Journal of Machine Learning **ISSN**: 2790-2048(e), 2790-203X(p)

A Mathematical Framework for Learning Probability Distributions

Hongkang Yang *[1]

[1]Program in Applied and Computational Mathematics, Princeton University, Princeton, NJ 08544, USA.

Abstract. The modeling of probability distributions, specifically generative modeling and density estimation, has become an immensely popular subject in recent years by virtue of its outstanding performance on sophisticated data such as images and texts. Nevertheless, a theoretical understanding of its success is still incomplete. One mystery is the paradox between memorization and generalization: In theory, the model is trained to be exactly the same as the empirical distribution of the finite samples, whereas in practice, the trained model can generate new samples or estimate the likelihood of unseen samples. Likewise, the overwhelming diversity of distribution learning models calls for a unified perspective on this subject. This paper provides a mathematical framework such that all the well-known models can be derived based on simple principles. To demonstrate its efficacy, we present a survey of our results on the approximation error, training error and generalization error of these models, which can all be established based on this framework. In particular, the aforementioned paradox is resolved by proving that these models enjoy implicit regularization during training, so that the generalization error at early-stopping avoids the curse of dimensionality. Furthermore, we provide some new results on landscape analysis and the mode collapse phenomenon.

Keywords:
Generative modeling,
Density estimation,
Generalization error,
Memorization,
Implicit regularization.

Article Info.:
Volume: 1
Number: 4
Pages: 373 - 431
Date: December/2022
doi.org/10.4208/jml.221202

Article History:
Received: 02/12/2022
Accepted: 22/12/2022

Communicated by:
Weinan E

1 Introduction

The popularity of machine learning models in recent years is largely attributable to their remarkable versatility in solving highly diverse tasks with good generalization power. Underlying this diversity is the ability of the models to learn various mathematical objects such as functions, probability distributions, dynamical systems, actions and policies, and often a sophisticated architecture or training scheme is a composition of these modules. Besides fitting functions, learning probability distributions is arguably the most widely-adopted task and constitutes a great portion of the field of unsupervised learning. Its applications range from the classical density estimation [115, 119] which is important for scientific computing [13, 70, 118], to generative modeling with superb performance in image synthesis and text composition [15, 16, 94, 96], and also to pretraining tasks such as masked reconstruction that are crucial for large-scale models [16, 25, 28].

Despite the impressive performance of machine learning models in learning probability measures, this subject is less understood than the learning of functions or supervised learning. Specifically, there are several mysteries:

*Corresponding author. hongkang@princeton.edu

1. Unified framework. There are numerous types of models for representing and estimating distributions, making it difficult to gain a unified perspective for model design and comparison. One traditional categorization includes five model classes: the generative adversarial networks (GAN) [7,39], variational autoencoders (VAE) [64], normalizing flows (NF) [95,112], autoregressive models [85,92], and diffusion models [104,107]. Within each class, there are further variations that complicate the picture, such as the choice of integral probability metrics for GANs and the choice of architectures for normalizing flows that enable likelihood computations. Ideally, instead of a phenomenological categorization, one would prefer a simple theoretical framework that can derive all these models in a straightforward manner based on a few principles.

2. Memorization and curse of dimensionality. Perhaps the greatest difference between learning functions and learning probability distributions is that, conceptually, the solution of the latter problem must be trivial. On one hand, since the target distribution P_* can be arbitrarily complicated, any useful model must satisfy the property of universal convergence, namely the modeled distribution can be trained to converge to any given distribution (e.g. Section 5 will show that this property holds for several models). On the other hand, the target P_* is unknown in practice and only a finite sample set $\{\mathbf{x}_i\}_{i=1}^n$ is given (with the empirical distribution denoted by $P_*^{(n)}$). As a result, the modeled distribution P_t can only be trained with $P_*^{(n)}$ and inevitably exhibits memorization, i.e.

$$\lim_{t\to\infty} P_t = P_*^{(n)}.$$

Hence, training results in a trivial solution and does not provide us with anything beyond the samples we already have. This is different from regression where the global minimizer (interpolating solution) can still generalize well [36].

One related problem is the curse of dimensionality, which becomes more severe when estimating distributions instead of functions. In general, the distance between the hidden target and the empirical distribution scales badly with dimension d: For any absolutely continuous P_* and any $\delta > 0$ [120]

$$W_2\big(P_*, P_*^{(n)}\big) \gtrsim n^{-\frac{1}{d-\delta}},$$

where W_2 is the Wasserstein metric. This slow convergence sets a limit on the performance of all possible models: for instance, the following worst-case lower bound [103]:

$$\inf_A \sup_{P_*} \mathbb{E}_{\{X_i\}} \left[W_2^2\big(P_*, A(\{X_i\}_{i=1}^n)\big)\right]^{\frac{1}{2}} \gtrsim n^{-\frac{1}{d}},$$

where P_* is any distribution supported on $[0,1]^d$ and A is any estimator, i.e. a mapping from every n sample set $\{X_i\}_{i=1}^n \sim P_*$ to an estimated distribution $A(\{X_i\})$. Hence, to achieve a generalization error of ϵ, an astronomical sample size $\Omega(\epsilon^d)$ could be necessary in high dimensions.

These theoretical difficulties form a seeming paradox with the empirical success of distribution learning models, for instance, models that can generate novel and highly-realistic images [15,62,94] and texts [16,92].

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

3. Training and mode collapse. The training of distribution learning models is known to be more delicate than training supervised learning models, and exhibits several novel forms of failures. For instance, for the GAN model, one common issue is mode collapse [66,77,90,100], when a positive amount of mass in P_t becomes concentrated at a single point, e.g. an image generator could consistently output the same image. Another issue is mode dropping [129], when P_t fails to cover some of the modes of P_*. In addition, training may suffer from oscillation and divergence [19,91]. These problems are the main obstacle to global convergence, but the underlying mechanism remains largely obscure.

The goal of this paper is to provide some insights into these mysteries from a mathematical point of view. Specifically,

1. We establish a unified theoretical framework from which all the major distribution learning models can be derived. The diversity of these models is largely determined by two simple factors, the distribution representation and loss type. This formulation greatly facilitates our analysis of the approximation error, training error and generalization error of these models.

2. We survey our previous results on generalization error, and resolve the paradox between memorization and generalization. As illustrated in Fig. 1.1, despite that the model eventually converges to the global minimizer, which is the memorization solution, the training trajectory comes very close to the hidden target distribution. With early-stopping or regularized loss, the generalization error scales as
$$W_2(P_*, P_t) \quad \text{or} \quad \mathrm{KL}(P_* \| P_t) \lesssim n^{-\alpha}$$
for some constant $\alpha > 0$ instead of dimension-dependent terms such as α/d. Thereby, the model escapes from the curse of dimensionality.

3. We discuss our previous results on the rates of global convergence for some of the models. For the other models, we establish new results on landscape and critical points, and identify two mechanisms that can lead to mode collapse.

This paper is structured as follows. Section 2 presents a sketch of the popular distribution learning models. Section 3 introduces our theoretical framework and the derivations

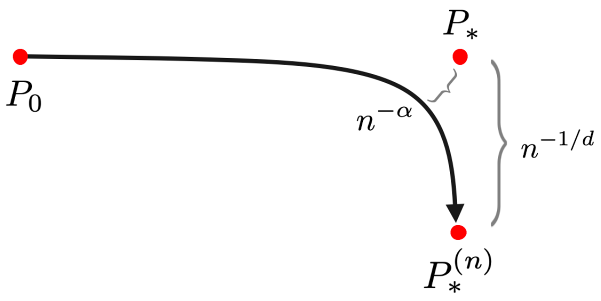

Figure 1.1: Generalization error during training.

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

of the models. Section 4 establishes the universal approximation theorems. Section 5 analyzes the memorization phenomenon. Section 6 discusses the generalization error of several representative distribution learning models. Section 7 analyzes the training process, loss landscape and mode collapse. All proofs are contained in Section 9. Section 8 concludes this paper with discussion on the remaining mysteries.

Here is a list of related works.

Mathematical framework: A framework for supervised learning has been proposed by [32, 35] with focus on the function representations, namely function spaces that can be discretized into neural networks. This framework is helpful for the analysis of supervised learning models, in particular, the estimation of generalization errors [31,33,36] that avoid the curse of dimensionality, and the determination of global convergence [24, 97]. Similarly, our framework for distribution learning emphasizes the function representation, as well as the new factor of distribution representation, and bound the generalization error through analogous arguments. Meanwhile, there are frameworks that characterizes distribution learning from other perspectives, for instance, statistical inference [54, 108], graphical models and rewards [12, 130], energy functions [134] and biological neurons [88].

Generalization ability: The section on generalization reviews our previous results on the generalization error estimates for potential-based model [126], GAN [127] and normalizing flow with stochastic interpolants [125]. The mechanism is that function representations defined by integral transforms or expectations [34,35] enjoy small Rademacher complexity and thus escape from the curse of dimensionality. Earlier works [31,33,36,37] used this mechanism to bound the generalization error of supervised learning models. Our analysis combines this mechanism with the training process to show that early-stopping solutions generalize well, and is related to concepts from supervised learning literature such as the frequency principle [123, 124] and slow deterioration [76].

Training and convergence: The additional factor of distribution representation further complicates the loss landscape, and makes training more difficult to analyze, especially for the class of "free generators" that will be discussed later. The model that attracted the most attention was GAN, and convergence has only been established in simplified settings [10, 38, 79, 121, 127] or for local saddle points [47, 73, 81]. In practice, GAN training is known to suffer from failures such as mode collapse and divergence [9,20,79]. Despite that these issues can be fixed using regularizations and tricks [44, 50, 66, 77, 90], the mechanism underlying this training instability is not well understood.

2 Model Overview

This section offers a quick sketch of the prominent models for learning probability distributions, while their derivations will be presented in Section 3. These models are commonly grouped into five categories: the generative adversarial networks (GAN), autoregressive models, variational autoencoders (VAE), normalizing flows (NF), and diffusion models. We assume access to samples drawn from the target distribution P_*, and the task

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

is to train a model to be able to generate more samples from the distribution (generative modeling) or compute its density function (density estimation).

1. GAN. The generative adversarial networks [39] model a distribution by transport

$$P = \mathrm{law}(X), \quad X = G(Z), \quad Z \sim \mathbb{P}. \tag{2.1}$$

The map $G : \mathbb{R}^d \to \mathbb{R}^d$ is known as the generator and $\mathbb{P}$ is a base distribution that is easy to sample (e.g. unit Gaussian). To solve for a generator such that $P = P_*$, the earliest GAN model considers the following optimization problem [39]:

$$\min_G \max_D \int \log\left(\frac{e^{D(\mathbf{x})}}{1+e^{D(\mathbf{x})}}\right) dP_*(\mathbf{x}) + \int \log\left(\frac{1}{1+e^{D(G(\mathbf{x}))}}\right) d\mathbb{P}(\mathbf{x}),$$

where $D : \mathbb{R}^d \to \mathbb{R}$ is known as the discriminator and this type of min-max losses are known as the adversarial loss. A well-known variant is the WGAN [7] defined by

$$\min_G \max_{\|\theta\|_\infty \leq 1} \int D_\theta(\mathbf{x}) dP_*(\mathbf{x}) - \int D_\theta\big(G(\mathbf{x})\big) d\mathbb{P}(\mathbf{x}),$$

where the discriminator D_θ is a neural network with parameter θ, which is bounded in l^∞ norm. For the other variants, a survey on the GAN models is given by [43].

2. VAE. The variational autoencoder proposed by [64] uses a randomized generator and its approximate inverse, known as the decoder and encoder, and we denote them by the conditional distributions $P(\cdot|\mathbf{z})$ and $Q(\cdot|\mathbf{x})$. Similar to (2.1), the distribution is modeled by $P = \int P(\cdot|\mathbf{z}) d\mathbb{P}(\mathbf{z})$ and can be sampled by $X \sim P(\cdot|Z), Z \sim \mathbb{P}$. VAE considers the following optimization problem:

$$\min_{P(\cdot|\mathbf{z})} \min_{Q(\cdot|\mathbf{x})} \iint -\log P(\mathbf{x}|\mathbf{z}) dQ(\mathbf{z}|\mathbf{x}) + \mathrm{KL}\big(Q(\cdot|\mathbf{x}) \big\| \mathbb{P}\big) dP_*(\mathbf{x}),$$

where KL is the Kullback–Leibler divergence. To simplify computation, $\mathbb{P}$ is usually set to be the unit Gaussian $\mathcal{N}$, and the decoder $P(\cdot|\mathbf{z})$ and encoder $Q(\cdot|\mathbf{x})$ are parameterized as diagonal Gaussians [64]. For instance, consider

$$P(\cdot|\mathbf{z}) = \mathcal{N}\big(G(\mathbf{z}), s^2 I\big), \quad Q(\cdot|\mathbf{x}) = \mathcal{N}\big(F(\mathbf{x}), v^2 I\big),$$

where G, F are parameterized functions and $s, v > 0$ are scalars. Then, we have

$$\begin{aligned}
&\int -\log P(\mathbf{x}|\mathbf{z}) dQ(\mathbf{z}|\mathbf{x}) \\
&= \int -\log P\big(\mathbf{x}\big|F(\mathbf{x}) + v\omega\big) d\mathcal{N}(\omega) \\
&= \int \frac{\big\|\mathbf{x} - G\big(F(\mathbf{x}) + v\omega\big)\big\|^2}{2s^2} d\mathcal{N}(\omega) + \frac{d}{2}\log(2\pi s^2), \\
&\mathrm{KL}\big(Q(\cdot|\mathbf{x}) \big\| \mathbb{P}\big) = \frac{\|F(\mathbf{x})\|^2}{2} + \frac{d}{2}\big(v^2 - \log v^2 - 1\big).
\end{aligned}$$

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

Up to constant, the VAE loss becomes

$$\min_{G,s}\min_{F,v}\iint \frac{\left\|\mathbf{x}-G\big(F(\mathbf{x})+v\omega\big)\right\|^2}{2s^2}d\mathcal{N}(\omega)+\frac{\|F(\mathbf{x})\|^2}{2}dP_*(\mathbf{x})+d(\log s-\log v)+\frac{d}{2}v^2.$$

3. Autoregressive. Consider sequential data $\mathbf{X}=[\mathbf{x}_1,\ldots,\mathbf{x}_l]$ such as text and audio. The autoregressive models represent a distribution P through factorization $P(\mathbf{X})=\prod_{i=1}^{l}P(\mathbf{x}_i|\mathbf{x}_{<i})$, and can be sampled by sampling iteratively from $X_i\sim P(X_i|X_{<i})$. These models minimize the loss

$$-\int\sum_{i=1}^{l}\log P(\mathbf{x}_i|\mathbf{x}_{<i})dP_*(\mathbf{X}).$$

There are several approaches to the parametrization of conditional distributions $P(\mathbf{x}_i|\mathbf{x}_{<i})$, depending on how to process the variable-length input $\mathbf{x}_{<i}$. The common options are the transformer networks [92, 116], recurrent networks [87], autoregressive networks [68, 87] and causal convolution [86]. For instance, consider Gaussian distributions parameterized by a recurrent network

$$P(\cdot|\mathbf{x}_{<i})=\mathcal{N}\big(m(\mathbf{h}_i),s^2(\mathbf{h}_i)I\big),\quad \mathbf{h}_i=f(\mathbf{h}_{i-1},\mathbf{x}_{i-1}),$$

where m,s,f are parameterized functions, and $\mathbf{h}$ is the hidden feature. Since

$$-\log P(\mathbf{x}_i|\mathbf{x}_{<i})=\frac{\|\mathbf{x}_i-m(\mathbf{h}_i)\|^2}{2s(\mathbf{h}_i)^2}+\frac{d}{2}\log\left(2\pi s(\mathbf{h}_i)^2\right)$$

the loss is equal, up to constant, to

$$\min_{m,s,f,\mathbf{h}_1}\int\sum_{i=1}^{l}\frac{\|\mathbf{x}_i-m(\mathbf{h}_i)\|^2}{2s(\mathbf{h}_i)^2}+d\log s(\mathbf{h}_i)\;dP_*(\mathbf{X}).$$

4. NF. The normalizing flows proposed by [111, 112] use a generator G (2.1) similar to GAN and VAE. The optimization problem is given by

$$\min_{G}-\int\log\det\nabla G^{-1}(\mathbf{x})+\log\mathbb{P}\big(G^{-1}(\mathbf{x})\big)dP_*(\mathbf{x}),$$

where $\mathbb{P}=\mathcal{N}$ is set to be the unit Gaussian and $\det\nabla G^{-1}$ is the Jacobian determinant of G^{-1}. To enable the calculation of these terms, the earliest approach [95, 111, 112] considers only the inverse $F=G^{-1}$ and models it by a concatenation of simple maps $F=F_1\circ\cdots\circ F_T$ such that each $\det\nabla F_\tau$ is easy to compute. The modeled distribution (2.1) cannot be sampled, but can serve as a density estimator, with the density given by

$$P(\mathbf{x}_T)=\mathcal{N}(\mathbf{x}_0)\prod_{\tau=1}^{T}\det\nabla F_\tau(\mathbf{x}_{\tau-1}),\quad \mathbf{x}_{\tau-1}:=F_\tau\circ\cdots\circ F_T(\mathbf{x}_T).$$

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

Up to constant, the loss becomes

$$\min_{F_1,\dots,F_T} \int \frac{\|\mathbf{x}_0\|^2}{2} + \sum_{\tau=1}^{T} -\log\det \nabla F_\tau(\mathbf{x}_{\tau-1})\, dP_*(\mathbf{x}_T).$$

A later approach [29,55,63,89] models the generator by $G = G_T \circ \cdots \circ G_1$ such that each G_τ is designed to be easily invertible with ∇G_τ being a triangular matrix. Then, the loss becomes

$$\min_{G_1,\dots,G_T} \int \frac{\|\mathbf{x}_0\|^2}{2} + \sum_{\tau=1}^{T} \operatorname{Tr}\big[\log \nabla G_\tau(\mathbf{x}_{\tau-1})\big] dP_*(\mathbf{x}_T), \quad \mathbf{x}_{\tau-1} = G_\tau^{-1} \circ \cdots \circ G_T^{-1}(\mathbf{x}_T).$$

Another approach [21,30,40] defines G as a continuous-time flow, i.e. solution to an ordinary differential equation (ODE)

$$G = G_0^1, \quad G_s^\tau(\mathbf{x}_s) = \mathbf{x}_\tau, \quad \frac{d}{d\tau}\mathbf{x}_\tau = V(\mathbf{x}_\tau, \tau) \tag{2.2}$$

for some time-dependent velocity field V. Then, the loss becomes

$$\min_V \int \frac{\|\mathbf{x}_0\|^2}{2} + \int_0^1 \operatorname{Tr}\big[\nabla_{\mathbf{x}} V(\mathbf{x}_\tau, \tau)\big] d\tau dP_*(\mathbf{x}_1), \quad \mathbf{x}_\tau = \big(G_\tau^1\big)^{-1}(\mathbf{x}_1).$$

A survey on normalizing flows is given by [65].

5. Monge-Ampère flow. A model that is closely related to the normalizing flows is the Monge-Ampère flow [131]. It is parameterized by a time-dependent potential function $\phi_\tau, \tau \in [0,1]$, and defines a generator by the ODE

$$G = G_0^1, \quad G_s^\tau(\mathbf{x}_s) = \mathbf{x}_\tau, \quad \frac{d}{d\tau}\mathbf{x}_\tau = \nabla\phi_\tau(\mathbf{x}_\tau)$$

such that the flow is driven by a gradient field. The model minimizes the following loss:

$$\min_\phi \int \frac{\|\mathbf{x}_0\|^2}{2} + \int_0^1 \Delta\phi_\tau(\mathbf{x}_\tau) d\tau dP_*(\mathbf{x}_1), \quad \mathbf{x}_\tau = \big(G_\tau^1\big)^{-1}(\mathbf{x}_1),$$

where $\Delta\phi = \sum_{i=1}^d \partial_i^2 \phi$ is the Laplacian.

6. Diffusion. The diffusion models [49,104,106] define the generator by a reverse-time SDE (stochastic differential equation)

$$G(X_T) = X_0, \quad X_T \sim \mathbb{P}, \quad dX_\tau = -\frac{\beta_\tau}{2}\big(X_\tau + 2\mathbf{s}(X_\tau, \tau)\big)d\tau + \sqrt{\beta_\tau} d\overline{W}_\tau,$$

where $\mathbf{s} : \mathbb{R}^d \times [0,T] \to \mathbb{R}^d$ is a time-dependent velocity field known as the score function [56], $\beta_\tau > 0$ is some noise scale, and $\overline{W}_\tau$ is a reverse-time Wiener process. The modeled

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

distribution is sampled by solving this SDE backwards in time from T to 0. The score function $\mathbf{s}$ is learned from the optimization problem

$$\min_{\mathbf{s}} \int_0^T \frac{\lambda_\tau}{2} \iint \Big\| \mathbf{s}\Big(e^{-\frac{1}{2}\int_0^\tau \beta_s ds}\mathbf{x}_0 + \sqrt{1-e^{-\int_0^\tau \beta_s ds}}\,\omega, \tau\Big) \\ + \frac{\omega}{\sqrt{1-e^{-\int_0^\tau \beta_s ds}}} \Big\|^2 d\mathcal{N}(\omega) dP_*(\mathbf{x}_0) d\tau,$$

where $\mathcal{N}$ is the unit Gaussian and $\lambda_\tau > 0$ is any weight. Besides the reverse-time SDE, another way to sample from the model is to solve the following reverse-time ODE, which yields the same distribution [107]

$$G(X_T) = X_0, \quad X_T \sim \mathbb{P}, \quad \frac{d}{d\tau}X_\tau = -\frac{\beta_\tau}{2}\big(X_\tau + \mathbf{s}(X_\tau, \tau)\big)d\tau.$$

A survey on diffusion models is given by [128].

7. NF interpolant. Finally, we introduce a model called normalizing flow with stochastic interpolants [2, 75]. It is analogous to the diffusion models, and yet is conceptually simpler. This model learns a velocity field $V : \mathbb{R}^d \times [0,1] \to \mathbb{R}^d$ from the optimization problem

$$\min_V \int_0^1 \iint \big\| V\big((1-\tau)\mathbf{x}_0 + \tau \mathbf{x}_1, \tau\big) - (\mathbf{x}_1 - \mathbf{x}_0) \big\|^2 d\mathbb{P}(\mathbf{x}_0) dP_*(\mathbf{x}_1) d\tau.$$

Then, the generator is defined through the ODE (2.2) and the modeled distribution is sampled by (2.1).

3 Framework

Previously, a mathematical framework for supervised learning was proposed by [35], which was effective for estimating the approximation and generalization errors of supervised learning models [31, 33, 34]. In particular, it helped to understand how neural network-based models manage to avoid the curse of dimensionality. The framework characterizes the models by four factors: the function representation (abstract function spaces built from integral transformations), the loss, the training scheme, and the discretization (e.g. how the continuous representations are discretized into neural networks with finite neurons).

This section presents a similar framework that unifies models for learning probability distributions. We focus on two factors: the distribution representation, which is a new factor and determines how distributions are parameterized by abstract functions, and the loss type, which specifies which metric or topology is imposed upon the distributions. A sketch of the categorization is given in Table 3.1. We show that the diverse families of distribution learning models can be simultaneously derived from this framework. The theoretical results in the latter sections, in particular the generalization error estimates, are also built upon this mathematical foundation.

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

Table 3.1: Categorization of distribution learning models based on distribution representation (row) and loss type (column) with the representative models. See Section 3.4 for a detailed description. Our theoretical results will focus on the marked categories.

	Density	Expectation	Regression
Potential	bias potential model	feasible	unknown
Free generator	NF, VAE, autoregressive	GAN	unknown
Fixed generator	upper bound	feasible	diffusion, NF interpolant, OT

3.1 Background

The basic task is to estimate a probability distribution given i.i.d. samples $\{\mathbf{x}_i\}_{i=1}^n$. We denote this unknown target distribution by P_* and the empirical distribution by

$$P_*^{(n)} = \frac{1}{n}\sum_{i=1}^{n}\delta_{\mathbf{x}_i}.$$

The underlying space is assumed to be Euclidean $\mathbb{R}^d$. To estimate a distribution may have several meanings depending on its usage: e.g. to obtain a random variable $X \sim P_*$, estimate the density function $P_*(\mathbf{x})$, or compute expectations $\int f dP_*$. The first task is known as generative modeling and the second as density estimation; these two problems are the focus of this paper, while the third task can be solved by them.

There are two general approaches to modeling a distribution, which can be figuratively termed as "vertical" and "horizontal", and an illustration is given by Fig. 3.1. Given a base distribution $\mathbb{P}$, the vertical approach reweighs the density of $\mathbb{P}$ to approximate the density of the target P_*, while the horizontal approach transports the mass of $\mathbb{P}$ towards the location of P_*. When a modeled distribution P is obtained and a distance $d(P, P_*)$ is needed to compute either the training loss or test error, the vertical approach measures the difference between the densities of P and P_* over each location, and is exemplified by the

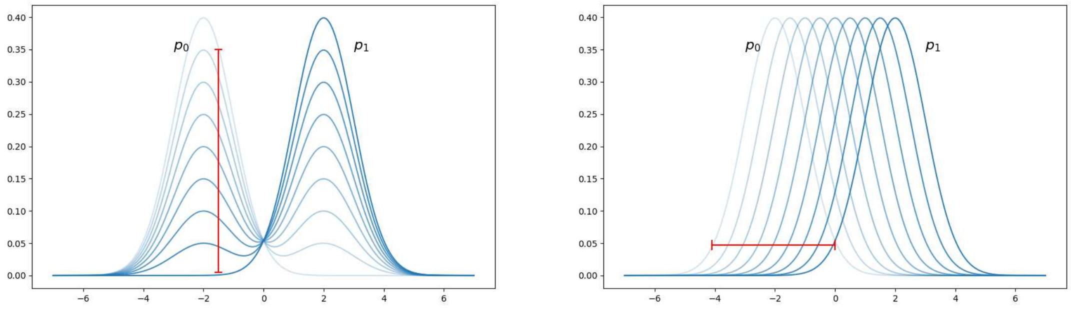

Figure 3.1: The vertical and horizontal perspectives on probability distributions. Left: the distribution $P_0 = \mathcal{N}(-2,1)$ is transformed to $P_1 = \mathcal{N}(2,1)$ by reweighing and the distance $d(P_0, P_1)$ is measured by the difference between densities. Right: P_0 is transformed to P_1 by transport, and the distance is measure by the displacement of mass.

