Notes on Unnormalized Probability Models

Zijing Ou
School of Computer Science and Engineering
Sun Yat-sen University

In this script, we motivate the energy based models (EBM) by interpreting it as a maximum entropy distribution, and then provides several methods on EBM learning.

1 Maximum Entropy Distribution

We begin by considering linear (mean-value) constraints on our distribution. In this case, we are given a function $f : \mathcal{X} \to \mathbb{R}$ and a scale $\alpha \in \mathbb{R}$, we wish to solve

$$\text{maximize } H(p) \quad \text{subject to } \mathbb{E}_p[f(x)] = \alpha \tag{1}$$

over distribution density $p(x), x \in \mathcal{X}$. Rewriting problem (1), we see that it is equivalent to

$$\begin{aligned} &\text{maximize } -\int p(x) \log p(x) \mathrm{d}x \\ &\text{subject to } \int p(x) f(x) \mathrm{d}x = \alpha, \quad p(x) \geq 0 \text{ for } x \in \mathcal{X}, \quad \int p(x) \mathrm{d}x = 1. \end{aligned} \tag{2}$$

Let

$$P_\alpha^{\text{lin}} := \{p(x) : \mathbb{E}_p[f(x)] = \alpha\} \tag{3}$$

be a set of distribution satisfying the expectation (linear) constraint $\mathbb{E}[f(x)] = \alpha$. We obtain the following theorem.

Theorem 1.1. *Let p_θ have density*

$$p_\theta(x) = \frac{\exp(-f(x)/T)}{Z}, \quad Z = \int \exp(-f(x)/T) \mathrm{d}x. \tag{4}$$

If $\mathbb{E}_{p_\theta}[f(x)] = \alpha$, then p_θ maximizes $H(p)$ over P_α^{lin}; moreover, the distribution p_θ is unique.

Proof. First, we write a Lagrangian for the problem (1). Introducing Lagrange multipliers $\lambda(x) \geq 0$ for the constraint $p(x) \geq 0$, $\eta_0 \in \mathbb{R}$ for the normalization constraint that $\int p(x) \mathrm{d}x = 1$, and η_1 for the constraints that $\mathbb{E}_p[f(x)] = \alpha$, we obtain the following Lagrangian:

$$\begin{aligned} \mathcal{L}(p, \eta_0, \theta_1, \lambda) = &\int p(x) \log p(x) \mathrm{d}x + \eta_1 \left(\int p(x) f(x) \mathrm{d}x - \alpha \right) \\ &+ \eta_0 \left(\int p(x) \mathrm{d}x - 1 \right) - \int \lambda(x) p(x) \mathrm{d}x. \end{aligned}$$

Now, heuristically treating the density $p = [p(x)]_{x \in \mathcal{X}}$ as a finite-dimentional vector (in the case that $\mathcal{X}$ is finite, this is completely rigorous), we take derivatives and obtain

$$\begin{aligned} \frac{\partial}{\partial p(x)} \mathcal{L}(p, \eta_0, \eta_1, \lambda) &= 1 + \log p(x) + \eta_1 f(x) + \eta_0 - \lambda(x) \\ &= 1 + \log p(x) + \eta_1 f(x) + \eta_0 - \lambda(x). \end{aligned}$$

Notes.

To find the minimizing p for the Lagrangian (the function is convex in p), we set this equal to zero to find that

$$p(x) = \exp(-\eta_1 f(x) - 1 - \eta_0 + \lambda(x)).$$

Now, we note that with this setting, we always have $p(x) > 0$, so that the constraint $p(x) > 0$ is unnecessary and (by complementary slackness) we have $\lambda(x) = 0$. In particular, by taking $\eta_0 = -1 + \log Z = -1 + \log \int \exp(\eta_1 f(x))\mathrm{d}x$, we have that that optimal density p should have the form

$$p_\theta(x) = \frac{\exp(-\eta_1 f(x))}{Z}. \tag{5}$$

Note that the maximum value of the entropy is

$$\begin{aligned} H_{\text{max}} &= -\int p_\theta(x) \log p_\theta(x)\mathrm{d}x = \log Z + \eta_1 \int p_\theta(x) f(x)\mathrm{d}x \\ &= 1 + \eta_0 + \eta_1 \alpha. \end{aligned} \tag{6}$$

