Fully Variational Noise-Contrastive Estimation

Christopher Zach(✉)

Chalmers University of Technology, Gothenburg, Sweden
zach@chalmers.se

Abstract. By using the underlying theory of proper scoring rules, we design a family of noise-contrastive estimation (NCE) methods that are tractable for latent variable models. Both terms in the underlying NCE loss, the one using data samples and the one using noise samples, can be lower-bounded as in variational Bayes, therefore we call this family of losses *fully variational noise-contrastive estimation*. Variational autoencoders are a particular example in this family and therefore can be also understood as separating real data from synthetic samples using an appropriate classification loss. We further discuss other instances in this family of fully variational NCE objectives and indicate differences in their empirical behavior.

1 Introduction

Estimating the parameters of a model distribution from a training set is an important research topic with applications in deep generative models (e.g. [5, 8, 15, 18, 24, 28]), out-of-distribution (OOD) or anomaly detection [16, 17, 23, 32] and representation learning [2, 4, 19, 22]. Maximum-likelihood estimation is the method of choice when the parametric model distribution is normalized and can be evaluated efficiently (which is the case for "elementary" probability distributions and for normalizing flows [24]). The expressiveness of a model distribution can be enhanced by introducing latent variables and by using an unnormalized distribution (also known as energy-based model). Both of these modifications prevent the maximum likelihood method from being applicable: latent variables often lead to intractable integrals or sums when computing the marginal likelihood, and likewise the normalization factor (also called the partition function) of an unnormalized model is typically intractable.

Latent variables are usually addressed by utilizing the evidence lower bound (ELBO) of the likelihood as in variational Bayes (e.g. [12]), and parameters of unnormalized models can be estimated from data by methods such as score matching [11] or noise-contrastive estimation (NCE, [9, 10]). NCE can intuitively be understood as learning a binary classifier separating training data from samples drawn from a fully known noise distribution. Variational NCE [26] aims to enable the estimation of unnormalized latent variable models from data by leverging the ELBO. It succeeds only partially, since the ELBO cannot be applied on all terms in the NCE objective, and an intractable marginal remains. In this work we derive modified instances of NCE that allow the application of the ELBO on

R. Gade et al. (Eds.): SCIA 2023, LNCS 13886, pp. 175–190, 2023.
https://doi.org/10.1007/978-3-031-31438-4_12

all terms, and the resulting objective is therefore free from intractable sums (or integrals). We call the resulting method *fully variational noise-contrastive estimation*. Interestingly, variational autoencoders [14,25] are one particular (and important) instance in this family of fully variational NCE methods.

2 Background

Proper scoring rules Let $\mathcal{P} \subseteq \mathbb{R}^d$, and let $G : \mathcal{P} \to \mathbb{R}$ be a differentiable convex mapping. The Bregman divergence between $p \in \mathcal{P}$ and $q \in \mathcal{P}$ is defined as

$$D_G(p\|q) \stackrel{\text{def}}{=} G(p) - \big(G(q) + (p-q)^\top \nabla G(q)\big), \tag{1}$$

i.e. $D_G(p\|q)$ is the error between $G(p)$ and the linearization (first-order Taylor expansion) of G at q. Convexity of G implies that $D_G(p\|q)$ is non-negative. If G is strictly convex, then $D_G(p\|q) = 0$ iff $p = q$.

Now let p and q be the parameters of a categorical distribution, i.e. $P(X = k|p) = p_k$ and $P(X = k|q) = q_k$ for a categorical random variable X with values in $\{1, \dots, d\}$. The domain $\mathcal{P}$ is therefore the probability simplex, $\mathcal{P} = \{p \in [0,1]^d : \sum_{k=1^d} p_k = 1\}$. In this setting $D_G(p\|q)$ can be stated as

$$\begin{aligned} D_G(p\|q) &= G(p) - G(q) + \mathbb{E}_{X\sim p}\left[\tfrac{d}{dq_X}G(q)\right] - \mathbb{E}_{X\sim q}\left[\tfrac{d}{dq_X}G(q)\right] \\ &= G(p) + \mathbb{E}_{X\sim p}\left[\tfrac{d}{dq_X}G(q) - G(q) - \mathbb{E}_{X'\sim q}\left[\tfrac{d}{dq_{X'}}G(q)\right]\right]. \end{aligned} \tag{2}$$

Minimizing $D_G(p\|q)$ w.r.t. q for fixed p is equivalent to

$$\begin{aligned} \arg\min_{q\in\mathcal{P}} D_G(p\|q) &= \arg\min_{q\in\mathcal{P}} -G(q) - \textstyle\sum_k (p_k - q_k)\tfrac{\partial}{\partial q_k}G(q) \\ &= \arg\max_{q\in\mathcal{P}} \mathbb{E}_{X\sim p}\left[\tfrac{\partial}{\partial q_X}G(q) + G(q) - \textstyle\sum_k q_k \tfrac{\partial}{\partial q_k}G(q)\right] \\ &= \arg\max_{q\in\mathcal{P}} \mathbb{E}_{X\sim p}\left[S(X,q)\right], \end{aligned} \tag{3}$$

where we defined the *proper scoring rule* (PSR) S as follows,

$$S(x,q) \stackrel{\text{def}}{=} \tfrac{\partial}{\partial q_x}G(q) + G(q) - \textstyle\sum_k q_k \tfrac{\partial}{\partial q_k}G(q). \tag{4}$$

Note that maximization w.r.t. q only requires samples from p, but does not need the knowledge of the distrbution p itself. Therefore proper scoring rules are one method to estimate distribution parameters when only samples from an unknown data distribution p are available.

If G is strictly convex, then the resulting PSR is a *strictly* PSR. If e.g. G is chosen as the negated Shannon entropy, then $S(x,q) = \log q_x$ is called the *logarithmic* scoring rule underlying maximum likelihood estimation and the cross-entropy loss in machine learning. It is an instance of a *local* PSR [20], which does not depend on any value of $q_{x'}$ for $x' \neq x$ (the score matching cost [11] being another example). We refer to [7] and [3] for an extensive overview and further examples of proper scoring rules.

