Lecture : Variational Inference

Riashat Islam
Slides courtesy of David Blei

Reasoning and Learning Lab
McGill University

31st October 2018

Probabilistic Machine Learning

- A probabilistic model is a joint distribution of hidden variables $\mathbf{z}$ and observed variables $\mathbf{x}$,

$$p(\mathbf{z}, \mathbf{x}).$$

- Inference about the unknowns is through the **posterior**, the conditional distribution of the hidden variables given the observations

$$p(\mathbf{z} \mid \mathbf{x}) = \frac{p(\mathbf{z}, \mathbf{x})}{p(\mathbf{x})}.$$

- For most interesting models, the denominator is not tractable. We appeal to **approximate posterior inference**.

Approximate Inference

- When using probabilistic graphical models, we will be interested in evaluating the posterior distribution $p(\mathbf{Z}|\mathbf{X})$ of the latent variables $\mathbf{Z}$ given the observed data $\mathbf{X}$.
- For example, in the EM algorithm, we need to evaluate the expectation of the complete-data log-likelihood with respect to the posterior distribution over the latent variables.
- For more complex models, it may be infeasible to evaluate the posterior distribution, or compute expectations with respect to this distribution.
- This typically occurs when working with high-dimensional latent spaces, or when the posterior distribution has a complex form, for which expectations are not analytically tractable (e.g. Boltzmann machines).
- We will examine a range of deterministic approximation schemes, some of which scale well to large applications.

Computational Challenge

Remember: the big challenge is computing the posterior distribution. There are several main approaches:

- Analytical integration: If we use "conjugate" priors, the posterior distribution can be computed analytically (we saw this in case of Bayesian linear regression).

- Gaussian (Laplace) approximation: Approximate the posterior distribution with a Gaussian. Works well when there is a lot of data compared to the model complexity (as posterior is close to Gaussian).

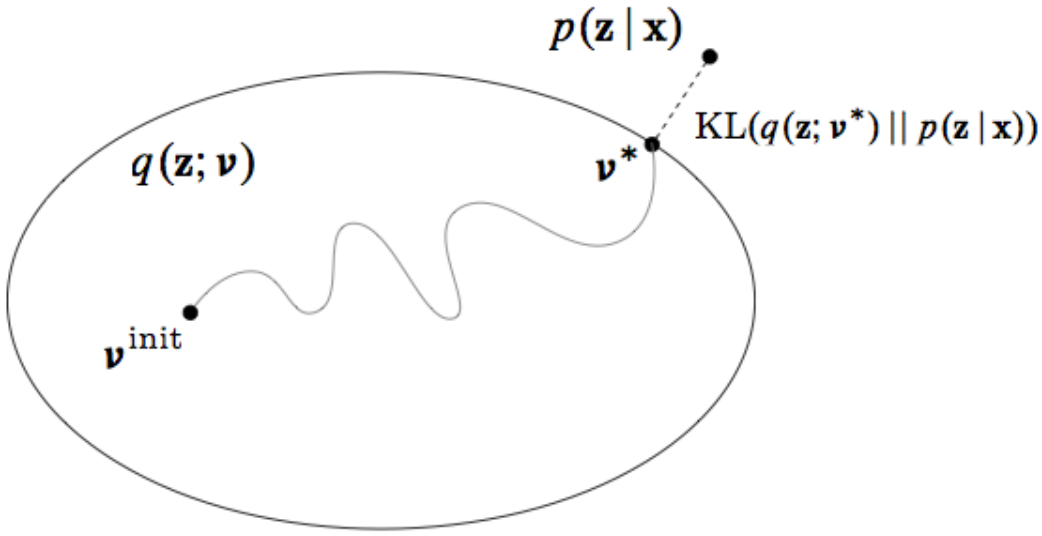
- Monte Carlo integration: The dominant current approach is Markov Chain Monte Carlo (MCMC) -- simulate a Markov chain that converges to the posterior distribution. It can be applied to a wide variety of problems.

- Variational approximation: A cleverer way to approximate the posterior. It often works much faster, but not as general as MCMC.

