Variational Annealing of GANs: A Langevin Perspective

Chenyang Tao [1] **Shuyang Dai** [1] **Liqun Chen** [1] **Ke Bai** [1] **Junya Chen** [2] **Chang Liu** [3] **Ruiyi Zhang** [1]
Georgiy Bobashev [4] **Lawrence Carin** [1]

Abstract

The generative adversarial network (GAN) has received considerable attention recently as a model for data synthesis, without an explicit specification of a likelihood function. There has been commensurate interest in leveraging likelihood estimates to improve GAN training. To enrich the understanding of this fast-growing yet almost exclusively heuristic-driven subject, we elucidate the theoretical roots of some of the empirical attempts to stabilize and improve GAN training with the introduction of likelihoods. We highlight new insights from variational theory of diffusion processes to derive a likelihood-based regularizing scheme for GAN training, and present a novel approach to train GANs with an unnormalized distribution instead of empirical samples. To substantiate our claims, we provide experimental evidence on how our theoretically-inspired new algorithms improve upon current practice.

1. Introduction

Modern applications of data science necessitate more expressive, robust and efficient probabilistic models, to capture the rich structure in complex datasets. These models generally fall into two major categories: likelihood-based and likelihood-free. The former explicitly assigns a likelihood function $p_G(x;\theta)$ with parameters θ to describe the data x, while the latter learns a model from which samples from the desired distribution may be drawn (but does not assign or learn a form for the distribution itself). Specifically, likelihood-free methods typically pass samples z from a pre-specified simple distribution $q(z)$ through a deterministic mapping $G(z;\theta) : \mathcal{Z} \to \mathcal{X}$, commonly known as the *generator*. These two paradigms intersect when $G(z;\theta)$ is a diffeomorphism between $\mathcal{Z}$ and $\mathcal{X}$ (Dinh et al., 2017).

While being more restrictive by construction, likelihood-based models have been widely studied within the statistics literature. Most likelihood-based models either directly minimize the Kullback-Leibler divergence $\mathrm{KL}(p_d \parallel p_G)$ between the data and model distributions (*e.g.*, maximal likelihood estimation (MLE)), or a variational bound to it (*e.g.*, variational inference (VI)). Recent advances on the scalability (Hoffman et al., 2013), flexibility (Kingma et al., 2016) and automation of inference procedures (Ranganath et al., 2014) have made such models appealing for modern applications.

In the absence of a tractable likelihood, the development of practical likelihood-free models has struggled (Diggle & Gratton, 1984; Beaumont et al., 2002; Didelot et al., 2011), for a lack of mathematical and computational tools. The recent introduction of the *generative adversarial network* (GAN) presented a simple and elegant solution to this difficulty (Goodfellow et al., 2014), revolutionizing the practice. The key idea behind GAN is the design of a critic $D(x;\omega)$, parameterized by ω, and a variational functional $V(p_d, p_G; D)$ that depends on the critic. The variational functional is chosen such that: (i) $V(p_d, p_G; D)$ can be estimated using samples from the respective distributions, and (ii) solving for $d(p_d, p_G) \triangleq \max_D V(p_d, p_G; D)$ establishes a proper discrepancy metric between distributions p_d and p_G (Li et al., 2015; Nowozin et al., 2016; Mohamed & Lakshminarayanan, 2016; Arjovsky et al., 2017). GAN matches p_G to p_d by seeking a Nash equilibrium of the adversarial minimax game $\min_G \max_D V(p_d, p_G; D)$ between the critic and generator.

Both modeling approaches have their limitations. Likelihood models are prone to learn a model that produces unrealistic samples (possibly because of a misspecification of the model form $p_G(x;\theta)$), while the major criticisms of adversarial models target their brittle training (Arjovsky & Bottou, 2017) and mode-trapping behaviors (Salimans et al., 2016). Considerable empirical efforts have been made to combine these two approaches, in the hope of marrying the

*Equal contribution [1]Electrical & Computer Engineering, Duke University, Durham, NC, USA [2]ISTBI, Fudan University, Shanghai, China [3]Computer Science & Technology, Tsinghua University, Beijing, China [4]RTI International, Research Triangle Park, NC, USA. Correspondence to: Chenyang Tao <chenyang.tao@duke.edu>, Lawrence Carin <lcarin@duke.edu>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

best of both. Adversarial strategies have been formulated to construct non-parametric likelihoods (Yeh et al., 2016; Rosca et al., 2017; 2018), estimate the intractable posterior (Mescheder et al., 2017), and ameliorate the sample-coverage issue (Makhzani et al., 2015; Tolstikhin et al., 2018) for likelihood-based learning. Similarly, likelihood estimates have also been used extensively to guide or stabilize adversarial distribution matching (Wang & Liu, 2016; Warde-Farley & Bengio, 2017; Che et al., 2017b; Li & Turner, 2018). While the exact workings of these methods are sometimes not well understood beyond heuristics, they have yielded promising results.

An alternative categorization of learning a generative sampler is based on whether the supervision is enforced through (a) empirical samples or (b) a (possibly unnormalized) distribution $u(x)$. The latter naturally arises in many statistical learning problems, for example Bayesian analysis (Welling & Teh, 2011) and reinforcement learning (Nachum et al., 2017). GANs have typically been trained under case (a), using samples from the underlying true distribution, and such models cannot directly leverage $u(x)$ (assuming $u(x)$ is difficult to sample). Recent attempts have been made to fill this gap: Stein variational gradient descent (SVGD) used the kernel trick to analytically derive the functional gradient of an implicitly defined critic based on $u(x)$ (Liu, 2017), while A-NICE-MC (Song et al., 2017) instead learned its transition kernel and proposal distribution wrt $u(x)$ through a GAN-style objective.

In this work we reconsider the integration of likelihood-based and likelihood-free approaches in generative modeling, and demonstrate how the re-introduction of likelihoods may improve GAN training. We set out by (i) presenting theoretical insights connecting likelihood regularized Kullback–Leibler GAN to the damped Langevin diffusion process, which motivates (ii) a novel *variational annealing* strategy to stabilize and facilitate general GAN training. In addition, (iii) we couple GANs with normalizing flows to train efficient neural samplers from unnormalized distributions $u(x)$. Our theory unifies many heuristically-driven practices in the literature, and the proposed algorithms represent a more principled attempt to address the challenges in GAN training from a likelihood-based perspective.