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

KL-divergence

$$\mathrm{KL}(P_*\|P) := \int \log \frac{P_*(\mathbf{x})}{P(\mathbf{x})} dP_*(\mathbf{x}), \tag{3.1}$$

while the horizontal approach measures the distance between the "particles" of P and P_*, and is exemplified by the 2-Wasserstein metric [60, 117]

$$W_2(P,P_*) := \inf_{\pi} \left(\int \|\mathbf{x}_0 - \mathbf{x}_1\|^2 d\pi(\mathbf{x}_0,\mathbf{x}_1) \right)^{\frac{1}{2}}, \tag{3.2}$$

where π is any coupling between P, P_* (i.e. a joint distribution in $\mathbb{R}^d \times \mathbb{R}^d$ whose marginal distributions are P, P_*). We will see that the vertical and horizontal approaches largely determine the distribution representation and loss type.

Finally, consider the operator law

$$P = \mathrm{law}(X),$$

which maps a random variable X to its distribution P. Similarly, given a random path $\{X_\tau, \tau \in [0,T]\}$, we obtain a path $P_\tau = \mathrm{law}(X_\tau)$ in the distribution space. In general, there can be infinitely many random variables that are mapped to the same distribution, e.g. let P and $\mathbb{P}$ be uniform over $[0,1]$, for any $k \in \mathbb{N}$, we can define the random variable $X_k = kZ\mathrm{mod}1$ with $Z \sim \mathbb{P}$, which all satisfy $P = \mathrm{law}(X_k)$. One drawback of this non-uniqueness is that, for many generative models, there can be plenty of global minima, which make the loss landscape non-convex and may lead to training failures, as we will show that this is inherent in mode collapse. One benefit is that, if the task is to learn some time-dependent distribution P_τ, one can select from the infinitely many possible random paths X_τ the one that is the easiest to compute, and therefore define a convenient loss.

For the notations, for any measurable subset Ω of $\mathbb{R}^d$, denote by $\mathcal{P}(\Omega)$ the space of probability measures over Ω, $\mathcal{P}_2(\Omega)$ the subspace of measures with finite second moments, and $\mathcal{P}_{ac}(\Omega)$ the subspace of absolutely continuous measures (i.e. have density functions). Denote by $\mathrm{sprt}P$ the support of a distribution. Given any two measures m_0, m_1, we denote by $m_0 \times m_1$ the product measure over the product space. We denote by t the training time and by τ the time that parameterizes flows.

3.2 Distribution representation

Since machine learning models at the basic level are good at learning functions, the common approach to learning distributions is to parameterize distributions by functions. There are three common approaches:

1. **Potential function.** Given any base distribution $\mathbb{P}$, define the modeled distribution P by

$$P = \frac{1}{Z} e^{-V} \mathbb{P}, \quad Z = \int e^{-V} d\mathbb{P}, \tag{3.3}$$

where V is a potential function and Z is for normalization. This parametrization is sometimes known as the Boltzmann distribution or exponential family.

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

2. Free generator. Given any measurable function $G : \mathbb{R}^d \to \mathbb{R}^d$, define the modeled distribution P by

$$P = \text{law}(X), \quad X = G(Z), \quad Z \sim \mathbb{P},$$

P is known as the transported measure or pushforward measure and denoted by $P = G\#\mathbb{P}$, while G is called the generator or transport map. Equivalently, P is defined as the measure that satisfies

$$P(A) = \mathbb{P}\big(G^{-1}(A)\big) \tag{3.4}$$

for all measurable sets A.

The name "free generator" is used to emphasize that the task only specifies the target distribution P_* to estimate, and we are free to choose any generator from the possibly infinite set of solutions $\{G \,|\, P_* = G\#\mathbb{P}\}$.

There are several common extensions to the generator. First, G can be modeled as a random function, such that $G(\mathbf{z}) \sim P(\cdot|\mathbf{z})$ for some conditional distribution $P(\cdot|\mathbf{z})$. Second, G can be induced by a flow. Let $V : \mathbb{R}^d \times [0,T] \to \mathbb{R}^d$ be a Lipschitz velocity field, and define G as the unique solution to the ODE

$$G = G_T, \quad G_\tau(\mathbf{x}) = \mathbf{x}_\tau, \quad \mathbf{x}_0 = \mathbf{x}, \quad \frac{d}{d\tau}\mathbf{x}_\tau = V(\mathbf{x}_\tau, \tau), \tag{3.5}$$

where G_τ is the flow map. Furthermore, if we define the interpolant distributions $P_\tau = G_\tau\#\mathbb{P}$, then they form a (weak) solution to the continuity equation

$$\partial_\tau P_\tau + \nabla \cdot (V_\tau P_\tau) = 0. \tag{3.6}$$

Specifically, for any smooth test function ϕ

$$\begin{aligned}\int \phi(\mathbf{x}) d(\partial_\tau P_\tau)(\mathbf{x}) &= \frac{d}{d\tau}\int \phi(\mathbf{x}) dP_\tau(\mathbf{x}) = \frac{d}{d\tau}\int \phi\big(G_\tau(\mathbf{x})\big) d\mathbb{P}(\mathbf{x}) \\ &= \int V\big(G_\tau(\mathbf{x}), \tau\big) \cdot \nabla\phi\big(G_\tau(\mathbf{x})\big) d\mathbb{P}(\mathbf{x}) \\ &= \int V(\mathbf{x}, \tau) \cdot \nabla\phi(\mathbf{x}) dP_\tau(\mathbf{x}) = -\int \phi(\mathbf{x}) d\big(\nabla \cdot (V_\tau P_\tau)\big)(\mathbf{x}).\end{aligned}$$

Third, one can restrict to a subset of the possibly infinite set of solutions $\{G \,|\, P_* = G\#\mathbb{P}\}$, specifically generators that are gradients of some potential functions $\{G = \nabla\psi\}$. By Brennier's theorem [14, 117], such potential function exists in very general conditions. Similarly, one can restrict the velocity fields in (3.5) to time-dependent gradient fields, $V_\tau = \nabla\phi_\tau$. By [117, Theorem 5.51], in general there exists a potential function ϕ_τ such that the flow G induced by $\nabla\phi_\tau$ satisfies $P_* = G\#\mathbb{P}$. Specifically, the interpolant distribution $P_\tau = G_\tau\#\mathbb{P}$ is the Wasserstein geodesic that goes from $\mathbb{P}$ to P_*. Finally, it is interesting to note that there is also a heuristic argument from [2] that justifies the restriction to $\nabla\phi_\tau$: Given any velocity field V_τ with the interpolant distribution P_τ generated by (3.5), consider the equation

$$\nabla \cdot (P_\tau \nabla\phi_\tau) = \nabla \cdot (P_\tau V_\tau).$$

By the theory of elliptic PDE, the solution ϕ_τ exists. Hence, we can always replace a velocity field by a gradient field that induces the same interpolant distributions P_τ.

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

3. Fixed generator. Contrary to the free generator, another approach is to choose a specific coupling between the base and target distributions $\pi \in \Pi(\mathbb{P}, P_*)$, where

$$\Pi(\mathbb{P}, P_*) = \left\{ \pi \in \mathcal{P}(\mathbb{R}^d \times \mathbb{R}^d) \,\Big|\, \int \pi d\mathbf{x}_0 = \mathbb{P}, \int \pi d\mathbf{x}_1 = P_* \right\}$$

and the generator G is represented as the conditional distribution $\pi(\cdot|\mathbf{x}_0)$.

One can further extend π into a random path $\{X_\tau, \tau \in [0,T]\}$ so that $\pi = \text{law}(X_0, X_T)$. Then, analogous to the construction (3.5), G can be represented as the ODE or SDE that drives the trajectories X_τ. Thanks to the non-uniqueness of law, one can further consider the interpolant distributions $P_\tau = \text{law}(X_\tau)$ and solve for the velocity field V in the continuity equation (3.6). Then, G can be represented as the solution to the ODE (3.5) with velocity V.

Currently, models of this category belong to either of the two extremes:

Fully deterministic: For some measurable function G,

$$\pi(\mathbf{x}_0, \mathbf{x}_1) = \delta_{G(\mathbf{x}_0)}(\mathbf{x}_1)\mathbb{P}(\mathbf{x}_0). \tag{3.7}$$

The generator G is usually set to be the optimal transport map from $\mathbb{P}$ to P_*. The idea is simple in one dimension, such that we sort the "particles" of $\mathbb{P}$ and P_* and match according to this ordering. This monotonicity in $\mathbb{R}$ can be generalized to the cyclic monotonicity in higher dimensions [117]. Couplings $\pi \in \Pi(\mathbb{P}, P_*)$ that are cyclic monotonic are exactly the optimal transport plans with respect to the squared Euclidean metric [117], namely minimizers of (3.2). Then, Brennier's theorem [14, 117] implies that, under general conditions, the problem (3.2) has unique solution, which has the form (3.7), and furthermore the generator is a gradient field $G = \nabla\psi$ of a convex function ψ.

Fully random: The coupling is simply the product measure

$$\pi = \mathbb{P} \times P_*. \tag{3.8}$$

At first sight, this choice is trivial and intractable, but the trick is to choose an appropriate random path X_τ such that either the dynamics of X_τ or the continuity equation (3.6) is easy to solve.

One of the simplest constructions, proposed by [2, 75], is to use the linear interpolation

$$X_\tau = (1-\tau)X_0 + \tau X_1, \quad (X_0, X_1) \sim \pi, \quad \tau \in [0,1]. \tag{3.9}$$

Then, to solve for the target velocity field in (3.6), define a joint distribution M_* over $\mathbb{R}^d \times [0,1]$

$$\begin{aligned} \int \phi(\mathbf{x},\tau) dM_*(\mathbf{x},\tau) &= \int_0^1 \int \phi(\mathbf{x},\tau) dP_\tau d\tau \\ &= \int_0^1 \iint \phi\big((1-\tau)\mathbf{x}_0 + \tau\mathbf{x}_1, \tau\big) d\mathbb{P}(\mathbf{x}_0) dP_*(\mathbf{x}_1) d\tau \end{aligned} \tag{3.10}$$

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

for any test function ϕ. Similarly, define the current density J_*, a vector-valued measure, by

$$\int \mathbf{f}(\mathbf{x},\tau)\cdot dJ_*(\mathbf{x},\tau) = \int_0^1 \iint (\mathbf{x}_1-\mathbf{x}_0)\cdot \mathbf{f}\big((1-\tau)\mathbf{x}_0+\tau\mathbf{x}_1,\tau\big) d\mathbb{P}(\mathbf{x}_0)dP_*(\mathbf{x}_1)d\tau \tag{3.11}$$

for any test function $\mathbf{f}$. Then, we can define a velocity field V_* by the Radon-Nikodym derivative

$$V_* = \frac{dJ_*}{dM_*}. \tag{3.12}$$

Each $V_*(\mathbf{x},\tau)$ is the weighted average of the velocities of the random lines (3.9) that pass through the point $(\mathbf{x},\tau)$ in spacetime. As shown in [2,125], under general assumptions, V_* is the solution to the continuity equation (3.6) and satisfies

$$G_*\#\mathbb{P} = P_*,$$

where G_* is the generator defined by the flow (3.5) of V_*.

A more popular construction by [49,104,106] uses the diffusion process

$$X_0 \sim P_*, \quad dX_\tau = -\frac{\beta_\tau}{2}X_\tau d\tau + \sqrt{\beta_\tau}dW_\tau, \tag{3.13}$$

where $\beta_\tau > 0$ is a non-decreasing function that represents the noise scale. Consider the coupling $\pi_\tau = \text{law}(X_\tau, X_0)$. The conditional distribution $\pi_\tau(\cdot|\mathbf{x}_0)$ is an isotropic Gaussian [107]

$$\pi_\tau(\cdot|\mathbf{x}_0) = \mathcal{N}\Big(e^{-\frac{1}{2}\int_0^\tau \beta_s ds}\mathbf{x}_0,\ (1-e^{-\int_0^\tau \beta_s ds})I\Big) \tag{3.14}$$

and the interpolant distributions $P_\tau = \text{law}(X_\tau)$ are given by

$$P_\tau = \int \pi_\tau(\cdot|\mathbf{x}_0)dP_*(\mathbf{x}_0). \tag{3.15}$$

Then,

$$\begin{aligned}\text{KL}(\pi\|\pi_\tau) &= \int \ln\frac{d\pi}{d\pi_\tau}d\pi = \iint \ln\frac{\mathcal{N}(\mathbf{x})}{\pi_\tau(\mathbf{x}|\mathbf{x}_0)}d\mathcal{N}(\mathbf{x})dP_*(\mathbf{x}_0)\\ &= \int \text{KL}\left(\mathcal{N}\ \|\ \mathcal{N}(e^{-\frac{1}{2}\int_0^\tau \beta_s ds}\mathbf{x}_0,(1-e^{-\int_0^\tau \beta_s ds})I)\right)dP_*(\mathbf{x}_0)\\ &= \frac{e^{-\int_0^\tau \beta_s ds}}{1-e^{-\int_0^\tau \beta_s ds}}\left(\int \|\mathbf{x}_0\|^2 dP_*(\mathbf{x}_0)+d\right)+d\ln(1-e^{-\int_0^\tau \beta_s ds}),\end{aligned}$$

where the last line follows from the formula for the KL divergence between multivariable Gaussians. Let $\mathbb{P}$ be the unit Gaussian $\mathcal{N}$. It follows that if P_* has finite second moments, then the coupling π_τ converges to the product measure $\pi = \mathbb{P}\times P_*$ exponentially fast.

By choosing T sufficiently large, we have $P_T \approx \mathbb{P}$. Then, a generative model can be defined by sampling from $X_T \sim \mathbb{P}$ and then going through a reverse-time process $X_0 =$

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

$G(X_T)$ to approximate the target P_*. One approach is to implement the following reverse-time SDE [6]:

$$X_T \sim \mathbb{P}, \quad dX_\tau = -\frac{\beta_\tau}{2}\big(X_\tau + 2\nabla_{\mathbf{x}} \log P_\tau(X_\tau)\big)d\tau + \sqrt{\beta_\tau}d\overline{W}_\tau,$$

which is solved from time T to 0, and $\overline{W}$ is the reverse-time Wiener process. This backward SDE is equivalent to the forward SDE (3.13) in the sense that, if $P_T = \mathbb{P}$, then they induce the same distribution of paths $\{X_\tau, \tau \in [0,T]\}$ [6], and in particular $P_* = \text{law}(X_0)$. (An analysis that accounts for the approximation error between P_T and $\mathbb{P}$ is given by [105].) The gradient field $\nabla_{\mathbf{x}} \log P_\tau$ is known as the score function [56], which is modeled by a velocity field $\mathbf{s} : \mathbb{R}^d \times [0,T] \to \mathbb{R}^d$, and then the generator G can be defined as the following random function:

$$G(\mathbf{x}_T) = X_0, \quad X_T = \mathbf{x}_T, \quad dX_\tau = -\frac{\beta_\tau}{2}\big(X_\tau + 2\mathbf{s}(X_\tau,\tau)\big)d\tau + \sqrt{\beta_\tau}d\overline{W}_\tau. \tag{3.16}$$

Another approach is to implement the following reverse-time ODE [107]:

$$X_T \sim \mathbb{P}, \quad \frac{d}{d\tau}X_\tau = V(X_\tau,\tau), \quad V(\mathbf{x},\tau) = -\frac{\beta_\tau}{2}\big(\mathbf{x}_\tau + \nabla_{\mathbf{x}} \log P_\tau(\mathbf{x}_\tau)\big).$$

This V is the solution to the continuity equation (3.6), and we similarly have $P_* = \text{law}(X_0)$ if $P_T = \mathbb{P}$. Then, the generator G can be defined as a deterministic function

$$G(\mathbf{x}_T) = \mathbf{x}_0, \quad \frac{d}{d\tau}\mathbf{x}_\tau = -\frac{\beta_\tau}{2}\big(\mathbf{x}_\tau + \mathbf{s}(\mathbf{x}_\tau,\tau)\big). \tag{3.17}$$

4. Mixture. Finally, we remark that it is possible to use a combination of these representations. For instance, [82] uses a normalizing flow model reweighed by a Boltzmann distribution, which is helpful for sampling from distributions with multiple modes while maintaining accurate density estimation. Another possibility is that one can first train a model with fixed generator representation as a stable initialization, and then finetune the trained generator as a free generator (e.g. using GAN loss) to improve sample quality.

3.3 Loss type

There are numerous ways to define a metric or divergence on the space of probability measures, which greatly contribute to the diversity of distribution learning models. One requirement, however, is that since the target distribution P_* is replaced by its samples $P_*^{(n)}$ during training, the term P_* almost always appears in the loss as an expectation.

The commonly used losses belong to three categories:

1. Density-based loss. The modeled distribution P participates in the loss as a density function. The default choice is the KL divergence (3.1), which is equivalent up to constant to the negative log-likelihood (NLL)

$$L(P) = -\int \log P(\mathbf{x})dP_*(\mathbf{x}). \tag{3.18}$$

In fact, we can show that NLL is in a sense the only possible density-based loss.

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

Proposition 3.1. *Let L be any loss function on $\mathcal{P}_{ac}(\mathbb{R}^d)$ that has the form*

$$L(P) = \int f\big(P(\mathbf{x})\big) dP_*(\mathbf{x}),$$

where f is some C^1 function on $(0, +\infty)$. If for any $P_* \in \mathcal{P}_{ac}(\mathbb{R}^d)$, the loss L is minimized by P_*, then

$$f(p) = c \log p + c'$$

for $c \leq 0$ and $c' \in \mathbb{R}$. The converse is obvious.

Besides the KL divergence, there are several other well-known divergences in statistics such as the Jensen–Shannon divergence and χ^2 divergence. Despite that they are infeasible by Proposition 3.1, certain weakened versions of these divergences can still be used as will be discussed later.

2. Expectation-based loss. The modeled distribution participate in the loss through expectations. Since both P_* and P are seen as linear operators over test functions, it is natural to define the loss as a dual norm

$$L(P) = \sup_{\|D\| \leq 1} \int D(\mathbf{x}) d(P - P_*)(\mathbf{x}), \tag{3.19}$$

where $\|\cdot\|$ is some user-specified functional norm. The test function D is often called the discriminator, and such loss is called an adversarial loss. If $\|\cdot\|$ is a Hilbert space norm, then the loss can also be defined by

$$L(P) = \sup_{D} \int D(\mathbf{x}) d(P - P_*)(\mathbf{x}) - \|D\|^2 \tag{3.20}$$

There are several classical examples of adversarial losses: If $\|\cdot\|$ is the C_0 norm, then (3.19) becomes the total variation norm $\|P - P_*\|_{\text{TV}}$. If $\|\cdot\|$ is the Lipschitz semi-norm, then (3.19) becomes the 1-Wasserstein metric by Kantorovich-Rubinstein theorem [61, 117]. If $\|\cdot\|$ is the RKHS norm with some kernel k, then (3.19) becomes the maximum mean discrepancy (MMD) [41], and (3.20) is the squared MMD

$$L(P) = \frac{1}{2} \iint k(\mathbf{x}, \mathbf{x}') \, d(P_* - P)(\mathbf{x}) d(P_* - P)(\mathbf{x}'),$$

which gives rise to the moment matching network [72].

In practice, the discriminator D is usually parameterized by a neural network, denoted by D_θ with parameter θ. One common choice of the norm $\|\cdot\|$ is simply the l^∞ norm on θ

$$L(P) = \sup_{\|\theta\|_\infty \leq 1} \int D_\theta \, d(P - P_*) \tag{3.21}$$

This formulation gives rise to the WGAN model [7], and the l^∞ bound can be conveniently implemented by weight clipping. The loss (3.21) and its variants are generally known as the neural network distances [8, 37, 45, 133].

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

The strength of the metric (e.g. fineness of its topology) is proportional to the size of the normed space of $\|\cdot\|$, or inversely proportional to the strength of $\|\cdot\|$. Once some global regularity such as the Lipschitz norm applies to the space, then the dual norm or L becomes continuous with respect to the underlying geometry (e.g. the W_1 metric), and is no longer permutation invariant like NLL (3.18) or total variation. If we further restrict D to certain sparse subspaces of the Lipschitz functions, in particular neural networks, then L becomes insensitive to the "high frequency" parts of $\mathcal{P}(\mathbb{R}^d)$. As we will demonstrate in Section 6, this property is the source of good generalization.

Note that there are some variants of the GAN loss that resemble (3.20) but whose norms $\|D\|$ are so weak that the dual norms are no longer well-defined. For instance, the loss with L^2 penalty from [122]

$$L(P) = \sup_\theta \int D_\theta \, d(P - P_*) - \|D_\theta\|^2_{L^2(P+P_*)}$$

or the loss with Lipschitz penalty $\|1 - \|\nabla D_\theta\|\|^2_{L^2(P)}$ from [44]. By [127, Proposition 5], in general we have $L(P) = \infty$. Nevertheless, if we consider one-time-scale training such that P and D are trained with similar learning rates, then this blow-up can be avoided [127].

Beyond the dual norms (3.19), one can also consider divergences. Despite that Proposition 3.1 has ruled out the use of divergences other than the KL divergence, one can consider the weakened versions of the dual of these divergences. For instance, given any parameterized discriminator D_θ, the Jensen–Shannon divergence can be bounded below by [39]

$$\begin{aligned} \mathrm{JS}(P, P_*) &= \frac{1}{2}\mathrm{KL}\left(P \middle\| \frac{P + P_*}{2}\right) + \frac{1}{2}\mathrm{KL}\left(P_* \middle\| \frac{P + P_*}{2}\right) \\ &= \sup_{q:\mathbb{R}^d \to [0,1]} \int \log q(\mathbf{x}) dP(\mathbf{x}) + \int \log\big(1 - q(\mathbf{x})\big) dP_*(\mathbf{x}) + 2\ln 2 \\ &\geq \sup_\theta \int \log\left(\frac{e^{D_\theta(\mathbf{x})}}{1 + e^{D_\theta(\mathbf{x})}}\right) dP(\mathbf{x}) + \int \log\left(\frac{1}{1 + e^{D_\theta(\mathbf{x})}}\right) dP_*(\mathbf{x}) + 2\ln 2. \end{aligned} \tag{3.22}$$

This lower bound gives rise to the earliest version of GAN [39]. GANs based on other divergences have been studied in [78,83].

3. Regression loss. The regression loss is used exclusively by the fixed generator representation discussed in Section 3.2. If a target generator G_* has been specified, then we simply use the L^2 loss over the base distribution $\mathbb{P}$

$$L(G) = \frac{1}{2}\|G - G_*\|^2_{L^2(\mathbb{P})}. \tag{3.23}$$

If a target velocity field V_* has been specified, then the loss is integrated over the interpolant distributions P_τ

$$L(V) = \frac{1}{2}\int_0^T \big\|V(\cdot,\tau) - V_*(\cdot,\tau)\big\|^2_{L^2(P_\tau)} d\tau \tag{3.24}$$

or equivalently, we use the $L^2(M_*)$ loss with the joint distribution M_* defined by (3.10).

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

3.4 Combination

Having discussed the distribution representations and loss types, we can now combine them to derive the distribution learning models in Table 3.1. Our focus will be on the highlighted four classes in the table.

Density + Potential. Since Proposition 3.1 indicates that the negative log-likelihood (NLL) (3.18) is the only feasible density-based loss, we simply insert the potential-based representation (3.3) into NLL, and obtain a loss in the potential function V,

$$L(V) = \int V dP_* + \ln \int e^{-V} d\mathbb{P}. \tag{3.25}$$

This formulation gives rise to the bias-potential model [13, 115], also known as variationally enhanced sampling.

