STA 4273: Minimizing Expectations

Lecture 5 - Variational Objectives I

Chris J. Maddison

University of Toronto

Announcements

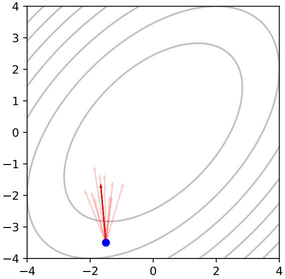
- Additonal office hours posted for next week.
- Questions, comments, concerns?

Stochastic computation graphs

- Today we will review stochastic computation graphs (SCG) framework.
  - Gradient Estimation Using Stochastic Computation Graphs (Schulman et al., 2015).
  - Credit Assignment Techniques in Stochastic Computation Graphs (Weber et al., 2019)
- Summarizes a great deal of the topics on gradient estimation in the last two weeks.

Stochastic computation graphs—basic idea

- Suppose we have a program that computes realizations of $f(X, \theta)$ with $X \sim q_\theta$.
  - X is a random variable with a prob. density q_θ.
  - $f : \mathcal{X} \times \mathbb{R}^D \to \mathbb{R}$ is a function.
- Can we automatically derive a program that computes an estimator of $\nabla_\theta \mathbb{E}_{X \sim q_\theta}[f(X, \theta)]$?

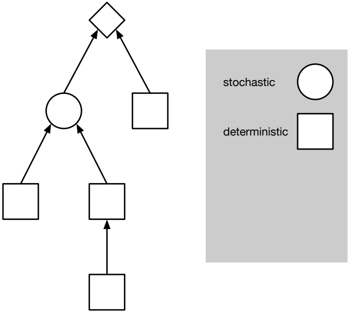


A SCG is a directed, acyclic graph $(\mathcal{V}, \mathcal{E})$.

- An edge in $\mathcal{E}$ from v to w means that w is a (random) function of v.

It has two types of nodes

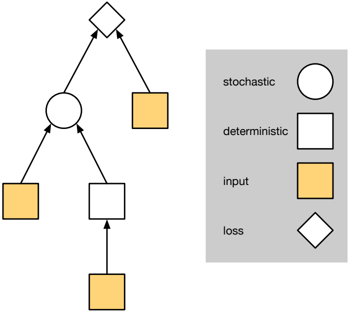
- Stochastic nodes $\mathcal{S} \subseteq \mathcal{V}$, which are conditionally independent r.v.s given their parents.
- Deterministic nodes $\mathcal{D} \subseteq \mathcal{V}$, which are deterministic functions of their parents.



Stochastic computation graphs

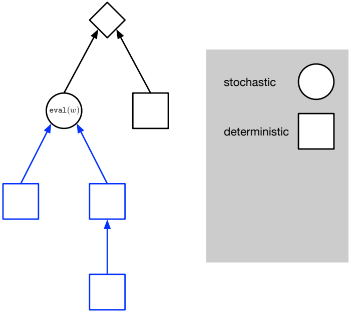
Deterministic nodes are further specialized

- Inputs are deterministic nodes that have no parents. Includes the parameters θ.
- Losses $\mathcal{L} \subseteq \mathcal{V}$ are the deterministic nodes whose average expectation we aim to minimize in θ.



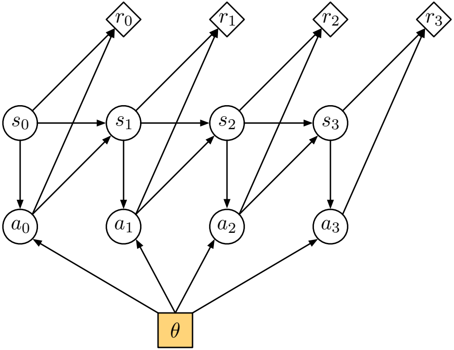
Stochastic computation graphs

- h_v are the parents of a node v.
- w descends from v, $v \prec w$, if a directed path from v to w exists.
  - Sim. $\mathcal{X} \prec w$ for $\mathcal{X} \subseteq \mathcal{V}$, if a directed path exists from some node in $\mathcal{X}$ to w.
- Can evalute a node, $\texttt{eval}(w)$.
  - Resolve the value of it's ancestors $\mathcal{A}_w = \{v : v \prec w\}$.
  - All inputs in $\mathcal{A}_w$ need to have their values given by a user or fixed.
  - Value of a stochastic node is a realization of the random variable.
- We use v synonymously with its value in a realization of the graph.



