Variational Auto-encoders

Nan Ye

School of Mathematics and Physics
The University of Queensland

Schedule

A tentative schedule is available on BlackBoard

- Week 1-2: machine learning basics
- Week 3-4: neural network basics
- Week 5-6: deep architectures
- Week 7-8: optimization
- Week 9-10: improving generalization
- Week 10-11: unsupervised learning
- Week 12: reinforcement learning

Generative Modelling

- Discriminative modelling aims to find a model for discriminating data points.
  - In a probabilistic context, we are interested in $P(Y \mid X)$.
  - Example application: if we know the class distribution for a digit image, we can label it with the most likely class.
- Generative modelling aims to find a model that generates data points.
  - In a probabilistic context, we are interested in $P(X, Y)$ or $P(X)$.
  - Example application: if we know the probability distribution on digit images, we can sample realistic digit images from it.

Maximum likelihood estimation

- Given $x_1, \ldots, x_n$ independently drawn from $P(X)$, we often estimate $P(X)$ using a parametric model $p_\theta(x)$ by maximizing the log-likelihood

$$\max_\theta \sum_i \ln p_\theta(x_i)$$

- Recall: if X is a continuous random variable, and $p_{\mu,\sigma^2}(x)$ is a Gaussian with mean μ and variance σ^2, their MLEs are

$$\hat{\mu} = \frac{1}{n}\sum_i x_i, \qquad \hat{\sigma^2} = \frac{1}{n}\sum_i (x_i - \hat{\mu})^2.$$

- When it is difficult to sample from $P(X)$, but possible to find a good approximation $p_\theta(X)$, we can sample from $p_\theta(X)$ instead.
- In practice, X may be a high-dimensional vector,
  - designing an approximate $p_\theta(x)$ can be hard,
  - MLE can be hard for $p_\theta(x)$.
- For example, finding a good $p_\theta(X)$ for digit images.

Latent variable models

- Latent variable models allow us to incorporate unobserved variables into the generative process.
- A latent variable model assumes that each x is generated in two steps
  - Sample a latent z from a prior distribution $P(z)$
  - Sample x from $P(x \mid z)$
- The probability of observing x is

$$P(x) = \int P(x \mid z)P(z)dz$$

A special case

- Latent variable models are a rich class of generative models.
- As a useful special case, a function $f(Z)$ of a random variable Z can be seen as a latent variable model.
  - Take $P(X \mid Z) = I(X = f(Z))$.
  - Then $P(x) = \int P(x \mid z)P(z)dz = P(f(Z) = x)$.
- By choosing suitable f, we can turn a simple distribution into a complex one.
  - E.g., take $Z \sim U[0, 1]$ and $X = -\frac{1}{\lambda} \ln Z$.
  - Then X follows an exponential distribution with mean λ.

Difficulty of MLE

- Consider estimating a parametric latent variable model

$$p_\theta(x) = \int p_\theta(x \mid z) p_\theta(z) dz$$

  using a sample $x_1, \ldots, x_m$.
- For MLE, we need to be able to compute $p_\theta(x)$.
- In general, we don't have a simple formula for computing $p_\theta(x)$.

- Simple Monte Carlo estimate doesn't work either
  - While we can sample $z_1, \ldots, z_m$ from $p_\theta(z)$ and estimate $p_\theta(x)$ using $\frac{1}{m} \sum_i p_\theta(x \mid z_i)$, it often happens that the sampled z_i's are unlikely to generate x.
  - Thus we need a large number of samples to accurately estimate $p_\theta(x)$.

Variational inference

- When optimizing an intractable objective, the variational approach finds an approximation to the objective, and optimize the approximation instead.
- For latent variable models, there is a lower bound of the log-likelihood $p_\theta(x)$ that contains KL-divergence terms.

KL-divergence

- The KL-divergence between two distributions $p(X)$ and $q(X)$ is

$$KL(p||q) = \mathbb{E}_{X\sim p} \ln \frac{p(X)}{q(X)} = \sum_x p(x) \ln \frac{p(x)}{q(x)}.$$

- The KL-divergence is always non-negative, with $KL(p||q) = 0$ when $p = q$.
- $KL(p||q)$ is often considered as a distance between p and q.
  - Strictly speaking, it's not a distance measure as its not symmetric.

