# SGD Learns One-Layer Networks in WGANs

**Qi Lei** [1]  **Jason D. Lee** [2]  **Alexandros G. Dimakis** [1]  **Constantinos Daskalakis** [3]

## Abstract

Generative adversarial networks (GANs) are a widely used framework for learning generative models. Wasserstein GANs (WGANs), one of the most successful variants of GANs, require solving a minmax optimization problem to global optimality, but are in practice successfully trained using stochastic gradient descent-ascent. In this paper, we show that, when the generator is a one-layer network, stochastic gradient descent-ascent converges to a global solution with polynomial time and sample complexity.

## 1. Introduction

Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) are a prominent framework for learning generative models of complex, real-world distributions given samples from these distributions. GANs and their variants have been successfully applied to numerous datasets and tasks, including image-to-image translation (Isola et al., 2017), image super-resolution (Ledig et al., 2017), domain adaptation (Tzeng et al., 2017), probabilistic inference (Dumoulin et al., 2016), compressed sensing (Bora et al., 2017) and many more. These advances owe in part to the success of Wasserstein GANs (WGANs) (Arjovsky et al., 2017; Gulrajani et al., 2017), leveraging the neural net induced integral probability metric to better measure the difference between a target and a generated distribution.

Along with the aforementioned empirical successes, there have been theoretical studies of the statistical properties of GANs—see e.g. (Zhang et al., 2018; Arora et al., 2017; 2018; Bai et al., 2018; Dumoulin et al., 2016) and their references. These works have shown that, with an appropriate design of the generator and discriminator, the global optimum of the WGAN objective identifies the target distri-

[1]University of Texas at Austin, TX [2]Princeton University, NJ [3]Massachusetts Institute of Technology, MA. Correspondence to: Qi Lei <leiqi@ices.utexas.edu>, Jason D. Lee <jasonlee@princeton.edu>.

bution with low sample complexity. However, these results cannot be algorithmically attained via practical GAN training algorithms.

On the algorithmic front, prior work has focused on the stability and convergence properties of gradient descent-ascent (GDA) and its variants in GAN training and more general min-max optimization problems; see e.g. (Nagarajan & Kolter, 2017; Heusel et al., 2017; Mescheder et al., 2017; 2018; Daskalakis et al., 2017; Daskalakis & Panageas, 2018a;b; Gidel et al., 2019; Liang & Stokes, 2019; Mokhtari et al., 2019; Jin et al., 2019; Lin et al., 2019; Lei et al., 2020; 2019; 2017) and their references. These works have studied conditions under which GDA converges to a globally optimal solution in the convex-concave objective, or local stability in the non-convex non-concave setting. These results do not ensure convergence to a globally optimal generator, or in fact even convergence to a locally optimal generator.

Thus a natural question is whether:

> *Are GANs able to learn high-dimensional distributions in polynomial time and polynomial/parametric sample complexity, and thus bypass the curse of dimensionality?*

The aforementioned prior works stop short of this goal due to a) the intractability of min-max optimization in the non-convex setting, and b) the curse of dimensionality in learning with Wasserstein distance in high dimensions (Bai et al., 2018).

A notable exception is Feizi et al. (2017) which shows that for WGANs with a linear generator and quadratic discriminator GDA succeeds in learning a Gaussian using polynomially many samples in the dimension.

In the same vein, we are the first to our knowledge to study the global convergence properties of stochastic GDA in the GAN setting, and establishing such guarantees for non-linear generators. In particular, we study the WGAN formulation for learning a single-layer generative model with some reasonable choices of activations including tanh, sigmoid and leaky ReLU.

**Our contributions.** For WGAN with a one-layer generator network using an activation from a large family of functions and a quadratic discriminator, we show that stochastic gradient descent-ascent learns a target distribution using

polynomial time and samples, under the assumption that the target distribution is realizable in the architecture of the generator. This is achieved by *simultaneously* satisfying the following two criterion:

1. Proving that stochastic gradient-descent attains a globally optimal generator in the metric induced by the discriminator,

2. Proving that appropriate design of the discriminator ensures a parametric $\mathcal{O}(\frac{1}{\sqrt{n}})$ statistical rate (Zhang et al., 2018; Bai et al., 2018) that matches the lower bound for learning one-layer generators as shown in (Wu et al., 2019).

