Energy Based Models

John Thickstun

The idea of an energy-based model [Hinton, 1999] is, rather than explicitly learning a probabilistic model $p_\theta(x)$ over a space $\mathcal{X}$, to instead learn an energy functional $E_\theta : \mathcal{X} \to \mathbb{R}$. This energy functional can be used to implicitly define a probability distribution, for example a Gibbs distribution

$$p_\theta(x) = \frac{1}{Z_\theta} e^{-E_\theta(x)}, \text{ where } Z_\theta = \int_{\mathcal{X}} e^{-E_\theta(y)}\, dy. \tag{1}$$

The point is that, while it is easy to construct a function $E_\theta(x)$, it can be quite challenging to enforce the constraint $\int_{\mathcal{X}} p_\theta(x) = 1$, or to compute the partition function Z_θ for a given energy function $E_\theta(x)$.

To use an energy-based model as a generative model, we need to solve two problems. First, we need a training procedure for optimizing the parameters of the energy function E_θ so that the implicit distribution $p_\theta(x)$ approximates the data generating distribution $p(x)$. And second, we need a sampling procedure for drawing samples $x \sim p_\theta$. Solutions to both problems should avoid calculation of the intractable integral Z_θ. For early approaches to this problem based on the contrastive divergence, see Hinton [2002] and Hinton et al. [2006]. For a modern, empirical realization of these ideas see Du and Mordatch [2019].

Langevin Dynamics

Setting aside for the moment the question of training an energy function, suppose we have a model E_θ and we want to sample from the implied distribution $x \sim p_\theta$. While directly sampling from p_θ is difficult, we can approximate samples using a Markov chain with stationary distribution p_θ. A convenient construction on $\mathcal{X} = \mathbb{R}^d$ is Langevin dynamics; this is a continuous Markov process with dynamics given by the stochastic differential equation

$$\frac{\partial x_t}{\partial t} = \nabla_x \log p_\theta(x_t)\, dt + \sqrt{2}\, dW_t, \tag{2}$$

where dW_t is a white noise process, given by the derivative of standard Brownian motion W_t. The Fokker-Planck equation shows that diffusion following these dynamics converges asymptotically to samples $x_t \sim p_\theta$, in the sense that $D(x_t \parallel p_\theta) \to 0$ as $t \to \infty$.

For implementation, we cannot exactly construct a diffusion x_t following the dynamics of Equation (3). In practice, we will discretize the diffusion and follow a discrete Markov chain driven by i.i.d. Gaussian noise $\varepsilon_t \sim \mathcal{N}(0, I)$:

$$x_{t+1} = x_t - \eta \nabla_x \log p_\theta(x_t) + \sqrt{2\eta}\varepsilon_t. \tag{3}$$

This can be viewed as the stochastic analog to an Euler discretization of a deterministic differential equation. As $\eta \to 0$, the approximation to the continuous dynamics of Equation (3) becomes more precise, but mixing will become more slow; an effective accelerated mixing algorithm based on simulated annealing [Neal, 2001] is presented in Song and Ermon [2019].

Score Matching

We can apply Langevin Dynamics to sample from an energy based model, because

$$\nabla_x \log p_\theta(x) = -\nabla_x E_\theta(x) - \nabla_x \log Z_\theta = -\nabla_x E_\theta(x). \tag{4}$$

In fact, we can be even more direct and simply model the gradient field of the log-density, also known as the score function $s : \mathbb{R}^d \to \mathbb{R}^d$ defined by $x \mapsto \nabla_x \log p(x)$. Want to estimate this score function using a neural parameterization $s_\theta : \mathbb{R}^d \to \mathbb{R}^d$, which implicitly defines an energy function $E_\theta : \mathbb{R}^d \to \mathbb{R}$ (by integration) and a density p_θ (by choosing the appropriate normalization Z_θ). We will now focus on learning this score function $s_\theta : \mathbb{R}^d \to \mathbb{R}^d$ that minimizes the Fisher divergence

$$\mathop{\mathbb{E}}_{x\sim p}\left[\frac{1}{2}\|s_\theta(x) - \nabla_x \log p(x)\|_2^2\right]. \tag{5}$$

