Probabilistic Modelling

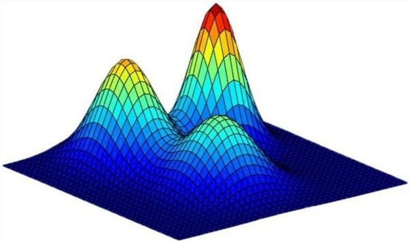
Claire Zhang, Mikail Khona

February 1, 2024

Outline

Problem Setup

Problem



- Given: $x_1, \ldots, x_N \sim p(x)$.
- Goal: Generate new samples according to $p(x)$.
- Uses:
  - Generative modelling
  - Unsupervised learning
  - Data augmentation

Variational Inference

Casual Model

View the X as the effect of a cause Z. Assume the data is generated as follows:

- $z \sim p(z)$
- $x \sim p(x|z)$

$$p_X(x) = \int p_Z(z) p_{X|Z}(x|z) dz$$

We approximate $p(z)$ and $p(x|z)$ with parametric $p_\theta(z)$ and $p_\theta(x|z)$.

Goal
Find θ such that $p_X(x) \approx p_\theta(x)$.

Variational Inference Goal

Goal
Find θ that minimizes

$$D_{KL}(p||p_\theta) = \int p(x) \log \frac{p(x)}{p_\theta(x)} dx$$

ie. maximizes the average log-likelihood of data (EV of):

$$\begin{aligned}
\log p_\theta(x) &= \log \int p_\theta(x, z) dz \\
&= \log \int \frac{p_\theta(x, z)}{q_\phi(z|x)} q_\phi(z|x) dz \\
&\geq \int q_\phi(z|x) \log \frac{p_\theta(x, z)}{q_\phi(z|x)} dz \\
&= \mathbb{E}_{q_\phi(z|x)} \left[\log \frac{p_\theta(z) p_\theta(x|z)}{q_\phi(z|x)} \right]
\end{aligned}$$

Evidence Lower Bound

$$
\begin{aligned}
\mathcal{L}(\theta, \phi; x) &:= \mathbb{E}_{q_\phi(z|x)}\left[\log \frac{p_\theta(z)p_\theta(x|z)}{q_\phi(z|x)}\right] \\
&= \mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x|z)\right] - D_{KL}(q_\phi(z|x)||p_\theta(z)) \\
&= \mathbb{E}_{p_\theta(\epsilon)}\left[\log p_\theta(x|\phi, \epsilon)\right] - D_{KL}(q_\phi(z|x)||p_\theta(z))
\end{aligned}
$$

if we **reparameterize** $z = g_\phi(x, \epsilon)$, where $\epsilon \sim p_\theta(\epsilon)$.

If g_ϕ is differentiable wrt. ϕ, $\mathcal{L}(\theta, \phi; x)$ can be maximized with gradient descent.

Sampling

We sample $\tilde{x} \sim p_\theta(x)$ using $p_\theta(z)$ and $p_\theta(x|z)$.

- $\tilde{z} \sim p_\theta(z)$
- $\tilde{x} \sim p_\theta(x|\tilde{z})$

Variational Autoencoder (VAE)

$$\begin{aligned}\mathcal{L}(\theta,\phi;x) &= \mathbb{E}_{p_\theta(\epsilon)}\left[\log p_\theta(x|\phi,\epsilon)\right] - D_{KL}(q_\phi(z|x)||p_\theta(z)) \\ &= [\text{Reconstruction Loss}] + [\text{Regularization}]\end{aligned}$$

- x is an image; z is a vector of latent variables.
- $q_\phi(z|x)$ is an inference model.
- $p_\theta(x|z)$ is a generative model.

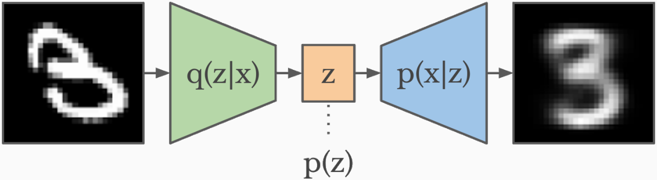


Figure 1: VAE Architecture

Score Matching

Langevin Sampling

Score matching provides another approach to generative modelling, inspired by Langevin dynamics.

