DSC291: Machine Learning with Few Labels

Variational Inference

Zhiting Hu

Lecture 7, January 24, 2023

UC San Diego
HALICIOĞLU DATA SCIENCE INSTITUTE

Recap: EM Algorithm

- The EM algorithm is coordinate-decent on $F(q, \theta)$
  - E-step: $q^{t+1} = \arg\min_q F\left(q, \theta^t\right) = p(\boldsymbol{z}|\boldsymbol{x}, \theta^t)$
  - M-step: $\theta^{t+1} = \arg\min_\theta F\left(q^{t+1}, \theta^t\right) = \text{argmax}_\theta \sum_z q^{t+1}(\boldsymbol{z}|\boldsymbol{x}) \log p(\boldsymbol{x}, \boldsymbol{z}|\theta)$

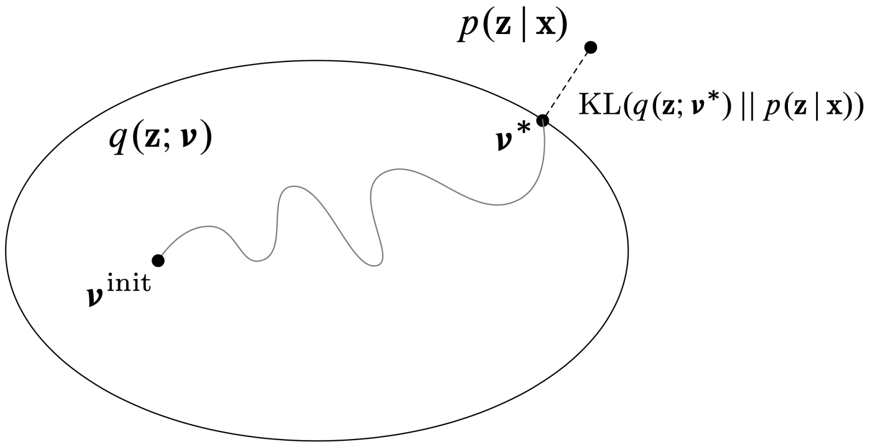
$$\ell(\theta; \boldsymbol{x}) = \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p(\boldsymbol{x}, \boldsymbol{z}|\theta)}{q(\boldsymbol{z}|\boldsymbol{x})}\right] + \text{KL}\big(q(\boldsymbol{z}|\boldsymbol{x}) \,||\, p(\boldsymbol{z}|\boldsymbol{x}, \theta)\big)$$

$$= -F(q, \theta) + \text{KL}\big(q(\boldsymbol{z}|\boldsymbol{x}) \,||\, p(\boldsymbol{z}|\boldsymbol{x}, \theta)\big)$$

- Limitation: need to be able to compute $p(\boldsymbol{z}|\boldsymbol{x}, \theta)$, not possible for more complicated models --- solution: Variational inference

Recap: Variational Inference

- We often cannot compute posteriors $p(\boldsymbol{z}|\boldsymbol{x},\theta)$, and so we need to approximate them, using variational methods.
- In variational Bayes, we'd like to find an approximation within some family that minimizes the KL divergence to the posterior, but we can't directly minimize this
- Therefore, we defined the ELBO, which we can maximize, and this is equivalent to minimizing the KL divergence.

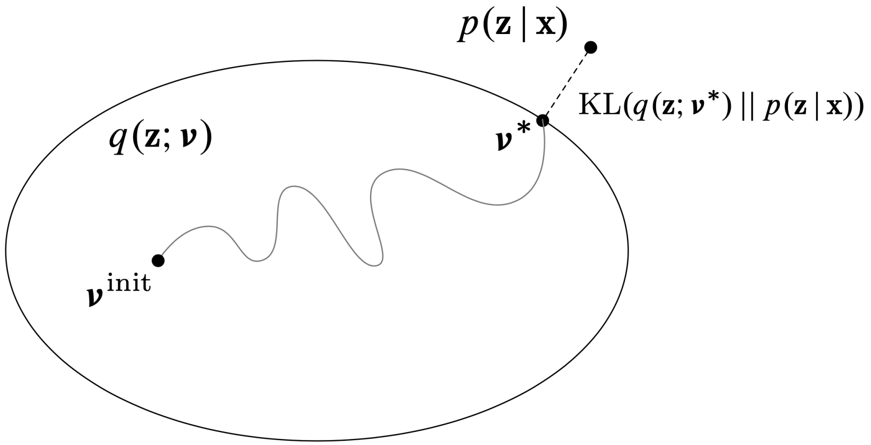


Evidence Lower Bound (ELBO)

$$\ell(\theta;\boldsymbol{x}) = \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x})}\left[\log\frac{p(\boldsymbol{x},\boldsymbol{z}|\theta)}{q(\boldsymbol{z}|\boldsymbol{x})}\right] + \mathrm{KL}\big(q(\boldsymbol{z}|\boldsymbol{x})\,||\,p(\boldsymbol{z}|\boldsymbol{x},\theta)\big)$$

Recap: Variational Inference

- We often cannot compute posteriors $p(\boldsymbol{z}|\boldsymbol{x},\theta)$, and so we need to approximate them, using variational methods.
- In variational Bayes, we'd like to find an approximation within some family that minimizes the KL divergence to the posterior, but we can't directly minimize this
- Therefore, we defined the ELBO, which we can maximize, and this is equivalent to minimizing the KL divergence.