Notation. In the following, ∇, $\nabla\cdot$ and Δ respectively denote the gradient, divergence, and Laplacian operators wrt the spatial variable x, and we use ∂_t for the temporal derivative. To simplify our discussions, we always assume all distributions are compactly supported on $\mathcal{X} \subseteq \mathbb{R}^d$, and use $\mathcal{P}$ to denote all distributions defined on $\mathcal{X}$. We interchangeably use $p_d(x)$ or $\mu(x)$ for the data distribution, and $p_G(x)$ or $\rho(x)$ for the model distribution throughout the text.

2. Generative Adversarial Net, Annealing and Likelihood Regularization

In this section we will develop theory to interpret enforcing likelihood regularization as a way to amortize GAN training through annealing. In particular, we derive analytical results for a particular case of GAN to motivate more general algorithms proposed in subsequent sections. Detailed derivations can be found in the Supplementary Material (SM).

2.1. Reverse-KL GAN

Consider the family of f-divergences (Csiszár, 1963) defined as $\mathbb{D}_f(p \parallel q) \triangleq \int f\left(\frac{p(x)}{q(x)}\right) q(x)\,\mathrm{d}x$, where $p(x), q(x)$ are probability densities and $f(r) : \mathbb{R}_+ \to \mathbb{R}$ is a convex function satisfying $f(1) = 0$. It can be readily verified from the Jensen's inequality that $\mathbb{D}_f(p \parallel q) \geq 0$, the equality holds when $p \overset{a.s.}{=} q$. A natural choice of f-divergence is the Kullback-Leibler (KL) divergence where $f(t) = -\log(t)$. Hereafter we call $\mathrm{KL}(p_G \parallel p_d)$ as the reverse KL (RKL) to distinguish it from the common forward form $\mathrm{KL}(p_d \parallel p_G)$ implicitly used in MLE. Below we leverage RKL-GAN as a case of particular interest, since an analytical result can be derived in this case.

Direct optimization of an f-divergence objective typically requires explicit access to the model likelihood $p_G(x)$, which can be difficult. Fortunately, this can be recast as a GAN training through its variational formulation

$$\mathbb{D}_f(p \parallel q) = \max_D \{\underbrace{\mathbb{E}_{X\sim p_d}[D(X)] - \mathbb{E}_{X'\sim p_G}[f^*(D(X'))]}_{V_f(p_d, p_G; D)}\},$$

where $f^*(r) \triangleq \sup_{u\in\mathrm{supp}(f)}\{ur - f(u)\}$ is the Frechet conjugate of $f(r)$. Using $V_f(p_G, p_d; D)$ as the variational objective for GAN training optimizes f-divergence without explicitly evaluating $p_G(x)$. For RKL, we have

$$V_{\mathrm{RKL}}(\rho, \mu; D) = \mathbb{E}_{X\sim\mu}[D(X)] + \mathbb{E}_{X'\sim\rho}[\log(-D(X'))].$$

It also turns out that the maximizing of $D(x)$ can be reformulated as the estimation of the (log-) likelihood ratio $r(x) = \log \frac{p_G(x)}{p_d(x)}$.

2.2. Sampling unnormalized distributions with annealed RKL-GAN

We are interested in sampling from probability density $\mu(x)$ described by $\mu(x) \propto \exp(-\psi(x))$, where $\psi(x) : \mathcal{X} \to [0, \infty)$ is a smooth function commonly known as the *potential*. To sample from $\mu(x)$, one can construct a Markov chain with $\mu(x)$ as its invariant measure. One standard construction, which we will revisit in detail later, is the Itô-Langevin diffusion, given by

$$\mathrm{d}X(t) = -\nabla\psi(X(t))\,\mathrm{d}t + \sqrt{2}\,\mathrm{d}W(t), \qquad (1)$$

where $W(t)$ is the standard Wiener process (Doob, 1953). Samples are obtained via subsampling a particle trajectory

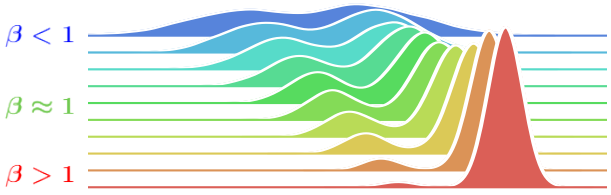

Figure 1. Illustration of damped diffusion in 1-D. Green denotes proper approximation to the target density ($\beta \approx 1$), cool colors denote under-damped diffusion ($\beta < 1$, mode covering) and warm colors denote over-damped diffusion ($\beta > 1$, mode seeking).

$\{x_t; t \in \mathbb{R}_+\}$ evolved according to (1) following a proper burn-in embargo.

Direct sampling from a Markov chain defining $\mu(x)$ can be challenging (Liu, 2008), and *simulated annealing* (SA) techniques have been developed to alleviate the difficulties (Kirkpatrick et al., 1983; Neal, 2001). In particular, SA introduces an *inverse temperature parameter* β to construct intermediate distributions $\mathcal{Q} = \{\mu_\beta\}_{\beta \in [0,1]}$, where $\mu_\beta(x) \propto \exp(-\beta\psi(x))$. $\mathcal{Q}$ continuously bridges a uniform distribution ($\beta = 0$) to the target distribution ($\beta = 1$), and one proceeds by evolving an ensemble of particles through a sequence of Markov chains respectively designed for $\{\mu_{\beta_k}\}_{k=1}^K$, where the $\{\beta_k\}$ are gradually annealed from 0 to 1.

We now consider optimizing a model distribution ρ towards an annealed target μ_β wrt the RKL objective $\mathrm{KL}(\rho \parallel \mu_\beta)$, under the assumption $\mu(x)$ can be directly evaluated. It is easy to show that

$$\mathrm{KL}(\rho \parallel \mu_\beta) = F_\mu(\rho; \beta) + C_\beta, \tag{2}$$

where $F_\mu(\rho; \beta) = \mathrm{KL}(\rho \parallel \mu) + (1 - \beta)\mathbb{E}_{X \sim \rho}[\log \mu(X)]$ and C_β is a constant independent of ρ. This readily implies that we can target a deformed distribution μ_β assuming we have a tractable μ and empirical samples from it, without explicit evaluations of the model likelihood $\rho(x)$ via applying the f-GAN technique described above to the RKL objective. We will refer to this procedure as *annealed RKL-GAN*.