## 2. Related Work

We briefly review relevant results in GAN training and learning generative models:

### 2.1. Optimization viewpoint

For standard GANs and WGANs with appropriate regularization, (Nagarajan & Kolter, 2017), (Mescheder et al., 2017) and (Heusel et al., 2017) establish sufficient conditions to achieve local convergence and stability properties for GAN training. At the equilibrium point, if the Jacobian of the associated gradient vector field has only eigenvalues with negative real-part, GAN training is verified to converge locally for small enough learning rates. A follow-up paper by (Mescheder et al., 2018) shows the necessity of these conditions by identifying a counterexample that fails to converge locally for gradient descent based GAN optimization. The lack of global convergence prevents the analysis from yielding any guarantees for learning the real distribution.

The work of (Feizi et al., 2017) described above has similar goals as our paper, namely understanding the convergence properties of basic dynamics in simple WGAN formulations. However, they only consider linear generators, which restrict the WGAN model to learning a Gaussian. Our work goes a step further, considering WGANs whose generators are one-layer neural networks with a broad selection of activations. We show that with a proper gradient-based algorithm, we can still recover the ground truth parameters of the underlying distribution.

More broadly, WGANs typically result in nonconvex-nonconcave min-max optimization problems. In these problems, a global min-max solution may not exist, and there are various notions of local min-max solutions, namely local min-local max solutions (Daskalakis & Panageas, 2018b), and local min solutions of the max objective (Jin et al., 2019), the latter being guaranteed to exist under mild conditions. In fact, (Lin et al., 2019) show that GDA is able to find stationary points of the max objective in nonconvex-

concave objectives. Given that GDA may not even converge for convex-concave objectives, another line of work has studied variants of GDA that exhibit global convergence to the min-max solution (Daskalakis et al., 2017; Daskalakis & Panageas, 2018a; Gidel et al., 2019; Liang & Stokes, 2019; Mokhtari et al., 2019), which is established for GDA variants that add negative momentum to the dynamics. While the convergence of GDA with negative momentum is shown in convex-concave settings, there is experimental evidence supporting that it improves GAN training (Daskalakis et al., 2017; Gidel et al., 2019).

### 2.2. Statistical viewpoint

Several works have studied the issue of mode collapse. One might doubt the ability of GANs to actually learn the distribution vs just memorize the training data (Arora et al., 2017; 2018; Dumoulin et al., 2016). Some corresponding cures have been proposed. For instance, (Zhang et al., 2018; Bai et al., 2018) show for specific generators combined with appropriate parametric discriminator design, WGANs can attain parametric statistical rates, avoiding the exponential in dimension sample complexity (Liang, 2018; Bai et al., 2018; Feizi et al., 2017).

Recent work of (Wu et al., 2019) provides an algorithm to learn the distribution of a single-layer ReLU generator network. While our conclusion appears similar, our focus is very different. Our paper targets understanding when a WGAN formulation trained with stochastic GDA can learn in polynomial time and sample complexity. Their work instead relies on a specifically tailored algorithm for learning truncated normal distributions (Daskalakis et al., 2018).

## 3. Preliminaries

**Notation.** We consider GAN formulations for learning a generator $G_A : \mathbb{R}^k \to \mathbb{R}^d$ of the form $\boldsymbol{z} \mapsto \boldsymbol{x} = \phi(A\boldsymbol{z})$, where $A$ is a $d \times k$ parameter matrix and $\phi$ some activation function. We consider discriminators $D_{\boldsymbol{v}} : \mathbb{R}^d \to \mathbb{R}$ or $D_V : \mathbb{R}^d \to \mathbb{R}$ respectively when the discriminator functions are parametrized by either vectors or matrices. We assume latent variables $\boldsymbol{z}$ are sampled from the normal $\mathcal{N}(0, I_{k \times k})$, where $I_{k \times k}$ denotes the identity matrix of size $k$. The real/target distribution outputs samples $\boldsymbol{x} \sim \mathcal{D} = G_{A^*}(\mathcal{N}(0, I_{k_0 \times k_0}))$, for some ground truth parameters $A^*$, where $A^*$ is $d \times k_0$, and we take $k \geq k_0$ for enough expressivity, taking $k = d$ when $k_0$ is unknown.

The Wasserstain GAN under our choice of generator and discriminator is naturally formulated as:

$$\min_{A \in \mathbb{R}^{d \times k}} \max_{\boldsymbol{v} \in \mathbb{R}^d} f(A, \boldsymbol{v}),\,^{[1]}$$

---

[1]We will replace $\boldsymbol{v}$ by matrix parameters $V \in \mathbb{R}^{d \times d}$ when

for $f(A, \boldsymbol{v}) \equiv \mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}} D_{\boldsymbol{v}}(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{z} \sim \mathcal{N}(0, I_{k \times k})} D_{\boldsymbol{v}}(G_A(\boldsymbol{z}))$.

