S&DS 365 / 665 Intermediate Machine Learning

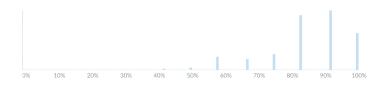
Variational Inference

October 7

Reminders

- Assignment 2 due Wednesday
- Midterm week from today, in class
- Material up to and including today's class
- Cheat sheet: One side of 8.5x11 sheet, handwritten
- Multiple review sessions scheduled (see Canvas)
- Quiz 3 scores posted; can discuss in review sessions
- Practice midterms posted





For Today

- Variational inference: The ELBO
- Derivations and examples
- Next time: Variational autoencoders (VAEs)



Recall: Inverting generative models

Template for generative model:

- Choose Z
- ② Given z, generate (sample) X

We often want to invert this:

- Given x
- What is Z that generated it?

Inverting models

Bayesian setup:

- **1** Choose θ
- 2 Given θ , generate (sample) X

Posterior inference:

- Given x
- **2** What is θ that generated it?

Approximate inference

If we have a random vector $Z \sim p(Z \mid x)$, we might want to compute the following:

- marginal probabilities $\mathbb{P}(Z_i = z \mid x)$
- marginal means $\mathbb{E}(Z_i = z \mid x)$
- most probable assignments $z^\star = \operatorname{\mathsf{arg}} \max_{Z} \mathbb{P}(\{Z_i = z_i\} \,|\, x)$
- maximum marginals $z_i^* = \arg \max_{z_i} \mathbb{P}(Z_i = z_i \mid x)$
- joint probability $\mathbb{P}(Z \mid x)$
- joint mean $\mathbb{E}(Z \mid x)$

Each of these quantities is intractable to calculate exactly, in general.

7

Gibbs sampler

Generic algorithm:

- Sample one component holding others fixed
- 2 Iterate

Gibbs sampler

Generic algorithm:

- Choose index *j* randomly
- **2** Sample θ_i fixing θ_i , $i \neq j$ and x
- 3 Iterate

Gibbs sampler

Generic algorithm:

- Choose index *j* randomly
- ② Sample θ_i from $\mathbb{P}(\theta_i | \theta_i, i \neq j, x)$.
- 3 Iterate

When run for sufficiently long, this will draw a sample from the posterior $\mathbb{P}(\theta \mid x)$.

8

Variational methods

- Gibbs sampling is stochastic approximation
- Variational methods iteratively refine deterministic approximations
- Variational and Markov chain approximations originated in physics

Stochastic approximation—Ising model

Gibbs sampler

Iterate until converged:

- **1** Choose vertex $s \in V$ at random
- 2 Sample z_s holding others fixed

$$egin{aligned} heta_{\mathcal{S}} &= \mathsf{sigmoid}\left(eta_{\mathcal{S}} + \sum_{t \in \mathcal{N}(\mathcal{S})} eta_{\mathit{st}} z_{t}
ight) \ Z_{\mathcal{S}} \,|\, heta_{\mathcal{S}} \sim \mathsf{Bernoulli}(heta_{\mathcal{S}}) \end{aligned}$$

You should verify that this is the correct conditional distribution of Z_s holding the others fixed.

10

 $sigmoid(u) = 1/(1 + e^{-u})$

Deterministic approximation

Mean field variational algorithm

Iterate until converged:

- **1** Choose vertex $s \in V$ at random
- 2 Update mean μ_s holding others fixed

$$\mu_s = \operatorname{sigmoid}\left(\beta_s + \sum_{t \in N(s)} \beta_{st} \, \mu_t\right)$$

1:

Deterministic vs. stochastic approximation

- The z_s variables are random
- The μ_s variables are deterministic
- The Gibbs sampler convergence is in distribution
- The mean field convergence is numerical
- The Gibbs sampler approximates the full distribution
- The mean field algorithm approximates the mean of each node