Probabilistic Model

- Suppose that we have a fully Bayesian model in which all parameters are given prior distributions.
- The model may have latent variables and parameters, and we will denote the set of all latent variables and parameters by $\mathbf{Z}$.
- We will also denote the set of all observed variables by $\mathbf{X}$.
- For example, we may be given a set of N i.i.d data points, so that $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ and $\mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_N\}$ (as we saw in our previous class).
- Our probabilistic model specifies the joint distribution $P(\mathbf{X}, \mathbf{Z})$.
- Our goal is to find approximate posterior distribution $P(\mathbf{Z}|\mathbf{X})$ and the model evidence $p(\mathbf{X})$.

Variational Inference



- VI turns **inference into optimization.**
- Posit a **variational family** of distributions over the latent variables,

$$q(\mathbf{z}; \boldsymbol{\nu})$$

- Fit the **variational parameters** $\boldsymbol{\nu}$ to be close (in KL) to the exact posterior.
  (There are alternative divergences, which connect to algorithms like EP, BP, and others.)

Variational Bound

- As in our previous lecture, we can decompose the marginal log-probability as:

$$\log p(\mathbf{X}) = \mathcal{L}(q) + \mathrm{KL}(q||p),$$

where

$$\mathcal{L}(q) = \int q(\mathbf{Z}) \ln \frac{p(\mathbf{X}, \mathbf{Z})}{q(\mathbf{Z})} \mathrm{d}\mathbf{Z}$$

$$\mathrm{KL}(q||p) = -\int q(\mathbf{Z}) \ln \frac{p(\mathbf{Z}|\mathbf{X})}{q(\mathbf{Z})} \mathrm{d}\mathbf{Z}.$$

- Note that parameters are now stochastic variables and are absorbed into **Z**.

- We can maximize the variational lower bound $\mathcal{L}(q)$ with respect to the distribution q(**Z**), which is equivalent to minimizing the KL divergence.

- If we allow any possible choice of q(**Z**), then the maximum of the lower bound occurs when:

$$q(\mathbf{Z}) = p(\mathbf{Z}|\mathbf{X}).$$

In this case KL divergence becomes zero.

Variational Bound

- As in our previous lecture, we can decompose the marginal log-probability as:

$$\log p(\mathbf{X}) = \mathcal{L}(q) + \mathrm{KL}(q||p),$$

- We will assume that the true posterior distribution is intractable.

- We can consider a restricted family of distributions q(**Z**) and then find the member of this family for which KL is minimized.

- Our goal is to restrict the family of distributions so that it contains only tractable distributions.

- At the same time, we want to allow the family to be sufficiently rich and flexible, so that it can provide a good approximation to the posterior.

- One option is to use parametric distributions q(**Z**|ω), governed by parameters ω.

- The lower bound then becomes a function of ω, and we can optimize the lower-bound to determine the optimal values for the parameters.

Motivation : Topic Modelling

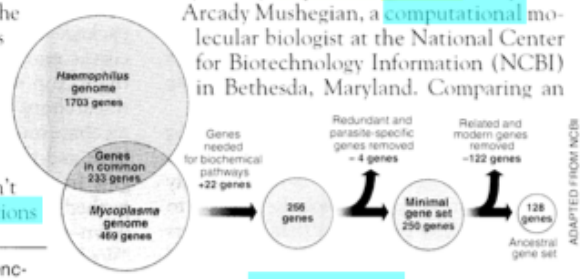
Topic models use posterior inference to discover the hidden thematic structure in a large collection of documents.

Example : Latent Dirichlet Allocation

Seeking Life's Bare (Genetic) Necessities

COLD SPRING HARBOR, NEW YORK—How many genes does an organism need to survive? Last week at the genome meeting here,* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.

Although the numbers don't match precisely, those predictions "are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing an



