  - Aggregate these computation to update the global parameter.

Gradients of The ELBO

- Recall that the λ-ELBO (update to a constant) is

$$L(\lambda) = \nabla_\lambda A_g(\lambda)^\top \left(\alpha + \sum_{i=1}^{n} \mathbb{E}_{\phi_i}(T(z_i, x_i)) - \lambda \right) + A_g(\lambda)$$

- Differentiating this w.r.t. λ yields

$$\nabla_\lambda L(\lambda) = \nabla_\lambda^2 A_g(\lambda) \left(\alpha + \sum_{i=1}^{n} \mathbb{E}_{\phi_i}(T(z_i, x_i)) - \lambda \right)$$

- Similarly

$$\nabla_{\phi_i} L(\phi_i) = \nabla_{\phi_i}^2 A_\ell(\phi_i) \left(\mathbb{E}_\lambda(\eta_\ell(x_i, \beta)) - \phi_i \right)$$

Natural Gradient

- The gradient of f at λ, $\nabla_\lambda f(\lambda)$ points in the same direction as the solution to

$$\arg\max_{d\lambda} f(x + d\lambda), \quad s.t. \ \|d\lambda\|^2 \leq \epsilon^2$$

for sufficiently small ϵ.

- The gradient direction implicitly depends on the Euclidean distance, which might not capture the distance between the parameterized probability distribution $q(\beta|\lambda)$.
- We can use *natural gradient* instead, which points in the same direction as the solution to

$$\arg\max_{d\lambda} f(x + d\lambda), \quad s.t. \ \mathrm{D}_{\mathrm{KL}}^{\mathrm{sym}}(q(\beta|\lambda), q(\beta|\lambda + d\lambda)) \leq \epsilon$$

for sufficiently small ϵ, where $\mathrm{D}_{\mathrm{KL}}^{\mathrm{sym}}$ is the symmetrized KL divergence.

Natural Gradient

- We manage the symmetrized KL divergence constraint with a Riemannian metric $G(\lambda)$

$$\mathrm{D}_{\mathrm{KL}}^{\mathrm{sym}}(q(\beta|\lambda), q(\beta|\lambda + d\lambda)) \approx d\lambda^\top G(\lambda) d\lambda$$

as $d\lambda \to 0$. G is the **Fisher information** matrix of $q(\beta|\lambda)$

$$G(\lambda) = \mathbb{E}_\lambda \left((\nabla_\lambda \log q(\beta|\lambda))(\nabla_\lambda \log q(\beta|\lambda))^\top \right)$$

- The **natural gradient** (Amari, 1998)

$$\hat{\nabla}_\lambda f(\lambda) \triangleq G(\lambda)^{-1} \nabla_\lambda f(\lambda)$$

- When $q(\beta|\lambda)$ is in the prescribed exponential family

$$G(\lambda) = \nabla_\lambda^2 A_g(\lambda)$$

- The natural gradient of the ELBO

$$\nabla_{\lambda}^{\text{nat}} L = \left(\alpha + \sum_{i=1}^{n} \mathbb{E}_{\phi_i}(T(z_i, x_i))\right) - \lambda$$
$$\nabla_{\phi_i}^{\text{nat}} L = \mathbb{E}_{\lambda}(\eta_\ell(x_i, \beta)) - \phi_i$$

  Classical coordinate ascent can be viewed as natural gradient descent with step size one

- Use the noisy natural gradient instead

$$\hat{\nabla}_{\lambda}^{\text{nat}} L(\lambda) = \alpha + n\mathbb{E}_{\phi_j}(T(z_j, x_j)) - \lambda, \quad j \sim \text{Uniform}(1, \ldots, n)$$

- This is a good noisy gradient
  - The expectation is the exact gradient (unbiased).
  - Depends merely on optimized local parameters (cheap).

Input: data $\mathbf{x}$, model $p(\beta, \mathbf{z}, \mathbf{x})$.