So one can get,

$$\eta_1 = \frac{\partial H_{\text{max}}}{\partial \alpha} =: \frac{1}{T}, \tag{7}$$

which defines the inverse temperature. Now we see the form of distribution we would like to have

$$p_\theta(x) = \frac{\exp(-f(x)/T)}{\int \exp(-f(x)/T)\mathrm{d}x}. \tag{8}$$

Next, we show that the distribution p_θ is unique. Assume there exists any other distribution $p \in P_\alpha^{\text{lin}}$, such that $p = \arg\max_p H(p)$. In this case, we may expand the entropy $H(p)$ as

$$\begin{aligned} H(p) &= -\int p \log p \mathrm{d}x = -\int p \log \frac{p}{p_\theta} - \int p \log p_\theta \mathrm{d}x \\ &= -KL(p||p_\theta) - \int p(x)[-f(x)/T - \log Z]\mathrm{d}x \\ &\overset{(*)}{=} -KL(p||p_\theta) - \int p_\theta(x)[-f(x)/T - \log Z]\mathrm{d}x \\ &= -KL(p||p_\theta) + H(p_\theta), \end{aligned}$$

where in the step $(*)$ we have used the fact that $\int p(x)f(x)\mathrm{d}x = \int p_\theta(x)f(x)\mathrm{d}x = \alpha$. As $KL(p||p_\theta) \geq 0$ unless $p = p_\theta$, we have shown that p_θ is the unique distribution maximizing the entropy, as desired. □

Note that T is generally set to be 1, which leads to the common form of energy base model

$$p(x) = \frac{\exp(-f(x))}{Z}, \quad Z = \int \exp(-f(x))\mathrm{d}x. \tag{9}$$

2 Contrastive Divergence

2.1 CD on Probability Fitting

Given a data distribution $p_d(x)$, of which we solely could sample its empirical distribution. Our target is using a function $f_\theta(x)$ with parameters θ to fit the probability of data. Specifically, we define the following energy based model

$$p_\theta(x) = \frac{\exp(-f_\theta(x))}{Z_\theta}, \tag{10}$$

where Z_θ, known as the partition function and indicated its dependency of parameters by the subscript θ, is defined as

$$Z_\theta = \int \exp(-f_\theta(x))dx. \tag{11}$$

Generally, Z_θ is intractable, especially in high dimensional scenarios. To learn the model parameters θ, one could maximize the probability of a set of training data $X = \{x_1, \ldots, x_N\}$, given as

$$p_\theta(X) = \prod_{i=1}^{N} p_\theta(x_i) = \prod_{i=1}^{N} \frac{\exp(-f_\theta(x_i))}{Z_\theta}. \tag{12}$$

Equivalently, we can minimize the negative log likelihood of $p_\theta(X)$, which is

$$\mathcal{L}(\theta) := \log Z_\theta + \frac{1}{N} \sum_{i=1}^{N} f_\theta(x_i). \tag{13}$$

The gradient ascent algorithm can be applied to optimize parameters, in which we have to compute the gradient of $\mathcal{L}(\theta)$

$$\begin{aligned}
\nabla_\theta \mathcal{L}(\theta) &= \nabla_\theta \log Z_\theta + \nabla_\theta \frac{1}{N} \sum_{i=1}^{N} f_\theta(x_i) \\
&= \frac{1}{Z_\theta} \nabla_\theta \int \exp(-f_\theta(x)) dx + \frac{1}{N} \sum_{i=1}^{N} \nabla_\theta f_\theta(x_i) \\
&= \int \frac{\exp(-f_\theta(x))}{Z_\theta} \nabla_\theta (-f_\theta(x)) dx + \mathbb{E}_{p_d(x)}[\nabla_\theta f_\theta(x)] \\
&= \mathbb{E}_{p_d(x)}[\nabla_\theta f_\theta(x)] - \mathbb{E}_{p_\theta(x)}[\nabla_\theta f_\theta(x)].
\end{aligned} \tag{14}$$