PSRs for binary RVs When X is a binary random variable, and therefore $x \in \{0,1\}$, then we only need one parameter $\mu \in [0,1]$ to characterize the corresponding Bernoulli distribution. For a differentiable convex function $G : [0,1] \to \mathbb{R}$ the induced Bregman divergence between $\mu \in [0,1]$ and $\nu \in [0,1]$ is given by

$$D_G(\mu \| \nu) = G(\mu) - G(\nu) - (\mu - \nu) G'(\nu) \tag{5}$$

and

$$\begin{aligned}\arg\min_{\nu \in [0,1]} D_G(\mu \| \nu) &= \arg\max_{\nu \in [0,1]} G(\nu) + (\mu - \nu) G'(\nu) \\ &= \arg\max_{\nu \in [0,1]} \mathbb{E}_{x \sim \mathrm{Ber}(\mu)} \left[G(\nu) + (x - \nu) G'(\nu) \right].\end{aligned} \tag{6}$$

The resulting PSR S is therefore

$$S(1, \nu) = G(\nu) + (1 - \nu) G'(\nu) \qquad S(0, 1-\nu) = G(\nu) - \nu G'(\nu). \tag{7}$$

G can be recovered via

$$G(\nu) = \nu S(1, \nu) + (1-\nu) S(0, 1-\nu) = \mathbb{E}_{x \sim \mathrm{Ber}(\mu)} \left[S(x, x\nu + (1-x)(1-\nu)) \right]. \tag{8}$$

Noise-contrastive estimation Noise-contrastive estimation (NCE, [9,10]) ultimately casts the estimation of parameters of an unknown data distribution as a binary classification problem. Let $\Omega \subseteq \mathbb{R}^n$ and X be a n-dimensional random vector. Let p_d the (unknown) data distrbution, p_θ a model distribution (with parameters θ) and p_n a user-specified noise distribution. Let Z be a (fair) Bernoulli RV that determines whether a sample is drawn from the data (respectively model) distribution or from the noise distribution p_d.[1] NCE applies the logarithmic PSR to match the posteriors,

$$P_{d,n}(Z = 1 | X = x) = \frac{p_d(x)}{p_d(x) + p_n(x)} \quad P_{\theta,n}(Z = 1 | X = x) = \frac{p_\theta(x)}{p_\theta(x) + p_n(x)}, \tag{9}$$

which yields the NCE objective

$$J_{\mathrm{NCE}}(\theta) = \mathbb{E}_{X \sim p_d} \left[\log \frac{p_\theta(X)}{p_\theta(X) + p_n(X)} \right] + \mathbb{E}_{X \sim p_n} \left[\log \frac{p_n(X)}{p_\theta(X) + p_n(X)} \right]. \tag{10}$$

After introducing $r_\theta(x) \stackrel{\mathrm{def}}{=} p_\theta(x) / p_n(x)$ this reads as

$$J_{\mathrm{NCE}}(\theta) = \mathbb{E}_{X \sim p_d} \left[-\log\left(1 + r_\theta(X)^{-1}\right) \right] + \mathbb{E}_{X \sim p_n} \left[-\log\left(1 + r_\theta(X)\right) \right], \tag{11}$$

establishing the connection to logistic regression. At first glance this is superficially similar to GANs [8], but it lacks e.g. the problematic min-max structure of GANs. In contrast to e.g. maximum likelihood estimation, NCE is applicable even when the model distribution is unnormalized, i.e.

$$p_\theta(x) = \tfrac{1}{Z(\theta)} p_\theta^0(x) \tag{12}$$

for an unnormalized model $p_\theta^0(x)$ and an intractable partition function $Z(\theta) = \sum_x p_\theta^0(x)$.[2] NCE allows to estimate the value of the partition function $Z(\theta)$ for

[1] We omit the possibility of using general Bernoulli RV for notational simplicity.

[2] For brevity we use sums to refer to marginalization of RV, but these sums should always be understood as the appropriate Lebesque integrals.

the obtained model parameters θ by augmenting the parameter vector to (θ, Z) and use the relation $p_\theta(x) = p^0_\theta(x)/Z$. Extensions to the basic NCE framework are discussed in [1,21].

NCE is not directly applicable to latent variable models, where the joint density $p_\theta(X, Z)$ is specified, but the induced marginal $p_\theta(X)$ is only indirectly given via

$$p_\theta(x) = \sum\nolimits_z p_\theta(x, z) = \sum\nolimits_z p_\theta(x|z) p_Z(z), \tag{13}$$

where we use a generative model for the joint $p_\theta(X, Z)$.

Using latent variable models greatly enhances the expressiveness of model distributions, but exact computation of the marginal $p_\theta(x)$ is often intractable. By noting that the term under the first expectation in Eq. 11 is concave w.r.t. $r_\theta(x)$, Variational NCE [26] proposes to apply the evidence lower bound (ELBO) to obtain a tractable variational lower bound for the first term in Eq. 11. Unfortunately, the second term in Eq. 11 is convex in r_θ and the ELBO does not apply here. Importance sampling is leveraged instead to estimate the intractable expectation inside the second term. In the following section we show how the ELBO can be applied on both terms in a slightly generalized version of NCE.

3 Fully Variational NCE

First, we generalize the NCE objective (Eq. 10) to arbitrary strictly proper scoring rules for binary random variables,

$$J_{S\text{-NCE}}(\theta) = \mathbb{E}_{x\sim p_d}\left[S\left(1, \tfrac{r_\theta(x)}{1+r_\theta(x)}\right)\right] + \mathbb{E}_{x\sim p_n}\left[S\left(0, \tfrac{1}{1+r_\theta(x)}\right)\right], \tag{14}$$

where r_θ is the density ratio, $r_\theta(x) \stackrel{\text{def}}{=} p_\theta(x)/p_n(x)$. $J_{S\text{-NCE}}$ is maximized w.r.t. the parameters θ in this formulation. Recall that $r_\theta(x)/(1+r_\theta(x))$ is the posterior of x being a sample drawn from the model p_θ, and $1/(1 + r_\theta(x))$ is the posterior for x being a noise sample. Our aim is to determine a convex function G such that both mappings

$$f_1(r) = S(1, r/(1 + r)) \qquad \text{and} \qquad f_0(r) = S(0, 1/(1 + r)) \tag{15}$$

are concave. If this is the case, then

$$\begin{aligned} f_k\big(r_\theta(x)\big) &= f_k\left(\frac{p_\theta(x)}{p_n(x)}\right) = f_k\left(\frac{\sum_z p_\theta(x, z)}{p_n(x)}\right) = f_k\left(\frac{\sum_z p_\theta(x, z) q_k(z|x)}{p_n(x) q_k(z|x)}\right) \\ &\geq \sum\nolimits_z q_k(z|x) f_k\left(\frac{p_\theta(x, z)}{p_n(x) q_k(z|x)}\right) = \mathbb{E}_{z\sim q_k(Z|x)}\left[f_k\left(\frac{p_\theta(x, z)}{p_n(x) q_k(z|x)}\right)\right] \end{aligned}$$

for $k \in \{0, 1\}$. $q_k(Z|X)$ is a posterior corresponding to the encoder part. Overall, $J_{S\text{-NCE}}$ in Eq. 14 can be lower bounded as follows,

$$J_{S\text{-NCE}}(\theta) = \mathbb{E}_{x\sim p_d}\left[f_1(r_\theta(x))\right] + \mathbb{E}_{x\sim p_n}\left[f_0(r_\theta(x))\right] \geq \max_{q_1,q_0} J_{S\text{-fvNCE}}(\theta, q_1, q_0) \tag{16}$$

with the r.h.s. defined as the *fully variational* NCE loss,

$$\begin{aligned} J_{S\text{-fvNCE}}(\theta, q_1, q_0) \stackrel{\text{def}}{=} \mathbb{E}_{x\sim p_d, z\sim q_1(Z|x)} \left[f_1\left(\frac{p_\theta(x,z)}{p_n(x)q_1(z|x)} \right) \right] \\ + \mathbb{E}_{x\sim p_n, z\sim q_0(Z|x)} \left[f_0\left(\frac{p_\theta(x,z)}{p_n(x)q_0(z|x)} \right) \right]. \end{aligned} \tag{17}$$

