Understanding Estimation and Generalization Error of Generative Adversarial Networks

Kaiyi Ji, Yi Zhou, and Yingbin Liang *Senior Member, IEEE*

***Abstract*—This paper investigates the estimation and generalization errors of the generative adversarial network (GAN) training. On the statistical side, we develop an upper bound as well as a minimax lower bound on the estimation error for training GANs. The upper bound incorporates the roles of both the discriminator and the generator of GANs, and matches the minimax lower bound in terms of the sample size and the norm of the parameter matrices of neural networks under ReLU activation. On the algorithmic side, we develop a generalization error bound for the stochastic gradient method (SGM) in training GANs. Such a bound justifies the generalization ability of the GAN training via SGM after multiple passes over the data and reflects the interplay between the discriminator and the generator. Our results imply that the training of the generator requires more samples than the training of the discriminator. This is consistent with the empirical observation that the training of the discriminator typically converges faster than that of the generator. The experiments validate our theoretical results.**

***Index Terms*—GAN training, estimation error, generalization error, neural networks, stochastic gradient method**

I. Introduction

GENERATIVE adversarial networks (GANs) [1] have been developed as a successful machine learning tool for learning complex high dimensional distributions, and have been applied to many applications in computer vision, medical science, etc. The GAN training is conducted through a min-max optimization problem, where the minimum and the maximum are taken over a class of generators and a class of discriminators, respectively, in order to guarantee a distribution p_y to be close enough to a target distribution. In particular, in the case that the discriminator class is sufficiently large, the GAN training amounts to finding a generator so that the generated distribution is close to the target distribution p_x with respect to the Jensen-Shannon distance [1]. In this paper, we adopt the so-called neural net distance [2], as given by

$$d_{\mathcal{F}_{nn}}(p_x, p_y) = \sup_{\mathbf{w}\in W}\left[\mathop{\mathbb{E}}_{\mathbf{x}\sim p_x} f(\mathbf{w};\mathbf{x}) - \mathop{\mathbb{E}}_{\mathbf{y}\sim p_y} f(\mathbf{w};\mathbf{y})\right], \quad (1)$$

where $f(\mathbf{w};\mathbf{x})$ denotes the output of a discriminator neural network with network parameters $\mathbf{w}$ and input data $\mathbf{x}$. Correspondingly, the GAN training is to solve the following optimization problem

$$\min_{\mathbf{v}\in V} \; d_{\mathcal{F}_{nn}}(p_x, p_{g(\mathbf{v};Z)}), \quad (1.5)$$

where $p_{g(\mathbf{v};Z)}$ is the generated distribution by a generator neural network $g(\mathbf{v};Z)$ with network parameters $\mathbf{v}$ and input random sample Z that is drawn from a given distribution p_z (e.g., Gaussian distribution).

In practical scenarios, the target distribution p_x is unknown a priori and the evaluation of the expectation in the neural net distance in eq. (1) is computationally expensive. Thus, one usually collects a set of training samples $S_x = \{\mathbf{x}_1, ..., \mathbf{x}_n\}$ and $S_z = \{Z_1, ..., Z_m\}$ from the distributions p_x and p_z, respectively, using which to evaluate the empirical distributions $\widehat{p}_x$ and $\widehat{p}_{g(\mathbf{v};Z)}$ instead. Then, the practical GAN training corresponds to solving the following empirical risk minimization

$$\text{(GAN training):} \quad \min_{\mathbf{v}\in V} \; d_{\mathcal{F}_{nn}}(\widehat{p}_x, \widehat{p}_{g(\mathbf{v};Z)}), \quad (2)$$

which takes the min-max form if we substitute the distance by eq. (1).

As the above GAN training is conducted on empirical distributions, it is therefore important to understand the estimation error induced by the obtained solution. In specific, denote $\widehat{\mathbf{v}}^*$ as the solution of the optimization in eq. (2). Then the corresponding estimation error is defined as

$$d_{\mathcal{F}_{nn}}(p_x, p_{g(\widehat{\mathbf{v}}^*;Z)}) - \inf_{\mathbf{v}\in V} d_{\mathcal{F}_{nn}}(p_x, p_{g(\mathbf{v};Z)}). \quad (3)$$

The estimation error of the GAN training has been studied in [3] for $\widehat{\mathbf{v}}^*$ being the minimizer of a different objective function $d_{\mathcal{F}_{nn}}(\widehat{p}_x, p_{g(\mathbf{v};Z)})$ (as compared to that in eq. (2)). [4] studied the same type of estimation error but took the discriminator class in the Sobolev space. As a result, the estimation error bounds in [3], [4] incorporates only the discriminator part, but does not capture the role of the generator and the interplay between them. This motivates us to address the following questions with respect to the estimation error of the GAN training in this paper.

Q1: Taking $\widehat{\mathbf{v}}^*$ as the minimizer of eq. (2), can we further characterize the impact of the generator and the interplay between the discriminator and the generator on the estimation error?

Q2: Is the developed estimation error bound tight enough with respect to the minimax lower bound?

Another important aspect of the GAN training is the generalization error corresponding to specific optimization algorithms used in practice. Since the GAN training optimization is

Kaiyi Ji is with the Department of Electrical and Computer Engineering, The Ohio State University, Columbus, OH 43220 USA (email: ji.367@osu.edu). Yi Zhou is with the Department of Electrical and Computer Engineering, University of Utah, Salt Lake City, UT 84101, USA (email: yi.zhou@utah.edu). Yingbin Liang is with the Department of Electrical and Computer Engineering, The Ohio State University, Columbus, OH 43220 USA (email: liang.889@osu.edu).

The work of K. Ji and Y. Liang was supported partially by the U.S. National Science Foundation under the grants CCF-1801855 and CCF-1900145.

Kaiyi Ji and Yi Zhou equally contribute to this work.

Manuscript received October 01, 2020; revised July 31, 2020.

typically solved by the stochastic gradient method (SGM) in practice, it is desirable to investigate the generalization error induced by the output of SGM. However, such a topic has not been explored in the existing literature for GANs, which we study in this paper. In specific, denoting $(\mathbf{w}_S, \mathbf{v}_S)$ as the discriminator and the generator trained by SGM with the dataset $S = S_x \cup S_z$, we are interested in the following generalization error of the output of SGM

$$d_{f(\mathbf{w}_S)}(p_x, p_{g(\mathbf{v}_S;Z)}) - d_{f(\mathbf{w}_S)}(\widehat{p}_x, \widehat{p}_{g(\mathbf{v}_S;Z)}), \qquad (4)$$

where $d_{f(\mathbf{w}_S)}(p_x, p_y) := \mathbb{E}_{\mathbf{x}\sim p_x} f(\mathbf{w}_S; \mathbf{x}) - \mathbb{E}_{\mathbf{y}\sim p_y} f(\mathbf{w}_S; \mathbf{y})$ denotes the plug-in neural net distance with the trained discriminator $\mathbf{w}_S$. Similar to the estimation error of the GAN training, an important aspect of the above generalization error is the interplay between the discriminator and the generator in the training process of SGM. Thus, in this paper, we address the following question with regard to the generalization error of SGM for GAN training.

Q3: Can we characterize the generalization error of SGM for the GAN training, which captures the interplay between the discriminator and the generator?

A. Our Contributions

We first develop an upper bound on the estimation error for the GAN training. In contrast to the existing results [3], [4], our upper bound captures the impact of the generator and the interplay between the generator and the discriminator on the estimation error. In particular, our proof requires nontrivial efforts into characterizing the bound via Rademacher complexity of a *compositional* function class of the discriminator and the generator in order to capture their interactions.

We then provide a minimax lower bound for the estimation error. In particular for ReLU networks, we show that the obtained lower bound matches the established upper bound in terms of both the sample size and the norm of the parameter matrices of neural networks. This shows that the generator trained by GANs via the neural net distance is nearly minimax-optimal. The technical proof exploits the Fano's inequality, with the major technical development lying in the construction of appropriate multiple hypothesis distributions and a proper neural network, and the development of a lower bound over neural network functions.

We further develop the first known generalization error bound for SGM in the GAN training under the stability framework [5][1]. The established bound shows that the GAN training via SGM can generalize well after multiple passes over the training data, and the corresponding proof reflects the interplay between the discriminator and generator in the training process of SGM. We note that the generalization error of SGM has been studied for the risk minimization problem under the stability framework [5]. As a comparison, the GAN training corresponds to a min-max problem due to the neural net distance, and the corresponding SGM update consists of a minimization step as well as a maximization step. Thus, the analysis of the corresponding generalization error requires substantial new development.

Furthermore, we note that our analysis is also applicable to the case where the number of samples of the generator is *unlimited*, i.e., the objective function of the GAN training in eq. (2) is $\min_{\mathbf{v}\in V} d_{\mathcal{F}_{nn}}(\widehat{p}_x, p_{g(\mathbf{v};Z)})$, where the estimation and generalization errors of the *generator* become zero. Our results still involve some developments for such a case. For example, we develop the first known minimax lower bound, which match upper bounds in both the sample size and the norm of neural net weights, and we also provide the first characterization on the stability of SGM in the GAN training. These two results are both new in the existing literature.

We provide experiments in training the widely-used deep convolutional GANs (DCGANs, [7]) over CIFAR-10 dataset, and study its the generalization performance in the new setting considered in our paper. Our results show that the generalization performance of DCGANs is improved with the increasing of the latent sample size m, and m needs to be larger than n to achieve a good quality and diversity of the generated images, which supports our theoretical results well.

B. Related Work

Estimation error of GANs: The generalization properties of GANs have been studied recently. In specific, [3] studied the estimation error of the GAN training where they considered the obtained generator $\widehat{\mathbf{v}}^*$ to be the minimizer of the objective function $d_{\mathcal{F}_{nn}}(\widehat{p}_x, p_{g(\mathbf{v};Z)})$ (as opposed to that in eq. (2)). The resulting estimation error bound captures only the impact of the discriminator neural network class, whereas our work further captures the impact of the generator neural network class. [8] provided a generator-dependent estimation error bound for MMD-GAN. In comparison, our work studies the GAN training based on the neural net distance (which corresponds to GAN training by neural networks) rather than MMD. Recently, [9] studied the effect of disconnected support on the estimation performance of GANs, and [10] explored the effectiveness of regularization on the generalization performance of the original GANs. The objectives in their studies do not take the same form as our objective in eq. (2).

Recently, a line of research studies have analyzed the performance of GANs from the density estimation perspective. [11] first studied the impact of the discriminator in the approximation error of GANs, and showed that GANs with certain restricted class of discriminators have a moment-matching effect like f-GANs. Second, it established a weak convergence to the target distribution for a class of GAN objective functions. [4] studied the rate of density estimation of GANs as a nonparametric estimation problem. By modeling the discriminator function class as s-Sobolev and the generator function class as t-Sobolev, [4] developed the first-known minimax lower bound $n^{-\frac{s+t}{2t+d}}$ for training Sobolev GANs, where n is the size of observed samples from the target distribution and d is the dimension. The developed theory has also been applied to parametric function classes including neural networks. In specific, [4] approximates neural networks by Sobolev function class and established minimax rate

[1]The stability here is with respect to replacement of samples, and is different from the stability in [6], which is with respect the dynamic system of GAN optimization (see Section III-B for details).

optimality in terms of the sample size n. [12]–[14] extended the results in [4] from several perspectives. Specifically, [13] extended the results in its early version [4] to a wider range of target distributions and a larger class of functional spaces, which include nonparametric spaces (e.g., Sobolev space and a reproducing kernel Hilbert space) and parametric spaces (e.g., leaky ReLU neural networks). In particular, [13] provided a tighter upper bound on density estimation in Sobolev GANs than that in [4], which matches the minimax lower bound established in [4]. In addition, the developed theories on the parametric function classes such including neural networks improved the results in [4] using a tool of pair regularization. [12] provided upper bounds and minimax lower bounds on the density estimation of a class of implicit generative models including GANs as a typical example. The developed results have been applied to different functional spaces including Sobolev space, a reproducing kernel Hilbert space and neural networks. In particular, for a s-Sobolev discriminator function class and a t-Sobolev generator function class, it provided a tight upper bound $n^{-\frac{s+t}{2t+d}}$ that matches the lower bound in [4]. By approximating neural networks using a Sobolev class, [12] provided a tighter upper bound than that in [4]. In addition, it provided some evidence in the statistical advantages of implicit generative models over conventional sampling approaches. [14] studied the statistical error of GANs by modeling the discriminator and generator as Besov function classes, and demonstrated the advantage of GANs over the conventional linear estimator.

The main developments in [12]–[14] and [4] approximate neural networks as classical nonparametric function classes, and leverage techniques based on these function classes. As a comparison, our developed results directly exploit the structures of neural networks and are given in a form of parameters of neural networks. In particular, our developed upper and lower bounds match with each others in terms of not only the sample size n (as done by [4], [12]–[14]) but also the norms of parameter matrices of neural networks.

Generalization error on deep neural networks: In the context of regression and classification, analysis of the generalization error of deep neural networks has been the theme of a number of recent papers (e.g., [15]–[18]).

[15] did not theoretically provide upper bounds on the generalization error, but proposed a simple method for computing Lipschitz norms of a class of neural networks such as fully connected networks, convolutional networks and residual networks. Such Lipschitz norms are then used for regularizing the model training via a constrained optimization procedure, and are shown to play an important role in controlling the generalization error. Our theory supports this empirical finding by showing that the upper and lower bounds almost matches in terms of such Lipschitz norms. [16] proposed a new loss function by adding a regularization of Lipschitz norms of neural networks for reducing the generalization error. It showed that such a Lipschitz regularization leads to a depth-independent generalization bound. As a comparison, our paper focuses on a non-regularization case, and shows that such Lipschitz norms of neural networks are important in controlling the generalization error. We would like to leave the regularized case for the future study, where one interesting part is to see whether such a regularization technique also helps to reduce the generalization error of GANs. [17] developed upper bounds on the generalization error under the Jacobian-norm based constraints on neural networks. The developments in [17] use a notion of Lipschitzness augmentation inspired by margin theory, which enables to exploit more data-dependent quantities that are empirically small. As a comparison, our paper analyzes the generalization error under a matrix-norm constraints in our eq. (8), which are different from those made in [17]. However, we would like to study the generalization error of GANs under such Jacobian-norm based constraints in the future work.

There have been some other works analyzing the generalization of classic regression problems via Rademacher complexity of neural networks. For example, [19]–[22] developed various bounds on the Rademacher complexity of neural networks for a class of distributions with bounded support, and [23] studied the average Rademacher complexity of neural networks over Gaussian variables. Our upper bounds on the estimation error are also developed based on the Rademacher complexity of neural networks, but the key difference from these works is that our analysis needs to handle the Rademacher complexity of a *compositional* function class.

Estimation of neural net distance: Another type of related but different problems focus on the estimation of the neural net distance , which does not include generator minimization in the GAN training. In particular, [2] established an upper bound on the difference between the empirical and true neural net distances, and [24] further established the minimax estimation optimality of the empirical neural net distance. In comparison, our work here studies the the neural net distance between target and generated distributions, whereas [2], [24] studied the difference between the empirical and true neural net distances.

Stability and generalization error: The stability approach was initially proposed by [25] to study the generalization error under the risk minimization framework, and [26] further extended the stability framework to characterize the generalization error of randomized learning algorithms. [27] developed various properties of stability on learning problems. In [5], the authors first applied the stability framework to study the expected generalization error for SGD, followed by a number of studies [28]–[31] that characterized the generalization error of SGD under various scenarios. In [32], the authors studied the generalization error of several first-order algorithms for loss functions satisfying the gradient dominance [33] and the quadratic growth conditions. [34] studied the stability of online learning algorithms. More recently, [35], [36] improves the probablistic generalization bounds for uniformly-stable algorithms, and [37] relaxes the constraint on choice of the learning rate and the assumption on boundedness of the gradient of SGD to achieve a probablistic generalization guarantee. More recently, [38] studied the generalization error of SGD under a stagewise decreasing stepsize and [39] studied the generalization error of stochastic gradient Langevin dynamics in non-convex optimization, both via the algorithm stability approach. All the above work studied the generalization error

for the minimization problem, whereas this paper develops the stability-based method for analyzing the min-max problem specialized for the GAN training.