Initialize λ randomly. Set ρ_t appropriately.

repeat

Sample $j \sim \text{Unif}(1, \ldots, n)$.

Set local parameter $\phi \leftarrow \mathbb{E}_\lambda \left[\eta_\ell(\beta, x_j) \right]$.

Set intermediate global parameter

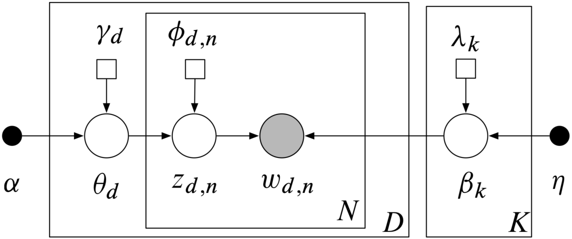
$$\hat{\lambda} = \alpha + n\mathbb{E}_\phi[t(Z_j, x_j)].$$

Set global parameter

$$\lambda = (1 - \rho_t)\lambda + \rho_t \hat{\lambda}.$$

until *forever*

Stochastic Variational Inference in LDA



Classic Coordinate Ascent

$$\phi_{d,n,k} \propto \exp\left(\mathbb{E}(\log \theta_{d,k}) + \mathbb{E}(\log \beta_{k,w_{d,n}})\right)$$

$$\gamma_d = \alpha + \sum_{n=1}^{N} \phi_{d,n}, \quad \lambda_k = \eta + \sum_{d=1}^{D} \sum_{n=1}^{N} \phi_{d,n,k} w_{d,n}$$

Stochastic Variational Inference in LDA

- Sample a document w_d uniform from the data set
- Estimate the local variational parameters using the current topics. For $n = 1, \ldots, N$

$$\phi_{d,n,k} \propto \exp\left(\mathbb{E}(\log \theta_{d,k}) + \mathbb{E}(\log \beta_{k,w_{d,n}})\right), \quad k = 1, \ldots, K$$

$$\gamma_d = \alpha + \sum_{n=1}^{N} \phi_{d,n}$$

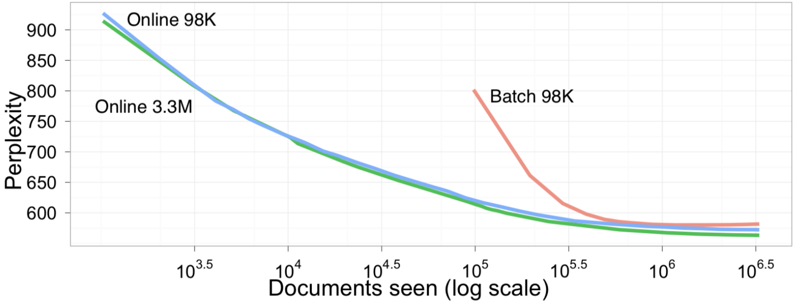
- Form the intermediate topics from those local parameters for noisy natural gradient

$$\hat{\lambda}_k = \eta + D \sum_{n=1}^{N} \phi_{d,n,k} w_{d,n}, \quad k = 1, \ldots, K$$

- Update topics using noisy natural gradient

$$\lambda = (1 - \rho_t)\lambda + \rho_t \hat{\lambda}$$

Stochastic Variational Inference in LDA



Documents analyzed	**2048**	**4096**	**8192**	**12288**	**16384**	**32768**	**49152**	**65536**
Top eight words	systems road made service announced national west language	systems health communication service billion language care road	service systems health companies market communication company billion	service systems companies business company billion health industry	service companies systems business company industry market billion	business service companies industry company management systems services	business service companies industry services company management public	business industry service companies services company management public

- Mean-field VI works for **conjugate-exponential models**, where the local optimal has closed-form solution.
- For more general models, we may not have this conditional conjugacy
  - Nonlinear Time Series Models
  - Deep Latent Gaussian Models
  - Generalized Linear Models
  - Stochastic Volatility Models
  - Bayesian Neural Networks
  - Sigmoid Belief Network
- While we may derive a model specific bound for each of these models (Knowles and Minka, 2011; Paisley et al., 2012), it would be better if there is a solution that does not entail model specific work.