Note that we allow in principle two separate encoders, q_1 and q_0, since the ELBO is applied at two places independently. For brevity we introduce the following short-hand notations for the joint distributions,

$$p_{d,k}(x,z) \stackrel{\text{def}}{=} p_d(x)q_k(z|x) \qquad\qquad p_{n,k}(x,z) \stackrel{\text{def}}{=} p_n(x)q_k(z|x), \tag{18}$$

resulting in a more compact expression for $J_{S\text{-fvNCE}}$,

$$J_{S\text{-fvNCE}}(\theta,q_1,q_0) = \mathbb{E}_{(x,z)\sim p_{d,1}} \left[f_1\left(\frac{p_\theta(x,z)}{p_{n,1}(x,z)} \right) \right] + \mathbb{E}_{(x,z)\sim p_{n,0}} \left[f_0\left(\frac{p_\theta(x,z)}{p_{n,0}(x,z)} \right) \right]. \tag{19}$$

From $p_\theta(x)p_\theta(z|x) = p_\theta(x,z)$ we deduce that the lower bound is tight, i.e. $J_{S\text{-NCE}}(\theta) = \max_{q_1,q_0} J_{S\text{-fvNCE}}(\theta, q_1, q_0)$ when the encoders q_1 and q_0 are equal to the model posterior, $q_1(Z|X) = q_0(Z|X) = p_\theta(Z|X)$ a.e. $J_{S\text{-fvNCE}}$ in Eq. 17 is formulated as a population loss, but the corresponding empirical risk can be immediately obtained by sampling from p_d, p_n and the encoder distributions.

Now the question is whether such concave mappings f_1 and f_0 satisfying Eq. 15 for a PSR S exist. Since common PSRs such as the logarithmic and the quadratic PSR violate these properties, existence of such a PSR is not obvious. The next section discusses how to construct such PSRs and provides examples.

4 A Family of Suitable Proper Scoring Rules

In this section we construct a pair (f_1, f_0) of concave mappings, such that the induced functions $S(1,\cdot)$ and $S(0,\cdot)$ in Eq. 15 form a PSR. The following result provides sufficient conditions on such a pair (f_1, f_0):

Lemma 1. *Let a pair of functions (f_0, f_1), $f_k : (0,\infty) \to \mathbb{R}$, satisfy the following:*

1. *Both f_1 and f_0 are concave,*
2. *f_1 and f_0 satisfy the compatibility condition*

$$f_0'(r) = -r f_1'(r) \tag{20}$$

   for all $r > 0$,
3. *the mapping $G(\mu) = \mu f_1(\mu/(1-\mu)) + (1-\mu) f_0(\mu/(1-\mu))$ is convex in $(0,1)$.*

Then S is a PSR. Such pairs (f_1, f_0) are said to have to double ELBO *property.*

Proof. We abbreviate $S_1(\mu) := S(1,\mu)$ and $S_0(1-\mu) := S(0,1-\mu)$ and recall the relations between S and G:

$$\begin{aligned} G(\mu) &= \mu S_1(\mu) + (1-\mu) S_0(1-\mu) \\ S_0(1-\mu) &= G(\mu) - \mu G'(\mu) \\ S_1(\mu) &= G(\mu) + (1-\mu) G'(\mu) = S_0(1-\mu) + G'(\mu) \end{aligned} \tag{21}$$

and therefore $G'(\mu) = S_1(\mu) - S_0(1-\mu)$. We calculate

$$G'(\mu) = S_1(\mu) - S_0(1-\mu) + \mu S_1'(\mu) - (1-\mu) S_0'(1-\mu) \tag{22}$$

Combining these relations implies that

$$\mu S_1'(\mu) - (1-\mu) S_0'(1-\mu) = 0 \iff S_0'(1-\mu) = \tfrac{\mu}{1-\mu} \cdot S_1'(\mu) \tag{23}$$

Now the relation between μ and r is $\mu = r/(1+r)$ and therefore $r = \mu/(1-\mu)$, which we use to express (f_1, f_0) in terms of (S_1, S_0),

$$f_1(r) = S_1(\mu) = S_1(r/(1+r)) \qquad f_0(r) = S_0(1-\mu) = S_0(1/(1+r)). \tag{24}$$

Using $d\mu/dr = (1+r)^{-2}$ and

$$f_1'(r) = \tfrac{1}{(1+r)^2} S_1'(\mu) \qquad\qquad f_0'(r) = -\tfrac{1}{(1+r)^2} S_0'(1-\mu),$$

the condition can be restated as

$$-(1+r)^2 f_0'(r) = r \cdot (1+r)^2 f_1'(r) \iff f_0'(r) = -r f_1'(r), \tag{25}$$

which is the second requirement on (f_1, f_0). Now if (f_1, f_0) satisfy Eq. 20, then (S_1, S_0) satisfy the relations of a binary PSR in Eq. 21 for an induced function G. If G is now convex, then (S_1, S_0) is a PSR. ☐

One consequence of the condition in Eq. 20 is, that f_1 is increasing and f_0 is decreasing or vice versa. This further implies that S cannot be symmetric, i.e.

$$S(1,\mu) \neq S(0,1-\mu), \tag{26}$$

and positive and negative samples are penalized differently in the overall loss. This is in contrast to many well-known PSR, which are symmetric (such as the logarithmic PSR used in NCE). The condition also implies that

$$f_0''(r) = -f_1'(r) - r f_1''(r) \overset{!}{\leq} 0.$$

Since f_1 is concave and $r \geq 0$, $-r f_1''(r) \geq 0$. This has to be compensated by f_1' increasing sufficiently fast with r. Since $f_1'(r) \geq -r f_1''(r) \geq 0$, f_1 is increasing and f_0 is decreasing in $\mathbb{R}_{\geq 0}$. This observation yields some intuition on $J_{S\text{-fvNCE}}$ in Eq. 17: the first term aims to align p_θ with $p_{d,1}$ by maximizing $p_\theta(x,z)/p_{n,1}(x,z)$ for real data (and its code), whereas the second term favors mis-alignment between p_θ and $p_{n,0}$ for noise samples (by minimizing the likelihood ratio $p_\theta(x,z)/p_{n,0}(x,z)$).