C. Practicality on Limited Number of Generator Samples

In this section, we provide some applications in domain transportation to motivate the practicality of the assumption on the limited number of generator's input samples. First, consider conditional GANs as an example. Following [40], the goal of conditional GANs is to find a generator $G(\mathbf{v};\cdot)$ parameterized by $\mathbf{v}$ from data $\mathbf{y}$ in a domain $\mathcal{Y}$ and a random noise vector Z, to data in another domain $\mathcal{X}$. Let p_x and p_y denote the distributions of x and y. Then, the objective function of conditional GANs is given by

$$\min_{\mathbf{v}\in V} \; d_{\mathcal{F}_{nn}}(p_x, p_{G(\mathbf{v};\mathbf{y},Z)}), \tag{5}$$

where $d_{\mathcal{F}_{nn}}(\cdot,\cdot)$ is the neural net distance and $p_{G(\mathbf{v};\mathbf{y},Z)}$ is the generated distribution by generator $G(\mathbf{v};\mathbf{y},Z)$ with data $y \sim p_y, z \sim p_z$. In practical scenarios, the distributions p_x and p_y of real images are unknown a priori. Thus, one usually collects a set of training samples $S_x = \{\mathbf{x}_1, ..., \mathbf{x}_n\}$ and $S_y = \{\mathbf{y}_1, ..., \mathbf{y}_m\}$ from the distributions p_x and p_y, using which to compute the empirical distributions $\widehat{p}_x$ and $\widehat{p}_{G(\mathbf{v};\mathbf{y},Z)}$. Then, the practical objective function of conditional GANs in [40], [41] corresponds to solve the empirical risk minimization $\min_{\mathbf{v}\in V} \; d_{\mathcal{F}_{nn}}(\widehat{p}_x, \widehat{p}_{G(\mathbf{v};\mathbf{y},Z)})$. Thus, in the above framework, the generalization error of the generator exists due to the limited number of the training samples $S_y = \{\mathbf{y}_1, ..., \mathbf{y}_m\}$, and hence our assumption and analysis apply to such a setting.

Another example is cycle-GANs [42], of which the goal is to learn mapping functions (which can be regarded as generator functions) between two different domains X and Y, given two sets of training samples $\{\mathbf{x}_i\}$ and $\{\mathbf{y}_i\}$. In such a setting, the number of input samples of generators is naturally limited, and hence falls into the framework we study in our paper.

In our experiments, we also validate this setting for DCGANs, where we show that with a proper choice of the latent sample size m, the generalization performance of DCGANs exhibits a behavior similar to the conventional setting where fresh latent samples are drawn at each iteration.

II. Estimation Error and Minimax Optimality

A. Problem Formulation

The goal of the GAN training is to obtain a generator so that the generated distribution is close to the target distribution. Suppose we obtain the desired generator $\widehat{\mathbf{v}}^*$ via the GAN training, i.e.,

$$\widehat{\mathbf{v}}^* = \arg\min_{\mathbf{v}\in V} \; d_{\mathcal{F}_{nn}}\left(\widehat{p}_x, \widehat{p}_{g(\mathbf{v};Z)}\right). \tag{6}$$

Next, consider the quantity $d_{\mathcal{F}_{nn}}(p_x, p_{g(\widehat{\mathbf{v}}^*;Z)})$ that measures the distance between the target distribution p_x and the generated distribution $p_{g(\widehat{\mathbf{v}}^*;Z)}$. Note that it is further decomposed as

$$\begin{aligned} d_{\mathcal{F}_{nn}}(p_x, p_{g(\widehat{\mathbf{v}}^*;Z)}) &= \underbrace{\inf_{\mathbf{v}\in V} d_{\mathcal{F}_{nn}}(p_x, p_{g(\mathbf{v};Z)})}_{\text{approximation error}} \\ &\quad + \underbrace{d_{\mathcal{F}_{nn}}(p_x, p_{g(\widehat{\mathbf{v}}^*;Z)}) - \inf_{\mathbf{v}\in V} d_{\mathcal{F}_{nn}}(p_x, p_{g(\mathbf{v};Z)})}_{\text{estimation error}}. \end{aligned}$$

The first two terms capture the performance gap between the empirically trained generator and the best-possible generator, and is referred to as the estimation error. The third term corresponds to the minimal distance between the target distribution p_x and the entire class of generator distributions, and is referred to as the approximation error. In the case where the generator function class is infinitely powerful, this term achieves zero approximation error. Our goal here is to study the estimation error that captures the statistical nature of the GAN training process.

We consider the following standard discriminator and generator classes of neural networks. The discriminator function class $\mathcal{F}_w := \{f(\mathbf{w};\cdot) : \mathbf{w} \in W\}$ is taken as a set of neural networks of the form:

$$\begin{aligned} f \in \mathcal{F}_w : \mathbf{x} \in \mathcal{X} := \{\mathbf{x} : \|\mathbf{x}\| \le B_x\} \longmapsto \\ \mathbf{w}_d^{\intercal}\sigma_{d-1}\left(\mathbf{W}_{d-1}\sigma_{d-2}\left(\cdots\sigma_1(\mathbf{W}_1\mathbf{x})\right)\right) \in \mathbb{R}, \end{aligned} \tag{7}$$

where $\mathbf{w}_d$ is the parameter vector of the output layer, $\mathbf{W}_i, i = 1, 2, ..., d-1$ are the parameter matrices of the intermediate layers, and each $\sigma_i(\cdot)$ denotes the entry-wise activation function of layer i for $i = 1, 2, \ldots, d-1$, i.e., for an input $\mathbf{r} \in \mathbb{R}^t$, $\sigma_i(\mathbf{r}) := [\sigma_i(r_1), \sigma_i(r_2), ..., \sigma_i(r_t)]^T$. We assume that each $\sigma_i(\cdot)$ is $L_w(i)$-Lipschitz, i.e., $\|\sigma_i(r_1) - \sigma_i(r_2)\| \le L_w(i)\|r_1 - r_2\|$ for any $r_1, r_2 \in \mathbb{R}$. The generator class $\mathcal{G}_v := \{g(\mathbf{v};\cdot) : \mathbf{v} \in V\}$ is taken as a set of neural networks of the form:

$$\begin{aligned} g \in \mathcal{G}_v : Z \in \mathcal{Z} := \{Z : \|Z\| \le B_z\} \longmapsto \\ \mathbf{V}_s\phi_{s-1}\left(\mathbf{V}_{s-1}\phi_{s-2}\left(\cdots\phi_1(\mathbf{V}_1 Z)\right)\right), \end{aligned} \tag{8}$$

where $\mathbf{V}_i$, $i = 1, ..., s$ are parameter matrices and each activation function $\phi_i(\cdot)$ is entry-wise and assumed to be $L_v(i)$-Lipschitz. Moreover, for the discriminator and generator neural networks, we consider the following compact parameter sets, as adopted in [21], [22], [43].

$$\begin{aligned} W :=& \prod_{i=1}^{d-1} \left\{\mathbf{W}_i \in \mathbb{R}^{p_i \times p_{i+1}} : \|\mathbf{W}_i\|_F \le M_w(i)\right\} \\ &\times \left\{\mathbf{w}_d \in \mathbb{R}^{p_d} : \|\mathbf{w}_d\| \le M_w(d)\right\}, \\ V :=& \prod_{i=1}^{s} \left\{\mathbf{V}_i \in \mathbb{R}^{q_i \times q_{i+1}} : \|\mathbf{V}_i\|_F \le M_v(i)\right\}. \end{aligned} \tag{9}$$

B. Upper Bound on Estimation Error

In this subsection, we develop an upper bound on the estimation error. We first introduce the following notion of Rademacher complexity.

Definition 1 (Rademacher complexity)**.** *Let $\mathcal{F}_W := \{f(\mathbf{w};\mathbf{x}) : \mathbf{w} \in W\}$ be a function class. Then, the Rademacher complexity $\mathcal{R}(\mathcal{F}_W)$ corresponding to the n samples $\mathbf{x}_1, ..., \mathbf{x}_n$ is defined as*

$$\mathcal{R}(\mathcal{F}_W) = \mathbb{E}_{\mathbf{x},\epsilon} \sup_{\mathbf{w}\in W} \left| \frac{1}{n} \sum_{i=1}^{n} \epsilon_i f(\mathbf{w};\mathbf{x}_i) \right|,$$

where $\epsilon_1, \epsilon_2 ..., \epsilon_n$ are independent random variables uniformly chosen from $\{-1, 1\}$. Similarly, for a compositional function class $\mathcal{H}_{W\times V} := \{f(\mathbf{w}; g(\mathbf{v}; Z)) : \mathbf{w} \times \mathbf{v} \in W \times V\}$, we define the Rademacher complexity $\mathcal{R}(\mathcal{H}_{W\times V})$ corresponding to the m samples $Z_1, ..., Z_m$ as

$$\mathcal{R}(\mathcal{H}_{W\times V}) = \mathbb{E}_{Z,\epsilon} \sup_{\mathbf{w}\in W, \mathbf{v}\in V} \left| \frac{1}{m} \sum_{i=1}^{m} \epsilon_i f(\mathbf{w}; g(\mathbf{v}; Z_i)) \right|.$$

Based on the above notion of Rademacher complexity, we establish the following result on the estimation error of GAN.

Theorem 1. *Let $\mathcal{P}_X$ be the class of Borel probability measures over the compact domain $\mathcal{X}$. Let $\mathcal{F}_w$ and $\mathcal{G}_v$ be the discriminator and generator classes given by eq.* (7) *and eq.* (8)*, respectively. Consider a target distribution $p_x \in \mathcal{P}_X$ and the trained generator $\widehat{\mathbf{v}}^*$ given by eq.* (6)*. Then, with probability at least $1 - 2\delta$ over the randomness of the training samples,*

$$\begin{aligned} d_{\mathcal{F}_{nn}}(p_x, p_{g(\widehat{\mathbf{v}}^*;Z)}) - \inf_{\mathbf{v}\in V} d_{\mathcal{F}_{nn}}(p_x, p_{g(\mathbf{v};Z)}) \\ \leq 4\mathcal{R}(\mathcal{F}_W) + 4\mathcal{R}(\mathcal{H}_{W\times V}) \\ + 2U_w\sqrt{2\log\frac{1}{\delta}}\Big(\frac{B_x}{\sqrt{n}} + \frac{B_z U_v}{\sqrt{m}}\Big), \quad (10) \end{aligned}$$

where parameters $U_w = \prod_{i=1}^{d} M_w(i) \prod_{i=1}^{d-1} L_w(i)$ and $U_v = \prod_{j=1}^{s} M_v(j) \prod_{i=1}^{s-1} L_v(i)$.

Proof. See Section A-A. □

Theorem 1 relates the estimation error to the Rademacher complexity of the discriminator $\mathcal{R}(\mathcal{F}_W)$ and the Rademacher complexity of the compositional function class $\mathcal{R}(\mathcal{H}_{W\times V})$. This is due to the fact that the generator is composed into the discriminator in the formulation of the objective function in the GAN training. In other related works [3], [4], they study the estimation error of the GAN training with the objective function $d_{\mathcal{F}_{nn}}\left(\widehat{p}_x, p_{g(\mathbf{v};Z)}\right)$ that is based on the empirical distribution $\widehat{p}_x$ and the true distribution $p_{g(\mathbf{v};Z)}$. As a comparison, our GAN objective function in eq. (2) uses the empirical distribution for both parts. This is more desired as one can only access a finite number of samples from the distributions in practical GAN training. Moreover, their results incorporate only the discriminator function class, and do not capture the impact of the generator function class. Technically, the proof of Theorem 1 requires much more efforts to capture the interplay between the discriminator and generator in characterizing the estimation error.

Based on Theorem 1 and upper bounds on the Rademacher complexity for neural networks, we further obtain the following result.

Corollary 1. *Consider the same setting as that in Theorem 1, and we assume that the activation functions $\sigma_i(\cdot), i = 1, .., d-1$ and $\phi_j(\cdot)$, $j = 1, ..., s-1$ are positive-homogeneous, i.e., $\sigma_i(\alpha r) = \alpha\sigma_i(r)$ and $\phi_j(\alpha r) = \alpha\phi_j(r)$ for any $\alpha \geq 0$ and $r \in \mathbb{R}$. Then, with probability at least $1 - 2\delta$ over the randomness of samples $\{\mathbf{x}_i\}_{i=1}^n$ and $\{Z_i\}_{i=1}^m$, we have*

$$\begin{aligned} d_{\mathcal{F}_{nn}}(p_x, p_{g(\widehat{\mathbf{v}}^*;Z)}) - \inf_{\mathbf{v}\in V} d_{\mathcal{F}_{nn}}(p_x, p_{g(\mathbf{v};Z)}) \\ \leq \frac{4B_x U_w\sqrt{3d}}{\sqrt{n}} + \frac{4U_w U_v B_z\sqrt{3(s+d-1)}}{\sqrt{m}} \\ + 2U_w\sqrt{2\log\frac{1}{\delta}}\left(\frac{B_x}{\sqrt{n}} + \frac{B_z U_v}{\sqrt{m}}\right), \quad (11) \end{aligned}$$

where parameters $U_w = \prod_{i=1}^{d} M_w(i) \prod_{i=1}^{d-1} L_w(i)$ and $U_v = \prod_{j=1}^{s} M_v(j) \prod_{i=1}^{s-1} L_v(i)$.

Proof. See Section A-B. □

The estimation error bound in Corollary 1 has a larger constant coefficient for the sample size m than that for the sample size n. This is because the generator network is composed into the discriminator network in the GAN objective function, which further yields a higher Rademacher complexity. This also justifies in part that the training of the generator is more difficult than that of the discriminator, and suggests to use more samples from p_z than that from the target distribution p_x to balance the estimation error. We note that the homogeneity assumption holds for widely-used ReLU-type activation function, and hence Corollary 1 applies to the typical ReLU networks.

Next, we consider the scenario in which the generator function class $\mathcal{G}_v$ is large enough so that the corresponding distribution class contains the target distribution. Also, we assume that the sample size m at the generator scales faster than the sample size n of the target distribution, as they are drawn from a *known* distribution. Then, based on Theorem 1, we further obtain the following result.

Corollary 2. *Consider the same setting as that in Theorem 1, and assume that activation functions $\sigma_i(\cdot), i = 1, .., d-1$ and $\phi_j(\cdot)$, $j = 1, ..., s-1$ are positive-homogeneous, i.e., $\sigma_i(\alpha r) = \alpha\sigma_i(r)$ and $\phi_j(\alpha r) = \alpha\phi_j(r)$ for any $\alpha \geq 0$ and $r \in \mathbb{R}$. Suppose that the generator function class is large enough such that $p_x \in \mathcal{P}_{\mathcal{G}_v}$, and that the number m of samples drawn from the distribution p_z scales faster than n. Then, with probability at least $1 - 2\delta$, we have*

$$\begin{aligned} &d_{\mathcal{F}_{nn}}(p_x, p_{g(\widehat{\mathbf{v}}^*;Z)}) \leq \\ &\mathcal{O}\left(\frac{\big(4\sqrt{3d} + 2\sqrt{2\log\frac{1}{\delta}}\big) B_x \prod_{i=1}^{d} M_w(i) \prod_{i=1}^{d-1} L_w(i)}{\sqrt{n}}\right). \end{aligned}$$

Proof. By $p_x \in \mathcal{P}_{\mathcal{G}_v}$, we have $\inf_{\mathbf{v}\in V} d_{\mathcal{F}_{nn}}(p_x, p_{g(\mathbf{v};Z)}) = 0$, which, in conjunction with Corollary 1 and the assumption that m scales faster than n, yields the proof. □

Since the samples $\{Z_i\}_{i=1}^m$ are i.i.d. generated from a *known* distribution (e.g., uniform distributions), we can access arbitrarily large number of these samples, e.g., $m = \infty$. For this reason, we are more interested in whether the rate $n^{-1/2}$ obtained by the estimator $p_{g(\widehat{\mathbf{v}}^*;Z)}$ under GAN framework is optimal? To address this issue, we investigate minimax

estimation of the target distribution p_x based on finite samples, as shown in the following subsection.

