DSC291: Machine Learning with Few Labels

Unsupervised Learning

Zhiting Hu

Lecture 5, January 20, 2023

UC San Diego
HALICIOĞLU DATA SCIENCE INSTITUTE

Recap: EM Algorithm

- Observed variables $\boldsymbol{x}$, latent variables $\boldsymbol{z}$
- To learn a model $p(\boldsymbol{x}, \boldsymbol{z}|\theta)$, we want to maximize the marginal log-likelihood

$$\ell(\theta; \boldsymbol{x}) = \log p(\boldsymbol{x}|\theta) = \log \sum_{z} p(\boldsymbol{x}, \boldsymbol{z}|\theta)$$

  - But it's too difficult
- EM algorithm:
  - maximize a lower bound of $\ell(\theta; \boldsymbol{x})$
  - Or equivalently, minimize an upper bound of $-\ell(\theta; \boldsymbol{x})$
- Key equation:

Evidence Lower Bound (ELBO)

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

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

Variational free energy

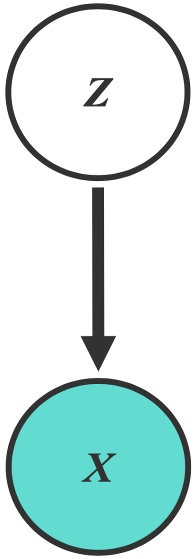
Recap: EM Algorithm

- The EM algorithm is coordinate-decent on $F(q, \theta)$
  - E-step: $q^{t+1} = \arg\min_q F\left(q, \theta^t\right) = p(\boldsymbol{z}|\boldsymbol{x}, \theta^t)$
    - the posterior distribution over the latent variables given the data and the current parameters
  - M-step: $\theta^{t+1} = \arg\min_\theta F\left(q^{t+1}, \theta^t\right) = \text{argmax}_\theta \sum_z q^{t+1}(\boldsymbol{z}|\boldsymbol{x}) \log p(\boldsymbol{x}, \boldsymbol{z}|\theta)$

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

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

Example: Gaussian Mixture Models (GMMs)



- Consider a mixture of K Gaussian components:
  - Z is a latent class indicator vector:

$$p(z_n) = \mathrm{multi}(z_n : \pi) = \prod_k (\pi_k)^{z_n^k}$$

  - X is a conditional Gaussian variable with a class-specific mean/covariance

$$p(x_n \mid z_n^k = 1, \mu, \Sigma) = \frac{1}{(2\pi)^{m/2}|\Sigma_k|^{1/2}} \exp\left\{-\tfrac{1}{2}(x_n - \mu_k)^T \Sigma_k^{-1}(x_n - \mu_k)\right\}$$

  - The likelihood of a sample:

mixture component

mixture proportion

$$\begin{aligned} p(x_n|\mu,\Sigma) &= \sum_k p(z^k = 1 \mid \pi)\, p(x, \mid z^k = 1, \mu, \Sigma) \\ &= \sum_{z_n} \prod_k \left((\pi_k)^{z_n^k} N(x_n : \mu_k, \Sigma_k)^{z_n^k}\right) = \sum_k \pi_k N(x, \mid \mu_k, \Sigma_k) \end{aligned}$$

Example: Gaussian Mixture Models (GMMs)

- Consider a mixture of K Gaussian components
- The expected complete log likelihood

$$\mathbb{E}_q\left[\ell_c(\boldsymbol{\theta}; x, z)\right] = \sum_n \mathbb{E}_q\left[\log p\left(z_n \mid \pi\right)\right] + \sum_n \mathbb{E}_q\left[\log p\left(x_n \mid z_n, \mu, \Sigma\right)\right]$$

$$= \sum_n \sum_k \mathbb{E}_q\left[z_n^k\right] \log \pi_k - \frac{1}{2} \sum_n \sum_k \mathbb{E}_q\left[z_n^k\right] \left((x_n - \mu_k)^T \Sigma_k^{-1} (x_n - \mu_k) + \log|\Sigma_k| + C \right)$$

- E-step: computing the posterior of z_n given the current estimate of the parameters (i.e., π , μ, Σ)

$p(z_n^k = 1, x, \mu^{(t)}, \Sigma^{(t)})$

$$p(z_n^k = 1 \mid x, \mu^{(t)}, \Sigma^{(t)}) = \frac{\pi_k^{(t)} N(x_n, \mid \mu_k^{(t)}, \Sigma_k^{(t)})}{\sum_i \pi_i^{(t)} N(x_n, \mid \mu_i^{(t)}, \Sigma_i^{(t)})}$$