We use $\boldsymbol{a}_i$ to denote the $i$-th row vector of $A$. We sometimes omit the 2 subscript, using $\|\boldsymbol{x}\|$ to denote the 2-norm of vector $\boldsymbol{x}$, and $\|X\|$ to denote the spectral norm of matrix $X$ when there is no ambiguity. $\mathbb{S}^n \subset \mathbb{R}^{n \times n}$ represents all the symmetric matrices of dimension $n \times n$. We use $Df(X_0)[B]$ to denote the directional derivative of function $f$ at point $X_0$ with direction $B$: $Df(X_0)[B] = \lim_{t \to 0} \frac{f(X_0 + tB) - f(X_0)}{t}$.

### 3.1. Motivation and Discussion

To provably learn one-layer generators with nonlinear activations, the design of the discriminator must strike a delicate balance:

1. (Approximation.) The discriminator should be large enough to be able to distinguish the true distribution from incorrect generated ones. To be more specific, the max function $g(A) = \max_{\boldsymbol{x}} f(A, V)$ captures some distance from our learned generator to the target generators. This distance should only have global minima that correspond to the ground truth distribution.

2. (Generalizability.) The discriminator should be small enough so that it can be learned with few samples. In fact, our method guarantees an $\mathcal{O}(1/\sqrt{n})$ parametric rate that matches the lower bound established in (Wu et al., 2019).

3. (Stability.) The discriminator should be carefully designed so that simple local algorithms such as gradient descent ascent can find the global optimal point.

Further, min-max optimization with non-convexity in either side is intractable. In fact, gradient descent ascent does not even yield last iterate convergence for bilinear forms, and it requires more carefully designed algorithms like Optimistic Gradient Descent Ascent (Daskalakis & Panageas, 2018b) and Extra-gradient methods (Korpelevich, 1976). In this paper we show a stronger hardness result. We show that for simple bilinear forms with ReLU activations, it is NP-hard to even find a stationary point.

**Theorem 1.** *Consider the min-max optimization on the following ReLU-bilinear form:*

$$\min_{\boldsymbol{x}} \max_{\boldsymbol{y}} \left\{ f(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i=1}^{n} \phi(A_i \boldsymbol{x} + \boldsymbol{b}_i)^\top \boldsymbol{y} \right\},$$

*where $\boldsymbol{x} \in \mathbb{R}^d$, $A_i \in \mathbb{R}^{\mathcal{O}(d) \times d}$ and $\phi$ is ReLU activation. As long as $n \geq 4$, the problem of checking whether $f$ has any stationary point is NP-hard in $d$.*

necessary.

We defer the proof to the Appendix where we show 3SAT is reducible to the above problem. This theorem shows that in general, adding non-linearity (non-convexity) in min-max forms makes the problem intractable. However, we are able to show gradient descent ascent finds global minima for training one-layer generators with non-linearity. This will rely on carefully designed discriminators, regularization and specific structure that we considered.

Finally we note that understanding the process of learning one-layer generative model is important in practice as well. For instance, Progressive GAN (Karras et al., 2017) proposes the methodology to learn one-layer at a time, and grow both the generator and discriminator progressively during the learning process. Our analysis implies further theoretical support for this kind of progressive learning procedure.

## 4. Warm-up: Learning the Marginal Distributions

As a warm-up, we ask whether a simple linear discriminator is sufficient for the purposes of learning the marginal distributions of all coordinates of $\mathcal{D}$. Notice that in our setting, the $i$-th output of the generator is $\phi(x)$ where $x \sim \mathcal{N}(0, \|\boldsymbol{a}_i\|^2)$, and is thus solely determined by $\|\boldsymbol{a}_i\|_2$. With a linear discriminator $D_{\boldsymbol{v}}(\boldsymbol{x}) = \boldsymbol{v}^\top \boldsymbol{x}$, our minimax game becomes:

$$\min_{A \in \mathbb{R}^{d \times k}} \max_{\boldsymbol{v} \in \mathbb{R}^d} f_1(A, \boldsymbol{v}), \tag{1}$$

for $f_1(A, \boldsymbol{v}) \equiv \mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}} \left[ \boldsymbol{v}^\top \boldsymbol{x} \right] - \mathbb{E}_{\boldsymbol{z} \sim \mathcal{N}(0, I_{k \times k})} \left[ \boldsymbol{v}^\top \phi(A\boldsymbol{z}) \right]$.

