Tutorial on Variational Autoencoder and its Gradient Estimators

Raymond A. Yeh

University of Illinois at Urbana-Champaign

February 21, 2019
Motivation

HOLY 💩, MAN!! LOOK AT THIS!!

"STUDY FINDS 50% OF PEOPLE BORED BY STATISTICS."

Cyanide and Happiness © Explosm.net
Suppose we are interested in modeling the distribution of

$$p_{\theta}(\mathbf{x}, \mathbf{z}) = p_{\theta}(\mathbf{x}|\mathbf{z}) p(\mathbf{z})$$  \hspace{1cm} \text{(1)}$$

where only $\mathbf{x}$ is observed and $\mathbf{z}$ is an unobserved variable.
Motivation

- Suppose we are interested in modeling the distribution of

\[ p_\theta(x, z) = p_\theta(x|z)p(z) \]  

where only \( x \) is observed and \( z \) is an unobserved variable.

- To apply maximum-likelihood,

\[ p_\theta(x) = \int p_\theta(x, z) \, dz \]  

Is this integral tractable?

- Can we approximate it? 

\[ p_\theta(x) \approx \sum_{z^{(i)}} p_\theta(x|z^{(i)}) \], 

where \( z^{(i)} \sim p(z) \).
Variational inference

- Sampling problem $\rightarrow$ optimization problem.
- Evidence Lower Bound (ELBO)

\[
\log p_\theta(x) = \int q_\phi(z|x) \log p_\theta(x) \, dz \\
= \int q_\phi(z|x) \log \left( p_\theta(x) \frac{p_\theta(z|x)q_\phi(z|x)}{p_\theta(z|x)q_\phi(z|x)} \right) \, dz \\
= \int q_\phi(z|x) \log \left( \frac{p_\theta(x,z)}{q_\phi(z|x)} \right) \, dz - \int q_\phi(z|x) \log \left( \frac{p_\theta(z|x)}{q_\phi(z|x)} \right) \, dz \\
= \mathbb{E}_{q_\phi} \left[ \log \frac{p_\theta(x,z)}{q_\phi(z|x)} \right] + \text{KL}(q_\phi \| p_\theta) \\
\geq \mathbb{E}_{q_\phi} \left[ \log \frac{p_\theta(x,z)}{q_\phi(z|x)} \right] = \mathcal{L}(\phi, \theta)
\]
Evidence lower bound (ELBO)

- When is ELBO tight? $\mathbb{E}_{q_{\phi}} \left[ \log \frac{p_{\theta}(x,z)}{q_{\phi}(z|x)} \right] + KL(q_{\phi} \parallel p_{\theta}) \geq \mathcal{L}(\phi, \theta)$
  - To get the tightest bound, find $q_{\phi}$ such that maximizes ELBO.
Evidence lower bound (ELBO)

- When is ELBO tight? \( \mathbb{E}_{q_\phi} \left[ \log \frac{p_\theta(x,z)}{q_\phi(z|x)} \right] + KL(q_\phi \| p_\theta) \geq \mathcal{L}(\phi, \theta) \)
  - To get the tightest bound, find \( q_\phi \) such that maximizes ELBO.

- Further decompose:
  \[ \mathbb{E}_{q_\phi} \left[ \log \frac{p_\theta(x,z)}{q_\phi(z|x)} \right] = \mathbb{E}_{q_\phi} [\log (p_\theta(x|z))] - KL(q_\phi(z|x) \| p(z)) \]
  - Estimate the first term using Monte Carlo samples.
  - KL can be computed analytically, if \( q \) and \( p \) are “simple”.

Side Note: EM algorithm is choosing \( q_\phi(z|x) \) as \( p_{\theta_1}(z|x) \), i.e. assumes the computation of the posterior is tractable. Need to choose a “flexible” \( q_\phi(z|x) \) that is also easy to sample from. How? Deep nets!
Evidence lower bound (ELBO)

- When is ELBO tight? \( \mathbb{E}_{q}\left[ \log \frac{p_{\theta}(x,z)}{q_{\phi}(z|x)} \right] + KL(q_{\phi}||p_{\theta}) \geq \mathcal{L}(\phi, \theta) \)
  - To get the tightest bound, find \( q_{\phi} \) such that maximizes ELBO.