Langevin Sampling

- Start with a random x_0.
- Iteratively update it according to

$$x_{t+1} = x_t + \frac{\epsilon}{2} \nabla_x \log p(x_t) + \epsilon^{1/2} \eta_t$$

where $\eta_t \sim \mathcal{N}(0, I)$.

Score Matching

Define the **score** of $p(x)$ as $\psi(x) = \nabla_x \log p(x)$. Langevin sampling doesn't require knowing $p(x)$ *exactly*, only $\psi(x)$.

Let's directly model $\psi(x)$ by $\psi_\theta(x)$, our objective being to minimize the expected squared error:

$$J(\theta) = \mathbb{E}_{p(x)} \left[\frac{1}{2} \left\| \psi_\theta(x) - \psi(x) \right\|^2 \right]$$

Under diffrentiability and regularity assumptions,

$$J(\theta) = \mathbb{E}_{p(x)} \partial \ \psi_\theta(x) + \frac{1}{2} \psi_\theta(x)^2 + C$$

which can be minimized by gradient descent.

Annealed Langevin Sampling

Langevin sampling doesn't work well on mixed or low-dim data.

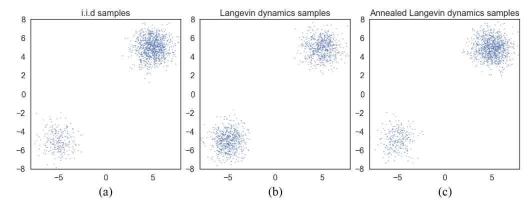


Figure 2: Sampling with true scores

Annealed Langevin Sampling (ALS)
Model long-range rates of change of $\log p$. Have $\psi(x, \sigma)$ approximate the change in $\log p(x)$ caused by adding $\mathcal{N}(0, \sigma^2)$ noise to x.

Noise Conditional Score Network

Noise Conditional Score Networks (NCSN) optimizes for ALS:

Algorithm 1 Annealed Langevin dynamics.

Require: $\{\sigma_i\}_{i=1}^{L}, \epsilon, T$.

Initialize $\tilde{\mathbf{x}}_0$
for $i \leftarrow 1$ to L **do**
    $\alpha_i \leftarrow \epsilon \cdot \sigma_i^2 / \sigma_L^2$ $\triangleright$ α_i is the step size.
    for $t \leftarrow 1$ to T **do**
        Draw $\mathbf{z}_t \sim \mathcal{N}(0, I)$
        $\tilde{\mathbf{x}}_t \leftarrow \tilde{\mathbf{x}}_{t-1} + \frac{\alpha_i}{2} \mathbf{s}_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}_{t-1}, \sigma_i) + \sqrt{\alpha_i}\, \mathbf{z}_t$
    end for
    $\tilde{\mathbf{x}}_0 \leftarrow \tilde{\mathbf{x}}_T$
end for
return $\tilde{\mathbf{x}}_T$

$$\ell(\boldsymbol{\theta}; \sigma) \triangleq \frac{1}{2} \mathbb{E}_{p_{\text{data}}(\mathbf{x})} \mathbb{E}_{\tilde{\mathbf{x}} \sim \mathcal{N}(\mathbf{x}, \sigma^2 I)} \left[\left\| \mathbf{s}_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}, \sigma) + \frac{\tilde{\mathbf{x}} - \mathbf{x}}{\sigma^2} \right\|_2^2 \right]$$

$$\mathcal{L}(\boldsymbol{\theta}; \{\sigma_i\}_{i=1}^{L}) \triangleq \frac{1}{L} \sum_{i=1}^{L} \lambda(\sigma_i) \ell(\boldsymbol{\theta}; \sigma_i)$$

Experiments

- synthetic data   mixture of gaussians
- all parametric models were NN, except z=g(x, eps) = mu + eps * sigma
- Code [github link]

PLOTS

Summary

Summary

Variational autoencoders (VAEs) and score matching are two methods for learning a generative model from data.

VAE
Model x using latent variables.

- Assume prior $z \sim p_\theta(z)$.
- Learn an inference model $q_\phi(z|x)$ and generative model $p_\theta(x|z)$.
- Sample $\tilde{z} \sim p_\theta(z)$, $\tilde{x} \sim p_\theta(x|\tilde{z})$.

Score Matching
Match first moment ψ of $\log p$.

- Learn score estimator $\psi(x)$, or $\psi(x, \sigma)$
- Sample with (annealed) langevin sampling.