Evidence Lower Bound (ELBO)

$$\ell(\theta;\boldsymbol{x}) = \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x})}\left[\log\frac{p(\boldsymbol{x},\boldsymbol{z}|\theta)}{q(\boldsymbol{z}|\boldsymbol{x})}\right] + \mathrm{KL}\big(q(\boldsymbol{z}|\boldsymbol{x})\,||\,p(\boldsymbol{z}|\boldsymbol{x},\theta)\big)$$

- How do we choose the variational family $q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{v})$?

Mean Field Variational Inference

- A popular family of variational approximations

- In this type of variational inference, we assume the variational distribution over the latent variables factorizes as

$$q(\boldsymbol{z}) = q(z_1, \ldots, z_m) = \prod_{j=1}^{m} q(z_j)$$

  - (where we omit variational parameters for ease of notation)
  - We refer to $q(z_j)$, the variational approximation for a single latent variable, as a "local variational approximation"

- In the above expression, the variational approximation $q(z_j)$ over each latent variable z_j is independent

Mean Field Variational Inference

- Note that this is a fairly general setup; we can also partition the latent variables $z_1, \ldots, z_m$ into R groups $z_{G_1}, \ldots, z_{G_R}$, and use the approximation:

$$q(z_1, \ldots, z_m) = q(z_{G_1}, \ldots, z_{G_R}) = \prod_{r=1}^{R} q(z_{G_r})$$

  - Often called “generalized mean field” versus (the above) “naïve mean field”.

Mean Field Variational Inference

- Note that this is a fairly general setup; we can also partition the latent variables $z_1, \ldots, z_m$ into R groups $z_{G_1}, \ldots, z_{G_R}$, and use the approximation:

$$q(z_1, \ldots, z_m) = q(z_{G_1}, \ldots, z_{G_R}) = \prod_{r=1}^{R} q(z_{G_r})$$

  - Often called "generalized mean field" versus (the above) "naïve mean field".

- Typically, this approximation does not contain the true posterior (because the latent variables are dependent).
  - E.g.: in the (Bayesian) mixture of Gaussians model, all of the cluster assignments z_i for $i = 1, \ldots, n$ are dependent on each other and on the cluster locations $\mu_{1:K}$ given data.

Optimizing the ELBO in Mean Field Variational Inference

How do we optimize the ELBO in mean field variational inference?

- Typically, we use coordinate ascent optimization.
- I.e. we optimize each latent variable's variational approximation $q(z_j)$ in turn while holding the others fixed.
  - At each iteration we get an updated "local" variational approximation.
  - And we iterate through each latent variable until convergence.

Optimizing the ELBO in Mean Field Variational Inference

- Recall that the ELBO is defined as:

$$\mathcal{L} = \mathbb{E}_q[\log p(\boldsymbol{x}, \boldsymbol{z})] - \mathbb{E}_q[\log q(\boldsymbol{z})]$$

- Note that we can decompose the entropy term of the ELBO (using the mean field variational approximation) as:

$$\mathbb{E}_q[\log q(z_{1:m})] = \sum_{j=1}^{m} \mathbb{E}_{q_j}[\log q(z_j)]$$

- Therefore, under the mean field approximation, the ELBO can be written:

$$\mathcal{L} = \mathbb{E}_{q_j}\mathbb{E}_{q_{-j}}[\log p(\boldsymbol{x}, \boldsymbol{z})] - \sum_{j=1}^{m} \mathbb{E}_{q_j}[\log q(z_j)]$$

Optimizing the ELBO in Mean Field Variational Inference

- Therefore, under the mean field approximation, the ELBO can be written:

$$\mathcal{L} = \mathbb{E}_{q_j}\mathbb{E}_{q_{-j}}[\log p(\boldsymbol{x}, \boldsymbol{z})] - \sum_{j=1}^{m} \mathbb{E}_{q_j}[\log q(z_j)]$$

- Next, we want to derive the coordinate ascent update for a latent variable z_j , keeping all other latent variables fixed.
  - i.e. we want the $\mathrm{argmax}_{q_j}\mathcal{L}$.