The Fisher divergence provides us with another measure of the distance between two probability distributions, analogous to KL divergence:

$$D_{\text{Fisher}}(p \parallel q) \equiv \mathop{\mathbb{E}}_{x\sim p}\left[\frac{1}{2}\left\|\nabla_x \log \frac{p(x)}{q(x)}\right\|^2\right]. \tag{6}$$

A precise connection between Fisher divergence and the rate of change in KL-divergence over smoothed versions of p sand q. Define $\tilde{x}_t = x + \sqrt{t}\varepsilon_x$ and $\tilde{y}_t = y + \sqrt{t}\varepsilon_y$, where $x \sim p$, $y \sim q$, and $\varepsilon_x, \varepsilon_y \sim \mathcal{N}(0, I)$ (independent samples). Let $p_t(\tilde{x}_t)$ and $q_t(\tilde{y}_t)$ denote the densities of $\tilde{x}_t$ and $\tilde{y}_t$ respectively. Adding Gaussian noise to x, y corresponds to smoothing of their probability densities (Gaussian convolution).

Proposition 1. *[Lyu, 2012] Under mild regularity conditions,*

$$\frac{d}{dt} D(p_t \parallel q_t) = -D_{\text{Fisher}}(p_t \parallel q_t). \tag{7}$$

Because Fisher divergence is non-negative, integrating we see that $D(p_t \parallel q_t) \to 0$ as $t \to \infty$, and this convergence is monotonic.

Implicit Score Matching

We can't compute the score matching objective, because it required evaluation of (gradients of) the unknown density $p(x)$. But it turns out that we can minimize it implicitly.

Proposition 2. *(Implicit Score Matching) [Hyvärinen, 2005]*

$$\arg\min_\theta \mathop{\mathbb{E}}_{x\sim p}\left[\frac{1}{2}\|s_\theta(x) - \nabla_x \log p(x)\|_2^2\right] = \arg\min_\theta \mathop{\mathbb{E}}_{x\sim p}\left[\operatorname{tr}\left(\nabla_x s_\theta(x)\right) + \frac{1}{2}\|s_\theta(x)\|_2^2\right]. \tag{8}$$

Proof. Expanding the quadratic and dropping the constant term, we have

$$\arg\min_\theta \mathop{\mathbb{E}}_{x\sim p}\left[\frac{1}{2}\|s_\theta(x) - \nabla_x \log p(x)\|_2^2\right] = \arg\min_\theta \mathop{\mathbb{E}}_{x\sim p}\left[\frac{1}{2}\|s_\theta(x)\|^2 - s_\theta(x)^T \nabla_x \log p(x)\right]. \tag{9}$$

So we just need to show that the inner product term is equivalent to $\mathrm{tr}(\nabla_x s_\theta(x))$. Applying integration by parts, we find that

$$\begin{aligned}
\mathop{\mathbb{E}}_{x\sim p}\left[s_\theta(x)^T \nabla_x \log p(x)\right] &= \sum_{i=1}^{d} \int_{\mathcal{X}} s_\theta(x)_i \frac{\partial \log p(x)}{\partial x_i} p(x)\, dx \\
&= \sum_{i=1}^{d} \int_{\mathcal{X}} s_\theta(x)_i \frac{\partial p(x)}{\partial x_i}\, dx \\
&= -\sum_{i=1}^{d} \int_{\mathcal{X}} \frac{s_\theta(x)_i}{\partial x_i} p(x)\, dx \\
&= -\int_{\mathcal{X}} \mathrm{tr}\left(\nabla_x s_\theta(x)\right) p(x)\, dx = -\mathop{\mathbb{E}}_{x\sim p}\left[\mathrm{tr}\left(\nabla_x s_\theta(x)\right)\right].
\end{aligned}$$

□

Sliced Score Matching

The right-hand side of Equation (8) is interesting because it can be approximated by monte carlo, and evaluation of the objective only involves our model s_θ. But this is not yet a convenient objective for modeling, because the quantity $\mathrm{tr}(\nabla_x s_\theta(x))$ is a second-order statistic; it is the trace of the Hessian of the log-likelihood $\log p_\theta(x)$. Evaluating this quantity scales like $O(d)$ in the dimensionality of $x \in \mathbb{R}^d$. We can create a tractable objective [**?**] by minimizing Equation (5) along random projections $v \sim r$, e.g. from a Gaussian $r = \mathcal{N}(0, I)$:

$$L(\theta, v) \equiv \mathop{\mathbb{E}}_{x\sim p}\left[\frac{1}{2}\left(v^T s_\theta(x) - v^T \nabla_x \log p(x)\right)^2\right]. \tag{10}$$