2.3. Likelihood-based annealing

We now consider the implications of annealed RKL-GAN (see Figure 1). When $\beta = 1$, the likelihood term vanishes, and minimizing $F_\mu(\rho; \beta)$ recovers the target distribution $\mu(x)$. When $\beta < 1$, the stationary solution demonstrates a mode-covering behavior, putting more mass at low density regions. In the asymptotic limit $\beta \to 0$, the solution maximizes the Shannon entropy, producing a uniform distribution. Consequently, isolated modes will be joined under a sufficiently small β, making it easier to train a generator that properly distribute its mass. When $\beta > 1$, the solution reverts to a mode-seeking behavior, putting extra mass at more probable regions. In the asymptotic limit $\beta \to \infty$, the solution degenerates to point masses on the modes of $\mu(x)$.

We synthesize the standard results for Langevin systems in the theorem below to assist our development of theories, which reveal more fundamental connections between Langevin diffusion and annealed RKL-GAN.

Theorem 2.1 (Damped Langevin diffusion, Fokker-Plank equation and free energy minimization (Jordan et al., 1998; Reichl, 2016))**.** *Consider damped Langevin system $dX(t) = -\nabla\psi(X(t))\,dt + \sqrt{2\beta^{-1}}\,dW(t)$ with damping factor $\beta > 0$. Let $\rho(x, t)$ be the density of an infinite ensemble of particles governed by the damped Langevin dynamics described above, then:*

- *the temporal evolution of $\rho(x, t)$ is governed by the Fokker-Plank equation*
$$\partial_t \rho_t = \nabla \cdot (\rho_t \nabla\psi) + \beta^{-1}\Delta\rho_t\ ; \tag{3}$$
- *the stationary distribution of the above Langevin system is given by the Gibbs distribution $\rho_s(x; \beta) \propto \exp(-\beta\psi(x))$;*
- *the stationary distribution $\rho_s(x; \beta)$ also solves the variational problem $\arg\min_{\rho \in \mathcal{P}} F_\mu(\rho; \beta)$.*

From an information-theoretic perspective, $F_\mu(\rho; \beta) = \beta\mathcal{E}_\mu(\rho) + \mathcal{S}(\rho)$ is the free energy, where $\mathcal{E}_\mu(\rho) \triangleq \int_{\mathcal{X}} \psi(x)\rho(x)\,\mathrm{d}x$ and $\mathcal{S}(\rho) \triangleq \int_{\mathcal{X}} \rho(x)\log\rho(x)\,\mathrm{d}x$ are respectively known as the evidence and entropy functionals. The evidence term tells how likely an observation will occur under the ground truth and the entropy quantifies the diversity of a distribution. So intuitively, $F_\mu(\rho; \beta)$ seeks a balance between: (i) synthesizing more likely samples via minimizing the negative log-likelihood $\psi(x)$, and (ii) encouraging the diversity of synthesized samples via maximizing the entropy of $\rho(x)$. This intuition has been repeatedly exploited by researchers to design regularization schemes for generative modeling (Li & Turner, 2018; Che et al., 2017a; Anonymous, 2019).

Now let us rearrange the terms in the free energy functional so that the RKL term appears explicitly:

$$F_\mu(\rho; \beta) = \mathrm{KL}(\rho \parallel \mu) + (\beta - 1)\mathcal{E}_\mu(\rho) \tag{4}$$
$$= \beta\,\mathrm{KL}(\rho \parallel \mu) + (1 - \beta)\mathcal{S}(\rho). \tag{5}$$

Equation (4) rediscovers the annealed RKL-GAN objective. The second terms in (4) and (5) respectively represent likelihood and entropy regularization used in the literature to regularize GAN training (Warde-Farley & Bengio, 2017; Li & Turner, 2018). This implies for RKL-GAN these two regularization schemes are equivalent, and the solutions actually converge to an annealed surrogate instead of the target. Let

$$\mathcal{L}_\mu(\rho; \lambda) = \mathrm{KL}(\rho \parallel \mu) + \lambda\mathcal{R}_\mu(\rho) \tag{6}$$

be the general form of loss used in regularized training of RKL-GAN (which is essentially the annealed RKL-GAN), where λ is the regularization parameter and $\mathcal{R}_\mu(\rho)$ is the choice of regularizer. The following Corollary characterizes the target solutions for these two regularization schemes.

Corollary 2.2. Likelihood and entropy regularized RKL-GAN respectively converge to $\rho^*_{\text{lik}}(x) \propto \exp(-(\lambda+1)\psi(x))$ and $\rho^*_{\text{ent}}(x) \propto \exp(-(\lambda+1)^{-1}\psi(x))$.

2.4. Understanding annealed RKL-GAN in the continuous-time limit

While Theorem 2.1 links the annealed RKL-GAN to the damped Langevin system through their asymptotic solutions, better understanding of their relations can be established in the continuous-time limit. It suffices to look at the unannealed case, as generalization to the annealed case is straightforward.

It is simpler to consider RKL-GAN as a particle-based system. In the update of RKL-GAN, a particle x is propagated through $x_+ \leftarrow x + \epsilon\nabla \log \frac{\rho(x)}{\mu(x)}$, where ϵ is considered as a small time step. Now treat ϵ as an infinitesimal unit, then $\nabla \log \frac{\rho(x)}{\mu(x)}$ is the *velocity* of the particle. Recalling the fact the mass of ρ is conserved, we then have the continuity equation for the temporal evolution of model density ρ_t as

$$\partial_t \rho_t + \nabla \cdot (\rho_t \nabla \log \frac{\rho_t}{\mu}) = 0. \tag{7}$$

It can be readily recognized that this is the Fokker-Plank equation of the Langevin system in (1). With this in mind, we can derive the asymptotic convergence rate for the annealed RKL-GAN.

Theorem 2.3 (Convergence rate of continuous-time annealed RKL-GAN)**.** *If annealed RKL-GAN is solved exactly, we have*

$$KL(\rho_t \parallel \mu) \le \mathcal{O}(t^{-1}) + \delta_\beta, \tag{8}$$

where $\delta_\beta = \| \log \mu_\beta(x) - \log \mu(x) \|_{L_\infty(\mathcal{X})}$.