Density + Free generator. In order to insert the transport representation $P = G\#\mathbb{P}$ into NLL (3.18), we need to be able to compute the density $P(\mathbf{x})$. For simple cases such as when $\mathbb{P}$ is Gaussian and G is affine, the density $P(\mathbf{x})$ has closed form expression. Yet, in realistic scenarios when P needs to satisfy the universal approximation property and thus has complicated forms, one has to rely on indirect calculations. There are three common approaches:

1. Change of variables (for normalizing flows): If G is a C^1 diffeomorphism, the density of P is given by the change of variables formula

$$P(\mathbf{x}) = \det \nabla G^{-1}(\mathbf{x})\, \mathbb{P}\big(G^{-1}(\mathbf{x})\big).$$

Usually $\mathbb{P}$ is set to the unit Gaussian $\mathcal{N}$. Then, the NLL loss (3.18) becomes

$$\begin{aligned} L(G) &= -\int \log\det \nabla G^{-1}(\mathbf{x}) + \log \mathbb{P}\big(G^{-1}(\mathbf{x})\big) dP_*(\mathbf{x}) \\ &= \int \log\det \nabla G\big(G^{-1}(\mathbf{x})\big) + \frac{1}{2}\|G^{-1}(\mathbf{x})\|^2 dP_*(\mathbf{x}) + \text{constant}. \end{aligned}$$

If G is modeled by a flow $\{G_\tau, \tau \in [0,1]\}$ (3.5) with velocity field V, then its Jacobian satisfies

$$\begin{aligned} \frac{d}{d\tau} \det \nabla G_\tau(\mathbf{x}_0) &= \frac{d}{d\tau} \det \nabla \left(\mathbf{x}_0 + \int_0^\tau V\big(G_s(\mathbf{x}_0), s\big) ds \right) \\ &= \frac{d}{d\tau} \det \left(I + \int_0^\tau \nabla V\big(G_s(\mathbf{x}_0), s\big) \nabla G_s(\mathbf{x}_0) ds \right) \\ &= \text{Tr}\Big[\big(\nabla V(G_s(\mathbf{x}_0), s) \nabla G_s(\mathbf{x}_0)\big) \big(\nabla G_s(\mathbf{x}_0)\big)^{-1} \Big] \det \nabla G_\tau(\mathbf{x}_0) \\ &= \text{Tr}\big[\nabla V(G_s(\mathbf{x}_0), s)\big] \det \nabla G_\tau(\mathbf{x}_0). \end{aligned}$$

It follows that

$$\log\det \nabla G(\mathbf{x}_0) = \int_0^1 \text{Tr}\big[\nabla V(G_\tau(\mathbf{x}_0), \tau)\big] d\tau$$

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

and this is known as Abel's formula [113]. Hence, we obtain the loss of the normalizing flow model [21,112]

$$\begin{aligned}
L(V) &= \iint_0^1 \operatorname{Tr}\big[\nabla V(\mathbf{x}_\tau,\tau)\big]d\tau + \frac{1}{2}\|\mathbf{x}_0\|^2 dP_*(\mathbf{x}_1), \\
\mathbf{x}_\tau &:= G_\tau(G^{-1}(\mathbf{x}_1)).
\end{aligned} \tag{3.26}$$

Moreover, if the velocity field is defined by a gradient field $V(\cdot,\tau)=\nabla\phi_\tau$ as discussed in Section 3.2, then the loss has the simpler form

$$L(\phi) = \iint_0^1 \Delta\phi_\tau(\mathbf{x}_\tau)d\tau + \frac{1}{2}\|\mathbf{x}_0\|^2 dP_*(\mathbf{x}_1),$$

which leads to the Monge-Ampère flow model [131].

One potential shortcoming of NF is that diffeomorphisms might not be suitable for the generator when the target distribution P_* is singular, e.g. concentrated on low-dimensional manifolds, which is expected for real data such as images. To approximate P_*, the generator G needs to shrink the mass of $\mathbb{P}$ onto negligible sets, and thus G^{-1} blows up. As G^{-1} is involved in the loss, it can cause the training process to be unstable.

2. Variational lower bound (for VAE): Unlike NF, the variational autoencoders do not require the generator to be invertible, and instead use its posterior distribution. The generator can be generalized to allow for random output, and we define the conditional distribution

$$P(\cdot|\mathbf{z}) = \operatorname{law}(X), \quad X = G(\mathbf{z}).$$

The generalized inverse can be defined as the conditional distribution $Q_*(\cdot|\mathbf{x})$ that satisfies

$$P(\mathbf{x}|\mathbf{z})\mathbb{P}(\mathbf{z}) = P(\mathbf{x})Q_*(\mathbf{z}|\mathbf{x}), \quad P(\mathbf{x}) = \int P(\mathbf{x}|\mathbf{z})d\mathbb{P}(\mathbf{z})$$

in the distribution sense. If the generator is deterministic, i.e. $P(\cdot|\mathbf{z})=\delta_{G(\mathbf{z})}$, and invertible, then $Q_*(\cdot|\mathbf{x})$ is simply $\delta_{G^{-1}(\mathbf{x})}$. It follows that the KL divergence (3.1) can be written as

$$\begin{aligned}
\operatorname{KL}(P_*\|P) &= \int \log\frac{P_*(\mathbf{x})}{P(\mathbf{x})} + \operatorname{KL}\big(Q_*(\cdot|\mathbf{x})\big\|Q_*(\cdot|\mathbf{x})\big)dP_*(\mathbf{x}) \\
&= \min_{Q(\cdot|\mathbf{x})} \iint \log\frac{P_*(\mathbf{x})}{P(\mathbf{x})} + \operatorname{KL}\big(Q(\cdot|\mathbf{x})\big\|Q_*(\cdot|\mathbf{x})\big)dP_*(\mathbf{x}) \\
&= \min_{Q(\cdot|\mathbf{x})} \iint \log\frac{P_*(\mathbf{x})Q(\mathbf{z}|\mathbf{x})}{P(\mathbf{x})Q_*(\mathbf{z}|\mathbf{x})}dQ(\mathbf{z}|\mathbf{x})dP_*(\mathbf{x}) \\
&= \min_{Q(\cdot|\mathbf{x})} \iint \log\frac{P_*(\mathbf{x})Q(\mathbf{z}|\mathbf{x})}{P(\mathbf{x}|\mathbf{z})\mathbb{P}(\mathbf{z})}dQ(\mathbf{z}|\mathbf{x})dP_*(\mathbf{x}) \\
&= \min_{Q(\cdot|\mathbf{x})} \operatorname{KL}\big(P_*Q(\cdot|\mathbf{x}) \,\|\, P(\cdot|\mathbf{z})\mathbb{P}\big).
\end{aligned}$$

This is an example of the variational lower bound [64], and the NLL loss (3.18) now becomes

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

$$\min_{P(\cdot|\mathbf{z})}\min_{Q(\cdot|\mathbf{x})}\iint -\log P(\mathbf{x}|\mathbf{z})dQ(\mathbf{z}|\mathbf{x})+\mathrm{KL}\big(Q(\mathbf{z}|\mathbf{x})\big\|\mathbb{P}(\mathbf{z})\big)dP_*(\mathbf{x}),$$

which is the loss of VAE. To make the problem more solvable, the decoder $P(\cdot|\mathbf{z})$ and encoder $Q(\cdot|\mathbf{x})$ are usually parameterized by diagonal Gaussian distributions [64]

$$P(\cdot|\mathbf{z})=\mathcal{N}\big(G(\mathbf{z}),\mathrm{diag}(e^{\mathbf{s}(\mathbf{z})})\big),\quad Q(\cdot|\mathbf{x})=\mathcal{N}\big(F(\mathbf{x}),\mathrm{diag}(e^{\mathbf{v}(\mathbf{x})})\big),$$

where $G,F,\mathbf{s},\mathbf{v}$ are parameterized functions $\mathbb{R}^d\to\mathbb{R}^d$ such as neural networks, and exp is taken entry-wise. Using the formula for KL divergence between Gaussians

$$\begin{aligned}&\mathrm{KL}\big(\mathcal{N}(m_0,\Sigma_0)\big\|\mathcal{N}(m_1,\Sigma_1)\big)\\&=\frac{1}{2}\left[\log\frac{\det\Sigma_1}{\det\Sigma_0}-d+\mathrm{Tr}\big[\Sigma_1^{-1}\Sigma_0\big]+(m_1-m_0)^T\Sigma_1^{-1}(m_1-m_0)\right],\end{aligned}$$

we can show that, up to constant, the VAE loss equals

$$\begin{aligned}\min_{G,s}\min_{F,v}\frac{1}{2}\iint&\frac{\|\mathbf{x}-G(F(\mathbf{x})+e^{\mathbf{v}(\mathbf{x})}\odot\omega)\|^2}{e^{\mathbf{s}(F(\mathbf{x})+e^{\mathbf{v}(\mathbf{x})}\odot\omega)}}\\&+\sum_{i=1}^d\mathbf{s}_i\big(F(\mathbf{x})+e^{\mathbf{v}(\mathbf{x})}\odot\omega\big)d\mathcal{N}(\omega)\\&+\|F(\mathbf{x})\|^2+\sum_{i=1}^d e^{\mathbf{v}_i(\mathbf{x})}-\mathbf{v}_i(\mathbf{x})\,dP_*(\mathbf{x}),\end{aligned}$$

where $\odot$ is entry-wise product. This loss resembles the classical autoencoder [1,102]

$$\min_{F,G}\int\frac{\|\mathbf{x}-G(F(\mathbf{x}))\|^2}{2}dP_*(\mathbf{x}),$$

and thus $P(\cdot|\mathbf{z}),Q(\cdot|\mathbf{x})$ are addressed by the decoder and encoder.

3. Factorization (for autoregressive model): To model a distribution P over sequential data $\mathbf{X}=[\mathbf{x}_1,\ldots,\mathbf{x}_l]$, one can choose a generator G that is capable of processing variable-length inputs $[\mathbf{x}_1,\ldots,\mathbf{x}_i]$, such as the Transformer network [116] or recurrent networks [99], and define the distribution by

$$\begin{aligned}&P(\mathbf{X})=\prod_{i=1}^l P(\mathbf{x}_i|\mathbf{x}_1,\ldots,\mathbf{x}_{i-1}),\\&P(\cdot|\mathbf{x}_1,\ldots,\mathbf{x}_{i-1})=\mathrm{law}(X_i),\quad X_i\sim G(Z|\mathbf{x}_1,\ldots,\mathbf{x}_{i-1}),\quad Z\sim\mathbb{P}.\end{aligned}$$

Then, NLL (3.18) is reduced to

$$-\int\log P(\mathbf{X})dP_*(\mathbf{X})=-\int\sum_{i=1}^l\log P(\mathbf{x}_i|\mathbf{x}_1,\ldots,\mathbf{x}_{i-1})dP_*(\mathbf{x}).$$

Usually, each $P(\cdot|\mathbf{x}_1,\ldots,\mathbf{x}_{i-1})$ has a simple parametrization such as Gaussian or Softmax so that $\log P(\mathbf{x}_i|\mathbf{x}_1,\ldots,\mathbf{x}_{i-1})$ is tractable [86,92].

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

Expectation + Free generator. By the definition (3.4) of the transport representation $P = G\#\mathbb{P}$,

$$\int f(\mathbf{x})dP(\mathbf{x}) = \int f\big(G(\mathbf{z})\big)d\mathbb{P}(\mathbf{z})$$

for all measurable functions f. Then, the classical GAN loss (3.22) becomes

$$\min_G \max_D \int \log\left(\frac{e^{D(G(\mathbf{z}))}}{1+e^{D(G(\mathbf{z}))}}\right) d\mathbb{P}(\mathbf{z}) + \int \log\left(\frac{1}{1+e^{D(\mathbf{x})}}\right) dP_*(\mathbf{x}). \tag{3.27}$$

Similarly, the WGAN loss (3.21) becomes

$$\min_G \max_{\|\theta\|_\infty \le 1} \int D_\theta\big(G(\mathbf{z})\big)d\mathbb{P}(\mathbf{z}) - \int D_\theta(\mathbf{x})dP_*(\mathbf{x}). \tag{3.28}$$

Regression + Fixed generator. For the case with fully deterministic coupling (3.7), a target generator G_* is provided by numerical optimal transport, and then fitted by a parameterized function G with the regression loss (3.23). This formulation leads to the generative model [132]. (Moreover, a few models with some technical variations [4,5,98] are related to this category, but for simplicity we do not describe them here.)

For the case with fully random coupling (3.8), we fit either the score function $\nabla \log P_\tau$ from (3.15) or the velocity field V_* from (3.12) using the regression loss (3.24). Note that the targets (3.15, 3.12) are both defined by expectations and thus the loss (3.24) cannot be computed directly. Thanks to the linearity of expectation, we can expand the loss to make the computation tractable.

Model the score function $\nabla \log P_\tau$ by a velocity field $\mathbf{s} : \mathbb{R}^d \times [0,T] \to \mathbb{R}^d$ and let $\lambda_\tau > 0$ be a user-specified weight. The regression loss can be written as

$$\begin{aligned}
L(\mathbf{s}) &:= \frac{1}{2}\int_0^T \lambda_\tau \|\mathbf{s}(\cdot,\tau) - \nabla \log P_\tau\|^2_{L^2(P_\tau)} d\tau \\
&= \int_0^T \lambda_\tau \int \frac{1}{2}\|\mathbf{s}(\mathbf{x},\tau)\|^2 - \mathbf{s}(\mathbf{x},\tau)\cdot\nabla \log P_\tau(\mathbf{x})\; dP_\tau(\mathbf{x})d\tau + C \\
&= \int_0^T \lambda_\tau \int \frac{1}{2}\|\mathbf{s}(\mathbf{x},\tau)\|^2 dP_\tau(\mathbf{x}) - \lambda_\tau \int \mathbf{s}(\mathbf{x},\tau)\cdot\nabla P_\tau(\mathbf{x})d\mathbf{x}\, d\tau + C \\
&= \int_0^T \lambda_\tau \iint \frac{1}{2}\|\mathbf{s}(\mathbf{x},\tau)\|^2 d\pi_\tau(\mathbf{x}|\mathbf{x}_0)dP_*(\mathbf{x}_0) \\
&\quad - \lambda_\tau \int \mathbf{s}(\mathbf{x},\tau)\cdot\nabla \int \pi_\tau(\mathbf{x}|\mathbf{x}_0)dP_*(\mathbf{x}_0)d\mathbf{x}\, d\tau + C \\
&= \int_0^T \lambda_\tau \iint \frac{1}{2}\|\mathbf{s}(\mathbf{x},\tau)\|^2 - \mathbf{s}(\mathbf{x},\tau)\cdot\nabla\log\pi_\tau(\mathbf{x}|\mathbf{x}_0)\; d\pi_\tau(\mathbf{x}|\mathbf{x}_0)dP_*(\mathbf{x}_0)d\tau + C \\
&= \int_0^T \frac{\lambda_\tau}{2} \iint \|\mathbf{s}(\mathbf{x},\tau) - \nabla\log\pi_\tau(\mathbf{x}|\mathbf{x}_0)\|^2 d\pi_\tau(\mathbf{x}|\mathbf{x}_0)dP_*(\mathbf{x}_0)d\tau + C.
\end{aligned}$$

Since the conditional distribution $\pi_\tau(\mathbf{x}|\mathbf{x}_0)$ is the isotropic Gaussian (3.14), this loss is straightforward to evaluate. Thus, we obtain the loss of the score-based diffusion models [49,104,106,107]

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02
OPEN ACCESS

$$L(\mathbf{s}) = \int_0^T \frac{\lambda_\tau}{2} \iint \left\| \mathbf{s}\Big(e^{-\frac{1}{2}\int_0^\tau \beta_s ds}\mathbf{x}_0 + \sqrt{1-e^{-\int_0^\tau \beta_s ds}}\,\omega, \tau\Big) \right.$$
$$\left. + \frac{\omega}{\sqrt{1-e^{-\int_0^\tau \beta_s ds}}} \right\|^2 d\mathcal{N}(\omega) dP_*(\mathbf{x}_0) d\tau. \tag{3.29}$$

Similarly, for the velocity field V_* (3.12), using the definitions (3.10,3.11) of the joint distribution M_* and current density J_*, we can write the regression loss as

$$\begin{aligned}
L(V) &:= \frac{1}{2}\int_0^1 \|V(\cdot,\tau) - V_*(\cdot,\tau)\|^2_{L^2(P_\tau)} d\tau \\
&= \int \frac{1}{2}\|V(\mathbf{x},\tau)\|^2 - V(\mathbf{x},\tau)\cdot V_*(\mathbf{x},\tau)\, dM_*(\mathbf{x},\tau) + C \\
&= \int \frac{1}{2}\|V(\mathbf{x},\tau)\|^2 dM_*(\mathbf{x},\tau) - \int V \cdot dJ_* + C \\
&= \int_0^1 \iint \frac{1}{2}\big\|V\big((1-\tau)\mathbf{x}_0 + \tau\mathbf{x}_1, \tau\big)\big\|^2 d\mathbb{P}(\mathbf{x}_0) dP_*(\mathbf{x}_1) d\tau \\
&\quad - \int_0^1 \iint (\mathbf{x}_1 - \mathbf{x}_0)\cdot V\big((1-\tau)\mathbf{x}_0 + \tau\mathbf{x}_1, \tau\big) d\mathbb{P}(\mathbf{x}_0) dP_*(\mathbf{x}_1) d\tau + C \\
&= \frac{1}{2}\int_0^1 \iint \big\|V\big((1-\tau)\mathbf{x}_0 + \tau\mathbf{x}_1, \tau\big) - (\mathbf{x}_1 - \mathbf{x}_0)\big\|^2 d\mathbb{P}(\mathbf{x}_0) dP_*(\mathbf{x}_1) d\tau + C.
\end{aligned} \tag{3.30}$$

Thus, we obtain the loss of normalizing flow with stochastic interpolants [2,75,125].

Other classes. Finally, we briefly remark on the rest of the classes in Table 3.1. For the combination "Density + Fixed generator", it has been shown by [105] that the regression loss upper bounds the KL divergence. Specifically, if in the loss L (3.29) we set the weight by $\lambda_\tau = \beta_\tau$ where β_τ is the noise scale in the SDE (3.13), then given any score function $\mathbf{s}$,

$$\mathrm{KL}(P_* \| G_{\mathbf{s}}\#\mathbb{P}) \leq L(\mathbf{s}) + \mathrm{KL}(P_T \| \mathbb{P}), \tag{3.31}$$

where $G_{\mathbf{s}}$ is the generator defined by the reverse-time SDE (3.16) with the score $\mathbf{s}$. The result also holds for $G_{\mathbf{s}}$ defined by the reverse-time ODE (3.17) under a self-consistency assumption: let $(G_{\mathbf{s}})_1^\tau$ be the reverse-time flow map of (3.17), then

$$\mathbf{s}(\mathbf{x},\tau) = \nabla \log \big((G_{\mathbf{s}})_1^\tau \#\mathbb{P}\big)(\mathbf{x}).$$

The combinations "Expectation + Potential" and "Expectation + Fixed generator" are feasible, but we are not aware of representative models. The combinations "Regression + Potential" and "Regression + Free generator" do not seem probable, since there is no clear target to perform regression.

Remark 3.1 (Empirical loss). As discussed in Section 3.3, the loss is almost always an expectation in the target distribution P_*. Indeed, one can check that all the loss functions introduced in this section can be written in the abstract form

$$L(f) = \int F(f, \mathbf{x}) dP_*(\mathbf{x}),$$

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

where f is the parameter function and F depends on the model. Thus, if only a finite sample set of P_* is available, as is usually the case in practice, one can define the empirical loss

$$L^{(n)}(f) = \int F(f, \mathbf{x}) dP_*^{(n)}(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^{n} F(f, \mathbf{x}_i), \tag{3.32}$$

where $P_*^{(n)}$ is the empirical distribution.

3.5 Function representation

Having parameterized the distributions and losses by abstract functions, the next step is to parameterize these functions by machine learning models such as neural networks. There is much freedom in this choice, such that any parametrization used in supervised learning should be applicable to most of the functions we have discussed. These include the generators and discriminators of GANs, the means and variances of the decoder and encoder of VAE, the potential function of the bias potential model, the score function of score-based diffusion models, and the velocity field of normalizing flows with stochastic interpolants. Some interesting applications are given by [27,58,92,96].

One exception is the generator G of the normalizing flows (3.26), which needs to be invertible with tractable Jacobian. As mentioned in Section 2, one approach is to parameterize G as a sequence of invertible blocks whose Jacobians have closed-form formula (Example designs can be found in [29,55,63,89]). Another approach is to represent G as a flow, approximate this flow with numerical schemes, and solve for the traces $\text{Tr}[\nabla V]$ in (3.26) (Examples of numerical schemes are given by [21,40,131]).

For the theoretical analysis in the rest of this paper, we need to fix a function representation. Since our focus is on the phenomena that are unique to learning distributions (e.g. memorization), we keep the function representation as simple as possible, while satisfying the minimum requirement of the universal approximation property among distributions (and thus the capacity for memorization). Specifically, we use the random feature functions [35,93,126].

Definition 3.1 (Random feature functions)**.** *Let $\mathcal{H}(\mathbb{R}^d, \mathbb{R}^k)$ be the space of functions that can be expressed as*

$$f_{\mathbf{a}}(\mathbf{x}) = \mathbb{E}_{\rho(\mathbf{w},b)}\big[\mathbf{a}(\mathbf{w},b)\ \sigma(\mathbf{w}\cdot\mathbf{x}+b)\big], \tag{3.33}$$

where $\rho \in \mathcal{P}(\mathbb{R}^{d+1})$ is a fixed parameter distribution and $\mathbf{a} \in L^2(\rho, \mathbb{R}^k)$ is a parameter function. For simplicity, we use the notation $\mathcal{H}$ when the input and output dimensions d, k are clear.

Definition 3.2 (RKHS norm)**.** *For any subset $\Omega \subseteq \mathbb{R}^d$, consider the quotient space*

$$\mathcal{H}(\Omega) = \mathcal{H}/\{f_{\mathbf{a}} \equiv \mathbf{0} \ \ on \ \ \Omega\}$$

with the norm

$$\|f\|_{\mathcal{H}(\Omega)} = \inf\big\{\|\mathbf{a}\|_{L^2(\rho)} \mid f = f_{\mathbf{a}} \ \ on \ \ \Omega\big\}.$$

We use the notation $\|\cdot\|_{\mathcal{H}}$ if Ω is clear from context. By [26,93], *$\mathcal{H}(\Omega)$ is a Hilbert space and $\|\cdot\|_{\mathcal{H}(\Omega)}$ is equal to the RKHS norm (reproducing kernel Hilbert space) induced by the kernel*

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

$$k(\mathbf{x},\mathbf{x}') = \mathbb{E}_\rho\big[\sigma(\mathbf{w}\cdot\mathbf{x}+b)\sigma(\mathbf{w}\cdot\mathbf{x}'+b)\big].$$

Furthermore, given any distribution $\mathbb{P} \in \mathcal{P}(\Omega)$, we can define the following integral operator $K : L^2(\mathbb{P}) \to L^2(\mathbb{P})$:

$$K(f)(\mathbf{x}) = \int k(\mathbf{x},\mathbf{x}')f(\mathbf{x}')d\mathbb{P}(\mathbf{x}'). \tag{3.34}$$

Definition 3.3 (Time-dependent random feature function). *Given any $V \in \mathcal{H}(\mathbb{R}^{d+1},\mathbb{R}^d)$, one can define a flow by*

$$G_V(\mathbf{x}_0) = \mathbf{x}_1, \quad \frac{d}{d\tau}\mathbf{x}_\tau = V(\mathbf{x}_\tau,\tau).$$

Define the flow-induced norm

$$\|V\|_{\mathcal{F}} = \exp\|V\|_{\mathcal{H}}.$$

Our results adopt either of the following settings:

Assumption 3.1. Assume that the activation σ is ReLU $\sigma(x) = \max(x,0)$. Assume that the parameter distribution ρ is supported on the l^1 sphere $\{\|\mathbf{w}\|_1 + |b| = 1\}$ and has a positive and continuous density over this sphere.

Assumption 3.2. Assume that the activation σ is sigmoid $\sigma(x) = e^x/(1+e^x)$. Assume that ρ has a positive and continuous density function over $\mathbb{R}^{d+1}$ and also bounded variance

$$\int(\|\mathbf{w}\|^2 + b^2)d\rho(\mathbf{w},b) \leq 1.$$

Given either assumption, the universal approximation theorems [51, 109] imply that the space $\mathcal{H}(K,\mathbb{R}^k)$ is dense among the continuous functions $C(K,\mathbb{R}^k)$ with respect to the C_0 norm for any compact subset $K \subseteq \mathbb{R}^d$. Also, by Lemma 9.1, $\mathcal{H}(\mathbb{R}^d,\mathbb{R}^k)$ is dense in $L^2(P,\mathbb{R}^k)$ for any distribution $P \in \mathcal{P}(\mathbb{R}^d)$.

The random feature functions (3.33) can be seen as a simplified form of neural networks, e.g. if we replace the parameter distribution ρ by a finite sample set $\{(\mathbf{w}_i,b_i)\}_{i=1}^m$, then (3.33) becomes a 2-layer network with m neurons and frozen first layer weights. Similarly, for the flow G_V in Definition 3.3, if the ODE is replaced by a forward Euler scheme, then G_V becomes a deep residual network whose layers share similar weights. Beyond the random feature functions, one can extend the analysis to the Barron functions [11,34] and flow-induced functions [31], which are the continuous representations of 2-layer networks and residual networks.

3.6 Training rule

The training of distribution learning models is very similar to training supervised learning models, such that one chooses from the many algorithms for gradient descent and optimizes the function parameters. One exception is the GANs, whose losses are min-max problems of the form (3.27,3.28) and are usually solved by performing gradient descent on the generator and gradient ascent on the discriminator [39].

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

For the theoretical analysis in this paper, we use the continuous time gradient descent. Specifically, given any loss $L(f)$ over $L^2(\mathbb{P},\mathbb{R}^k)$ for some $\mathbb{P}\in\mathcal{P}(\mathbb{R}^d)$, we parameterize f by the random feature function $f_{\mathbf{a}}$ from Definition 3.1 and denote the loss by $L(\mathbf{a})=L(f_{\mathbf{a}})$. Given any initialization $\mathbf{a}_0\in L^2(\rho,\mathbb{R}^k)$, we define the trajectory $\{\mathbf{a}_t, t\geq 0\}$ by the dynamics

$$\frac{d}{dt}\mathbf{a}_t=-\nabla_{\mathbf{a}}L\big|_{\mathbf{a}_t}=-\int\nabla_f L(\mathbf{x})\,\sigma(\mathbf{w}\cdot\mathbf{x}+b)d\mathbb{P}(\mathbf{x}). \tag{3.35}$$

It follows that the function $f_t=f_{\mathbf{a}_t}$ evolves by

$$\frac{d}{dt}f_t=\mathbb{E}_{\rho(\mathbf{w},b)}\left[\frac{d}{dt}\mathbf{a}_t(\mathbf{w},b)\,\sigma(\mathbf{w}\cdot\mathbf{x}+b)\right]=-K(\nabla_f L), \tag{3.36}$$

where K is the integral operator defined in (3.34). Similarly, given the empirical loss $L^{(n)}$ (3.32), we define the empirical training trajectory

$$\frac{d}{dt}f_t^{(n)}=-K\big(\nabla_f L^{(n)}\big). \tag{3.37}$$

By default, we use the initialization

$$\mathbf{a}_0=\mathbf{a}_0^{(n)}\equiv\mathbf{0} \tag{3.38}$$

or equivalently $f_0=f_0^{(n)}\equiv\mathbf{0}$.