C. Minimax Lower Bound

We establish a minimax lower bound for the estimation error in the following theorem. The main challenge of the proof of Theorem 2 lies in the construction of appropriate multiple hypothesis distributions and a proper neural network with further development of a lower bound for the neural network functions.

Theorem 2 (Minimax lower bound)**.** *Let $\mathcal{F}_w$ be the discriminator function class given by eq.* (7) *and $\widehat{p}_n$ be any estimator of the target distribution p_x constructed based on the samples $\{\mathbf{x}_i\}_{i=1}^n$. Then, we have*

$$\inf_{\widehat{p}_n} \sup_{p_x \in \mathcal{P}_X} \mathbb{P}\Big\{ d_{\mathcal{F}_{nn}}(\widehat{p}_n, p_x) \geq \frac{C(\mathcal{P}_X)}{\sqrt{n}} \Big\} > 0.42, \tag{12}$$

where the constant $C(\mathcal{P}_X)$ is given by

$$C(\mathcal{P}_X) = 0.015\big(M_w(d)\sigma_{d-1}\left(\cdots(M_w(1)B_x)\right) - M_w(d)\sigma_{d-1}\left(\cdots(-M_w(1)B_x)\right)\big). \tag{13}$$

Proof. See Section B . □

To elaborate, Theorem 2 shows that no matter what algorithm we use to construct the estimator $\widehat{p}_n$ of the target distribution based on the training samples, there exists at least one particular target distribution that makes the corresponding estimation error larger than $\frac{C(\mathcal{P}_X)}{\sqrt{n}}$ with a non-vanishing probability. In other words, this minimax lower bound characterizes the fundamental limit of learning the target distribution via the GAN training based on the neural net distance.

The constant $C(\mathcal{P}_X)$ in the above lower bound takes a complicated form, and its exact value in general depends on the activation function and the neural network structure. To further illustrate, we consider the case in which all the activation functions are ReLU functions and obtain the following result.

Corollary 3. *Consider the same setting as that in Theorem 2 and assume that all activation functions in eq.* (7) *are ReLU, i.e., $\sigma_i(x) = \max(0, x)$ for $i = 1, 2, ..., d-1$. Then, we have*

$$\inf_{\widehat{p}_n} \sup_{p_x \in \mathcal{P}_X} \mathbb{P}\Big\{ d_{\mathcal{F}_{nn}}(\widehat{p}_n, p_x) \geq \frac{0.015 B_x \prod_{i=1}^d M_w(i)}{\sqrt{n}} \Big\} > 0.42.$$

Proof. The proof follows directly from Theorem 2 by letting $\sigma_i(x) = \max(0, x), i = 1, ..., d-1$ in eq. (13). □

Comparing the lower bound in Corollary 3 with the upper bound developed in Corollary 2, it can be seen that the two bounds nearly match each other, and share the same terms $n^{-1/2}$ and $B_x \prod_{i=1}^d M_w(i)$ (note that $L_w(i) = 1$ under ReLU activation). This implies that these quantities play an important role in determining the estimation performance of the GAN training (under ReLU activation). We note that the upper bound in Corollary 2 has an additional depth-dependent term $\sqrt{d}$, which stays at the constant level for reasonably deep neural networks (e.g., $d = 100$). Thus, the generator trained by GANs under ReLU networks is nearly-minimax optimal, and this justifies in part the use of GANs to learn the target distribution.

III. Generalization Error of SGM

In Section II, we study the estimation error assuming that we obtain the optimal solution of the GAN training, i.e., $\widehat{\mathbf{v}}^* = \arg\min_{\mathbf{v} \in V} d_{\mathcal{F}_{nn}}\left(\widehat{p}_x, \widehat{p}_{g(\mathbf{v};Z)}\right)$, where the analysis does not depend on the specific algorithm that is used to obtain the optimal solution. In practice, we typically do not obtain the exact optimal solution. Furthermore, the GAN training is typically performed by the stochastic gradient method (SGM), which significantly affects the generalization performance. Hence, in this section, we focus on SGM in the GAN training, and analyze the non-asymptotical generalization error of SGM.

A. Problem Formulation and SGM

Recall that $S_x = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ and $S_z = \{Z_1, \ldots, Z_m\}$ are the data samples generated from the distributions p_x and p_z, respectively. Here, we rewrite the objective function in the GAN training introduced in eq. (1.5) and eq. (2) more explicitly as follows.

$$\begin{aligned} L(\mathbf{w}, \mathbf{v}) &:= d_{f(\mathbf{w})}(p_x, p_{g(\mathbf{v},Z)}) \\ &= \mathbb{E}_{\mathbf{x}\sim p_x} f(\mathbf{w}; \mathbf{x}) - \mathbb{E}_{Z\sim p_z} f(\mathbf{w}; g(\mathbf{v}; Z)), \\ L_S(\mathbf{w}, \mathbf{v}) &:= d_{f(\mathbf{w})}(\widehat{p}_x, \widehat{p}_{g(\mathbf{v},Z)}) \\ &= \frac{1}{n}\sum_{i=1}^n f(\mathbf{w}; \mathbf{x}_i) - \frac{1}{m}\sum_{j=1}^m f(\mathbf{w}; g(\mathbf{v}; Z_j)). \end{aligned}$$

Here, L_S denotes the empirical risk of the GAN objective, which measures the distance between the empirical target distribution $\widehat{p}_x$ (e.g., already seen real images) and the empirical generated distribution $\widehat{p}_{g(\mathbf{v},Z)}$ (e.g., already generated images), whereas L denotes the population risk of the GAN objective, which measures the distance between the true target distribution p_x (e.g., unseen real images) and the generated distribution $p_{g(\mathbf{v},Z)}$ (e.g., images to be generated). Let $(\mathbf{w}_S, \mathbf{v}_S)$ denote the trained discriminator and generator by SGM with the dataset S. Then, we expect that the obtained generator can further generate images that are close to the unseen real images. Such a goal naturally motivates us to study the generalization error in eq. (4), whose expected value can be rewritten as

$$\text{(Generalization error):} \quad \mathbb{E}_S \mathbb{E}_{\text{sgm}}\big[L(\mathbf{w}_S, \mathbf{v}_S) - L_S(\mathbf{w}_S, \mathbf{v}_S)\big],$$

where the expectation is taken over the random draw of the dataset $S = S_x \cup S_z$ and the random sampling of SGM.

Next, we specify the SGM for the GAN training. Note that the GAN objective in eq. (2) corresponds to solving the min-max optimization problem

$$\min_{\mathbf{v}\in V} \max_{\mathbf{w}\in W} L_S(\mathbf{w}, \mathbf{v}), \tag{14}$$

and the updates of the corresponding SGM can be written as, for $t = 0, \ldots, T-1$,

$$\text{(SGM):} \quad \begin{cases} \mathbf{w}_{t+1} &= \mathbf{w}_t + \eta_t \nabla_{\mathbf{w}} L_S(\mathbf{w}_t, \mathbf{v}_t; \mathbf{x}_{\xi_t}, Z_{\zeta_t}), \\ \mathbf{v}_{t+1} &= \mathbf{v}_t - \eta_t \nabla_{\mathbf{v}} L_S(\mathbf{w}_t, \mathbf{v}_t; Z_{\zeta_t}), \end{cases}$$

where $\xi_t \in \{1, \ldots, n\}$ and $\zeta_t \in \{1, \ldots, m\}$ are sample indices that are sampled uniformly at random, and the stochastic gradients of the discriminator and the generator take the forms

$$\nabla_{\mathbf{w}} L_S(\mathbf{w}_t, \mathbf{v}_t; \mathbf{x}_{\xi_t}, Z_{\zeta_t}) = \nabla_{\mathbf{w}} f(\mathbf{w}_t; \mathbf{x}_{\xi_t}) - \nabla_{\mathbf{w}} f(\mathbf{w}_t; g(\mathbf{v}_t; Z_{\zeta_t})),$$
$$\nabla_{\mathbf{v}} L_S(\mathbf{w}_t, \mathbf{v}_t; Z_{\zeta_t}) = -\nabla_{\mathbf{v}} g(\mathbf{v}_t; Z_{\zeta_t}) \nabla_g f(\mathbf{w}_t; g(\mathbf{v}_t; Z_{\zeta_t})).$$

That is, SGM conducts a stochastic gradient ascent step on the discriminator parameter $\mathbf{w}$ and followed by a stochastic gradient descent step on the generator parameter $\mathbf{v}$. The min-max structure of the optimization problem and the alternating scheme of SGM makes the updates of the discriminator and the updates of the generator interact and compete with each other. Also, it can be seen that the stochastic gradient of the generator has a very different structure from that of the discriminator.

B. Stability Approach

We adopt the stability-based approach for analyzing the generalization error for the GAN training. The stability framework has been established for providing generalization error bounds for risk *minimization* problems (see references in Section I-B). The GAN training is a *min-max* problem, which does not fall directly under the existing framework for stability-based analysis. Hence, in this subsection, we first introduce two notions of *uniform stability* for SGM in the GAN training and then develop a bound on the generalization error for the GAN training based on such stability quantities.

Throughout the paper, we denote $\bar{S} = \bar{S}_x \cup \bar{S}_y$, where $\bar{S}_x$ and $\bar{S}_y$ are data sets that differ from S_x and S_y in one data sample, respectively. We also use $(\mathbf{w}_S, \mathbf{v}_S)$ and $(\mathbf{w}_{\bar{S}}, \mathbf{v}_{\bar{S}})$ to denote the outputs of SGM in the GAN training with the data sets S and $\bar{S}$, respectively.

Definition 2 (Stability and Generalization for GANs)**.** *SGM is said to be ϵ_f uniform-discriminator stable if for any $S, \bar{S}$ that*

$$\sup_{S,\bar{S},\mathbf{x}} \mathbb{E}_{sgm} |f(\mathbf{w}_S; \mathbf{x}) - f(\mathbf{w}_{\bar{S}}; \mathbf{x})| \leq \epsilon_f.$$

Moreover, it is said to be ϵ_g uniform-generator stable if for any $S, \bar{S}$ that

$$\sup_{S,\bar{S},Z} \mathbb{E}_{sgm} |f(\mathbf{w}_S; g(\mathbf{v}_S; Z)) - f(\mathbf{w}_{\bar{S}}; g(\mathbf{v}_{\bar{S}}; Z))| \leq \epsilon_g.$$

Definition 2 introduces the stability notions for SGM in training the discriminator and the generator of GANs. In particular, the discriminator-stability of SGM is similar to the uniform stability of the SGD introduced in [5]. On the other hand, the generator-stability of SGM measures the stability of the composition of both the discriminator and the generator with respect to the data perturbation. We note that the stability notion in Definition 2 for GANs is different from that proposed in [6], which corresponds to the stability of the dynamic system associated with the corresponding optimization.

Based on the above stability notions, we obtain the following characterization of the generalization error for SGM in training GANs.

Proposition 1. *Let SGM be ϵ_f uniform-discriminator stable and ϵ_g uniform-generator stable. Then, the generalization error induced by the output of SGM in the GAN training satisfies*

$$\mathbb{E}_S \mathbb{E}_{sgm} \big[L(\mathbf{w}_S, \mathbf{v}_S) - L_S(\mathbf{w}_S, \mathbf{v}_S) \big] \leq \epsilon_f + \epsilon_g.$$

Proof. See Section C. □

Proposition 1 bounds the generalization error of SGM in the GAN training in terms of its discriminator-stability and generator-stability. The proof requires special treatments for both the discriminator and the generator, and the obtained upper bound involves their corresponding stability notions. This is different from the stability result developed in [5], where the generalization error is bounded by the stability of the single objective function.

The generalization error induced by SGM in training GANs is affected by the algorithm stability of both the discriminator and the generator, which interact with each other due to the competence between the two networks in the training process. Thus, bounding the uniform-generator stability and the uniform-discriminator stability in Proposition 1 is the key to understanding the generalization error of SGM for training GANs, and we further explore them in the next subsection.

C. Stability Bounds for SGM in the GAN Training

To understand the generalization error of SGM for training GANs, we adopt the following assumptions on the discriminator and generator of GANs.

Assumption 1. *The discriminator and generator of GANs satisfy:*

1) *For all $\mathbf{w}, \mathbf{x}$, $f(\cdot; \mathbf{x})$ and $f(\mathbf{w}; \cdot)$ are σ_f^w-Lipschitz and σ_f^x-Lipschitz, respectively;*
2) *For all $\mathbf{w}, \mathbf{x}$, $\nabla_{\mathbf{w}} f(\cdot; \mathbf{x})$ and $\nabla_{\mathbf{x}} f(\mathbf{w}; \cdot)$ are L_f^w-Lipschitz and L_f^x-Lipschitz, respectively;*
3) *For all Z, $g(\cdot; Z)$ is σ_g-Lipschitz and $\nabla_{\mathbf{v}} g(\cdot; Z)$ is L_g-Lipschitz.*

Assumption 1 essentially assumes that the discriminator is Lipschitz in terms of either $\mathbf{w}$ or $\mathbf{x}$ and the generator is Lipschitz in terms of $\mathbf{v}$, which are standard assumptions adopted in stability analysis of SGM [5]. In particular, for fully-connected deep neural network models discussed in the previous section, the Lipschitz constants σ_f^x and σ_g reduce to the quantities U_w and U_v defined in Theorem 1. We note that the bivariate Lipschitz property in the item 1 of Assumption 1 is introduced to the discriminator due to its composition with the generator in the GAN training. Based on Assumption 1, we obtain the following result.

Lemma 1. *Let Assumption 1 hold. Apply SGM to solve the ERM in eq.* (14) *with the data sets S and $\bar{S}$, respectively, and denote the corresponding outputs as $(\mathbf{w}_S, \mathbf{v}_S)$ and $(\mathbf{w}_{\bar{S}}, \mathbf{v}_{\bar{S}})$, respectively. Then, the stabilities ϵ_f, ϵ_g of GAN satisfy*

$$\epsilon_f + \epsilon_g \leq 2\sigma_f^x \sup_{S,\bar{S}} \mathbb{E}_{sgm} \|\mathbf{w}_S - \mathbf{w}_{\bar{S}}\| + \sigma_f^w \sigma_g \sup_{S,\bar{S}} \mathbb{E}_{sgm} \|\mathbf{v}_S - \mathbf{v}_{\bar{S}}\|.$$

Proof. See Section D. □

By Lemma 1, the generalization error of SGM for the GAN training is bounded by the stability of both the discriminator parameter and the generator parameter with respect to the data perturbations. In particular, as the discriminator and generator are involved and competing with each other in the SGM updates, their corresponding stabilities also affect each other.

Next, we characterize how the stability of the discriminator interact with that of the generator in the training process of

GAN with the SGM. For simplicity of presentation, we denote $\delta_t^w := \|\mathbf{w}_{t,S} - \mathbf{w}_{t,\bar{S}}\|$ and $\delta_t^v = \|\mathbf{v}_{t,S} - \mathbf{v}_{t,\bar{S}}\|$, and obtain the following result.