Equation 20 immediately allows to establish one pair (f_1, f_0) satisfying the double ELBO property: we choose $f_1(r) = \log r$, which yields $f_0'(r) = -1$ and therefore $f_0(r) = -r$. Both f_1 and f_0 are concave. Further,

$$S_1(\mu) = \log \tfrac{\mu}{1-\mu} \qquad\qquad S_0(1-\mu) = -\tfrac{\mu}{1-\mu} \tag{27}$$

and therefore

$$G(\mu) = \mu S_1(\mu) + (1-\mu) S_0(1-\mu) = \mu\left(\log \tfrac{\mu}{1-\mu} - 1\right), \tag{28}$$

which is convex in $(0,1)$. Thus, we have established the existence of one PSR allowing the ELBO being applied on both terms as in Eq. 16. This example can be generalized to the following parametrized family of PSRs:

Lemma 2. *A family of PSRs satisfying the double ELBO property is given by*

$$f_1(r) = \log(r+\beta) \qquad\qquad f_0(r) = \beta \log(r+\beta) - r \tag{29}$$

for any $\beta \geq 0$,.

Proof. This follows from

$$f_0'(r) = -r f_1'(r) = -\tfrac{r}{r+\beta} = -\tfrac{r+\beta-\beta}{r+\beta} = -1 + \tfrac{\beta}{r+\beta} \implies f_0(r) = \beta \log(r+\beta) - r.$$

Further, G'' can be calculated as

$$G''(\mu) = -\frac{1}{(1-\mu)^2(\beta\mu - \mu - \beta)} = \frac{1}{(1-\mu)^2(\mu + \beta(1-\mu))} > 0, \tag{30}$$

which establishes the convexity of G (due to $(1-\mu)^2 > 0$ and $\mu + \beta(1-\mu) > 0$ for $\mu \in (0,1)$ and $\beta \geq 0$). $\square$

A 2-parameter family of PSRs is given next.

Lemma 3. *For* $\alpha \in (0,1]$ *and* $\beta \geq 0$ *we choose*

$$f_1(r) = \tfrac{1}{\alpha}(r+\beta)^\alpha \qquad\qquad f_0(r) = -\tfrac{1}{\alpha+1}(r+\beta)^{\alpha+1}.$$

This pair induces a strictly PSR satisfying the double ELBO property.

Proof. Both f_1 and f_0 are clearly concave. We deduce

$$f_1'(r) = (r+\beta)^{\alpha-1} \qquad\qquad f_0'(r) = -r(r+\beta)^{\alpha-1} = -r f_1'(r), \tag{31}$$

hence (f_1, f_0) satisfy the condition in Eq. 20. $G''(\mu)$ can be calculated as

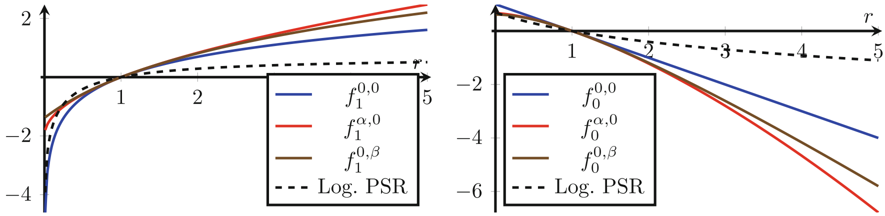
$$G''(\mu) = \left(\frac{\mu + \beta(1-\mu)}{1-\mu}\right)^\alpha \cdot \frac{\mu + \beta(2-\alpha)(1-\mu)}{(1-\mu)^2(\mu + \beta(1-\mu))^2}. \tag{32}$$

The first factor is positive for $\alpha \in (0,1]$, $\beta \geq 0$ and $\mu \in (0,1)$. Analogously, the second factor is positive since the numerator is positive for the allowed values of (μ, α, β), and the denominator is a product of squares. $\square$

Since

$$\lim_{\alpha\to 0^+} f_0'(r;\alpha,\beta) = -1 \implies \lim_{\alpha\to 0^+} f_1'(r;\alpha,\beta) = (r+\beta)^{-1}, \tag{33}$$

we deduce that the limit $\alpha \to 0^+$ yields the pair (f_1, f_0) from Lemma 2 (up to constants independent of r).



(a) S_1 (b) S_0

Fig. 1. Several pairs (f_1, f_0), in particular $(f_1^{0,0}, f_0^{0,0})$, $(f_1^{\alpha,0}, f_0^{\alpha,0})$ and $(f_1^{0,\beta}, f_0^{0,\beta})$ for $\alpha = 1/2$ and $\beta = 1$ (solid curves). Both f_1 and f_0 are concave functions. The pair (f_1, f_0) induced by the logarithmic PSR is shown for reference (dashed curve, which is concave in (a), but convex in (b)).

For visualization purposes it is convenient to normalize f_1 and f_0 such that $f_1(1) = f_0(1) = 0$ and $f_1'(1) = 1$ (and therefore $f_0'(1) = -1$). With such normalization the above pairs are given by

$$\begin{aligned} f_1(r;\alpha,\beta) &= \tfrac{(1+\beta)^{1-\alpha}}{\alpha}\big((r+\beta)^\alpha - (1+\beta)^\alpha\big) \\ f_0(r;\alpha,\beta) &= -\tfrac{(1+\beta)^{1-\alpha}}{\alpha(\alpha+1)}\big((\alpha r-\beta)(r+\beta)^\alpha - (\alpha-\beta)(1+\beta)^\alpha\big). \end{aligned} \tag{34}$$

Few instances of $(f_1^{\alpha,\beta}, f_0^{\alpha,\beta})$ are depicted in Fig. 1. We further introduce the fully variational NCE loss parametrized by (α, β),

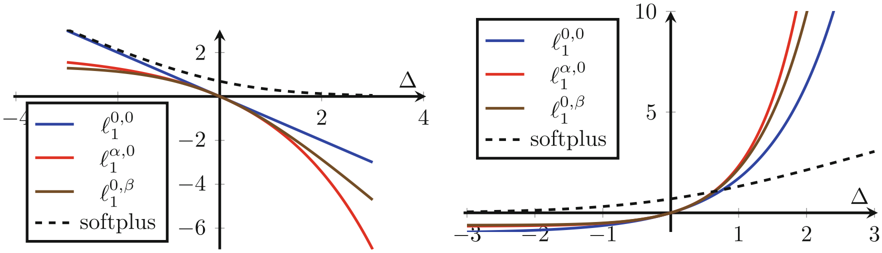
$$\begin{aligned} J^{\alpha,\beta}_{\text{fvNCE}}(\theta, q_1, q_0) \stackrel{\text{def}}{=} &\, \mathbb{E}_{(x,z)\sim p_{d,1}}\left[f_1\left(\frac{p_\theta(x,z)}{p_{n,1}(x,z)};\alpha,\beta\right)\right] \\ &+ \mathbb{E}_{(x,z)\sim p_{n,0}}\left[f_0\left(\frac{p_\theta(x,z)}{p_{n,0}(x,z)};\alpha,\beta\right)\right]. \end{aligned} \tag{35}$$