Finite-horizon MDP—example

Finite-horizon MDP with policy $\pi_\theta(a_t|s_t)$.



$\tau = (s_0, a_0 .. s_3, a_3)$, $r_t = r(s_t, a_t)$, $r(\tau) = \sum_{t=0}^{3} r_t$, $J(\theta) = \mathbb{E}[r(\tau)]$.

Notation

- If y is a function of x (may be a random function), then
  - $\partial y/\partial x$ is the direct derivative of y with respect to x.
  - dy/dx is the total derivative of y with respect to x, taking into account all paths from x to y.
  - If y is a random function of x, then $\partial y/\partial x = 0$ by convention.

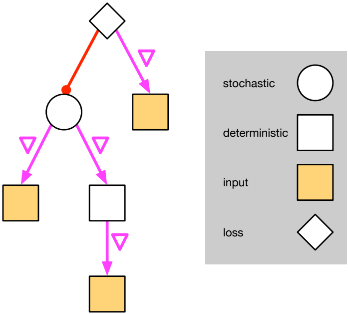
Stochastic computation graphs

- For $L := \sum_{\ell \in \mathcal{L}} \ell$ are interested in:

$$\nabla_\theta J(\theta) = \mathbb{E}[L] = \mathbb{E}\left[\sum_{\ell \in \mathcal{L}} \ell\right]$$

- Stochastic nodes block gradients.
- Then we have $\nabla_\theta J(\theta) =$

$$\mathbb{E}\left[\sum_{\substack{v \in \mathcal{S} \\ \theta \prec v}} L \frac{d \log p(v|h_v)}{d\theta} + \sum_{\substack{\ell \in \mathcal{L} \\ \theta \prec \ell}} \frac{d\ell}{d\theta}\right]$$

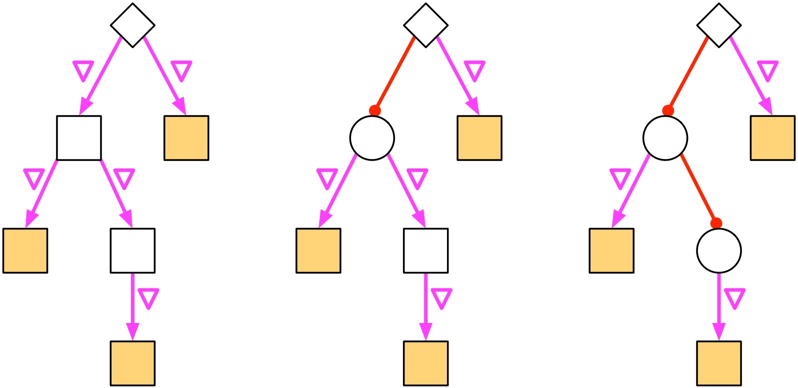


Stochastic computation graphs

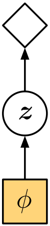
$$\mathbb{E}\left[\sum_{\substack{v \in \mathcal{S} \\ \theta \prec v}} L \frac{d \log p(v|h_v)}{d\theta} + \sum_{\substack{\ell \in \mathcal{L} \\ \theta \prec \ell}} \frac{d\ell}{d\theta}\right]$$

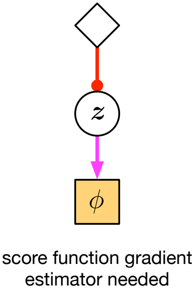
- We are ignoring smoothness assumptions needed to make this formal, but at the very least we need the differentiability of all edges
- Note, any paths from θ to v that include a stochastic node will contribute 0 to the total derivative by convention.
- Pathwise gradients usually contribute very little variance.
- Score function gradients or REINFORCE contribute the most variance.
- *Usually*. There are exceptions in which score function estimators are lower variance.

Stochastic computation graphs—examples



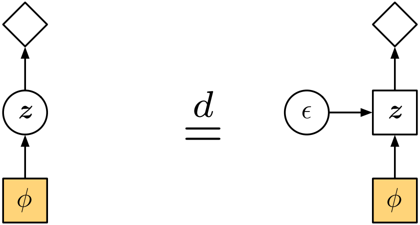
Stochastic computation graphs—examples





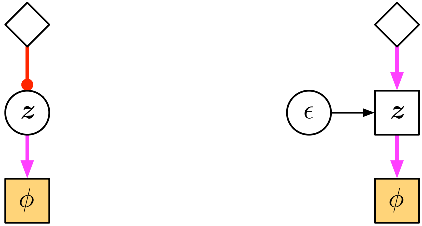
If a path from a loss to an input is blocked by a stochastic node, we must use score function estimators.