Though we can exploit Monte Carlo to estimate $\nabla_\theta \mathcal{L}(\theta)$, the hardnesss arises from sampling from $p_\theta(x)$, since we cannot obtain its close form due to the notorious partition function. As introduced in [1], we can sidestep this issue by using MCMC sampling technique [2]. Specifically, given an initial sample $x^{(0)} \sim p_d(x)$, we can apply k-step MCMC iteration to generate $x^{(k)}$ for $p_\theta(x)$, which has been turn out that $\lim_{k \to \infty} x^{(k)} \sim p_\theta(x)$. Consequently, the (14) can be approximated by CD-k estimator

$$\nabla_\theta \mathcal{L}(\theta) \approx \nabla_\theta \log f_\theta(x^{(0)}) - \nabla_\theta \log f_\theta(x^{(k)}). \tag{15}$$

2.2 Examples for Latent Variable Models

Energy-based latent variable model is a popular nowadays thanks to its expressive modeling ability, whose general form can be expressed by in terms of observation data x and latent variables z, with the density function

$$p_\theta(x, z) = \frac{e^{-E_\theta(x,z)}}{Z_\theta}, \tag{16}$$

where $Z_\theta = \int e^{-E_\theta(x,z)} dx dz$ is the normalized term. In terms of maximum likelihood estimation, we have to compute the gradients of $\log p_\theta(x)$ with respect to θ

$$\begin{aligned}
\nabla_\theta \log p_\theta(x) &= \nabla_\theta \left(\log \left[\int e^{-E_\theta(x,z)} dz \right] - \log Z_\theta \right) \\
&= -\frac{\int e^{-E_\theta(x,z)} \nabla_\theta E_\theta(x, z) dz}{\int e^{-E_\theta(x,z)} dz} - \frac{\nabla_\theta Z_\theta}{Z_\theta} \\
&= -\frac{1/Z_\theta \int e^{-E_\theta(x,z)} \nabla_\theta E_\theta(x, z) dz}{1/Z_\theta \int e^{-E_\theta(x,z)} dz} - \frac{\nabla_\theta \int e^{-E_\theta(x,z)} dx dz}{Z_\theta} \\
&= -\frac{\int p_\theta(x, z) \nabla_\theta E_\theta(x, z) dz}{p_\theta(x)} + \frac{\int e^{-E_\theta(x,z)} \nabla_\theta E_\theta(x, z) dx dz}{Z_\theta} \\
&= -\int p_\theta(z|x) \nabla_\theta E_\theta(x, z) dz + \int p_\theta(x, z) \nabla_\theta E_\theta(x, z) dx dz \\
&= -\mathbb{E}_{p_\theta(z|x)}[\nabla_\theta E_\theta(x, z)] + \mathbb{E}_{p_\theta(x,z)}[\nabla_\theta E_\theta(x, z)].
\end{aligned} \tag{17}$$

In many cases, such as the RBM model, the first term has a closed form. While the second term is more difficult to deal with since we have to draw samples from $p_\theta(x, z)$, which is usually intractable. The Contrastive Divergence algorithm [1] addresses this issue by a finite-step MCMC to generates approximated samples from $p_\theta(x, z)$. However, this approximation is often insufficient and introduces additional bias.

2.3 Unbiased Contrastive Divergence Algorithm

Recently, [3] proposed a new framework to remove bias of CD. The key idea of unbiased CD algorithm is that we can compute expectations of random variables after finite many steps of Markov Chain by introducing another Markov chain, which is strongly related to the theory of unbiased MCMC developed by [4].