$p(x, \mu^{(t)}, \Sigma^{(t)})$

Example: Gaussian Mixture Models (GMMs)

- E-step: computing the posterior of z_n given the current estimate of the parameters (i.e., π , μ, Σ)

$$
\begin{aligned}
p(z^k = 1 \mid \boldsymbol{x}) &= \frac{p(z^k = 1) p(\boldsymbol{x} \mid z^k = 1)}{p(\boldsymbol{x})} \\
&= \frac{p(z^k = 1) p(\boldsymbol{x} \mid z^k = 1)}{\sum_{j=1}^{K} p(z^j = 1) p(\boldsymbol{x} \mid z^j = 1)} \\
&= \frac{\pi_k \mathcal{N}(\boldsymbol{x} \mid \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x} \mid \mu_j, \Sigma_j)} \\
&:= \gamma_k
\end{aligned}
$$

Example: Gaussian Mixture Models (GMMs)

- M-step: computing the parameters given the current estimate of z_n
  - Once we have $q^{t+1}(z^k|x) = p(z^k|x, \theta^t) = \gamma^k$, we can compute the expected likelihood:

$$\theta^{t+1} = \mathrm{argmax}_\theta \sum_k q^{t+1}(z^k = 1|x) \log p(x, z^k = 1|\theta)$$

$$\begin{aligned} & \mathbb{E}_{q^{t+1}} [\log (p(\boldsymbol{x}, z \mid \boldsymbol{\theta}))] \\ = & \sum_k \gamma_k \left(\log p\left(z^k = 1|\boldsymbol{\theta}\right) + \log P\left(\boldsymbol{x} \mid z^k = 1, \boldsymbol{\theta}\right)\right) \\ = & \sum_k \gamma_k \log \pi_k + \sum_k \gamma_k \log \mathcal{N}(\boldsymbol{x}; \mu_k, \Sigma_k) \end{aligned}$$

  - We need to fit K Gaussians, just need to weight examples by γ_k

Example: Gaussian Mixture Models (GMMs)

- M-step: computing the parameters given the current estimate of z_n

$$\pi_k^* = \arg\max \langle l_c(\boldsymbol{\theta}) \rangle, \qquad \Rightarrow \frac{\partial}{\partial \pi_k} \langle l_c(\boldsymbol{\theta}) \rangle = 0, \forall k, \quad \text{s.t.} \sum_{\text{k}} \pi_k = 1$$

$$\Rightarrow \quad \pi_k^* = \sum_n \langle z_n^k \rangle_{q^{(t)}} \Big/ N = \sum_n \tau_n^{k(t)} \Big/ N = \langle n_k \rangle \Big/ N$$

$$\mu_k^* = \arg\max \langle l(\boldsymbol{\theta}) \rangle, \qquad \Rightarrow \quad \mu_k^{(t+1)} = \frac{\sum_n \tau_n^{k(t)} x_n}{\sum_n \tau_n^{k(t)}}$$

$$\Sigma_k^* = \arg\max \langle l(\boldsymbol{\theta}) \rangle, \qquad \Rightarrow \quad \Sigma_k^{(t+1)} = \frac{\sum_n \tau_n^{k(t)} (x_n - \mu_k^{(t+1)})(x_n - \mu_k^{(t+1)})^T}{\sum_n \tau_n^{k(t)}}$$

Fact:

$$\frac{\partial \log |\mathrm{A}^{-1}|}{\partial \mathrm{A}^{-1}} = \mathrm{A}^T$$

$$\frac{\partial \mathbf{x}^T \mathrm{A} \mathbf{x}}{\partial \mathrm{A}} = \mathbf{x}\mathbf{x}^T$$

EM Algorithm for GMM: Quick Summary

- Initialize the means μ_k , covariances Σ_k and mixing coefficients π_k
- Iterate until convergence:
  - E-step: Evaluate the posterior given current parameters