Stochastic computation graphs—examples



Suppose we can reparamterize $z = g(\epsilon, \phi)$ for some random variable ϵ and differentiable g.

Stochastic computation graphs—examples

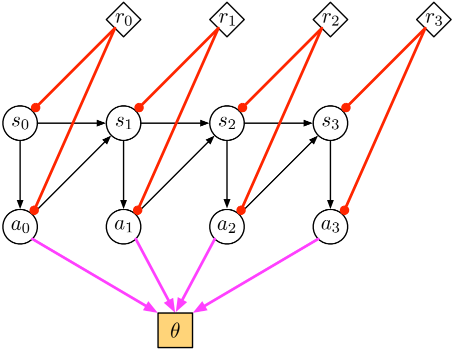


score function gradient estimator needed

pathwise gradient estimator available

Now we can use pathwise (which is typically lower variance!).

Finite-horizon MDP—example



$$\nabla J(\theta) = \mathbb{E}_{\tau \sim p}\left[\sum_{t=0}^{3} \left(\sum_{t=0}^{3} r_t\right) \nabla \log \pi_\theta(a_t|s_t)\right]$$

Stochastic computation graphs

- The most important thing is not the formal details of this framework (unless you will implement a new TensorFlow package), but that you get the intuitions.
- We will now define values, baselines, and critics on general SCGs.
- The reason is that these are powerful techniques for lowering the variance of gradient estimators and this framework can help you develop an intuition for designing new techniques.

Values

- Let $\mathcal{X} \subseteq \mathcal{V}$. Let x be an assignment of possible values to variables in $\mathcal{X}$. The value function of x for a scalar function S of the nodes is

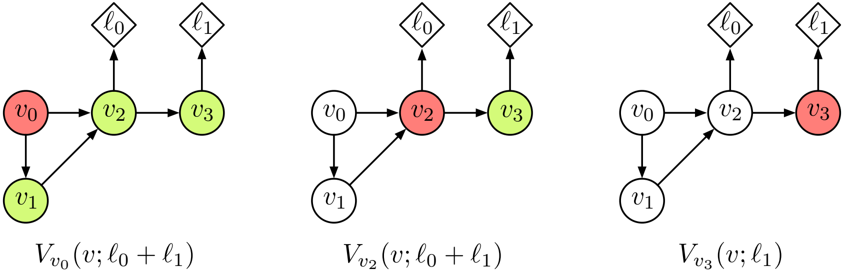
$$V_{\mathcal{X}}(x; S) = \mathbb{E}[S(\mathcal{V}) \,|\, \mathcal{X} = x]$$

- $S(\mathcal{V})$ is typically the cost-to-go of $\mathcal{X}$, i.e., the sum of loss nodes that descend from $\mathcal{X}$.

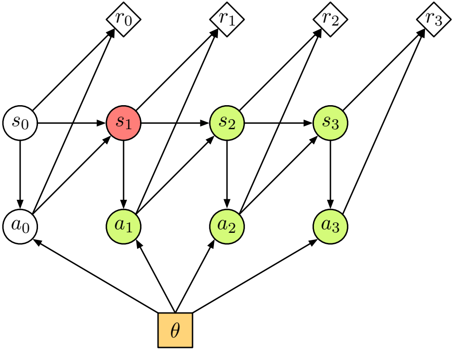
$$S(\mathcal{V}) = L(\mathcal{X}) := \sum_{\substack{\ell \in \mathcal{L} \\ \mathcal{X} \prec \ell}} \ell$$

Values

Red nodes are conditioned on; green nodes are marginalized.