3.7 Test error

For our theoretical analysis, given a modeled distribution P and target distribution P_*, we measure the test error by either the Wasserstein metric $W_2(P_*,P)$ or KL-divergence $\mathrm{KL}(P_*\|P)$. As discussed in Section 1, W_2 exhibits the curse of dimensionality, while KL is stronger than W_2. Thus, they are capable of detecting memorization and can distinguish the solutions that generalize well.

In addition, one advantage of the W_2 metric is that it can be related to the regression loss.

Proposition 3.2 ([127, Proposition 21]). *Given any base distribution $\mathbb{P}\in\mathcal{P}_{2,ac}(\mathbb{R}^d)$ and any target distribution $P_*\in\mathcal{P}_2(\mathbb{R}^d)$, for any $G\in L^2(\mathbb{P},\mathbb{R}^d)$*

$$W_2(P_*,G\#\mathbb{P})=\inf\Big\{\|G-G_*\|_{L^2(\mathbb{P})}\mid G_*\in L^2(\mathbb{P},\mathbb{R}^d),\ P_*=G_*\#\mathbb{P}\Big\}.$$

So effectively, the W_2 test error is the L^2 error with the closest target generator.

One remark is that these test losses are only applicable to theoretical analysis. In practice, we only have a finite sample set from P_*, and the curse of dimensionality becomes an obstacle to meaningful evaluation.

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

4 Universal Approximation Theorems

It is not surprising that distribution learning models equipped with neural networks satisfy the universal approximation property in the space of probability distributions. The significance is that the models in general have the capacity for memorization. This section confirms that the universal approximation property holds for all three distribution representations introduced in Section 3.2. Since our results are proved with the random feature functions (Definitions 3.1 and 3.3), they hold for more expressive function parametrizations such as 2-layer and deep neural networks.

For the free generator representation, the following result is straightforward.

Proposition 4.1. *Given either Assumption* 3.1 *or* 3.2, *for any base distribution* $\mathbb{P} \in \mathcal{P}_{2,ac}(\mathbb{R}^d)$, *the set of distributions generated by the random feature functions* $\mathcal{H}(\mathbb{R}^d, \mathbb{R}^d)$ *are dense with respect to the* W_2 *metric. Specifically, for any* $P_* \in \mathcal{P}_2(\mathbb{R}^d)$,

$$\inf_{G \in \mathcal{H}} W_2(P_*, G\#\mathbb{P}) = 0.$$

In particular, $G\#\mathbb{P}$ can approximate the empirical distribution $P_*^{(n)}$.

4.1 Potential representation

Consider the potential-based representation (3.3). Let $K \subseteq \mathbb{R}^d$ be any compact set with positive Lebesgue measure, let $\mathcal{P}_{ac}(K) \cap C(K)$ be the space of distributions with continuous density functions, and let the base distribution $\mathbb{P}$ be uniform over K.

Proposition 4.2. *Given either Assumption* 3.1 *or* 3.2, *the set of probability distributions*

$$\mathcal{P}_{\mathcal{H}} = \left\{ \frac{1}{Z} e^{-V} \mathbb{P} \;\middle|\; V \in \mathcal{H}(\mathbb{R}^d, \mathbb{R}) \right\}$$

is dense in

- $\mathcal{P}(K)$ *under the Wasserstein metric* W_p *(*$1 \leq p < \infty$*),*
- $\mathcal{P}_{ac}(K)$ *under the total variation norm* $\|\cdot\|_{TV}$*,*
- $\mathcal{P}_{ac}(K) \cap C(K)$ *under KL divergence.*

4.2 Flow-based free generator

For the normalizing flows, we have seen in Section 3.4 the two common approaches for modeling the generator G, i.e. continuous-time flow (2.2) or concatenation of simple diffeomorphisms. Both approaches have an apparent issue, that they do not satisfy the universal approximation property among functions. Since G is always a diffeomorphism, it cannot approximate for instance functions that are overlapping or orientation-reversing (such as $x \mapsto |x|$ and $x \mapsto -x$). Hence, the approach of Proposition 4.1 is not applicable.

Nevertheless, to transport probability distributions, it is sufficient to restrict to specific kinds of generators, for instance gradient fields $\nabla\phi$ according to Brennier's theorem [14, 117]. Using flows induced by random feature functions, we have the following result.

Proposition 4.3. *Given Assumption* 3.2*, for any base distribution $\mathbb{P} \in \mathcal{P}_{2,ac}(\mathbb{R}^d)$, the following set of distributions is dense in $\mathcal{P}_2(\mathbb{R}^d)$ with respect to the W_2 metric*

$$\mathcal{P}_G = \big\{G_V\#\mathbb{P} \mid V \in \mathcal{H}(\mathbb{R}^{d+1}, \mathbb{R}^d)\big\},$$

where G_V is given by Definition 3.3*.*

4.3 Fixed generator

For the fixed generator representation, we analyze the normalizing flow with stochastic interpolants (3.12) instead of the score-based diffusion models (3.13), since the former has a simpler formulation. As the target velocity field V_* has been specified, we show a stronger result than Propositions 4.1 and 4.3, such that we can simultaneously bound the W_2 test error and the L^2 training loss.

Proposition 4.4 ([125, Proposition 3.2])**.** *Given Assumption* 3.2*, assume that the base distribution $\mathbb{P}$ is compactly-supported and has C^2 density. For any target distribution P_* that is compactly-supported, let V_* be the velocity field* (3.12)*. Then, for any $\epsilon > 0$, there exists a velocity field $V_\epsilon \in \mathcal{H}(\mathbb{R}^{d+1}, \mathbb{R}^d)$ with induced generator $G_\epsilon = G_{V_\epsilon}$ given by Definition* 3.3*, such that*

$$\begin{aligned}&W_2(P_*, G_\epsilon\#\mathbb{P}) < \epsilon,\\ &\|V_* - V_\epsilon\|_{L^2(M_*)} = \sqrt{2}\sqrt{L(V_\epsilon) - L(V_*)} < \epsilon,\end{aligned}$$

where M_* is the joint distribution (3.10) *and L is the loss* (3.30)*.*

5 Memorization

The previous section has shown that the distribution learning models, from all known distribution representations, satisfy the universal approximation property. In particular, they are capable of approximating the empirical distribution $P_*^{(n)}$ and thus have the potential for memorization. This section confirms that memorization is inevitable for some of the models. Specifically, we survey our results on the universal convergence property, that is, the ability of a model to converge to any given distribution during training. We believe that this property holds for other models as well, and it should be satisfied by any desirable model for learning probability distributions.

5.1 Bias-potential model

Recall that the bias-potential model is parameterized by potential functions V (3.3) and minimizes the loss L (3.25). For any compact set $K \subseteq \mathbb{R}^d$ with positive Lebesgue measure, let the base distribution $\mathbb{P}$ be uniform over K. Parameterize V by random feature functions $\mathcal{H}(\mathbb{R}^d,\mathbb{R})$, and define the training trajectory V_t by continuous time gradient descent (3.36) on L with any initialization $V_0 \in \mathcal{H}$. Denote the modeled distribution by $P_t = (1/Z)e^{-V_t}\mathbb{P}$. Similarly, let $V_t^{(n)}$ be the training trajectory on the empirical loss (3.32) and denote its modeled distribution by $P_t^{(n)}$.

Proposition 5.1 ([126, Lemma 3.8]). *Given Assumption* 3.1*, for any target distribution $P_* \in \mathcal{P}(K)$, if P_t has only one weak limit, then P_t converges weakly to P_**

$$\lim_{t\to\infty} W_2(P_*,P_t) = 0.$$

Corollary 5.1 (Memorization, [126, Proposition 3.7]). *Given Assumption* 3.1*, the training trajectory $P_t^{(n)}$ can only converge to the empirical distribution $P_*^{(n)}$. Moreover, both the test error and the norm of the potential function diverge*

$$\lim_{t\to\infty} KL\big(P_*\|P_t^{(n)}\big) = \lim_{t\to\infty} \big\|V_t^{(n)}\big\|_{\mathcal{H}} = \infty.$$

In the setting of Proposition 5.1, a limit point always exists, but we have to exclude the possibility of more than one limit point, e.g. the trajectory P_t may converge to a limit circle. We believe that with a more refined analysis one can prove that such exotic scenario cannot happen.

5.2 GAN discriminator

As will be demonstrated in Section 7, the training and convergence of models with the free generator representation is in general difficult to analyze. Thus, we consider the simplified GAN model from [127] such that the representation $G\#\mathbb{P}$ replaced by a density function P.

Consider the GAN loss (3.20), and parameterize the discriminator by $D \in \mathcal{H}(\mathbb{R}^d,\mathbb{R})$. Equivalently, we set the penalty term $\|D\|$ to be the RKHS norm $\|\cdot\|_{\mathcal{H}}$ and the loss (3.20) becomes

$$\begin{aligned} L(P) &= \sup_D \int D(\mathbf{x})d(P-P_*)(\mathbf{x}) - \|D\|_{\mathcal{H}}^2 \\ &= \max_a \iint a(\mathbf{w},b)\sigma(\mathbf{w}\cdot\mathbf{x}+b)d\rho(\mathbf{w},b)d(P-P_*)(\mathbf{x}) - \|a\|_{L^2(\rho)}^2 \\ &= \frac{1}{2}\iint k(\mathbf{x},\mathbf{x}')d(P-P_*)(\mathbf{x})d(P-P_*)(\mathbf{x}'), \end{aligned} \tag{5.1}$$

where k is the kernel function from Definition 3.2. This loss is an instance of the maximum mean discrepancy [41]. Model the density P as a function in $L^2([0,1]^d)$, and define the training trajectory P_t by continuous time gradient descent

$$\frac{d}{dt}P_t = -\nabla_P L(P_t) = -k*(P_t-P_*), \tag{5.2}$$

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

where

$$(k * P)(\mathbf{x}) = \int k(\mathbf{x}, \mathbf{x}')dP(\mathbf{x}').$$

Let Π_Δ be the nearest point projection from $L^2([0,1]^d)$ to the convex subset $\mathcal{P}([0,1]^d) \cap L^2([0,1]^d)$. We measure the test error by $W_2(P_*, \Pi_\Delta(P))$.

Proposition 5.2 ([127, Lemma 13]). *Given Assumption* 3.1*, for any target distribution $P_* \in \mathcal{P}([0,1]^d)$ and any initialization $P_0 \in L^2([0,1]^d)$, the distribution $\Pi_\Delta(P_t)$ converges weakly to P_**

$$\lim_{t\to\infty} W_2\big(P_*, \Pi_\Delta(P_t)\big) = 0.$$

6 Generalization Error

Despite that Sections 4 and 5 have demonstrated that distribution learning models have the capacity for memorization, this section shows that solutions with good generalization are still achievable. For the four classes highlighted in Table 3.1, we show that their models escape from the curse of dimensionality with either early stopping or regularization. Specifically, their generalization errors scale as $\mathcal{O}(n^{-\alpha})$ where α are absolute constants, instead of dimension-dependent terms such as α/d.

These results depend on either of the two forms of regularizations:

- Implicit regularization: As depicted in Fig. 6.1 (left), the training trajectory P_t comes very close to the hidden target distribution P_* before eventually turning towards the empirical distribution $P_*^{(n)}$.
- Explicit regularization: Analogous to the above picture, we consider some regularized loss $L^{(n)} + R(\lambda)$ with strength $\lambda \geq 0$. With an appropriate regularization strength, the minimizer P_λ becomes very close to the hidden target P_*.

The mechanism underlying both scenarios is that the function representations of the models are insensitive to the sampling error $P_* - P_*^{(n)}$. Thus, we resolve the seeming paradox between good generalization and the inevitable memorization.

Without a good function representation, this behavior cannot be guaranteed. For instance, as argued in [127], if a distribution P_t is trained by Wasserstein gradient flow (i.e. without any function parametrization) on the empirical loss

$$L^{(n)}(P) = W_2\big(P_*^{(n)}, P\big)$$

and if the initialization $P_0 \neq P_*$ is in $\mathcal{P}_{2,ac}(\mathbb{R}^d)$, then the training trajectory P_t follows the W_2 geodesic that connects P_0 and $P_*^{(n)}$. Since the Wasserstein manifold has positive curvature [3], the geodesic in general can never come close to the hidden target P_*, as depicted in Fig. 6.1 (right).

In the following five subsections, we survey our results on three models that have rigorous proofs, and then analyze two models with heuristic calculations.

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

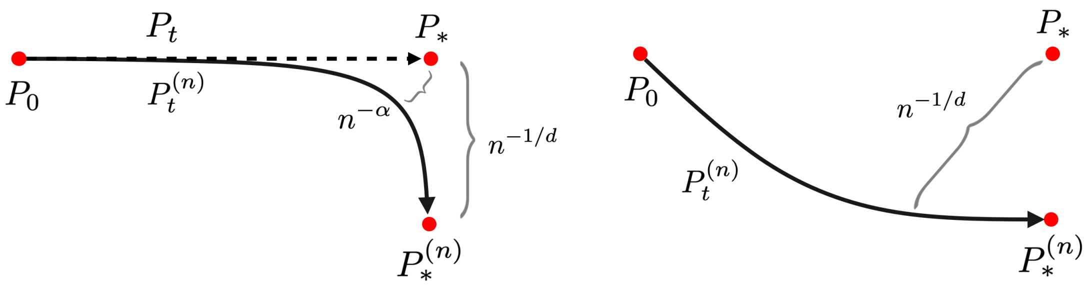

Figure 6.1: Left: Implicit regularization enables $P_t^{(n)}$ to stay close to P_t and thus approximate P_* better than $P_*^{(n)}$. Right: Wasserstein gradient flow on W_2 loss.

6.1 Bias-potential model

We start with the bias-potential model (3.25) since it enjoys the most convexity and thus the arguments are the most transparent.

Consider the domain $\Omega = [0,1]^d$ with base distribution $\mathbb{P} \in \mathcal{P}(\Omega)$. Let $V_t, V_t^{(n)} \in \mathcal{H}$ be potential functions trained on the population loss L (3.25) and the empirical loss $L^{(n)}$ (3.32) respectively, using continuous time gradient descent (3.36). Denote their induced distributions (3.3) by $P_t, P_t^{(n)}$.

Theorem 6.1 ([126, Theorem 3.3]). *Given Assumption* 3.1*, assume that the target distribution* P_* *has the form* (3.3) *with a potential function* $V_* \in \mathcal{H}$*. For any* $\delta > 0$*, with probability* $1-\delta$ *over the sampling of* $P_*^{(n)}$*,*

$$KL\big(P_* \| P_t^{(n)}\big) \leq \frac{\|V_*\|_{\mathcal{H}}^2}{2t} + \frac{8\sqrt{2\log 2d} + 2\sqrt{2\log(2/\delta)}}{\sqrt{n}}t. \tag{6.1}$$

Corollary 6.1. *Given the condition of Theorem* 6.1*, if we choose an early-stopping time* T *such that*

$$T = \Theta\left(\|V_*\|_{\mathcal{H}}\left(\frac{n}{\log d}\right)^{\frac{1}{4}}\right),$$

then the test error satisfies

$$KL\big(P_* \| P_T^{(n)}\big) \lesssim \|V_*\|_{\mathcal{H}}\left(\frac{\log d}{n}\right)^{\frac{1}{4}}.$$

Hence, the generalization error escapes from the curse of dimensionality.

The two terms in the upper bound (6.1) are the training error and generalization gap. The former is a consequence of convexity, while the latter follows from the observation that the landscapes of L and $L^{(n)}$ differ very little

$$\left\|\frac{\delta L - L^{(n)}}{\delta a}\right\|_{L^2(\rho)} \leq \sup_{\|V\|_{\mathcal{H}}\leq 1}\left|\int V(\mathbf{x})d\big(P_* - P_*^{(n)}\big)(\mathbf{x})\right|$$

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

$$\lesssim Rad_n\big(\{\|V\|_{\mathcal{H}} \leq 1\}\big) + \frac{\sqrt{\log 1/\delta}}{\sqrt{n}},$$

where the Rad_n term is the Rademacher complexity and scales as $\mathcal{O}(1/\sqrt{n})$. Then, V_t, $V_t^{(n)}$ remain close during training

$$\|V_t - V_t^{(n)}\|_{\mathcal{H}} \lesssim \int_0^t \left\| \frac{\delta L - L^{(n)}}{\delta a} \right\|_{L^2(\rho)} \lesssim \frac{t}{\sqrt{n}},$$

which confirms the depiction in Fig. 6.1 (left).

In the meantime, the bias potential model also generalizes well in the explicit regularization setting. Here we consider the Ivanov and Tikhonov regularizations [57,84,114].

Proposition 6.1 ([126, Proposition 3.9]). *Given Assumption* 3.1, *assume that the target P_* is generated by a potential $V_* \in \mathcal{H}$. Let $V_R^{(n)}$ be the minimizer of the regularized loss*

$$\min_{\|V\|_{\mathcal{H}} \leq R} L^{(n)}(V),$$

where R is any constant such that $R \geq \|V_*\|_{\mathcal{H}}$. For any $\delta > 0$, with probability $1-\delta$ over the sampling of $P_*^{(n)}$, the distribution $P_R^{(n)}$ generated by the potential $V_R^{(n)}$ satisfies

$$KL\big(P_* \| P_R^{(n)}\big) \leq \frac{8\sqrt{2\log 2d} + 2\sqrt{2\log(2/\delta)}}{\sqrt{n}} R.$$

Proposition 6.2. *Given the condition of Proposition* 6.1, *for any $\delta > 0$, let $V_\lambda^{(n)}$ be the minimizer to the regularized loss*

$$\min_{V \in \mathcal{H}} L^{(n)}(V) + \frac{\lambda}{\sqrt{n}} \|V\|_{\mathcal{H}}, \quad \lambda \geq 4\sqrt{2\log 2d} + \sqrt{2\log(2/\delta)}.$$

With probability $1-\delta$ over the sampling of $P_*^{(n)}$, the distribution $P_\lambda^{(n)}$ generated by $P_R^{(n)}$ satisfies

$$KL\big(P_* \| P_\lambda^{(n)}\big) \leq \frac{2\lambda \|V_*\|_{\mathcal{H}}}{\sqrt{n}}.$$

6.2 Normalizing flow with stochastic interpolants

Consider the normalizing flow with stochastic interpolants (3.30), and model the velocity field V and generator G_V by Definition 3.3. Denote by V_t, $V_t^{(n)}$ the training trajectories on the population and empirical losses (3.30, 3.32) using gradient flow (3.36, 3.37). Denote the generated distributions by $P_t = G_{V_t}\#\mathbb{P}$ and $P_t^{(n)} = G_{V_t^{(n)}}\#\mathbb{P}$.

First, we bound the generalization gap.

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

Theorem 6.2 ([125, Theorem 3.4]). *Given Assumption* 3.2, *for any compactly-supported base and target distributions $\mathbb{P}$ and P_*, if the velocity field V_** (3.12) *satisfies $V_* \in \mathcal{H}(\mathbb{R}^{d+1}, \mathbb{R}^d)$, then with probability $1-\delta$ over the sampling of $P_*^{(n)}$,*

$$W_2\big(P_t, P_t^{(n)}\big) \leq \|V_*\|_{\mathcal{F}} \frac{1}{\sqrt{n}} \left(\left(1 + \sqrt{2\ln 2} + \sqrt{2\ln\left(\frac{2}{\delta}\right)} \right) \left(\frac{4}{3} R t^{\frac{3}{2}} + 2Rt \right) + \sqrt{R^2 + 2t} \right),$$

where R is the radius

$$R = \sup\big\{ \|\mathbf{x}\| \,\big|\, \mathbf{x} \in sprt\mathbb{P} \cup sprt P_* \big\}.$$

Next, to estimate the generalization error, we need a sharper norm to bound the training error.

Definition 6.1 ([125, Proposition 2.4]). *Given any distribution $M \in \mathcal{P}(\mathbb{R}^{d+1})$, let K be the integral operator* (3.34) *over $L^2(M)$. Given Assumption* 3.2, [125, *Proposition* 2.4] *implies that K is a symmetric positive compact operator, and thus have an eigendecomposition with positive eigenvalues $\{\lambda_i\}_{i=1}^\infty$ and eigenfunctions $\{\phi_i\}_{i=1}^\infty$, which form an orthonormal basis of $L^2(M)$. Define the subspace $\mathcal{H}^2(M) \subseteq L^2(M)$ with the following norm:*

$$\|V\|_{\mathcal{H}^2(M)} = \sum_{i=1}^\infty \frac{\|\mathbf{v}_i\|^2}{\lambda_i^2},$$

where $\mathbf{v}_i$ are the coefficients in the decomposition $V = \sum_i \mathbf{v}_i \phi_i$. Besides, we have $\|V\|_{\mathcal{H}(sprtM)} \leq \|V\|_{\mathcal{H}^2(M)}$ and thus $\|V\|_{\mathcal{F}} \leq \exp \|V\|_{\mathcal{H}^2(M)}$.

Theorem 6.3 ([125, Theorem 3.7]). *Given Assumption* 3.2, *assume that the base distribution $\mathbb{P}$ is compactly-supported and has a C^2 density. Let P_* be any compactly-supported target distribution such that the velocity V_** (3.12) *satisfies $V_* \in \mathcal{H}^2(M_*)$, where M_* is the joint distribution* (3.10). *Then,*

$$\begin{aligned} W_2\big(P_*, P_t^{(n)}\big) \leq{}& \frac{\|V_*\|_{\mathcal{F}} \|V_*\|_{\mathcal{H}^2(M_*)}}{2\sqrt{t}} + \|V_*\|_{\mathcal{F}} \frac{1}{\sqrt{n}} \\ &\times \left(\left(1 + \sqrt{2\ln 2} + \sqrt{2\ln\left(\frac{2}{\delta}\right)} \right) \left(\frac{4}{3} R t^{\frac{3}{2}} + 2Rt \right) + \sqrt{R^2 + 2\, t} \right), \end{aligned}$$

where the radius $R = \sup\{\|\mathbf{x}\| \mid \mathbf{x} \in sprt\mathbb{P} \cup sprt P_*\}$.

In short, the generalization error scales as

$$W_2\big(P_*, P_t^{(n)}\big) \lesssim \frac{1}{\sqrt{t}} + \frac{1}{\sqrt{n}}\, t^{\frac{3}{2}},$$

and thus with early-stopping $T = \Theta(n^{1/4})$, the model escapes from the curse of dimensionality

$$W_2\big(P_*, P_T^{(n)}\big) \lesssim n^{\frac{1}{8}}.$$

The condition $V_* \in \mathcal{H}^2(M_*)$ may seem strict, so we present the following corollary that holds for general target distributions.

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

Corollary 6.2 ([125, Corollary 3.8]). *Given Assumption* 3.2*, assume that the base distribution* $\mathbb{P}$ *is compactly-supported and has* C^2 *density. Let* P_* *be any compactly-supported target distribution. For any* $\epsilon > 0$*, there exists a distribution* $M_\epsilon \in \mathcal{P}(\mathbb{R}^{d+1})$ *and a velocity field* $V_\epsilon \in \mathcal{H}^2(M_\epsilon)$ *such that*

$$\begin{aligned}W_2\big(P_*, P_t^{(n)}\big) < \frac{\|V_*\|_{\mathcal{F}}\|V_*\|_{\mathcal{H}^2(M_\epsilon)}}{2\sqrt{t}} + \epsilon\left(\|V_\epsilon\|_{\mathcal{F}}\, t^{\frac{3}{2}} + 1\right)\\ + \|V_*\|_{\mathcal{F}} \frac{1}{\sqrt{n}}\left(\left(1+\sqrt{2\ln 2}+\sqrt{2\ln\left(\frac{2}{\delta}\right)}\right)\left(\frac{4}{3}Rt^{\frac{3}{2}} + 2Rt\right) + \sqrt{R^2+2}\, t\right).\end{aligned}$$

6.3 GAN

Similar to Section 5.2, we consider the simplified GAN model such that the modeled distribution is represented by a density function P. We show that the discriminator D alone is sufficient to enable good generalization.

With the discriminator modeled by $D \in \mathcal{H}(\mathbb{R}^d, \mathbb{R})$, the GAN loss (3.20) becomes the maximum mean discrepancy L in (5.1). Consider the training trajectory $P_t \in L^2([0,1]^d)$ defined by (5.2). Similarly, define the empirical loss $L^{(n)}$ and empirical training trajectory $P_t^{(n)}$ by

$$\begin{aligned}L^{(n)}(P) &= \frac{1}{2}\iint k(\mathbf{x},\mathbf{x}')d\big(P - P_*^{(n)}\big)^2(\mathbf{x},\mathbf{x}'),\\ P_t^{(n)} &= -\nabla_P L^{(n)}\big(P_t^{(n)}\big) = -k * \big(P_t^{(n)} - P_*^{(n)}\big).\end{aligned}$$

Fix some initialization $P_0 = P_0^{(n)} \in L^2([0,1]^d)$. We measure the test error by $W_2(P_*, \Pi_\Delta(P))$, where Π_Δ is the nearest point projection onto $\mathcal{P}([0,1]^d) \cap L^2([0,1]^d)$.