A variational bound

- Variational inference makes use of the following bound (evidence lower bound or ELBO)

$$\ln p_\theta(x) \geq L_{\theta,\phi}(x) = \mathbb{E}_{z\sim q_\phi}(\ln p_\theta(x \mid z)) - KL(q_\phi(z)||p_\theta(z)),$$

  where $q_\phi(z)$ is an arbitrary distribution that is easy to sample from.
- In fact, $\ln p_\theta(x) - L_{\theta,\phi}(x) = KL(q_\phi(z)||p_\theta(z \mid x))$.
- If $q_\phi(z)$ is the same as the posterior $p_\theta(z \mid x)$, then the lower bound equals to $\ln p_\theta(x)$.
- Thus $q_\phi(z)$ is often chosen to be a distribution $q_\phi(z \mid x)$ that depends on x, so that we can get good approximation for all x's.

- Instead of maximizing the log-likelihood $\ln p_\theta(x)$, we can now maximize its lower bound $L_{\theta,\phi}(x)$.
- However, the lower bound itself can be hard to compute or approximate.
- In addition, while we can sample $z_1, \ldots, z_m$ from $q_\phi(z \mid x)$ and estimate $L_{\theta,\phi}(x)$ with

$$\frac{1}{m}\sum_i \ln p_\theta(x \mid z_i) - \frac{1}{m}\sum_i \ln \frac{q_\phi(z_i \mid x)}{p_\theta(z_i)},$$

this approximation is not differentiable, and thus we cannot use SGD to optimize the bound.

The reparametrization trick

- In a lot of cases, we can reparametrize $q_\phi(z \mid x)$ as $z = g_\phi(\epsilon, x)$
  - ϵ is a random variable
  - distribution of ϵ does not depend on ϕ.
- We can estimate $L_{\theta,\phi}$ by

$$\frac{1}{m}\sum_i \ln p_\theta(x \mid z_i) - \frac{1}{m}\sum_i \ln \frac{q_\phi(z_i \mid x)}{p_\theta(z_i)},$$

  where $z_i = g_\phi(\epsilon_i, x)$ and $\epsilon_i \sim p(\epsilon)$.
- This is a differentiable estimator, and now we can use SGD to optimize $L_{\theta,\phi}$.

Variational Auto-encoders (VAEs)

- VAEs have emerged as one of the most popular approaches to unsupervised learning of complicated distributions.
- Advantages
  - weak assumptions
  - small approximation error (use high capacity models),
  - efficient training via backpropagation