VI for Bayesian Logistic Regression

- The logistic regression model

$$y_i \sim \text{Bernoulli}(p_i),\ p_i = \frac{1}{1+\exp(-x_i^\top \beta)}. \quad \beta \sim \mathcal{N}(0, I_d)$$

- The mean-field approximation

$$q(\beta) = \prod_{j=1}^{d} \mathcal{N}(\beta_j | \mu_j, \sigma_j^2)$$

- The ELBO is

$$L(\mu, \sigma^2) = \mathbb{E}_q(\log p(\beta) + \log p(y|x, \beta) - \log q(\beta))$$

VI for Bayesian Logistic Regression

$$
\begin{aligned}
L(\mu, \sigma^2) &= \mathbb{E}_q(\log p(\beta) - \log q(\beta) + \log p(y|x, \beta)) \\
&= -\frac{1}{2}\sum_{j=1}^{d}(\mu_j^2 + \sigma_j^2) + \frac{1}{2}\sum_{j=1}^{d}\log \sigma_j^2 + \mathbb{E}_q \log p(y|x, \beta) + \text{Const} \\
&= \frac{1}{2}\sum_{j=1}^{d}(\log \sigma_j^2 - \mu_j^2 - \sigma_j^2) + Y^\top X\mu - \mathbb{E}_q(\log(1 + \exp(X\beta)))
\end{aligned}
$$

- We can not compute the expectation term
- This hides the objective dependence on the variational parameters, making it hard to directly optimize.

- Let $p(x, \theta)$ be the joint probability (i.e., the posterior up to a constant), and $q_\phi(\theta)$ be our variational approximation
- The ELBO is

$$L(\phi) = \mathbb{E}_q(\log p(x, \theta) - \log q_\phi(\theta))$$

- Instead of requiring a closed-form lower bound and differentiating afterwards, we can take derivatives directly
- As shown later, this leads to a stochastic optimization approach that handles massive data sets as well.

- Compute the gradient

$$
\begin{aligned}
\nabla_\phi L &= \nabla_\phi \mathbb{E}_q(\log p(x,\theta) - \log q_\phi(\theta)) \\
&= \int \nabla_\phi q_\phi(\theta)(\log p(x,\theta) - \log q_\phi(\theta))\, d\theta \\
&\qquad\qquad - q_\phi(\theta)\nabla_\phi \log q_\phi(\theta)\, d\theta \\
&= \int q_\phi(\theta)\nabla_\phi \log q_\phi(\theta)(\log p(x,\theta) - \log q_\phi(\theta)) \\
&\qquad\qquad - q_\phi(\theta)\nabla_\phi \log q_\phi(\theta)\, d\theta \\
&= \mathbb{E}_q\left(\nabla_\phi \log q_\phi(\theta)(\log p(x,\theta) - \log q_\phi(\theta) - 1)\right)
\end{aligned}
$$

Using $\nabla_\phi \log q_\phi \theta = \frac{\nabla_\phi q_\phi(\theta)}{q_\phi(\theta)}$

Score Function Estimator

- Recall that

$$\nabla_\phi L = \mathbb{E}_q \left(\nabla_\phi \log q_\phi(\theta)(\log p(x,\theta) - \log q_\phi(\theta) - 1)\right)$$

- Note that

$$\mathbb{E}_q \nabla_\phi \log q_\phi(\theta) = 0$$

- We can simplify the gradient as follows

$$\nabla_\phi L = \mathbb{E}_q \left(\nabla_\phi \log q_\phi(\theta)(\log p(x,\theta) - \log q_\phi(\theta))\right)$$