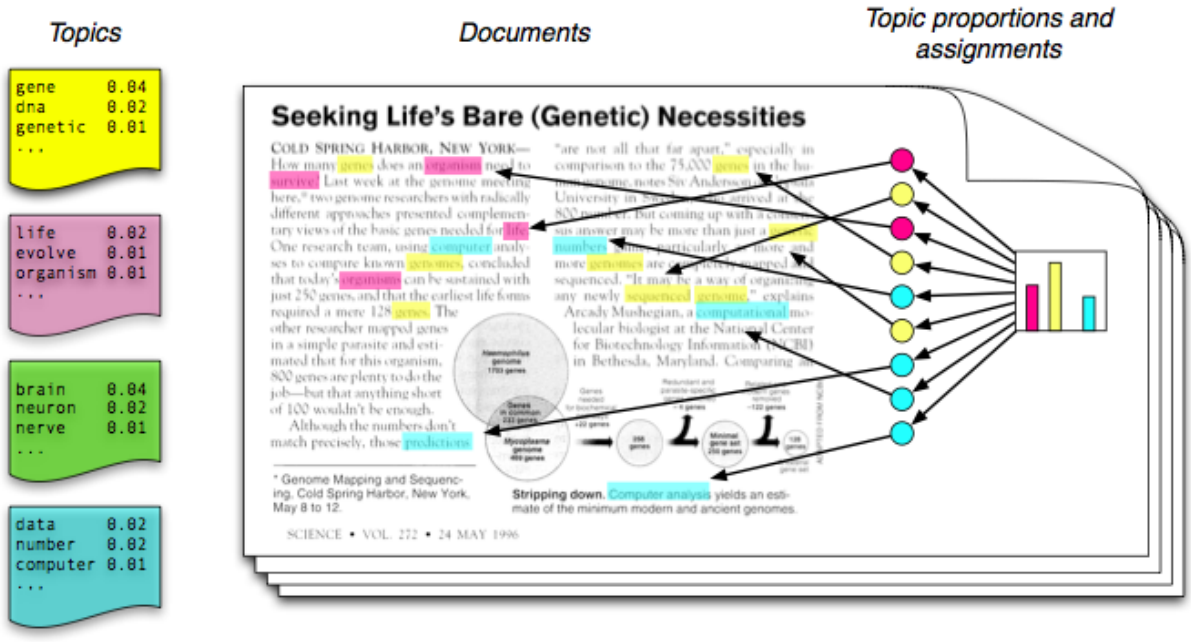
Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.

* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

SCIENCE • VOL. 272 • 24 MAY 1996

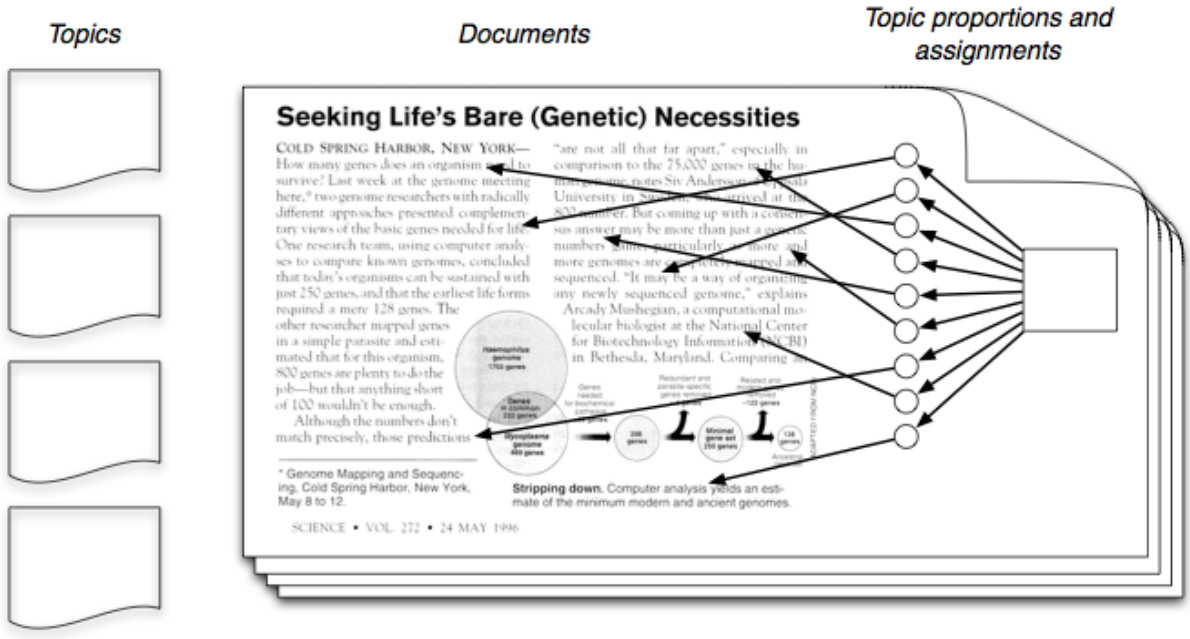
Documents exhibit multiple topics.