In particular, we want to compute $\mathbb{E}_{\mathcal{M}}[f(x)]$, where in the expression of (17) $\mathcal{M}$ denotes $p_\theta(x, z)$ and $f(x)$ denotes $\nabla_\theta E_\theta(x, z)$. If there exists two Markov chains $\{a_t\}$ and $\{b_t\}$ such that $\mathbb{E}[f(a_t)] \to \mathbb{E}[f(x)]$ as $t \to \infty$ and $\mathbb{E}[f(a_t)] = \mathbb{E}[f(b_t)]$ for all $t \geq 0$. Furthermore, if they satisfy that for some random time τ, $a_t = b_{t-1}$ for all $t \geq \tau$, then we have

$$\begin{aligned}
\mathbb{E}_{\mathcal{M}}[f(x)] &= \mathbb{E}_{\mathcal{M}}\left[f(a_1) + \sum_{t=2}^{\infty}(f(a_t) - f(a_{t-1}))\right] \\
&= \mathbb{E}_{\mathcal{M}}\left[f(a_1) + \sum_{t=2}^{\infty}(f(a_t) - f(b_{t-1}))\right] \\
&= \mathbb{E}_{\mathcal{M}}\left[f(a_1) + \sum_{t=2}^{\tau-1}(f(a_t) - f(b_{t-1}))\right],
\end{aligned}$$

where the second identity holds since $\mathbb{E}[f(a_t)] = \mathbb{E}[f(b_t)]$ for all $t \geq 0$, and the third one is due to the fact that $a_t = b_{t-1}$ for all $t \geq \tau$. Thus, we only need to compute the finite number of expectations since infinitely many terms are cancelled out. Such an idea seems rather simple, but the construction of the chain $\{b_t\}$, which satisfies two conditions: (i) $\mathbb{E}[f(a_t)] = \mathbb{E}[f(b_t)]$ for all $t \geq 0$; (ii) $a_t = b_{t-1}$ for all $t \geq \tau$, is a highly non-trivial task. We recommendedly defer to [3] for more details.

3 Noise Contrastive Estimation

3.1 NCE on Probability Fitting

To address the notorious normalization issue, one naive strategy is regarding it as a learnable parameter. Specifically, the model is parameterized in terms of an unnormalized distribution f_θ and a learned parameter Z_θ corresponding to the normalizing constant

$$p_\theta(x) = \exp(-f_\theta(x))/(Z_\theta). \tag{18}$$

Ideally, the maximum log-likelihood estimation can be applied to optimize parameter θ. However it fails in this scenario since the model faces a trivial solution that when $Z_\theta = 1$, the log-likelihood will be infinity.

Noise contrastive estimation address this issue by introducing a noise distribution $p_n(x)$, and the model is learned by distinguishing the sample from p_d and p_n. Following [5], assuming that noise samples are k times more frequent that data sample, we construct a mixture distribution

$$p_m(x) = \frac{1}{k+1}p_d(x) + \frac{k}{k+1}p_n(x). \tag{19}$$

Then the posterior probability that samples x came from the data distribution is

$$\begin{aligned}
p(D = 1|x) &= \frac{p(D = 1)p(x|D = 1)}{p(D = 1)p(x|D = 1) + p(D = 0)p(x|D = 0)} \\
&= \frac{\frac{1}{k+1}p_d(x)}{\frac{1}{k+1}p_d(x) + \frac{k}{k+1}p_n(x)} \\
&= \frac{p_d(x)}{p_d(x) + kp_n(x)}.
\end{aligned} \tag{20}$$

Since we would like to fit p_θ to p_d, we use p_θ in place of p_d in (20), making the posterior probability a function of the model parameter θ

$$p_\theta(D=1|x) = \frac{p_\theta(x)}{p_\theta(x) + kp_n(x)}. \tag{21}$$

To learn the model by distinguishing samples from data and noise distribution, we maximize the following objective function, which is equivalent to maximize log-likelihood estimation of Bernoulli distribution

$$\begin{aligned}\mathcal{J}(\theta) &= (k+1)\mathbb{E}_{p_m(x)}\left[\mathbb{I}_{[D(x)=1]}\log\frac{p_\theta(x)}{p_\theta(x)+kp_n(x)} + \mathbb{I}_{[D(x)=0]}\log\frac{kp_n(x)}{p_\theta(x)+kp_n(x)}\right] \\ &= \mathbb{E}_{p_d(x)}\left[\log\frac{p_\theta(x)}{p_\theta(x)+kp_n(x)}\right] + k\mathbb{E}_{p_n(x)}\left[\log\frac{kp_n(x)}{p_\theta(x)+kp_n(x)}\right].\end{aligned} \tag{22}$$