Theorem 6.4 ([127, Theorem 2]). *Given Assumption* 3.1*, for any target density function* P_* *such that* $P_* - P_0 \in \mathcal{H}$*, with probability* $1-\delta$ *over the sampling of* $P_*^{(n)}$*,*

$$W_2\big(P_*, \Pi_\Delta(P_t^{(n)})\big) \leq \sqrt{d}\frac{\|P_* - P_0\|_{\mathcal{H}}}{\sqrt{t}} + \sqrt{d}\,\frac{4\sqrt{2\log 2d} + \sqrt{2\log(2/\delta)}}{\sqrt{n}} t.$$

It follows that with an early stopping time $T = \Theta(n^{1/3})$, the generalization error scales as $\mathcal{O}(n^{-1/6})$ and escapes from the curse of dimensionality.

Part of this result can be extended to the usual case with a generator. As shown in [127], if we consider the GAN loss $L(G) = L(G\#\mathbb{P})$, then the generator is insensitive to the difference between the landscapes of the population loss L and empirical loss $L^{(n)}$,

$$\left\|\frac{\delta L}{\delta G} - \frac{\delta L^{(n)}}{\delta G}\right\|_{L^2(\mathbb{P})} = \big\|\nabla k * \big(P_* - P_*^{(n)}\big)\big\|_{L^2(\mathbb{P})} \lesssim \frac{1}{\sqrt{n}}.$$

The mechanism is that the sampling error $P_* - P_*^{(n)}$ is damped by D.

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

Furthermore, we can model the generator as a random feature function, $G \in \mathcal{H}(\mathbb{R}^d, \mathbb{R}^d)$, and consider the population trajectory G_t and empirical trajectory $G_t^{(n)}$, which are trained respectively on $L, L^{(n)}$ with continuous time gradient descent (3.36,3.37) and with the same initialization. Assume that the activation σ is C^1, then the difference $G_t - G_t^{(n)}$ grows slowly at $t = 0$

$$\frac{d}{dt}\big(G_t - G_t^{(n)}\big)(\mathbf{z})\big|_{t=0} = k * \nabla\big(k * (P_* - P_*^{(n)})\big).$$

Note that the sampling error $P_* - P_*^{(n)}$ is damped twice by both G and D.

Finally, we remark that the generalization error of GANs have been studied in several related works using other kinds of simplified models, for instance when the generator G is a linear map [38], one-layer network (without hidden layer, of the form $G(\mathbf{x}) = \sigma(A\mathbf{x})$ for $A \in \mathbb{R}^{d\times d}$) [69, 121], or polynomial with bounded degree [71].

6.4 Score-based diffusion model

As a further demonstration of the techniques from the previous sections, this section presents an informal estimation of the generalization error of the score-based diffusion model (3.29). By heuristic calculations, we derive a bound in the implicit regularization setting that resembles Theorem 6.3.

Model the score function by $\mathbf{s} \in \mathcal{H}(\mathbb{R}^{d+1}, \mathbb{R}^d)$. Let $\mathbf{s}_t, \mathbf{s}_t^{(n)}$ be the training trajectories on the population loss L (3.29) and empirical loss $L^{(n)}$ (3.32) using gradient flow (3.36, 3.37) with zero initialization $\mathbf{s}_0 = \mathbf{s}_0^{(n)} \equiv \mathbf{0}$. Model the generators $G_t, G_t^{(n)}$ by the reverse-time SDE (3.16) with scores $\mathbf{s}_t, \mathbf{s}_t^{(n)}$. Denote the generated distributions by $P_t = G_t\#\mathbb{P}$ and $P_t^{(n)} = G_t^{(n)}\#\mathbb{P}$.

For any target distribution P_*, denote the target score function by $\mathbf{s}_* = \nabla \log P_\tau$ with P_τ given by (3.15). By inequality (3.31),

$$\mathrm{KL}\big(P_* \| P_t^{(n)}\big) \leq L\big(\mathbf{s}_t^{(n)}\big) - L(\mathbf{s}_*) + \mathrm{KL}(P_T \| \mathbb{P}).$$

Assume that $\mathbf{s}_* \in \mathcal{H}$. Then, by convexity (see for instance [126, Proposition 3.1])

$$L\big(\mathbf{s}_t\big) - L\big(\mathbf{s}_*\big) \lesssim \frac{\|\mathbf{s}_*\|_{\mathcal{H}}^2}{t}.$$

Meanwhile, by the growth rate bound $\|\mathbf{s}_t\|_{\mathcal{H}}, \|\mathbf{s}_t^{(n)}\|_{\mathcal{H}} \lesssim \sqrt{t}$ from [125, Proposition 5.3],

$$L\big(\mathbf{s}_t^{(n)}\big) - L\big(\mathbf{s}_t\big) \lesssim \Big(\|V_t^{(n)}\|_{C_0} + \|V_t\|_{C_0}\Big)\, \|\mathbf{s}_t - \mathbf{s}_t^{(n)}\|_{\mathcal{H}} \lesssim \sqrt{t}\|\mathbf{s}_t - \mathbf{s}_t^{(n)}\|_{\mathcal{H}}.$$

Then, using a calculation analogous to the proof of [125, Theorem 3.4]

$$\|\mathbf{s}_t - \mathbf{s}_t^{(n)}\|_{\mathcal{H}} \lesssim Rad_n\Big(\{\|\mathbf{s}\|_{\mathcal{H}} \leq \sqrt{t}\}\Big)\, t \lesssim \frac{t^{\frac{3}{2}}}{\sqrt{n}}.$$

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

Combining these inequalities, we obtain

$$\mathrm{KL}\big(P_* \| P_t^{(n)}\big) \lesssim \frac{\|\mathbf{s}_*\|_{\mathcal{H}}^2}{t} + \frac{t^2}{\sqrt{n}} + \mathrm{KL}(P_T \| \mathbb{P}).$$

Hence, if we ignore the approximation error $\mathrm{KL}(P_T\|\mathbb{P})$ due to finite T, the generalization error with early stopping scales as $\mathcal{O}(n^{-1/6})$ and escapes from the curse of dimensionality.

6.5 Normalizing flow

This section presents an informal estimation of the generalization error of the normalizing flow model (3.26). We conjecture an upper bound in the explicit regularization setting that resembles Proposition 6.1.

Let the velocity field be modeled by $V \in \mathcal{H}(\mathbb{R}^{d+1}, \mathbb{R}^d)$, let G_V be the flow map from Definition 3.3, and define the reverse-time flow map for $\tau \in [0,1]$,

$$F_V(\mathbf{x}_1, \tau) = \mathbf{x}_\tau, \quad \frac{d}{d\tau}\mathbf{x}_\tau = V(\mathbf{x}_\tau, \tau).$$

Let $L, L^{(n)}$ be the population and empirical losses (3.26,3.32). Let the base distribution be the unit Gaussian $\mathbb{P} = \mathcal{N}$. For any target distribution $P_* \in \mathcal{P}_2(\mathbb{R}^d)$ such that $P_* = G_{V_*}\#\mathbb{P}$ for some $V_* \in \mathcal{H}$, and for any $R \geq \|V_*\|_{\mathcal{F}}$, consider the problem with explicit regularization

$$\min_{\|V\|_{\mathcal{F}} \leq R} L^{(n)}(V).$$

Let $V_R^{(n)}$ be a minimizer. It follows that

$$\begin{aligned} L\big(V_R^{(n)}\big) &\leq L^{(n)}\big(V_R^{(n)}\big) + \sup_{\|V\|_{\mathcal{F}} \leq R} L(V) - L^{(n)}(V) \\ &\leq L^{(n)}(V_*) + \sup_{\|V\|_{\mathcal{F}} \leq R} L(V) - L^{(n)}(V) \\ &\leq L(V_*) + 2 \sup_{\|V\|_{\mathcal{F}} \leq R} L(V) - L^{(n)}(V). \end{aligned}$$

Then,

$$\begin{aligned} \sup_{\|V\|_{\mathcal{F}} \leq R} L(V) - L^{(n)}(V) \leq & \sup_{\|V\|_{\mathcal{F}} \leq R} \iint_0^1 \mathrm{Tr}\big[\nabla V\big(F_V(\mathbf{x}_\tau, \tau), \tau\big)\big] d\tau d\big(P_* - P_*^{(n)}\big)(\mathbf{x}) \\ & + \sup_{\|V\|_{\mathcal{F}} \leq R} \iint_0^1 \frac{1}{2} \|F_V(\mathbf{x}, 1)\|^2 d\big(P_* - P_*^{(n)}\big)(\mathbf{x}). \end{aligned}$$

Denote the two terms by the random variables A, B. Using the techniques of [31, Theorem 2.11] and [46, Theorem 3.3] for bounding the Rademacher complexity of flow-induced functions, one can try to bound the following expectations:

$$\mathbb{E}[A] \lesssim \frac{R}{\sqrt{n}} \mathbb{E}\Big[\max_{1 \leq i \leq n} \|X_i\| \;\Big|\; X_i \sim^{i.i.d.} P_*\Big] \lesssim \frac{R^2}{\sqrt{n}},$$

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

$$\mathbb{E}[B] \lesssim \frac{R^2}{\sqrt{n}} \mathbb{E}\Big[\max_{1\leq i\leq n} \|X_i\|^2 \mid X_i \sim^{i.i.d.} P_* \Big] \lesssim \frac{R^4}{\sqrt{n}}.$$

Meanwhile, for the random fluctuations $A - \mathbb{E}[A], B - \mathbb{E}[B]$, one can apply the extension of McDiarmid's inequality to sub-Gaussian random variables [67], and try to show that, with probability $1-\delta$ over the sampling of $P_*^{(n)}$,

$$A - \mathbb{E}[A] \lesssim \frac{R^2\sqrt{\log 1/\delta}}{\sqrt{n}}, \quad B - \mathbb{E}[B] \lesssim \frac{R^4\sqrt{\log 1/\delta}}{\sqrt{n}}.$$

Combining these inequalities, one can conjecture that the solution $P_R^{(n)} = G_{V_R^{(n)}}\#\mathbb{P}$ satisfies

$$\mathrm{KL}\big(P_* \| P_R^{(n)}\big) = L\big(V_R^{(n)}\big) - L(V_*) \lesssim \frac{1+\sqrt{\log 1/\delta}}{\sqrt{n}} R^4.$$

7 Training

This section studies the training behavior of distribution learning models, and illustrates the differences between the three distribution representations discussed in Section 3.2. On one hand, we survey our results on the global convergence rates of models with the potential representation and fixed generator representation. On the other hand, we present new results on the landscape of models with the free generator representation, and analyze the mode collapse phenomenon of GANs.

7.1 Potential and fixed generator

Models with these two representations are easier to analyze since their losses are usually convex over abstract functions. Specifically, this section considers the bias-potential model (3.25) and normalizing flow with stochastic interpolants (3.30)

$$\begin{aligned} &L(V) = \int V dP_* + \ln \int e^{-V} d\mathbb{P}, \\ &L(V) = \frac{1}{2}\int_0^1 \iint \big\|V\big((1-\tau)\mathbf{x}_0 + \tau\mathbf{x}_1, \tau\big) - (\mathbf{x}_1 - \mathbf{x}_0)\big\|^2 d\mathbb{P}(\mathbf{x}_0) dP_*(\mathbf{x}_1) d\tau. \end{aligned}$$

If we choose a convex function representation for V, then the optimization problem becomes convex.

To estimate the rate of convergence, one approach is to bound the test error by a stronger norm. The following toy example shows how to bound the L^2 error by the RKHS norm $\|\cdot\|_{\mathcal{H}}$.

Example 7.1 (Kernel regression). Fix any base distribution $\mathbb{P} \in \mathcal{P}(\mathbb{R}^d)$ and assume that the activation σ is bounded. For any target function $f_* \in \mathcal{H}$, consider the regression loss

$$L(f) = \frac{1}{2}\|f - f_*\|^2_{L^2(\mathbb{P})}.$$

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

Parameterize f by $f_a \in \mathcal{H}$ and train the parameter function a by continuous time gradient descent (3.35) with initialization $a \equiv 0$. Then,

$$L(f_t) \leq \frac{\|f_*\|_{\mathcal{H}}^2}{2t}.$$

Proof one. Denote the loss by $L(a) = L(f_a)$. Since σ is bounded, L has continuous Fréchet derivative in $a \in L^2(\rho)$, so the gradient descent (3.35) is well-defined. Choose $a_* \in L^2(\rho)$ such that $f_* = f_{a_*}$ and $\|f_*\|_{\mathcal{H}} = \|a_*\|_{L^2(\rho)}$. Define the Lyapunov function

$$E(t) = t\left(L(a_t) - L(a_*)\right) + \frac{1}{2}\|a_* - a_t\|_{L^2(\rho_0)}^2.$$

Then,

$$\begin{aligned}\frac{d}{dt}E(t) &= \left(L(a_t) - L(a_*)\right) + t \cdot \frac{d}{dt}L(a_t) + \left\langle a_t - a_*, \frac{d}{dt}a_t \right\rangle_{L^2(\rho_0)} \\ &\leq \left(L(a_t) - L(a_*)\right) - \left\langle a_t - a_*, \nabla L(a_t)\right\rangle_{L^2(\rho_0)}.\end{aligned}$$

By convexity, for any a_0, a_1,

$$L(a_0) + \langle a_1 - a_0, \nabla L(a_0)\rangle \leq L(a_1).$$

Hence, $(d/dt)E \leq 0$. We conclude that $E(t) \leq E(0)$. □

Proof two. Since a_t evolves by (3.35), f_t evolves by (3.36)

$$\frac{d}{dt}f_t = -K(f_t - f_*), \tag{7.1}$$

where K is the integral operator (3.34) over $L^2(\mathbb{P})$. Since K is symmetric, positive semidefinite and compact, there exists an eigendecomposition with non-negative eigenvalues $\{\lambda_i\}_{i=1}^\infty$ and eigenfunctions $\{\phi_i\}_{i=1}^\infty$ that form an orthonormal basis of $L^2(\mathbb{P})$. Consider the decomposition

$$f_t = \sum_{i=1}^\infty c_t^i \phi_i, \quad f_* = \sum_{i=1}^\infty c_*^i \phi_i.$$

It is known that the RKHS norm satisfies [26, 93]

$$\|f_*\|_{\mathcal{H}}^2 = \sum_{i=1}^\infty \frac{(c_*^i)^2}{\lambda_i}.$$

Since $c_0^i = 0$ by assumption, (7.1) implies that $c_t^i = (1 - e^{-\lambda_i t})c_*^i$. Hence,

$$\begin{aligned}L(f_t) &= \frac{1}{2}\sum_{i=1}^\infty \left(c_t^i - c_*^i\right)^2 = \frac{1}{2}\sum_{i=1}^\infty \left(c_*^i\right)^2 e^{-2\lambda_i t} \\ &\leq \frac{1}{2}\sum_{i=1}^\infty \frac{(c_*^i)^2}{\lambda_i} \sup_{\lambda \geq 0} \lambda e^{-2\lambda t} = \frac{\|f_*\|_{\mathcal{H}}^2}{4et}.\end{aligned}$$

It completes the proof. □

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

Using similar arguments, we have the following bounds on the test error of models trained on the population loss.

Proposition 7.1 (Bias-potential, [126, Proposition 3.1]). *Given the setting of Theorem* 6.1*, the distribution P_t generated by the potential V_t satisfies*

$$KL(P_*\|P_t) \leq \frac{\|V_*\|^2_{\mathcal{H}}}{2t}.$$

Proposition 7.2 (NF, [125, Proposition 3.5]). *Given the setting of Theorem* 6.3*, the distribution $P_t = G_{V_t}\#\mathbb{P}$ generated by the trajectory V_t satisfies*

$$W_2(P_*, P_t) \leq \frac{\|V_*\|_{\mathcal{F}}\|V_*\|_{\mathcal{H}^2(M_*)}}{2\sqrt{t}}.$$

Corollary 7.1 (NF, [125, Corollary 3.6]). *Given the setting of Corollary* 6.2*, let P_* be any compactly-supported target distribution and V_* be the target velocity field* (3.12)*. For any $\epsilon > 0$, there exists a distribution $M_\epsilon \in \mathcal{P}(\mathbb{R}^{d+1})$ and velocity field $V_\epsilon \in \mathcal{H}^2(M_\epsilon)$ such that*

$$W_2(P_*, P_t) < \frac{\|V_*\|_{\mathcal{F}}\|V_*\|_{\mathcal{H}^2(M_\epsilon)}}{2\sqrt{t}} + \epsilon\Big(\|V_\epsilon\|_{\mathcal{F}} t^{\frac{3}{2}} + 1\Big).$$

It seems probable that the $\mathcal{O}(\epsilon t^{3/2})$ term can be strengthened to $\mathcal{O}(\epsilon)$, which would imply universal convergence, i.e. convergence to any target distribution.

7.2 Free generator: Landscape

Models with the free generator representation are more difficult to analyze, since the loss $L(G)$ is not convex in G. For instance, it is straightforward to check that if $\mathbb{P}$ and P_* are uniform over $[0,1]$, then the solution set $\{G_* \,|\, G_*\#\mathbb{P} = P_*\}$, or equivalently the set of minimizers of L, is an infinite and non-convex subset of $L^2(\mathbb{P})$, and thus L is non-convex.

Despite the non-convexity, there is still hope for establishing global convergence through a careful analysis of the critical points. For instance, one can conjecture that there are no spurious local minima and that the saddle points can be easily avoided. This section offers two results towards this intuition: a characterization of critical points for general loss functions of the form $L(G\#\mathbb{P})$, and a toy example such that global convergence can be determined from initialization.

In general, no matter how we parameterize the generator, the modeled distribution $P_t = G_t\#\mathbb{P}$ satisfies the continuity equation during training: Assume that $G(\mathbf{x},\theta)$ is C^1 in $\mathbf{x} \in \mathbb{R}^d$ and $\theta \in \Theta$, where Θ is some Hilbert space. Then, given any path θ_t that is C^1 in t, for any smooth test function ϕ,

$$\begin{aligned}\frac{d}{dt}\int \phi dP_t &= \frac{d}{dt}\int \phi\big(G(\mathbf{x},\theta_t)\big)d\mathbb{P}(\mathbf{x})\\ &= \int \nabla\phi\big(G(\mathbf{x},\theta_t)\big)\cdot\nabla_\theta G(\mathbf{x},\theta_t)\dot{\theta}_t d\mathbb{P}(\mathbf{x})\\ &= \int \nabla\phi(\mathbf{x})\cdot V_t(\mathbf{x})dP_t(\mathbf{x}) = -\int \phi d\nabla\cdot(V_tP_t),\end{aligned}$$

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

where the velocity field V_t is defined by

$$\int \mathbf{f}(\mathbf{x}) \cdot V_t(\mathbf{x})\, dP_t(\mathbf{x}) = \int \mathbf{f}\big(G(\mathbf{x},\theta_t)\big) \cdot \nabla_\theta G(\mathbf{x},\theta_t)\dot{\theta}_t\, d\mathbb{P}(\mathbf{x})$$

for any test function $\mathbf{f} \in L^2(P_t, \mathbb{R}^d)$. Thus, P_t is a weak solution to the continuity equation

$$\partial_t P_t + \nabla \cdot (V_t P_t) = 0.$$

In particular, the equation implies that no matter how G is parameterized, the "particles" of P_t during training can only move continuously without jumps or teleportation.

For abstraction, it is helpful to consider the Wasserstein gradient flow [3, 101]: For any loss function L over $\mathcal{P}_2(\mathbb{R}^d)$ and any initialization $P_0 \in \mathcal{P}_2(\mathbb{R}^d)$, define the training trajectory P_t by

$$\partial_t P_t + \nabla \cdot \left(P_t \nabla \frac{\delta L}{\delta P}(P_t) \right) = 0,$$

where $\delta_P L$ is the first variation, which satisfies

$$\frac{d}{d\epsilon} L\big(P + \epsilon(Q - P)\big)\big|_{\epsilon=0} = \int \frac{\delta L}{\delta P}(P_t) d(Q - P)$$

for any bounded and compactly-supported density function Q.

The Wasserstein gradient flow abstracts away the parametrization $\theta \mapsto G_\theta$ of the generator, and thus simplifies the analysis of critical points. To relate to our problem, the following result shows that the Wasserstein gradient flow often shares the same critical points as the parameterized loss $L(\theta) = L(G_\theta\#\mathbb{P})$, and thus we are allowed to study the former instead.

Proposition 7.3 (Comparison of critical points). *Given any loss L over $\mathcal{P}_2(\mathbb{R}^d)$, assume that the first variation $\delta_P L$ exists, is C^1, and $\nabla_{\mathbf{x}} \delta_P L(P) \in L^2(P, \mathbb{R}^d)$ for all $P \in \mathcal{P}_2(\mathbb{R}^d)$. Define the set of critical points*

$$CP = \left\{ P \in \mathcal{P}_2(\mathbb{R}^d) \;\middle|\; \nabla_{\mathbf{x}} \delta_P L(P)(\mathbf{x}) = \mathbf{0} \text{ for } P \text{ almost all } \mathbf{x} \right\}.$$

Given any base distribution $\mathbb{P} \in \mathcal{P}_2(\mathbb{R}^k)$ and any generator $G_\theta : \mathbb{R}^k \to \mathbb{R}^d$ parameterized by $\theta \in \Theta$, where Θ is a Hilbert space. Assume that $G_\theta(\mathbf{x})$ is Lipschitz in $\mathbf{x}$ for any θ, and C^1 in θ for any $\mathbf{x}$, and that the gradient $\nabla_\theta G_\theta$ at any θ is a continuous linear operator $\Theta \to L^2(\mathbb{P}, \mathbb{R}^d)$. Define the set of generated distributions $\mathcal{P}_\Theta = \{G_\theta\#\mathbb{P},\ \theta \in \Theta\}$ and the set of critical points

$$CP_\Theta = \left\{ G_\theta\#\mathbb{P} \;\middle|\; \theta \in \Theta, \nabla_\theta L(G_\theta\#\mathbb{P}) = 0 \right\}.$$

Then, $CP \cap \mathcal{P}_\Theta \subseteq CP_\Theta$. Furthermore, if the parametrization G_θ is the random feature functions $G_{\mathbf{a}} \in \mathcal{H}(\mathbb{R}^k, \mathbb{R}^d)$, and either Assumption 3.1 or 3.2 holds, then

$$CP_\Theta = CP \cap \mathcal{P}_\Theta.$$

Below is an example use case of this proposition.

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

Example 7.2. Consider the GAN loss $L(P)$ in (5.1) induced by discriminators that are random feature functions (3.33). Assume that the target distribution $P_* \in \mathcal{P}_{2,ac}(\mathbb{R}^d)$ is radially symmetric, the parameter distribution ρ in (3.33) is radially symmetric in $\mathbf{w}$ conditioned on any b, and that the activation σ is C^1. Consider the point mass $P = \delta_{\mathbf{0}}$. Then,

$$\begin{aligned}
\nabla \frac{\delta L}{\delta P}(P)(\mathbf{0}) &= \nabla \int k(\mathbf{x},\mathbf{x}')d(P-P_*)(\mathbf{x}')\big|_{\mathbf{x}=\mathbf{0}} \\
&= \iint \mathbf{w}\sigma'(\mathbf{w}\cdot\mathbf{0}+b)\sigma(\mathbf{w}\cdot\mathbf{x}'+b)d(P-P_*)(\mathbf{x}')d\rho(\mathbf{w},b) \\
&= \frac{1}{2}\iint \mathbf{w}\sigma'(b)\sigma(\mathbf{w}\cdot\mathbf{x}'+b)d(P-P_*)(\mathbf{x}')d\rho(\mathbf{w}|b)d\rho(b) \\
&\quad + \frac{1}{2}\iint (-\mathbf{w})\sigma'(b)\sigma\big((-\mathbf{w})\cdot(-\mathbf{x}')+b\big)d(P-P_*)(\mathbf{x}')d\rho(\mathbf{w}|b)d\rho(b) \\
&= \mathbf{0}.
\end{aligned}$$

Thus, $\nabla\delta_P L(P) = \mathbf{0}$ for P almost all $\mathbf{x}$, and $P \in CP$. It follows from Proposition 7.3 that P is a critical point ($P \in CP_\Theta$) for any parameterized generator G_θ that can express the constant zero function.

A probability measure P is called singular if P cannot be expressed as a density function ($P \in \mathcal{P}(\mathbb{R}^d) - \mathcal{P}_{ac}(\mathbb{R}^d)$). The critical points of an expectation-based loss are often singular distributions such as $\delta_{\mathbf{0}}$ from the previous example. The following toy model shows that the critical points may consist of only global minima and saddle points that are singular.

Consider a one-dimensional setting. Model the generator by any function $G \in L^2(\mathbb{P},\mathbb{R})$, where the base distribution $\mathbb{P}$ is conveniently set to be uniform over $[0,1]$. Given any target distribution $P_* \in \mathcal{P}_{2,ac}(\mathbb{R})$ and any initialization $G_0 \in L^2(\mathbb{P},\mathbb{R})$, consider the dynamics

$$\frac{d}{dt}G_t(z) = \int x \, d\pi_t\big(x\big|G_t(z)\big) - G_t(z), \tag{7.2}$$

where $\pi_t(\cdot|\cdot)$ is the conditional distribution of the optimal transport plan $\pi_t \in \mathcal{P}(\mathbb{R}\times\mathbb{R})$ between $P_t = G_t\#\mathbb{P}$ and P_*. By Brennier's theorem [117], since P_* is absolutely continuous, the optimal transport plan is unique.

Note that if P_t is also absolutely continuous, then Brennier's theorem implies that the transport plan is deterministic: $\pi(\cdot|x_0) = \delta_{\nabla\phi(x_0)}$ for some convex potential ϕ. Since the first variation of $W_2^2(P,P_*)$ is exactly the potential ϕ [101], the dynamics (7.2) when restricted to $\{G \,|\, G\#\mathbb{P} \in \mathcal{P}_{ac}(\mathbb{R})\}$ becomes equivalent to the gradient flow on the loss

$$L(G) = W_2^2(G\#\mathbb{P}, P_*).$$

Since the dynamics (7.2) is not everywhere differentiable, we consider stationary points instead of critical points, and extend the definition of saddle points: A stationary point $G \in L^2(\mathbb{P},\mathbb{R}^d)$ is a generalized saddle point if for any $\epsilon > 0$, there exists a perturbation h ($\|h\|_{L^2(\mathbb{P})} < \epsilon$) such that $L(G+h) < L(G)$.