Proposition 2 (Stability of SGM for GANs)**.** *Let Assumption 1 hold. Apply SGM to solve the ERM in eq.* (14) *with the datasets S and $\bar{S}$, respectively, and denote the corresponding outputs as $(\mathbf{w}_S, \mathbf{v}_S)$ and $(\mathbf{w}_{\bar{S}}, \mathbf{v}_{\bar{S}})$, respectively. Denote the stepsize as $\eta_t > 0$. Then, the stabilities of both the discriminator and the generator satisfy*

$$\mathbb{E}_{sgm}\begin{bmatrix}\delta_{t+1}^w \\ \delta_{t+1}^v\end{bmatrix} \leq \begin{bmatrix}1+2\eta_t L_f^w & \eta_t L_f^x \sigma_g \\ \eta_t L_f^w \sigma_g & 1+\eta_t(\sigma_f^x L_g + \sigma_g^2 L_f^x)\end{bmatrix} \mathbb{E}_{sgm}\begin{bmatrix}\delta_t^w \\ \delta_t^v\end{bmatrix} + 2\eta_t \begin{bmatrix}\frac{m+n}{mn}\sigma_f^w \\ \frac{\sigma_g \sigma_f^x}{m}\end{bmatrix}. \tag{15}$$

Proof. See Section E. □

Eq. (15) characterizes how the discriminator stability δ_t^w interacts with the generator stability δ_t^v along the iteration path of SGM. It can be seen that the matrix in eq. (15) has nonzero off-diagonal entries, implying that the stability of the discriminator is correlated with that of the generator in the training process. Also, these two stabilities are affected by very different problem parameters due to the asymmetric roles that the discriminator and the generator play in the GAN objective.

We note that the recursive stability bound in Proposition 2 is the key to obtain the final stability bound. In particular, the analysis of the proposition requires to carefully examine the stochastic samples sampled by the SGM in the updates of the discriminator and generator. Specifically, in the proof, we first analyze the stability of the discriminator δ_{t+1}^w by examining four different cases of stochastic samples: $\xi_t =$ or $\neq 1$ and $\zeta_t =$ or $\neq 1$. We bound δ_{t+1}^w in each separate case and the weighted average of these bounds leads to the desired bound. Similarly, in the analysis of the stability of the generator δ_{t+1}^v, we examine two different cases of stochastic samples: $\zeta_t =$ or $\neq 1$ to obtain the desired bound.

Based on Proposition 2, we obtain the following result on the generalization error of SGM for the GAN training.

Theorem 3. *Let Assumption 1 hold. Apply SGM to solve the ERM in eq.* (14) *with the dataset S and denote the corresponding output at T-th iteration as $(\mathbf{w}_{T,S}, \mathbf{v}_{T,S})$. Choose the stepsize $\eta_t = \frac{c}{t \log t}$ with $c \leq (2L_f^w + \sigma_f^x L_g + \sigma_g^2 L_f^x)^{-1}$. Then, the generalization error of SGM satisfies*

$$\mathbb{E}_S \mathbb{E}_{sgm}\big[L(\mathbf{w}_{T,S}, \mathbf{v}_{T,S}) - L_S(\mathbf{w}_{T,S}, \mathbf{v}_{T,S})\big] \leq 2\sqrt{(2\sigma_f^x)^2 + (\sigma_f^w \sigma_g)^2}\left(\frac{\sigma_f^w + \sigma_g \sigma_f^x}{m} + \frac{\sigma_f^w}{n}\right)\log T.$$

Proof. See Section F. □

Theorem 3 provides a generalization error bound for SGM in training GANs. The generalization bound conveys several insights. First, the bound vanishes as the number of samples $m, n \to \infty$, implying that SGM can generalize well in training GANs given enough data samples. Second, the bound has a logarithm dependence on T. Hence, SGM can still generalize well after conducting multiple passes over the data during the training process. Last, the coefficient associated with m (the sample size of S_z) is larger than that associated with n (the sample size of S_x). This implies that the training of the generator requires more samples than that required by that of the discriminator. This is consistent with the empirical observations that the training of the discriminator typically converges faster than that of the generator, and agrees with Theorem 1 on the estimation error. Moreover, for the fully-connected deep neural network models discussed in the previous section, the Lipschitz constants σ_f^x and σ_g can be specified by the quantities U_w and U_v defined in Theorem 1. Hence, one can incoporate these quantities into the above bound to obtain a generalization error of SGM under neural network models.

IV. Experiments

In this section, we conduct experiments on the GAN training to validate our theoretical results from two perspectives: 1) justifying the new setting considered in our theory, where the number of noise samples by the generator is assumed to be limited, and 2) validating our generalization bounds under such this new setting.

A. Parameter Setting and Performance Metrics

Our implementation is adapted from the existing repository [44] (github.com/xuqiantong/GAN-Metrics) on DCGANs. The experiment is performed on one commonly used dataset CIFAR-10, which consists of 50000 training samples and 10000 test samples. We adopt the same hyper-parameter (i.e., learning rate, optimizer, batch size, noise dimension) settings as in [44], and use 2000 test samples for performance evaluation of the trained generative model based on three metrics: Frechet Inception Distance (FID) [45], Model Score (MS) [46], and generalization error, which we describe as follows.

Frechet Inception Distance (FID): Let P_r and P_g be the true distribution and the generated distribution, respectively. Given the inception network's feature function f, FID models $f(P_r)$ and $f(P_g)$ as two multivariate Gaussians with means μ_r and μ_g and covariances Σ_r and Σ_g. Then, the FID between P_r and P_g is given by

$$\text{FID}(P_r, P_g) = \|\mu_r - \mu_g\| + \text{Trace}(\Sigma_r + \Sigma_g - 2(\Sigma_r \Sigma_g)^{1/2}).$$

The lower FID value implies a better quality and diversity of the generated image.

Model Score (MS): MS is an improved version of the Inception Score (IS) given by

$$\text{MS}(P_g) = \exp(\mathbb{E}_{x \sim P_g}(D_{KL}(P_\phi(y|x)||P_\phi(y)) - D_{KL}(P_\phi(y)||P_\phi(y^*)))),$$

where $P_\phi(y|x)$ is the label distribution of x predicted by a classification model ϕ, and $P_M(y)$ and $P_M(y^*)$ are the marginal distributions over P_g and P_r, respectively. The higher MS value implies a better quality and diversity of the generated images.

Generalization Error (GE): Both MS and FID measure the quality of generated samples rather than the generalization error

studied in our theory. To capture the generalization performance, let $\mathbf{w}_S$ and $\mathbf{v}_S$ be the well-trained discriminator and generator network parameters. Then,

$$d_{f(\mathbf{w}_S)}(p_x, p_y) = \mathbb{E}_{\mathbf{x}\sim p_x} f(\mathbf{w}_S; \mathbf{x}) - \mathbb{E}_{\mathbf{y}\sim p_y} f(\mathbf{w}_S; \mathbf{y})$$

represents the distance between the two distributions p_x and p_y, measured by the trained discriminator $\mathbf{w}_S$. Here, we use $\mathbf{w}_S$ to approximate the neural net distance defined in eq. (1) for the ease of empirical evaluation. Then, if $\widehat{p}_x$ and $\widehat{p}_g$ denote the empirical distributions over the **training samples**, then $d_{f(\mathbf{w}_S)}(\widehat{p}_x, \widehat{p}_{g(\mathbf{v}_S;Z)})$ represents the training performance of the trained generator $\mathbf{v}_S$; and if $\widetilde{p}_x, \widetilde{p}_g$ denote the empirical distributions over the **test samples**, then $d_{f(\mathbf{w}_S)}(\widetilde{p}_x, \widetilde{p}_{g(\mathbf{v}_S;Z)})$ represents the test performance of the trained generator $\mathbf{v}_S$. Therefore, it is natural to define the distance between the training and test performances as the generalization error, which is given by

$$\text{GE} := d_{f(\mathbf{w}_S)}(\widetilde{p}_x, \widetilde{p}_{g(\mathbf{v}_S;Z)}) - d_{f(\mathbf{w}_S)}(\widehat{p}_x, \widehat{p}_{g(\mathbf{v}_S;Z)}).$$

B. GAN Training with Limited Latent Samples

The conventional GAN training considers a setting where the number of latent samples is supposed to be unlimited, and draws *fresh* latent samples at each iteration. As a comparison, our theory considers the setting where the latent (noise) samples of the generator are constrained to a finite set. To implement such a setting, we draw a limited number of latent samples $\{z_1, ..., z_m\}$ to construct a set Z_{pre} before training. Then, at each iteration during the training, we randomly choose latent samples from the set Z_{pre} rather than sampling fresh data.

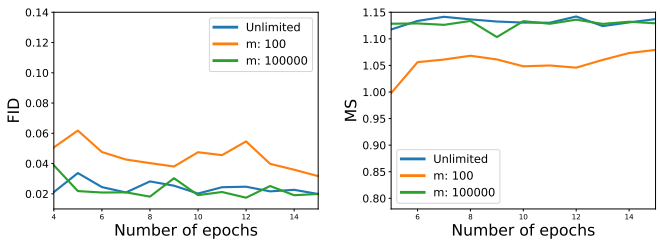

Fig. 1. FID and MS scores of DCGAN under different sizes of latent sample set. The left figure plots FID v.s. the number of epochs and the right figure plots MS v.s. the number of epochs. A lower FID or a higher MS implies a better quality of generated sample. Scores are averaged over 10 trials with different random seeds.

Our results are shown in Figure 1, where the "unlimited" line refers to the conventional setting with fresh latent samples drawn at each iteration. It can be seen that the FID and MS curves of the $m = 100000$ and unlimited cases are very close to each other. This validates the limited latent sample setting used in our theoretical analysis.

In Figure 2 and Figure 3, we plot the generated images at epochs 0 and 20, respectively. It can be seen that for both the unlimited and $m = 100000$ cases, the quality and diversity of the generated images are improved as the algorithm runs. We want to mention that the low-quality generated samples do not necessarily mean poor generalization and may be due to poor optimization performance.

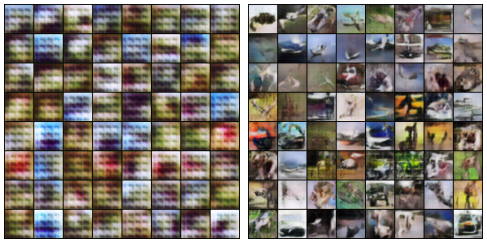

Fig. 2. Generated images at epoch 0 (left) and epoch 20 (right) with $m = 100000$.

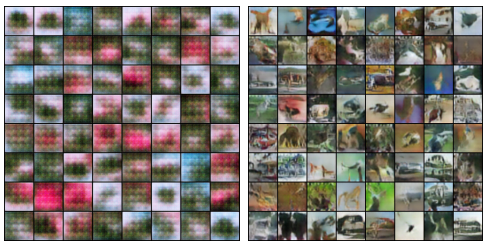

Fig. 3. Generated images at epoch 0 (left) and epoch 20 (right) with unlimited latent samples.

C. Generalization Performance of GANs

As shown in Table I, GE of DCGAN decreases as the size m of latent sample set increases. This verifies our theory that the generalization error decreases with respect to the number m of latent samples.

TABLE I
GENERALIZATION PERFORMANCE OF GANS V.S. SAMPLE SIZE m

m	100	500	1000	50000	100000	unlimited
GE	0.058	0.053	0.051	0.041	0.031	0.031

V. CONCLUSION

In this paper, we provided an upper bound as well as a minimax lower bound on the estimation error. Our upper bound captures the interplay between the discriminator and the generator on the estimation error, and furthermore matches the lower bound in terms of the convergence rate with respect to the sample size and the norm of the parameter matrices of neural networks. We also developed the generalization error bound for the stochastic gradient method (SGM) in training GANs. Such a bound not only characterizes how the discriminator interacts with the generator in the training process, but also shows that SGM can generalize well in training GANs with multiple passes over the data. Our experiments on DCGANs validate our theoretical results.

APPENDIX A
PROOF OF THEOREM 1 AND COROLLARY 1

A. Proof of Theorem 1

Theorem 1. *Let $\mathcal{P}_X$ be the class of Borel probability measures over the compact domain $\mathcal{X}$. Let $\mathcal{F}_w$ and $\mathcal{G}_v$ be the discriminator and generator classes given by eq.* (7) *and eq.* (8),

respectively. Consider a target distribution $p_x \in \mathcal{P}_X$ and the trained generator $\widehat{\mathbf{v}}^*$ given by eq. (6)*. Then, with probability at least $1-2\delta$ over the randomness of the training samples,*

$$\begin{aligned} d_{\mathcal{F}_{nn}}(p_x, p_{g(\widehat{\mathbf{v}}^*;Z)}) - \inf_{\mathbf{v}\in V} d_{\mathcal{F}_{nn}}(p_x, p_{g(\mathbf{v};Z)}) \\ \leq & 4\mathcal{R}(\mathcal{F}_W) + 4\mathcal{R}(\mathcal{H}_{W\times V}) \\ & + 2U_w\sqrt{2\log\frac{1}{\delta}}\Big(\frac{B_x}{\sqrt{n}} + \frac{B_z U_v}{\sqrt{m}}\Big), \quad (10) \end{aligned}$$

where parameters $U_w = \prod_{i=1}^{d} M_w(i)\prod_{i=1}^{d-1} L_w(i)$ and $U_v = \prod_{j=1}^{s} M_v(j)\prod_{i=1}^{s-1} L_v(i)$.

Proof. Let $\widetilde{\mathbf{v}} = \arg\min_{\mathbf{v}\in V} = d_{\mathcal{F}_{nn}}\left(\widehat{p}_x, p_{g(\mathbf{v};Z)}\right)$. We have

$$\begin{aligned} & d_{\mathcal{F}_{nn}}(p_x, p_{g(\widehat{\mathbf{v}}^*;Z)}) - \inf_{\mathbf{v}\in V} d_{\mathcal{F}_{nn}}(p_x, p_{g(\mathbf{v};Z)}) \\ & = \underbrace{d_{\mathcal{F}_{nn}}(p_x, p_{g(\widehat{\mathbf{v}}^*;Z)}) - d_{\mathcal{F}_{nn}}(\widehat{p}_x, p_{g(\widehat{\mathbf{v}}^*;Z)})}_{\text{(I)}} \\ & \quad + \underbrace{\inf_{\mathbf{v}\in V} d_{\mathcal{F}_{nn}}(\widehat{p}_x, p_{g(\mathbf{v};Z)}) - \inf_{\mathbf{v}\in V} d_{\mathcal{F}_{nn}}(p_x, p_{g(\mathbf{v};Z)})}_{\text{(II)}} \\ & \quad + \underbrace{d_{\mathcal{F}_{nn}}(\widehat{p}_x, p_{g(\widehat{\mathbf{v}}^*;Z)}) - \inf_{\mathbf{v}\in V} d_{\mathcal{F}_{nn}}(\widehat{p}_x, p_{g(\mathbf{v};Z)})}_{\text{(III)}}. \quad (16) \end{aligned}$$

We next upper-bound eq. (16) through the following three steps.

Step 1: bound (I). By the definition of $d_{\mathcal{F}_{nn}}\left(p_x, p_{g(\mathbf{v};Z)}\right)$ given by eq. (1), we have

$$\begin{aligned} & d_{\mathcal{F}_{nn}}(p_x, p_{g(\widehat{\mathbf{v}}^*;Z)}) - d_{\mathcal{F}_{nn}}(\widehat{p}_x, p_{g(\widehat{\mathbf{v}}^*;Z)}) \\ & = \sup_{\mathbf{w}\in W}\left[\mathbb{E}_{\mathbf{x}\sim p_x} f(\mathbf{w};\mathbf{x}) - \mathbb{E}_{Z\sim p_z} f(\mathbf{w}; g(\widehat{\mathbf{v}}^*; Z))\right] \\ & \quad - \sup_{\mathbf{w}\in W}\left[\mathbb{E}_{\mathbf{x}\sim \widehat{p}_x} f(\mathbf{w};\mathbf{x}) - \mathbb{E}_{Z\sim p_z} f(\mathbf{w}; g(\widehat{\mathbf{v}}^*; Z))\right] \\ & \overset{\text{(i)}}{\leq} \sup_{\mathbf{w}\in W}\left|\mathbb{E}_{\mathbf{x}\sim p_x} f(\mathbf{w};\mathbf{x}) - \mathbb{E}_{\mathbf{x}\sim \widehat{p}_x} f(\mathbf{w};\mathbf{x})\right|, \quad (17) \end{aligned}$$

where (i) follows from the inequality that $\sup x - \sup y \leq \sup(x-y) \leq \sup|x-y|$.