- Removing the parts that do not depend on $q(z_j)$, we can write:

$$\mathcal{L} = \mathbb{E}_{q_j}\mathbb{E}_{q_{-j}}[\log p(\boldsymbol{x}, \boldsymbol{z})] - \mathbb{E}_{q_j}[\log q(z_j)] + \mathrm{const.}$$

Optimizing the ELBO in Mean Field Variational Inference

- To find this argmax, we take the derivative of $\mathcal{L}$ w.r.t $q(z_j)$ and and set the derivative to zero :

$$\frac{d\mathcal{L}}{dq(z_j)} = \mathbb{E}_{q_j}\mathbb{E}_{q_{-j}}[\log p(\boldsymbol{x}, \boldsymbol{z})] - \log q(z_j) - 1 = 0$$

- From this, we arrive at the coordinate ascent update:

$$q^*(z_j) \propto \exp\left\{ \mathbb{E}_{q_{-j}}[\log p(\boldsymbol{x}, \boldsymbol{z})] \right\}$$

Optimizing the ELBO in Mean Field Variational Inference

- The coordinate ascent update:

$$q^*(z_j) \propto \exp\left\{ \mathbb{E}_{q_{-j}}[\log p(\boldsymbol{x}, \boldsymbol{z})] \right\}$$

  - The optimal solution for factor $q(z_j)$ is obtained simply by considering the log of the joint distribution over all observed and latent variables and then taking the expectation with respect to all of the other factors $q(z_k)$, $k \neq j$, then taking exponential and normalizing

- Note that the only assumption we made so far is the mean-field factorization:

$$q(\boldsymbol{z}) = q(z_1, \ldots, z_m) = \prod_{j=1}^{m} q(z_j)$$

  - We haven't yet made any assumptions on the form of $q(z_j)$

Simple example:

- Consider a univariate Gaussian distribution $p(x) = \mathcal{N}(x|\mu, \tau^{-2})$, given a dataset $\mathcal{D} = \{x_1, \ldots, x_N\}$:

$$p(\mathcal{D}|\mu,\tau) = \left(\frac{\tau}{2\pi}\right)^{N/2} \exp\left\{-\frac{\tau}{2}\sum_{n=1}^{N}(x_n-\mu)^2\right\}$$

$$p(\mu|\tau) = \mathcal{N}\left(\mu|\mu_0, (\lambda_0\tau)^{-1}\right)$$

$$p(\tau) = \mathrm{Gam}(\tau|a_0, b_0)$$

  - $\mathrm{Gam}(\tau|a_0, b_0) = \frac{1}{\Gamma(\mathrm{a})} b^a \lambda^{a-1} \exp(-b\lambda)$: gamma distribution

- For this simple problem the posterior distribution can be found exactly. But we use it as an example for tutorial anyway

$$q^*(z_j) \propto \exp\left\{ \mathbb{E}_{q_{-j}}[\log p(\boldsymbol{x}, \boldsymbol{z})] \right\}$$

Simple example:

$$p(\mathcal{D}|\mu,\tau) = \left(\frac{\tau}{2\pi}\right)^{N/2} \exp\left\{-\frac{\tau}{2}\sum_{n=1}^{N}(x_n-\mu)^2\right\}$$

$$\begin{aligned} p(\mu|\tau) &= \mathcal{N}\left(\mu|\mu_0, (\lambda_0\tau)^{-1}\right) \\ p(\tau) &= \mathrm{Gam}(\tau|a_0, b_0) \end{aligned}$$

- Introduce the factorized variational approximation: $q(\mu,\tau) = q_\mu(\mu)q_\tau(\tau)$
- Solution to q_μ:

$$\begin{aligned} \ln q_\mu^\star(\mu) &= \mathbb{E}_\tau\left[\ln p(\mathcal{D}|\mu,\tau) + \ln p(\mu|\tau)\right] + \text{const} \\ &= -\frac{\mathbb{E}[\tau]}{2}\left\{\lambda_0(\mu-\mu_0)^2 + \sum_{n=1}^{N}(x_n-\mu)^2\right\} + \text{const.} \end{aligned}$$

  - We can see q_μ^* is a Gaussian $\mathcal{N}(x|\mu_N, \lambda_N^{-1})$:

$$\begin{aligned} \mu_N &= \frac{\lambda_0\mu_0 + N\overline{x}}{\lambda_0 + N} \\ \lambda_N &= (\lambda_0 + N)\mathbb{E}[\tau] \end{aligned}$$

$$q^*(z_j) \propto \exp\left\{ \mathbb{E}_{q_{-j}}[\log p(\boldsymbol{x}, \boldsymbol{z})] \right\}$$

Simple example:

$$p(\mathcal{D}|\mu,\tau) = \left(\frac{\tau}{2\pi}\right)^{N/2} \exp\left\{-\frac{\tau}{2}\sum_{n=1}^{N}(x_n-\mu)^2\right\}$$

$$\begin{aligned} p(\mu|\tau) &= \mathcal{N}\left(\mu|\mu_0,(\lambda_0\tau)^{-1}\right) \\ p(\tau) &= \mathrm{Gam}(\tau|a_0,b_0) \end{aligned}$$

- Introduce the factorized variational approximation: $q(\mu,\tau) = q_\mu(\mu)q_\tau(\tau)$
- Solution to q_τ:

