Bayesian statistics

Variational inference

Leonardo Egidi

2024/2025

Università di Trieste

Table of contents i

Variational inference: why? i

- One of the core problems of modern statistics is to approximate difficult-to-compute probability densities: this is relevant especially in Bayesian statistics, which frames all inference about unknown quantities as a calculation about the posterior.
- Variational inference (VI) is widely used to approximate posterior densities for Bayesian models, an alternative strategy to MCMC sampling.
- Compared to MCMC, variational inference tends to be faster and easier to scale to large data—it has been applied to problems such as large-scale document analysis, computational neuroscience, and computer vision.
- But variational inference has been studied less rigorously than MCMC, and its statistical properties are less well understood.
- However, there are problems for which we cannot easily use the MCMC approach, such as when *datasets are large or models are very complex*.

Variational inference: why? ii

- Rather than use sampling, the main idea behind variational inference is to use *optimization*. First, we posit a family of approximate densities $\mathcal{Q}$. This is a set of densities over the latent variables. Then, we try to find the member of that family that minimizes the Kullback-Leibler (KL) divergence to the exact posterior:

$$q^*(\boldsymbol{\theta}) = \underset{q(\boldsymbol{\theta}) \in \mathcal{Q}}{\operatorname{argmin}} \ \mathrm{KL}(q(\boldsymbol{\theta}) || \pi(\boldsymbol{\theta}|y)). \tag{1}$$

- Finally, we approximate the posterior with the optimized member of the family $q^*(\cdot)$.
- One of the key ideas behind variational inference is to choose $\mathcal{Q}$ to be flexible enough to capture a density close to $\pi(\boldsymbol{\theta}|y)$, but simple enough for efficient optimization.

Comparing VI and MCMC i

- MCMC algorithms sample a *Markov chain*; variational algorithms solve an optimization problem. MCMC algorithms approximate the posterior with samples from the chain; variational algorithms approximate the posterior with the result of the optimization.
- MCMC methods tend to be more computationally intensive than variational inference but they also provide guarantees of producing (asymptotically) exact samples from the target density.
- Variational inference does not enjoy such guarantees—it can only find a density close to the target—but tends to be faster than MCMC.
- VI is suited to large datasets and scenarios where we want to quickly explore many models; MCMC is suited to smaller datasets and scenarios where we happily pay a heavier computational cost for more precise samples.
- Another factor is the **geometry of the posterior distribution**: see the posterior distribution in *mixture models*.

Comparing VI and MCMC ii

- Exploring the interplay between model complexity and inference (and between variational inference and MCMC) is an exciting avenue for future research.
- The relative accuracy of variational inference and MCMC is still unknown.
- Moreover, as we'll see, there is a strong connection between VI and the expectation and maximization (EM) algorithm.
- Modern research on variational inference focuses on several aspects:
  - tackling Bayesian inference problems that involve massive data;
  - using improved optimization methods for solving Equation (1) (which is usually subject to local minima);
  - developing generic variational inference algorithms that are easy to apply to a wide class of models;
  - and increasing the accuracy of variational inference, for example, by stretching the boundaries of $\mathcal{Q}$ while managing complexity in optimization.

VI: the setup

Start with VI

- The goal of VI is to approximate a conditional density of latent variables given observed variables.
- We use a family of densities over the latent variables, parameterized by free "*variational parameters.*" The optimization finds the member of this family, that is, the setting of the parameters, which is closest in KL divergence to the conditional of interest.
- The fitted variational density then serves as a proxy for the exact conditional density.
- The inference problem is to compute the conditional density of the latent variables given the observations, $\pi(\boldsymbol{\theta}|y)$. As widely known, the denominator contains the marginal density of the observations, also called the evidence:

$$p(y) = \int p(\boldsymbol{\theta}, y) d\boldsymbol{\theta}. \tag{2}$$

- For many models, this evidence integral is unavailable in closed form or requires exponential time to compute.

The ELBO i

- Each $q(\boldsymbol{\theta}) \in \mathcal{Q}$ is a candidate approximation to the exact conditional. Once found, $q^*(\cdot)$ is the best approximation of the conditional, within the family $\mathcal{Q}$.
- However, this objective is not computable because it requires computing the logarithm of the evidence, $\log p(y)$ in Equation (2).
- To see why, recall that KL divergence is;

$$\begin{aligned} \mathrm{KL}(q(\boldsymbol{\theta})||\pi(\boldsymbol{\theta}|y)) &= \int \log\left(\frac{q(\boldsymbol{\theta})}{\pi(\boldsymbol{\theta}|y)}\right) q(\boldsymbol{\theta}) d\boldsymbol{\theta} \\ &= \mathrm{E}[\log q(\boldsymbol{\theta})] - \mathrm{E}[\log \pi(\boldsymbol{\theta}|y)] \\ &= \mathrm{E}[\log q(\boldsymbol{\theta})] - \mathrm{E}[\log p(\boldsymbol{\theta}, y)] + \log p(y), \end{aligned} \tag{3}$$

where the expectations are taken with respect to $q(\boldsymbol{\theta})$. This reveals its dependence on $\log p(y)$.