2.5. A gradient flow interpretation

It is helpful to further the understanding of the dynamics from a gradient flow perspective. Let $F(q) : \mathcal{P} \to \mathbb{R}$ be a functional on the space of probability measure, in our case, the anneal RKL. Let $\mathcal{H}$ be some function space, then functional gradient $\nabla_{\mathcal{H}} F(q) \in \mathcal{H}$ is defined as the one satisfying $F(q + f\,\mathrm{d}t) = F(q) + \langle \nabla_{\mathcal{H}} F(q), f\,\mathrm{d}t \rangle_{q\mathcal{H}}$ for all $f \in \mathcal{H}$, $\langle \cdot, \cdot \rangle_{q\mathcal{H}}$ is a q-weighted inner product and $\mathrm{d}t$ is taken to be infinitesimally small. For RKL, closed-form expressions for $\nabla_{\mathcal{H}} F(q)$ are available for particular choices of $\mathcal{H}$: $f_{L_2}(x) \triangleq \nabla \log \frac{\rho(x)}{\mu(x)}$ for $L_2(\rho)$ and $f_\kappa(x) \triangleq \mathbb{E}_{X' \sim \rho}[\mathcal{A}_\mu \otimes \kappa(X', x)]$ for a RKHS generated by kernel $\kappa(x, x')$, where $\mathcal{A}_\mu$ is the Stein operator for target measure μ, and $\otimes$ is the Kronecker product (Liu, 2017). In practice, these functional gradients are generally not computationally tractable, so their stochastic estimates are used to update model ρ. In Liu & Wang (2016), the authors leveraged the kernel trick to derive a tractable empirical estimator for $f_\kappa(x)$; while as in (annealed) RKL-GAN, $f_{L_2}(x)$ is estimated through a function approximator (*e.g.*, neural net) updated by stochastic gradient descent (the critic updates). The quality of the approximation affects the empirical convergence rate, as the asymptotic convergence rate is an upper bound on the improvement, and it only holds if the updates are exactly aligned with functional gradients. So the maximizing step can be understood as searching for the best descent direction for the generator update.

3. Variational Annealing

Inspired by the discussions above, we propose to anneal general GAN training via imposing a controlled likelihood regularization, and we refer to this procedure as *variational annealing* (VA). More specifically, we reformulate the adversarial game as [1]

$$\min_\rho \{ \max_D \{ V(\mu, \rho; D) \} - \lambda_t \mathbb{E}_{X' \sim \rho}[\log \mu(X')] \}. \tag{9}$$

To simplify the discussion, we assume that the training operates under the continuous-time limit, $i.e.$, the maximization step is solved accurately and instantly, and the evolution of ρ follows the functional gradient of the regularized loss. For RKL-GAN, in order for $\rho_t \to \mu$ we must have $\lambda_t \to 0$, otherwise it will result in an over- ($\lambda < 0$) or under- ($\lambda > 0$) dispersed sampler. Nevertheless, there are also situations for which one may want to keep λ_∞ nonzero, $e.g.$, to promote model robustness ($\lambda < 0$) or dismiss outliers ($\lambda > 0$). In general, we would expect other annealed GANs to behave qualitatively similar to annealed RKL-GAN.

From an implementation perspective, VA implicitly assumes the availability of both: 1) empirical data samples for estimation of $V(\mu, \rho; D)$, and 2) a tractable (unnormalized) distribution for computation of $\log \mu(X')$. Unfortunately, in practice we usually only have access to one of these two, *e.g.*, either a collection of data samples or a model likelihood. In either case, solving VA directly as defined in (9) is not possible. We now describe how to build surrogates to implement VA.

3.1. Replacing likelihood with empirical score function estimator

Generally speaking, density estimation is a challenging task on its own. Fortunately for practical purposes, the requirement of a tractable μ can be relaxed to its *score function*, given as $S_\mu(x) = \nabla_x \log \mu(x)$. To see this, let $G(z; \theta)$ be the parametrized generator, and via the chain rule we have the gradient from the regularizer as $\nabla_\theta \log \mu(G(z;\theta)) = \nabla_\theta G(z;\theta)^T S_\mu(G(z;\theta))$. For modern auto-differentiating learning platforms, we can use $\mathcal{R}_\mu(z) = G(z;\theta)^T \text{StopGrad}\{S_\mu(G(z;\theta))\}$ to replace the $\log \mu(G(z;\theta))$ term in the objective, where $\text{StopGrad}(\cdot)$ is the platform-specific API that stops the gradients computation during back-propagation.

There are a number of ways to estimate the score function

[1] The regularizer can also be the entropy term, just flip the sign of λ. For RKL-GAN, these two are equivalent.

from data (Hyvärinen, 2005; Gutmann & Hyvärinen, 2010; Li & Turner, 2018). Here we choose the *denoising auto-encoder* (DAE) estimator proposed in (Alain & Bengio, 2014) due to its simple construction and robustness to complex distributions. A mapping $\phi_\sigma(x) : \mathcal{X} \to \mathcal{X}$ is called a DAE if it minimizes the L_2 reconstruction loss of samples corrupted with *i.i.d.* Gaussian noise with intensity σ, *i.e.*

$$\phi_\sigma = \arg\min_\varphi \mathbb{E}_{X\sim\mu,\xi\sim\mathcal{N}(0,\sigma^2)}[\|X - \varphi(X+\xi)\|_2^2]. \quad (10)$$

Theorem 1 from Alain & Bengio (2014) states that $\nabla \log \mu = s_\sigma(x) + o(1)$ as $\sigma \to 0$, where $s_\sigma(x) = (\phi_\sigma(x) - x)/\sigma^2$ and we have dropped its dependence on μ to simplify our notations. In our experiments, we train a DAE modeled by a neural net optimized wrt (10) with a sufficiently small σ, and then plug that into the score function estimator $s_\sigma(x)$. Since the generator update does not involve back-propagation through the DAE, this scheme is very efficient.

3.2. Addressing unnormalized distributions via generative flows

When learning based on an unnormalized distribution $u(x)$, we can no longer implement the D-maximization step because of the lack of ground truth samples. However, for f-GANs, it is still possible to do so through a bridging measure ν that is both easy to sample and evaluate. Recall that for f-GANs the D-maximization is equivalent to the estimation of density ratio $r(x)$. We can estimate $r_1(x) = \log \frac{\rho(x)}{\nu(x)}$ with the regular f-GAN D-maximization step through samples from both ρ and ν, and evaluate $r_2(x) = \log \frac{\nu(x)}{\mu(x)} = \psi(x) - \gamma(x)$ directly assuming $\nu(x) \propto \exp(-\gamma(x))$. Since $r(x) = r_1(x) + r_2(x)$, we can now estimate $D(x)$ subsequent updates in f-GANs. To distinguish it from regular GANs which learn from empirical samples, we call it *unnormalized GAN* (U-GAN).