Notice that when the activation $\phi$ is an odd function, such as the tanh activation, the symmetric property of the Gaussian distribution ensures that $\mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}}[\boldsymbol{v}^\top \boldsymbol{x}] = 0$, hence the linear discriminator in $f_1$ reveals no information about $A^*$. Therefore specifically for odd activations (or odd plus a constant activations), we instead use an adjusted rectified linear discriminator $D_{\boldsymbol{v}}(\boldsymbol{x}) \equiv \boldsymbol{v}^\top R(\boldsymbol{x} - C)$ to enforce some bias, where $C = \frac{1}{2}(\phi(x) + \phi(-x))$ for all $x$, and $R$ denotes the ReLU activation. Formally, we slightly modify our loss function as:

$$\bar{f}_1(A, \boldsymbol{v}) \equiv \mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}} \left[ \boldsymbol{v}^\top R(\boldsymbol{x} - C) \right] \\ - \mathbb{E}_{\boldsymbol{z} \sim \mathcal{N}(0, I_{k \times k})} \left[ \boldsymbol{v}^\top R(\phi(A\boldsymbol{z}) - C) \right]. \tag{2}$$

We will show that we can learn each marginal of $\mathcal{D}$ if the activation function $\phi$ satisfies the following.

**Assumption 1.** *The activation function $\phi$ satisfies either one of the following:*
*1. $\phi$ is an odd function plus constant, and $\phi$ is monotone increasing;*
*2. The even component of $\phi$, i.e. $\frac{1}{2}(\phi(x) + \phi(-x))$, is positive and monotone increasing on $x \in [0, \infty)$.*

**Remark 1.** *All common activation functions like (Leaky) ReLU, tanh or sigmoid function satisfy Assumption 1.*

**Lemma 1.** *Suppose $A^* \neq 0$. Consider $f_1$ with activation that satisfies Assumption 1.2 and $\bar{f}_1$ with activation that satisfies Assumption 1.1. The stationary points of such $f_1$ and $\bar{f}_1$ yield parameters $A$ satisfying $\|\boldsymbol{a}_i\| = \|\boldsymbol{a}_i^*\|, \forall i \in [d]$.*

To bound the capacity of the discriminator, WGAN adds an Lipschitz constraint: $\|D_{\boldsymbol{v}}\| \leq 1$, or simply $\|\boldsymbol{v}\|_2 \leq 1$. To make the training process easier, we instead regularize the discriminator. For the regularized formulation we have:

**Theorem 2.** *In the same setting as Lemma 1, alternating gradient descent-ascent with proper learning rates on*

$$\min_A \max_{\boldsymbol{v}} \{f_1(A, \boldsymbol{v}) - \|\boldsymbol{v}\|^2/2\},$$
$$\text{or respectively} \quad \min_A \max_{\boldsymbol{v}} \{\bar{f}_1(A, \boldsymbol{v}) - \|\boldsymbol{v}\|^2/2\},$$

*recovers $A$ such that $\|\boldsymbol{a}_i\| = \|\boldsymbol{a}_i^*\|, \forall i \in [d]$.*

All the proofs of the paper can be found in the appendix. We show that all local min-max points in the sense of (Jin et al., 2019) of the original problem are global min-max points and recover the correct norm of $\boldsymbol{a}_i^*, \forall i$. Notice for the source data distribution $\boldsymbol{x} = (x_1, x_2, \cdots x_d) \sim \mathcal{D}$ with activation $\phi$, the marginal distribution of each $x_i$ follows $\phi(\mathcal{N}(0, \|\boldsymbol{a}_i^*\|^2))$ and is determined by $\|\boldsymbol{a}_i^*\|$. Therefore we have learned the marginal distribution for each entry $i$. It remains to learn the joint distribution.

# 5. Learning the Joint Distribution

In the previous section, we utilize a (rectified) linear discriminator, such that each coordinate $v_i$ interacts with the $i$-th random variable. With the (rectified) linear discriminator, WGAN learns the correct $\|\boldsymbol{a}_i\|$, for all $i$. However, since there's no interaction between different coordinates of the random vector, we do not expect to learn the joint distribution with a linear discriminator.