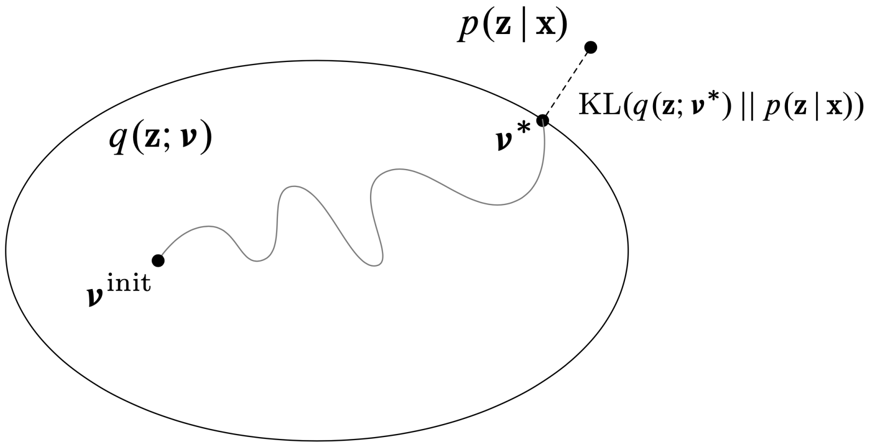
$$\begin{aligned} \ln q_\tau^\star(\tau) &= \mathbb{E}_\mu\left[\ln p(\mathcal{D}|\mu,\tau) + \ln p(\mu|\tau)\right] + \ln p(\tau) + \text{const} \\ &= (a_0-1)\ln\tau - b_0\tau + \frac{N}{2}\ln\tau \\ &\quad -\frac{\tau}{2}\mathbb{E}_\mu\left[\sum_{n=1}^{N}(x_n-\mu)^2 + \lambda_0(\mu-\mu_0)^2\right] + \text{const} \end{aligned}$$

  - We can see q_τ^* is a gamma distribution $\mathrm{Gam}(\tau|a_N,b_N)$:

$$\begin{aligned} a_N &= a_0 + \frac{N}{2} \\ b_N &= b_0 + \frac{1}{2}\mathbb{E}_\mu\left[\sum_{n=1}^{N}(x_n-\mu)^2 + \lambda_0(\mu-\mu_0)^2\right] \end{aligned}$$

Quick Recap

- We often cannot compute posteriors, and so we need to approximate them, using variational methods.
- In variational Bayes, we'd like to find an approximation within some family that minimizes the KL divergence to the posterior, but we can't directly minimize this
- Therefore, we defined the ELBO, which we can maximize, and this is equivalent to minimizing the KL divergence.



Evidence Lower Bound (ELBO)

$$\ell(\theta; \boldsymbol{x}) = \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p(\boldsymbol{x}, \boldsymbol{z}|\theta)}{q(\boldsymbol{z}|\boldsymbol{x})}\right] + \mathrm{KL}\big(q(\boldsymbol{z}|\boldsymbol{x}) \mid\mid p(\boldsymbol{z}|\boldsymbol{x}, \theta)\big)$$

Quick Recap

- We defined a family of approximations called "mean field" approximations, in which there are no dependencies between latent variables

$$q(\boldsymbol{z}) = q(z_1, \ldots, z_m) = \prod_{j=1}^{m} q(z_j)$$

- We optimize the ELBO with coordinate ascent updates to iteratively optimize each local variational approximation under mean field assumptions

$$q^*(z_j) \propto \exp\left\{ \mathbb{E}_{q_{-j}}[\log p(\boldsymbol{x}, \boldsymbol{z})] \right\}$$

Key Takeaways

- KL Divergence

$$\mathrm{KL}(q(\boldsymbol{x}) \,||\, p(\boldsymbol{x})) = \sum_{\boldsymbol{x}} q(\boldsymbol{x}) \log \frac{q(\boldsymbol{x})}{p(\boldsymbol{x})}$$

- The key equation of EM and VI

Evidence Lower Bound (ELBO)

$$\ell(\theta; \boldsymbol{x}) = \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p(\boldsymbol{x}, \boldsymbol{z}|\theta)}{q(\boldsymbol{z}|\boldsymbol{x})}\right] + \mathrm{KL}\big(q(\boldsymbol{z}|\boldsymbol{x}) \,||\, p(\boldsymbol{z}|\boldsymbol{x}, \theta)\big)$$

  - Free energy $F(q, \theta)$

- EM: E-step and M-step optimizing ELBO w.r.t q and θ
- Mean-field VI: optimizing ELBO w.r.t factorized q components

Stochastic VI; Black-box VI

VI with coordinate ascent

Example: Bayesian mixture of Gaussians

- Treat the mean μ_k and cluster proportion π as latent variables

$$\mu_k \sim \mathcal{N}(0, \tau^2) \text{ for } k = 1, \ldots, K$$

$$\pi \sim Dirichlet(\boldsymbol{\alpha})$$

- For each data $i = 1, \ldots, n$

$$z_i \sim \text{Cat}(\pi).$$

$$x_i \sim \mathcal{N}(\mu_{z_i}, \sigma^2).$$

- We have
  - observed variables $x_{1:n}$
  - latent variables $\mu_{1:k}$, π and $z_{1:n}$
  - Hyper-parameters $\{\tau^2, \sigma^2\}$

VI with coordinate ascent

Example: Bayesian mixture of Gaussians

Assume mean-field $q(\mu_{1:K}, \pi, z_{1:n}) = \prod_k q(\mu_k) q(\pi) \prod_i q(z_i)$

- Initialize the global variational distributions $q(\mu_k)$ and $q(\pi)$
- Repeat:
  - For each data example $i \in \{1,2, \dots, D\}$
    - Update the local variational distribution $q(z_i)$
  - End for
  - Update the global variational distributions $q(\mu_k)$ and $q(\pi)$
- Until ELBO converges

- What if we have millions of data examples? This could be very slow.