The ELBO ii

- Because we cannot compute the KL, we optimize an alternative objective that is equivalent to the KL up to an added constant,

$$\boxed{\text{ELBO}(q) = \mathrm{E}[\log p(\boldsymbol{\theta}, y)] - \mathrm{E}[\log q(\boldsymbol{\theta})].} \qquad (4)$$

- This function is called the evidence lower bound (ELBO) (see below). The ELBO is the negative KL divergence of Equation (3) plus $\log p(y)$, which is a constant with respect to $q(\boldsymbol{\theta})$. *Maximizing the ELBO is equivalent to minimizing the KL divergence*.
- We could rewrite the ELBO as:

$$\begin{aligned} \text{ELBO}(q) = &\ \mathrm{E}[\log p(\boldsymbol{\theta})] + \mathrm{E}[\log p(y|\boldsymbol{\theta})] - \mathrm{E}[\log q(\boldsymbol{\theta})] \\ = &\ \mathrm{E}[\log p(y|\boldsymbol{\theta})] - \text{KL}(q(\boldsymbol{\theta})||p(\boldsymbol{\theta})), \end{aligned} \qquad (5)$$

where the first term is an expected likelihood, and the second term is the negative divergence between the variational density and the prior. Thus, the variational objective mirrors the *usual balance between likelihood and prior*.

The ELBO iii

- Another property of the ELBO is that it lower-bounds the (log) evidence, $\log p(y) \geq \text{ELBO}(q)$ for any $q(\boldsymbol{\theta})$. This explains the name. To see this notice that Equation (3) and (4) give the following expression of the evidence:

$$\log p(y) = \text{KL}(q(\boldsymbol{\theta}) || \pi(\boldsymbol{\theta}|y)) + \text{ELBO}(q). \tag{6}$$

  The bound then follows from the fact that $\text{KL}(\cdot) \geq 0$.

- We notice that the first term of the ELBO in Equation (4) is the expected complete log-likelihood, which is optimized by the EM algorithm.
- Unlike variational inference, EM assumes the expectation under $\pi(\boldsymbol{\theta}|y)$ is computable and uses it in otherwise difficult parameter estimation problems.
- Unlike EM, variational inference does not estimate fixed model parameters—it is often used in a Bayesian setting where classical parameters are treated as latent variables.

The Mean-Field Variational family i

- The complexity of the family determines the complexity of the optimization; it is more difficult to optimize over a complex family than a simple family.
- We focus on the mean-field variational family, where the latent variables are mutually independent and each governed by a distinct factor in the variational density. A generic member of the mean-field variational family is:

$$q(\boldsymbol{\theta}) = \prod_{j=1}^{m} q_j(\theta_j). \tag{7}$$

  Each latent variable θ_j is governed by its own variational factor, the density $q_j(\theta_j)$. In optimization, these variational factors are chosen to maximize the ELBO of Equation (4).
- Notice we have not specified the parametric form of the individual variational factors. In principle, each can take on any parametric form appropriate to the corresponding random variable.

The Mean-Field Variational family ii

- The mean-field family is expressive because it can capture any marginal density of the latent variables. However, it cannot capture correlation between them. Seeing this in action reveals some of the intuitions and limitations of mean-field variational inference,as shown by two-dimensional Gaussian distributions in Figure 1.

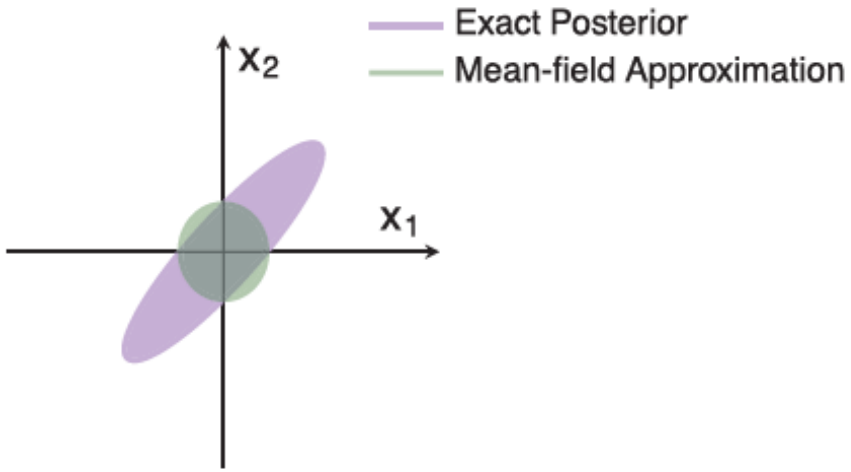


Figure 1. Visualizing the mean-field approximation to a two-dimensional Gaussian posterior. The ellipses show the effect of mean-field factorization. (The ellipses are 2σ contours of the Gaussian distributions.)

The Mean-Field Variational family iii

- While the variational approximation has the same mean as the original density, its covariance structure is, by construction, decoupled. Further, the marginal variances of the approximation under-represent those of the target density.
- The KL divergence in (3) penalizes placing mass in $q(\cdot)$ on areas where $\pi(\cdot)$ has little mass, but penalizes less the reverse.
- One way to expand the family is to add dependencies between the variables: this is called structured variational inference.

Coordinate Ascent Mean-Field VI (CAVI) algorithm

CAVI algorithm i

- Using the ELBO and the mean-field family, we have cast approximate conditional inference as an optimization problem.
- The CAVI algorithm iteratively optimizes each factor of the mean-field variational density, while holding the others fixed. It climbs the ELBO to a local optimum.
- We first state a result. Consider the j-th latent variable θ_j. The *complete conditional* of θ_j is its conditional density given all of the other latent variables in the model and the observations, $p(\theta_j|\boldsymbol{\theta}_{-j}, y)$.
- The optimal $q_j(\theta_j)$ is then proportional to the exponentiated expected log of the complete conditional,

$$q^*(\theta_j) \propto \exp\{\mathrm{E}_{-j}[\log p(\theta_j|\boldsymbol{\theta}_{-j}, y)]\}. \tag{8}$$

The expectation in (8) is with respect to the variational density over $\boldsymbol{\theta}_{-j}$, that is $\prod_{\ell \neq j} q_\ell(\theta_\ell)$.