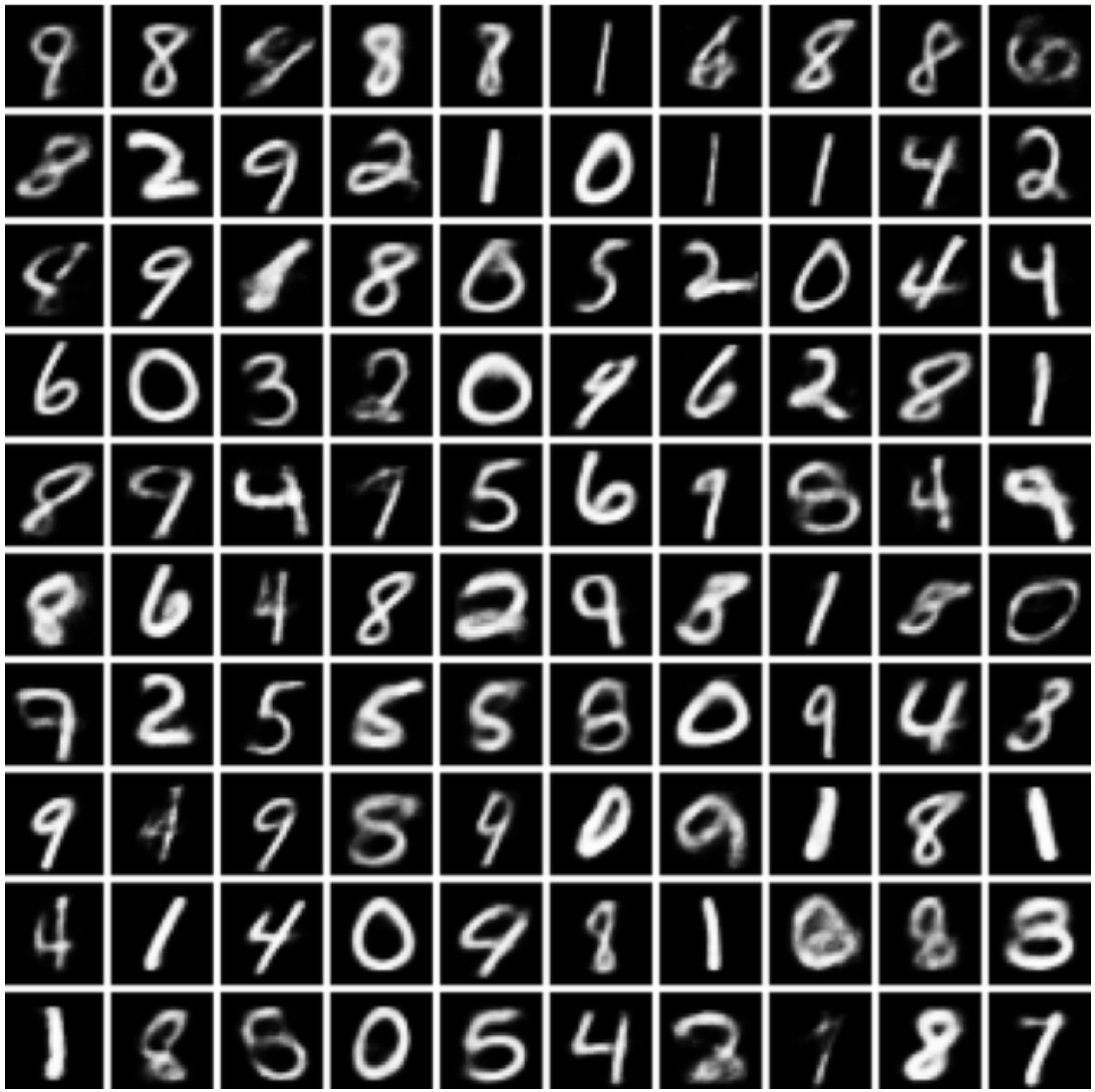
- VAEs are based on very different mathematical principles as compared to classical autoencoders.
- The *variational* part comes from all the variational inference principle that we just covered.
- The *autoencoder* part comes from the neural nets used to represent $q_\phi(z \mid x)$ and $p_\theta(x \mid z)$
  - q_ϕ encodes x to a distribution on z
  - p_θ decodes z to a distribution on x

A typical architecture

- The prior $p_\theta(z)$ is $N(0, I)$.
- The encoder network computes $\mu_\phi(x)$ and $\Sigma_\phi(x)^{1/2}$.
- The code z is computed as $\mu_\phi(x) + \Sigma_\phi^{1/2}(x)\epsilon$, where $\epsilon \sim N(0, I)$.
  - Here $q_\phi(z \mid x) = N(z; \mu_\phi(x), \Sigma_\phi(x))$.
- The decoder computes $\mu_\theta(z)$ and $\Sigma_\theta^{1/2}(z)$.
  - Here $p_\theta(x \mid z) = N(x; \mu_\theta(z), \Sigma_\theta(z))$.
- Now we can compute the differentiable estimator for ELBO and perform SGD.

Generating new samples

- We first draw z from $N(0, I)$, instead of using the encoder to generate z.
- We then run the decoder on z to draw a new data point $\mu_\theta(z)$.