Here, we arrive at the final objective of noise contrastive estimation, which can be further approximated using Monte Carlo sampling

$$\mathcal{J}(\theta) \approx \log\frac{p_\theta(x)}{p_\theta(x)+kp_n(x)} + \sum_{i=1}^{k}\log\frac{kp_n(\tilde{x}_i)}{p_\theta(\tilde{x}_i))+kp_n(\tilde{x}_i))}, \text{where } x\sim p_d, \tilde{x}_i \sim p_n. \tag{23}$$

Note that the weights $\frac{kp_n(\tilde{x}_i)}{p_\theta(\tilde{x}_i))+kp_n(\tilde{x}_i))}$ are always lying in $(0,1)$, which make NCE-based learning very stable compared with MLE. Interestingly, as indicated in [6], simply set $Z_\theta = 1$, instead of learning it, do not affect the performance of models.

Understanding NCE To fully understande the insight behind NCE, we take the gradient of $\mathcal{J}(\theta)$ with respect to θ

$$\begin{aligned}\nabla\theta\mathcal{J}(\theta) &= \mathbb{E}_{p_d(x)}\left[\nabla_\theta\log\frac{p_\theta(x)}{p_\theta(x)+kp_n(x)}\right] + k\mathbb{E}_{p_n(x)}\left[\nabla_\theta\log\frac{kp_n(x)}{p_\theta(x)+kp_n(x)}\right] \\ &= \mathbb{E}_{p_d(x)}\left[\frac{kp_n(x)}{p_\theta(x)+kp_n(x)}\nabla_\theta\log p_\theta(x)\right] - k\mathbb{E}_{p_n(x)}\left[\frac{p_\theta(x)}{p_\theta(x)+kp_n(x)}\nabla_\theta\log p_\theta(x)\right] \\ &= \int\frac{kp_n(x)}{p_\theta(x)+kp_n(x)}(p_d(x)-p_\theta(x))\nabla_\theta\log p_\theta(x)dx.\end{aligned}$$

Then as $k\to\infty$, we have

$$\begin{aligned}\nabla\theta\mathcal{J}(\theta) &= \int(p_d(x)-p_\theta(x))\nabla_\theta\log p_\theta(x)dx \\ &= \mathbb{E}_{p_d(x)}[\nabla_\theta\log p_\theta(x)] - \mathbb{E}_{p_\theta(x)}[\nabla_\theta\log p_\theta(x)].\end{aligned} \tag{24}$$

Actually, this is the gradient of log-likelihood estimation. To show this, we have

$$\begin{aligned}\nabla_\theta\mathbb{E}_{p_d(x)}[\log p_\theta(x)] &= \mathbb{E}_{p_d(x)}\left[\nabla_\theta(-f_\theta(x)) - \nabla_\theta\log Z_\theta\right] \\ &= \mathbb{E}_{p_d(x)}\left[\nabla_\theta(-f_\theta(x)) - \frac{\int\exp(-f_\theta(x))\nabla_\theta(-f_\theta(x))dx}{Z_\theta}\right] \\ &= \mathbb{E}_{p_d(x)}[\nabla_\theta(-f_\theta(x))] - \mathbb{E}_{p_\theta(x)}[\nabla_\theta(-f_\theta(x))].\end{aligned} \tag{25}$$

As Z_θ is set to be a constant, (24) is equal to (25). That is, as $k\to\infty$, the gradient of NCE is equivalent to the maximum likelihood gradient.