CAVI algorithm ii

- Equivalently, Equation (8) is proportional to the exponentiated log of the joint,

$$q^*(\theta_j) \propto \exp\{\mathrm{E}_{-j}[\log p(\theta_j, \boldsymbol{\theta}_{-j}, y)]\}. \tag{9}$$

- Because of the mean-field property—all the latent variables are independent—the expectations on the right-hand side of (8) and (9) do not involve the j-th variational factor.
- We maintain a set of variational factors $q_\ell(\theta_\ell)$. We iterate through them, updating $q_\ell(\theta_\ell)$ using Equation (9). CAVI goes uphill on the ELBO of Equation (4), eventually finding a local optimum. Details are provided by Algorithm 1.

CAVI algorithm iii

Algorithm 1: Coordinate ascent variational inference (CAVI)

Input: A model $p(y, \theta)$, a data set y
Output: A variational density $q(\boldsymbol{\theta}) = \prod_{j=1}^{m} q_j(\theta_j)$
Initialize: Variational factors $q_j(\theta_j)$
while *the ELBO has not converged* **do**
  for $j \in \{1, \ldots, m\}$ **do**
    Set $q_j(\theta_j) \propto \exp\{\mathrm{E}_{-j}[\log p(\theta_j | \boldsymbol{\theta}_{-j}, y)]\}$
  end
  Compute $\mathrm{ELBO}(q) = \mathrm{E}[\log p(\boldsymbol{\theta}, y)] - \mathrm{E}[\log q(\boldsymbol{\theta})]$
end
return $q(\boldsymbol{\theta})$

- Finally, CAVI is closely related to Gibbs sampling, the classical workhorse of approximate inference. The GS maintains a realization of the latent variables and iteratively samples from each variable's complete conditional. Equation (8) uses the same complete conditional. It takes the expected log, and uses this quantity to iteratively set each variable's variational factor. Equation (8) is called the *coordinate update*.

CAVI algorithm iv

- *Derivation*: We now derive the coordinate update in Equation (9). Rewrite the ELBO of (4) as a function of the j-th variational factor $q_j(\theta_j)$, absorbing into a constant the terms that do not depend on it:

$$\text{ELBO}(q_j) = \mathrm{E}_j[\mathrm{E}_{-j}[\log p(\theta_j, \boldsymbol{\theta}_{-j}, y)]] - \mathrm{E}_j[\log q_j(\theta_j)] + \text{const.} \qquad (10)$$

  We have rewritten the first term of the ELBO using iterated expectation. The second term we have decomposed, using the independence of the variables (i.e., the mean-field assumption) and retaining only the term that depends on $q_j(\theta_j)$.

- Up to an added constant, the objective function in Equation (10) is equal to the negative KL divergence between $q_j(\theta_j)$ and $q_j^*(\theta_j)$ from Equation (9). Thus, we maximize the ELBO with respect to q_j when we set $q_j(\theta_j) = q_j^*(\theta_j)$.

CAVI algorithm v

- *Initialization*: The ELBO is (generally) a nonconvex objective function. CAVI only guarantees convergence to a local optimum, which can be sensitive to initialization.
- *Assessing convergence*: Monitoring the ELBO in CAVI is simple; we typically declare convergence once the change in ELBO falls below some small threshold. However, computing the ELBO of the full dataset may be undesirable.

Bayesian mixture of Gaussians i

- Consider a Bayesian mixture of unit-variance univariate Gaussians. There are K mixture components, corresponding to K Gaussian distributions with means $\boldsymbol{\mu} = \{\mu_1, \ldots, \mu_K\}$, assigned a common prior distribution $\mathcal{N}(0, \sigma^2)$, whereas the prior variance σ^2 is a hyperparameter. c_i denotes the cluster assignment, it indicates which latent cluster y_i comes from and is drawn from a categorical distribution over $\{1, \ldots, K\}$. We then draw data y_i from the corresponding Gaussian $\mathcal{N}(c_i^T \boldsymbol{\mu}, 1)$.
- The full model is:

$$\begin{aligned} y_i | c_i, \boldsymbol{\mu} &\sim \mathcal{N}(c_i^T \boldsymbol{\mu}, 1), && i = 1, \ldots, n \\ c_i &\sim \text{categorical}(1/K, \ldots, 1/K), && k = 1, \ldots, K \\ \mu_k &\sim \mathcal{N}(0, \sigma^2), && k = 1, \ldots, K. \end{aligned} \tag{11}$$

Bayesian mixture of Gaussians ii

- For a sample of size n, the joint density of latent and observed variables is:

$$p(\boldsymbol{\mu}, \boldsymbol{c}, y) = p(\boldsymbol{\mu}) \prod_{i=1}^{n} p(c_i) p(y_i | c_i, \boldsymbol{\mu}). \tag{12}$$

- Here the evidence is:

$$p(y) = \int p(\boldsymbol{\mu}) \prod_{i=1}^{n} \sum_{c_i} p(c_i) p(y_i | c_i, \boldsymbol{\mu}) d\boldsymbol{\mu}. \tag{13}$$

The integral in (13) does not reduce to a product of one-dimensional integrals over the μ_k's. Computing the evidence remains exponential in K, hence intractable.