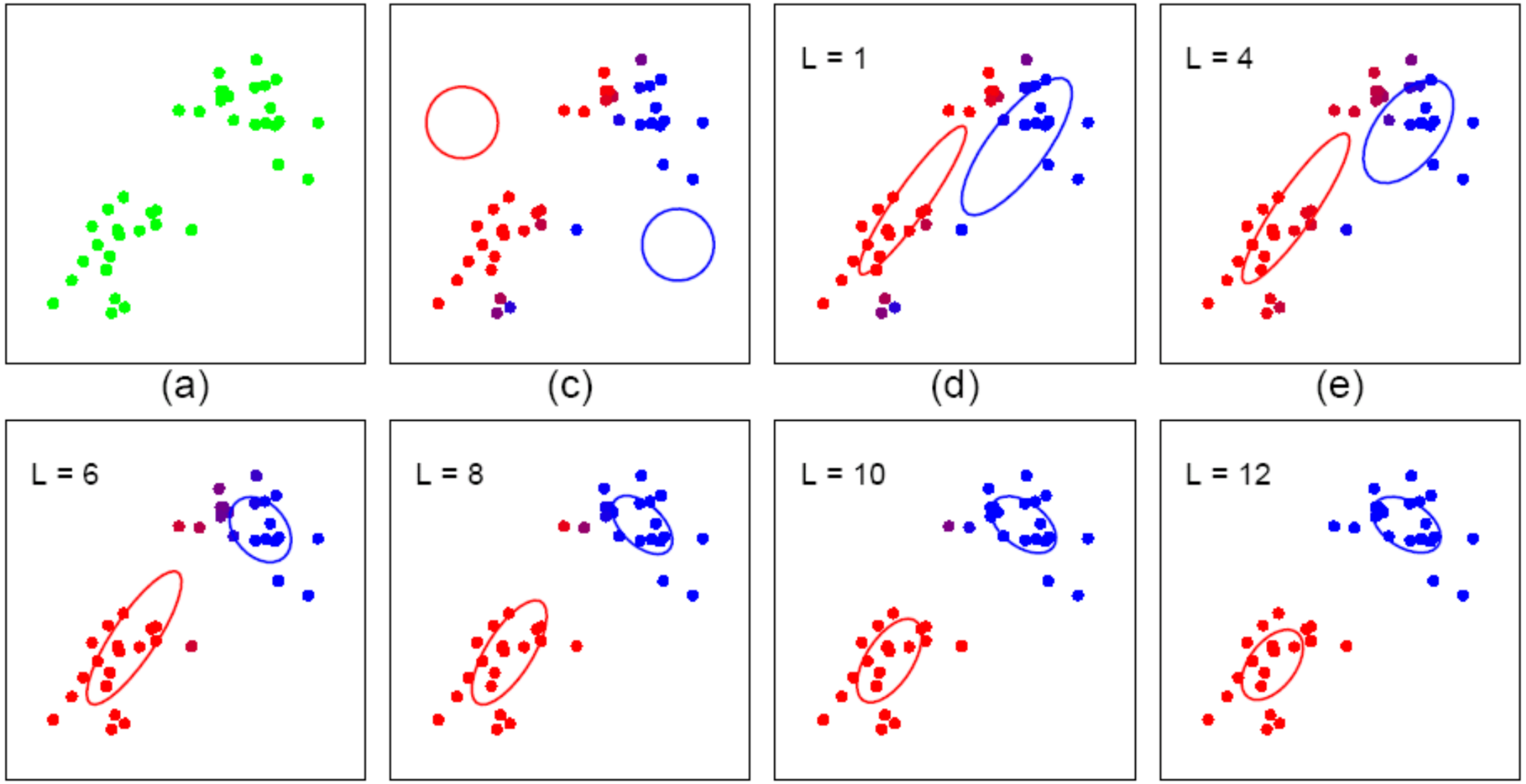
$$p(z^k = 1 \mid \boldsymbol{x}) = \frac{\pi_k \mathcal{N}(\boldsymbol{x} \mid \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x} \mid \mu_j, \Sigma_j)} := \gamma_k$$

  - M-step: Re-estimate the parameters given current posterior

$$\begin{aligned}
&\mathbb{E}_{q^{t+1}}\left[\log\left(p\left(\boldsymbol{x}, z \mid \boldsymbol{\theta}\right)\right)\right] \\
=& \sum_k \gamma_k \left(\log p\left(z^k = 1 | \boldsymbol{\theta}\right) + \log P\left(\boldsymbol{x} \mid z^k = 1, \boldsymbol{\theta}\right)\right) \\
=& \sum_k \gamma_k \log \pi_k + \sum_k \gamma_k \log \mathcal{N}\left(\boldsymbol{x}; \mu_k, \Sigma_k\right)
\end{aligned}$$

Example: Gaussian Mixture Models (GMMs)

- Start: "guess" the centroid μ_k and covariance Σ_k of each of the K clusters
- Loop:



(a) (c) (d) (e) (f) (g) (h) (i)

Summary: EM Algorithm

- A way of maximizing likelihood function for latent variable models. Finds MLE of parameters when the original (hard) problem can be broken up into two (easy) pieces
  - Estimate some “missing” or “unobserved” data from observed data and current parameters.
  - Using this “complete” data, find the maximum likelihood parameter estimates.