Step 2: bound (II). Let $\mathbf{v}^* = \arg\min_{\mathbf{v}\in V} d_{\mathcal{F}_{nn}}(p_x, p_{g(\mathbf{v};Z)})$. Similarly to (i), we obtain

$$\begin{aligned} & \inf_{\mathbf{v}\in V} d_{\mathcal{F}_{nn}}(\widehat{p}_x, p_{g(\mathbf{v};Z)}) - \inf_{\mathbf{v}\in V} d_{\mathcal{F}_{nn}}(p_x, p_{g(\mathbf{v};Z)}) \\ & \leq d_{\mathcal{F}_{nn}}(\widehat{p}_x, p_{g(\mathbf{v}^*;Z)}) - d_{\mathcal{F}_{nn}}(p_x, p_{g(\mathbf{v}^*;Z)}) \\ & \overset{\text{(i)}}{\leq} \sup_{\mathbf{w}\in W}\left|\mathbb{E}_{\mathbf{x}\sim p_x} f(\mathbf{w};\mathbf{x}) - \mathbb{E}_{\mathbf{x}\sim \widehat{p}_x} f(\mathbf{w};\mathbf{x})\right|, \end{aligned}$$

where (i) follows from the same steps as in eq. (17).

Step 3: bound (III). This step captures the role of the generator in the estimation error. Letting $\widetilde{\mathbf{v}} = \arg\min_{\mathbf{v}\in V} d_{\mathcal{F}_{nn}}\left(\widehat{p}_x, p_{g(\mathbf{v};Z)}\right)$, we can obtain

$$\begin{aligned} & d_{\mathcal{F}_{nn}}(\widehat{p}_x, p_{g(\widehat{\mathbf{v}}^*;Z)}) - \inf_{\mathbf{v}\in V} d_{\mathcal{F}_{nn}}(\widehat{p}_x, p_{g(\mathbf{v};Z)}) \\ & \leq d_{\mathcal{F}_{nn}}(\widehat{p}_x, p_{g(\widehat{\mathbf{v}}^*;Z)}) - d_{\mathcal{F}_{nn}}(\widehat{p}_x, \widehat{p}_{g(\widehat{\mathbf{v}}^*;Z)}) \\ & \quad + d_{\mathcal{F}_{nn}}(\widehat{p}_x, \widehat{p}_{g(\widetilde{\mathbf{v}};Z)}) - d_{\mathcal{F}_{nn}}(\widehat{p}_x, p_{g(\widetilde{\mathbf{v}};Z)}) \\ & \leq \sup_{\mathbf{w}\in W}\left|\mathbb{E}_{Z\sim p_z} f(\mathbf{w}; g(\widehat{\mathbf{v}}^*; Z)) - \mathbb{E}_{Z\sim \widehat{p}_z} f(\mathbf{w}; g(\widehat{\mathbf{v}}^*; Z)\right| \\ & \quad + \sup_{\mathbf{w}\in W}\left|\mathbb{E}_{Z\sim p_z} f(\mathbf{w}; g(\widetilde{\mathbf{v}}; Z)) - \mathbb{E}_{Z\sim \widehat{p}_z} f(\mathbf{w}; g(\widetilde{\mathbf{v}}; Z)\right| \\ & \leq 2\sup_{\mathbf{v}\in V}\sup_{\mathbf{w}\in W}\left|\mathbb{E}_{Z\sim p_z} f(\mathbf{w}; g(\mathbf{v}; Z)) - \mathbb{E}_{Z\sim \widehat{p}_z} f(\mathbf{w}; g(\mathbf{v}; Z)\right| \\ & = 2\sup_{\mathbf{w}\in W,\mathbf{v}\in V}\left|\mathbb{E}_{Z\sim p_z} f(\mathbf{w}; g(\mathbf{v}; Z)) - \mathbb{E}_{Z\sim \widehat{p}_z} f(\mathbf{w}; g(\mathbf{v}; Z)\right|. \end{aligned}$$

Combining the above three steps yields

$$\begin{aligned} & d_{\mathcal{F}_{nn}}(p_x, p_{g(\widehat{\mathbf{v}}^*;Z)}) \\ & \leq \inf_{\mathbf{v}} d_{\mathcal{F}_{nn}}(p_x, p_{g(\mathbf{v};Z)}) + 2\sup_{\mathbf{w}}\left|\mathbb{E}_{\mathbf{x}\sim p_x} f(\mathbf{w};\mathbf{x}) - \mathbb{E}_{\mathbf{x}\sim \widehat{p}_x} f(\mathbf{w};\mathbf{x})\right| \\ & + 2\sup_{\mathbf{w}\in W,\mathbf{v}\in V}\left|\mathbb{E}_{Z\sim p_z} f(\mathbf{w}; g(\mathbf{v}; Z)) - \mathbb{E}_{Z\sim \widehat{p}_z} f(\mathbf{w}; g(\mathbf{v}; Z)\right|. \\ & = \inf_{\mathbf{v}} d_{\mathcal{F}_{nn}}(p_x, p_{g(\mathbf{v};Z)}) + \underbrace{2\sup_{\mathbf{w}}\left|\mathbb{E}_{\mathbf{x}\sim p_x} f(\mathbf{w};\mathbf{x}) - \frac{1}{n}\sum_{i=1}^{n} f(\mathbf{w};\mathbf{x}_i)\right|}_{F(\mathbf{x}_1,\ldots,\mathbf{x}_n)} \\ & + \underbrace{2\sup_{\mathbf{w},\mathbf{v}}\left|\mathbb{E}_{Z\sim p_z} f(\mathbf{w}; g(\mathbf{v}; Z)) - \frac{1}{m}\sum_{i=1}^{m} f(\mathbf{w}; g(\mathbf{v}; Z_i)\right|}_{G(Z_1,\ldots,Z_m)}. \quad (18) \end{aligned}$$

Our next step is to upper-bound $G(Z_1, ..., Z_m)$ in eq. (18). Based on the inequality that $\sup|x| - \sup|y| \leq \sup(|x|-|y|) \leq \sup|x-y|$, we have, for any $Z_1, ..., Z_i, ..., Z_m, Z_i'$

$$\begin{aligned} & |G(Z_1, ...,Z_i, ..., Z_m) - G(Z_1, ..., Z_i', ..., Z_m)| \\ & \leq \sup_{\mathbf{w}\in W,\mathbf{v}\in V}\left|f(\mathbf{w}; g(\mathbf{v}; Z_i)) - f(\mathbf{w}; g(\mathbf{v}; Z_i'))\right|/m, \end{aligned}$$

which, using the Cauchy-Schwarz inequality, is upper-bounded by $2Q_z/m$ with Q_z given by

$$Q_z = B_z\prod_{i=1}^{d-1} L_w(i)\prod_{i=1}^{s-1} L_v(i)\prod_{i=1}^{d} M_w(i)\prod_{i=1}^{s} M_v(i).$$

Combining the above result with the standard McDiarmid's inequality, we have, with probability at least $1-\delta$,

$$\begin{aligned} & G(Z_1, ..., Z_i, ..., Z_m) \\ & \leq \mathbb{E}_Z G(Z_1, ..., Z_i, ..., Z_m) + 2Q_z\sqrt{\frac{\log(1/\delta)}{2m}}. \quad (19) \end{aligned}$$

The expectation term in eq. (19) is upper-bounded through the following standard steps

$$\begin{aligned} & \mathbb{E}_Z G(Z_1, ..., Z_i, ..., Z_m) \\ & = \mathbb{E}_Z \sup_{\mathbf{w}\in W}\left|\mathbb{E}_{\widetilde{Z}}\frac{1}{m}\sum_{i=1}^{m} f(\mathbf{w}; g(\mathbf{v}; \widetilde{Z}_i)) - \frac{1}{m}\sum_{i=1}^{m} f(\mathbf{w}; g(\mathbf{v}; Z_i))\right| \\ & \overset{\text{(i)}}{\leq} \mathbb{E}_{Z,\widetilde{Z}} \sup_{\mathbf{w}\in W}\left|\frac{1}{m}\sum_{i=1}^{m} f(\mathbf{w}; g(\mathbf{v}; \widetilde{Z}_i)) - \frac{1}{m}\sum_{i=1}^{m} f(\mathbf{w}; g(\mathbf{v}; Z_i))\right| \\ & = \mathbb{E}_{Z,\widetilde{Z},\epsilon} \sup_{\mathbf{w}\in W}\left|\frac{1}{m}\sum_{i=1}^{m} \epsilon_i\big(f(\mathbf{w}; g(\mathbf{v}; \widetilde{Z}_i)) - f(\mathbf{w}; g(\mathbf{v}; Z_i))\big)\right| \\ & \leq 2\mathbb{E}_{Z,\epsilon}\sup_{\mathbf{w}}\left|\frac{1}{m}\sum_{i=1}^{m}\epsilon_i f(\mathbf{w}; g(\mathbf{v}; Z_i))\right| = 2\mathcal{R}(\mathcal{H}_{W\times V}) \quad (20) \end{aligned}$$

where (i) follows from the Jensen's inequality. Using an approach similar to eq. (20), we have, with probability at least $1-\delta$

$$F(\mathbf{x}_1, ..., \mathbf{x}_i, ..., \mathbf{x}_n) \leq 2\mathcal{R}(\mathcal{F}_W) + 2Q_x\sqrt{\frac{\log(1/\delta)}{2n}}, \quad (21)$$

where $Q_x = B_x\prod_{i=1}^{d-1} L_w(i)\prod_{i=1}^{d} M_w(i)$.

Combining eqs. (18), (20) and (21), and a union bound implies that

$$d_{\mathcal{F}_{nn}}(p_x,p_{g(\widehat{\mathbf{v}}^*;Z)}) - \inf_{\mathbf{v}\in V} d_{\mathcal{F}_{nn}}(p_x,p_{g(\mathbf{v};Z)}) \leq 4\mathcal{R}(\mathcal{F}_W) + 4\mathcal{R}(\mathcal{H}_{W\times V}) + \frac{2Q_x\sqrt{2\log(1/\delta)}}{\sqrt{n}} + \frac{2Q_z\sqrt{2\log(1/\delta)}}{\sqrt{m}}. \tag{22}$$

B. Proof of Corollary 1

We first upper bound the Rademacher complexity in Theorem 1. First note that

$$\mathcal{R}(\mathcal{F}_W) = \mathbb{E}_{\mathbf{x},\epsilon}\sup_{\mathbf{w}\in W}\left|\frac{1}{n}\sum_{i=1}^n \epsilon_i f(\mathbf{w};\mathbf{x}_i)\right| = \mathbb{E}_{\mathbf{x}}\left(\mathbb{E}_\epsilon\left(\sup_{\mathbf{w}\in W}\left|\frac{1}{n}\sum_{i=1}^n \epsilon_i f(\mathbf{w};\mathbf{x}_i)\right|\right)\Big|\mathbf{x}_1,...,\mathbf{x}_n\right). \tag{23}$$

Conditioned on $\mathbf{x}_1,...,\mathbf{x}_n$, we define

$$\widetilde{\mathcal{R}}_n(\mathcal{F}_W) = \mathbb{E}_\epsilon \sup_{\mathbf{w}\in W}\left|\frac{1}{n}\sum_{i=1}^n \epsilon_i f(\mathbf{w};\mathbf{x}_i)\right|.$$

Letting $F_i(\mathbf{x}) = \sigma_{i-1}(\mathbf{W}_{i-1}\sigma_{i-2}(\cdots\sigma_1(\mathbf{W}_1\mathbf{x}))$, we have

$$\begin{aligned} &n\lambda\widetilde{\mathcal{R}}_n(\mathcal{F}_W)\\ &= \lambda\mathbb{E}_\epsilon \sup_{F,\mathbf{w},\mathbf{W}}\Big|\sum_{i=1}^n \epsilon_i \mathbf{w}_d^T\sigma_{d-1}(\mathbf{W}_{d-1}F_{d-1}(\mathbf{x}_i))\Big|\\ &= \log\Big(\exp\lambda\Big(\mathbb{E}_\epsilon \sup_{F,\mathbf{w},\mathbf{W}}\Big|\sum_{i=1}^n \epsilon_i \mathbf{w}_d^T\sigma_{d-1}(\mathbf{W}_{d-1}F_{d-1}(\mathbf{x}_i))\Big|\Big)\Big)\\ &\overset{\text{(i)}}{\leq} \log\Big(\mathbb{E}_\epsilon \sup_{F,\mathbf{w},\mathbf{W}}\exp\lambda\Big(\Big|\sum_{i=1}^n \epsilon_i \mathbf{w}_d^T\sigma_{d-1}(\mathbf{W}_{d-1}F_{d-1}(\mathbf{x}_i))\Big|\Big)\Big)\\ &\leq \log\Big(\mathbb{E}_\epsilon \sup_{F,\mathbf{W}}\exp\lambda\Big(M_w(d)\Big\|\sum_{i=1}^n \epsilon_i \sigma_{d-1}(\mathbf{W}_{d-1}F_{d-1}(\mathbf{x}_i))\Big\|\Big)\Big), \end{aligned} \tag{24}$$

where (i) follows from the Jensen's inequality. In our case, we have $\|\mathbf{x}_i\| \leq B_x$ for $i = 1,...,n$. Thus, combining eq. (24) with the steps of the proof of Theorem 1 in [21], we have

$$\hat{\mathcal{R}}_n(\mathcal{F}_W) \leq \frac{B_x\prod_{i=1}^d M_w(i)\prod_{i=1}^{d-1}L_w(i)(\sqrt{2d\log 2}+1)}{\sqrt{n}},$$

which, in conjunction with eq. (23) and the fact that $\sqrt{2d\log 2}+1 \leq \sqrt{3d}$, implies that

$$\mathcal{R}(\mathcal{F}_W) = \mathbb{E}_{\mathbf{x}}(\hat{\mathcal{R}}_n(\mathcal{F}_W)) \leq \frac{B_x\prod_{i=1}^d M_w(i)\prod_{i=1}^{d-1}L_w(i)\sqrt{3d}}{\sqrt{n}}.$$

We next upper-bound $\mathcal{R}(\mathcal{H}_{W\times V})$, which is given by

$$\mathcal{R}(\mathcal{H}_{W\times V}) = \underbrace{\mathbb{E}_{Z,\epsilon}\sup_{\mathbf{w}\in W,\mathbf{v}\in V}\left|\frac{1}{m}\sum_{i=1}^m \epsilon_i f(\mathbf{w};g(\mathbf{v};Z_i))\right|}_{\mathbb{E}_Z(\hat{\mathcal{R}}_m(\mathcal{H}_{W\times V}))}.$$

Note that each function $f(\mathbf{w};g(\mathbf{v};\cdot))$ can be regarded as a concatenation of the discriminator and generator neural networks, which takes the form as $f(\mathbf{w};g(\mathbf{v};\cdot)) = \mathbf{w}_d^T\sigma_{d-1}(\mathbf{W}_{d-1}\cdots\sigma_1(\mathbf{W}_1\mathbf{V}_s\phi_{s-1}(\cdots\phi_1(\mathbf{V}_1(\cdot)))))$. Then, taking an approach similar to eq. (24) and using the fact that $\|\mathbf{W}_1\mathbf{V}_s\|_F \leq M_w(1)M_v(s)$, we finishes the proof. □

APPENDIX B
PROOF OF THEOREM 2

Theorem 2 (Minimax lower bound)**.** *Let $\mathcal{F}_w$ be the discriminator function class given by eq.* (7) *and $\widehat{p}_n$ be any estimator of the target distribution p_x constructed based on the samples $\{\mathbf{x}_i\}_{i=1}^n$. Then, we have*

$$\inf_{\widehat{p}_n}\sup_{p_x\in\mathcal{P}_X}\mathbb{P}\Big\{d_{\mathcal{F}_{nn}}(\widehat{p}_n,p_x) \geq \frac{C(\mathcal{P}_X)}{\sqrt{n}}\Big\} > 0.42, \tag{12}$$

where the constant $C(\mathcal{P}_X)$ is given by

$$C(\mathcal{P}_X) = 0.015\big(M_w(d)\sigma_{d-1}(\cdots(M_w(1)B_x)) - M_w(d)\sigma_{d-1}(\cdots(-M_w(1)B_x))\big). \tag{13}$$

Proof. The following Fano's inequality (Theorem 2.5 in [47]) is useful in the proof.