To proceed, a natural idea is to use a quadratic discriminator $D_V(\boldsymbol{x}) := \boldsymbol{x}^\top V \boldsymbol{x} = \langle \boldsymbol{x}\boldsymbol{x}^\top, V \rangle$ to enforce component interactions. Similar to the previous section, we study the regularized version:

$$\min_{A \in \mathbb{R}^{d \times k}} \max_{V \in \mathbb{R}^{d \times d}} \{f_2(A, V) - \frac{1}{2}\|V\|_F^2\}, \quad (3)$$

where

$$f_2(A, V)$$
$$= \mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}} D_V(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{z} \sim \mathcal{N}(0, I_{k \times k})} D_V(\phi(A\boldsymbol{z}))$$
$$= \langle \mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}} \left[ \boldsymbol{x}\boldsymbol{x}^\top \right] - \mathbb{E}_{\boldsymbol{z} \sim \mathcal{N}(0, I_{k \times k})} \left[ \phi(A\boldsymbol{z})\phi(A\boldsymbol{z})^\top \right], V \rangle.$$

By adding a regularizer on $V$ and explicitly maximizing

over $V$:

$$g(A) \equiv \max_V \left\{ f_2(A, V) - \frac{1}{2}\|V\|_F^2 \right\}$$
$$= \frac{1}{2} \left\| \mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}} \left[ \boldsymbol{x}\boldsymbol{x}^\top \right] \right.$$
$$\left. - \mathbb{E}_{\boldsymbol{z} \sim \mathcal{N}(0, I_{k \times k})} \left[ \phi(A\boldsymbol{z})\phi(A\boldsymbol{z})^\top \right] \right\|_F^2.$$

In the next subsection, we first focus on analyzing the second-order stationary points of $g$, then we establish that gradient descent ascent converges to second-order stationary points of $g$.

## 5.1. Global Convergence for Optimizing the Generating Parameters

We first assume that both $A$ and $A^*$ have unit row vectors, and then extend to general case since we already know how to learn the row norms from Section 4. To explicitly compute $g(A)$, we rely on the property of Hermite polynomials. Since normalized Hermite polynomials $\{h_i\}_{i=0}^\infty$ forms an orthonomal basis in the functional space, we rewrite the activation function as $\phi(\boldsymbol{x}) = \sum_{i=0}^\infty \sigma_i h_i$, where $\sigma_i$ is the $i$-th Hermite coefficient. We use the following claim:

**Claim 1** ((Ge et al., 2017) Claim 4.2). *Let $\phi$ be a function from $\mathbb{R}$ to $\mathbb{R}$ such that $\phi \in L^2(\mathbb{R}, e^{-x^2/2})$, and let its Hermite expansion be $\phi = \sum_{i=1}^\infty \sigma_i h_i$. Then, for any unit vectors $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^d$, we have that*

$$\mathbb{E}_{\boldsymbol{x} \sim \mathcal{N}(0, I_{d \times d})} \left[ \phi(\boldsymbol{u}^\top \boldsymbol{x})\phi(\boldsymbol{v}^\top \boldsymbol{x}) \right] = \sum_{i=0}^\infty \sigma_i^2 (\boldsymbol{u}^\top \boldsymbol{v})^i.$$

Therefore we could compute the value of $f_2$ explicitly using the Hermite polynomial expansion:

$$f_2(A, V) = \left\langle \sum_{i=0}^\infty \sigma_i^2 \left( (A^*(A^*)^\top)^{\circ i} - (AA^\top)^{\circ i} \right), V \right\rangle.$$

Here $X^{\circ i}$ is the Hadamard power operation where $(X^{\circ i})_{jk} = (X_{jk})^i$. Therefore we have:

$$g(A) = \frac{1}{2} \left\| \sum_{i=0}^\infty \sigma_i^2 \left( (A^*(A^*)^\top)^{\circ i} - (AA^\top)^{\circ i} \right) \right\|_F^2$$

We reparametrize with $Z = AA^\top$ and define $\tilde{g}(Z) = g(A)$ with individual component functions $\tilde{g}_{jk}(z) \equiv \frac{1}{2}(\sum_{i=0}^\infty \sigma_i^2((z_{jk}^*)^i - z^i))^2$. Accordingly $z_{jk}^* = \langle \boldsymbol{a}_j^*, \boldsymbol{a}_k^* \rangle$ is the $(j, k)$-th component of the ground truth covariance matrix $A^*(A^*)^\top$.