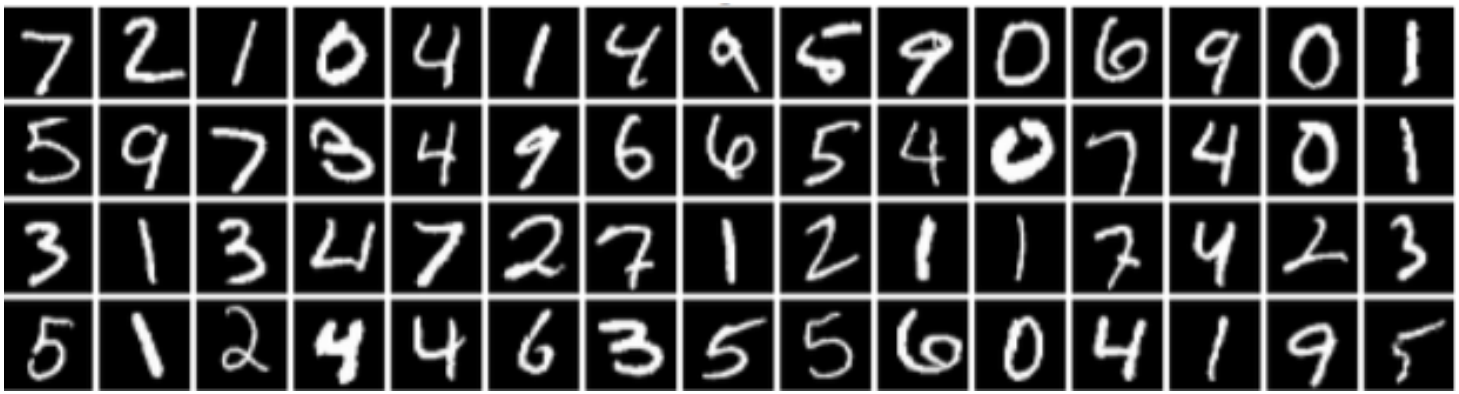


Images generated by a VAE trained on MNIST

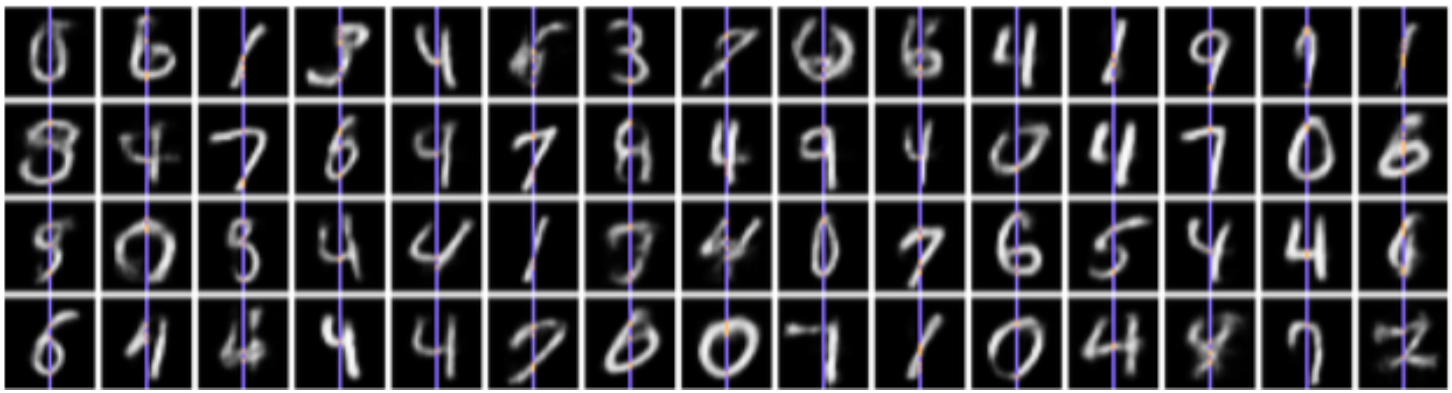
Doersch, Tutorial on variational autoencoders, 2016

Conditional VAE

- If we are given part of an image, and we want to complete the image, we can use the given part as an additional input to the encoder and decoder network in VAE.
- The resulting model is known as conditional VAE.



Ground truth



Images generated by a CVAE given by the blue part

Doersch, Tutorial on variational autoencoders, 2016

Your Turn

Which of the following statement is correct? (Multiple choice)

(a) A mixture of gaussians is a latent variable model

(b) A variational autoencoder is a latent variable model

(c) ELBO is an upper bound on the log-likelihood of a latent variable model

(d) The reparametrization trick improves computational efficiency of variational inference

What You Need to Know

- Discriminative vs generative modelling
- Latent variable models
  - hierarchical structure in the data generation process
  - only the outcomes but not the intermediate variables are observed
- Variational inference
  - deals with intractability of MLE by optimizing a lower bound
- Variational auto-encoder
  - uses neural nets to represent $q_\phi(z \mid |x)$ and $p_\theta(x \mid z)$ in variational inference