Mean-field family for Gaussian mixtures i

- Now, suppose to assume the mean-field variational family, that contains approximate posterior densities of the form:

$$q(\boldsymbol{\mu}, \boldsymbol{c}) = \prod_{k=1}^{K} q(\mu_k; m_k, s_k^2) \prod_{i=1}^{n} q(c_i; \varphi_i). \tag{14}$$

- The factor $q(\mu_k; m_k, s_k^2)$ is a Gaussian distribution on the k-th mixture component's mean parameter; its mean is m_k and its variance is σ_k^2. The factor $q(c_i; \varphi_i)$ is a distribution on the i-th observation's mixture assignment; its assignment probabilities are a K-vector φ_i.
- With the variational family in place, we have completely specified the variational inference problem for the mixture of Gaussians. The ELBO is defined by the joint model density in Equation (12) and the mean-field variational family in Equation (14).

Mean-field family for Gaussian mixtures ii

- Two types of variational parameters: categorical parameters φ_i for approximating the posterior cluster assignment of the i-th data point and Gaussian parameters m_k and s_k^2 for approximating the posterior of the k-th mixture component.
- We combine the joint and the mean-field family to form the ELBO for the mixture of Gaussians.

$$\begin{aligned}\text{ELBO}(\boldsymbol{m}, \boldsymbol{s}^2, \boldsymbol{\varphi}) = & \sum_{k=1}^{K} \mathrm{E}[\log p(\mu_k); m_k, s_k^2)] + \\ & \sum_{i=1}^{n} (\mathrm{E}[\log p(c_i); \varphi_i] + \mathrm{E}[\log p(y_i | c_i, \boldsymbol{\mu}); \varphi_i, \boldsymbol{m}, \boldsymbol{s}^2]) - \\ & \sum_{i=1}^{n} \mathrm{E}[\log q(c_i; \varphi_i)] - \sum_{k=1}^{K} \mathrm{E}[\log q(\mu_k; m_k, s_k^2)].\end{aligned} \tag{15}$$

CAVI for mixtures i

- In each term, we have made explicit the dependence on the variational parameters. Each expectation can be computed in closed form.
- The CAVI algorithm updates each variational parameter in turn. We first derive the *update for the variational cluster assignment factor*; we then derive the *update for the variational mixture component factor*.
- Using Equation (9):

$$q^*(c_i; \varphi_i) \propto \exp\{\log p(c_i) + \mathrm{E}[\log p(y_i | c_i, \boldsymbol{\mu}); \boldsymbol{m}, \boldsymbol{s}^2]\}. \tag{16}$$

The expectation in the second term is over the mixture components $\boldsymbol{\mu}$.

CAVI for mixtures ii

- For the second term in Equation (16), we can write: $p(y_i|c_i, \boldsymbol{\mu}) = \prod_{k=1}^{K} p(y_i|\mu_k)^{c_{ik}}$, and then we can compute the expected log probability as (after some steps, see the paper Blei et al. (2017) for technical details, and/or try on your own!):

$$\mathrm{E}[\log p(y_i|c_i, \boldsymbol{\mu})] = \sum_k c_{ik}(\mathrm{E}[\mu_k; m_k, s_k^2[y_i - \mathrm{E}[\mu_k^2; m_k, s_k^2]/2) + \text{const.} \tag{17}$$

  In each line we remove terms that are constant with respect to c_i.

- Thus, the variational update for the i-th cluster assignment is

$$\varphi_{ik} \propto \exp\{\mathrm{E}[\mu_k; m_k, s_k^2]y_i - \mathrm{E}[\mu_k^2; m_k, s_k^2]/2\}. \tag{18}$$

CAVI for mixtures iii

- Now we turn to the variational density $q(\mu_k; m_k, s_k^2)$ of the k-th mixture component. Again we use Equation (9) and write down the joint density up to a normalizing constant:

$$q(\mu_k) \propto \exp\left\{\log p(\mu_k) + \sum_{i=1}^{n} \mathrm{E}[\log p(y_i|c_i, \boldsymbol{\mu}]; \varphi_i, \boldsymbol{m}_{-k}, s_{-k}^2]\right\}. \quad (19)$$

- Recall φ_{ik} is the probability that the i-th observation come from the k-th cluster. Because c_i is an indicator vector, $\varphi_{ik} = \mathrm{E}[c_{ik}; \varphi_i]$. We compute now the unnormalized logarithm for the coordinate-optimal in Equation (14). As Blei et al. (2017) report (see them for computational details):

$$\log q(\mu_k) = \left(\sum_i \varphi_{ik} y_i\right) \mu_k - \left(1/2\sigma^2 + \sum_i \varphi_{ik}/2\right) \mu_k^2 + const. \quad (20)$$

CAVI for mixtures iv

- This calculation in (20) reveals that the coordinate-optimal variational density of μ_k is an exponential family with sufficient statistics $\{\mu_k, \mu_k^2\}$ and natural parameters $\{\sum_i \varphi_{ik} y_i, -1/2\sigma^2 - \sum_i \varphi_{ik}/2\}$, that is, a Gaussian.
- Expressed in terms of the variational mean and variance, the updates for $q(\mu_k)$ are

$$m_k = \frac{\sum_i \varphi_{ik} y_i}{1/\sigma^2 + \sum_i \varphi_{ik}}, \quad s_k^2 = \frac{1}{1/\sigma^2 + \sum_i \varphi_{ik}}. \tag{21}$$

These updates relate closely to the complete conditional density of the k-th component in the mixture model. The complete conditional is a posterior Gaussian given the data assigned to the k-th component. The variational update is a weighted complete conditional, where each data point is weighted by its variational probability of being assigned to component k. For all the details, see Algorithm 2.