In our implementation, we approached bridging measure ν through generative flows, given in the form of

$$x = f_K \circ f_{K-1} \cdots f_2 \circ f_1(z), z \sim q(z), \quad (11)$$

where the $\{f_k\}$ are invertible mappings with easy-to-compute Jacobians and $q(z)$ is some simple noise distribution. Denoting $z_k = f_k(z_{k-1}), x = z_{K+1}$ and $z_0 = z$, then we can back-track the likelihood as $\log \nu(x) = \log q(z) - \sum_{k=1}^K \log|\det \frac{\partial f_k}{\partial z_{k-1}}|$. Generally speaking, ν should stay close to ρ to allow for more accurate estimation for $r_1(x)$. Therefore, we optimize the MLE objective $\mathbb{E}_{X'\sim\rho_t}[\log \nu_t(X')]$ wrt ν to track the evolving model distribution ρ_t.

We choose the *masked auto-regressive flow* (MAF) to build our bridging measure ν (Papamakarios et al., 2017). MAF is a special case of the *backward shift-scale* flow

$$z_{k+1} = t(z_{k+1}) + s(z_{k+1}) \odot z_k, \quad (12)$$

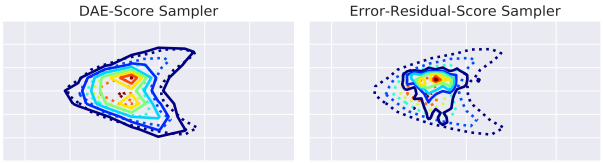

Figure 2. Comparison of different score function estimators. Solid and dotted contours respectively represent model and ground-truth densities.

where $t(z), s(z)$ respects the causal dependency of an auto-regressive flow so that Jacobians are triangular by design and can be computed in parallel. The backward design renders MAF fast to evaluate but slow to generate, because the forward pass (*e.g.*, $z \to x$) resembles a Gaussian elimination process. This feature is particularly suited for the VA f-GAN framework, because we only need to backpropagate the gradients for the (simpler) evaluation step: both in the G-maximization step and MLE step. In practice, MAF blocks are stacked and interleaved with permutation layers to construct flexible flows.

4. Related Work

We further discuss the proposed VA-GAN framework in the context of existing literature. In particular, we highlight how the VA procedure unifies popular regularization schemes, and point out a few mis-practices from prior arts.

4.1. Likelihood and entropy regularization

Regularizing GAN training with likelihood or entropy terms has been considered, with *denoising feature matching* (DFM-GAN) (Warde-Farley & Bengio, 2017) and *implicit gradient regularizer* (IGR-GAN) (Li & Turner, 2018) representing prominent examples from the respective categories. These works are mostly heuristically driven, in the hope that the imposed regularizer can stablize GAN training or encourage sample diversity. We rigorously show the equivalence of these two schemes under the annealed RKL-GAN framework, *e.g.*, positive entropy regularization is negative likelihood regularization, and answered the question of what distributions they are converging to. By appealing to the idea of annealing, we present a more principled scheme for regularized GAN training.

It is also worth noting that the misuse of likelihood estimators is prevalent in the literature (*e.g., amortized SVGD, PPGN, etc.*). For image models, in particular, the exponentiated reconstruction loss of an auto-encoder (AE), *e.g.*, $\tilde{\mu} \sim \exp(-\|x - \phi(x)\|_2^2)$, has been widely used as a substitute for the (unnormalized) likelihood (Wang & Liu, 2016). We identify two major caveats: (i) the training objective of the auto-encoder does not involve the normalizing constant (only evidence, no entropy), so it is not a valid estimator for $\mu(x)$; (ii), if ϕ is a DAE then $(\nabla_\theta x)^T(x - \phi(x))$ is a valid estimator for the parameter gradients, however taking $\nabla_\theta \|x - \phi(x)\|_2^2$ unnecessarily back-propagates gradients

through ϕ, which not only biases the gradients but also slows computation. As such, we surmise the results from existing likelihood regularized models are subject to aggregated mode collapsing and slower training. See Figure 2 for an intuitive illustration with the toy "banana" distribution.

4.2. Explicit versus implicit noise annealing

The notation of annealing has also been explored previously in the GAN literature. The most commonly adopted practice is to inject instance noise to the data and the generated samples, and gradually anneal the noise intensity to zero, so that GAN training may be improved (Arjovsky & Bottou, 2017; Mehrjou et al., 2017). We refer to this strategy as explicit annealing, and the model distribution ρ converges to $[\mu * \kappa](x)$, where $*$ is the convolution operator and κ denotes the noise density. Note for the explicit scheme the support of an annealed solution deviates from the ground truth, especially when the data resides on a low-dimensional submanifold. In contrast, the proposed variational annealing provides an implicit scheme that does not explicitly invoke the data corruption process [2] and always stays on the data manifold.

4.3. Generative modeling as diffusion process

Diffusion-based constructions are the most popular approaches towards drawing complex samples from an (unnormalized) distribution in the machine learning community. This is done with standard Langevin schemes (*e.g.*, PPGN, (Nguyen et al., 2017)), and more recently with deterministic schemes (*e.g.*, SVGD, MMD-GAN, OT-GAN, π-SGLD, (Chen et al., 2018)) and hybrid schemes (*e.g.*, SPOS, (Zhang et al., 2018)). In particular, *Stein variational gradient descent* (SVGD) also optimizes the RKL objective $\mathrm{KL}(p_G \parallel p_d)$ (Liu & Wang, 2016). It computes the functional gradient in a *Reproducing Kernel Hilbert Space* $\mathcal{H}$, which enjoys a tractable expression that only involves the kernel, the score function of μ and sampling from ρ. Our work unveils the connection between Langevin diffusion and GAN optimization, and endows GAN models with the ability to sample from unnormalized distributions.

4.4. Integrating generative flow with GANs

Attempts have been made to marry intractable GANs with tractable generative flows. A brute-force solution is considered in Grover et al. (2017) by directly combining a GAN loss with the MLE loss. It basically flips the likelihood and sample terms in our VA framework, and falls victim to the computational issues we discussed in Sec 3.2. Song et al. (2017) proposed a NICE approach, using generative flows as a proposal distribution and GAN training to learn a transition kernel to carry out Metropolis-Hasting sampling, *i.e.*, one still needs to simulate a (costly) Markov chain to obtain samples. The procedure we described can be considered as amortized training for generative flows, and contrastive training for GANs (Gutmann & Hyvärinen, 2010).

[2] Except when we train the DAE, but that is independent of the GAN training and with fixed noise input.