**Assumption 2.** *The activation function $\phi$ is an odd function plus constant. In other words, its Hermite expansion $\phi = \sum_{i=0}^\infty \sigma_i h_i$ satisfies $\sigma_i = 0$ for even $i \geq 2$. Additionally we assume $\sigma_1 \neq 0$.*

**Remark 2.** *Common activations like tanh and sigmoid satisfy Assumption 2.*

**Lemma 2.** *For activations including leaky ReLU and functions satisfying Assumption 2, $\tilde{g}(Z)$ has a unique stationary point where $Z = A^*(A^*)^\top$.*

Notice $\tilde{g}(Z) = \sum_{jk} \tilde{g}_{jk}(z_{jk})$ is separable across $z_{jk}$, where each $\tilde{g}_{jk}$ is a polynomial scalar function. Lemma 2 comes from the fact that the only zero point for $\tilde{g}'_{jk}$ is $z_{jk} = z^*_{jk}$, for odd activation $\phi$ and leaky ReLU. Then we migrate this good property to the original problem we want to solve:

**Problem 1.** *We optimize over function $g$ when $\|a^*_i\| = 1, \forall i$:*

$$\min_A \left\{ g(A) \equiv \frac{1}{2} \left\| \sum_{i=0}^{\infty} \sigma_i^2 \left( (A^*(A^*)^\top)^{\circ i} - (AA^\top)^{\circ i} \right) \right\|_F^2 \right\}$$

*s.t.* $a_i^\top a_i = 1, \forall i.$

Existing work (Journée et al., 2008) connects $\tilde{g}(Z)$ to the optimization over factorized version for $g(A)$ ($g(A) \equiv \tilde{g}(AA^\top)$). Specifically, when $k = d$, all second-order stationary points for $g(A)$ are first-order stationary points for $\tilde{g}(Z)$. Though $\tilde{g}$ is not convex, we are able to show that its first-order stationary points are global optima when the generator is sufficiently expressive, i.e., $k \geq k_0$. In reality we won't know the latent dimension $k_0$, therefore we just choose $k = d$ for simplicity. We get the following conclusion:

**Theorem 3.** *For activations including leaky ReLU and functions satisfying Assumption 2, when $k = d$, all second-order KKT points for problem 1 are global minima. Therefore alternating projected gradient descent-ascent on Eqn. (3) converges to $A$ such that $AA^\top = A^*(A^*)^\top$.*

The extension for non-unit vectors is straightforward, and we defer the analysis to the Appendix.

This main theorem demonstrates the success of gradient descent ascent on learning the ground truth generator. This result is achieved by analyzing two factors. One is the geometric property of our loss function, i.e., all second-order KKT points are global minima. Second, all global minima satisfy $AA^\top = A^*(A^*)^\top$, and for the problem we considered, i.e., one-layer generators, retrieving parameter $AA^\top$ is sufficient in learning the whole generating distribution.

## 6. Finite Sample Analysis

In the previous section, we demonstrate the success of using gradient descent ascent on the population risk. This leaves us the question on how many samples do we need to achieve

small error. In this section, we analyze Algorithm 1, i.e., gradient descent ascent on the following empirical loss:

$$\tilde{f}^{(t)}_{m,n}(A, V)$$
$$= \left\langle \frac{1}{m} \sum_{i=1}^{m} \phi(Az_i^{(t)})\phi(Az_i^{(t)})^\top - \frac{1}{n} \sum_{i=1}^{n} x_i x_i^\top, V \right\rangle$$
$$- \frac{1}{2}\|V\|^2.$$

Notice in each iteration, gradient ascent with step-size 1 finds the optimal solution for $V$. By Danskin's theorem (Danskin, 2012), our min-max optimization is essentially gradient descent over $\tilde{g}^{(t)}_{m,n}(A) \equiv \max_V \tilde{f}^{(t)}_{m,n}(A, V) = \frac{1}{2}\|\frac{1}{m}\sum_{i=1}^{m} \phi(Az_i^{(t)})\phi(Az_i^{(t)})^\top - \frac{1}{n}\sum_{i=1}^{n} x_i x_i^\top\|_F^2$ with a batch of samples $\{z_i^{(t)}\}$, i.e., stochastic gradient descent for $f_n(A) \equiv \mathbb{E}_{z_i \sim \mathcal{N}(0, I_{k \times k}), \forall i \in [m]}[\tilde{g}_{m,n}(A)]$.

Therefore to bound the difference between $f_n(A)$ and the population risk $g(A)$, we analyze the sample complexity required on the observation side ($x_i \sim \mathcal{D}, i \in [n]$) and the mini-batch size required on the learning part ($\phi(Az_j), z_j \sim \mathcal{N}(0, I_{k \times k}), j \in [m]$). We will show that with large enough $n, m$, the algorithm specified in Algorithm 1 that optimizes over the empirical risk will yield the ground truth covariance matrix with high probability.

Our proof sketch is roughly as follows:

1. With high probability, projected stochastic gradient descent finds a second order stationary point $\hat{A}$ of $f_n(\cdot)$ as shown in (Ge et al., 2015; Lee et al., 2019; 2016).