We can replace this projected loss with an equivalent quantity that can be estimated from samples (Proposition 2):

$$\begin{aligned}
\arg\min_\theta \mathop{\mathbb{E}}_{v\sim r} L(\theta, v) &= \arg\min_\theta \mathop{\mathbb{E}}_{v\sim r} v^T \mathop{\mathbb{E}}_{x\sim p}\left[\frac{1}{2}\|s_\theta(x) - \nabla_x \log p(x)\|^2\right] v & (11)\\
&= \arg\min_\theta \mathop{\mathbb{E}}_{v\sim r} v^T \mathop{\mathbb{E}}_{x\sim p}\left[\mathrm{tr}\left(\nabla_x s_\theta(x)\right) + \frac{1}{2}\|s_\theta(x)\|_2^2\right] v & (12)\\
&= \arg\min_\theta \mathop{\mathbb{E}}_{\substack{v\sim r\\ x\sim p}}\left[v^T \nabla_x s_\theta(x) v + \frac{1}{2}\left(v^T s_\theta(x)\right)^2\right]. & (13)
\end{aligned}$$

Crucially, this objective involves only Hessian-vector products, which can be computed in time complexity independent of the data dimension. The following proposition shows that, so long as our random projections $v \sim r$ span the space $\mathbb{R}^d$, we can recover the data generating distribution $p(x)$ by minimizing the expected loss $L(\theta, v)$.

Proposition 3. *[Song, Garg, Shi, and Ermon, 2019] Suppose $p(x) = p_{\theta^*}(x)$ for some value of the parameters θ^* (the data-generating distribution is realizable). If r is positive definite, i.e. $\mathbb{E}_{v\sim r}[vv^T] \succ 0$, then*

$$\mathop{\mathbb{E}}_{v\sim r} L(\theta, v) = 0 \textit{ if and only if } \theta = \theta^*. \tag{14}$$

Proof. Suppose $\mathbb{E}_{v\sim r} L(\theta, v) = 0$ (the converse is clearly true). Note that $L(\theta, v) \geq 0$ and therefore for any x,

$$0 = \mathop{\mathbb{E}}_{v\sim r}\left[\frac{1}{2}\left(v^T s_\theta(x) - v^T \nabla_x \log p(x)\right)^2\right] \tag{15}$$

$$= \mathop{\mathbb{E}}_{v\sim r}\left[\frac{1}{2} v^T \left(s_\theta(x) - \nabla_x \log p(x)\right)\left(s_\theta(x) - \nabla_x \log p(x)\right)^T v\right] \tag{16}$$

$$= \frac{1}{2}\left(s_\theta(x) - \nabla_x \log p(x)\right)^T \mathop{\mathbb{E}}_{v\sim r}\left[vv^T\right]\left(s_\theta(x) - \nabla_x \log p(x)\right). \tag{17}$$

Because $\mathbb{E}_{v\sim r}[vv^T] \succ 0$, we deduce that $s_\theta(x) - \nabla_x \log p(x) = 0$. □

References

Yilun Du and Igor Mordatch. Implicit generation and modeling with energy based models. In *Advances in Neural Information Processing Systems*, pages 3608–3618, 2019. (document)

Geoffrey Hinton, Simon Osindero, Max Welling, and Yee-Whye Teh. Unsupervised discovery of nonlinear structure using contrastive backpropagation. *Cognitive science*, 2006. (document)

Geoffrey E Hinton. Products of experts. 1999. (document)

Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 2002. (document)

Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 2005. 2

Siwei Lyu. Interpretation and generalization of score matching. In *Uncertainty in Artificial Intelligence*, 2012. 1

Radford M Neal. Annealed importance sampling. *Statistics and computing*, 2001. (document)

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *Advances in Neural Information Processing Systems*, 2019. (document)

Yang Song, Sahaj Garg, Jiaxin Shi, and Stefano Ermon. Sliced score matching: A scalable approach to density and score estimation. In *Uncertainty in Artificial Intelligence*, 2019. 3