Stochastic VI

Example: Bayesian mixture of Gaussians

Assume mean-field $q(\mu_{1:K}, \pi, z_{1:n}) = \prod_k q(\mu_k) q(\pi) \prod_i q(z_i)$

- Initialize the global variational distributions $q(\mu_k)$ and $q(\pi)$
- Repeat:
  - Sample a data example $i \in \{1,2,\dots,D\}$
  - Update the local variational distribution $q(z_i)$
  - Update the global variational distributions $q(\mu_k)$ and $q(\pi)$ with **natural gradient ascent**
- Until ELBO converges

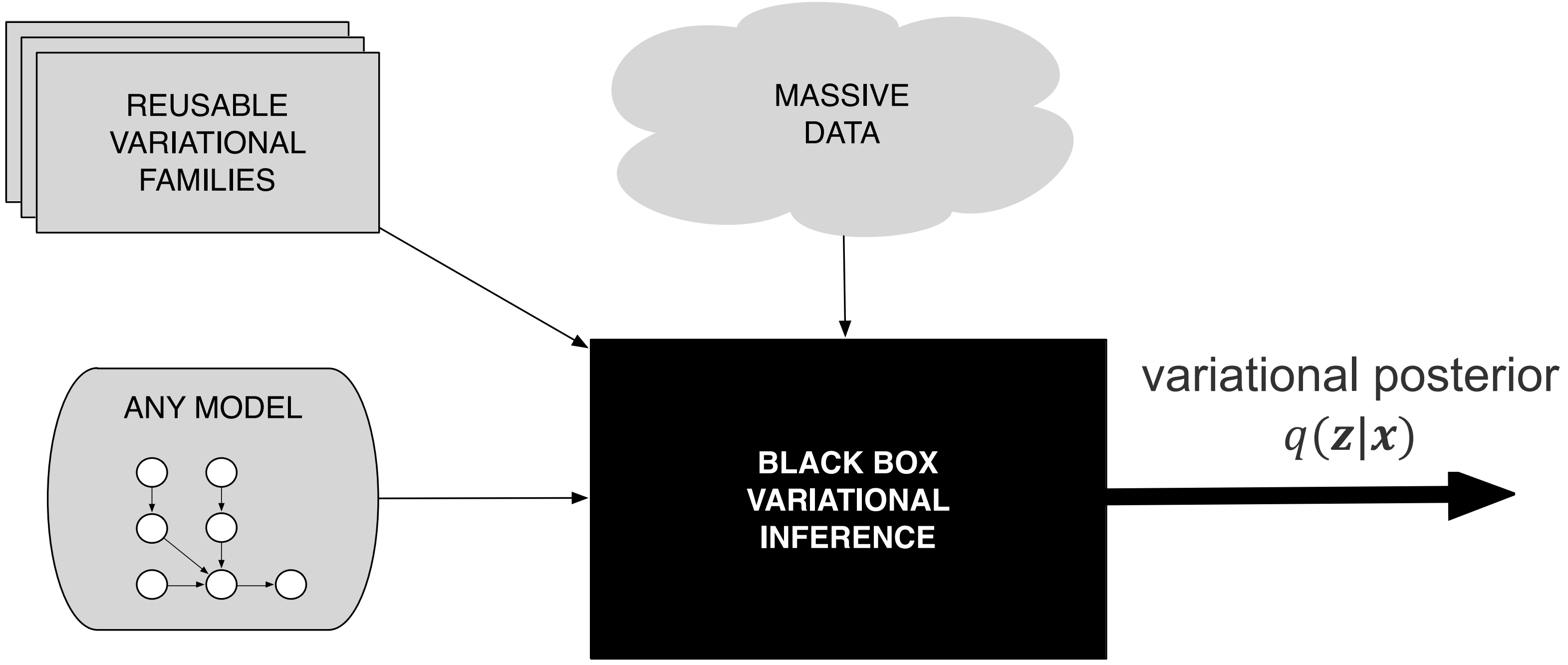
- (Setting natural gradient $= 0$ gives the traditional mean-field update)

[Hoffman et al., Stochastic Variational Inference, 2013]

Black-box Variational Inference (BBVI)

- We have derived variational inference specific for Bayesian Gaussian (mixture) models
- There are innumerable models
- Can we have a solution that does not entail model-specific work?