2. For sufficiently large $m$, our empirical objective, though a biased estimator of the population risk $g(\cdot)$, achieves good $\epsilon$-approximation to the population risk on both the gradient and Hessian (Lemmas 4&5). Therefore $\hat{A}$ is also an $\mathcal{O}(\epsilon)$-approximate second order stationary point (SOSP) for the population risk $g(A)$.

3. We show that any $\epsilon$-SOSP $\hat{A}$ for $g(A)$ yields an $\mathcal{O}(\epsilon)$-first order stationary point (FOSP) $\hat{Z} \equiv \hat{A}\hat{A}^\top$ for the semi-definite programming on $\tilde{g}(Z)$ (Lemma 6).

4. We show that any $\mathcal{O}(\epsilon)$-FOSP of function $\tilde{g}(Z)$ induces at most $\mathcal{O}(\epsilon)$ absolute error compared to the ground truth covariance matrix $Z^* = A^*(A^*)^\top$ (Lemma 7).

### 6.1. Observation Sample Complexity

For simplicity, we assume the activation and its gradient satisfy Lipschitz continuous, and let the Lipschitz constants be 1 w.l.o.g.:

**Assumption 3.** *Assume the activation is 1-Lipschitz and 1-smooth.*

To estimate observation sample complexity, we will bound the gradient and Hessian for the population risk and empiri-

---

**Algorithm 1** Online stochastic gradient descent ascent on WGAN

---

1: **Input:** $n$ training samples: $\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots \boldsymbol{x}_n$, where each $\boldsymbol{x}_i \sim \phi(A^* \boldsymbol{z}), \boldsymbol{z} \sim \mathcal{N}(0, I_{k \times k})$, learning rate for generating parameters $\eta$, number of iterations $T$.

2: Random initialize generating matrix $A^{(0)}$.

3: **for** $t = 1, 2, \cdots, T$ **do**

4:      Generate $m$ latent variables $\boldsymbol{z}_1^{(t)}, \boldsymbol{z}_2^{(t)}, \cdots, \boldsymbol{z}_m^{(t)} \sim \mathcal{N}(0, I_{k \times k})$ for the generator. The empirical function becomes

$$\tilde{f}_{m,n}^{(t)}(A, V) = \left\langle \frac{1}{m} \sum_{i=1}^{m} \phi(A \boldsymbol{z}_i^{(t)}) \phi(A \boldsymbol{z}_i^{(t)})^\top - \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{x}_i \boldsymbol{x}_i^\top, V \right\rangle - \frac{1}{2} \|V\|^2$$

5:      Gradient ascent on $V$ with optimal step-size $\eta_V = 1$:

$$V^{(t)} \leftarrow V^{(t)} - \eta_V \nabla_V \tilde{f}_{m,n}^{(t)}(A^{(t-1)}, V^{(t-1)}).$$

6:      Sample noise $\boldsymbol{e}$ uniformly from unit sphere

7:      Projected Gradient Descent on $A$, with constraints $C = \{A | (AA^\top)_{ii} = (A^* A^{*\top})_{ii}\}$:

$$A^{(t)} \leftarrow \text{Proj}_C(A^{(t-1)} - \eta(\nabla_A \tilde{f}_{m,n}^{(t)}(A^{(t-1)}, V^{(t)}) + \boldsymbol{e})).$$

8: **end for**

9: **Output:** $A^{(T)}(A^{(T)})^\top$

---

cal risk on the observation samples:

$$g(A)$$
$$\equiv \frac{1}{2} \left\| \mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}} \left[ \boldsymbol{x} \boldsymbol{x}^\top \right] - \mathbb{E}_{\boldsymbol{z} \sim \mathcal{N}(0, I_{k \times k})} \left[ \phi(A \boldsymbol{z}) \phi(A \boldsymbol{z})^\top \right] \right\|_F^2,$$
$$g_n(A)$$
$$\equiv \frac{1}{2} \left\| \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{x}_i \boldsymbol{x}_i^\top - \mathbb{E}_{\boldsymbol{z} \sim \mathcal{N}(0, I_{k \times k})} \left[ \phi(A \boldsymbol{z}) \phi(A \boldsymbol{z})^\top \right] \right\|_F^2.$$

We calculate the gradient estimation error due to finite samples.