- Further decompose:
  \[ \mathbb{E}_{q_{\phi}} \left[ \log \frac{p_{\theta}(x,z)}{q_{\phi}(z|x)} \right] = \mathbb{E}_{q_{\phi}} \left[ \log (p_{\theta}(x|z)) \right] - KL(q_{\phi}(z|x)||p(z)) \]
  - Estimate the first term using Monte Carlo samples.
  - KL can be computed analytically, if \( q \) and \( p \) are “simple”.

- Side Note: EM algorithm is choosing \( q_{\phi}(z|x) \) as \( p_{\theta_{t-1}}(z|x) \), i.e. assumes the computation of the posterior is tractable.
Evidence lower bound (ELBO)

- When is ELBO tight? $\mathbb{E}_{q_\phi} \left[ \log \frac{p_\theta(x,z)}{q_\phi(z|x)} \right] + KL(q_\phi \| p_\theta) \geq \mathcal{L}(\phi, \theta)$
  - To get the tightest bound, find $q_\phi$ such that maximizes ELBO.

- Further decompose:
  $\mathbb{E}_{q_\phi} \left[ \log \frac{p_\theta(x,z)}{q_\phi(z|x)} \right] = \mathbb{E}_{q_\phi} \left[ \log p_\theta(x|z) \right] - KL(q_\phi(z|x) \| p(z))$
  - Estimate the first term using Monte Carlo samples.
  - KL can be computed analytically, if $q$ and $p$ are “simple”.

- **Side Note:** EM algorithm is choosing $q_\phi(z|x)$ as $p_{\theta^{t-1}}(z|x)$, i.e. assumes the computation of the posterior is tractable.

- Need to choose a “flexible” $q_\phi(z|x)$ that is also easy to sample from. How? Deep nets!
Variational AutoEncoder (VAE)

Variational AutoEncoder models both $p_\theta(x|z)$ and $q_\phi(z|x)$ with deep networks:

- Encoder: $q_\phi(z|x) \sim \mathcal{N}(\mu_\phi(x), \sigma_\phi(x) \cdot I)$
- Decoder: $p_\theta(x|z) \sim \mathcal{N}(\mu_\theta(z), c \cdot I)$
Variational AutoEncoder (VAE)

- Variational AutoEncoder models both $p_\theta(x|z)$ and $q_\phi(z|x)$ with deep networks:
  - Encoder: $q_\phi(z|x) \sim \mathcal{N}(\mu_\phi(x), \sigma_\phi(x) \cdot I)$
  - Decoder: $p_\theta(x|z) \sim \mathcal{N}(\mu_\theta(z), c \cdot I)$

- How to learn $\phi$?:
  - Reparameterization Trick:
    $z \sim \mathcal{N}(\mu, \sigma)$ is equivalent to $\mu + \sigma \cdot \epsilon$ where $\epsilon \sim \mathcal{N}(0, 1)$.
  - Sample $z$ from $q$ is a deterministic function of $\epsilon$.
  - Use standard backpropagation for training
Variational AutoEncoder (VAE)

- Variational AutoEncoder models both $p_\theta(x|z)$ and $q_\phi(z|x)$ with deep networks:
  - Encoder: $q_\phi(z|x) \sim \mathcal{N}(\mu_\phi(x), \sigma_\phi(x) \cdot I)$
  - Decoder: $p_\theta(x|z) \sim \mathcal{N}(\mu_\theta(z), c \cdot I)$

- How to learn $\phi$?:
  - Reparameterization Trick:
    - $z \sim \mathcal{N}(\mu, \sigma)$ is equivalent to $\mu + \sigma \cdot \epsilon$ where $\epsilon \sim \mathcal{N}(0, 1)$.
    - Sample $z$ from $q$ is a deterministic function of $\epsilon$.
    - Use standard backpropagation for training

- How to learn $\theta$? Standard backpropagation.
Overall pipeline

Input Image  →  Inference  →  Generative  →  Reconstructed Image

Latent Distribution

Credit: Visualizing MNIST using a Variational Autoencoder
Applications

(a) Learned Frey Face manifold  
(b) Learned MNIST manifold

Credit: Kingma et al., 2013
Extension to the family of $q_\phi(z)$

- Variational inference with normalizing flows
Extension to the family of $q_\phi(z)$
- Variational inference with normalizing flows

Reparameterization trick with discrete latent variable
- Categorical Reparameterization with Gumbel-Softmax
Since the original VAE paper...