- This is known as score function estimator or REINFORCE gradients (Williams, 1992; Ranganath et al., 2014; Minh et al., 2014)

Monte Carlo Estimate

$$\nabla_\phi L = \mathbb{E}_q \left(\nabla_\phi \log q_\phi(\theta)(\log p(x,\theta) - \log q_\phi(\theta)) \right)$$

- Unbiased stochastic gradients via **Monte Carlo**!

$$\frac{1}{S}\sum_{s=1}^{S} \nabla_\phi \log q_\phi(\theta_s)(\log p(x,\theta_s) - \log q_\phi(\theta_s)), \quad \theta_s \sim q_\phi(\theta)$$

- The requirements for inference
  - Sampling from $q_\phi(\theta)$
  - Evaluating $\nabla_\phi \log q_\phi(\theta)$
  - Evaluating $\log p(x,\theta)$ and $\log q_\phi(\theta)$
- This is called **Black Box Variational Inference** (BBVI): no model specific work! (Ranganath et al., 2014)

Algorithm 1: Basic Black Box Variational Inference

Input : Model $\log p(\mathbf{x}, \mathbf{z})$,
Variational approximation $q(\mathbf{z}; \boldsymbol{\nu})$

Output : Variational Parameters: $\boldsymbol{\nu}$

while *not converged* **do**

$\mathbf{z}[s] \sim q$ // **Draw** S **samples from** q

$\rho = t$-th value of a Robbins Monro sequence

$\boldsymbol{\nu} = \boldsymbol{\nu} + \rho \frac{1}{S} \sum_{s=1}^{S} \nabla_{\boldsymbol{\nu}} \log q(\mathbf{z}[s]; \boldsymbol{\nu})(\log p(\mathbf{x}, \mathbf{z}[s]) - \log q(\mathbf{z}[s]; \boldsymbol{\nu}))$

$t = t + 1$

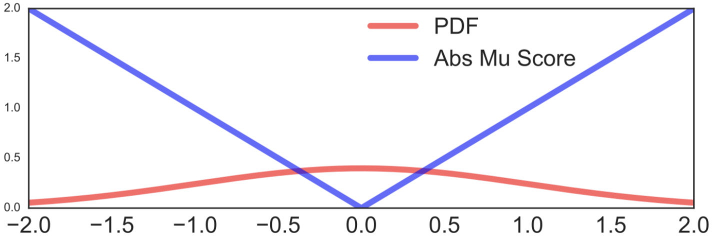
end

Ranganath et al., 2014

Basic BBVI Doesn't Work

Variance of the gradient can be a problem

$$\mathrm{Var}_{q_\phi(\theta)} = \mathbb{E}_q\left((\nabla_\phi \log q_\phi(\theta)(\log p(x,\theta) - \log q_\phi(\theta)) - \nabla_\phi L)^2\right)$$

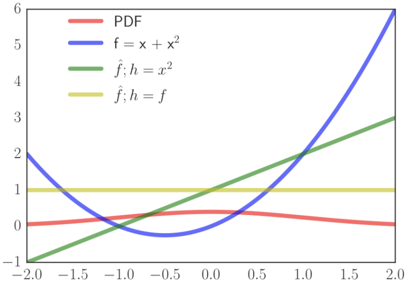


Adapted from Blei, Ranganath and Mohamed

- magnitude of $\log p(x,\theta) - \log q_\phi(\theta)$ varies widely
- rare values sampling
- too much variance to be useful

- To make BBVI work in practice, we need methods to reduce the variance of naive Monte Carlo estimates
- **Control Variates**. To reduce the variance of Monte Carlo estimates of $\mathbb{E}(f(x))$, we replace f with $\hat{f}$ such that $\mathbb{E}(\hat{f}(x)) = \mathbb{E}(f(x))$. A general class

$$\hat{f}(x) = f(x) - a(h(x) - \mathbb{E}h(x))$$



- a can be chosen to minimize the variance.
- h is a function of our choice. Good h have high correlation with the original function f.