Example : Latent Dirichlet Allocation



- Each **topic** is a distribution over words
- Each **document** is a mixture of corpus-wide topics
- Each **word** is drawn from one of those topics

Example : Latent Dirichlet Allocation

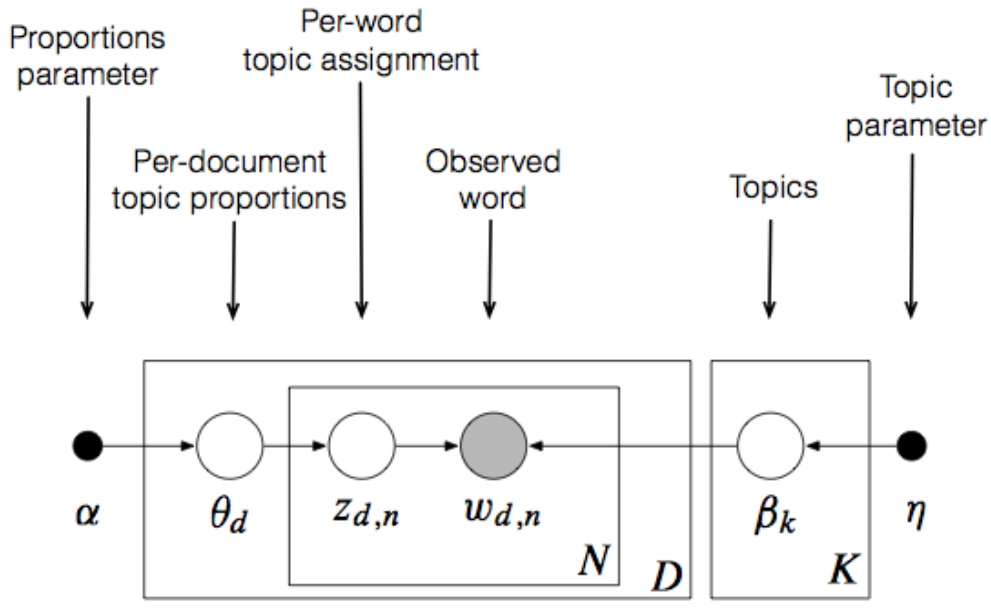


- But we only observe the documents; everything else is hidden.
- So we want to calculate the posterior

$$p(\text{topics, proportions, assignments} \mid \text{documents})$$

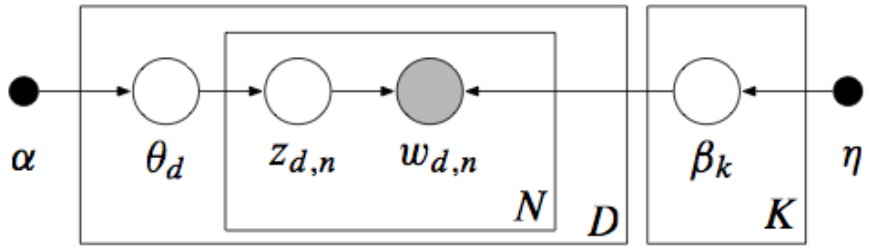
(Note: millions of documents; billions of latent variables)

LDA as Graphical Model



- Encodes **assumptions** about data with a factorization of the joint
- Connects assumptions to **algorithms** for computing with data
- Defines the **posterior** (through the joint)

Posterior Inference in LDA



- The posterior of the latent variables given the documents is

$$p(\beta, \theta, \mathbf{z} \mid \mathbf{w}) = \frac{p(\beta, \theta, \mathbf{z}, \mathbf{w})}{\int_\beta \int_\theta \sum_{\mathbf{z}} p(\beta, \theta, \mathbf{z}, \mathbf{w})}.$$

- We can't compute the denominator, the marginal $p(\mathbf{w})$.
- We use approximate inference.

The Evidence Lower Bound (ELBO)

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

- KL is intractable; VI optimizes the **evidence lower bound** (ELBO) instead.
  - It is a lower bound on $\log p(\mathbf{x})$.
  - Maximizing the ELBO is equivalent to minimizing the KL.
- The ELBO trades off two terms.
  - The first term prefers $q(\cdot)$ to place its mass on the MAP estimate.
  - The second term encourages $q(\cdot)$ to be diffuse.
- Caveat: The ELBO is not convex.