5. Experiments

We consider a wide range of synthetic and real-world tasks to validate our models experimentally, and benchmark them against competing baselines. All experiments are implemented with TensorFlow and run on a single NVIDIA TITANX GPU. Detailed modeling specifications are provided in the Supplementary Material (SM). Code for our experiments are available from `https://www.github.com/author_name/lgan`.

5.1. Variational annealing

Effect of likelihood regularization Our theory predicts stronger likelihood regularization (larger λ) can lead to mode collapse. This, unfortunately, might possibly encourage a better discrimination-based evaluation metrics such as *Inception Score* (IS) (Salimans et al., 2016), but other distribution-based metrics like *Fréchet Inception Distance* (FID) (Heusel et al., 2017) will be worse. To examine this hypothesis, we applied VA with fixed annealing, *i.e.*, constant λ throughout training, to the RKL-GAN and other popular GAN variants, namely the vanilla GAN (JSD-GAN) and Wasserstein-GAN (W-GAN), on the Cifar10 dataset [3]. We vary the regularization parameter λ and in Table 1 report the corresponding IS and FID scores for each model. Note that our goal is to qualitatively verify our hypotheses, not to necessarily achieve the state of the art. As expected, for positive annealing ($\lambda > 0$) all models deteriorate when an excessive likelihood penalty is enforced.

Positive & negative annealing While existing literature converged on the stabilizing positive annealing ($\lambda > 0$), our theory predicts the counter-intuitive negative annealing ($\lambda < 0$) actually promotes sample diversity, as it encourages exploration. To verify, we trained our model with negative annealing and in Table 1 summarize the scores and training dynamics wrt IS and FID using both annealing strategies. We observe VA with proper negative annealing yields competitive, if not better, results. We also present the trained generator samples for 32×32 Cifar10 and CelebA datasets using negative annealing in Figure 3 for visual inspection.

Dynamic annealing In order to match the target distribution, we should gradually anneal the regularization parameter λ to 0 towards the end of training. We call this practice dynamic annealing to distinguish it from the fixed annealing discussed above. In this experiment, we consider three different dynamic annealing schemes: positive monotonic annealing (PMA), negative monotonic annealing (NMA) and oscillatory annealing (OA). We visualize these three dynamic annealing schemes in Figure 4. PMA is appealing

[3] The DFM implementation from `https://github.com/pfnet-research/chainer-gan-lib` is used as our codebase.

Table 1. Quantative results for variational annealing on Cifar10.

	Static Annealing											Dynamic Annealing		
λ	-50	-10	-1	-0.1	-0.01	0	0.01	0.1	1	10	50	PMA	NMA	OA
Inception score (higher is considered better)														
RKL-GAN	6.24	6.37	6.35	6.33	6.35	6.25	6.24	6.35	6.41	6.19	6.17	6.56	**7.08**	7.05
JSD-GAN	6.68	6.84	6.64	6.35	6.61	6.29	6.67	6.30	6.93	6.48	6.22	6.80	**6.99**	6.96
W-GAN	5.77	6.14	6.29	6.86	6.62	5.93	6.22	6.54	5.95	6.00	6.00	6.95	**6.92**	6.91
FID score (lower is considered better)														
RKL-GAN	38.4	34.5	36.7	36.5	37.0	36.5	37.2	36.1	38.8	36.0	37.3	34.4	29.2	**28.9**
JSD-GAN	34.9	30.9	35.19	36.6	33.0	37.4	33.5	34.9	30.7	32.75	34.7	30.9	31.0	**29.1**
W-GAN	44.1	40.6	38.6	31.4	30.4	42.8	39.43	33.6	41.4	41.6	40.2	29.3	29.8	**29.0**

Figure 3. Cifar10 and CelebA generation results with negative static annealing.

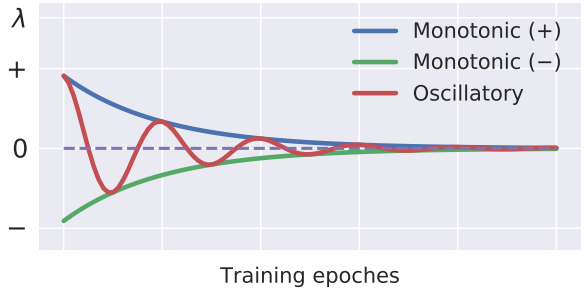

Figure 4. Different dynamic regularization schemes.

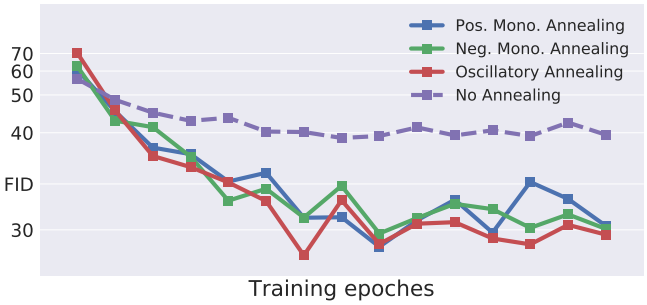

Figure 5. Learning dynamics with dynamic annealing.

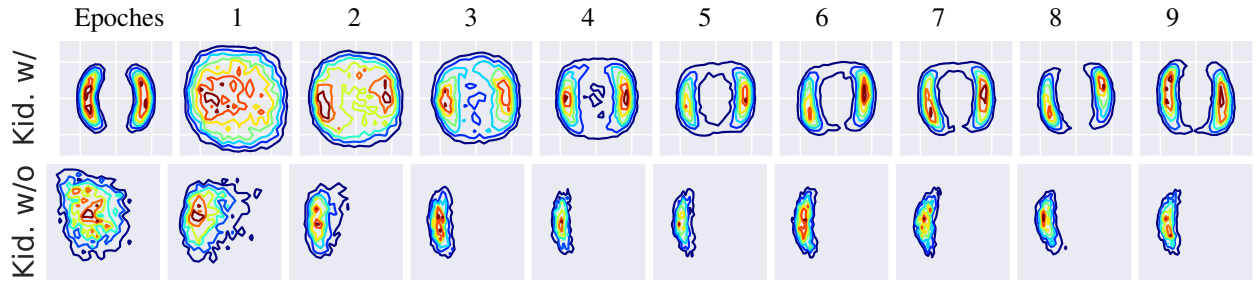

Figure 6. Learning from an unnormalized density to sample the kidney distribution. Top left: target distribution; bottom left: model distribution initialization; 'w/' with variational annealing; 'w/o' without annealing.

when the model distribution is away from the data manifold, as it attracts model distributions to the more plausible regions. NMA is especially helpful when the target distribution has isolated modes, as it disperses the samples to make the transition to missed modes easier. OA combines consolidation with exploration: using positive annealing to consolidate the patterns learned, and using negative annealing to explore new patterns.