Control Variates for VI

$$\hat{f}(x) = f(x) - a(h(x) - \mathbb{E}h(x))$$

- For variational inference, we need h functions with known q expectation
- A commonly used one is $h(\theta) = \nabla_\phi \log q_\phi(\theta)$, where

$$\mathbb{E}_q(\nabla_\phi \log q_\phi(\theta)) = 0, \quad \forall q$$

- The variance of $\hat{f}$ is

$$\text{Var}(\hat{f}) = \text{Var}(f) + a^2\text{Var}(h) - 2a\text{Cov}(f, h)$$

and the optimal scaling is $a^* = \text{Cov}(f, h)/\text{Var}(h)$. In practice this can be estimated using the empirical variance and covariance on the samples

Baseline

- When $h(\theta) = \nabla_\phi \log q_\phi(\theta)$, the control variate gradient is

$$\nabla_\phi L = \mathbb{E}_q \left(\nabla_\phi \log q_\phi(\theta)(\log p(x,\theta) - \log q_\phi(\theta) - a)\right)$$

  and a is called a **baseline**.
- Baselines can be constant, or input-dependent $a(x)$.
- While we can estimate the baseline using the samples as before, people often use a *model-agnostic* baseline to *centre the learning signal* (Minh and Gregor, 2014)

$$\rho = \arg\min_\rho \mathbb{E}_q(\ell(x,\theta,\phi) - a_\rho(x))^2$$

  where the learning signal is

$$\ell(x,\theta,\phi) = \log p(x,\theta) - \log q_\phi(\theta)$$

Rao-Blackwellization

- We can use Rao-Blackwellization to reduce the variance by integrating out some random variables.
- Consider the mean-field variational family

$$q(\theta) = \prod_{i=1}^{d} q_i(\theta_i|\phi_i)$$

- Let $q_{(i)}$ be the distribution of variables that depend on the ith variable (i.e., the Markov blanket of θ_i and θ_i), and let $p_i(x, \theta_{(i)})$ be the terms in the joint probability that depend on those variables.

$$\nabla_{\phi_i} L = \mathbb{E}_{q_{(i)}} \left(\nabla_{\phi_i} \log q_i(\theta_i|\phi_i)(\log p_i(x, \theta_{(i)}) - \log q_i(\theta_i|\phi_i)) \right)$$

- This can be combined with control variates.

The Reparameterization Trick

- Another commonly used variance reduction technique is **the reparameterization trick** (Kingma et al., 2014; Rezende et al., 2014)
- The Reparameterization

$$\theta = g_\phi(\epsilon), \ \ \epsilon \sim q_\epsilon(\epsilon) \quad \Longrightarrow \quad \theta \sim q_\phi(\theta)$$

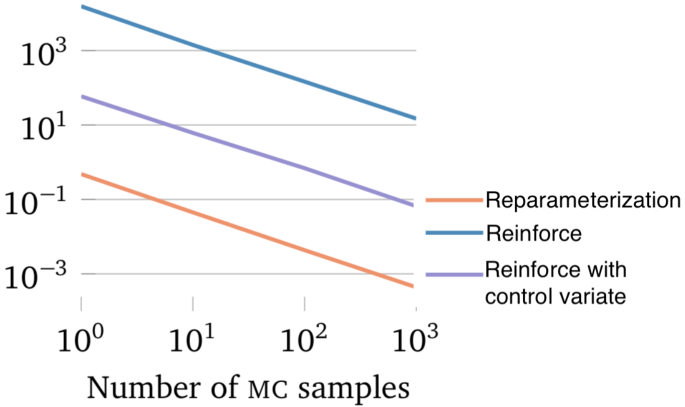
- Example:

$$\theta = \epsilon\sigma + \mu, \ \ \epsilon \sim \mathcal{N}(0,1) \quad \Longleftrightarrow \quad \theta \sim \mathcal{N}(\mu, \sigma^2)$$