Proposition 7.4. *For any target distribution $P_* \in \mathcal{P}_{2,ac}(\mathbb{R})$, the stationary points of the dynamics* (7.2) *consist only of global minima and generalized saddle points. If G is a generalized saddle point,*

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

then $G\#\mathbb{P}$ is a singular distribution, and there exists x such that $(G\#\mathbb{P})(\{x\}) > 0$. Moreover, global convergence holds

$$\lim_{t\to\infty} W_2(P_*, P_t) = 0,$$

if and only if the initialization $P_0 \in \mathcal{P}_{2,ac}(\mathbb{R})$.

This toy example confirms the intuition that despite the loss $L(G)$ is nonconvex, global convergence is still achievable. Moreover, all saddle points have one thing in common, that part of the mass has collapsed onto one point.

7.3 Free generator: Mode collapse

The previous section has presented a general study of the critical points of models with free generator, while this section focuses on a particular training failure that is common to GANs, the mode collapse phenomenon. Mode collapse is characterized as the situation when the generator G_t during training maps a positive amount of mass of $\mathbb{P}$ onto the same point [66,77,90,100], and is identified as the primary obstacle for GAN convergence. This characterization is analogous to the saddle points from Proposition 7.4 that are also singular distributions. Despite that in the setting of Proposition 7.4, the singular distributions can be avoided and the toy model enjoys global convergence, how mode collapse occurs in practice remains a mystery.

To provide some insight into the mode collapse phenomenon, we demonstrate with toy examples two mechanisms that can lead to mode collapse.

Denote by $U[x,y]$ the uniform distribution over the interval $[x,y]$. Let the base and target distributions be $\mathbb{P} = P_* = U[0,1]$. Model the generator by $G(x) = ax$, and discriminator by $D(x) = b\phi(x)$ for some ϕ to be specified. Consider the following GAN loss based on (3.20):

$$\min_a \max_b L(a,b) = \int D\, d(G\#\mathbb{P} - P_*) + \frac{c}{2}|b|^2,$$

where $c \geq 0$ is the strength of regularization. Train a, b by continuous time gradient flow

$$\frac{d}{dt}a_t = -\partial_a L(a_t, b_t), \quad \frac{d}{dt}b_t = \partial_b L(a_t, b_t)$$

with initialization $a_0 > 0$ and $b_0 = 0$. Mode collapse happens when $G_t\#\mathbb{P} = [-a_t, a_t]$ becomes a singular distribution, i.e. when $a_t = 0$.

Case one. This example shows that a non-differentiable discriminator can lead to mode collapse. Set

$$\phi(x) = |x| = \frac{\text{ReLU}(x) + \text{ReLU}(-x)}{2}.$$

Restricting to the half space $\{(a,b), a > 0\}$, the loss and training dynamics become

$$\begin{aligned} &L(a,b) = \frac{a-1}{2}b - \frac{cb^2}{2}, \\ &\frac{d}{dt}a_t = -\frac{b}{2}, \quad \frac{d}{dt}b_t = \frac{a-1}{2} - cb. \end{aligned}$$

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

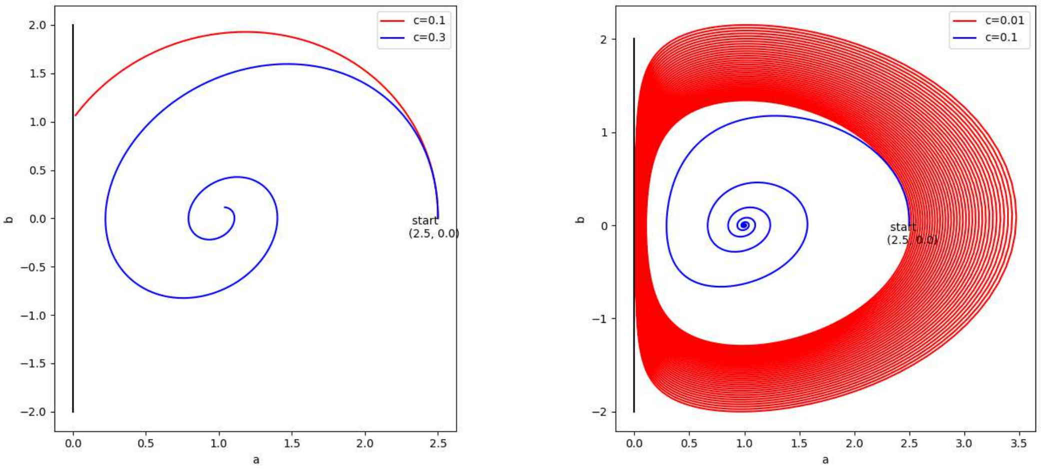

Figure 7.1: Mode collapse during GAN training. Left: Case 1. Right: Case 2. The horizontal and vertical axes are the parameters a and b. The red curves are trained with small regularization c and end up in mode collapse $a = 0$, while the blue curves have larger c and converge to the stationary point $(1, 0)$. The initialization is $(2.5, 0)$ and the learning rate for Case 2 is $\gamma = 0.1$.

Assume that $c \in [0, 1)$. Then, the unique solution is given by

$$\begin{aligned} a_t &= 1 + (a_0 - 1)e^{-\frac{ct}{2}}\Big(\cos\big(\sqrt{1-c^2}\,t/2\big) + \frac{c}{\sqrt{1-c^2}}\sin\big(\sqrt{1-c^2}\,t/2\big)\Big), \\ b_t &= \frac{a_0}{\sqrt{1-c^2}}e^{-\frac{ct}{2}}\sin\big(\sqrt{1-c^2}\,t/2\big). \end{aligned}$$

At time $T = 2\pi/\sqrt{1-c^2}$, we have

$$a_T = 1 - (a_0 - 1)e^{-\frac{c\pi}{\sqrt{1-c^2}}}.$$

Thus, if $a_0 > 1 + e^{c\pi/\sqrt{1-c^2}}$, then the trajectory a_t must hit the line $\{a = 0\}$ at some $t \in (0, T)$, and thus mode collapse happens.

This process is depicted in Fig. 7.1 (left). In general, one can conjecture that mode collapse may occur at the locations where ∇D_t is discontinuous.

Case two. This example shows that the accumulation of numerical error due to the finite learning rate can lead to mode collapse. This result is analogous to [79] which shows that if P_t is a point mass, numerical error can cause it to diverge to infinity.

Set $\phi = (1/2)x^2$. The loss and training dynamics become

$$\begin{aligned} L(a, b) &= \frac{a^2 - 1}{6}b - \frac{cb^2}{2}, \\ \frac{d}{dt}a_t &= -\frac{ab}{3}, \quad \frac{d}{dt}b_t = \frac{a^2 - 1}{6} - cb. \end{aligned}$$

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

Given a learning rate $\gamma > 0$, consider the discretized training dynamics

$$a_{(k+1)\gamma} = a_{k\gamma} - \gamma \frac{a_{k\gamma} b_{k\gamma}}{3}, \quad b_{(k+1)\gamma} = b_{k\gamma} + \gamma \left(\frac{a_{k\gamma}^2 - 1}{6} - c b_{k\gamma} \right)$$

for every $k \in \mathbb{N}$.

In general, given a discrete dynamics in the form of $\theta_{(k+1)\gamma} = \theta_{k\gamma} + \gamma f(\theta_{k\gamma})$, one can fit the sequence $\theta_{k\gamma}$ by the continuous time solution of an ODE $(d/dt)\theta_t = \tilde{f}(\theta_t)$, where $\tilde{f} = f + \gamma h$ for some function h. Assume that the two trajectories match at each $t = k\gamma$,

$$\begin{aligned} f(\theta_{k\gamma}) &= \gamma^{-1}(\theta_{(k+1)\gamma} - \theta_{k\gamma}) \\ &= \gamma^{-1} \int_0^\gamma \tilde{f}(\theta_{k\gamma+s}) ds = \int_0^1 (f + \gamma h)(\theta_{(k+s)\gamma}) ds \\ &= \int_0^1 f(\theta_{k\gamma}) + s\gamma \nabla f(\theta_{k\gamma}) \cdot f(\theta_{k\gamma}) + \gamma h(\theta_{k\gamma}) + \mathcal{O}(\gamma^2) ds \\ &= f(\theta_{k\gamma}) + \frac{\gamma}{2} \nabla f(\theta_{k\gamma}) \cdot f(\theta_{k\gamma}) + \gamma h(\theta_{k\gamma}) + \mathcal{O}(\gamma^2). \end{aligned}$$

It follows that

$$\tilde{f} = f - \frac{\gamma}{2} \nabla f \cdot f + \mathcal{O}(\gamma^2).$$

Plugging in $\theta_t = [a_t, b_t]$, we obtain the following approximate ODE:

$$\begin{aligned} \frac{d}{dt} a_t &= -\frac{ab}{3} + \gamma \left(-\frac{ab^2}{18} + \frac{a(a^2-1)}{36} - \frac{cab}{6} \right), \\ \frac{d}{dt} b_t &= \frac{a^2-1}{6} - cb + \gamma \left(\frac{a^2 b}{18} + \frac{c(a^2-1)}{12} - \frac{c^2 b}{2} \right). \end{aligned}$$

Define the energy function

$$H(a,b) = \frac{b^2}{2} + \frac{a^2-1}{4} - \frac{\log a}{2}.$$

Then

$$\frac{d}{dt} H(a_t, b_t) = -cb^2 + \gamma \left(\frac{(a^2+1)b^2}{36} + \frac{(a^2-1)^2}{72} - \frac{c^2 b^2}{2} \right).$$

Thus, if $c \leq \min(1/17, \gamma/144)$,

$$\frac{d}{dt} H(a_t, b_t) \geq \frac{\gamma}{72} \left[(a^2+1)b^2 + (a^2-1)^2 \right] > 0.$$

The energy H is strictly convex and its only minimizer is the stationary point $(a,b) = (1,0)$ where $H = 0$. Assume that the initialization $(a_0, b_0) \neq (1,0)$ and $a_0 > 0$. Since the energy is non-decreasing, $H(a_t, b_t) \geq H(a_0, b_0) > 0$, and thus the trajectory will never enter the set $S_0 = \{(a,b), H(a,b) < H(a_0,b_0)\}$. Since S_0 is an open set that contains $(1,0)$, there exists $r > 0$ such that the open ball $B_r((1,0)) \subseteq S_0$. Since $a_t \geq 0$, the trajectory (a_t, b_t) satisfies

$$\frac{d}{dt}H(a_t,b_t) \geq \frac{\gamma}{72}\big[(a^2+1)b^2 + (a+1)^2(a-1)^2\big] \geq \frac{\gamma r^2}{72}$$

and thus

$$H(a_t,b_t) \geq H(a_0,b_0) + \frac{\gamma r^2}{72}t.$$

Consider the four subsets

$$\begin{aligned} S_1 &= \{b \leq 0, a > 1\}, & S_2 &= \{b > 0, 0 < a \leq 1\}, \\ S_3 &= \{b \geq 0, 0 < a < 1\}, & S_4 &= \{b < 0, a \geq 1\} \end{aligned}$$

that partition the space $\{(a,b)\,|\,(a,b) \neq (1,0), a > 0\}$. Then, the trajectory (a_t, b_t) repeatedly moves from S_1 to S_2 to S_3 to S_4 and back to S_1. In particular, it crosses the line $\{0 < a < 1, b = 0\}$ for times $t_1, t_2 \ldots$ that go to infinity. It follow that

$$\begin{aligned} \inf_{t\geq 0} a_t \leq \inf_{i\in\mathbb{N}} a_{t_i} &\leq \inf_{i\in\mathbb{N}} \exp\left[-2\left(\frac{-\log a_{t_i}}{2}\right)\right] \\ &\leq \inf_{i\in\mathbb{N}} \exp\big[-2H(a_{t_i}, b_{t_i})\big] \\ &\leq \inf_{i\in\mathbb{N}} \exp\left[-2\left(H(a_0,b_0) + \frac{\gamma r^2}{72}t_i\right)\right] = 0. \end{aligned}$$

Hence, a_t converges to 0 exponentially fast, and then numerical underflow would lead to mode collapse. This process is depicted in Fig. 7.1 (right).

In summary, these two examples indicate two possible causes for mode collapse. On one hand, if the discriminator is non-smooth, then the gradient field ∇D_t may squeeze the distribution $G_t\#\mathbb{P}$ at the places where ∇D_t is discontinuous and thus form a singular distribution. On the other hand, the numerical error due to the finite learning rate may amplify the oscillatory training dynamics of G_t and D_t, and if this error is stronger than the regularization on D_t, then the norm of G_t can diverge and lead to collapse.

Note that, however, if we consider two-time-scale training such that b_t is always the maximizer, then the loss becomes proportional to $L(a) = (|a|-1)^2$ or $(a^2-1)^2$, and training converges to the global minimum as long as $a_0 \neq 0$.

8 Discussion

This paper studied three aspects of the machine learning of probability distributions.

First, we proposed a mathematical framework for generative models and density estimators that allows for a unified perspective on these models. Abstractly speaking, the diversity of model designs mainly arises from one factor, the vertical or horizontal way of modeling discussed in Section 3.1. When applied to distribution representation, it leads to the options of potential representation and transport representation, and the latter ramifies into the free generator and fixed generator depending on whether a probabilistic coupling can be fixed. Similarly, when applied to the loss, it leads to three loss types, depending on whether the difference is measured vertically or horizontally, with or without a fixed target. Then, the rest of the design process is to try to realize each category in Table 3.1 by satisfying the various constraints of implementation, for instance, whether to compute the density of $G\#\mathbb{P}$ directly or indirectly, and which random path is chosen to achieve the product coupling $\mathbb{P} \times P_*$. Thereby, all the major models are derived. By isolating the factors of distribution representation, loss type and function representation, this framework allows for a more transparent study of training and landscape (who depend more on distribution representation and loss type) and generalization error (which depends more on function representation).

Second, we studied the seeming conflict between the memorization phenomenon and the generalization ability of the models, and reviewed our results that resolve this conflict. On one hand, we confirmed that the models satisfy the universal approximation property (some models even enjoy universal convergence) and thus memorization is inevitable. On the other hand, function representations defined by expectations are insensitive to the sampling error $P_* - P_*^{(n)}$, so that the training trajectory tends to approximate the hidden target distribution before eventually diverging towards memorization. In particular, our results established generalization error bounds of the form $\mathcal{O}(n^{-\alpha})$ with $\alpha = 1/4, 1/6, 1/8$ for several models. There should be room for improvement, but for now we are content that the models can escape from the curse of dimensionality. Considering that this generalization ability is mostly an effect of the function representations, it seems reasonable to expect that good generalization is enjoyed by all models regardless of the choice of distribution representation and loss type.

Third, we discussed the training dynamics and loss landscape. For the potential and fixed generator representations, the convexity of their distribution parametrizations and loss functions enable the estimation of the rates of global convergence. For the free generator representation, despite the loss is non-convex, we demonstrated that the critical points have tractable forms, and also identified two general mechanisms common to the min-max training of GANs that can provably lead to mode collapse. It seems worthwhile to devote more effort in the design of models with the fixed generator representation, since they are as expressive as the free generator models while their convexity greatly eases training. It is not clear at this moment whether the product coupling $\mathbb{P} \times P_*$ is the best choice for the fixed generators, and whether the diffusion SDE and linear interpolants are the most effective random paths, so there is much to explore.

To conclude, we list a few interesting topics that have not been covered in this paper.

- Unstructured data: Our analysis was conducted only in the Euclidean space, whereas most of the applications of distribution learning models involve unstructured data such as images, texts and molecules. One thing of practical importance is that the performance of generative models for unstructured data is judged by human perception or perceptual loss [59], which can greatly differ from the Euclidean metric and thus the W_2 metric. To train the model to have higher fidelity, one approach is to use the adversarial loss of GANs such that the hidden features of the discriminators can be seen as an embedding space that captures fidelity. A related approach is to rely on a pretrained feature embedding as in [53, 96].

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02
OPEN ACCESS

- Prior knowledge: For supervised tasks involving unstructured data, it is often helpful to instill prior knowledge from humans into the models through self-supervised pretraining. A well-known example is the approximate invariance of image classification with respect to color distortion and cropping, and models that are pretrained to be insensitive to these augmentations can achieve higher test accuracy after training [18, 22, 42]. It could be beneficial to try to boost distribution learning models by prior knowledge. One example is given by [131] such that a generative model for sampling a thermodynamic system is designed to respect the spatial symmetry of the system.
- Conditional generation: In practice, people are more interested in estimating conditional distributions, e.g. generate images conditioned on text descriptions [94, 96]. Incorporating a context variable can be done by simply allowing an additional input variable in the parameter functions [80], but it can also be accomplished with tricks that minimize additional training [107].
- Factor discovery: For generative models, instead of using $G\#\mathbb{P}$ as a blackbox for sampling, it could be useful to train G in a way such that $\mathbb{P}$ has semantic meaning, e.g. for image synthesis, given a random image $G(Z), Z \sim \mathbb{P}$ of a face, one coordinate of Z may control hair style and another may control head orientation. This unsupervised task is known as factor discovery [110], and some solutions are provided by [23,48,62] with application to semantic photo editing [74].
- Density distillation: The basic setting considered in this paper is to estimate a target distribution given a sample set; yet, another task common to scientific computing is to train a generative model to sample from a distribution $(1/Z)e^{-V}$ given a potential function V. One popular approach [17, 70, 82, 131] is to use a modified normalizing flow with the reverse KL-divergence.

9 Proofs

9.1 Loss function

Proof of Proposition 3.1. For any $P_*, P \in \mathcal{P}_{ac}(\mathbb{R}^d)$, the assumption on global minimum implies that

$$\lim_{t\to 0^+} \frac{1}{t}\big(L((1-t)P_* + tP) - L(P_*)\big) = \int f'(P(\mathbf{x}))(P(\mathbf{x}) - P_*(\mathbf{x}))dP_*(\mathbf{x}) \geq 0.$$

Define the map $g(p) = pf'(p)$. Then,

$$\mathbb{E}_P\big[g(P_*(\mathbf{x}))\big] \geq \mathbb{E}_{P_*}\big[g(P_*(\mathbf{x}))\big].$$

Assume for contradiction that g is nonconstant on $(0,\infty)$, then there exist $a, b > 0$ such that $g(a) < g(b)$. Let $A, B \subseteq \mathbb{R}^d$ be two disjoint hyperrectangles with volumes $1/2a$ and $1/2b$. Define P as the uniform distribution over A, and P_* as

$$P_* = a\mathbf{1}_A + b\mathbf{1}_B.$$

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

Then the above inequality is violated. It follows that g is constant and f has the form $c\log p + c'$. Finally, the assumption on global minimum implies that $c \leq 0$. □

9.2 Universal approximation theorems

Proof of Proposition 4.1. By Brennier's theorem ([14] and [117, Theorem 2.12]), since both $\mathbb{P}$ and P_* have finite second moments and $\mathbb{P}$ is absolutely continuous, there exists an $L^2(\mathbb{P}, \mathbb{R}^d)$ function G_* such that $G_* \# \mathbb{P} = P_*$. By Lemma 9.1, there exists a sequence $\{G_n\}_{n=1}^\infty \subset \mathcal{H}$ that converges to G_* in $L^2(\mathbb{P})$ norm. Hence,

$$\lim_{n\to\infty} W_2(G\#\mathbb{P}, G_n\#\mathbb{P}) \leq \lim_{n\to\infty} \|G - G_n\|_{L^2(\mathbb{P})} = 0.$$

The proof is complete. □

The preceding proof uses the following lemma, which is a slight extension of the classical universal approximation theorem [51, 52] to cases with possibly unbounded base distributions $\mathbb{P}$. Such extension is needed since in practice $\mathbb{P}$ is usually set to be the unit Gaussian.

Lemma 9.1. *Given either Assumption* 3.1 *or* 3.2, *for any* $\mathbb{P} \in \mathcal{P}(\mathbb{R}^d)$ *and any* $k \in \mathbb{N}$, *the space* $\mathcal{H}(\mathbb{R}^d, \mathbb{R}^k)$ *is dense in* $L^2(\mathbb{P}, \mathbb{R}^k)$.

Proof. It suffices to consider the case with output dimension $k = 1$. Denote $\mathcal{H}$ by $\mathcal{H}_\sigma$ to emphasize the choice of the activation σ.

Given Assumption 3.1, the activation σ is ReLU. Define

$$\sigma_*(x) = \int \sigma(x+1+\epsilon) - 2\sigma(x+\epsilon) + \sigma(x-1+\epsilon) dh(\epsilon),$$

where h is a continuous distribution supported in $[-0.1, 0.1]$. Since ReLU is homogeneous, σ_* can be expressed by (3.33) with some bounded parameter function a, and thus $\sigma_* \in \mathcal{H}_\sigma(\mathbb{R}, \mathbb{R})$. Similarly, given Assumption 3.2, the activation σ is sigmoid. Define

$$\sigma_*(x) = \int \sigma(x+1+\epsilon) - \sigma(x-1+\epsilon) dh(\epsilon).$$

Since the parameter distribution ρ is bounded below by some positive constant over the ball B_3, the function σ_* can be expressed by (3.33) with bounded parameter function a, and thus $\sigma_* \in \mathcal{H}_\sigma(\mathbb{R}, \mathbb{R})$.

Hence, we always have $\sigma_* \in \mathcal{H}_\sigma(\mathbb{R}, \mathbb{R})$ and it is integrable ($\|\sigma_*\|_{L^1(\mathbb{R})} < \infty$). Define the subspace $\mathcal{H}^c_{\sigma_*} \subseteq \mathcal{H}_{\sigma_*}$ of functions f_a whose parameter functions a are compactly-supported. Since $\mathcal{H}^c_{\sigma_*} \subseteq \mathcal{H}_\sigma$, it suffices to show that $\mathcal{H}^c_{\sigma_*}$ is dense in $L^2(\mathbb{P})$. Without loss of generality, we denote σ_* by σ. Since σ is L^1, its Fourier transform $\hat{\sigma}$ is well-defined. Since σ is not constant zero, there exists a constant $c \neq 0$ such that $\hat{\sigma}(c) \neq 0$. Perform scaling if necessary, assume that $\hat{\sigma}(1) = 1$.

It suffices to approximate the subspace $C_c^\infty(\mathbb{R}^d)$, which is dense in $L^2(\mathbb{P})$. Fix any $f \in C_c^\infty$. Its Fourier transform $\hat{f}$ is integrable. Then, Step 1 of the proof of [52, Theorem 3.1] implies that,

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

$$f(\mathbf{x}) = \int_{\mathbb{R}} \int_{\mathbb{R}^d} \sigma(\mathbf{w}\cdot\mathbf{x}+b) m(\mathbf{w},b)\, d\mathbf{w}db, \quad m(\mathbf{w},b) = \mathrm{Re}\big[\hat{f}(\mathbf{w})e^{-2\pi i b}\big].$$

For any $R>0$, define the signed distribution m_R by

$$m_R(\mathbf{w},b) = \mathrm{Re}\big[\hat{f}(\mathbf{w})e^{-2\pi i b}\big]\mathbf{1}_{[-R,R]^{d+1}}(\mathbf{w},b).$$

Define the function f_R by

$$\begin{aligned} f_R(\mathbf{x}) &= \int \sigma(\mathbf{w}\cdot\mathbf{x}+b) dm_R(\mathbf{w},b) \\ &= \int a_R(\mathbf{w},b)\sigma(\mathbf{w}\cdot\mathbf{x}+b) d\rho(\mathbf{w},b). \end{aligned}$$

If Assumption 3.2 holds, then define the parameter function a_R by the compactly supported function

$$a_R(\mathbf{w},b) = \frac{m_R(\mathbf{w},b)}{\rho(\mathbf{w},b)}.$$

Then,

$$\|f_R\|_{\mathcal{H}} \le \|a_R\|_{L^2(\rho)} \le \frac{\|f\|_{L^1(\mathbb{R}^d)}}{\sqrt{\min_{\mathbf{w},b\in[-R,R]^{d+1}} \rho(\mathbf{w},b)}} < \infty.$$

If Assumption 3.1 holds, then define a_R by the following function over the l^1 sphere $\{\|\mathbf{w}\|_1 + |b| = 1\}$:

$$a_R(\mathbf{w},b) = \frac{\int_0^\infty m_R(\lambda\mathbf{w},\lambda b)d\lambda}{\rho(\mathbf{w},b)}.$$

Then,

$$\|f_R\|_{\mathcal{H}} \le \|a_R\|_{L^2(\rho)} \le \frac{\|f\|_{L^1(\mathbb{R}^d)}}{\sqrt{\min_{\|\mathbf{w}\|+|b|=1} \rho(\mathbf{w},b)}} < \infty.$$

Thus, we always have $f_R \in \mathcal{H}_\sigma^c(\mathbb{R}^d,\mathbb{R})$.