Lemma 2 (Fano's inequality)**.** *For $M \geq 2$, assume that there exist M hypotheses $\theta_0,...,\theta_M \in \Theta$ satisfying (i) $d(\theta_i,\theta_j) \geq 2s > 0$ for all $0 \leq i < j \leq M$; (ii) $\frac{1}{M}\sum_{i=1}^M \mathrm{KL}(P_{\theta_i}\|P_{\theta_0}) \leq \alpha\log M$, $0 < \alpha \leq 1/8$, where $d(\cdot,\cdot)$ is a semi-distance and P_θ is a probability measure with respect to the randomness of data D. Then, we have*

$$\inf_{\hat{\theta}}\sup_{\theta\in\Theta}\mathbb{P}_{D\sim P_\theta}\Big\{d(\hat{\theta},\theta) \geq s\Big\} \geq \frac{\sqrt{M}}{1+\sqrt{M}}\left(1-2\alpha-\frac{2\alpha}{\log M}\right),$$

where the infimum is taken over all estimators $\hat{\theta}$ of θ constructed based on the data D.

In our setting, we let Θ be a hypothesis set that contains all distributions in $\mathcal{P}_X$ and choose $d(\cdot,\cdot)$ in Lemma 2 as the neural distance $d_{\mathcal{F}_{nn}}(\cdot,\cdot)$. To use Lemma 2, we need to choose M distributions $\{p_i \in \mathcal{P}_X, i = 0,...,M\}$ such that (i) $d_{\mathcal{F}_{nn}}(p_i,p_j) \geq 2s > 0$ for all $0 \leq i < j \leq M$ and (ii) $\frac{1}{M}\sum_{i=1}^M \mathrm{KL}(P_{\theta_i}\|P_{\theta_0}) \leq \alpha\log M$. In our proof, we choose $M = 2$ and consider the following three hypothesis distributions

$$p_0(\mathbf{x}) = \begin{cases}\frac{1}{2}, & \mathbf{x}=\mathbf{x}_1\\ \frac{1}{2}, & \mathbf{x}=-\mathbf{x}_1\end{cases} \quad p_1(\mathbf{x}) = \begin{cases}\frac{1}{2}-\epsilon_x, & \mathbf{x}=\mathbf{x}_1\\ \frac{1}{2}+\epsilon_x, & \mathbf{x}=-\mathbf{x}_1\end{cases}$$

$$p_2(\mathbf{x}) = \begin{cases}\frac{1}{2}-2\epsilon_x, & \mathbf{x}=\mathbf{x}_1\\ \frac{1}{2}+2\epsilon_x, & \mathbf{x}=-\mathbf{x}_1,\end{cases} \tag{25}$$

where $\|\mathbf{x}_1\| = B_x$ and $\epsilon_x = \log(2)n^{-\frac{1}{2}}/10 < 1/4$.

First, we lower-bound $d_{\mathcal{F}_{nn}}(p_i,p_j)$. For $0 \leq i < j \leq 2$, we have

$$\begin{aligned} d_{\mathcal{F}_{nn}}(p_i,p_j) &= \sup_{\mathbf{w}\in W}\big|\mathbb{E}_{\mathbf{x}\sim p_i}f(\mathbf{w};\mathbf{x}) - \mathbb{E}_{\mathbf{x}\sim p_j}f(\mathbf{w};\mathbf{x})\big|\\ &= (j-i)\epsilon_x\sup_{\mathbf{w}\in W}|f(\mathbf{w};\mathbf{x}_1) - f(\mathbf{w};-\mathbf{x}_1)|\\ &\geq \epsilon_x\sup_{\mathbf{w}\in W}|f(\mathbf{w};\mathbf{x}_1) - f(\mathbf{w};-\mathbf{x}_1)|. \end{aligned} \tag{26}$$

Next, we select $\widetilde{\mathbf{w}} \in W$ in eq. (7) as following.

- $\widetilde{\mathbf{w}}_d(1) = M_w(d), \widetilde{\mathbf{w}}_d(i) = 0$ for $i = 2, 3, ..., n_d$,
- For $i = 2, ..., d-1$, $\widetilde{\mathbf{W}}_i(1,1) = M_w(i), \widetilde{\mathbf{W}}_i(s,t) = 0$, for $(s,t) \neq (1,1)$,
- $\|\widetilde{\mathbf{W}}_1(1)\| = \mathbf{w}_1 = M_w(1)\mathbf{x}_1/\|\mathbf{x}_1\|, \widetilde{\mathbf{W}}_1(s) = \mathbf{0}$, for $2 \leq s \leq n_1$, (27)

where $\widetilde{\mathbf{w}}_d(i)$ refers to the i^{th} coordinate of $\widetilde{\mathbf{w}}_d$, $\widetilde{\mathbf{W}}_i(s,t)$ denotes the $(s,t)^{th}$ entry of $\widetilde{\mathbf{W}}$, $\widetilde{\mathbf{W}}_1(s)$ is the s^{th} column vector of $\widetilde{\mathbf{W}}_1^T$.

Combining eqs. (26) and (27) yields

$$\begin{aligned} d_{\mathcal{F}_{nn}}(p_i, p_j) \geq & \epsilon_x (M_w(d)\sigma_{d-1}(\cdots(M_w(1)B_x)) \\ & - M_w(d)\sigma_{d-1}(\cdots(-M_w(1)B_x))). \end{aligned} \quad (28)$$

Next, we upper-bound $\frac{1}{2}\sum_{i=1}^{2} \mathrm{KL}(P_{\theta_i} \| P_{\theta_0})$. Using the properties of KL-divergence, we obtain

$$\begin{aligned} &\mathrm{KL}(P_{\theta_i} \| P_{\theta_0}) = n\mathrm{KL}(p_i \| p_0) \\ =& n\left(\frac{1}{2} - i\epsilon_x\right)\log(1 - 2i\epsilon_x) + n\left(\frac{1}{2} + i\epsilon_x\right)\log(1 + 2i\epsilon_x) \\ =& \frac{n}{2}\log(1 - 4i^2\epsilon_x^2) + ni\epsilon_x \log\left(1 + \frac{4i\epsilon_x}{1 - 2i\epsilon_x}\right) \\ \overset{\text{(i)}}{\leq}& 4ni^2\epsilon_x^2, \end{aligned}$$

where (i) follows from the fact that $\log(1 + x) \leq x$. Then, we obtain

$$\frac{1}{2}\sum_{i=1}^{2} \mathrm{KL}(P_{\theta_i} \| P_{\theta_0}) \leq 10n\epsilon_x^2 \leq \frac{\log^2(2)}{10}.$$

Combining the above inequality with eq. (28) and Lemma 2, we have

$$\begin{aligned} \inf_{\hat{p}_n} \sup_{p_x \in \mathcal{P}_X} & \mathbb{P}\left\{ d_{\mathcal{F}_{nn}}(\hat{p}_n, p_x) \geq C(\mathcal{P}_X) n^{-1/2} \right\} \\ & \geq \frac{\sqrt{2}}{1 + \sqrt{2}}\left(\frac{4}{5} - \frac{\log(2)}{5}\right) > 0.42, \end{aligned}$$

where the constant $C(\mathcal{P}_X)$ is given by

$$\begin{aligned} C(\mathcal{P}_X) = & \frac{\log(2)}{20}(M_w(d)\sigma_{d-1}(\cdots(M_w(1)B_x)) \\ & - M_w(d)\sigma_{d-1}(\cdots(-M_w(1)B_x))), \end{aligned}$$

which, combined with the fact that $\log(2)/20 \geq 0.015$, finishes the proof. □

APPENDIX C
PROOF OF PROPOSITION 1

Proposition 1. *Let SGM be ϵ_f uniform-discriminator stable and ϵ_g uniform-generator stable. Then, the generalization error induced by the output of SGM in the GAN training satisfies*

$$\mathbb{E}_S \mathbb{E}_{sgm}\big[L(\mathbf{w}_S, \mathbf{v}_S) - L_S(\mathbf{w}_S, \mathbf{v}_S)\big] \leq \epsilon_f + \epsilon_g.$$

Proof. We denote $S = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \cup \{Z_1, \ldots, Z_m\}, S' = \{\mathbf{x}'_1, \ldots, \mathbf{x}'_n\} \cup \{Z'_1, \ldots, Z'_m\}, S^{i,j} = \{\mathbf{x}_1, \ldots, \mathbf{x}'_i, \ldots, \mathbf{x}_n\} \cup \{Z_1, \ldots, Z'_j, \ldots, Z_m\}$. Note that

$$\begin{aligned} \mathbb{E}_S \mathbb{E}_A\left[L_S(\mathbf{w}_S, \mathbf{v}_S)\right] = & \underbrace{\mathbb{E}_S \mathbb{E}_A\left[\frac{1}{n}\sum_{i=1}^{n} f(\mathbf{w}_S; \mathbf{x}_i)\right]}_{P} \\ & - \underbrace{\mathbb{E}_S \mathbb{E}_A\left[\frac{1}{m}\sum_{j=1}^{m} f(\mathbf{w}_S; g(\mathbf{v}_S; Z_j))\right]}_{Q}. \end{aligned}$$

The two terms P, Q can be further rewritten as

$$\begin{aligned} P &= \mathbb{E}_{SS'}\mathbb{E}_A\left[\frac{1}{n}\sum_{i=1}^{n} f(\mathbf{w}_{S^{i,j}}; \mathbf{x}'_i)\right] \\ &= \mathbb{E}_S \mathbb{E}_A\Big[\mathbb{E}_{\mathbf{x}} f(\mathbf{w}_S; \mathbf{x})\Big] + \mathbb{E}_{SS'}\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}_A[f(\mathbf{w}_{S^{i,j}}; \mathbf{x}'_i) - f(\mathbf{w}_S; \mathbf{x}'_i)] \\ Q &= \mathbb{E}_{SS'}\mathbb{E}_A\left[\frac{1}{m}\sum_{j=1}^{m} f(\mathbf{w}_{S^{i,j}}; g(\mathbf{v}_{S^{i,j}}; Z'_j))\right] \\ &= \mathbb{E}_{SS'}\mathbb{E}_A\left[\frac{1}{m}\sum_{j=1}^{m} f(\mathbf{w}_S; g(\mathbf{v}_S; Z'_j)) - \frac{1}{m}\sum_{j=1}^{m} f(\mathbf{w}_S; g(\mathbf{v}_S; Z'_j))\right. \\ &\quad \left. + \frac{1}{m}\sum_{j=1}^{m} f(\mathbf{w}_{S^{i,j}}; g(\mathbf{v}_{S^{i,j}}; Z'_j))\right] \\ &= \mathbb{E}_S \mathbb{E}_A\Big[\mathbb{E}_Z f(\mathbf{w}_S; g(\mathbf{v}_S; Z))\Big] \\ &\quad - \mathbb{E}_{SS'}\frac{1}{m}\sum_{j=1}^{m}\mathbb{E}_A\Big[f(\mathbf{w}_S; g(\mathbf{v}_S; Z'_j)) - f(\mathbf{w}_{S^{i,j}}; g(\mathbf{v}_{S^{i,j}}; Z'_j))\Big], \end{aligned}$$

where we have used the fact that $\mathbf{x}'_i$ and Z'_j are i.i.d. copies of $\mathbf{x}_i$ and Z_j, respectively. Subtracting Q from P yields that

$$\begin{aligned} & P - Q \\ &= \mathbb{E}_S \mathbb{E}_A\Big[L(\mathbf{w}_S, \mathbf{v}_S)\Big] + \mathbb{E}_{SS'}\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}_A\Big[f(\mathbf{w}_{S^{i,j}}; \mathbf{x}'_i) - f(\mathbf{w}_S; \mathbf{x}'_i)\Big] \\ &\quad + \mathbb{E}_{SS'}\frac{1}{m}\sum_{j=1}^{m}\mathbb{E}_A\Big[f(\mathbf{w}_S; g(\mathbf{v}_S; Z'_j)) - f(\mathbf{w}_{S^{i,j}}; g(\mathbf{v}_{S^{i,j}}; Z'_j))\Big]. \end{aligned}$$

Note that $P - Q = \mathbb{E}_S \mathbb{E}_A\Big[L_S(\mathbf{w}_S, \mathbf{v}_S)\Big]$. Thus, we conclude that

$$\begin{aligned} & \mathbb{E}_S \mathbb{E}_A\Big[L(\mathbf{w}_S, \mathbf{v}_S) - L_S(\mathbf{w}_S, \mathbf{v}_S)\Big] \\ &= \mathbb{E}_{SS'}\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}_A\big|f(\mathbf{w}_{S^{i,j}}; \mathbf{x}'_i) - f(\mathbf{w}_S; \mathbf{x}'_i)\big| \\ &\quad + \mathbb{E}_{SS'}\frac{1}{m}\sum_{j=1}^{m}\mathbb{E}_A\big|f(\mathbf{w}_S; g(\mathbf{v}_S; Z'_j)) - f(\mathbf{w}_{S^{i,j}}; g(\mathbf{v}_{S^{i,j}}; Z'_j))\big| \\ &\leq \sup_{S, \bar{S}, \mathbf{x}} \mathbb{E}_A\big|f(\mathbf{w}_{\bar{S}}; \mathbf{x}) - f(\mathbf{w}_S; \mathbf{x})\big| \\ &\quad + \sup_{S, \bar{S}, Z} \mathbb{E}_A\big|f(\mathbf{w}_S; g(\mathbf{v}_S; Z)) - f(\mathbf{w}_{\bar{S}}; g(\mathbf{v}_{\bar{S}}; Z))\big| \\ &= \epsilon_f + \epsilon_g, \end{aligned}$$

where $\bar{S}$ is any such $S^{i,j}$. □

APPENDIX D
PROOF OF LEMMA 1

Lemma 1. *Let Assumption 1 hold. Apply SGM to solve the ERM in eq.* (14) *with the data sets S and $\bar{S}$, respectively, and denote the corresponding outputs as $(\mathbf{w}_S, \mathbf{v}_S)$ and $(\mathbf{w}_{\bar{S}}, \mathbf{v}_{\bar{S}})$, respectively. Then, the stabilities ϵ_f, ϵ_g of GAN satisfy*

$$\epsilon_f + \epsilon_g \leq 2\sigma_f^x \sup_{S,\bar{S}} \mathbb{E}_{sgm}\|\mathbf{w}_S - \mathbf{w}_{\bar{S}}\| + \sigma_f^w \sigma_g \sup_{S,\bar{S}} \mathbb{E}_{sgm}\|\mathbf{v}_S - \mathbf{v}_{\bar{S}}\|.$$