Example 2: A finite mixture model

Fix two distributions F_0 and F_1 , with densities $f_0(x)$ and $f_1(x)$, and form the mixture model

$$heta \sim \mathsf{Beta}(lpha,eta) \ X \, | \, heta \sim heta \mathsf{F}_1 + (\mathsf{1} - heta) \mathsf{F}_0.$$

Example 2: A finite mixture model

Fix two distributions F_0 and F_1 , with densities $f_0(x)$ and $f_1(x)$, and form the mixture model

$$heta \sim \mathrm{Beta}(lpha,eta) \ X \, | \, heta \sim heta F_1 + (1- heta) F_0.$$

The likelihood for data x_1, \ldots, x_n is

$$p(x_{1:n}) = \int_0^1 \text{Beta}(\theta \mid \alpha, \beta) \prod_{i=1}^n (\theta f_1(x_i) + (1-\theta) f_0(x_i)) d\theta.$$

Our goal is to approximate the posterior $p(\theta \mid x_{1:n})$

Stochastic approximation

Gibbs sampler

- **1** Sample $Z_i | \theta, x_{1:n}$ for i = 1, ..., n
- **2** Sample $\theta \mid z_{1:n}, x_{1:n}$

The first step is carried out by sampling

$$Z_i = \begin{cases} 1 & \text{with probability } \propto \theta f_1(x_i) \\ 0 & \text{with probability } \propto (1 - \theta) f_0(x_i) \end{cases}$$

Stochastic approximation

Gibbs sampler

- **1** Sample $Z_i | \theta, x_{1:n}$ for i = 1, ..., n
- **2** Sample $\theta \mid z_{1:n}, x_{1:n}$

The second step is carried out by sampling

$$\theta \sim \text{Beta}\left(\sum_{i=1}^{n} z_i + \alpha, n - \sum_{i=1}^{n} z_i + \beta\right).$$

Posterior over θ is approximated as *mixture* of Beta distributions; number of components is n+1

Variational inference: Strategy

- We'd like to compute $p(\theta, z | x)$, but it's too complicated.
- Strategy: Approximate as $q(\theta, z)$ that has a "nice" form
- q is a function of variational parameters, optimized for each x.
- Maximize a lower bound on p(x).

The ELBO is the following lower bound on $\log p(x)$:

$$\log p(x) = \int \sum_{z} q(z, \theta) \log p(x) d\theta$$

The ELBO is the following lower bound on $\log p(x)$:

$$\log p(x) = \int \sum_{z} q(z, \theta) \log p(x) d\theta$$
$$= \sum_{z} \int q(z, \theta) \log \left(\frac{p(x, z, \theta) q(z, \theta)}{p(z, \theta \mid x) q(z, \theta)} \right) d\theta$$

The ELBO is the following lower bound on $\log p(x)$:

$$\log p(x) = \int \sum_{z} q(z, \theta) \log p(x) d\theta$$

$$= \sum_{z} \int q(z, \theta) \log \left(\frac{p(x, z, \theta) q(z, \theta)}{p(z, \theta \mid x) q(z, \theta)} \right) d\theta$$

$$= \sum_{z} \int q(z, \theta) \log \left(\frac{p(x, z, \theta)}{q(z, \theta)} \right) d\theta + \sum_{z} \int q(z, \theta) \log \left(\frac{q(z, \theta)}{p(z, \theta \mid x)} \right) d\theta$$

The ELBO is the following lower bound on $\log p(x)$:

$$\log p(x) = \int \sum_{z} q(z,\theta) \log p(x) d\theta$$

$$= \sum_{z} \int q(z,\theta) \log \left(\frac{p(x,z,\theta) q(z,\theta)}{p(z,\theta \mid x) q(z,\theta)} \right) d\theta$$

$$= \sum_{z} \int q(z,\theta) \log \left(\frac{p(x,z,\theta)}{q(z,\theta)} \right) d\theta + \sum_{z} \int q(z,\theta) \log \left(\frac{q(z,\theta)}{p(z,\theta \mid x)} \right) d\theta$$