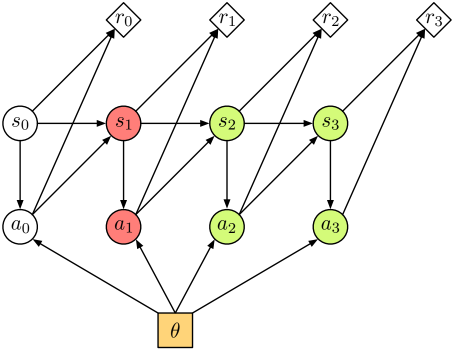


(Omitting the θ input from which all nodes descend.)



$$V_{s_1}(s; L(s_1)) = \mathbb{E}\left[\sum_{t=1}^{3} r_t \,\middle|\, s_1 = s\right] = V_1^{\pi}(s)$$

Finite-horizon MDP—example



$$V_{\{s_1,a_1\}}(s, a; L(\{s_1, a_1\})) = \mathbb{E}\left[\sum_{t=1}^{3} r_t \,\middle|\, s_1 = s, a_1 = a\right] = Q_1^{\pi}(s, a)$$

Baselines

- A baseline for a node v is a scalar-valued function $B(\mathcal{V})$ of the node values in $\mathcal{V}$ such that

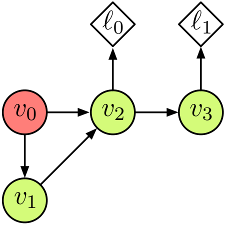
$$\mathbb{E}\left[\frac{d \log p(v|h_v)}{d\theta} B(\mathcal{V})\right] = 0$$

- Important fact: if $\mathcal{B} \subseteq \mathcal{V}$ is such that for all $b \in \mathcal{B}$, b is not a descendant of w, $w \nprec b$, and $B(\mathcal{B})$ is a scalar-valued function, then

$$\mathbb{E}\left[\frac{d \log p(v|h_v)}{d\theta} B(\mathcal{B})\right] = \mathbb{E}\left[\mathbb{E}\left[\frac{d \log p(v|h_v)}{d\theta}\middle| h_v\right] \mathbb{E}\left[B(\mathcal{B})|h_v\right]\right] = 0$$

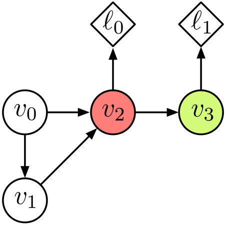
Baselines

Values can be used as baselines. Which are valid baselines for v_2?



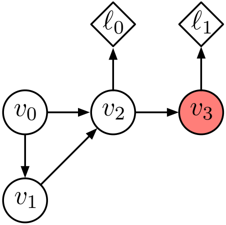
$V_{v_0}(v; \ell_0 + \ell_1)$

VALID



$V_{v_2}(v; \ell_0 + \ell_1)$

INVALID



$V_{v_3}(v; \ell_1)$

INVALID

Baselines

- Application: $L(\theta) - L(v)$ is a valid baseline for v, so we can quickly get the following identity:

$$\mathbb{E}\left[\sum_{\substack{v \in \mathcal{S} \\ \theta \prec v}} L \frac{d \log p(v|h_v)}{d\theta} + \sum_{\substack{\ell \in \mathcal{L} \\ \theta \prec \ell}} \frac{d\ell}{d\theta}\right]$$

$$= \mathbb{E}\left[\sum_{\substack{v \in \mathcal{S} \\ \theta \prec v}} L(v) \frac{d \log p(v|h_v)}{d\theta} + \sum_{\substack{\ell \in \mathcal{L} \\ \theta \prec \ell}} \frac{d\ell}{d\theta}\right]$$

Critics

- A critic for a node v is a scalar-valued function $Q(\mathcal{V})$ of the node values in $\mathcal{V}$ such that

$$\mathbb{E}\left[\frac{d \log p(v|h_v)}{d\theta} L(v)\right] = \mathbb{E}\left[\frac{d \log p(v|h_v)}{d\theta} Q(\mathcal{V})\right]$$

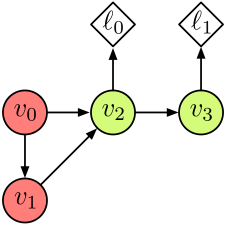
- Can be designed easily using the tower property of expectation:

$$\mathbb{E}\left[\frac{d \log p(v|h_v)}{d\theta} L(v)\right] = \mathbb{E}\left[\frac{d \log p(v|h_v)}{d\theta} \mathbb{E}[L(v)|v, h_v]\right]$$

So $Q_v(\mathcal{V}) = \mathbb{E}[L(v)|v, h_v] = V(v, h_v; L(v))$ is a valid critic.

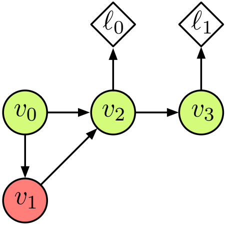
Critics

Values can be used as critics. Which are valid critics for v_1?