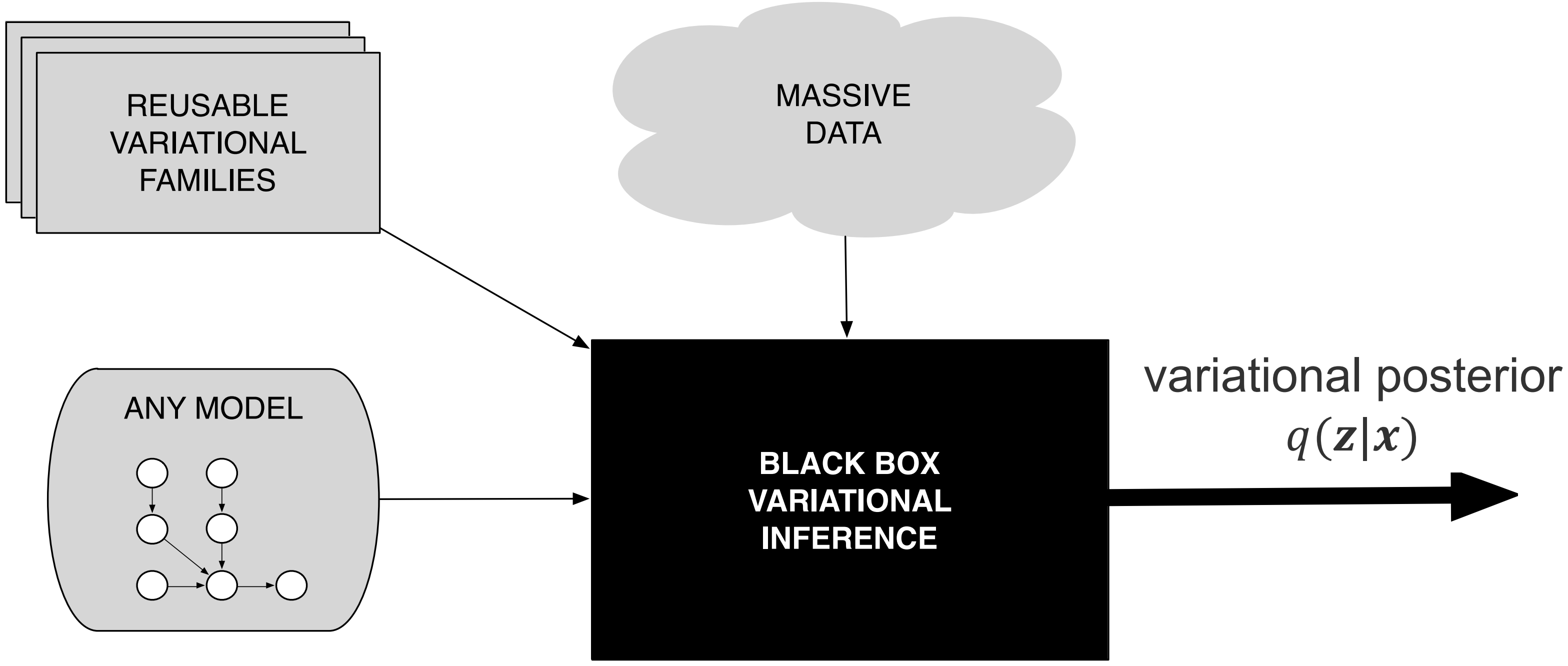
Black-box Variational Inference (BBVI)



- Easily use variational inference with **any model**
- Perform inference with **massive data**
- **No mathematical work** beyond specifying the model

(Courtesy: Blei et al., 2018)

Black-box Variational Inference (BBVI)



- Sample from $q(.)$
- Form noisy gradients (without model-specific computation)
- Use stochastic optimization

(Courtesy: Blei et al., 2018)

Black-box Variational Inference (BBVI)

- Probabilistic model: $\boldsymbol{x}$ -- observed variables, $\boldsymbol{z}$ -- latent variables
- Variational distribution $q_\lambda(\boldsymbol{z}|\boldsymbol{x})$ with parameters λ, e.g.,
  - Gaussian mixture distribution:
    - "A Gaussian mixture model is a universal approximator of densities, in the sense that any smooth density can be approximated with any specific nonzero amount of error by a Gaussian mixture model with enough components." (Deep Learning book, pp.65)
  - Deep neural networks

- ELBO:

$$\mathcal{L}(\lambda) = \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{\lambda})}[\log p(\boldsymbol{x}, \boldsymbol{z})] - \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{\lambda})}[\log q(\boldsymbol{z}|\lambda)]$$

- Want to compute the gradient w.r.t variational parameters λ

[Ranganath et al.,14]

The General Problem: Computing Gradients of Expectations

- When the objective function $\mathcal{L}$ is defined as an expectation of a (differentiable) test function $f_\lambda(\boldsymbol{z})$ w.r.t. a probability distribution $q_\lambda(\boldsymbol{z})$

$$\mathcal{L} = \mathbb{E}_{q_{\boldsymbol{\lambda}}(\boldsymbol{z})}[f_\lambda(\boldsymbol{z})]$$

- Computing exact gradients w.r.t. the parameters λ is often unfeasible
- Need stochastic gradient estimates
    - The score function estimator (a.k.a log-derivative trick, REINFORCE)
    - The reparameterization trick (a.k.a the pathwise gradient estimator)