$$\geq \sum_{z} \int q(z,\theta) \log \left(\frac{p(x,z,\theta)}{q(z,\theta)} \right) d\theta$$

$$= H(q) + \mathbb{E}_{q}(\log p(x,z,\theta))$$

16

The ELBO is the following lower bound on $\log p(x)$:

$$\log p(x) = \int \sum_{z} q(z,\theta) \log p(x) d\theta$$

$$= \sum_{z} \int q(z,\theta) \log \left(\frac{p(x,z,\theta) q(z,\theta)}{p(z,\theta \mid x) q(z,\theta)} \right) d\theta$$

$$= \sum_{z} \int q(z,\theta) \log \left(\frac{p(x,z,\theta)}{q(z,\theta)} \right) d\theta + \sum_{z} \int q(z,\theta) \log \left(\frac{q(z,\theta)}{p(z,\theta \mid x)} \right) d\theta$$

$$\geq \sum_{z} \int q(z,\theta) \log \left(\frac{p(x,z,\theta)}{q(z,\theta)} \right) d\theta$$

$$= H(q) + \mathbb{E}_{q}(\log p(x,z,\theta))$$

We maximize this over the parameters of q



The inequality above uses concavity of the logarithm:

$$\log\left(\sum_{\alpha} \mathbf{w}_{\alpha} \mathbf{x}_{\alpha}\right) \geq \sum_{\alpha} \mathbf{w}_{\alpha} \log \mathbf{x}_{\alpha}$$

The inequality above uses concavity of the logarithm:

$$\log\left(\sum_{\alpha} w_{\alpha} x_{\alpha}\right) \geq \sum_{\alpha} w_{\alpha} \log x_{\alpha}$$

So, if $q_{\alpha} \geq 0$ and $p_{\alpha} \geq 0$ sum (or integrate) to one, then

$$0 = \log \left(\sum_lpha p_lpha
ight) = \log \left(\sum_lpha q_lpha rac{p_lpha}{q_lpha}
ight) \geq \sum_lpha q_lpha \log \left(rac{p_lpha}{q_lpha}
ight)$$

17

The inequality above uses concavity of the logarithm:

$$\log\left(\sum_{\alpha} w_{\alpha} x_{\alpha}\right) \geq \sum_{\alpha} w_{\alpha} \log x_{\alpha}$$

So, if $q_{\alpha} \ge 0$ and $p_{\alpha} \ge 0$ sum (or integrate) to one, then

$$0 = \log \left(\sum_lpha p_lpha
ight) = \log \left(\sum_lpha q_lpha rac{p_lpha}{q_lpha}
ight) \geq \sum_lpha q_lpha \log \left(rac{p_lpha}{q_lpha}
ight)$$

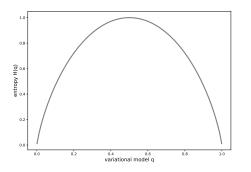
Therefore

$$\sum_{lpha} q_lpha \log \left(rac{q_lpha}{p_lpha}
ight) \geq 0$$

17

The ELBO is $H(q) + \mathbb{E}_q(\log p)$

The entropy term H(q) encourages q to be spread out:



The cross-entropy $\mathbb{E}_q \log p$ tries to match q to p

Example 2: A finite mixture model

Fix two distributions F_0 and F_1 , with densities $f_0(x)$ and $f_1(x)$, and form the mixture model

$$heta \sim \mathsf{Beta}(lpha,eta) \ X \, | \, heta \sim heta \mathsf{F}_1 + (\mathsf{1} - heta) \mathsf{F}_0.$$

Example 2: A finite mixture model

Fix two distributions F_0 and F_1 , with densities $f_0(x)$ and $f_1(x)$, and form the mixture model

$$heta \sim ext{Beta}(lpha, eta) \ X \, | \, heta \sim heta F_1 + (1 - heta) F_0.$$