We would like to get a better understanding of these PSRs in terms of losses used for binary classification. Recall that

$$r = \tfrac{p}{q} \quad \mu = \tfrac{p}{p+q} = \tfrac{1}{1+1/r} = \tfrac{r}{1+r} = \sigma(\varDelta) \quad r = \tfrac{\mu}{1-\mu} = \tfrac{\sigma(\varDelta)}{1-\sigma(\varDelta)} = \tfrac{\sigma(\varDelta)}{\sigma(-\varDelta)} = \exp(\varDelta).$$

Here $\varDelta$ is the logit of the binary classifier. We minimize a classification loss, hence we consider the negated PSRs. Thus, we obtain for the logarithmic PSR,

$$\begin{aligned} -\log(\mu) &= -\log(\sigma(\varDelta)) = \log(1+\exp(-\varDelta)) = \text{soft-plus}(-\varDelta) \\ -\log(1-\mu) &= -\log(1-\sigma(\varDelta)) = -\log(\sigma(-\varDelta)) = \text{soft-plus}(\varDelta), \end{aligned}$$



(a) Class 1 (b) Class 0

Fig. 2. The PSRs from Fig. 1 reinterpreted as binary classification losses in terms of log-ratios $\Delta = \log r$. The soft-plus loss corresponds to the logarithmic PSR.

where soft-plus$(u) \stackrel{\text{def}}{=} \log(1 + e^u)$. Inserting $f_1(r) = \log(r + \beta)$ and $f_0(r) = \beta \log(r + \beta) - r$ yields

$$-f_1(r) = -\log(r+\beta) = -\log(e^\Delta+\beta) \doteq -\text{soft-max}(\Delta, \log\beta) = \text{soft-min}(-\Delta, -\log\beta)$$
$$-f_0(r) = r - \beta\log(r+\beta) = e^\Delta + \beta\,\text{soft-min}(-\Delta, -\log\beta)$$

Finally, $f_1(r) = r^\alpha/\alpha$, $f_0(r) = -r^{\alpha+1}/(\alpha+1)$ results in

$$-f_1(r) = -\tfrac{1}{\alpha} r^\alpha = -\tfrac{1}{\alpha} e^{\alpha\Delta} \qquad -f_0(r) = \tfrac{1}{\alpha+1} r^{\alpha+1} = \tfrac{1}{\alpha+1} e^{(\alpha+1)\Delta}.$$

Graphically, the difference between the logistic classification loss and the double-ELBO losses is, that the logistic loss solely penalizes incorrect predictions and the double ELBO losses strongly favor true positives instead (as shown in Fig. 2).

We conclude this section by a noting that non-negative linear combinations of double ELBO pairs have the double ELBO property as well:

Corollary 1. *The set of pairs with the double ELBO property is a convex cone.*

This follows from the linearity of the relations Eq. 20 and Eq. 8.

5 Instances of Fully Variational NCE

In this section we discuss several instances of $J^{\alpha,\beta}_{\text{fvNCE}}$ for specific choices of α and β. For easier identification of known frameworks we focus on normalized model distributions p_θ, but the extension to unnormalized models is straightforward.

5.1 Variational Auto-Encoders: $(\alpha, \beta) = (0, 0)$

We choose $(\alpha, \beta) = (0, 0)$ in the 2-parameter family given in Lemma 3, i.e. $f_1(r) = \log r$ and $f_0(r) = -r$. The resulting fully variational NCE objective therefore is given by

$$J^{0,0}_{\text{fvNCE}}(\theta, q_1, q_0) = \mathbb{E}_{(x,z)\sim p_{d,1}}\left[\log\left(\frac{p_\theta(x,z)}{p_{n,1}(x,z)}\right)\right] - \mathbb{E}_{(x,z)\sim p_{n,0}}\left[\frac{p_\theta(x,z)}{p_{n,0}(x,z)}\right]. \quad (36)$$

We first focus on the second term:

$$\mathbb{E}_{(x,z)\sim p_{n,0}}\left[\frac{p_\theta(x,z)}{p_{n,0}(x,z)}\right] = \sum_{x,z:p_{n,0}(x,z)>0} p_\theta(x,z) \le 1. \tag{37}$$

Now if $\mathrm{supp}(p_\theta) \subseteq \mathrm{supp}(p_{n,0})$, then the r.h.s. of Eq. 37 is exactly 1, otherwise it is bounded by 1 from above.[3] We assume that $\mathrm{supp}(p_\theta) \subseteq \mathrm{supp}(p_{n,0})$, then the last term in Eq. 36 is 1, and since $\mathbb{E}_{x\sim p_d}[\log p_n(x)]$ is constant, we obtain

$$J^{0,0}_{\mathrm{fvNCE}}(\theta, q_1, q_0) \doteq \mathbb{E}_{x\sim p_d, z\sim q_1(Z|x)}\left[\log\left(\frac{p_\theta(x,z)}{q_1(z|x)}\right)\right]. \tag{38}$$

After factorizing $p_\theta(x,z) = p_\theta(x|z)p_Z(z)$ this can be identified as the variational autoencoder loss (up to constants independent of θ and q_1),

$$J^{0,0}_{\mathrm{fvNCE}}(\theta, q_1) \doteq \underbrace{\mathbb{E}_{x\sim p_d}\left[\mathbb{E}_{z\sim q_1(Z|x)}\left[\log p_\theta(x|z)\right] - D_{KL}(q_1(Z|x)\|p_Z)\right]}_{\overset{\mathrm{def}}{=} J_{\mathrm{VAE}}(\theta, q_1)}. \tag{39}$$

Thus, in this setting standard VAE training can be understood as variance-reduced implementation of $J^{0,0}_{\mathrm{fvNCE}}$ (since the stochastic second term becomes a closed-form constant). If $\mathrm{supp}(p_\theta) \not\subseteq \mathrm{supp}(p_{n,0})$, then

$$-\mathbb{E}_{(x,z)\sim p_{n,0}}\left[\frac{p_\theta(x,z)}{p_{n,0}(x,z)}\right] \ge -1 \tag{40}$$

and optimizing the VAE loss J_{VAE} is maximizing a lower bound of $J^{0,0}_{\mathrm{fvNCE}}$. Now let $q_0(Z|X)$ be a deterministic encoder, i.e. $q_0(z|x) = \mathbf{1}[z = g_0(x)]$. In this setting

$$\begin{aligned} J^{0,0}_{\mathrm{fvNCE}}(\theta, q_1, q_0) &\doteq J_{\mathrm{VAE}}(\theta, q_1) - \mathbb{E}_{x\sim p_n}\left[\frac{p_\theta(x, g_0(x))}{p_n(x)}\right] \\ &= J_{\mathrm{VAE}}(\theta, q_1) - \sum\nolimits_x p_\theta(x, g_0(x)). \end{aligned} \tag{41}$$