Stochastic Gradients of ELBO

Variational Inference Recipe

Start with a model:

$$p(\mathbf{z}, \mathbf{x})$$

Variational Inference Recipe

Choose a variational approximation:

$$q(\mathbf{z}; \boldsymbol{\nu})$$

Variational Inference Recipe

Write down the ELBO:

$$\mathscr{L}(\boldsymbol{\nu}) = \mathbb{E}_{q(\mathbf{z};\boldsymbol{\nu})}[\log p(\mathbf{x}, \mathbf{z}) - \log q(\mathbf{z}; \boldsymbol{\nu})]$$

Variational Inference Recipe

Compute the expectation(integral):

Example: $\mathscr{L}(v) = xv^2 + \log v$

Variational Inference Recipe

Take derivatives:

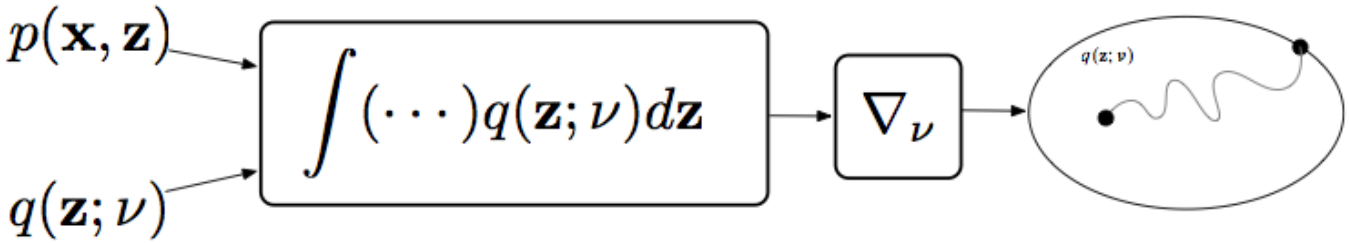
Example: $\nabla_{\nu}\mathcal{L}(\nu) = 2x\nu + \frac{1}{\nu}$

Variational Inference Recipe

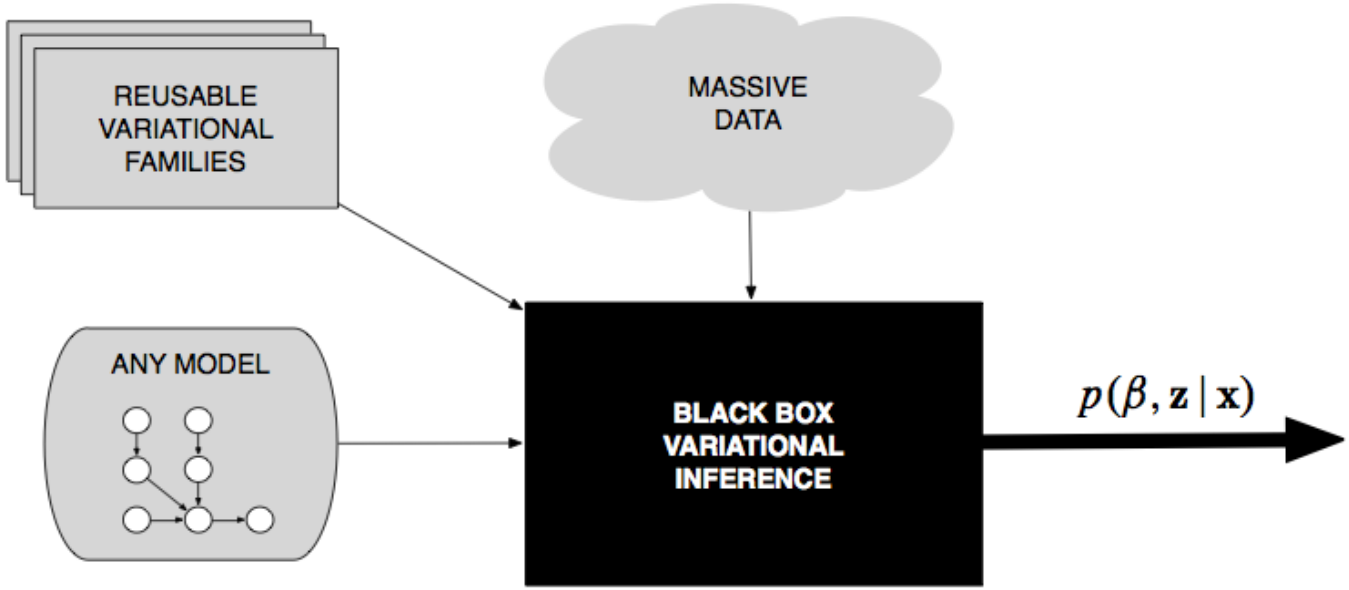
Optimize:

$$\nu_{t+1} = \nu_t + \rho_t \nabla_\nu \mathcal{L}$$

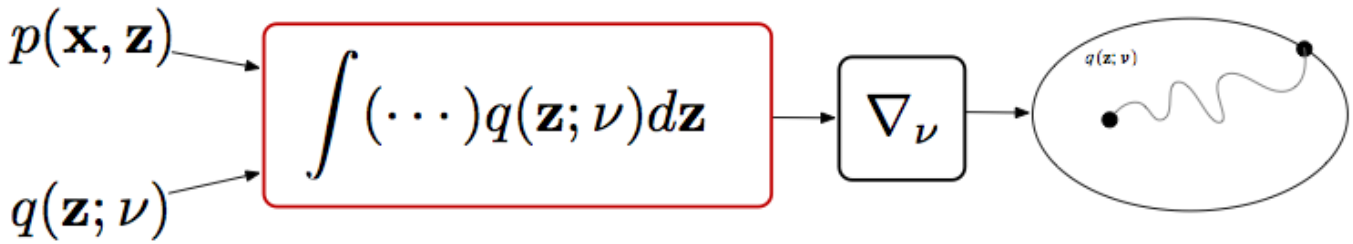
Variational Inference Recipe



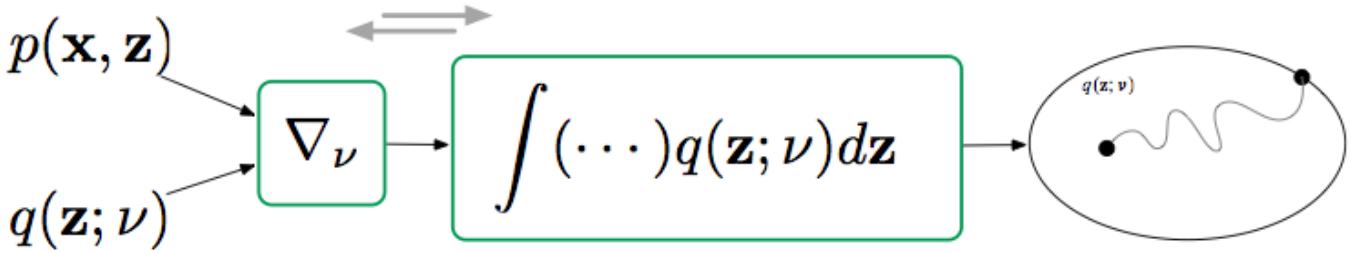
Black Box Variational Inference



Problem of Classical VI



New VI Recipe



Use stochastic optimization!

Computing Gradients of Expectations

- Define

$$g(\mathbf{z}, \nu) = \log p(\mathbf{x}, \mathbf{z}) - \log q(\mathbf{z}; \nu)$$

- What is $\nabla_\nu \mathcal{L}$

$$\begin{aligned}
\nabla_\nu \mathcal{L} &= \nabla_\nu \int q(\mathbf{z}; \nu) g(\mathbf{z}, \nu) d\mathbf{z} \\
&= \int \nabla_\nu q(\mathbf{z}; \nu) g(\mathbf{z}, \nu) + q(\mathbf{z}; \nu) \nabla_\nu g(\mathbf{z}, \nu) d\mathbf{z} \\
&= \int q(\mathbf{z}; \nu) \nabla_\nu \log q(\mathbf{z}; \nu) g(\mathbf{z}, \nu) + q(\mathbf{z}; \nu) \nabla_\nu g(\mathbf{z}, \nu) d\mathbf{z} \\
&= \mathbb{E}_{q(\mathbf{z};\nu)}[\nabla_\nu \log q(\mathbf{z}; \nu) g(\mathbf{z}, \nu) + \nabla_\nu g(\mathbf{z}, \nu)]
\end{aligned}$$