3.2 Examples on Prediction Models

In prediction models, we are supposed to predict $y\in\mathcal{Y}$ from $x\in\mathcal{X}$. Following the basic idea of NCE, we can construct a joint distribution

$$p_d(i, x, y_1, \ldots, y_N) := \frac{1}{N}p_{xy}(x, y_i)\prod_{j\neq i}p_y(y_j),$$

where $p_{xy}(xy)$ represents the joint probability of x, y and $p_y(y)$ is the marginal distribution of labels y. Consequently, we can generate the samples by first drawing an index $i \in \{1, \ldots, N\}$ uniformly at random and for $j = 1, \ldots, N$ drawing $(x, y_i) \sim p_{xy}$ if $j = i$ but else drawing $y_j \sim p_y$. This yields a conditional distribution

$$p_d(i|x, y_1, \ldots, y_N) = \frac{p_{xy}(y_i|x) \prod_{j \neq i} p_y(y_j)}{\sum_{k=1}^{N} p_{xy}(y_k|x) \prod_{j \neq k} p_y(y_j)} = \frac{\frac{p_{xy}(y_i|x)}{p_y(y_i)}}{\sum_{k=1}^{N} \frac{p_{xy}(y_k|x)}{p_y(y_k)}}. \tag{26}$$

The intuition of NCE is that infer which of N samples of $\{y_1, \ldots, y_N\}$ is from the joint distribution $p_{xy}(xy)$. To this end, we further construct the following distribution with the score function $f_\theta(x, y)$

$$p_\theta(i|x, y_1, \ldots, y_N) = \frac{f_\theta(x, y_i)}{\sum_{j=1}^{N} f_\theta(x, y_j)}. \tag{27}$$

Guiding by the insight of NCE, We train the model by minimizing the conditional entropy between $p_d(i, x, y_1, \ldots, y_N)$ and $p_\theta(i|x, y_1, \ldots, y_N)$

$$\mathcal{L}_\theta := \mathbb{E}_{p_d(i,x,y_1,\ldots,y_N)}[-\log p_\theta(i|x, y_1, \ldots, y_N)]. \tag{28}$$

We further assume p_θ is **universal**, that is, it is expressive enough to model p_d such that $p_\theta(i|x, y_1, \ldots, y_N) = p_d(i|x, y_1, \ldots, y_N)$ for some θ. This assumption seems to hold in practice with neural network, though it might require an exponentially large parameter space. Under this assumption, we find that compared with the formula expressions of equation 26 and 27, the optimal parameter θ^* satisfies

$$f_\theta(x, y) \propto \frac{p_{y|x}(xy)}{p_y(y)}.$$

Using this results, we can rewrite the training objective in the case of optimal solution as

$$\begin{aligned}
\mathcal{L}_{\theta^*} &= -\mathbb{E}_{p_d}\left[\log \frac{f_{\theta^*}(x, y_i)}{\sum_{j=1}^{N} f_{\theta^*}(x, y_j)}\right] \\
&= \mathbb{E}_{p_d}\left[\log \frac{\frac{p_{xy}(y_i|x)}{p_y(y_i)} + \sum_{j \neq i} \frac{p_{xy}(y_j|x)}{p_y(y_j)}}{\frac{p_{xy}(y_i|x)}{p_y(y_i)}}\right] \\
&= \mathbb{E}_{p_d} \log\left[1 + \frac{p_y(y_i)}{p_{xy}(y_i|x)} \sum_{j \neq i} \frac{p_{xy}(y_j|x)}{p_y(y_j)}\right] \\
&\approx \mathbb{E}_{p_d} \log\left[1 + \frac{p_y(y_i)}{p_{xy}(y_i|x)}(N-1)\mathbb{E}_{p_y(y_j)}\left[\frac{p_{xy}(y_j|x)}{p_y(y_j)}\right]\right] \quad \text{(the law of large numbers)} \\
&= \mathbb{E}_{p_d} \log\left[1 + \frac{p_y(y_i)}{p_{xy}(y_i|x)}(N-1)\right] \quad (y_j \text{ is independent of } x) \\
&\geq \mathbb{E}_{p_d} \log\left[\frac{p_y(y_i)}{p_{xy}(y_i|x)}\right] \quad (p_{xy}(y_i|x) > p_y(y_i)) \\
&= -I(x; y_i) + \log N.
\end{aligned} \tag{29}$$

Therefore, $I(x; y_i) = I(x; y) \geq \log N - \mathcal{L}_\theta^{\text{opt}} \geq \log N - \mathcal{L}_\theta$, that is, minimizing $\mathcal{L}_\theta$ over θ corresponds to maximizing a parameterized lower bound of $I(x; y)$, and for this reason this estimation is sometimes called "InfoNCE".