Algorithm 2: CAVI for a Gaussian mixture model

Input: Data set y, number of components K, prior variance of component means σ^2
Output: Variational densities $q(\mu_k; m_k, s_k^2)$ (Gaussian) and $q(c_i; \varphi_i)$ (K-categorical)
Initialize: Variational parameters $\boldsymbol{m} = m_{1:K}, \boldsymbol{s}^2 = s^2_{1:K}$, and $\boldsymbol{\varphi} = \varphi_{1:n}$.
while *the ELBO has not converged* **do**
  for $i \in \{1, \ldots, n\}$ **do**
    Set $\varphi_{ik} \propto \exp\{\mathrm{E}[\mu_k; m_k, s_k^2] y_i - \mathrm{E}[\mu_k^2; m_k, s_k^2]/2\}$
  end
  for $k \in \{1, \ldots, K\}$ **do**
    Set $m_k \leftarrow \frac{\sum_i \varphi_{ik} y_i}{1/\sigma^2 + \sum_i \varphi_{ik}}$
    Set $s_k^2 \leftarrow \frac{1}{1/\sigma^2 + \sum_i \varphi_{ik}}$
  end
  Compute ELBO$(\boldsymbol{m}, \boldsymbol{s}^2, \boldsymbol{\varphi})$
end
return $q(\boldsymbol{m}, \boldsymbol{s}^2, \boldsymbol{\varphi})$

CAVI for mixtures vi

- The algorithm 2 requires computing the ELBO in (15). We use the ELBO to track the progress of the algorithm and assess when it has converged.
- Once we have a fitted variational density, we can use it as we would use the posterior. For instance, we can assign points to their most likely mixture assignment $\hat{c}_i$ and estimate cluster means with their variational means m_k.

VI with exponential families

VI with exponential families

- The mixture of Gaussians is one member of the important class of models where each complete conditional is in the exponential family.
- This includes a number of widely used models, such as Bayesian mixtures of exponential families, factorial mixture models, matrix factorization models, certain hierarchical regression models (e.g., linear regression, probit regression, Poisson regression), stochastic blockmodels of networks, hierarchical mixtures of experts, and a variety of mixed-membership models.
- Working in this family simplifies variational inference: it is easier to derive the corresponding CAVI algorithm, and it enables variational inference to scale up to massive data.

Complete conditionals in the exponential family

- Consider the generic model $p(y, \theta)$ and suppose each complete conditional is in the exponential family:

$$\pi(\theta_j | \boldsymbol{\theta}_{-j}, y) = h(\theta_j) \exp\{\eta_j(\boldsymbol{\theta}_{-j}, y)^T \theta_j - a(\eta_j(\boldsymbol{\theta}_{-j}, y))\}, \tag{22}$$

  where θ_j is its own sufficient statistic, $h(\cdot)$ is a base measure, and $a(\cdot)$ is the log normalizer.

- Consider mean-field variational inference for this class of models, where we fit $q(\boldsymbol{\theta}) = \prod_j q_j(\theta_j)$. The coordinate update is:

$$q(\theta_j) \propto h(\theta_j) \exp\{\mathrm{E}[\eta_j(\boldsymbol{\theta}_{-j}, y)]^T \theta_j\}. \tag{23}$$

  Each one is in the same exponential family as its corresponding complete conditional.

- Let ν_j denote the variational parameter for the j-th variational factor. When we update each factor, we set its parameter equal to the expected parameter of the complete conditional, $\nu_j = \mathrm{E}[\eta_j(\boldsymbol{\theta}_{-j}, y)]$.

Conjugate models i

- One important special case of exponential family models are conditionally conjugate models with local and global variables. Models like this come up frequently in Bayesian statistics and statistical machine learning, where the global variables are the "parameters" and the local variables are per-data-point latent variables.
- Let β be a vector of global latent variables, which potentially govern any of the data. Let z be a vector of local latent variables, whose i-th component only governs data in the i-th "context." The joint density is

$$p(\beta, z, y) = \pi(\beta) \prod_{i=1}^{n} p(z_i, y_i | \beta). \tag{24}$$

The mixture of Gaussian in (11) is an example. The global variables are the mixture components; the i-th local variable is the cluster assignment for data point y_i.

Conjugate models ii

- We will assume that the modeling terms of Equation (24) are chosen to ensure each complete conditional is in the exponential family:

$$p(z_i, y_i|\beta) = h(z_i, y_i) \exp\{\beta^T t(z_i, y_i) - a(\beta)\}, \tag{25}$$

  where $t(\cdot, \cdot)$ is the sufficient statistic.

- Next, we take the prior on the global variables to be the corresponding conjugate prior

$$\pi(\beta) = h(\beta) \exp\{\alpha^T [\beta, -a(\beta)] - a(\alpha)\}. \tag{26}$$

  This prior has natural (hyper)parameter $\alpha = [\alpha_1, \alpha_2]^T$ and sufficient statistics that concatenate the global variable and its log normalizer in the density of the local variables.

Conjugate models iii

- With the conjugate prior, the complete conditional of the global variables is in the same family. Its natural parameter is

$$\hat{\alpha} = \left[\alpha_1 + \sum_{i=1}^{n} t(z_i, y_i), \alpha_2 + n \right]^T .$$

- Given β and y_i, the local variable z_i is conditionally independent of the other local variables z_{-i} and other data y_{-i}. This follows from the form of the joint density in Equation (24). Thus

$$p(z_i | y_i, \beta, z_{-i}, y_{-i}) = p(z_i | y_i, \beta), \tag{27}$$

  and we further assume that this density is in an exponential family.

- We now describe CAVI for this general class of models.

- Write $q(\beta|\lambda)$ for the variational posterior approximation on β; we call λ the "*global variational parameter*." It indexes the same exponential family density as the prior.