Proof. By Proposition 1 and Assumption 1, we obtain that

$$\begin{aligned}
&\mathbb{E}_S\mathbb{E}_{\text{sgm}}\big[L(\mathbf{w}_S, \mathbf{v}_S) - L_S(\mathbf{w}_S, \mathbf{v}_S)\big] \\
&\quad\leq \sup_{S,\bar{S},\mathbf{x}} \mathbb{E}_{\text{sgm}}\big|f(\mathbf{w}_{\bar{S}}; \mathbf{x}) - f(\mathbf{w}_S; \mathbf{x})\big| \\
&\qquad + \sup_{S,\bar{S},Z} \mathbb{E}_{\text{sgm}}\big|f(\mathbf{w}_S; g(\mathbf{v}_S; Z)) - f(\mathbf{w}_{\bar{S}}; g(\mathbf{v}_{\bar{S}}; Z))\big| \\
&\quad\leq \sup_{S,\bar{S},\mathbf{x}} \mathbb{E}_{\text{sgm}}\big|f(\mathbf{w}_{\bar{S}}; \mathbf{x}) - f(\mathbf{w}_S; \mathbf{x})\big| \\
&\qquad + \sup_{S,\bar{S},Z} \mathbb{E}_{\text{sgm}}\big|f(\mathbf{w}_S; g(\mathbf{v}_S; Z)) - f(\mathbf{w}_{\bar{S}}; g(\mathbf{v}_S; Z))\big| \\
&\qquad + \sup_{S,\bar{S},Z} \mathbb{E}_{\text{sgm}}\big|f(\mathbf{w}_{\bar{S}}; g(\mathbf{v}_S; Z)) - f(\mathbf{w}_{\bar{S}}; g(\mathbf{v}_{\bar{S}}; Z))\big| \\
&\quad\leq 2\sigma_f^w \mathbb{E}_{\text{sgm}}\|\mathbf{w}_{\bar{S}} - \mathbf{w}_S\| + \sigma_f^x \sigma_g \mathbb{E}_{\text{sgm}}\|\mathbf{v}_S - \mathbf{v}_{\bar{S}}\|,
\end{aligned}$$

which finishes the proof. □

APPENDIX E
PROOF OF PROPOSITION 2

Proposition 2 (Stability of SGM for GANs)**.** *Let Assumption 1 hold. Apply SGM to solve the ERM in eq.* (14) *with the datasets S and $\bar{S}$, respectively, and denote the corresponding outputs as $(\mathbf{w}_S, \mathbf{v}_S)$ and $(\mathbf{w}_{\bar{S}}, \mathbf{v}_{\bar{S}})$, respectively. Denote the stepsize as $\eta_t > 0$. Then, the stabilities of both the discriminator and the generator satisfy*

$$\begin{aligned}
\mathbb{E}_{sgm}\begin{bmatrix}\delta_{t+1}^w \\ \delta_{t+1}^v\end{bmatrix} &\leq \begin{bmatrix}1 + 2\eta_t L_f^w & \eta_t L_f^x \sigma_g \\ \eta_t L_f^w \sigma_g & 1 + \eta_t(\sigma_f^x L_g + \sigma_g^2 L_f^x)\end{bmatrix}\mathbb{E}_{sgm}\begin{bmatrix}\delta_t^w \\ \delta_t^v\end{bmatrix} \\
&\quad + 2\eta_t \begin{bmatrix}\frac{m+n}{mn}\sigma_f^w \\ \frac{\sigma_g \sigma_f^x}{m}\end{bmatrix}. \qquad (15)
\end{aligned}$$

Proof. Consider a pair of fixed data sets $S, \bar{S}$. Without of loss of generality, we assume that $S, \bar{S}$ are different at the samples with index 1, i.e., S contains $\mathbf{x}_1, Z_1$ and $\bar{S}$ contains $\mathbf{x}_1', Z_1'$, respectively.

We first bound δ_{t+1}^w. At iteration t, we consider the following four cases.

Case 1: $\xi_t \neq 1, \zeta_t \neq 1$**.**

By the uniform sampling, this case occurs with probability $\frac{n-1}{n}\frac{m-1}{m}$. Also, the update rule of SGD implies that

$$\begin{aligned}
\delta_{t+1}^w &= \|\mathbf{w}_{t,S} - \mathbf{w}_{t,\bar{S}} + \eta_t(\nabla_{\mathbf{w}} f(\mathbf{w}_{t,S}; \mathbf{x}_{\xi_t}) - \nabla_{\mathbf{w}} f(\mathbf{w}_{t,\bar{S}}; \mathbf{x}_{\xi_t}) \\
&\qquad + \nabla_{\mathbf{w}} f(\mathbf{w}_{t,\bar{S}}; g(\mathbf{v}_{t,\bar{S}}; Z_{\zeta_t})) - \nabla_{\mathbf{w}} f(\mathbf{w}_{t,S}; g(\mathbf{v}_{t,S}; Z_{\zeta_t})))\| \\
&\leq \delta_t^w + \eta_t L_f^w \delta_t^w \\
&\qquad + \eta_t\|\nabla_{\mathbf{w}} f(\mathbf{w}_{t,\bar{S}}; g(\mathbf{v}_{t,\bar{S}}; Z_{\zeta_t})) - \nabla_{\mathbf{w}} f(\mathbf{w}_{t,S}; g(\mathbf{v}_{t,\bar{S}}; Z_{\zeta_t})) \\
&\qquad + \nabla_{\mathbf{w}} f(\mathbf{w}_{t,S}; g(\mathbf{v}_{t,\bar{S}}; Z_{\zeta_t})) - \nabla_{\mathbf{w}} f(\mathbf{w}_{t,S}; g(\mathbf{v}_{t,S}; Z_{\zeta_t}))\| \\
&\leq (1 + \eta_t L_f^w)\delta_t^w + \eta_t L_f^w \delta_t^w + \eta_t L_f^x \sigma_g \delta_t^v \\
&= (1 + 2\eta_t L_f^w)\delta_t^w + \eta_t L_f^x \sigma_g \delta_t^v.
\end{aligned}$$

Case 2: $\xi_t = 1, \zeta_t \neq 1$**.**

This case occurs with probability $\frac{1}{n}\frac{m-1}{m}$. The update rule of SGD implies that

$$\begin{aligned}
\delta_{t+1}^w &= \|\mathbf{w}_{t,S} - \mathbf{w}_{t,\bar{S}} + \eta_t(\nabla_{\mathbf{w}} f(\mathbf{w}_{t,S}; \mathbf{x}_1) - \nabla_{\mathbf{w}} f(\mathbf{w}_{t,\bar{S}}; \mathbf{x}_1') \\
&\qquad + \nabla_{\mathbf{w}} f(\mathbf{w}_{t,\bar{S}}; g(\mathbf{v}_{t,\bar{S}}; Z_{\zeta_t})) - \nabla_{\mathbf{w}} f(\mathbf{w}_{t,S}; g(\mathbf{v}_{t,S}; Z_{\zeta_t})))\| \\
&\leq \delta_t^w + 2\eta_t \sigma_f^w + \eta_t L_f^x \sigma_g \delta_t^v.
\end{aligned}$$

Case 3: $\xi_t \neq 1, \zeta_t = 1$**.**

This case occurs with probability $\frac{n-1}{n}\frac{1}{m}$. The update rule of SGD implies that

$$\begin{aligned}
\delta_{t+1}^w &= \|\mathbf{w}_{t,S} - \mathbf{w}_{t,\bar{S}} + \eta_t(\nabla_{\mathbf{w}} f(\mathbf{w}_{t,S}; \mathbf{x}_{\xi_t}) - \nabla_{\mathbf{w}} f(\mathbf{w}_{t,\bar{S}}; \mathbf{x}_{\xi_t}) \\
&\qquad + \nabla_{\mathbf{w}} f(\mathbf{w}_{t,\bar{S}}; g(\mathbf{v}_{t,\bar{S}}; Z_1)) - \nabla_{\mathbf{w}} f(\mathbf{w}_{t,S}; g(\mathbf{v}_{t,S}; Z_1')))\| \\
&\leq \delta_t^w + \eta_t L_f^w \delta_t^w + 2\eta_t \sigma_f^w \\
&= (1 + \eta_t L_f^w)\delta_t^w + 2\eta_t \sigma_f^w.
\end{aligned}$$

Case 4: $\xi_t = 1, \zeta_t = 1$**.**

This case occurs with probability $\frac{1}{n}\frac{1}{m}$. The update rule of SGD implies that

$$\begin{aligned}
\delta_{t+1}^w &= \|\mathbf{w}_{t,S} - \mathbf{w}_{t,\bar{S}} + \eta_t(\nabla_{\mathbf{w}} f(\mathbf{w}_{t,S}; \mathbf{x}_1) - \nabla_{\mathbf{w}} f(\mathbf{w}_{t,\bar{S}}; \mathbf{x}_1') \\
&\qquad + \nabla_{\mathbf{w}} f(\mathbf{w}_{t,\bar{S}}; g(\mathbf{v}_{t,\bar{S}}; Z_1)) - \nabla_{\mathbf{w}} f(\mathbf{w}_{t,S}; g(\mathbf{v}_{t,S}; Z_1')))\| \\
&\leq \delta_t^w + 4\eta_t \sigma_f^w.
\end{aligned}$$

Based on the results of the above four cases, we take the expectation over the randomness of the SGD and obtain that

$$\begin{aligned}
\mathbb{E}_A\delta_{t+1}^w &\leq \frac{(n-1)(m-1)}{nm}\big[(1 + 2\eta_t L_f^w)\mathbb{E}_A\delta_t^w + \eta_t L_f^x \sigma_g \mathbb{E}_A\delta_t^v\big] \\
&\qquad + \frac{(m-1)}{nm}\big[\mathbb{E}_A\delta_t^w + 2\eta_t\sigma_f^w + \eta_t L_f^x \sigma_g \mathbb{E}_A\delta_t^v\big] \\
&\qquad + \frac{(n-1)}{nm}\big[(1 + \eta_t L_f^w)\mathbb{E}_A\delta_t^w + 2\eta_t\sigma_f^w\big] \\
&\qquad + \frac{1}{nm}\big[\mathbb{E}_A\delta_t^w + 4\eta_t\sigma_f^w\big] \\
&\leq \big(1 + 2\eta_t L_f^w\big)\mathbb{E}_A\delta_t^w + \eta_t L_f^x \sigma_g \mathbb{E}_A\delta_t^v \\
&\qquad + \frac{2(m+n)\eta_t\sigma_f^w}{mn}. \qquad (29)
\end{aligned}$$

Next, we bound δ_{t+1}^v. At iteration t, we consider the following two cases.

Case 1: $\zeta_t \neq 1$**.**

This case occurs with probability $\frac{m-1}{m}$. Also, the update rule of SGD implies that

$$\begin{aligned}
\delta_{t+1}^v &= \|\mathbf{v}_{t,S} - \mathbf{v}_{t,\bar{S}} + \eta_t(\nabla_{\mathbf{v}} g(\mathbf{v}_{t,S}; Z_{\zeta_t})\nabla_g f(\mathbf{w}_{t,S}; g(\mathbf{v}_{t,S}; Z_{\zeta_t})) \\
&\qquad - \nabla_{\mathbf{v}} g(\mathbf{v}_{t,\bar{S}}; Z_{\zeta_t})\nabla_g f(\mathbf{w}_{t,\bar{S}}; g(\mathbf{v}_{t,\bar{S}}; Z_{\zeta_t})))\| \\
&\leq \delta_t^v + \eta_t\sigma_g(L_f^w\delta_t^w + L_f^x\sigma_g\delta_t^v) + \eta_t\sigma_f^x L_g\delta_t^v \\
&= (1 + \eta_t L_f^x\sigma_g^2 + \eta_t\sigma_f^x L_g)\delta_t^v + \eta_t L_f^w\sigma_g\delta_t^w.
\end{aligned}$$

Case 2: $\zeta_t = 1$**.**

This case occurs with probability $\frac{1}{m}$. Also, the update rule of SGD implies that

$$\begin{aligned}
\delta_{t+1}^v &= \|\mathbf{v}_{t,S} - \mathbf{v}_{t,\bar{S}} + \eta_t(\nabla_{\mathbf{v}} g(\mathbf{v}_{t,S}; Z_1)\nabla_g f(\mathbf{w}_{t,S}; g(\mathbf{v}_{t,S}; Z_1)) \\
&\qquad - \nabla_{\mathbf{v}} g(\mathbf{v}_{t,\bar{S}}; Z_1')\nabla_g f(\mathbf{w}_{t,\bar{S}}; g(\mathbf{v}_{t,\bar{S}}; Z_1')))\| \\
&\leq \delta_t^v + 2\eta_t\sigma_f^x\sigma_g.
\end{aligned}$$

Based on the results of the above two cases, we take the expectation over the randomness of SGD and obtain that

$$\begin{aligned}\mathbb{E}_A\delta_{t+1}^v &\le \frac{m-1}{m}\big[(1+\eta_t L_f^x\sigma_g^2+\eta_t\sigma_f^x L_g)\delta_t^v+\eta_t L_f^w\sigma_g\delta_t^w\big]\\ &\quad+\frac{1}{m}\big[\delta_t^v+2\eta_t\sigma_f^x\sigma_g\big]\\ &\le (1+\eta_t L_f^x\sigma_g^2+\eta_t\sigma_f^x L_g)\mathbb{E}_A\delta_t^v+\eta_t L_f^w\sigma_g\mathbb{E}_A\delta_t^w\\ &\quad+\frac{2}{m}\eta_t\sigma_f^x\sigma_g. \end{aligned}\tag{30}$$

Combining eq. (29) and eq. (30) yields that

$$\begin{aligned}\mathbb{E}_A\begin{bmatrix}\delta_{t+1}^w\\ \delta_{t+1}^v\end{bmatrix} &\le \begin{bmatrix}1+2\eta_t L_f^w & \eta_t L_f^x\sigma_g\\ \eta_t L_f^w\sigma_g & 1+\eta_t(\sigma_f^x L_g+\sigma_g^2 L_f^x)\end{bmatrix}\mathbb{E}_A\begin{bmatrix}\delta_t^w\\ \delta_t^v\end{bmatrix}\\ &\quad+2\eta_t\begin{bmatrix}\frac{m+n}{mn}\sigma_f^w\\ \frac{\sigma_g\sigma_f^x}{m}\end{bmatrix}.\end{aligned}$$

Then, the proof is complete. □

APPENDIX F
PROOF OF THEOREM 3

Theorem 3. *Let Assumption 1 hold. Apply SGM to solve the ERM in eq.* (14) *with the dataset S and denote the corresponding output at T-th iteration as $(\mathbf{w}_{T,S},\mathbf{v}_{T,S})$. Choose the stepsize $\eta_t=\frac{c}{t\log t}$ with $c\le(2L_f^w+\sigma_f^x L_g+\sigma_g^2 L_f^x)^{-1}$. Then, the generalization error of SGM satisfies*

$$\begin{aligned}&\mathbb{E}_S\mathbb{E}_{sgm}\big[L(\mathbf{w}_{T,S},\mathbf{v}_{T,S})-L_S(\mathbf{w}_{T,S},\mathbf{v}_{T,S})\big]\\ &\qquad\le 2\sqrt{(2\sigma_f^x)^2+(\sigma_f^w\sigma_g)^2}\bigg(\frac{\sigma_f^w+\sigma_g\sigma_f^x}{m}+\frac{\sigma_f^w}{n}\bigg)\log T.\end{aligned}$$

Proof. Define the following two quantities

$$\boldsymbol{U}_t=\begin{bmatrix}1+2\eta_t L_f^w & \eta_t L_f^x\sigma_g\\ \eta_t L_f^w\sigma_g & 1+\eta_t(\sigma_f^x L_g+\sigma_g^2 L_f^x)\end{bmatrix},\mathbf{b}=\begin{bmatrix}\frac{m+n}{mn}\sigma_f^w\\ \frac{\sigma_g\sigma_f^x}{m}\end{bmatrix}.$$

Then, the recursion property in Proposition 2 can be rewritten as

$$\mathbb{E}_A\begin{bmatrix}\delta_{t+1}^w\\ \delta_{t+1}^v\end{bmatrix}\le\boldsymbol{U}_t\mathbb{E}_A\begin{bmatrix}\delta_t^w\\ \delta_t^v\end{bmatrix}+2\eta_t\mathbf{b}.\tag{31}$$

Applying eq. (31) recursively and noting that $\delta_0^w=\delta_0^v=0$, we obtain that

$$\mathbb{E}_A\begin{bmatrix}\delta_{t+1}^w\\ \delta_{t+1}^v\end{bmatrix}\le\sum_{l=0}^{t}\bigg(\prod_{k=l+1}^{t}\boldsymbol{U}_k\bigg)2\eta_l\mathbf{b}.$$