Computing Gradients of Expectations w/ score function

- Loss: $\mathcal{L} = \mathbb{E}_{q_{\boldsymbol{\lambda}}(\boldsymbol{z})}[f_{\lambda}(\boldsymbol{z})]$
- Log-derivative trick: $\nabla_{\lambda} q_{\lambda} = q_{\lambda}\, \nabla_{\lambda} \log q_{\lambda}$
- Gradient w.r.t. λ:

$$\nabla_{\lambda}\mathcal{L} = \mathbb{E}_{q_{\boldsymbol{\lambda}}(\boldsymbol{z})}[f_{\lambda}(\boldsymbol{z}) \underline{\nabla_{\lambda}\log q_{\lambda}(\boldsymbol{z})} + \nabla_{\lambda} f_{\lambda}(\boldsymbol{z})]$$

  - score function: the gradient of the log of a probability distribution
- Compute noisy unbiased gradients with Monte Carlo samples from q_{λ}

$$\nabla_{\lambda}\mathcal{L} \approx \frac{1}{S}\sum_{s=1}^{S} f_{\lambda}(\boldsymbol{z}_s)\nabla_{\lambda}\log q_{\lambda}(\boldsymbol{z}_s) + \nabla_{\lambda} f_{\lambda}(\boldsymbol{z}_s) \qquad \text{where } \boldsymbol{z}_s \sim q_{\lambda}(\boldsymbol{z})$$

- Pros: generally applicable to any distribution $q(z|\lambda)$
- Cons: empirically has high variance → slow convergence
  - To reduce variance: Rao-Blackwellization, control variates, importance sampling, ...

Computing Gradients of Expectations w/ reparametrization trick

- Loss: $\mathcal{L} = \mathbb{E}_{q_\lambda(\boldsymbol{z})}[f_\lambda(\boldsymbol{z})]$
- Assume that we can express the distribution $q_{\boldsymbol{\lambda}}(\boldsymbol{z})$ with a transformation

$$\begin{aligned} \epsilon &\sim s(\epsilon) \\ z &= t(\epsilon, \lambda) \end{aligned} \quad \Leftrightarrow \quad z \sim q(z|\lambda)$$

  - E.g.,

$$\begin{aligned} \epsilon &\sim Normal(0, 1) \\ z &= \epsilon\sigma + \mu \end{aligned} \quad \Leftrightarrow \quad z \sim Normal(\mu, \sigma^2)$$

- Reparameterization gradient

$$\mathcal{L} = \mathbb{E}_{\boldsymbol{\epsilon} \sim s(\boldsymbol{\epsilon})}[f_\lambda(\boldsymbol{z}(\boldsymbol{\epsilon}, \lambda))]$$

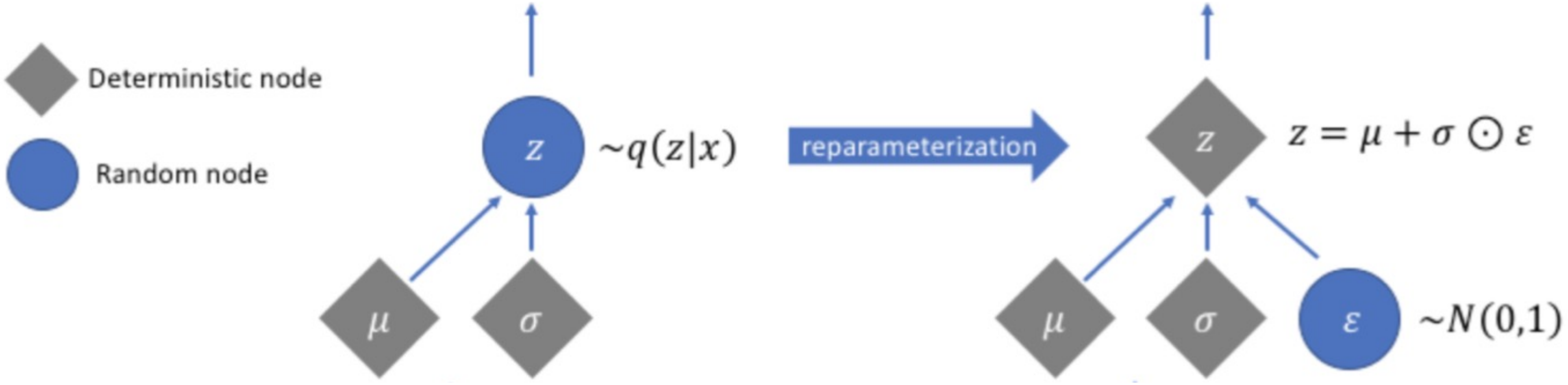
$$\nabla_\lambda \mathcal{L} = \mathbb{E}_{\boldsymbol{\epsilon} \sim \boldsymbol{s}(\boldsymbol{\epsilon})}[\nabla_{\boldsymbol{z}} f_\lambda(\boldsymbol{z})\, \nabla_\lambda t(\epsilon, \lambda)]$$

- Pros: empirically, lower variance of the gradient estimate
- Cons: Not all distributions can be reparameterized

Reparameterization trick

- Reparametrizing Gaussian distribution

$$\begin{aligned} \epsilon &\sim Normal(0,1) \\ z &= \epsilon\sigma + \mu \end{aligned} \iff z \sim Normal(\mu, \sigma^2)$$