Intuitively, $J^{0,0}_{\mathrm{fvNCE}}$ aims to autoencode real data well, but at the same time prefers poor reconstructions for arbitrary inputs. $J^{0,0}_{\mathrm{fvNCE}}$ uses importance weighting to estimate $\sum_x p_\theta(x, g_0(x))$. This term only becomes relevant in the objective if the two encoders q_1 and q_0 are tied in some way (otherwise g_0 may map the input to a constant code that is unlikely to be sampled from q_1).

It is interesting to note that deterministic (and tied) encoders yield somewhat different objectives when comparing classical autoencoders, VAEs and the fully variational NCE:

$$J_{\mathrm{AE}}(\theta, g) = \mathbb{E}_{x\sim p_d}\left[\log p_\theta(x|g(x))\right] \tag{42}$$

$$J_{\mathrm{VAE}}(\theta, g) = J_{\mathrm{AE}}(\theta, g) + \mathbb{E}_{x\sim p_d}\left[\log p_Z(g(x))\right] - \gamma \tag{43}$$

$$J^{0,0}_{\mathrm{fvNCE}}(\theta, g) = J_{\mathrm{VAE}}(\theta, g) - \sum\nolimits_x p_\theta(x, g(x)), \tag{44}$$

[3] If we use unnormalized models p^0_θ, then Eq. 37 is bounded by $Z(\theta)$.

where $\gamma := \max_z \log p_Z(z)$ is introduced to ensure $\log p_Z(z) - \gamma \le 0$,[4] which allows us to obtain the following chain of inequalities,

$$J_{\mathrm{AE}}(\theta, g) \ge J_{\mathrm{VAE}}(\theta, g) \ge J^{0,0}_{\mathrm{fvNCE}}(\theta, g). \tag{45}$$

$J^{0,0}_{\mathrm{fvNCE}}$ can be also interpreted as a well-justified instance of regularized autoencoders [6]. When using tied stochastic encoders $q_0 = q_1$ satisfying $\mathrm{supp}(p_\theta) \subseteq \mathrm{supp}(p_{n,0})$, using the empirical version the 2nd expectation in Eq. 36 (instead of dropping it due to being a constant) can be beneficial in scenarios explicitly requiring poor reconstruction of certain inputs. The downside is a higher variance in the empirical loss and its gradients. Overall, a variational autoencoder can be generally understood as variance-reduced instance of fully variational NCE.

5.2 "Robustified" VAEs: $(\alpha, \beta) = (0, 1)$

Now we consider the pair $f_1(r) = \log(1 + r)$ and $f_0(r) = \log(1 + r) - r$. We read

$$\begin{aligned} J^{0,1}_{\mathrm{fvNCE}}(\theta, q_1, q_0) &= \mathbb{E}_{(x,z)\sim p_{d,1}}\left[\log\left(1 + \frac{p_\theta(x,z)}{p_{n,1}(x,z)}\right)\right] \\ &\quad + \mathbb{E}_{(x,z)\sim p_{n,0}}\left[\log\left(1 + \frac{p_\theta(x,z)}{p_{n,0}(x,z)}\right) - \frac{p_\theta(x,z)}{p_{n,0}(x,z)}\right]. \end{aligned} \tag{46}$$

We assume $\mathrm{supp}(p_\theta) \subseteq \mathrm{supp}(p_{n,0})$, then the 3rd term can be dropped (see Sect. 5.1). With tied encoders $q = q_1 = q_0$ we arrive at a near-symmetric cost

$$\begin{aligned} J^{0,1}_{\mathrm{fvNCE}} &\doteq \mathbb{E}_{(x,z)\sim p_{d,1}}\left[\log\left(1 + \frac{p_\theta(x,z)}{p_{n,1}(x,z)}\right)\right] + \mathbb{E}_{(x,z)\sim p_{n,0}}\left[\log\left(1 + \frac{p_\theta(x,z)}{p_{n,0}(x,z)}\right)\right] \\ &= \mathbb{E}_{(x,z)\sim p_{d,1}}\left[\log\left(1 + \frac{p_\theta(x,z)}{p_n(x)q(z|x)}\right)\right] + \mathbb{E}_{(x,z)\sim p_{n,1}}\left[\log\left(1 + \frac{p_\theta(x,z)}{p_n(x)q(z|x)}\right)\right] \\ &= \mathbb{E}_{(x,z)\sim p_{d,1}}\left[\text{soft-plus}(\varDelta(x,z)\right] + \mathbb{E}_{(x,z)\sim p_{n,1}}\left[\text{soft-plus}(\varDelta(x,z)\right], \end{aligned} \tag{47}$$

where we introduced the shorthand notation $\varDelta(x,z) = \log p_\theta(x,z) - \log p_n(x) - \log q(z|x)$. This lower bound is tight if $q(z|x) = p_\theta(z|x) = p_\theta(x,z)/p_\theta(x)$. In this case the ratio inside the log simplifies to

$$\frac{p_\theta(x,z)}{p_n(x)q(z|x)} = \frac{p_\theta(x,z)p_\theta(x)}{p_n(x)p_\theta(x,z)} = \frac{p_\theta(x)}{p_n(x)} \tag{48}$$

and $\varDelta(x,z) = \log p_\theta(x) - \log p_n(x)$. Note that $\log p_n(x)$ is expected to be small for real samples x and large for noise samples. $J^{0,1}_{\mathrm{fvNCE}}$ can be interpreted as a version of VAEs aiming to reconstruct both real and noise samples well, but is based on a robustified reconstruction error (but with different and sample dependent truncation values for real and noise samples). In practice this cost appears to behave similar to AEs and VAEs (see Sect. 6.2 and Table 1).

[4] This is only necessary for continuous latent variables as pmf's are always in $[0, 1]$.