By Lemma 1 and the above inequality, we further obtain that

$$\begin{aligned}&\mathbb{E}_S\mathbb{E}_A\big[L(\mathbf{w}_{t,S},\mathbf{v}_{t,S})-L_S(\mathbf{w}_{t,S},\mathbf{v}_{t,S})\big]\\ &\le\sqrt{(2\sigma_f^x)^2+(\sigma_f^w\sigma_g)^2}\sup_{S,\bar S}\bigg\|\mathbb{E}_A\begin{bmatrix}\delta_t^w\\ \delta_t^v\end{bmatrix}\bigg\|\\ &\le\sqrt{(2\sigma_f^x)^2+(\sigma_f^w\sigma_g)^2}\bigg\|\sum_{l=0}^{t-1}\bigg(\prod_{k=l+1}^{t-1}\boldsymbol{U}_k\bigg)2\eta_l\mathbf{b}\bigg\|\\ &\le\sqrt{(2\sigma_f^x)^2+(\sigma_f^w\sigma_g)^2}\sum_{l=0}^{t-1}\bigg\|\bigg(\prod_{k=l+1}^{t-1}\boldsymbol{U}_k\bigg)2\eta_l\mathbf{b}\bigg\|\\ &\le 2\sqrt{(2\sigma_f^x)^2+(\sigma_f^w\sigma_g)^2}\|\mathbf{b}\|\sum_{l=0}^{t-1}\eta_l\bigg\|\prod_{k=l+1}^{t-1}\boldsymbol{U}_k\bigg\|\\ &\le 2\sqrt{(2\sigma_f^x)^2+(\sigma_f^w\sigma_g)^2}\|\mathbf{b}\|\sum_{l=0}^{t-1}\eta_l\prod_{k=l+1}^{t-1}\|\boldsymbol{U}_k\|.\end{aligned}\tag{32}$$

Next, we evaluate the operator norm of $\boldsymbol{U}_k$, i.e., the quantity $\sqrt{\lambda_{\max}(\boldsymbol{U}_k^\top\boldsymbol{U}_k)}$. Note that $\boldsymbol{U}_k^\top\boldsymbol{U}_k$ is a 2×2 matrix. We calculate its eigenvalue and obtain that

$$\|\boldsymbol{U}_k\|=\sqrt{\lambda_{\max}(\boldsymbol{U}_k^\top\boldsymbol{U}_k)}=\sqrt{\frac{\gamma}{2}+\sqrt{\frac{\gamma^2}{4}-\beta}},$$

where

$$\begin{aligned}\gamma&=(1+2\eta_t L_f^w)^2+(\eta_t L_f^w\sigma_g)^2\\ &\quad+(\eta_t L_f^x\sigma_g)^2+(1+\eta_t(\sigma_f^x L_g+\sigma_g^2 L_f^x))^2,\\ \beta&=\big((1+2\eta_t L_f^w)(1+\eta_t(\sigma_f^x L_g+\sigma_g^2 L_f^x))\\ &\quad-(\eta_t L_f^w\sigma_g)(\eta_t L_f^x\sigma_g)\big)^2.\end{aligned}$$

Note that the stepsize η_t for SGD is chosen to decrease to zero. Thus, we can ignore the higher order terms that contain η_t^2 and obtain that

$$\|\boldsymbol{U}_k\|\le\sqrt{\frac{\gamma}{2}+\sqrt{\frac{\gamma^2}{4}-\beta}}\lesssim 1+\eta_k(2L_f^w+\sigma_f^x L_g+\sigma_g^2 L_f^x).$$

Let $\lambda=2\sqrt{(2\sigma_f^x)^2+(\sigma_f^w\sigma_g)^2}$. Substituting the above bound into eq. (32) and noting that $\eta_t=\frac{c}{t\log t}$, we obtain that

$$\begin{aligned}&\mathbb{E}_S\mathbb{E}_A\big[L(\mathbf{w}_{t,S},\mathbf{v}_{t,S})-L_S(\mathbf{w}_{t,S},\mathbf{v}_{t,S})\big]\\ &\le\lambda\|\mathbf{b}\|\sum_{l=0}^{t-1}\eta_l\prod_{k=l+1}^{t-1}\big(1+\eta_k(2L_f^w+\sigma_f^x L_g+\sigma_g^2 L_f^x)\big)\\ &\le\lambda\|\mathbf{b}\|\sum_{l=0}^{t-1}\frac{c}{l\log l}\prod_{k=l+1}^{t-1}\exp\big(\frac{c}{k\log k}(2L_f^w+\sigma_f^x L_g+\sigma_g^2 L_f^x)\big)\\ &\le\lambda\|\mathbf{b}\|\sum_{l=0}^{t-1}\frac{c}{l\log l}\exp\big(c(2L_f^w+\sigma_f^x L_g+\sigma_g^2 L_f^x)\log\frac{\log t}{\log l}\big)\\ &\le\lambda\|\mathbf{b}\|\sum_{l=0}^{t-1}\frac{c}{l\log l}\bigg(\frac{\log t}{\log l}\bigg)^{c(2L_f^w+\sigma_f^x L_g+\sigma_g^2 L_f^x)}\\ &\le\lambda\|\mathbf{b}\|(\log t)^{c(2L_f^w+\sigma_f^x L_g+\sigma_g^2 L_f^x)}\\ &\le\lambda\bigg(\frac{\sigma_f^w+\sigma_g\sigma_f^x}{m}+\frac{\sigma_f^w}{n}\bigg)\log t.\end{aligned}$$

Then, the proof is complete. □

ACKNOWLEDGMENT

The authors would like to thank the associate editor Dr. Kamalika Chaudhuri for her constructive suggestions to improve the presentation of this paper. They would also like to thank the anonymous reviewers for their insightful and helpful comments.

REFERENCES

[1] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in Advances in neural information processing systems (NIPS), 2014, pp. 2672–2680.

[2] S. Arora, R. Ge, Y. Liang, T. Ma, and Y. Zhang, "Generalization and equilibrium in generative adversarial nets (GANs)," in Proc. 34rd International Conference on Machine Learning (ICML), 2017.

[3] P. Zhang, Q. Liu, D. Zhou, T. Xu, and X. He, "On the discrimination-generalization tradeoff in GANs," in Proc. International Conference on Learning Representations (ICLR), 2018.

[4] T. Liang, "How well can generative adversarial networks (GAN) learn densities: A nonparametric view," arXiv preprint arXiv:1712.08244, 2017.

[5] M. Hardt, B. Recht, and Y. Singer, "Train faster, generalize better: Stability of stochastic gradient descent," in Proc. 33rd International Conference on Machine Learning (ICML), 2016, pp. 1225–1234.

[6] V. Nagarajan and J. Z. Kolter, "Gradient descent GAN optimization is locally stable," in Proc. Advances in Neural Information Processing Systems (NIPS), 2017, pp. 5585–5595.

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

[8] G. K. Dziugaite, D. M. Roy, and Z. Ghahramani, "Training generative neural networks via maximum mean discrepancy optimization," in Proc. of the 31st Conference on Uncertainty in Artificial Intelligence (UAI), 2015.

[9] M. Imaizumi and K. Fukumizu, "Understanding gans via generalization analysis for disconnected support," in International Conference on Learning Representations (ICLR), 2019.

[10] H. Thanh-Tung, T. Tran, and S. Venkatesh, "Improving generalization and stability of generative adversarial networks," in International Conference on Learning Representations (ICLR), 2019.

[11] S. Liu, O. Bousquet, and K. Chaudhuri, "Approximation and convergence properties of generative adversarial learning," in Proc. Advances in Neural Information Processing Systems (NIPS), 2017, pp. 5551–5559.

[12] S. Singh, A. Uppal, B. Li, C.-L. Li, M. Zaheer, and B. Póczos, "Nonparametric density estimation under adversarial losses," in Advances in Neural Information Processing Systems (NeurIPS), 2018, pp. 10 225–10 236.

[13] T. Liang, "On how well generative adversarial networks learn densities: Nonparametric and parametric results," arXiv preprint arXiv:1811.03179, 2018.

[14] A. Uppal, S. Singh, and B. Póczos, "Nonparametric density estimation & convergence rates for gans under Besov IPM losses," in Advances in Neural Information Processing Systems (NeurIPS), 2019, pp. 9089–9100.

[15] H. Gouk, E. Frank, B. Pfahringer, and M. J. Cree, "Regularisation of neural networks by enforcing lipschitz continuity," Machine Learning, pp. 1–24, 2020.

[16] A. M. Oberman and J. Calder, "Lipschitz regularized deep neural networks converge and generalize," arXiv preprint arXiv:1808.09540, 2018.

[17] C. Wei and T. Ma, "Data-dependent sample complexity of deep neural networks via lipschitz augmentation," in Advances in Neural Information Processing Systems (NeurIPS), 2019, pp. 9725–9736.

[18] T. Xu, Y. Zhou, K. Ji, and Y. Liang, "When will gradient methods converge to max-margin classifier under ReLU models?" arXiv preprint arXiv:1806.04339, 2018.

[19] M. Anthony and P. L. Bartlett, Neural network learning: Theoretical foundations. Cambridge University Press, 2009.

[20] P. L. Bartlett, D. J. Foster, and M. J. Telgarsky, "Spectrally-normalized margin bounds for neural networks," in Advances in Neural Information Processing Systems (NeurIPS), 2017, pp. 6240–6249.

[21] N. Golowich, A. Rakhlin, and O. Shamir, "Size-independent sample complexity of neural networks," in Conference On Learning Theory (COLT), 2018, pp. 297–299.

[22] B. Neyshabur, R. Tomioka, and N. Srebro, "Norm-based capacity control in neural networks," in Proc. Conference on Learning Theory (COLT), 2015, pp. 1376–1401.

[23] S. Oymak, "Learning compact neural networks with regularization," in International Conference on Machine Learning (ICML), 2018, pp. 3966–3975.

[24] K. Ji and Y. Liang, "Minimax estimation of neural net distance," in Advances in Neural Information Processing Systems (NeurIPS), 2018, pp. 3845–3854.

[25] O. Bousquet and A. Elisseeff, "Stability and generalization," Journal of Machine Learning Research, vol. 2, pp. 499–526, Mar. 2002.

[26] A. Elisseeff, T. Evgeniou, and M. Pontil, "Stability of randomized learning algorithms," Journal of Machine Learning Research, vol. 6, pp. 55–79, Dec. 2005.

[27] S. Shalev-Shwartz, O. Shamir, N. Srebro, and K. Sridharan, "Learnability, stability and uniform convergence," Journal of Machine Learning Research, vol. 11, pp. 2635–2670, Dec. 2010.

[28] I. Kuzborskij and C. Lampert, "Data-dependent stability of stochastic gradient descent," in International Conference on Machine Learning (ICML). PMLR, 2018, pp. 2815–2824.

[29] W. Mou, L. Wang, X. Zhai, and K. Zheng, "Generalization bounds of SGLD for non-convex learning: Two theoretical viewpoints," in Conference on Learning Theory (COLT). PMLR, 2018, pp. 605–638.

[30] D. Yin, A. Pananjady, M. Lam, D. Papailiopoulos, K. Ramchandran, and P. Bartlett, "Gradient diversity: a key ingredient for scalable distributed learning," in International Conference on Artificial Intelligence and Statistics (AISTATS). PMLR, 2018, pp. 1998–2007.

[31] Y. Zhou, Y. Liang, and H. Zhang, "Generalization error bounds with probabilistic guarantee for SGD in nonconvex optimization," arXiv:1802.06903v1, 2018.

[32] Z. Charles and D. Papailiopoulos, "Stability and generalization of learning algorithms that converge to global optima," in International Conference on Machine Learning (ICML). PMLR, 2018, pp. 745–754.

[33] K. Ji, Z. Wang, Y. Zhou, and Y. Liang, "Improved zeroth-order variance reduced algorithms and analysis for nonconvex optimization," in International Conference on Machine Learning (ICML), 2019, pp. 3100–3109.

[34] T. Poggio, S. Voinea, and R. L., "Online learning, stability, and stochastic gradient descent," ArXiv: 1105.4701v3, 2011.

[35] V. Feldman and J. Vondrak, "Generalization bounds for uniformly stable algorithms," in Proc. Advances in Neural Information Processing Systems (NeurIPS), 2018, pp. 9747–9757.

[36] ——, "High probability generalization bounds for uniformly stable algorithms with nearly optimal rate," in Proc. Conference on Learning Theory (COLT), vol. 99, 25–28 Jun 2019, pp. 1270–1279.

[37] Y. Lei and K. Tang, "Stochastic composite mirror descent: Optimal bounds with high probabilities," in Proc. Advances in Neural Information Processing Systems (NeurIPS), 2018, pp. 1519–1529.

[38] Z. Yuan, Y. Yan, R. Jin, and T. Yang, "Stagewise training accelerates convergence of testing error over SGD," in Advances in Neural Information Processing Systems (NeurIPS), 2019, pp. 2608–2618.

[39] J. Li, X. Luo, and M. Qiao, "On generalization error bounds of noisy gradient methods for non-convex learning," in International Conference on Learning Representations (ICLR), 2020.

[40] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), 2016, pp. 2536–2544.

[41] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), 2017, pp. 1125–1134.

[42] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in Proceedings of the IEEE international conference on computer vision (ICCV), 2017, pp. 2223–2232.

[43] T. Salimans and D. P. Kingma, "Weight normalization: A simple reparameterization to accelerate training of deep neural networks," in Proc. Advances in Neural Information Processing Systems (NIPS), 2016, pp. 901–909.

[44] Q. Xu, G. Huang, Y. Yuan, C. Guo, Y. Sun, F. Wu, and K. Weinberger, "An empirical study on evaluation metrics of generative adversarial networks," arXiv preprint arXiv:1806.07755, 2018.

[45] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," in Advances in neural information processing systems (NeurIPS), 2017, pp. 6626–6637.

[46] T. Che, Y. Li, A. P. Jacob, Y. Bengio, and W. Li, "Mode regularized generative adversarial networks," in International Conference on Learning Representations (ICLR), 2017.
[47] A. B. Tsybakov, Introduction to nonparametric estimation. Springer, New York, 2008.

Kaiyi Ji Kaiyi Ji is currently a fifth-year PhD student at the Department of Electrical and Computer Engineering, The Ohio State University (OSU). He was a visiting student at the Department of Electrical Engineering, Princeton University for a short term in 2020. He obtained his B.E. degree from University of Science and Technology of China in 2016. His research interest lies in statistical machine learning, large-scale optimization, reinforcement learning, and data-driven design for networks and database systems. He received the University Fellowship at OSU in 2016 and the Presidential Fellowship at OSU in 2020.

Yi Zhou Yi Zhou is currently an Assistant Professor affiliated with the Department of Electrical and Computer Engineering at the University of Utah. Before, he was a postdoc fellow affiliated with the Department of Electrical and Computer Engineering at Duke University. He received the Ph.D. degree in Electrical and Computer Engineering from The Ohio State University in 2018. Dr. Zhou's research interests include machine learning, reinforcement learning, deep learning, nonconvex optimization and distributed optimization. His paper received an invitation for spotlight presentation in NeurIPS 2018. He served as a program committee member in the Workshop on Theoretical Foundations and Applications of Deep Generative Models in ICML 2018.

Yingbin Liang Dr. Yingbin Liang (S'0-M'05-SM'16) is currently a Professor at the Department of Electrical and Computer Engineering at the Ohio State University (OSU). She received the Ph.D. degree in Electrical Engineering from the University of Illinois at Urbana-Champaign in 2005, and served on the faculty of University of Hawaii and Syracuse University before she joined OSU. Dr. Liang's research interests include machine learning, optimization, information theory and statistical signal processing. Dr. Liang received the National Science Foundation CAREER Award and the State of Hawaii Governor Innovation Award in 2009. She also received EURASIP Best Paper Award for the EURASIP Journal on Wireless Communications and Networking in 2014. She served as an Associate Editor for the Shannon Theory of the IEEE Transactions on Information Theory during 2013-2015.