$V_{\{v_0,v_1\}}(u, v; \ell_0 + \ell_1)$

VALID



$V_{v_1}(v; \ell_0 + \ell_1)$

INVALID

Why? $L(v_1)$ is not conditionally independent of $d \log p(v_1|v_0)/d\theta$ given v_1.

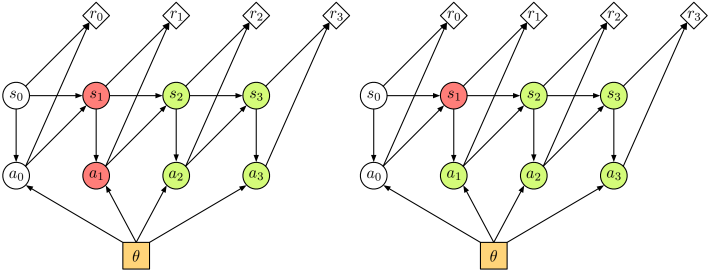
Baselines and critics

Critics and baselines are motivated by the following fact. Let Q_v and B_v be critics and baselines, respectively, for each stochastic node v, then

$$\nabla_\theta J(\theta) = \mathbb{E}\left[\sum_{\substack{v \in \mathcal{S} \\ \theta \prec v}} (Q_v(\mathcal{V}) - B_v(\mathcal{V})) \frac{d \log p(v|h_v)}{d\theta} + \sum_{\substack{\ell \in \mathcal{L} \\ \theta \prec \ell}} \frac{d\ell}{d\theta}\right]$$

Depending on the choice of Q_v and B_v we can *greatly* reduce variance, while remaining unbiased.

Finite-horizon MDP—example



$$\nabla J(\theta) = \mathbb{E}\left[\sum_{t=0}^{T}\left(\sum_{t'=t}^{T} r_{t'}\right)\frac{d\log \pi_\theta(a_t|s_t)}{d\theta}\right]$$

$$= \mathbb{E}\left[\sum_{t=0}^{T}\left(Q_t^{\pi}(s_t, a_t) - V_t^{\pi}(s_t)\right)\frac{d\log \pi_\theta(a_t|s_t)}{d\theta}\right]$$

Stochastic computation graphs

- Framework includes other generalizations.
- Weber et al. (2019) define the following.
  - Generalized Bellman equation.
  - "Bootstrapping" methods, i.e., generalizations of TD learning.
  - Some other slightly more exotic variance reduction ideas.
  - Lots to explore, some of which may not really have been widely applied. Opportunity?
- Let's look at an application: variational autoencoders.

Variational autoencoders

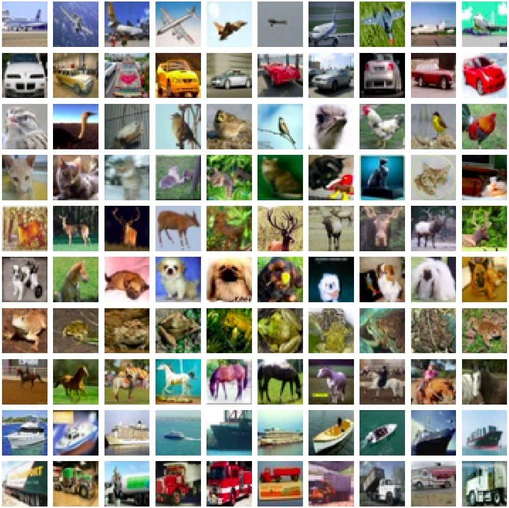
Modelling high-dimensional, multi-modal data

MNIST handwritten digit dataset.

(Vahdat and Kautz, 2020)

Modelling high-dimensional, multi-modal data

CIFAR-10 small natural image dataset.



(Krizhevsky, 2009)

Modelling high-dimensional, multi-modal data

CelebA large images of celebreties

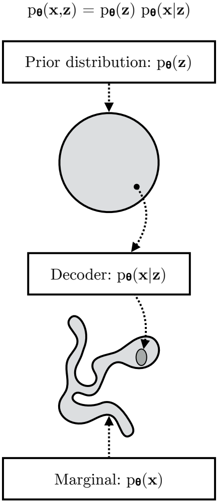
(Liu et al., 2015)

Variational autoencoders

1. Variational autoencoders (VAEs) are latent variables models for high dimensional data $\mathbf{x} \in \mathbb{R}^n$.
2. A latent variable model is specified in terms of a joint distribution between $\mathbf{x}$ and a latent variable $\mathbf{z} \in \mathbb{R}^m$ that factorizes as follows:

$$p_\theta(\mathbf{x}, \mathbf{z}) = p_\theta(\mathbf{z}) p_\theta(\mathbf{x}|\mathbf{z})$$

3. Latent variable models are an expressive class, because the marginal $p_\theta(\mathbf{x})$ can be very complex due to the likelihood $p_\theta(\mathbf{x}|\mathbf{z})$ warping the probability mass of a simple prior $p_\theta(\mathbf{z})$.