4 Primal-Dual view of MLE

Duality provides an alternative strategy to solve the intractable issue of the log partition term [7, 8, 9]. Specifically, given an unnormalized probability density function

$$p_\theta(x) = \exp(-f_\theta(x) - \log Z_\theta), \tag{30}$$

where $Z_\theta = \int_x \exp(-f_\theta(x))\mathrm{d}x$ is the partition function, the log-partition function can be estimated by using its dual form

$$\log Z_\theta = \max_q \mathbb{E}_q[-f(x)] + H(q), \tag{31}$$

where $H(q) = -\mathbb{E}_q[\log q]$ is the entropy of $q(\cdot)$, which leads to a primal-dual view of the MLE

$$\max_\theta \mathbb{E}_{p_d}[\log p_\theta(x)] = \max_\theta \min_q \mathbb{E}_{p_d}[-f(x)] - \mathbb{E}_q[-f(x)] - H(q), \tag{32}$$

which bypasses the explicit computation of the partition function. To understand this duality, we use Jensen's inequality which is commonly used in variational analyses

$$\begin{aligned}\log Z_\theta &= \log \int q(x) \frac{\exp(-f_\theta(x))}{q(x)} \mathrm{d}x \\ &\geq \mathbb{E}_q[-f_\theta(x) - \log q(x)] \\ &= \mathbb{E}_q[-f_\theta(x)] + H(q).\end{aligned}$$

It can be further shown that the equality holds when $q = p$, via an addictive KL term

$$\begin{aligned}\mathbb{E}_q[-f_\theta(x)] + H(q) + KL(q||p) &= \int q(x) \log \frac{\exp(-f_\theta(x))}{q(x)} \frac{q(x)}{p_\theta(x)} \mathrm{d}x \\ &= \int q(x) \log \frac{\exp(-f_\theta(x))}{p_\theta(x)} \mathrm{d}x \\ &= \int q(x) \log Z_\theta \mathrm{d}x = \log Z_\theta \\ \log Z_\theta &= \mathbb{E}_q[-f_\theta(x)] + H(q), \quad when\ q(x) = p_\theta(x).\end{aligned}$$

An alternative perspective derived from Fenchel inequality is shown in [10]. However, the design of $q(x)$ is nontrival. We recommend the readers refer to [8] for more details.

References

[1] Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.

[2] Radford M Neal et al. Mcmc using hamiltonian dynamics. *Handbook of markov chain monte carlo*, 2(11):2, 2011.

[3] Yixuan Qiu, Lingsong Zhang, and Xiao Wang. Unbiased contrastive divergence algorithm for training energy-based latent variable models. In *International Conference on Learning Representations*, 2019.

[4] PE Jacob, J O'Leary, and YF Atchadé. Unbiased markov chain monte carlo with couplings. arxiv e-prints, page. *arXiv preprint arXiv:1708.03625*, 2017.

[5] Michael U Gutmann and Aapo Hyvärinen. Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. *Journal of Machine Learning Research*, 13(2), 2012.

[6] Andriy Mnih and Yee Whye Teh. A fast and simple algorithm for training neural probabilistic language models. *arXiv preprint arXiv:1206.6426*, 2012.

[7] Hanjun Dai, Rishabh Singh, Bo Dai, Charles Sutton, and Dale Schuurmans. Learning discrete energy-based models via auxiliary-variable local exploration. *arXiv preprint arXiv:2011.05363*, 2020.

[8] Bo Dai, Zhen Liu, Hanjun Dai, Niao He, Arthur Gretton, Le Song, and Dale Schuurmans. Exponential family estimation via adversarial dynamics embedding. *arXiv preprint arXiv:1904.12083*, 2019.

[9] Bo Dai, Hanjun Dai, Arthur Gretton, Le Song, Dale Schuurmans, and Niao He. Kernel exponential family estimation via doubly dual embedding. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2321–2330. PMLR, 2019.

[10] Martin J Wainwright and Michael Irwin Jordan. *Graphical models, exponential families, and variational inference*. Now Publishers Inc, 2008.