Summary: EM Algorithm

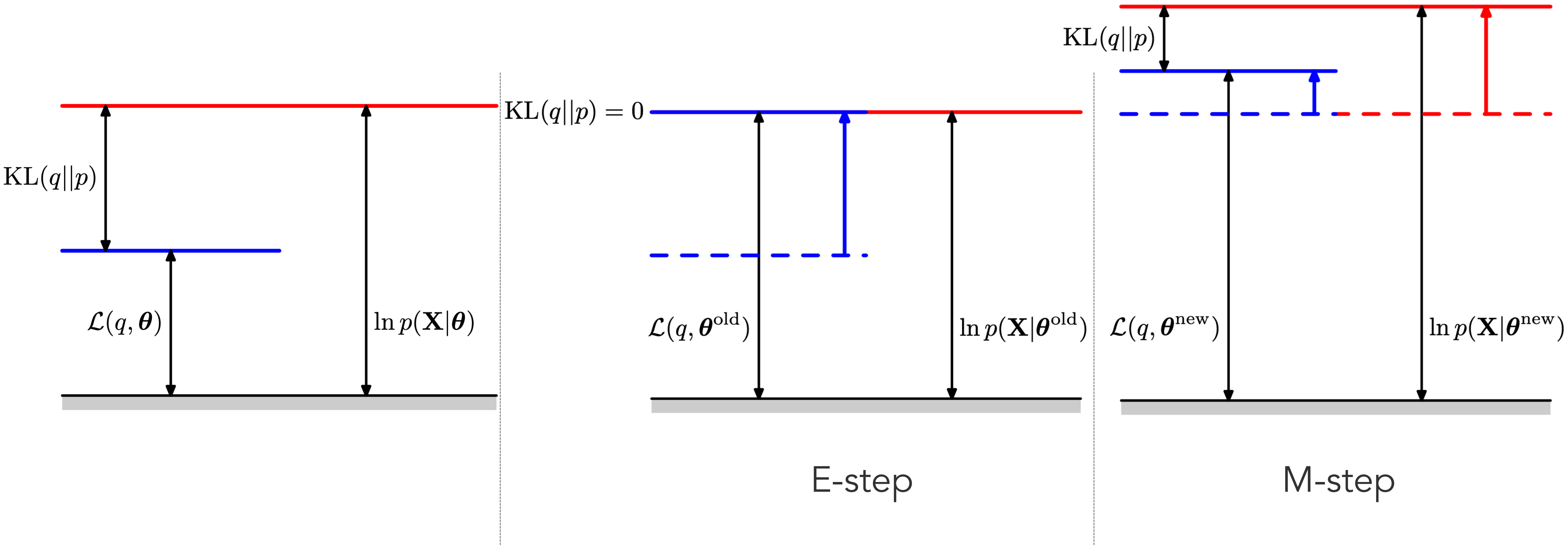
- The EM algorithm is coordinate-decent on $F(q,\theta)$
  - E-step: $q^{t+1} = \arg\min_q F\left(q, \theta^t\right) = p(\boldsymbol{z}|\boldsymbol{x}, \theta^t)$
  - M-step: $\theta^{t+1} = \arg\min_\theta F\left(q^{t+1}, \theta^t\right) = \text{argmax}_\theta \sum_z q^{t+1}(\boldsymbol{z}|\boldsymbol{x})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

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

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

Each EM iteration guarantees to improve the likelihood

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



[PRML, Chap 9.4]

EM Variants

- Sparse EM
  - Do not re-compute exactly the posterior probability on each data point under all models, because it is almost zero.
  - Instead keep an “active list” which you update every once in a while.

- Generalized (Incomplete) EM:
  - It might be hard to find the ML parameters in the M-step, even given the completed data. We can still make progress by doing an M-step that improves the likelihood a bit (e.g. gradient step).