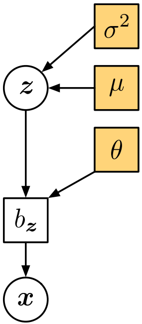
(Kingma and Welling, 2019)

Variational autoencoders—example

1. Consider the binary data case, $\mathbf{x} \in \{0,1\}^n$.
2. Consider a deep Gaussian latent variable model.

$$\mathbf{z} \sim \mathcal{N}(\mu, \sigma^2 I)$$
$$\mathbf{x}_i \sim \text{Bernoulli}(b_{\mathbf{z},i}) \text{ indept.}$$

   where $b_{\mathbf{z}} = \mathcal{NN}_\theta(\mathbf{z})$ is computed using a neural network $\mathcal{NN}_\theta : \mathbb{R}^m \to \mathbb{R}^n$ with parameters θ.
3. The marginal $p_\Theta(\mathbf{x})$ can be multimodal and expressive.

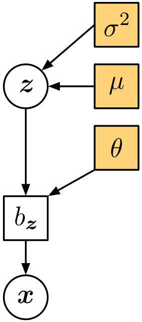


Variational autoencoders

1. Let $\Theta = (\theta, \mu, \sigma^2)$. How can we do maximum likelihood over Θ in this model?
2. What we want is

$$\arg\max_{\Theta} \log p_{\Theta}(\mathbf{x})$$

   but $p_{\Theta}(\mathbf{x}) = \int p_{\Theta}(\mathbf{x}, \mathbf{z})\, d\mathbf{z}$ is too expensive to compute.
3. The basic idea behind the variational autoencoder is to optimize a tractable variational lower bound on $\log p_{\Theta}(\mathbf{x})$, in fact the ELBO (Lecture 1)!



Evidence lower bound

Recall the evidence lower bound (ELBO)

$$\text{ELBO}(\Theta, \phi, \mathbf{x}) = \mathbb{E}_{\mathbf{z}\sim q_\phi}\left[\log \frac{p_\Theta(\mathbf{z}, \mathbf{x})}{q_\phi(\mathbf{z}|\mathbf{x})}\right] = \log p_\Theta(\mathbf{x}) - KL(q_\phi(\mathbf{z}|\mathbf{x}) \parallel p_\Theta(\mathbf{z}|\mathbf{x}))$$

Where

- q_ϕ is a density in a parametric family of probability densities.
- The objective is called the ELBO, because:

$$\text{ELBO}(\Theta, \phi, \mathbf{x}) \leq \log p_\Theta(\mathbf{x})$$

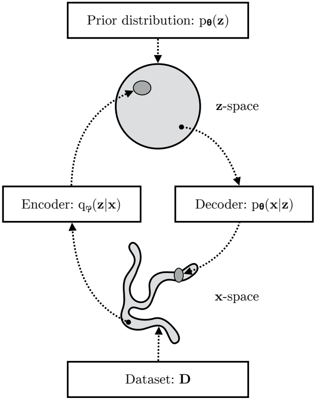
Idea: what if we optimized the ELBO in terms of Θ, ϕ?

Variational autoencoders

1. Approximate maximum likelihood for VAEs is carried out by introducing a approximate posterior $q_\phi(\mathbf{z}|\mathbf{x})$.
2. To fit a VAE, optimize ELBO using gradient ascent as a surrogate for the the marginal likelihood of $\mathbf{x}$,

$$\mathbb{E}_{\mathbf{z}\sim q_\phi}\left[\log \frac{p_\Theta(\mathbf{z},\mathbf{x})}{q_\phi(\mathbf{z}|\mathbf{x})}\right]$$

3. The key question is then how to estimate $\nabla_\Theta \text{ELBO}(\Theta,\phi,\mathbf{x})$ and $\nabla_\phi \text{ELBO}(\Theta,\phi,\mathbf{x})$



(Kingma and Welling, 2019)