10708 Homework 3

Fan Jiang
fanj

April 12, 2017

1 Variational Autoencoders

1.1 Derivations

1.1.1

$\log P(x) = \int_z q_\phi(z|x) \log p_\theta(x) dz$
$= \int_z q_\phi(z|x) \log \frac{p_\theta(x,z)}{p_\theta(z|x)}$
$= \int_z q_\phi(z|x) \log \frac{q_\phi(z|x)}{p_\theta(z|x)} \frac{p_\theta(x,z)}{q_\phi(z|x)}$
$= \int_z q_{\phi(z|x)} \log \frac{q_\phi(z|x)}{p_\theta(z|x)} + \int_z q_{\phi(z|x)} \log \frac{p_\theta(x,z)}{q_\phi(z|x)}$
$= KL(q_\phi(z|x)||p_\theta(z|x)) + \int_z q_{\phi(z|x)} \log \frac{p_\theta(x,z)}{q_\phi(z|x)}$
$\geq \int_z q_\phi(z|x) \log \frac{p_\theta(x,z)}{q_\phi(z|x)}$
$:= L(\theta, \phi; x)$

1.1.2

In the WAKE phase,
Generate examples from $q_\phi(z|x^i)$, from training data x^i.
Use the sample z and input x^i as target to update the generator network parameter θ, i.e. performs one step of gradient ascent update with respect to maximum likelihood.

The **optimization objective** for this phase is:
$\max_\theta L(\theta, \phi; x) = \max_\theta \int_z q_\phi(z|x) \log \frac{p_\theta(x,z)}{q_\phi(z|x)} = \max \int_z q_\phi(z|x) \log p_\theta(z|x) - \int_z q_\theta(z|x) \log q_\phi(z|x)$
$= \max_\theta E_{q_\phi(z|x)}[\log p_\theta(x|z)] = \max_\theta \frac{1}{N} \sum_{i=1}^N \log p_\theta(x^i|z)$

In the SLEEP phase,
Start with random hidden variables z^i drawn from prior p(z) in the top layer, then use top-down to generate each following layer. At the end, generate an unbiased sample x^i from the generative model.
Then train recognition weights using the data generated $(x^i, z^i)_{i=1}^N$.

The **optimization objective** for this phase is to maximize $F'(\theta, \phi; x) = -\log p(x) + KL(p(z|x)||q_\phi(z|x))$ w.r.t. $q_\phi(z|x)$:
$\max_\phi E_{p_\theta(z,x)}[\log q_\phi(z|x)] = \max_\phi \frac{1}{N} \sum_{i=1}^N \log q_\phi(z^i|x^i)$

Advantages:
1. This algorithm is unsupervised. It does not need any labels, all weights are updated iteratively by the samples generated in the model.
2. This algorithm does not require communicating methods that sending error information to all of the connections. Instead, each layer compare the input and the top-down reconstruction, and try to minimize the "description length".

Disadvantages:
1. At first few iterations, the data generated might be very different true data, but we still use the generated data to train the recognition weights. This is wasteful.
2. The recognition weights update is the gradient of the variational bound on the log probability. This can lead to mode-averaging.
3. We are assuming the prior is independent when generating the generative weights in the top layer, but it might not be the case because of explaining away effects.
4. This algorithm may not converge.

1.1.3

The stochastic estimate of the ELBO used as the objective:
From question 1.1.1,
$\log p(x) \geq \int_z q_\phi(z|x) \log \frac{p_\theta(x,z)}{q_\phi(z|x)} = E_{q_\phi(z|x)}[\log p_\theta(x|z)p(z)] - \int_z q_\phi(z|x) \log q_\phi(z|x)$
$= E_{q_\phi(z|x)}[\log p_\theta(x|z)] - KL(q_\phi(z|x)||p(z)) := L(\theta, \phi; x)$.

Optimize L w.r.t. $p_\theta(x|z)$ is the same with the wake phase.
$\max_\theta E_{q_\phi(z|x)}[\log p_\theta(x|z)]$

Optimize L w.r.t. $q_\phi(z|x)$ uses the reparameterization trick.
Make $q_\phi(z^i|x^i) = N(z^i; \mu^i, \sigma^{2i} I)$, then $z^{(i,l)} = \mu^i + \sigma^i \ \epsilon^l$, where $\epsilon \sim N(0, I)$.

Then the update rule for $q_\phi(z|x)$ is:
$L(\theta, \phi; x) = E_{q_\phi(z|x)}[\log p_\theta(x, z)] - KL(q_\phi(z|x)||p(z))$
$= E_{\epsilon \sim N(0,I)}[\log p_\theta(x, z_\phi(\epsilon))] - KL(q_\phi(z|x)||p(z))$

$\nabla_\phi E_{q_\phi(z|x)}[\log p_\theta(x, z)] = E_{\epsilon \sim N(0,I)}[\nabla_\phi \log p_\theta(x, z_\phi(\epsilon))]$
KL distance can be computed and differentiated analytically.