- Compute the gradient via the reparameterization trick

$$\begin{aligned} \nabla_\phi L &= \nabla_\phi \mathbb{E}_{q_\phi(\theta)} (\log p(x,\theta) - \log q_\phi(\theta)) \\ &= \nabla_\phi \mathbb{E}_{q_\epsilon(\epsilon)} (\log p(x, g_\phi(\epsilon)) - \log q_\phi(g_\phi(\epsilon))) \\ &= \mathbb{E}_{q_\epsilon(\epsilon)} \nabla_\phi (\log p(x, g_\phi(\epsilon)) - \log q_\phi(g_\phi(\epsilon))) \end{aligned}$$

Variance Comparison



Kucukelbir et al., 2016

Score Function

- Differentiates the density $\nabla_\phi q_\phi(\theta)$
- Works for general models, including both discrete and continuous models.
- Works for large class of variational approximations
- May suffer from large variance

Reparameterization

- Differentiates the function $\nabla_\phi(\log p(x,\theta) - \log q_\phi(\theta))$
- Requires differentiable models
- Requires variational approximation to have form $\theta = g_\phi(\epsilon)$
- Better behaved variance in general

- Scale up previous stochastic variational inference methods to large data set via **data subsampling**.
- Replace the log joint distribution with unbiased stochastic estimates

$$\log p(x, \theta) \simeq \log p(\theta) + \frac{n}{m} \sum_{i=1}^{m} \log p(x_{t_i} | \theta), \quad m \ll n$$

- Example: score function estimator

$$\hat{\nabla}_\phi L = \frac{1}{S} \sum_{s=1}^{S} \nabla_\phi \log q_\phi(\theta_s) \left(\log p(\theta_s) + \frac{n}{m} \sum_{i=1}^{m} \log p(x_{t_i} | \theta_s) - \log q_\phi(\theta_s) \right), \quad \theta_s \sim q_\phi(\theta)$$

Summary

- When the data size is large, we can use **stochastic optimization** to scale up VI.
- For conditional exponential models, we can use noisy natural gradient.
- For general models, naive stochastic gradient estimators may have large variance, variance reduction techniques are often required.
  - Score function estimator (for both discrete and continuous latent variable)
  - The reparameterization trick (for continuous variable, and requires reparameterizable variational family)
- We can also combine score function estimators with the reparameterization trick for more general and robust stochastic gradient estimators (Ruiz et al., 2016)

References

- S. Amari. Natural gradient works efficiently in learning. Neural computation, 10(2):251–276, 1998.
- Hoffman, M., Blei, D., Wang, C., and Paisley, J. (2013). Stochastic variational inference. Journal of Machine Learning Research, 14:1303–1347.
- D. Knowles and T. Minka. Non-conjugate variational message passing for multinomial and binary regression. In Advances in Neural Information Processing Systems, 2011.
- J. Paisley, D. Blei, and M. Jordan. Variational Bayesian inference with stochastic search. International Conference in Machine Learning, 2012.

References

- Williams, R. J. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. In Machine Learning, pages 229–256.
- R. Ranganath, S. Gerrish, and D. Blei. Black box variational inference. In Artificial Intelligence and Statistics, 2014.
- Rezende, D. J., Mohamed, S., and Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. In International Conference on Machine Learning, pages 1278–1286.
- D. P. Kingma and M. Welling. Auto-encoding variational Bayes. In International Conference on Learning Representations, 2014.

References

- A. Mnih and K. Gregor. Neural variational inference and learning in belief networks. Advances in Neural Information Processing Systems, 2014.
- F. R. Ruiz, M. Titsias, and D. Blei. The generalized reparameterization gradient. Advances in Neural Information Processing Systems, 2016.