5.3 Weighted Squared Distance: $(\alpha, \beta) = (1, 0)$

As a last example we consider $f_1(r) = r$ and $f_0(r) = -r^2/2$:

$$J^{1,0}_{\text{fvNCE}}(\theta, q_1, q_0) = \mathbb{E}_{(x,z)\sim p_{d,1}}\left[\frac{p_\theta(x,z)}{p_{n,1}(x,z)}\right] - \frac{1}{2}\mathbb{E}_{(x,z)\sim p_{n,0}}\left[\left(\frac{p_\theta(x,z)}{p_{n,0}(x,z)}\right)^2\right] \quad (49)$$

Note that the encoder q_1 cancels in the first term, as

$$\mathbb{E}_{(x,z)\sim p_{d,1}}\left[\frac{p_\theta(x,z)}{p_{n,1}(x,z)}\right] = \sum_{x,z}\frac{p_d(x)q_1(z|x)p_\theta(x,z)}{p_n(x)q_1(z|x)} = \mathbb{E}_{\substack{x\sim p_d\\ z\sim p_Z}}\left[\frac{p_\theta(x|z)}{p_n(x)}\right]. \quad (50)$$

Therefore q_1 does not appear in the r.h.s. of Eq. 49 and can be omitted. Further, the last term in $J^{1,0}_{\text{fvNCE}}$ is the (Neyman) χ^2-divergence between $p_\theta(X, Z)$ and $p_n(X)q_0(Z|X)$. After some algebraic manipulations it can be shown that $J^{1,0}_{\text{fvNCE}}$ is (up to constants) a weighted squared distance,

$$J^{1,0}_{\text{fvNCE}}(\theta, q_0) \doteq \frac{1}{2}\sum_{x,z}\frac{\left(p_\theta(x,z) - p_{d,0}(x,z)\right)^2}{p_{n,0}(x,z)}. \quad (51)$$

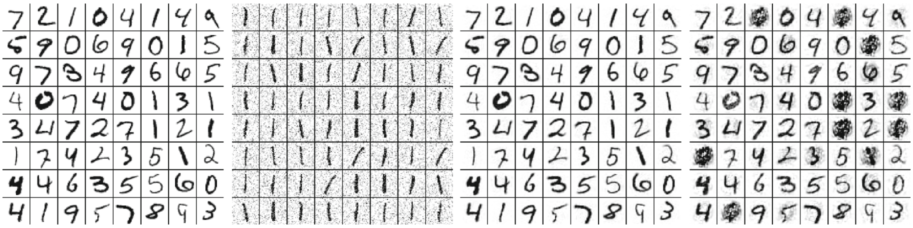
Overall the aim is to minimize the weighted squared distance between the generative joint model $p_\theta(X, Z)$ and the data-encoder induced one $p_d(X)q(Z|X)$. In contrary to the setting where $\alpha = 0$ (or α is at least small) and therefore it is natural to model $\log p_\theta$, it seems more natural to model p_θ directly (instead of the log-likelihood) in Eq. 49. Hence, the choice $\alpha = 1$ is connected to density ratio estimation [29,30], that typically uses shallow mixture models to represent the density ratio p_θ/p_n. In fact, $J^{1,0}_{\text{fvNCE}}$ in Eq. 49 is closely related to least-squares importance fitting [13] when $q_0 = q_1$.

6 Numerical Experiments

In this section we illustrate the difference in the behavior of several instances of $J^{\alpha,\beta}_{\text{fv-NCE}}$—in particular in comparison with classical autoencoders and VAEs—on toy examples.

6.1 Noise-Penalized Variational Autoencoders

First, we demonstrate the capability to steer the behavior of an 784-256-784 autoencoder (with deterministic encoder) by using $J^{0,0}_{\text{fv-NCE}}$ (Eq. 44). The noise distribution p_n is a kernel density estimate of inputs depicting the digit "1" from a validation set. Since the cost for false positives induced by $-f^{0,0}_0 = r$ is higher than the cost for false negatives ($-f^{0,0}_1(r) = -\log r$), anything resembling a digit "1" is expected to be poorly reconstructed—even when those digits appear frequently in the training data. Figure 3 visually verifies this on test inputs. This feature of Eq. 44 is useful when training data for OOD detection is contaminated by outliers, but a collection of outliers is available; or when an autoencoder-based OOD detector is required to identify certain patterns as OOD.



(a) Test inputs (b) Samples from p_n (c) VAE (d) $J^{0,0}_{\text{fv-NCE}}$

Fig. 3. The impact of the 2nd term in Eq. 44 on the reconstruction of test inputs (a). p_n is chosen as a kernel-density estimator of several digits in a validation set showing "1", with samples shown in (b). Reconstructions of the inputs using a VAE-trained encoder-decoder are given in (c), and (d) shows the corresponding reconstruction for a encoder-decoder trained using $J^{0,0}_{\text{fv-NCE}}$ (Eq. 44). Input patches showing a "1" are poorly reconstructed (as intended).

6.2 Stronger Noise Penalization Using $J^{\alpha,0}_{\text{fv-NCE}}$

Since $f_0^{\alpha,0}$ penalizes false positives stronger than $f_1^{\alpha,0}$ does for false negatives, we expect different solutions for different choices of α. With infinite data and correctly specified models $\log p_\theta$, all PSRs will return the same solution (up to the issue of local maxima), but we only have finite training data and clearly underspecified models.

We fix the decoder variance to $\sigma^2_{\text{dec}} = 1/8^2$ and use a kernel density estimate with bandwidth $\sigma_{\text{kde}} = 2\sigma_{\text{dec}}$ as noise distribution p_n. By setting $\alpha > 0$, noise samples (which are near the training data in this setting) force the model p_θ to explicitly concentrate on the training data. Samples $x \sim p_n$ have a larger reconstruction error as compared to the VAE setting ($\alpha = 0$). Table 1 lists average decoding log-likelihoods for several values of α. VAEs reconstruct noise samples worse than standard autoencoders (AEs) due to their latent code regularization. This behavior is generally amplified for increasing α, as the difference between the average reconstruction error grows with α. We also include $J^{0,1}_{\text{fvNCE}}$ (Sect. 5.2) for reference, which behaves in practice similar to VAEs. Figure 4 visualizes the decreasing reconstruction quality of samples drawn from p_n.

In order to avoid vanishing gradients when $\alpha > 0$ in the initial training phase, in view of Cor. 1 we use actually a linear combination of $J^{\alpha,0}_{\text{fvNCE}}$ (with weight 0.9) and $J^{0,0}_{\text{fvNCE}}$ (with weight 0.1) as training loss. Table 1 lists the values for two ReLU-based MLP networks (trained from the same random initial weights) obtained after 100 epochs. Since the log-ratios such as $\log r = \log p_\theta(x,z) - \log p_{n,1}(x,z)$ can attain large magnitudes, expressions such as r^α and $r^{\alpha+1}$ are evaluated using a "clipped" exponential function: we use the first-order approximation $e^T(u - T + 1)$ when $u > T$ for a threshold value T, which is chosen as $T = 10$ in our implementation.

Table 1. Average log-likelihood $\log p_\theta(x|g(x))$ in nats Higher values indicate lower reconstruction error.