Using $\nabla_\nu \log q = \frac{\nabla_\nu q}{q}$

Score Function Gradients of ELBO

Score Function Estimator

Recall

$$\nabla_\nu \mathcal{L} = \mathbb{E}_{q(\mathbf{z};\nu)}[\nabla_\nu \log q(\mathbf{z};\nu) g(\mathbf{z},\nu) + \nabla_\nu g(z,\nu)]$$

Simplify:

$$\mathbb{E}_q[\nabla_\nu g(\mathbf{z},\nu)] = \mathbb{E}_q[\nabla_\nu \log q(\mathbf{z};\nu)] = 0$$

Gives the gradient:

$$\nabla_\nu \mathcal{L} = \mathbb{E}_{q(\mathbf{z};\nu)}[\nabla_\nu \log q(\mathbf{z};\nu)(\log p(\mathbf{x},\mathbf{z}) - \log q(\mathbf{z};\nu))]$$

Sometimes called likelihood ratio or REINFORCE gradients

[Glynn 1990; Williams, 1992; Wingate+ 2013; Ranganath+ 2014; Mnih+ 2014]

Noisy Unbiased Gradients

Gradient: $\mathbb{E}_{q(\mathbf{z};\nu)}[\nabla_\nu \log q(\mathbf{z};\nu)(\log p(\mathbf{x},\mathbf{z}) - \log q(\mathbf{z};\nu))]$

Noisy unbiased gradients with Monte Carlo!

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

where $\mathbf{z}_s \sim q(\mathbf{z};\nu)$

Black Box Variational Inference

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

Pathwise Gradients of ELBO

Pathwise Estimator

Assume

1. $\mathbf{z} = t(\boldsymbol{\epsilon}, \boldsymbol{\nu})$ for $\boldsymbol{\epsilon} \sim s(\boldsymbol{\epsilon})$ implies $\mathbf{z} \sim q(\mathbf{z}; \boldsymbol{\nu})$
   Example:

$$\epsilon \sim \text{Normal}(0, 1)$$
$$z = \epsilon\sigma + \mu$$
$$\rightarrow z \sim \text{Normal}(\mu, \sigma^2)$$

2. $\log p(\mathbf{x}, \mathbf{z})$ and $\log q(\mathbf{z})$ are differentiable with respect to $\mathbf{z}$

Pathwise Estimator

Recall

$$\nabla_\nu \mathcal{L} = \mathbb{E}_{q(\mathbf{z};\nu)}[\nabla_\nu \log q(\mathbf{z};\nu) g(\mathbf{z},\nu) + \nabla_\nu g(z,\nu)]$$

Rewrite using using $\mathbf{z} = t(\epsilon, \nu)$

$$\nabla_\nu \mathcal{L} = \mathbb{E}_{s(\epsilon)}[\nabla_\nu \log s(\epsilon) g(t(\epsilon,\nu),\nu) + \nabla_\nu g(t(\epsilon,\nu),\nu)]$$

To differentiate:

$$\begin{aligned}\nabla \mathcal{L}(\nu) &= \mathbb{E}_{s(\epsilon)}[\nabla_\nu g(t(\epsilon,\nu),\nu)] \\ &= \mathbb{E}_{s(\epsilon)}[\nabla_\mathbf{z}[\log p(\mathbf{x},\mathbf{z}) - \log q(\mathbf{z};\nu)]\nabla_\nu t(\epsilon,\nu) - \nabla_\nu \log q(\mathbf{z};\nu)] \\ &= \mathbb{E}_{s(\epsilon)}[\nabla_\mathbf{z}[\log p(\mathbf{x},\mathbf{z}) - \log q(\mathbf{z};\nu)]\nabla_\nu t(\epsilon,\nu)]\end{aligned}$$

This is also known as the reparameterization gradient.

[Glasserman 1991; Fu 2006; Kingma+ 2014; Rezende+ 2014; Titsias+ 2014]

Score Function vs Pathwise Estimator

Score Function

- Differentiates the density $\nabla_\nu q(z; \nu)$
- Works for discrete and continuous models
- Works for large class of variational approximations
- Variance can be a big problem

Pathwise

- Differentiates the function $\nabla_z[\log p(\mathbf{x}, \mathbf{z}) - \log q(\mathbf{z}; \nu)]$
- Requires differentiable models
- Requires variational approximation to have form $\mathbf{z} = t(\epsilon, \nu)$
- Generally better behaved variance