In Figure 5 we show the evolution of the FID score for the above three annealing schemes using W-GAN; and the IS and FID scores produced are summarized alongside those from static annealing in Table 1. Additionally, we also directly estimate the likelihood using *annealed importance sampling* (AIS) (Neal, 2001; Wu et al., 2017) and report the results in Table 2. As observed, consistent with our hypothesis, dynamically annealed training significantly improved results. Armed with the strength from both positive and negative annealing, OA enjoyed even better performance evidenced by all three evaluation metrics considered here. We remark that the FID dynamics for OA appeared to be

Table 2. Likelihood evaluations on Cifar-10. (*nats per dimension*, lower is better)

	PMA	NMA	OA
RKL-GAN	2.41	2.55	**2.38**
JSD-GAN	2.27	2.43	**2.26**
W-GAN	2.96	2.58	**2.37**

Table 3. Results for Bayesian Logistic regression.

Model	U-GAN	SVGD	SGLD	MLE
heart	.48	.45	.51	.43
bcancer	.08	.08	.09	.09
german	.52	.50	.53	.50
sonar	.57	.55	.57	.52
ringnorm	.52	.51	.54	.51
splice	.39	.35	.40	.35
debiate	.51	.50	.52	.50
twonorm	.06	.06	.06	.06

more wiggly because of the exploration phases involved. Training without annealing clearly gets stuck at a bad local minimum and ceases to improve half-way thorough training. We have also repeated this experiment several times and the results are stable.

5.2. Sampling unnormalized distributions

Toy examples We applied the flow-augmented U-GAN to toy distributions to demonstrate its ability to learn to draw from unnormalized distributions, without empirical samples. In Figure 6, we tested its performance on a kidney distribution, a classic example with well-separated modes. In this experiment, we trained our model both with and without annealing the λ factor. Without the annealing, the mass of the model distribution quickly collapsed into one of the modes and trapped there. The annealed solution, as expected, firstly dispersed the mass all over the domain and then gradually condensed it towards the target. This highlights the necessity of annealing when dealing with multi-modal distributions. In Figure 7, we showed that U-GAN framework enables the free-form generator to learn more expressive distributions permitted by the generative flow. We handicapped the generative flow by dropping all its permutation layers, compromising its ability to capture the bifurcation structure of the target density; however, the U-GAN generator still correctly learned the target. More examples can be found in the SM.

5.3. Bayes Regression

Bayesian regression We next consider using U-GAN for Bayesian logistic regression. We assign a Gaussian prior $w \sim \mathcal{N}(0, \alpha^{-1})$ for the regression weights w, and a Gamma prior $\alpha \sim \Gamma(1, 10^{-2})$ for the precision parameter α. We collectively denote $\theta = (w, \log \alpha)$. The mini-batch estimator $\log \hat{\mu}(\theta|\mathcal{D}_n) = \log p_0(\theta) + \frac{n}{m} \sum_{k=1}^{m} \log p(x_{i_k}, y_i|\theta)$ is used for the posterior likelihood, where $\mathcal{D}_n = \{(x_i, y_i)\}_{i=1}^n$ is a size n data and $\{i_k\}_{k=1}^m$ is a size m mini-batch. The MAP estimate to initialize our model distribution. We report the test negative log likelihood in Table 3. We note that compared with SVGD, our UGAN results are more consistent with the ground-truth SGLD results.

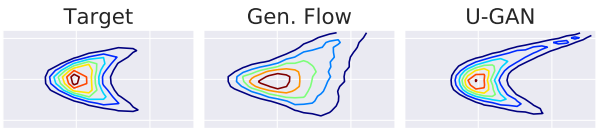

Figure 7. U-GAN training with restricted flow.

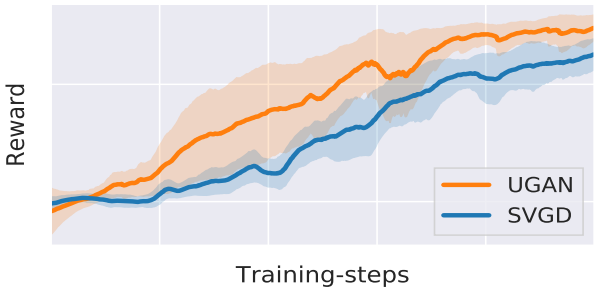

Figure 8. Soft-Q Learning results.

Energy-based deep reinforcement learning In our last experiment we consider an application in reinforcement learning (RL). It can be shown that, under the maximal entropy heuristics, the optimal stochastic policy $\pi(a|s)$ follows an unnormalized distribution given by $\exp(Q(a,s)/\tau)$, where (a, s) denotes the action and state pair and $Q(a, s)$ is known as the Q-function. To update the policy new actions are drawn from π to get reward from environment. In soft Q-learning, a sampling network is trained using amortized SVGD to match the predicted optimal policy distribution (Haarnoja et al., 2017). Here we replace SVGD with the proposed U-GAN, and compare the performance on the OpenAI *gym* tasks. Figure 8 shows a typical reward evolution for U-GAN based training.

6. Conclusion

We have considered the problem of likelihood-based regularization for generative adversarial nets. Via establishing a direct link to the Langevin diffusion, we are able to derive analytical results for the likelihood-regularized RKL-GAN, which unifies the practice of likelihood and entropy regularization. This also allows us to view dynamic likelihood regularization as implicit annealing for GANs, and that counter-intuitive negative annealing actually promotes the learning of multi-modal distributions. We also enhance GANs with generative flows to enable flexible learning from unnormalized distributions. We provide solid empirical evidence to validate our claims. In future work, we intend to further our study in this direction, investigating alternative strategies for likelihood-assisted GAN learning.

Acknowledgements

This research was supported in part by DARPA, DOE, NIH, ONR and NSF.

References

Alain, G. and Bengio, Y. What regularized auto-encoders learn from the data-generating distribution. *The Journal of Machine Learning Research*, 15(1):3563–3593, 2014.

Anonymous. Distributional concavity regularization for gans. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forums?id=SklEEnC5tQ. ICLR Anonymous submission.

Arjovsky, M. and Bottou, L. Towards principled methods for training generative adversarial networks. In *NIPS Workshop*. 2017.

Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein generative adversarial networks. In *ICML*, 2017.