(a) 784-128-784

Method/(α, β)	$x \sim p_d$	$x \sim p_n$	Difference
AE	766	-1138	1904
VAE	765	-1609	2374
$(1/256, 0)$	749	-1665	2414
$(1/64, 0)$	698	-1863	2561
$(1/16, 0)$	753	-41818	2571
$(0, 1)$	736	-1656	2392

(b) 784-256-128-256-784

Method/(α, β)	$x \sim p_d$	$x \sim p_n$	Difference
AE	756	-919	1675
VAE	769	-1373	2142
$(1/256, 0)$	774	-1402	2176
$(1/64, 0)$	748	-1520	2268
$(1/16, 0)$	777	-1463	2240
$(0, 1)$	775	-41372	2147

(a) Noise samples (b) AE: -1138 nats (c) VAE: -1609 nats (d) $\alpha=\frac{1}{16}$: -1818 nats

Fig. 4. Reconstruction of samples $x \sim p_n$ (a). VAEs (c) and $J^{\alpha,0}_{\text{fvNCE}}$ (d) increasingly force such samples to be poorly reconstructed compared to AEs (b), while maintaining a similar reconstruction error for training data $x \sim p_d$ (see Table 1).

7 Conclusion

In this work we propose fully variational noise-contrastive estimation as a tractable method to apply noise-contrastive estimation on latent variable models. As with most variational inference methods, the resulting empirical loss only needs samples from the data, noise and encoder distributions. We are largely interested in the existence and basic properties of such framework and unravel a connection with variational autoencoders. In light of this connection, VAEs are now justified to be steered explicitly towards poorly reconstructing samples from a user-specified noise distribution.

The utility of our framework for improved OOD detection and enabling general energy-based decoder models is left as future work. Further, the highly asymmetric nature of the classification loss suggests a potential but yet-to-explore connection with one-class SVMs [27] and support vector data description [31].

Acknowledgement. This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

References

1. Ceylan, C., Gutmann, M.U.: Conditional noise-contrastive estimation of unnormalised models. In: International Conference on Machine Learning, pp. 726–734. PMLR (2018)
2. Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., Abbeel, P.: Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In: Advances in Neural Information Processing Systems, vol. 29 (2016)
3. Dawid, A.P., Musio, M.: Theory and applications of proper scoring rules. METRON **72**(2), 169–183 (2014). https://doi.org/10.1007/s40300-014-0039-y
4. Dayan, P., Hinton, G.E., Neal, R.M., Zemel, R.S.: The helmholtz machine. Neural Comput. **7**(5), 889–904 (1995)
5. Dhariwal, P., Nichol, A.: Diffusion models beat gans on image synthesis. Adv. Neural. Inf. Process. Syst. **34**, 8780–8794 (2021)
6. Ghosh, P., Sajjadi, M.S., Vergari, A., Black, M.: From variational to deterministic autoencoders. In: 8th International Conference on Learning Representations (2020)
7. Gneiting, T., Raftery, A.E.: Strictly proper scoring rules, prediction, and estimation. J. Am. Stat. Assoc. **102**(477), 359–378 (2007)
8. Goodfellow, I.J., et al.: Generative adversarial nets. In: NIPS (2014)
9. Gutmann, M., Hyvärinen, A.: Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, pp. 297–304 (2010)
10. Gutmann, M.U., Hyvärinen, A.: Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. J. Mach. Learn. Res. **13**(Feb), 307–361 (2012)
11. Hyvärinen, A.: Estimation of non-normalized statistical models by score matching. J. Mach. Learn. Res. **6**(4) (2005)
12. Jordan, M.I., Ghahramani, Z., Jaakkola, T.S., Saul, L.K.: An introduction to variational methods for graphical models. Mach. Learn. **37**(2), 183–233 (1999)
13. Kanamori, T., Hido, S., Sugiyama, M.: A least-squares approach to direct importance estimation. J. Mach. Learn. Res. **10**, 1391–1445 (2009)
14. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. In: Proceedings of the 2nd International Conference on Learning Representations (ICLR) (2014)
15. Kingma, D.P., Dhariwal, P.: Glow: Generative flow with invertible 1x1 convolutions. In: Advances in Neural Information Processing Systems, vol. 31 (2018)
16. Kirichenko, P., Izmailov, P., Wilson, A.G.: Why normalizing flows fail to detect out-of-distribution data. Adv. Neural. Inf. Process. Syst. **33**, 20578–20589 (2020)
17. Liu, W., Wang, X., Owens, J., Li, Y.: Energy-based out-of-distribution detection. Adv. Neural. Inf. Process. Syst. **33**, 21464–21475 (2020)
18. Mirza, M., Osindero, S.: Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784 (2014)
19. Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)
20. Parry, M., Dawid, A.P., Lauritzen, S., et al.: Proper local scoring rules. Ann. Stat. **40**(1), 561–592 (2012)
21. Pihlaja, M., Gutmann, M., Hyvärinen, A.: A family of computationally efficient and simple estimators for unnormalized statistical models. In: Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence, pp. 442–449 (2010)

22. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434 (2015)
23. Ren, J., et al.: Likelihood ratios for out-of-distribution detection. In: Advances in Neural Information Processing Systems, vol. 32 (2019)
24. Rezende, D., Mohamed, S.: Variational inference with normalizing flows. In: International conference on machine learning, pp. 1530–1538. PMLR (2015)
25. Rezende, D.J., Mohamed, S., Wierstra, D.: Stochastic backpropagation and approximate inference in deep generative models. In: Proceedings of the 31st International Conference on Machine Learning (ICML-14), pp. 1278–1286 (2014)
26. Rhodes, B., Gutmann, M.U.: Variational noise-contrastive estimation. In: The 22nd International Conference on Artificial Intelligence and Statistics, pp. 2741–2750. PMLR (2019)
27. Schölkopf, B., Platt, J.C., Shawe-Taylor, J., Smola, A.J., Williamson, R.C.: Estimating the support of a high-dimensional distribution. Neural Comput. **13**(7), 1443–1471 (2001)
28. Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S., Poole, B.: Score-based generative modeling through stochastic differential equations. In: International Conference on Learning Representations (2020)
29. Sugiyama, M., Suzuki, T., Kanamori, T.: Density ratio estimation in machine learning. Cambridge University Press (2012)
30. Sugiyama, M., Suzuki, T., Kanamori, T.: Density-ratio matching under the bregman divergence: a unified framework of density-ratio estimation. Ann. Inst. Stat. Math. **64**(5), 1009–1044 (2012)
31. Tax, D.M., Duin, R.P.: Support vector data description. Mach. Learn. **54**(1), 45–66 (2004)
32. Zenati, H., Romain, M., Foo, C.S., Lecouat, B., Chandrasekhar, V.: Adversarially learned anomaly detection. In: 2018 IEEE International conference on data mining (ICDM), pp. 727–736. IEEE (2018)