Conjugate models iv

- Similarly, let the variational posterior $q(z_i|\varphi_i)$ on each local variable z_i be governed by a "*local variational parameter*" φ_i. It indexes the same exponential family density as the local complete conditional. CAVI iterates between updating each local variational parameter and updating the global variational parameter.
- The local variational update is

$$\varphi_i = \mathrm{E}[\eta(\beta, y_i)]$$

  This is an application of $\nu_j = \mathrm{E}[\eta_j(\boldsymbol{\theta}_{-j}, y)]$, where we take the expectation of the natural parameter of the complete conditional in Equation (27).
- The global variational update applies the same technique. It is

$$\lambda = \left[\alpha_1 + \sum_{i=1}^{n} \mathrm{E}_{\varphi_i}[t(z_i, y_i)], \alpha_2 + n\right]^T.$$

Conjugate models v

- CAVI optimizes the ELBO by iterating between local updates of each local parameter and global updates of the global parameters.
- The ELBO is

$$\begin{aligned} \text{ELBO} = & \left(\alpha_1 + \sum_{i=1}^{n} \mathrm{E}_{\varphi_i}[t(z_i, y_i)] \right)^T \mathrm{E}_\lambda[\beta] \\ & - (\alpha_2 + n)\mathrm{E}_\lambda[a(\beta)] - \mathrm{E}[\log q(\beta, z)]. \end{aligned} \tag{28}$$

  This is the ELBO in applied to the joint in Equation (24) and the corresponding mean-field variational density; we have omitted terms that do not depend on the variational parameters.
- CAVI for the mixture of Gaussians model (Algorithm 2) is an instance of this method.
- See Blei et al. (2017) for full computational details.

Stochastic variational Inference (SVI)

Stochastic variational inference i

- Modern applications of probability models often require analyzing massive data. However, most posterior inference algorithms do not easily scale. CAVI is no exception, particularly in the conditionally conjugate setting of the previous section.
- The reason is that the coordinate ascent structure of the algorithm requires iterating through the entire dataset at each iteration. As the dataset size grows, each iteration becomes more computationally expensive.
- An alternative to coordinate ascent is *gradient-based optimization*, which climbs the ELBO by computing and following its gradient at each iteration. This perspective is the key to scaling up variational inference using stochastic variational inference (SVI), a method that combines natural gradients and stochastic optimization.

Stochastic variational inference ii

- SVI focuses on optimizing the global variational parameters λ of a conditionally conjugate model. The flow of computation is simple. The algorithm maintains a current estimate of the global variational parameters. It repeatedly (a) subsamples a data point from the full dataset; (b) uses the current global parameters to compute the optimal local parameters for the subsampled data point; and (c) adjusts the current global parameters in an appropriate way. SVI is detailed in Algorithm 3.
- In gradient-based optimization, the natural gradient accounts for the geometric structure of probability parameters.

Stochastic variational inference iii

Algorithm 3: SVI for conditionally conjugate models

Input: A model $p(y, \theta)$, a data set y, step size sequence ϵ_t
Output: Global variational density $q_\lambda(\beta)$ **Initialize**: Variational parameters λ_0
while *TRUE* **do**
  Choose a data point uniformly at random,
  $t \sim \text{Unif}(1, \ldots, n)$
  Optimize its local variational parameters
  $\varphi_t^* = \mathrm{E}[\eta(\beta, y_t)]$
  Compute the coordinate update as though y_t were repeated n times,
  $\hat{\lambda} = \alpha + n\mathrm{E}[\varphi_t^* f(\theta_t, y_t)]$
  Update the global variational parameter,
  $\lambda_t = (1 - \epsilon_t)\lambda_{t-1} + \epsilon_t \hat{\lambda}_t$
  end
return λ

- Conditionally conjugate models enjoy simple natural gradients of the ELBO of the form:

Stochastic variational inference iv

$$g(\lambda) = \mathrm{E}_{\varphi}[\hat{\alpha}] - \lambda, \tag{29}$$

that is the difference between the coordinate updates $\mathrm{E}_{\varphi}[\hat{\alpha}]$ and the variational parameters λ at which we are evaluating the gradient.

- We can use this natural gradient in a gradient-based optimization algorithm. At each iteration, we update the global parameters

$$\lambda_t = \lambda_{t-1} + \epsilon_t g(\lambda_t), \tag{30}$$

where ϵ_t is the step size.

- Substituting Equation (29) into the second term reveals a special structure,

$$\lambda_t = (1 - \epsilon_t)\lambda_{t-1} + \epsilon_t \mathrm{E}_{\varphi}[\hat{\alpha}] \tag{31}$$

Stochastic variational inference v

- Notice this does not require additional types of calculations other than those for coordinate ascent updates. At each iteration, we first compute the coordinate update. We then adjust the current estimate to be a weighted combination of the update and the current variational parameter. With massive data, this is prohibitively expensive.
- SVI solves this problem by using the natural gradient in a stochastic optimization algorithm.
- We could perhaps construct a *noisy unbiased natural gradient* by sampling an index t from the data. The noisy gradient only requires calculations from the coordinate ascent algorithm.
- We emphasize that SVI requires no new derivation beyond what is needed for CAVI. Any implementation of CAVI can be immediately scaled up to a stochastic algorithm.

Softwares and implementation

Available softwares

- There are many packages/softwares around to deal with VI methods.
- In this course, the `R` code accompanying the lecture (see the official Moodle page) shows how to use the `CmdStan` ecosystem.
- It relies on Stan and works as a wrapper: you compile in C++ a Stan model and then you can fit it by using MCMC techniques, VI methods (such as the "Pathfinder" algorithm or the ADVI procedure), Laplace approximation, penalized likelihood estimation.
- Then, you can use many of the packages of the Stan ecosystem to display posterior estimates (`posterior`, `bayesplot`), to compare models (`loo`), and so on.
- SVI can be used in `Pyro`, another probabilistic programming language.
- `PyMC` allows to use VI in Python.
- `LINFA` is another Python package for VI methods.
- VIBES is a software package which allows variational inference to be performed automatically on a Bayesian network.
- And many others!