Beaumont, M. A., Zhang, W., and Balding, D. J. Approximate bayesian computation in population genetics. *Genetics*, 162(4):2025–2035, 2002.

Che, T., Li, Y., Jacob, A. P., Bengio, Y., and Li, W. Mode regularized generative adversarial networks. 2017a.

Che, T., Li, Y., Zhang, R., Hjelm, R. D., Li, W., Song, Y., and Bengio, Y. Maximum-likelihood augmented discrete generative adversarial networks. *arXiv preprint arXiv:1702.07983*, 2017b.

Chen, C., Zhang, R., Wang, W., Li, B., and Chen, L. A unified particle-optimization framework for scalable bayesian sampling. In *UAI*, 2018.

Csiszár, I. Eine informationstheoretische Ungleichung und ihre Anwendung auf den Beweis der Ergodizitat von Markoffschen Ketten. *Publ. Math. Inst. Hungar. Acad*, 8: 85–108, 1963.

Didelot, X., Everitt, R. G., Johansen, A. M., Lawson, D. J., et al. Likelihood-free estimation of model evidence. *Bayesian analysis*, 6(1):49–76, 2011.

Diggle, P. J. and Gratton, R. J. Monte carlo methods of inference for implicit statistical models. *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 193–227, 1984.

Dinh, L., Sohl-Dickstein, J., and Bengio, S. Density estimation using real nvp. In *ICLR*, 2017.

Doob, J. L. *Stochastic processes*, volume 7. Wiley New York, 1953.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In *NIPS*, 2014.

Grover, A., Dhar, M., and Ermon, S. Flow-gan: Combining maximum likelihood and adversarial learning in generative models. *arXiv preprint arXiv:1705.08868*, 2017.

Gutmann, M. and Hyvärinen, A. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *AISTATS*, pp. 297–304, 2010.

Haarnoja, T., Tang, H., Abbeel, P., and Levine, S. Reinforcement learning with deep energy-based policies. In *ICML*, 2017.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NIPS*, 2017.

Hoffman, M. D., Blei, D. M., Wang, C., and Paisley, J. Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1):1303–1347, 2013.

Hyvärinen, A. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(Apr):695–709, 2005.

Jordan, R., Kinderlehrer, D., and Otto, F. The variational formulation of the Fokker–Planck equation. *SIAM journal on mathematical analysis*, 29(1):1–17, 1998.

Kingma, D. P., Salimans, T., and Welling, M. Improving variational inference with inverse autoregressive flow. In *NIPS*, 2016.

Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P. Optimization by simulated annealing. *science*, 220(4598):671–680, 1983.

Li, Y. and Turner, R. E. Gradient estimators for implicit models. In *ICLR*, 2018.

Li, Y., Swersky, K., and Zemel, R. Generative moment matching networks. In *ICML*, 2015.

Liu, J. S. *Monte Carlo strategies in scientific computing*. Springer Science & Business Media, 2008.

Liu, Q. Stein variational gradient descent as gradient flow. In *NIPS*, pp. 3118–3126, 2017.

Liu, Q. and Wang, D. Stein variational gradient descent: A general purpose bayesian inference algorithm. In *NIPS*, 2016.

Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I., and Frey, B. Adversarial autoencoders. *arXiv preprint arXiv:1511.05644*, 2015.

Mehrjou, A., Schölkopf, B., and Saremi, S. Annealed generative adversarial networks. *arXiv preprint arXiv:1705.07505*, 2017.

Mescheder, L., Nowozin, S., and Geiger, A. Adversarial variational Bayes: unifying variational autoencoders and generative adversarial networks. In *ICML*, 2017.

Mohamed, S. and Lakshminarayanan, B. Learning in implicit generative models. *arXiv preprint arXiv:1610.03483*, 2016.

Nachum, O., Norouzi, M., Xu, K., and Schuurmans, D. Bridging the gap between value and policy based reinforcement learning. In *NIPS*, pp. 2775–2785, 2017.

Neal, R. M. Annealed importance sampling. *Statistics and computing*, 11(2):125–139, 2001.

Nguyen, A., Clune, J., Bengio, Y., Dosovitskiy, A., and Yosinski, J. Plug & play generative networks: Conditional iterative generation of images in latent space. In *CVPR*, 2017.

Nowozin, S., Cseke, B., and Tomioka, R. f-GAN: Training generative neural samplers using variational divergence minimization. In *NIPS*, 2016.

Papamakarios, G., Murray, I., and Pavlakou, T. Masked autoregressive flow for density estimation. In *NIPS*, pp. 2335–2344, 2017.

Ranganath, R., Gerrish, S., and Blei, D. Black box variational inference. In *AISTATS*, 2014.

Reichl, L. E. *A modern course in statistical physics*. John Wiley & Sons, 2016.

Rosca, M., Lakshminarayanan, B., Warde-Farley, D., and Mohamed, S. Variational approaches for auto-encoding generative adversarial networks. *arXiv preprint arXiv:1706.04987*, 2017.

Rosca, M., Lakshminarayanan, B., and Mohamed, S. Distribution matching in variational inference. *arXiv preprint arXiv:1802.06847*, 2018.

Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., and Chen, X. Improved techniques for training GANs. In *NIPS*, 2016.

Song, J., Zhao, S., and Ermon, S. A-NICE-MC: Adversarial training for MCMC. In *NIPS*, pp. 5140–5150, 2017.

Tolstikhin, I., Bousquet, O., Gelly, S., and Schoelkopf, B. Wasserstein auto-encoders. In *ICLR*, 2018.

Wang, D. and Liu, Q. Learning to draw samples: With application to amortized mle for generative adversarial learning. *arXiv:1611.01722*, 2016.

Warde-Farley, D. and Bengio, Y. Improving generative adversarial networks with denoising feature matching. In *ICLR*, 2017.

Welling, M. and Teh, Y. W. Bayesian learning via stochastic gradient langevin dynamics. In *ICML*, pp. 681–688, 2011.

Wu, Y., Burda, Y., Salakhutdinov, R., and Grosse, R. On the quantitative analysis of decoder-based generative models. In *ICLR*. 2017.

Yeh, R., Chen, C., Lim, T. Y., Hasegawa-Johnson, M., and Do, M. N. Semantic image inpainting with perceptual and contextual losses. *arXiv preprint arXiv:1607.07539*, 2016.

Zhang, J., Zhang, R., and Chen, C. Stochastic particle-optimization sampling and the non-asymptotic convergence theory. *arXiv preprint arXiv:1809.01293*, 2018.