The approximation error is bounded by

$$\|f - f_R\|_{L^2(\mathbb{P})}^2 \le \iint \iint \int h_R(\mathbf{x},\mathbf{w},b,\mathbf{w}',b') dP(\mathbf{x}) d\mathbf{w}d\mathbf{w}'dbdb',$$

where

$$\begin{aligned} h_R(\mathbf{x},\mathbf{w},b,\mathbf{w}',b') = {}& |\sigma(\mathbf{w}\cdot\mathbf{x}+b)||\hat{f}(\mathbf{w})|\mathbf{1}_{\mathbb{R}^d-[-R,R]^{d+1}}(\mathbf{w},b) \\ & \times |\sigma(\mathbf{w}'\cdot\mathbf{x}+b')||\hat{f}(\mathbf{w}')|\mathbf{1}_{\mathbb{R}^d-[-R,R]^{d+1}}(\mathbf{w}',b'). \end{aligned}$$

Note that $0 \le h_R \le h_0$, and $h_R \to 0$ pointwise, and h_0 is integrable

$$\iint \iint \int h_0(\mathbf{x},\mathbf{w},b,\mathbf{w}',b') dP(\mathbf{x}) d\mathbf{w}d\mathbf{w}'dbdb' \le \|\sigma\|_{L^1(\mathbb{R})}^2 \|\hat{f}\|_{L^1(\mathbb{R}^d)}^2 < \infty.$$

Hence, the dominated convergence theorem implies that $\lim_{R\to\infty} \|f - f_R\|_{L^2(\mathbb{P})} \to 0$, which completes the proof. □

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

Proof of Proposition 4.2. If Assumption 3.1 holds, then [109] implies that $\mathcal{H}$ is dense in $C(K)$ with respect to the supremum norm. Else, Assumption 3.2 holds, then [52] implies that $\mathcal{H}$ is dense in $C(K)$. Thus, we can apply [126, Proposition 2.1] to conclude the proof. □

Proof of Proposition 4.3. It suffices to approximate the compactly-supported distributions, which are dense with respect to the W_2 metric. Fix any compactly-supported distribution P. Choose $R > 0$ such that the support is contained in B_R. Assume that the base distribution $\mathbb{P}$ has full support over $\mathbb{R}^d$. The case without full support will be discussed in the end.

By Brennier's theorem [117, Theorem 2.12], there exists a convex function ψ over $\mathbb{R}^d$ such that $\nabla\psi\#\mathbb{P} = P$. For any $\delta > 0$, we can define the mollified function ψ_δ

$$\psi_\delta(\mathbf{x}) = h_\delta * \psi = \int h\left(\frac{\mathbf{y}}{\delta}\right)\psi(\mathbf{x}-\mathbf{y})d\mathbf{y} = \int h\left(\frac{\mathbf{x}-\mathbf{y}}{\delta}\right)\psi(\mathbf{y})d\mathbf{y},$$

where h is a mollifier (i.e. h is C^∞, non-negative, supported in the unit ball, and $\int h = 1$). Then, ψ_δ is C^∞ and convex, and

$$\lim_{\delta\to 0} W_2(P, \nabla\psi_\delta\#\mathbb{P}) \le \lim_{\delta\to 0}\|\nabla\psi - h_\delta * \nabla\psi\|_{L^2(\mathbb{P})} = 0.$$

Thus, without loss of generality, we can assume that ψ is C^∞.

Define the time-dependent transport map for $\tau \in [0,1]$

$$T_\tau(\mathbf{x}) = (1-\tau)\mathbf{x} + \tau\nabla\psi(\mathbf{x}) = \nabla\left[(1-\tau)\frac{\|\mathbf{x}\|^2}{2} + \tau\psi\right],$$

which is C^∞ in $\mathbf{x}$ and τ, and define the distributions $P_\tau = T_\tau\#\mathbb{P}$, which are known as McCann interpolation. Then, define the vector field

$$V_\tau = \nabla\psi \circ T_\tau^{-1}.$$

The Jacobian of T_τ is positive definite for $\tau \in [0,1)$

$$\nabla_\mathbf{x} T_\tau = \text{Hess}\left[(1-\tau)\frac{\|\mathbf{x}\|^2}{2} + \tau\psi\right] \ge (1-\tau)I,$$

so the inverse function theorem implies that the inverse T_τ^{-1} exists and is C^∞ over $(\mathbf{x},\tau) \in \mathbb{R}^d \times [0,1)$, with

$$\nabla_\mathbf{x} T_\tau^{-1} \le \frac{1}{1-\tau}I, \quad \partial_\tau T_\tau^{-1}(\mathbf{x}) = -\nabla_\mathbf{x} T^{-1}(\mathbf{x}) \cdot \partial_\tau T_\tau\big(T_\tau^{-1}(\mathbf{x})\big).$$

Since $\lim_{\delta\to 0^+} W_2(P_{1-\delta}, P) = 0$, we can replace the approximation target P by the sequence $\{P_{1-1/n}\}$, restrict τ to the interval $[0, 1-1/n]$, and thus assume without loss of generality that $\sup\nabla_\mathbf{x} T_\tau^{-1} < \infty$. It follows that T_τ^{-1} and V_τ are C^∞ over $\mathbb{R}^d \times [0,1]$. By Picard-Lindelöf theorem, the ODE $\dot{\mathbf{x}}_\tau = V_\tau(\mathbf{x}_\tau)$ has unique solution locally, and it is straightforward to check that $\mathbf{x}_t = T_t(\mathbf{x}_0)$ is exactly the solution.

For any $r > R+1$ and any $\epsilon \in (0,1)$, the universal approximation theorem [51] implies that there exists a random feature function $V_{r,\epsilon} \in \mathcal{H}(\mathbb{R}^{d+1}, \mathbb{R}^d)$ such that

$$\|V - V_{r,\epsilon}\|_{C^0(B_r \times [0,1])} < \epsilon. \tag{9.1}$$

Denote its flow map (3.5) by G_τ. For any $\mathbf{x}_0 \in B_r$, let $\mathbf{x}_\tau, \mathbf{y}_\tau$ denote the solutions to the ODEs

$$\dot{\mathbf{x}}_\tau = V(\mathbf{x}_\tau, \tau), \quad \dot{\mathbf{y}}_\tau = V_{r,\epsilon}(\mathbf{y}_\tau, \tau), \quad \mathbf{x}_0 = \mathbf{y}_0.$$

Recall that $V_\tau(\mathbf{x}_\tau)$ is simply the vector $\mathbf{x}_1 - \mathbf{x}_0$, where $\mathbf{x}_1 \in \text{sprt}P \subseteq B_R$. Then, for any $\mathbf{x} \in B_r - B_{R+1}$ and $\tau \in [0,1]$,

$$\frac{\mathbf{x}}{\|\mathbf{x}\|} \cdot V(\mathbf{x}, \tau) \leq -1, \quad \frac{\mathbf{x}}{\|\mathbf{x}\|} \cdot V_{r,\epsilon}(\mathbf{x}, \tau) \leq -1 + \epsilon$$

and this remains true despite the surgeries we performed on V (the mollification of ψ and the restriction of t to $[0, 1-1/n]$). It follows that the solutions $\mathbf{x}_\tau, \mathbf{y}_\tau$ are contained in B_r. By [113, Theorem 2.8],

$$\|\mathbf{x}_1 - \mathbf{y}_1\| \leq \|V - V_{r,\epsilon}\|_{C^0(B_r \times [0,1])} \frac{e^{\|\nabla V\|_{C^0(B_r \times [0,1])}} - 1}{\|\nabla V\|_{C^0(B_r \times [0,1])}} < \epsilon \exp \|\nabla V\|_{C^0(B_r \times [0,1])}.$$

Thus

$$W_2\big(T_1\#\mathbb{P}|_{B_r}, G_1\#\mathbb{P}|_{B_r}\big) < \epsilon \exp \|\nabla V\|_{C^0(B_r \times [0,1])}.$$

Meanwhile, for any $m > 0$, define the random feature function $g_m \in \mathcal{H}$

$$g_m(\mathbf{x}, \tau) = m \int \mathbf{n}\, \sigma\big(m(\mathbf{n} \cdot \mathbf{x} - (r+1))\big) d(h_{1/m} * U\mathbb{S}_{d-1})(\mathbf{n}),$$

where $U\mathbb{S}_{d-1}$ is the uniform distribution over the unit sphere and $h_{1/m}$ is a mollifier. Since σ is sigmoid, as $m \to \infty$, we have $\|g_m(\mathbf{x})\| \to 0$ uniformly over B_r while $\|g_m(\mathbf{x})\| \to \infty$ uniformly over $\mathbb{R}^d - B_{r+2}$. Replace $V_{r,\epsilon}$ by $V_{r,\epsilon} - g_m$ with m sufficiently large enough such that (9.1) continues to hold, while for all $\mathbf{x} \in \mathbb{R}^d - B_{r+2}$ and $t \in [0,1]$

$$\frac{\mathbf{x}}{\|\mathbf{x}\|} V_{r,\epsilon}(\mathbf{x}, t) < 0.$$

It follows that $\|G_1(\mathbf{x}_0)\| \leq \|\mathbf{x}_0\|$ for all $\mathbf{x}_0 \in \mathbb{R}^d - B_{r+2}$. Thus

$$W_2\big(T_1\#\mathbb{P}|_{\mathbb{R}^d - B_r}, G_1\#\mathbb{P}|_{\mathbb{R}^d - B_r}\big) < \int_{\mathbb{R}^d - B_{r+2}} (\|\mathbf{x}\| + r + 2)^2 d\mathbb{P}(\mathbf{x}).$$

Hence,

$$\begin{aligned} &\lim_{r \to \infty} \lim_{\epsilon \to 0} W_2(P, G_1\#\mathbb{P}) \\ \leq &\lim_{r \to \infty} \lim_{\epsilon \to 0} W_2\big(T_1\#\mathbb{P}|_{B_r}, G_1\#\mathbb{P}|_{B_r}\big) + W_2\big(T_1\#\mathbb{P}|_{\mathbb{R}^d - B_r}, G_1\#\mathbb{P}|_{\mathbb{R}^d - B_r}\big) \end{aligned}$$

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

$$\begin{aligned}
&\leq \lim_{r\to\infty}\lim_{\epsilon\to 0}\epsilon \exp\|\nabla V\|_{C^0(B_r\times[0,1])} + \int_{\mathbb{R}^d - B_{r+2}} (\|\mathbf{x}\| + r + 2)^2 d\mathbb{P}(\mathbf{x})\\
&\leq \lim_{r\to\infty}\int_{\mathbb{R}^d - B_{r+2}} (\|\mathbf{x}\| + r + 2)^2 d\mathbb{P}(\mathbf{x}) \leq 0.
\end{aligned}$$

We conclude that P is a limit point of $\mathcal{G}\#\mathbb{P}$.

Finally, consider the general case when the support of the base distribution $\mathbb{P}$ is not necessarily $\mathbb{R}^d$. For any $\epsilon \in (0,1)$, define $\mathbb{P}_\epsilon = (1-\epsilon)\mathbb{P} + \epsilon\mathcal{N}$, where $\mathcal{N}$ is the unit Gaussian distribution. Choose a velocity field $V_\epsilon \in \mathcal{H}(\mathbb{R}^{d+1}, \mathbb{R}^d)$ such that its flow G_τ satisfies $W_2(P, G_1\#\mathbb{P}_\epsilon) < \epsilon$. Then

$$\begin{aligned}
W_2(P, G_1\#\mathbb{P}) &\leq W_2(P, G_1\#\mathbb{P}_\epsilon) + W_2(G_1\#\mathbb{P}_\epsilon, G_1\#\mathbb{P})\\
&< \epsilon + W_2(G_1\#\epsilon\mathcal{N}, G_1\#\epsilon\mathbb{P})\\
&\leq \epsilon + W_2(\epsilon G_1\#\mathcal{N}, \epsilon\delta_{\mathbf{0}}) + W_2(\epsilon\delta_{\mathbf{0}}, \epsilon G_1\#\mathbb{P})\\
&\leq \epsilon + \epsilon\left(\sqrt{\int \|\mathbf{x}\|^2 d\mathcal{N}(\mathbf{x})} + \sqrt{\int \|\mathbf{x}\|^2 d\mathbb{P}(\mathbf{x})}\right).
\end{aligned}$$

Taking $\epsilon \to 0$ completes the proof. □

9.3 Generalization error

Proof of Proposition 6.2. This proof is a slight modification of the proof of Proposition 6.1 ([126, Proposition 3.9]). By [126, Eq. (17)], with probability $1-\delta$ over the sampling of $P_*^{(n)}$,

$$|L(a) - L^{(n)}(a)| \leq \frac{4\sqrt{2\log 2d} + \sqrt{2\log(2/\delta)}}{\sqrt{n}}\|a\|_{L^2(\rho)}. \tag{9.2}$$

Since the regularized loss is strongly convex in a, the minimizer $a_\lambda^{(n)}$ exists and is unique. Then

$$\begin{aligned}
L(a_\lambda^{(n)}) &\leq L^{(n)}(a_\lambda^{(n)}) + \frac{4\sqrt{2\log 2d} + \sqrt{2\log(2/\delta)}}{\sqrt{n}}\|a_\lambda^{(n)}\|_{L^2(\rho)}\\
&\leq L^{(n)}(a_\lambda^{(n)}) + \frac{\lambda}{\sqrt{n}}\|a_\lambda^{(n)}\|_{L^2(\rho)}\\
&\leq L^{(n)}(a_*) + \frac{\lambda}{\sqrt{n}}\|a_*\|_{L^2(\rho)}\\
&\leq L(a_*) + \left(\lambda + \frac{4\sqrt{2\log 2d} + \sqrt{2\log(2/\delta)}}{\sqrt{n}}\right)\|a_*\|_{L^2(\rho)},
\end{aligned}$$

where the first and last inequalities follow from (9.2) and the third inequality follows from the fact that $a_\lambda^{(n)}$ is the global minimizer. Hence,

$$\mathrm{KL}(P_*\|P_\lambda^{(n)}) = L(a_\lambda^{(n)}) - L(a_*) \leq \frac{2\lambda\|a_*\|_{L^2(\rho)}}{\sqrt{n}}.$$

The proof is complete. □

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

9.4 Training and loss landscape

Proof of Proposition 7.3. Since G_θ is Lipschitz for any $\theta \in \Theta$, the set of generated distributions $\mathcal{P}_\Theta \subseteq \mathcal{P}_2(\mathbb{R}^d)$, so the loss $L(\theta) = L(G_\theta \# \mathbb{P})$ is well-defined. Denote the evaluation of any linear operator l over the Hilbert space Θ by $\langle l, \theta \rangle$. By assumption, for any $\theta_0, \theta_1 \in \Theta$, the velocity field

$$\langle \nabla_\theta G_{\theta_0}, \theta_1 \rangle \in L^2(\mathbb{P}, \mathbb{R}^k).$$

Thus, we can define a linear operator $V_\theta : \Theta \to L^2(G_\theta \# \mathbb{P}, \mathbb{R}^d)$ by

$$\int f(\mathbf{x}) \cdot \langle V_{\theta_0}, \theta_1 \rangle(\mathbf{x}) \, d(G_\theta \# \mathbb{P})(\mathbf{x}) := \int f\big(G(\mathbf{x})\big) \cdot \langle \nabla_\theta G_{\theta_0}, \theta_1 \rangle(\mathbf{x}) \, d\mathbb{P}(\mathbf{x}),$$

where $f \in L^2(G_{\theta_0} \# \mathbb{P}, \mathbb{R}^d)$ is any test function. This operator is continuous since $\|V_\theta\|_{op} \leq \|\nabla_\theta G_\theta\|_{op} < \infty$.

For any perturbation $h \in \Theta$,

$$\begin{aligned}
\frac{d}{d\epsilon} L(\theta + \epsilon h)\big|_{\epsilon=0} &= \lim_{\epsilon \to 0} \frac{1}{\epsilon} \big(L(G_{\theta+\epsilon h} \# \mathbb{P}) - L(G_\theta \# \mathbb{P}) \big) \\
&= \lim_{\epsilon \to 0} \frac{1}{\epsilon} \int \frac{\delta L}{\delta P}\Big|_{G_{\theta+\epsilon h} \# \mathbb{P}} d(G_{\theta+\epsilon h} \# \mathbb{P} - G_\theta \# \mathbb{P}) \\
&= \lim_{\epsilon \to 0} \frac{1}{\epsilon} \int \frac{\delta L}{\delta P}\big(G_{\theta+\epsilon h}(\mathbf{x})\big) - \frac{\delta L}{\delta P}\big(G_\theta(\mathbf{x})\big) d\mathbb{P}(\mathbf{x}) \\
&= \lim_{\epsilon \to 0} \frac{1}{\epsilon} \int \nabla \frac{\delta L}{\delta P}\big(G_\theta(\mathbf{x})\big) \cdot \big(G_{\theta+\epsilon h}(\mathbf{x}) - G_\theta(\mathbf{x})\big) d\mathbb{P}(\mathbf{x}) \\
&= \int \nabla \frac{\delta L}{\delta P}\big(G_\theta(\mathbf{x})\big) \cdot \langle \nabla_\theta G_\theta, h \rangle(\mathbf{x}) \, d\mathbb{P}(\mathbf{x}) \\
&= \int \nabla \frac{\delta L}{\delta P}(\mathbf{x}) \cdot \langle V_\theta, h \rangle(\mathbf{x}) \, d(G_\theta \# \mathbb{P})(\mathbf{x}).
\end{aligned}$$

Hence, the loss L is differentiable in θ and the derivative is the following operator:

$$\langle \nabla_\theta L(\theta), \cdot \rangle = \int \nabla \frac{\delta L}{\delta P}(\mathbf{x}) \cdot \langle V_\theta, \cdot \rangle(\mathbf{x}) \, d(G_\theta \# \mathbb{P})(\mathbf{x}).$$

For any $\theta \in \Theta$ such that $G_\theta \# \mathbb{P} \in CP$,

$$\langle \nabla_\theta L(\theta), \cdot \rangle = \int \mathbf{0} \cdot \langle V_\theta, \cdot \rangle(\mathbf{x}) \, d(G_\theta \# \mathbb{P})(\mathbf{x}) = 0.$$

Thus, $CP \cap \mathcal{P}_\Theta \subseteq CP_\Theta$.

For the second claim, we first verify that the random feature functions G_θ satisfy the smoothness assumptions: For any $\mathbf{a} \in L^2(\rho, \mathbb{R}^d)$ and $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^k$,

$$\begin{aligned}
\|G_{\mathbf{a}}(\mathbf{x}) - G_{\mathbf{a}}(\mathbf{x}')\| &\leq \int \|\mathbf{a}(\mathbf{w}, b)\| |\sigma(\mathbf{w} \cdot \mathbf{x} + b) - \sigma(\mathbf{w} \cdot \mathbf{x}' + b)| d\rho(\mathbf{w}, b) \\
&\leq \|\mathbf{a}\|_{L^2(\rho)} \|\sigma\|_{\text{Lip}} \sqrt{\int \|\mathbf{w}\|^2 d\rho(\mathbf{w}, b)} \|\mathbf{x} - \mathbf{x}'\|
\end{aligned}$$

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

$$\leq C\|\mathbf{a}\|_{L^2(\rho)}\|\mathbf{x}-\mathbf{x}'\|,$$

where $C = \sqrt{d}$ if Assumption 3.1 holds and $C = 1$ if Assumption 3.2 holds. Thus, $G_{\mathbf{a}}$ is Lipschitz. Meanwhile, since the parametrization $\mathbf{a} \mapsto G_{\mathbf{a}}$ is linear

$$\langle \nabla_{\mathbf{a}} G_{\mathbf{a}}(\mathbf{x}), \mathbf{a}' \rangle = G_{\mathbf{a}'}.$$

Then, for any $\mathbf{a}, \mathbf{a}' \in L^2(\rho, \mathbb{R}^d)$ and any $\mathbf{x} \in \mathbb{R}^k$

$$\begin{aligned}
\|\langle \nabla_{\mathbf{a}} G_{\mathbf{a}}(\mathbf{x}), \mathbf{a}' \rangle\| = \|G_{\mathbf{a}'}(\mathbf{x})\| &\leq \|\mathbf{a}'\|_{L^2(\rho)} \left(|\sigma(0)| + \sqrt{\int \|\mathbf{w}\|^2 + |b|^2 d\rho(\mathbf{w}, b)} \sqrt{\|\mathbf{x}\|^2+1} \right) \\
&\leq C\|\mathbf{a}'\|_{L^2(\rho)} \sqrt{\|\mathbf{x}\|^2+1}, \\
\|\langle \nabla_{\mathbf{a}} G_{\mathbf{a}}(\cdot), \mathbf{a}' \rangle\|_{L^2(\mathbb{P})} = \|G_{\mathbf{a}'}\|_{L^2(\mathbb{P})} &\leq C \left(1 + \sqrt{\int \|\mathbf{x}\|^2 d\mathbb{P}(\mathbf{x})} \right) \|\mathbf{a}'\|_{L^2(\rho)},
\end{aligned}$$

where $C = \sqrt{d+1}$ if Assumption 3.1 holds and $C = 1$ if Assumption 3.2 holds. Thus, $G_{\mathbf{a}}(\mathbf{x})$ is C^1 in $\mathbf{a}$ for any $\mathbf{x}$, and $\nabla_{\mathbf{a}} G_{\mathbf{a}}$ is a continuous operator $L^2(\rho, \mathbb{R}^d) \to L^2(\mathbb{P}, \mathbb{R}^d)$.

Consider any $\mathbf{a} \in L^2(\rho, \mathbb{R}^d)$ such that $G_{\mathbf{a}}\#\mathbb{P} \in CP_{\Theta}$. Then, for any $\mathbf{a}' \in L^2(\rho, \mathbb{R}^d)$,

$$\begin{aligned}
0 &= \int \nabla \frac{\delta L}{\delta P}\big(G_{\mathbf{a}}(\mathbf{x})\big) \cdot \big\langle \nabla_{\mathbf{a}} G_{\mathbf{a}}, \mathbf{a}' \big\rangle(\mathbf{x}) \, d\mathbb{P}(\mathbf{x}) \\
&= \int \mathbf{a}'(\mathbf{w}, b) \cdot \int \nabla \frac{\delta L}{\delta P}\big(G_{\mathbf{a}}(\mathbf{x})\big) \sigma(\mathbf{w} \cdot \mathbf{x} + b) \, d\mathbb{P}(\mathbf{x}) d\rho(\mathbf{w}, b).
\end{aligned}$$

If σ is Lipschitz, then the integrand is continuous in $(\mathbf{w}, b)$, so for any $(\mathbf{w}, b) \in \text{sprt}\rho$, we have

$$\int \nabla \frac{\delta L}{\delta P}\big(G_{\mathbf{a}}(\mathbf{x})\big) \sigma(\mathbf{w} \cdot \mathbf{x} + b) \, d\mathbb{P}(\mathbf{x}) = \mathbf{0}.$$

Thus, if either Assumption 3.1 or 3.2 holds, the equality holds for all $(\mathbf{w}, b) \in \mathbb{R}^{k+1}$. It follows that, by universal approximation theorem [52], $\nabla \delta_P L(G_{\mathbf{a}}(\mathbf{x})) = \mathbf{0}$ for $\mathbb{P}$ almost all $\mathbf{x}$. Or equivalently, $\delta_P L(\mathbf{x}) = \mathbf{0}$ for $G_{\mathbf{a}}\#\mathbb{P}$ almost all $\mathbf{x}$. Hence, the reverse inclusion $CP_{\Theta} \subseteq CP \cap \mathcal{P}_{\Theta}$ holds. □

Proof of Proposition 7.4. For any $G \in L^2(\mathbb{P}, \mathbb{R}^d)$, let π be the optimal transport plan between $G\#\mathbb{P}$ and P_*. Define the function

$$m(x) = \int x' \, d\pi(x'|x). \tag{9.3}$$

Then, G is a stationary point if and only if $G = m \circ G$. If $G\#\mathbb{P}$ is absolutely continuous, then π is concentrated on the graph of $\nabla \phi$ for some convex function ϕ by Brennier's theorem [117]. Then, $m = \nabla \phi$, and $G\#\mathbb{P} = (m \circ G)\#\mathbb{P} = \nabla \phi\#(G\#\mathbb{P}) = P_*$, so G is a global minimum.

It follows that if G is a stationary point but not a global minimum, then $P = G\#\mathbb{P}$ must be singular. Since the cumulant function of P is non-decreasing, there are at most

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

countably many jumps. Thus, P can be expressed as $P = P_{ac} + P_{sg}$ where P_{ac} is a density function (the Lebesgue derivative of the continuous part of the cumulant) and P_{sg} is a countable sum of point masses at the jumps x_i. Since P_{sg} is nonzero, there exists $x_* \in \text{sprt}P$ such that $P(\{x_*\}) > 0$. Let S_+, S_- be a disjoint partition of $G^{-1}(\{x_*\})$ such that $\mathbb{P}(S_+) = \mathbb{P}(S_-) = P(x_*)/2$. Since P_* is absolutely continuous, Brennier's theorem implies that the transport plan $\pi(x_0, x_1)$ has the conditional distribution $\pi(\cdot|x_1) = \delta_{\nabla\psi(x_1)}$ for some convex potential ψ. Since $\nabla\psi$ is non-decreasing, there exists an interval $[x_-, x_+]$ such that $\nabla\psi$ maps $[x_-, x_+]$ to $\{x_*\}$ and $P_*([x_-, x_+]) = P(\{x_*\})$. Choose $x_o \in (x_-, x_+)$ such that $P_*([x_-, x_o]) = P_*((x_o, x_+]) = P(\{x_*\})/2$. Define the means

$$m_- = \frac{2}{P(\{x_*\})} \int_{x_-}^{x_o} x\, dP_*(x), \quad m_+ = \frac{2}{P(\{x_*\})} \int_{x_o}^{x_+} x\, dP_*(x).$$

Then, $x_- < m_- < x_o < m_+ < x_+$ and $x_* = (m_- + m_+)/2$. For any ϵ, define the function

$$h = \epsilon\big((x_+ - m_+)\mathbf{1}_{S_+} + (x_- - m_-)\mathbf{1}_{S_-}\big)$$

and the map

$$\forall x \in \text{sprt}P_*, \quad F(x) = \begin{cases} (1-\epsilon)x_* + \epsilon m_-, & \text{if} \quad x \in [x_-, x_o], \\ (1-\epsilon)x_* + \epsilon m_+, & \text{if} \quad x \in (x_o, x_+], \\ \nabla\psi(x), & \text{else.} \end{cases}$$

Then, for any $\epsilon \in (0,1)$,

$$\begin{aligned} & L(G) - L(G+h) \\ & \geq \frac{1}{2}\int |x - \nabla\psi(x)|^2 - |x - F(x)|^2 dP_*(x) \\ & = \frac{1}{2}\int_{x_-}^{x_o} |x - x_*|^2 - |x - (1-\epsilon)x_* - \epsilon m_-|^2 dP_*(x) \\ & \quad + \frac{1}{2}\int_{x_o}^{x_+} |x - x_*|^2 - |x - (1-\epsilon)x_* - \epsilon m_+|^2 dP_*(x) \\ & = \frac{\epsilon^2}{2}\int_{x_-}^{x_o} |x - x_*|^2 - |x - m_-|^2 dP_*(x) + \frac{\epsilon^2}{2}\int_{x_o}^{x_+} |x - x_*|^2 - |x - m_+|^2 dP_*(x) > 0. \end{aligned}$$

Thus, G is a generalized saddle point.