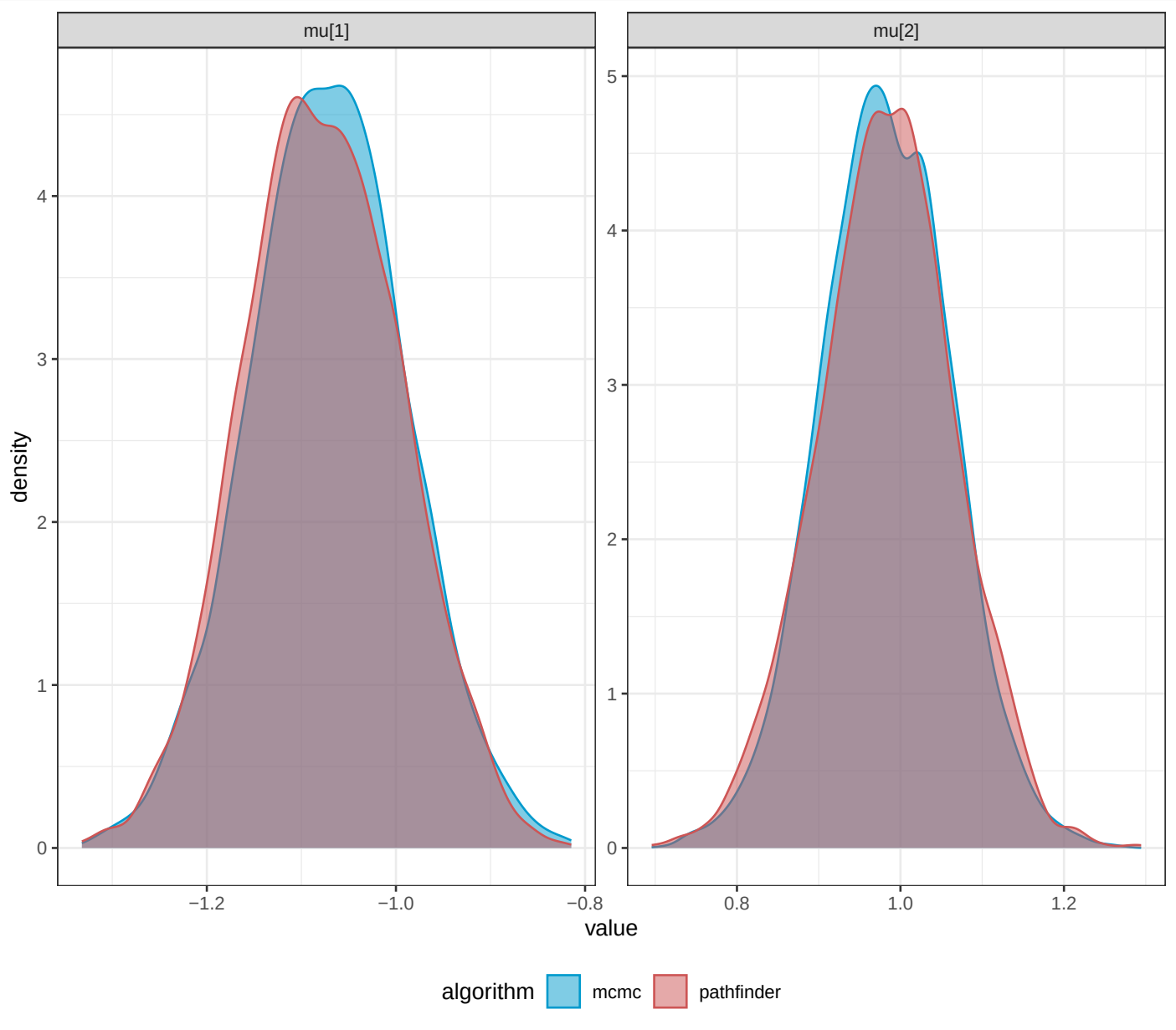
Two words about Pathfinder

- Variational method for approximately sampling from differentiable probability densities.
- Starting from a random initialization, Pathfinder locates normal approximations to the target density along a quasi-Newton optimization path (with L-BFGS procedure), with local covariance estimated using the inverse Hessian estimates produced by the optimizer.
- Pathfinder returns draws from the approximation with the lowest estimated Kullback-Leibler (KL) divergence to the target distribution.
- See Zhang et al. (2022) for further details. Check the R code accompanying the lecture in the Moodle page.

