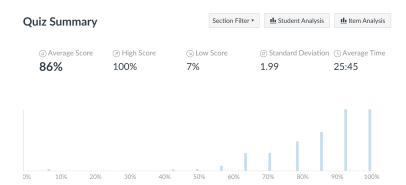
S&DS 365 / 665 Intermediate Machine Learning

## **Variational Inference**

October 9

#### Reminders

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



#### **Review sessions**

Chris: Thursday, Oct 12, 7pm

Ruixiao: Friday, Oct 13, 7pm

Zehao: Saturday, Oct 14, 4pm

## **For Today**

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



## Recall: Inverting generative models

Template for generative model:

- 1 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  $\theta$
- 2 Given  $\theta$ , generate (sample) X

#### Posterior inference:

- Given x
- **2** What is  $\theta$  that generated it?



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

• marginal probabilities  $\mathbb{P}(Z_i = z \mid x)$ 

8



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  $\mathbb{E}(Z_i = z \mid x)$



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  $z^* = \arg \max_{z} \mathbb{P}(\{Z_i = z_i\} \mid x)$



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  $z_i^* = \arg\max_{z_i} \mathbb{P}(Z_i = z_i \mid x)$

8

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  $z^* = \arg\max_{z} \mathbb{P}(\{Z_i = z_i\} \mid x)$
- maximum marginals  $z_i^* = \operatorname{arg\,max}_{z_i} \mathbb{P}(Z_i = z_i \mid x)$
- joint probability  $\mathbb{P}(Z \mid x)$

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  $\mathbb{E}(Z \mid x)$



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.

8

#### **Variational methods**

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

#### **Example: Ising model**

We have a graph with edges E and vertices V. Each node i has a random variable  $Z_i$  that can be "up" ( $Z_i = 1$ ) or "down" ( $Z_i = 0$ )

$$\mathbb{P}_{\beta}(z_1,\ldots,z_n) \propto \exp\left(\sum_{s\in V} \beta_s z_s + \sum_{(s,t)\in E} \beta_{st} z_s z_t\right)$$

This is called an "Ising model" and is central to statistical physics.

#### **Example: Ising model**

We have a graph with edges E and vertices V. Each node i has a random variable  $Z_i$  that can be "up" ( $Z_i = 1$ ) or "down" ( $Z_i = 0$ )

$$\mathbb{P}_{\beta}(z_1,\ldots,z_n) \propto \exp\left(\sum_{s\in V} \beta_s z_s + \sum_{(s,t)\in E} \beta_{st} z_s z_t\right)$$

E are the set of edges, V are the vertices. Imagine the  $Z_i$  are votes of politicians, and the edges encode the social network of party affiliations

10

#### Ising model, simple case

We have a graph with edges E and vertices V. Each node i has a random variable  $Z_i$  that can be "up" ( $Z_i = 1$ ) or "down" ( $Z_i = -1$ )

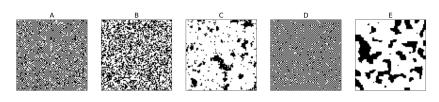
$$\mathbb{P}_{\beta}(z_1,\ldots,z_n)\propto \exp\left(\beta\sum_{(s,t)\in E}z_s\,z_t\right)$$

The sign of the parameter  $\beta$  effects whether neighboring states are encouraged to be the same ( $\beta > 0$ ) or different ( $\beta < 0$ )

*E* are the set of edges, *V* are the vertices.

1:

#### Ising model, simple case



(c) The figure above shows the result of running Gibbs sampling on a  $75 \times 75$  grid, using a fixed value of  $\alpha$  and five different values of  $\beta$ :

$$\beta\in\left\{-5,-\tfrac{1}{2},0,\tfrac{1}{2},5\right\}$$

Which is which? Explain your answer.

#### Ising model is the "discrete Gaussian"

The nodes can take values in  $\{0,1\}$  or  $\{-1,1\}$ —The two encodings are equivalent (exercise)

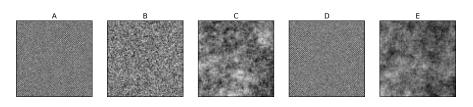
We can think of the Ising model as a "discrete Gaussian".

Why? The multivariate Gaussian and Ising models have exactly the same form, but the sample spaces are different.

The normalizing constant for the multivariate Gaussian has a closed form; not so for the Ising model.

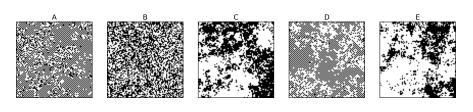
#### Ising model is the "discrete Gaussian"

Here is what we get when we sample Gaussians having the same parameters as in the midterm question:



#### Ising model is the "discrete Gaussian"

Here is what we get when we sample Gaussians having the same parameters as in the midterm question:



plotting the sign of  $z_s$ 

## Stochastic approximation

#### 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.

15

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

## **Deterministic approximation**

#### Mean field variational algorithm

Iterate until converged:

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

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

#### Determinnistic vs. stochastic approximation

- The z<sub>s</sub> variables are random
- The μ<sub>s</sub> 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

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

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

The first step is carried out by sampling

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

19

## 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  $\theta$  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).

#### Variational inference: The ELBO

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



#### Variational inference: The ELBO

The inequality above uses concavity of the logarithm:

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

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

$$0 = \log \left( \sum_lpha oldsymbol{p}_lpha 
ight) = \log \left( \sum_lpha oldsymbol{q}_lpha rac{oldsymbol{p}_lpha}{oldsymbol{q}_lpha} 
ight) \geq \sum_lpha oldsymbol{q}_lpha \log \left( rac{oldsymbol{p}_lpha}{oldsymbol{q}_lpha} 
ight)$$

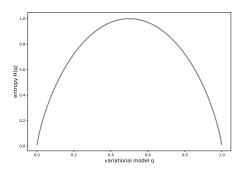
Therefore

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

#### Variational inference: The ELBO

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

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

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  $p(\theta \mid x_{1:n})$ 

#### Variational approximation

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 $(\gamma_1, \gamma_2)$  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

## Variational approximation

First, we have

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

Next we use identities such as

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

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

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  $(\gamma_1, \gamma_2)$ :

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

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

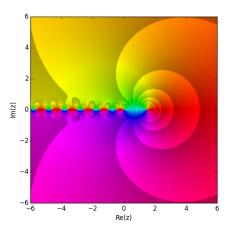
② Holding  $\gamma_1$  and  $\gamma_2$  fixed, set  $q_i$  to

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

After convergence, approximate posterior distribution over  $\theta$  is

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

# **Digamma function**



 $\psi(x)$  is the digamma function 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**

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

The likelihood for single data point x is

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

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  $\sigma^2$  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