Summary

- Supervised Learning
  - Maximum likelihood estimation (MLE)
  - Duality between MLE and Maximum Entropy Principle

- Unsupervised learning
  - Maximum likelihood estimation (MLE) with latent variables
  - EM algorithm for MLE

Variational Inference

Content adapted from CMU 10-708 Spring 2017

Inference

- Given a model, the goals of inference can include:
  - Computing the likelihood of observed data $p(\boldsymbol{x}^*)$
  - Computing the marginal distribution over a given subset of variables in the model $p(\boldsymbol{x}_A)$
  - Computing the conditional distribution over a subsets of nodes given a disjoint subset of nodes $p(\boldsymbol{x}_A|\boldsymbol{x}_B)$
  - Computing a mode of the density (for the above distributions) $\operatorname{argmax}_{\boldsymbol{x}} p(\boldsymbol{x})$
  - ....

Variational Inference

- Observed variables $\boldsymbol{x}$, latent variables $\boldsymbol{z}$
- Variational (Bayesian) inference, a.k.a. **variational Bayes**, is most often used to approximately infer the conditional distribution over the latent variables given the observations (and parameters)
  - i.e., the **posterior distribution** over the latent variables

$$p(\boldsymbol{z}|\boldsymbol{x},\theta) = \frac{p(\boldsymbol{z},\boldsymbol{x}|\theta)}{\sum_z \ p(\boldsymbol{z},\boldsymbol{x}|\theta)}$$

Motivating Example

- Why do we often need to use an approximate inference methods (such as variational Bayes) to compute the posterior distribution?

- It's because we cannot directly compute the posterior distribution for many interesting models
  - I.e. the posterior density is in an intractable form (often involving integrals) which cannot be easily analytically solved.

- As a motivating example, we will try to compute the posterior for a (Bayesian) mixture of Gaussians.

Bayesian mixture of Gaussians

- The mean μ_k is treated as a (latent) random variable

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

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

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

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

- We have
  - observed variables $x_{1:n}$
  - latent variables $\mu_{1:k}$ and $z_{1:n}$
  - parameters $\{\tau^2, \pi, \sigma^2\}$
- $p(x_{1:n}, z_{1:n}, \mu_{1:k} | \tau^2, \pi, \sigma^2) = \prod_{k=1}^{K} p(\mu_k) \prod_{i=1}^{n} p(z_i) p(x_i | z_i, \mu_{1:K})$

Bayesian mixture of Gaussians

- We can write the posterior distribution as

$$p(\mu_{1:K}, z_{1:n} | x_{1:n}) = \frac{\prod_{k=1}^{K} p(\mu_k) \prod_{i=1}^{n} p(z_i) p(x_i | z_i, \mu_{1:K})}{\int_{\mu_{1:K}} \sum_{z_{1:n}} \prod_{k=1}^{K} p(\mu_k) \prod_{i=1}^{n} p(z_i) p(x_i | z_i, \mu_{1:K})}$$

- The numerator can be computed for any choice of the latent variables
- The problem is the denominator (the marginal probability of the observations)
  - This integral cannot easily be computed analytically
- We need some approximation..

Variational Inference

The main idea behind variational inference:

- Choose a family of distributions over the latent variables $z_{1:m}$ with its own set of variational parameters ν, i.e.

$$q(z_{1:m}|\nu)$$

- Then, we find the setting of the parameters that makes our approximation q closest to the posterior distribution.
  - This is where optimization algorithms come in.

- Then we can use q with the fitted parameters in place of the posterior.
  - E.g. to form predictions about future data, or to investigate the posterior distribution over the hidden variables, find modes, etc.

Variational Inference

- We want to minimize the KL divergence between our approximation $q(\boldsymbol{z}|\boldsymbol{x})$ and our posterior $p(\boldsymbol{z}|\boldsymbol{x})$

$$\mathrm{KL}(q(\boldsymbol{z}|\boldsymbol{x})\ ||\ p(\boldsymbol{z}|\boldsymbol{x}))$$

  - But we can't actually minimize this quantity w.r.t q because $p(\boldsymbol{z}|\boldsymbol{x})$ is unknown

Evidence Lower Bound (ELBO)