The likelihood for data x_1, \ldots, x_n is

$$p(x_{1:n}) = \int_0^1 \text{Beta}(\theta \mid \alpha, \beta) \prod_{i=1}^n (\theta f_1(x_i) + (1-\theta) f_0(x_i)) d\theta.$$

Our goal is to approximate the posterior $p(\theta \mid x_{1:n})$

Our variational approximation is

$$q(z,\theta) = q(\theta \mid \gamma_1, \gamma_2) \prod_{i=1}^n q_i^{z_i} (1-q_i)^{(1-z_i)}$$

where $q(\theta \mid \gamma_1, \gamma_2)$ is a Beta (γ_1, γ_2) distribution, and $0 \le q_i \le 1$ are n free parameters.

Our variational approximation is

$$q(z,\theta) = q(\theta \mid \gamma_1, \gamma_2) \prod_{i=1}^n q_i^{z_i} (1-q_i)^{(1-z_i)}$$

where $q(\theta \mid \gamma_1, \gamma_2)$ is a Beta (γ_1, γ_2) distribution, and $0 \le q_i \le 1$ are n free parameters.

Need to maximize ELBO $H(q) + \mathbb{E}_q \log p$

Let's sketch part of the calculation

First, we have

$$\log p(x, \theta, z) = \log p(\theta \mid \alpha, \beta) + \sum_{i=1}^{n} \left\{ \log (\theta f_1(x_i))^{z_i} + \log ((1-\theta) f_0(x_i))^{(1-z_i)} \right\}$$

Next, take expectation with respect to q and use identities such as

$$\mathbb{E}_q \log \theta = \psi(\gamma_1) - \psi(\gamma_1 + \gamma_2)$$

for the digamma function $\psi(\cdot)$.

Next, take expectation with respect to q and use identities such as

$$\mathbb{E}_q \log \theta = \psi(\gamma_1) - \psi(\gamma_1 + \gamma_2)$$

for the digamma function $\psi(\cdot)$.

So, for example,

$$\mathbb{E}_{q} \log (\theta f_{1}(x_{i}))^{z_{i}} = q_{i} \mathbb{E}_{q} \log \theta + q_{i} \log f_{1}(x_{i})$$

$$= q_{i} (\psi(\gamma_{1}) - \psi(\gamma_{1} + \gamma_{2})) + q_{i} \log f_{1}(x_{i})$$

So, for example,

$$\mathbb{E}_{q} \log (\theta f_{1}(x_{i}))^{z_{i}} = q_{i} \mathbb{E}_{q} \log \theta + q_{i} \log f_{1}(x_{i})$$

$$= q_{i} (\psi(\gamma_{1}) - \psi(\gamma_{1} + \gamma_{2})) + q_{i} \log f_{1}(x_{i})$$

After some calculus and algebra, we end up with the following algorithm (see the notes on the course web page for more detail)

Variational algorithm for mixture

Variational inference

Iterate the following steps for variational parameters $q_{1:n}$ and (γ_1, γ_2) :

1 Holding q_i fixed, set $\gamma = (\gamma_1, \gamma_2)$ to

$$\gamma_1 = \alpha + \sum_{i=1}^n q_i$$
 $\gamma_2 = \beta + n - \sum_{i=1}^n q_i$

Variational algorithm for mixture

Variational inference

Iterate the following steps for variational parameters $q_{1:n}$ and (γ_1, γ_2) :

1 Holding q_i fixed, set $\gamma = (\gamma_1, \gamma_2)$ to

$$\gamma_1 = \alpha + \sum_{i=1}^n q_i$$
 $\gamma_2 = \beta + n - \sum_{i=1}^n q_i$

2 Holding γ_1 and γ_2 fixed, set q_i to

$$q_i = \frac{f_1(x_i) \exp \psi(\gamma_1)}{f_1(x_i) \exp \psi(\gamma_1) + f_0(x_i) \exp \psi(\gamma_2)}$$

Variational algorithm for mixture

Variational inference

Iterate the following steps for variational parameters $q_{1:n}$ and (γ_1, γ_2) :