[Courtesy: Tansey, 2016]

Reparameterization trick

- Reparametrizing Gaussian distribution

$$\begin{array}{l} \epsilon \sim Normal(0,1) \\ z = \epsilon\sigma + \mu \end{array} \Leftrightarrow \quad z \sim Normal(\mu, \sigma^2)$$

- Other reparameterizable distributions:

$$\begin{array}{l} \epsilon \sim Uniform(\epsilon) \\ z = F^{-1}(\epsilon) \end{array} \Leftrightarrow \quad z \sim q(z)$$

  - Tractable inverse CDF F^{-1}:
    - Exponential, Cauchy, Logistic, Rayleigh, Pareto, Weibull, Reciprocal, Gompertz, Gumbel, Erlang
  - Location-scale:
    - Laplace, Elliptical, Student's t, Logistic, Uniform, Triangular, Gaussian
  - Composition:
    - Log-Normal (exponentiated normal) Gamma (sum of exponentials) Dirichlet (sum of Gammas) Beta, Chi-Squared, F

[Courtesy: Tansey, 2016]

Computing Gradients of Expectations: Summary

- Loss: $\mathcal{L} = \mathbb{E}_{q_\lambda(\boldsymbol{z})}[f_\lambda(\boldsymbol{z})]$

- Score gradient

$$\nabla_\lambda \mathcal{L} = \mathbb{E}_{q_\lambda(\boldsymbol{z})}[f_\lambda(\boldsymbol{z}) \nabla_\lambda \log q_\lambda(\boldsymbol{z}) + \nabla_\lambda f_\lambda(\boldsymbol{z})]$$

  - Pros: generally applicable to any distribution $q(z|\lambda)$
  - Cons: empirically has high variance → slow convergence

- Reparameterization gradient

$$\nabla_\lambda \mathcal{L} = \mathbb{E}_{\boldsymbol{\epsilon} \sim \boldsymbol{s}(\boldsymbol{\epsilon})}[\nabla_{\boldsymbol{z}} f_\lambda(\boldsymbol{z})\, \nabla_\lambda t(\epsilon, \lambda)]$$

  - Pros: empirically, lower variance of the gradient estimate
  - Cons: Not all distributions can be reparameterized

Recall: Black-box Variational Inference (BBVI)

- Probabilistic model: $\boldsymbol{x}$ -- observed variables, $\boldsymbol{z}$ -- latent variables
- Variational distribution $q_\lambda(\boldsymbol{z}|\boldsymbol{x})$ with parameters λ, e.g.,
  - Gaussian mixture distribution:
    - "A Gaussian mixture model is a universal approximator of densities, in the sense that any smooth density can be approximated with any specific nonzero amount of error by a Gaussian mixture model with enough components." (Deep Learning book, pp.65)
  - Deep neural networks

$$\mathcal{L}(\lambda) \triangleq \mathrm{E}_{q_\lambda(z)}[\log p(x, z) - \log q(z)]$$

- ELBO:

$$\mathcal{L}(\lambda) = \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{\lambda})}[\log p(\boldsymbol{x}, \boldsymbol{z})] - \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{\lambda})}[\log q(\boldsymbol{z}|\lambda)]$$

- Want to compute the gradient w.r.t variational parameters λ

[Ranganath et al.,14]

BBVI with the score gradient

- ELBO:

$$\mathcal{L}(\lambda) = \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{\lambda})}[\log p(\boldsymbol{x}, \boldsymbol{z})] - \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{\lambda})}[\log q(\boldsymbol{z}|\lambda)]$$

- Gradient w.r.t. λ (using the log-derivative trick)

$$\nabla_\lambda \mathcal{L} = \mathrm{E}_q[\nabla_\lambda \log q(z|\lambda)(\log p(x, z) - \log q(z|\lambda))]$$

- Compute noisy unbiased gradients of the ELBO with Monte Carlo samples from the variational distribution

$$\nabla_\lambda \mathcal{L} \approx \frac{1}{S} \sum_{s=1}^{S} \nabla_\lambda \log q(z_s|\lambda)(\log p(x, z_s) - \log q(z_s|\lambda)),$$

$$\text{where } z_s \sim q(z|\lambda).$$

[Ranganath et al.,14]

BBVI with the reparameterization gradient

- ELBO:

$$\mathcal{L}(\lambda) = \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{\lambda})}[\log p(\boldsymbol{x}, \boldsymbol{z})] - \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{\lambda})}[\log q(\boldsymbol{z}|\lambda)]$$

- Gradient w.r.t. λ

$$\begin{aligned} \epsilon &\sim s(\epsilon) \\ z &= t(\epsilon, \lambda) \end{aligned} \quad \Leftrightarrow \quad z \sim q(z|\lambda)$$

$$\nabla_\lambda \mathcal{L} = \mathrm{E}_{\epsilon \sim s(\epsilon)}[\nabla_z[\log p(x, z) - \log q(z)] \nabla_\lambda t(\epsilon, \lambda)]$$

Questions?

Questions?