- Extension to the family of $q_\phi(z)$
  - Variational inference with normalizing flows
- Reparametrization trick with discrete latent variable
  - Categorical Reparameterization with Gumbel-Softmax
- Tighter Variational Bounds
  - Importance Weighted Autoencoders
Extension to the family of $q_\phi(z)$
- Variational inference with normalizing flows
Reparametrization trick with discrete latent variable
- Categorical Reparameterization with Gumbel-Softmax
Tighter Variational Bounds
- Importance Weighted Autoencoders
Lower variance gradient estimator
- Sticking the Landing: Simple, Lower-Variance Gradient Estimators for Variational Inference
Since the original VAE paper...

- Extension to the family of $q_\phi(z)$
  - Variational inference with normalizing flows
- Reparameterization trick with discrete latent variable
  - Categorical Reparameterization with Gumbel-Softmax
- Tighter Variational Bounds
  - Importance Weighted Autoencoders
- Lower variance gradient estimator
  - Sticking the Landing: Simple, Lower-Variance Gradient Estimators for Variational Inference
- Lots and lots of applications
  - Generative model with X using VAE
  - Semi-supervised learning
Since the original VAE paper...

- Extension to the family of $q_\phi(z)$
  - Variational inference with normalizing flows
- Reparametrization trick with discrete latent variable
  - Categorical Reparameterization with Gumbel-Softmax
- Tighter Variational Bounds
  - Importance Weighted Autoencoders

**Lower variance gradient estimator**
- Sticking the Landing: Simple, Lower-Variance Gradient Estimators for Variational Inference

- Lots and lots of applications
  - Generative model with X using VAE
  - Semi-supervised learning
Lower variance gradient estimator

- ELBO

\[
\mathcal{L}(\phi) = \mathbb{E}_{z \sim q_\phi(z|x)}[\log p(x|z)] - KL(q_\phi(z|x)\|p(z)) \\
= \mathbb{E}_{z \sim q_\phi(z|x)}[\log p(x|z) + \log p(z) - \log q_\phi(z|x)]
\] (3) (4)
Lower variance gradient estimator

- **ELBO**

\[
\mathcal{L}(\phi) = \mathbb{E}_{z \sim q_{\phi}(z|x)}[\log p(x|z)] - KL(q_{\phi}(z|x) \| p(z)) \\
= \mathbb{E}_{z \sim q_{\phi}(z|x)}[\log p(x|z)] + \log p(z) - \log q_{\phi}(z|x))
\]  \hspace{1cm} (3)

Gradient estimator, let \( z = t(\epsilon, \phi) \)

\[
\hat{\nabla}_{TD} = \nabla_{\phi}[\log p(x, z) - \log q_{\phi}(z|x)] \\
= \nabla_{z}[\log p(x, z) - \log q_{\phi}(z|x)]\nabla_{\phi} t(\epsilon, \phi) - \nabla_{\phi} \log q_{\phi}(z|x)
\]

For any finite samples of \( z \) the score function is not necessarily zero, even when \( q_{\phi}(z|x) = p_{\theta}(z|x) \).

Credit: Roeder et al., 2017
Remove the score function?

\[ \hat{\nabla}_{PD} = \nabla_z [\log p(x, z) - \log q_\phi(z|x)] \nabla_\phi t(\epsilon, \phi) - \nabla_\phi \log q_\phi(z|x) \]

path derivative

score function
Lower variance gradient estimator

- Remove the score function?
  \[ \hat{\nabla}_{PD} = \nabla_{z} [\log p(x, z) - \log q_{\phi}(z|x)] \nabla_{\phi} t(\epsilon, \phi) - \nabla_{\phi} \log q_{\phi}(z|x) \]  

  path derivative

  score function

- The score function has expected value of zero, thus \( \hat{\nabla}_{PD} \) is an unbiased estimator. Proof:

  \[
  \mathbb{E}_{q(z|x)} [\nabla_{\phi} \log q_{\phi}(z|x)] = \int \left( \nabla_{\phi} \log q_{\phi}(z|x) \right) q(z|x) dz \\
  = \int \left( \nabla_{\phi} q_{\phi}(z|x) \right) dz \\
  = \nabla_{\phi} \int q_{\phi}(z|x) dz = 0
  \]
Lower variance gradient estimator

Credit: Roeder et al., 2017