Advantages:
1. VAE reparameterization of the variational lower bound yields a simple differentiable unbiased estimator of the lower bound, which is easy to optimize using standard stochastic gradient ascent techniques.
2. The Approximate posterior inference is easier because we can use simple ancestral sampling, instead of using expensive iterative inference schemes (such as MCMC) per datapoint.

Disadvantages:
1. This model is not applicable to discrete latent variables.
2. Each element experience reconstruction error. Also this model is sensitive to irrelevant variance, for examples, translations.
3. For the simplicity of inference and learning, usually use a fixed standard normal distribution as prior.

1.1.4

By Jensen's inequality,
$logp(x) = logE_{z^i \sim q_\phi(z|x)}[\frac{1}{k}\sum_{i=1}^k \frac{p_\theta(x,z_i)}{q_\phi(z_i|x)}]$
$\geq E_{z^i \sim q_\phi(z|x)}[log\frac{1}{k}\sum_{i=1}^k \frac{p_\theta(x,z_i)}{q_\phi(z_i|x)}]$
$= L_k(x)$

We have shown that $logp(x) \geq L_k(x)$ for any given k > 0, next we have to show that $L_{k+1}(x) \geq L_k(x)$.
Let $M \subset \{1, ..., k+1\}$ and M has k elements. Then $E_{M=\{m1,...,m_k\}}[\frac{a_{m_1}+...+a_{m_k}}{k}] = \frac{a_1+...+a_{k+1}}{k+1}$. By Jensen's inequality, we got the following:
$L_{k+1}(x) = E_{z^1,...z^{k+1} \sim q_\phi(z|x)}[log\frac{1}{k+1}\sum_{i=1}^{k+1} \frac{p_{\theta(x,z_i)}}{q_\phi(z^i|x)}]$

$= E_{z^1,...z^{k+1} \sim q_\phi(z|x)}[logE_{m^1,...m^k} \frac{1}{k} \sum_{j=1}^{k} \frac{p_{\theta(x,z_{m^i})}}{q_\phi(z^{m^i}|x)}]$
$\geq E_{z^1,...z^{k+1} \sim q_\phi(z|x)}[E_{m^1,...m^k}[log \frac{1}{k} \sum_{j=1}^{k} \frac{p_{\theta(x,z_{m^i})}}{q_\phi(z^{m^i}|x)}]]$
$= E_{z^1,...z^k \sim q_\phi(z|x)}[log \frac{1}{k} \sum_{i=1}^{k} \frac{p_{\theta(x,z_i)}}{q_\phi(z^i|x)}]$
$= L_k(x)$

Combining above two parts, we can get $logp(x) \geq L_{k+1}(x) \geq L_k(x)$.

1.1.5

In order for $L_k(x) \to \log p(x)$ as $k \to \infty$, $\frac{p_\theta(x,z^i)}{q_\phi(z^i|x)}$ has to be bounded.
$L_{k \to \infty}(x) < \log p(x)$

2 Markov Chain Monte Carlo

2.1 Metropolis-Hastings

The proposal distribution I choose: $p(x'|x) = N(x' - x; 0, \sigma^2 I)$.
The acceptance can be calculated as:
$A(x'|x) = min(1, \frac{P(x')Q(x|x')}{P(x)Q(x'|x)})$
$= min(1, \frac{P(x')N(x-x';0,\sigma^2 I)}{P(x)N(x'-x;0,\sigma^2 I)})$
$= min(1, \frac{P(x')}{P(x)})$

The algorithm:
Initialize starting state x^0, set t = 0;
While samples have not converged:
$x = x^t$, t = t + 1;
sample $x^* \sim Q(x^*|x)$
sample $u \sim Uniform(0, 1)$:
if $u < A(x^*|x) = min(1, \frac{P(x')}{P(x)})$,
$x^t = x^*$;
else:
$x^t = x$.

2.2 Hamiltonian MCMC

2.2.1

$H(q, p) = \frac{1}{Z} \exp(-U(q)/T) \exp(-K(p)/T)$,
where $U(q) = -log[\pi(q)]$ and $K(p) = \sum_{i=1}^{d} \frac{p_i^2}{2}$.
where $\pi(q)$ indicates the mixture Gaussian model - $\sum_{i=1}^{m} \pi_i N(x; \mu_i, \Sigma_i)$. P are assumed to be independent Gaussians.

2.2.2

2.3 Effective sample size