1 Holding q_i fixed, set $\gamma = (\gamma_1, \gamma_2)$ to

$$\gamma_1 = \alpha + \sum_{i=1}^n q_i$$
 $\gamma_2 = \beta + n - \sum_{i=1}^n q_i$

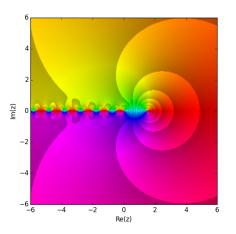
2 Holding γ_1 and γ_2 fixed, set q_i to

$$q_i = \frac{f_1(x_i) \exp \psi(\gamma_1)}{f_1(x_i) \exp \psi(\gamma_1) + f_0(x_i) \exp \psi(\gamma_2)}$$

After convergence, approximate posterior distribution over θ is

$$\widehat{p}(\theta \mid x_{1:n}) = \mathsf{Beta}(\theta \mid \gamma_1, \gamma_2)$$

Digamma function



 $\psi(\mathbf{x})$ is the digamma function $\label{eq:psi} \text{https://en.wikipedia.org/wiki/Digamma_function}$

Deterministic approximation

- Convergence is numerical, not stochastic
- Posterior is approximated as a single Beta
- Very similar algorithm is used for topic models

Example 3: More general mixtures

$$\theta \sim \mathsf{Dirichlet}(\alpha_1, \ldots, \alpha_k)$$

$$X \mid \theta \sim \theta_1 F_1 + \cdots + \theta_k F_k$$

Example 3: More general mixtures

$$heta \sim \mathsf{Dirichlet}(lpha_1, \dots, lpha_k)$$
 $X \mid heta \sim heta_1 F_1 + \dots + heta_k F_k$

The likelihood for single data point x is

$$p(x) = \int \mathsf{Dirichlet}(\theta \,|\, \alpha_1, \dots, \alpha_k) \left(\sum_{j=1}^k \theta_j f_j(x) \right) \, d\theta.$$

When distributions F_j are learned, this is a "topic model." Variational inference is one of the most useful ways of training topic models

- Notes on variational methods: Please read Sections 1–4
- The other sections are more advanced / specialized material
- Next up: Variational autoencoders
- VAEs not on midterm, but please review basics of variational methods discussed so far, including the ELBO



Q: What is the best q we could use?

Q: What is the best *q* we could use?

A: The true posterior $q(z, \theta \mid x) = p(z, \theta \mid x)$

Q: What is the best q we could use?

A: The true posterior $q(z, \theta \mid x) = p(z, \theta \mid x)$

Why? Because this maximizes the ELBO. Mathematically,

$$\sum_{z} \int q(z,\theta) \log \left(\frac{q(z,\theta)}{p(z,\theta \mid x)} \right) d\theta = 0$$

in this case, so the ELBO inequality is an equality.

Q: How does the ELBO regularize?

Q: How does the ELBO regularize?

A: The entropy term favors distributions that are "spread out"

Q: How does the ELBO regularize?

A: The entropy term favors distributions that are "spread out"

Why? For discrete distributions the maximum entropy distribution is uniform. For Gaussian, the entropy is $\log \sigma^2$ which favors σ^2 large.

Q: Is the ELBO easy to maximize?

Q: Is the ELBO easy to maximize?

A: No.

Q: Is the ELBO easy to maximize?

A: No.

Why? In general it is non-convex, and the solution depends on where we start an iterative algorithm. This is unlike the Gibbs sampler, which converges to the right thing if we wait long enough.

Next class: Variational autoencoders

- Variational autoencoders are generative models that are trained using variational inference
- The "decoder" is a neural net that generates from a latent variable
- The "encoder" approximates the posterior distribution with another neural network trained using variational inference

Summary

- Gibbs sampling makes stochastic approximations
- Variational methods make deterministic approximations
- General recipe: Maximize ELBO over variational parameters
- Gives a powerful approach to generative modeling