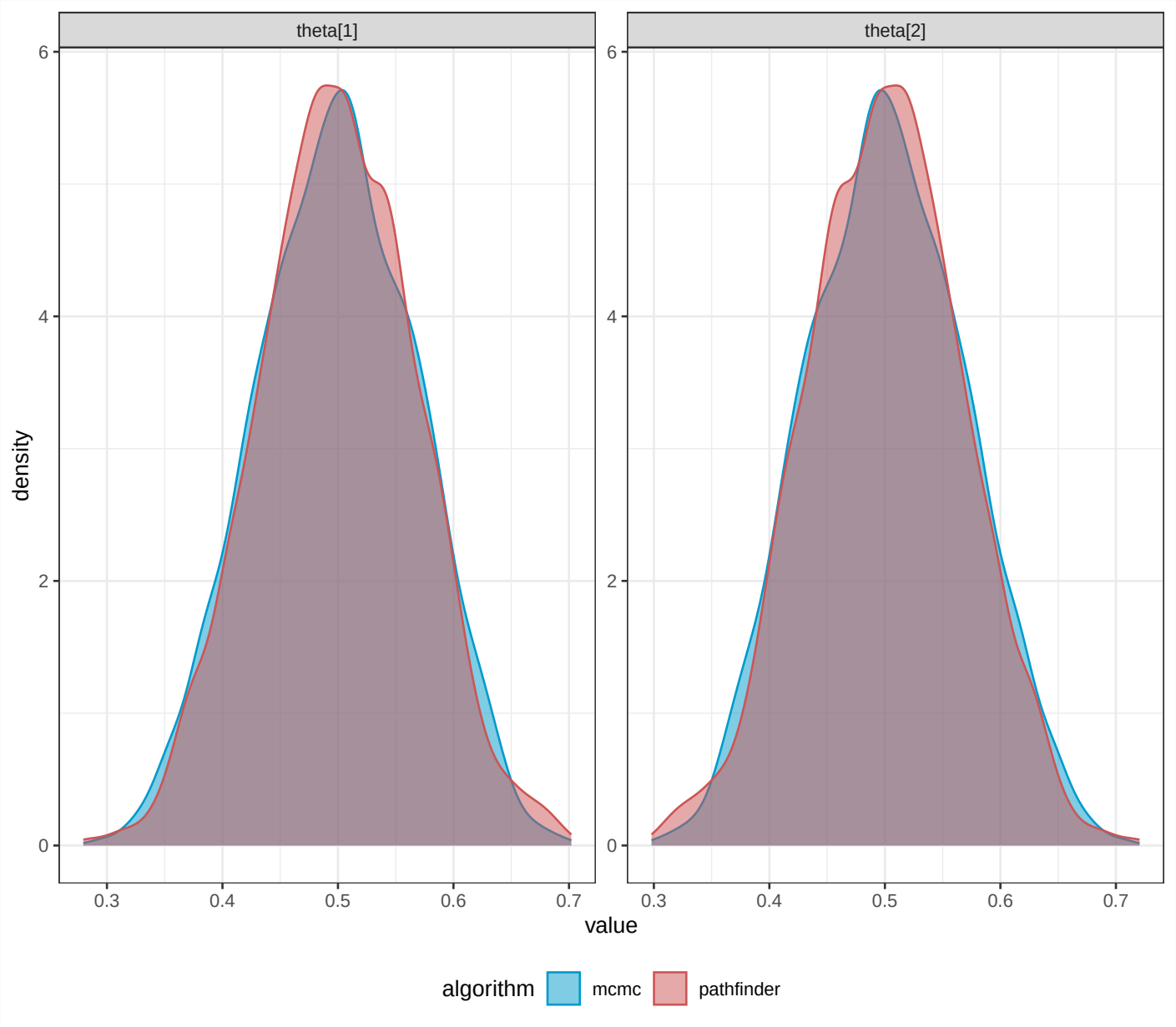
Pathfinder with CmdStan

```
# compile a model
file <- file.path(cmdstan_path(), "examples", "bernoulli", "bernoulli.stan")
mod <- cmdstan_model(file)
data_list <- list(N = 10, y = c(0,1,0,0,0,0,0,0,0,1))

# Variational 'pathfinder'
fit_pf <- mod$pathfinder(
  data = data_list,
  seed = 123,
  draws = 4000)

# plot fit
mcmc_hist(fit_pf$draws("theta"), binwidth = 0.025) +
  ggplot2::labs(subtitle = "Approximate posterior from pathfinder") +
  ggplot2::xlim(0, 1)
```

Bayesian mixture: Pathfinder VS MCMC, means



Bayesian mixture: Pathfinder VS MCMC, proportions



Bayesian mixture: Pathfinder VS MCMC

```
fit_pf$print()
    variable   mean median   sd  mad     q5    q95
 lp_approx__   2.74   3.06 1.24 1.04   0.22   4.11
 lp__        -62.41 -62.09 1.22 1.03 -64.86 -61.06
 theta[1]      0.50   0.50 0.07 0.07   0.39   0.61
 theta[2]      0.50   0.50 0.07 0.07   0.39   0.61
 mu[1]        -1.08  -1.08 0.08 0.08  -1.21  -0.94
 mu[2]         0.98   0.98 0.08 0.08   0.84   1.12
```

```
fit_mcmc$print()
 variable   mean median   sd  mad     q5    q95 rhat ess_bulk ess_tail
 lp__     -57.32 -57.01 1.19 0.99 -59.82 -56.01 1.00     1552     2270
 theta[1]   0.50   0.50 0.07 0.07   0.38   0.61 1.00     2767     2315
 theta[2]   0.50   0.50 0.07 0.07   0.39   0.62 1.00     2767     2315
 mu[1]     -1.07  -1.07 0.08 0.08  -1.21  -0.94 1.00     2300     2302
 mu[2]      0.98   0.98 0.08 0.08   0.85   1.11 1.00     4326     3251
```

Further reading

To properly grasp VI methods, I suggest you the further reading:

- *Variational inference: A review for statisticians.* by Blei, D. M., Kucukelbir, A., & McAuliffe, J. D. (2017). *Journal of the American statistical Association*, 112(518), 859-877.

More advanced papers and texts:

- *Pattern recognition and machine learning* by Bishop, C. (2006). Springer google schola, 2, 5-43.
- *Stochastic variational inference* by Hoffman, M. D., Blei, D. M., Wang, C., & Paisley, J. (2013). *The Journal of Machine Learning Research*.
- *Pathfinder: Parallel quasi-Newton variational inference.* by Zhang, L., Carpenter, B., Gelman, A., & Vehtari, A. (2022). *The Journal of Machine Learning Research*.