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

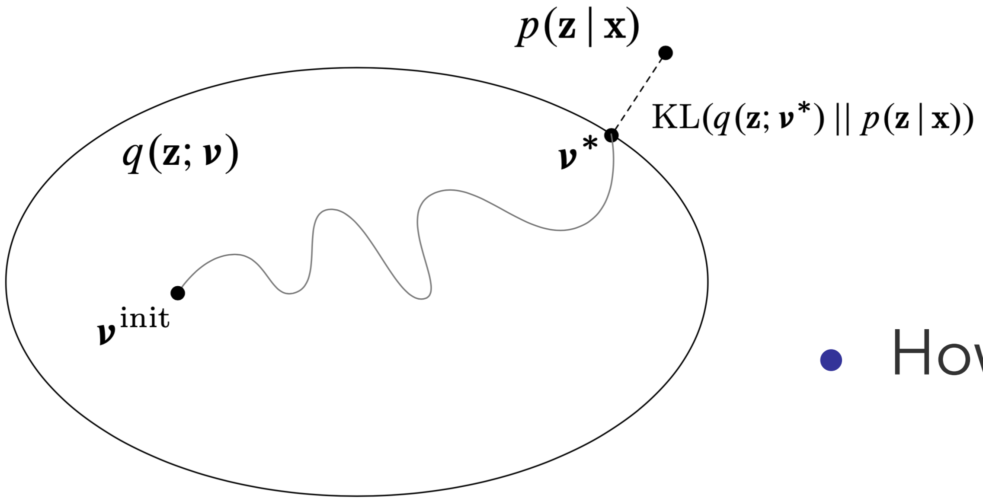
- The ELBO is equal to the negative KL divergence up to a constant $\ell(\theta;\boldsymbol{x})$
- We maximize the ELBO over q to find an "optimal approximation" to $p(\boldsymbol{z}|\boldsymbol{x})$

Variational Inference

- Choose a family of distributions over the latent variables $\boldsymbol{z}$ with its own set of variational parameters ν , i.e. $q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{v})$
- We maximize the ELBO over q to find an "optimal approximation" to $p(\boldsymbol{z}|\boldsymbol{x})$

$$\operatorname{argmax}_{\nu} \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{\nu})}\left[\log \frac{p(\boldsymbol{x}, \boldsymbol{z}|\theta)}{q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{\nu})}\right]$$

$$= \operatorname{argmax}_{\nu} \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{\nu})}[\log p(\boldsymbol{x}, \boldsymbol{z}|\theta)] - \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{\nu})}[\log q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{\nu})]$$



- 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 :

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

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

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

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 $\mathcal{D} = \{x_1, \ldots, x_N\}$:

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

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

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

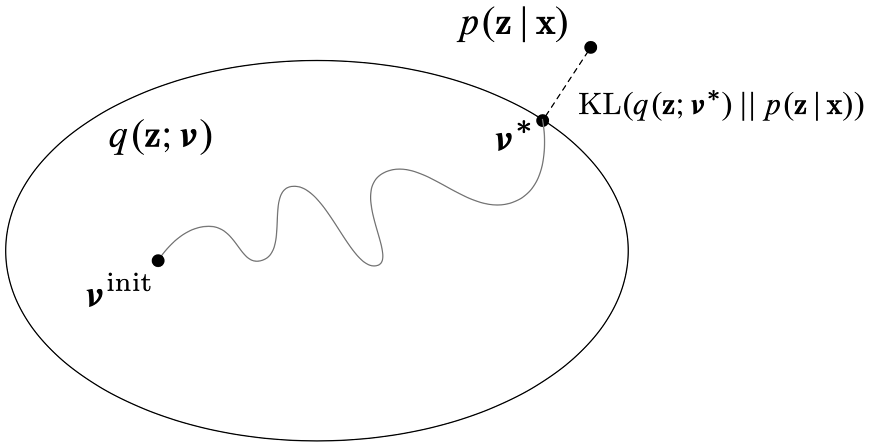
$$\begin{aligned} \ln q_\tau^\star(\tau) &= \mathbb{E}_\mu\left[\ln p(\mathcal{D}|\mu,\tau) + \ln p(\mu|\tau)\right] + \ln p(\tau) + \text{const} \\ &= (a_0-1)\ln\tau - b_0\tau + \frac{N}{2}\ln\tau \\ &\quad -\frac{\tau}{2}\mathbb{E}_\mu\left[\sum_{n=1}^{N}(x_n-\mu)^2 + \lambda_0(\mu-\mu_0)^2\right]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

  - 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)

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

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)

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

  - 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

Questions?

Questions?