For any initialization $G_0 \in L^2(\mathbb{P})$, let G_t be a trajectory defined by the dynamics (7.2). Let m_t be the function (9.3) defined by the optimal transport plan π_t between $P_t = G_t\#\mathbb{P}$ and P_*. Then, the dynamics (7.2) can be written as

$$\frac{d}{dt}G_t = m_t \circ G_t - G_t.$$

By cyclic monotonicity [117], the coupling π_t must be monotone: for any $(x_0, x_1), (x_0', x_1') \in \text{sprt}\pi_t$,

$$(x_0 - x_0')(x_1 - x_1') \geq 0.$$

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

Thus, the trajectory G_t is order-preserving

$$G_0(z) < G(z') \to G_t(z) < G_t(z').$$

It follows that the cumulant function

$$F(z) := P_t\big((-\infty, G_t(z)]\big) = \int \mathbf{1}_{G_t(z') \leq G_t(z)} d\mathbb{P}(z) = \int \mathbf{1}_{G_0(z') \leq G_0(z)} d\mathbb{P}(z)$$

is constant in t.

Since P_* is absolutely continuous, its cumulant function $F_*(x) = P_*((-\infty, x])$ is a continuous, non-decreasing function from $\mathbb{R}$ to $[0,1]$. For any $p \in [0,1]$, define its inverse by

$$F_*^{-1}(p) = \operatorname{argmin}_x \{F_*(x) \leq p\}.$$

It follows that for any z, the support of $\pi_t(\cdot|G_t(z))$ must lie in the closed interval

$$\begin{aligned} I(z) &= \Big[F_*^{-1}\big(P_t((-\infty, G_t(z)))\big), F_*^{-1}\big(P_t((-\infty, G_t(z)])\big)\Big] \\ &= \Big[F_*^{-1}\big(\lim_{\epsilon \to 0^+} F(z-\epsilon)\big), F_*^{-1}(F(z))\Big]. \end{aligned}$$

It follows that for any $z \in \operatorname{sprt}\mathbb{P}$, the conditional distribution $\pi(\cdot|G(z))$ is exactly P_* conditioned on the subset $I(z)$. Hence, the function

$$m_t \circ G_t(z) = \begin{cases} \dfrac{\int_{I(z)} x dP_*(x)}{P_*(I(z))}, & \text{if} \quad |I(z)| > 0, \\ F_*^{-1}\big(F(z)\big), & \text{else} \end{cases}$$

is constant in t. It follows that the trajectory G_t satisfies

$$\frac{d}{dt} G_t = m_0 \circ G_0 - G_t$$

and thus

$$G_t = e^{-t} G_0 + (1 - e^{-t}) m_0 \circ G_0.$$

It follows that P_t converges to $m_0 \# P_0$. If $P_0 \in \mathcal{P}_{ac}$, then as discussed above, m_0 is exactly the optimal transport map from P_0 to P_*, and thus we have global convergence. Else, there exists some x_0 such that $P_o(\{x_0\}) > 0$, so $(m_0 \# P_0)(\{m_0(x_0)\}) \geq P_0(\{x_0\}) > 0$, and thus $m_0 \# P_0$ is not absolutely continuous. Since $P_* \in \mathcal{P}_{ac}$, we have $m_0 \# P_0 \neq P_*$. □

Acknowledgments

I thank Weinan E for helpful discussions.

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

References

[1] D. H. Ackley, G. E. Hinton, and T. J. Sejnowski, A learning algorithm for Boltzmann machines, *Cogn. Sci.*, **9**(1):147–169, 1985.

[2] M. S. Albergo and E. Vanden-Eijnden, Building normalizing flows with stochastic interpolants, *arXiv: 2209.15571*, 2022.

[3] L. Ambrosio, N. Gigli, and G. Savaré, *Gradient Flows: In Metric Spaces and in the Space of Probability Measures*, Springer Science & Business Media, 2008.

[4] D. An, Y. Guo, N. Lei, Z. Luo, S.-T. Yau, and X. Gu, AE-OT: A new generative model based on extended semi-discrete optimal transport, *ICLR 2020*, 2019.

[5] D. An, Y. Guo, M. Zhang, X. Qi, N. Lei, and X. Gu, AE-OT-GAN: Training gans from data specific latent distribution, In: *European Conference on Computer Vision*, pp. 548–564, Springer, 2020.

[6] B. D. Anderson, Reverse-time diffusion equation models, *Stoch. Process Their Appl.*, **12**(3):313–326, 1982.

[7] M. Arjovsky, S. Chintala, and L. Bottou, Wasserstein GAN, *arXiv:1701.07875*, 2017.

[8] S. Arora, R. Ge, Y. Liang, T. Ma, and Y. Zhang, Generalization and equilibrium in generative adversarial nets (GANs), *arXiv:1703.00573*, 2017.

[9] S. Arora, A. Risteski, and Y. Zhang, Do GANs learn the distribution? Some theory and empirics, In: *International Conference on Learning Representations*, 2018.

[10] Y. Balaji, M. Sajedi, N. M. Kalibhat, M. Ding, D. Stöger, M. Soltanolkotabi, and S. Feizi, Understanding overparameterization in generative adversarial networks, *arXiv:2104.05605*, 2021.

[11] A. R. Barron and C.-H. Sheu, Approximation of density functions by sequences of exponential families, *Ann. Stat.*, pp. 1347–1369, 1991.

[12] Y. Bengio, T. Deleu, E. J. Hu, S. Lahlou, M. Tiwari, and E. Bengio, GFlowNet foundations, *arXiv:2111. 09266*, 2021.

[13] L. Bonati, Y.-Y. Zhang, and M. Parrinello, Neural networks-based variationally enhanced sampling, *Proc. Natl. Acad. Sci.*, **116**:17641–17647, 2019.

[14] Y. Brenier, Polar factorization and monotone rearrangement of vector-valued functions, *Commun. Pure Appl. Math.*, **44**(4):375–417, 1991.

[15] A. Brock, J. Donahue, and K. Simonyan, Large scale GAN training for high fidelity natural image synthesis, *arXiv:1809.11096*, 2018.

[16] T. B. Brown et al., Language models are few-shot learners, *arXiv:2005.14165*, 2020.

[17] Y. Cao and E. Vanden-Eijnden, Learning optimal flows for non-equilibrium importance sampling, *arXiv:2206.09908*, 2022.

[18] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, Emerging properties in self-supervised vision transformers, In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9650–9660, 2021.

[19] T. Chavdarova and F. Fleuret, SGAN: An alternative training of generative adversarial networks, In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 9407–9415, 2018.

[20] T. Che, Y. Li, A. Jacob, Y. Bengio, and W. Li, Mode regularized generative adversarial networks, *arXiv: 1612.02136*, 2016.

[21] R. T. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud, Neural ordinary differential equations, *Adv. neural inf. process. syst.*, pp. 6571–6583, 2018.

[22] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, A simple framework for contrastive learning of visual representations, In: *International Conference on Machine Learning*, pp. 1597–1607, PMLR, 2020.

[23] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets, *Adv. neural inf. process. syst.*, 29, 2016.

[24] L. Chizat and F. Bach, On the global convergence of gradient descent for over-parameterized models using optimal transport, *Adv. neural inf. process. syst.*, pp. 3036–3046, 2018.

[25] A. Chowdhery et al., PaLM: Scaling language modeling with pathways, *arXiv:2204.02311*, 2022.

[26] F. Cucker and S. Smale, On the mathematical foundations of learning, *Bull. Amer. Math. Soc.*, **39**(1):1–49, 2002.

[27] E. Denton, S. Chintala, S. Arthur, and R. Fergus, Deep generative image models using a Laplacian pyramid of adversarial networks, *Adv. neural inf. process. syst.*, pp. 1486–1494, 2015.

[28] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, *arXiv:1810.04805*, 2018.

[29] L. Dinh, J. Sohl-Dickstein, and S. Bengio, Density estimation using real NVP, *arXiv:1605.08803*, 2016.

[30] W. E, A proposal on machine learning via dynamical systems, *Commun. Math. Stat.*, **5**(1):1–11, 2017.

[31] W. E, C. Ma, and Q. Wang, A priori estimates of the population risk for residual networks, *arXiv:1903.02154*, 2019.

[32] W. E, C. Ma, S. Wojtowytsch, and L. Wu, Towards a mathematical understanding of neural network-based machine learning: what we know and what we don't, 2020.

[33] W. E, C. Ma, and L. Wu, A priori estimates for two-layer neural networks, *arXiv:1810.06397*, 2018.

[34] W. E, C. Ma, and L. Wu, Barron spaces and the compositional function spaces for neural network models, *arXiv:1906.08039*, 2019.

[35] W. E, C. Ma, and L. Wu, Machine learning from a continuous viewpoint, *arXiv:1912.12777*, 2019.

[36] W. E, C. Ma, and L. Wu, On the generalization properties of minimum-norm solutions for over-parameterized neural network models, *arXiv:1912.06987*, 2019.

[37] W. E and S. Wojtowytsch, Kolmogorov width decay and poor approximators in machine learning: Shallow neural networks, random feature models and neural tangent kernels, *arXiv:2005.10807*, 2020.

[38] S. Feizi, F. Farnia, T. Ginart, and D. Tse, Understanding GANs in the LQG setting: Formulation, generalization and stability, *IEEE Trans. Inf. Theory*, **1**(1):304–311, 2020.

[39] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, Generative adversarial nets, *Adv. neural inf. process. syst.*, pp. 2672–2680, 2014.

[40] W. Grathwohl, R. T. Chen, J. Bettencourt, I. Sutskever, and D. Duvenaud, FFJORD: Free-form continuous dynamics for scalable reversible generative models, *arXiv:1810.01367*, 2018.

[41] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, A kernel two-sample test, *J. Mach. Learn. Res.*, **13**(1):723–773, 2012.

[42] J.-B. Grill et al., Bootstrap your own latent-a new approach to self-supervised learning, *Adv. neural inf. process. syst.*, **33**:21271–21284, 2020.

[43] J. Gui, Z. Sun, Y. Wen, D. Tao, and J. Ye, A review on generative adversarial networks: Algorithms, theory, and applications, *IEEE Trans. Knowl. Data Eng.*, 2021.

[44] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, Improved training of wasserstein GANs, 2017.

[45] I. Gulrajani, C. Raffel, and L. Metz, Towards GAN benchmarks which require generalization, *arXiv:2001.03653*, 2020.

[46] J. Han, R. Hu, and J. Long, A class of dimensionality-free metrics for the convergence of empirical measures, *arXiv:2104.12036*, 2021.

[47] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, S. Hochreiter, GANs trained by a two time-scale update rule converge to a local Nash equilibrium, *Adv. neural inf. process. syst.*, pp. 6626–6637, 2017.

[48] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, beta-VAE: Learning basic visual concepts with a constrained variational framework, 2016.

[49] J. Ho, A. Jain, and P. Abbeel, Denoising diffusion probabilistic models, *Adv. neural inf. process. syst.*, **33**:6840–6851, 2020.

[50] Q. Hoang, T. D. Nguyen, T. Le, and D. Phung, Mgan: Training generative adversarial nets with multiple generators, In: *International Conference on Learning Representations*, 2018.

[51] K. Hornik, Approximation capabilities of multilayer feedforward networks, *Neural Netw.*, **4**(2): 251–257, 1991.

[52] K. Hornik, M. Stinchcombe, and H. White, Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks, *Neural Netw.*, **3**(5):551–560, 1990.

[53] X. Hou, L. Shen, K. Sun, and G. Qiu, Deep feature consistent variational autoencoder, In: *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1133–1141, IEEE, 2017.

[54] Z. Hu, Z. Yang, R. Salakhutdinov, and E. P. Xing, On unifying deep generative models, *arXiv:1706.00550*, 2017.

[55] C.-W. Huang, D. Krueger, A. Lacoste, and A. Courville, Neural autoregressive flows, *arXiv:1804.00779*, 2018.

[56] A. Hyvärinen and P. Dayan, Estimation of non-normalized statistical models by score matching, *J. Mach. Learn. Res.*, **6**(4), 2005.

[57] V. Ivanov, *The theory of approximate methods and their applications to the numerical solution of singular integral equations*, Vol. 2, Springer Science & Business Media, 1976.

[58] Y. Jiang, S. Chang, and Z. Wang, TransGAN: Two pure transformers can make one strong GAN, and that can scale up, *Adv. neural inf. process. syst.*, **34**:14745–14758, 2021.

[59] J. Johnson, A. Alahi, and L. Fei-Fei, Perceptual losses for real-time style transfer and super-resolution, In: *European Conference on Computer Vision*, pp. 694–711, Springer, 2016.

[60] L. V. Kantorovich, Mathematical methods of organizing and planning production, *Management Science*, **6**(4):366–422, 1960.

[61] L. Kantorovich and G. S. Rubinstein, On a space of totally additive functions, *Vestnik Leningrad. Univ*, **13**:52–59, 1958.

[62] T. Karras, S. Laine, and T. Aila, A style-based generator architecture for generative adversarial networks, In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4401–4410, 2019.

[63] D. P. Kingma, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, and M. Welling, Improved variational inference with inverse autoregressive flow, *Adv. neural inf. process. syst.*, pp. 4743–4751, 2016.

[64] D. P. Kingma and M. Welling, Auto-encoding variational bayes, *arXiv:1312.6114*, 2013.

[65] I. Kobyzev, S. J. Prince, and M. A. Brubaker, Normalizing flows: An introduction and review of current methods, *IEEE Trans. Pattern Anal. Mach. Intell.*, **43**(11):3964–3979, 2020.

[66] N. Kodali, J. Abernethy, J. Hays, Z. Kira, On convergence and stability of GANs, *arXiv:1705.07215*, 2017.

[67] A. Kontorovich, Concentration in unbounded metric spaces and algorithmic stability, In: *International Conference on Machine Learning*, pp. 28–36, PMLR, 2014.

[68] H. Larochelle and I. Murray, The neural autoregressive distribution estimator, In: *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pp. 29–37, 2011.

[69] Q. Lei, J. D. Lee, A. G. Dimakis, and C. Daskalakis, SGD learns one-layer networks in WGANs, *Proc. Mach. Learn. Res.*, **119**:5799–5808, 2020.

[70] S.-H. Li and L. Wang, Neural network renormalization group, *Phys. Rev. Lett.*, **121**(26):260601, 2018.

[71] Y. Li and Z. Dou, Making method of moments great again?–how can GANs learn distributions, *arXiv: 2003.04033*, 2020.

[72] Y. Li, K. Swersky, and R. Zemel, Generative moment matching networks, In: *International Conference on Machine Learning*, pp. 1718–1727, 2015.

[73] T. Lin, C. Jin, and M. Jordan, On gradient descent ascent for nonconvex-concave minimax problems, In: *International Conference on Machine Learning*, pp. 6083–6093, PMLR, 2020.

[74] H. Ling, K. Kreis, D. Li, S. W. Kim, A. Torralba, and S. Fidler, EditGAN: High-precision semantic image editing, *Adv. neural inf. process. syst.*, **34**:16331–16345, 2021.

[75] X. Liu, C. Gong, and Q. Liu, Flow straight and fast: Learning to generate and transfer data with rectified flow, *arXiv:2209.03003*, 2022.

[76] C. Ma, L. Wu, and E. Weinan, The slow deterioration of the generalization error of the random feature model, *Mathematical and Scientific Machine Learning*, pp. 373–389, PMLR, 2020.

[77] Q. Mao, H.-Y. Lee, H.-Y. Tseng, S. Ma, and M.-H. Yang, Mode seeking generative adversarial networks for diverse image synthesis, In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1429–1437, 2019.

[78] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. P. Smolley, On the effectiveness of least squares generative adversarial networks, *IEEE Trans. Pattern Anal. Mach. Intell.*, **41**(12):2947–2960, 2018.

[79] L. Mescheder, A. Geiger, and S. Nowozin, Which training methods for GANs do actually converge? In: *International Conference on Machine Learning*, pp. 3481–3490, PMLR, 2018.

[80] M. Mirza and S. Osindero, Conditional generative adversarial nets, *arXiv:1411.1784*, 2014.

[81] V. Nagarajan, J. Z. Kolter, Gradient descent GAN optimization is locally stable, *arXiv:1706.04156*, 2017.

[82] F. Noé, S. Olsson, J. Köhler, and H. Wu, Boltzmann generators: Sampling equilibrium states of many-

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02

body systems with deep learning, *Science*, **365**(6457):eaaw1147, 2019.

[83] S. Nowozin, B. Cseke, and R. Tomioka, f-GAN: Training generative neural samplers using variational divergence minimization, *Adv. neural inf. process. syst.*, pp. 271–279, 2016.

[84] L. Oneto, S. Ridella, and D. Anguita, Tikhonov, Ivanov and Morozov regularization for support vector machine learning, *Mach. Learn.*, **103**(1):103–136, 2016.

[85] A. Oord et al., Parallel WaveNet: Fast high-fidelity speech synthesis, In: *International Conference on Machine Learning*, pp. 3918–3926, 2018.

[86] A. V. D. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, WaveNet: A generative model for raw audio, *arXiv:1609.03499*, 2016.

[87] A. V. D. Oord, N. Kalchbrenner, and K. Kavukcuoglu, Pixel recurrent neural networks, *arXiv:1601.06759*, 2016.

[88] A. Ororbia and D. Kifer, The neural coding framework for learning generative models, *Nat. Commun.*, **13**(1):1–14, 2022.

[89] G. Papamakarios, T. Pavlakou, and I. Murray, Masked autoregressive flow for density estimation, *Adv. neural inf. process. syst.*, pp. 2338–2347, 2017.

[90] S. Pei, R. Y. Da Xu, S. Xiang, and G. Meng, Alleviating mode collapse in GAN via diversity penalty module, *arXiv:2108.02353*, 2021.

[91] A. Radford, L. Metz, and S. Chintala, Unsupervised representation learning with deep convolutional generative adversarial networks, *arXiv:1511.06434*, 2015.

[92] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, Improving language understanding by generative pre-training, 2018.

[93] A. Rahimi and B. Recht, Uniform approximation of functions with random bases, In: *2008 46th Annual Allerton Conference on Communication, Control, and Computing*, pp. 555–561, IEEE, 2008.

[94] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever, Zero-shot text-to-image generation, In: *International Conference on Machine Learning*, pp. 8821–8831, PMLR, 2021.

[95] D. J. Rezende and S. Mohamed, Variational inference with normalizing flows, *arXiv:1505.05770*, 2015.

[96] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, High-resolution image synthesis with latent diffusion models, In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022.

[97] G. M. Rotskoff and E. Vanden-Eijden, Trainability and accuracy of neural networks: An interacting particle system approach, *Stat.*, 1050:30, 2019.

[98] L. Rout, A. Korotin, and E. Burnaev, Generative modeling with optimal transport maps, *arXiv:2110.02999*, 2021.

[99] H. Salehinejad, S. Sankar, J. Barfett, E. Colak, and S. Valaee, Recent advances in recurrent neural networks, *arXiv:1801.01078*, 2017.

[100] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, Improved techniques for training GANs, *Adv. neural inf. process. syst.*, pp. 2234–2242, 2016.

[101] F. Santambrogio, Optimal transport for applied mathematicians, *Birkäuser*, **55**(58-63):94, 2015.

[102] J. Schmidhuber, Deep learning in neural networks: An overview, *Neural Netw.*, **61**:85–117, 2015.

[103] S. Singh, B. Póczos, Minimax distribution estimation in Wasserstein distance, *arXiv:1802.08855*, 2018.

[104] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, Deep unsupervised learning using nonequilibrium thermodynamics, In: *International Conference on Machine Learning*, pp. 2256–2265, PMLR, 2015.

[105] Y. Song, C. Durkan, I. Murray, and S. Ermon, Maximum likelihood training of score-based diffusion models, *Adv. neural inf. process. syst.*, **34**:1415–1428, 2021.

[106] Y. Song and S. Ermon, Generative modeling by estimating gradients of the data distribution, *Adv. neural inf. process. syst.*, 32, 2019.

[107] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, Score-based generative modeling through stochastic differential equations, *arXiv:2011.13456*, 2020.

[108] J. Su, Variational inference: A unified framework of generative models and some revelations, *arXiv:1807.05936*, 2018.

[109] Y. Sun, A. Gilbert, and A. Tewari, On the approximation properties of random ReLU features, *arXiv:*

1810.04374, 2018.

[110] E. G. Tabak and G. Trigila, Explanation of variability and removal of confounding factors from data through optimal transport, *Commun. Pure Appl. Math.*, **71**(1):163–199, 2018.

[111] E. G. Tabak and C. V. Turner, A family of nonparametric density estimation algorithms, *Commun. Pure Appl. Math.*, **66**(2):145–164, 2013.

[112] E. G. Tabak and E. Vanden-Eijnden, Density estimation by dual ascent of the log-likelihood, *Commun. Math. Sci.*, **8**(1):217–233, 2010.

[113] G. Teschl, *Ordinary Differential Equations and Dynamical Systems*, Chapter 3.4, pp. 83–84, AMS, 2012.

[114] A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-Posed Problems*, V. H. Winston & Sons, John Wiley & Sons, 1977.

[115] O. Valsson and M. Parrinello, Variational approach to enhanced sampling and free energy calculations, *Phys. Rev. Lett.*, 113, 2014.

[116] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, Attention is all you need, *Adv. neural inf. process. syst.*, 30, 2017.

[117] C. Villani, *Topics in Optimal Transportation*, Vol. 58, AMS, 2003.

[118] D. Wang et al., Efficient sampling of high-dimensional free energy landscapes using adaptive reinforced dynamics, *Nat. Comput. Sci.*, **2**(1):20–29, 2022.

[119] Z. Wang and D. W. Scott, Nonparametric density estimation for high-dimensional data–Algorithms and applications, *Wiley Interdisciplinary Reviews: Computational Statistics*, **11**(4):e1461, 2019.

[120] J. Weed and F. Bach, Sharp asymptotic and finite-sample rates of convergence of empirical measures in wasserstein distance, *Bernoulli*, **25**(4A):2620–2648, 2019.

[121] S. Wu, A. G. Dimakis, and S. Sanghavi, Learning distributions generated by one-layer relu networks, *Adv. neural inf. process. syst.*, **32**:8107–8117, 2019.

[122] K. Xu, C. Li, J. Zhu, and B. Zhang, Understanding and stabilizing GANs' training dynamics using control theory, In: *International Conference on Machine Learning*, pp. 10566–10575, PMLR, 2020.

[123] Z.-Q. J. Xu, Y. Zhang, and T. Luo, Overview frequency principle/spectral bias in deep learning, *arXiv: 2201.07395*, 2022.

[124] Z.-Q. J. Xu, Y. Zhang, T. Luo, Y. Xiao, and Z. Ma, Frequency principle: Fourier analysis sheds light on deep neural networks, *arXiv:1901.06523*, 2019.

[125] H. Yang, Generalization error of normalizing flows with stochastic interpolants, 2022. To appear.

[126] H. Yang and W. E, Generalization and memorization: The bias potential model, In: *Proceedings of the 2nd Mathematical and Scientific Machine Learning Conference*, J. Bruna, J. Hesthaven, and L. Zdeborova (Eds.), *Proceedings of Machine Learning Research*, **145**:1013–1043, PMLR, 2022.

[127] H. Yang and W. E, Generalization error of GAN from the discriminator's perspective, *Res. Math. Sci.*, **9**(1):1–31, 2022.

[128] L. Yang, Z. Zhang, Y. Song, S. Hong, R. Xu, Y. Zhao, Y. Shao, W. Zhang, B. Cui, and M.-H. Yang, Diffusion models: A comprehensive survey of methods and applications, *arXiv:2209.00796*, 2022.

[129] Y. Yazici, C.-S. Foo, S. Winkler, K.-H. Yap, and V. Chandrasekhar, Empirical analysis of overfitting and mode drop in GAN training, In: *2020 IEEE International Conference on Image Processing (ICIP)*, pp. 1651–1655, IEEE, 2020.

[130] D. Zhang, R. T. Chen, N. Malkin, and Y. Bengio, Unifying generative models with gflownets, *arXiv: 2209.02606*, 2022.

[131] L. Zhang, W. E, and L. Wang, Monge-Ampère flow for generative modeling, *arXiv:1809.10188*, 2018.

[132] O. Zhang, R.-S. Lin, Y. Gou, Optimal transport based generative autoencoders, *arXiv:1910.07636*, 2019.

[133] P. Zhang, Q. Liu, D. Zhou, T. Xu, and X. He, On the discrimination-generalization tradeoff in GANs, *arXiv:1711.02771*, 2017.

[134] J. Zhao, M. Mathieu, Y. LeCun, Energy-based generative adversarial network, *arXiv:1609.03126*, 2016.

DOI https://doi.org/10.4208/jml.221202 | Generated on 2025-01-14 09:16:02