**Claim 2.**

$$\nabla g(A) - \nabla g_n(A)$$
$$= 2 \mathbb{E}_{\boldsymbol{z}} \left[ diag(\phi'(A \boldsymbol{z})) (X - X_n) \phi(A \boldsymbol{z}) \boldsymbol{z}^\top \right],$$

*where $X = \mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}}[\boldsymbol{x} \boldsymbol{x}^\top]$, and $X_n = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{x}_i \boldsymbol{x}_i^\top$. The directional derivative with arbitrary direction $B$ is:*

$$D \nabla g(A)[B] - D \nabla g_n(A)[B]$$
$$= 2 \mathbb{E}_{\boldsymbol{z}} \left[ diag(\phi'(A \boldsymbol{z})) (X_n - X) \phi'(A \boldsymbol{z}) \circ (B \boldsymbol{z}) \boldsymbol{z}^\top \right]$$
$$+ 2 \mathbb{E}_{\boldsymbol{z}} \left[ diag(\phi''(A \boldsymbol{z}) \circ (B \boldsymbol{z})) (X_n - X) \phi(A \boldsymbol{z}) \boldsymbol{z}^\top \right]$$

**Lemma 3.** *Suppose the activation satisfies Assumption 3. We get*

$$\Pr[\|X - X_n\| \le \epsilon \|X\|] \ge 1 - \delta,$$

*for $n \ge \tilde{\Theta}(d/\epsilon^2 \log^2(1/\delta))^2$.*

---

[2] We will use $\tilde{\Theta}$ throughout the paper to hide log factors of $d$ for simplicity.

---

Bounding the relative difference between sample and population covariance matrices is essential for us to bound the estimation error in both gradient and its directional derivative. We can show the following relative error:

**Lemma 4.** *Suppose the activation satisfies Assumption 2&3. With samples $n \ge \tilde{\Theta}(d/\epsilon^2 \log^2(1/\delta))$, we get:*

$$\|\nabla g(A) - \nabla g_n(A)\|_2 \le \mathcal{O}(\epsilon d \|A\|_2),$$

*with probability $1 - \delta$. Meanwhile,*

$$\|D \nabla g(A)[B] - D \nabla g_n(A)[B]\|_2 \le \mathcal{O}(\epsilon d^{3/2} \|A\|_2 \|B\|_2),$$

*with probability $1 - \delta$.*

### 6.2. Bounding Mini-batch Size

Normally for empirical risk for supervised learning, the mini-batch size can be arbitrarily small since the estimator of the gradient is unbiased. However in the WGAN setting, notice for each iteration, we randomly sample a batch of random variables $\{\boldsymbol{z}_i\}_{i \in [m]}$, and obtain a gradient of

$$\tilde{g}_{m,n}(A) \equiv \frac{1}{2} \left\| \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{x}_i \boldsymbol{x}_i^\top - \frac{1}{m} \sum_{j=1}^{m} \phi(A \boldsymbol{z}_j) \phi(A \boldsymbol{z}_j)^\top \right\|_F^2,$$

in Algorithm 1. However, the finite sum is inside the Frobenius norm and the gradient on each mini-batch may no longer be an unbiased estimator for our target

$$g_n(A) = \frac{1}{2} \left\| \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{x}_i \boldsymbol{x}_i^\top - \mathbb{E}_{\boldsymbol{z}} \left[ \phi(A \boldsymbol{z}) \phi(A \boldsymbol{z})^\top \right] \right\|_F^2.$$

In other words, we conduct stochastic gradient descent over the function $f(A) \equiv \mathbb{E}_{\boldsymbol{z}}\tilde{g}_{m,n}(A)$. Therefore we just need to analyze the gradient error between this $f(A)$ and $g_n(A)$ (i.e. $\tilde{g}_{m,n}$ is almost an unbiased estimator of $g_n$). Finally with the concentration bound derived in last section, we get the error bound between $f(A)$ and $g(A)$.

**Lemma 5.** *The empirical risk $\tilde{g}_{m,n}$ is almost an unbiased estimator of $g_n$. Specifically, the expected function $f(A) = \mathbb{E}_{\boldsymbol{z}_i \sim \mathcal{N}(0, I_{k \times k}), i \in [m]}[\tilde{g}_{m,n}]$ satisfies:*

$$\|\nabla f(A) - \nabla g_n(A)\| \leq \mathcal{O}(\frac{1}{m}\|A\|^3 d^2).$$

*For arbitrary direction matrix $B$,*

$$\|D\nabla f(A)[B] - D\nabla g_n(A)[B]\| \leq \mathcal{O}(\frac{1}{m}\|B\|\|A\|^3 d^{5/2}).$$

In summary, we conduct concentration bound over the observation samples and mini-batch sizes, and show the gradient of $f(A)$ that Algorithm 1 is optimizing over has close gradient and Hessian with the population risk $g(A)$. Therefore a second-order stationary point (SOSP) for $f(A)$ (that our algorithm is guaranteed to achieve) is also an $\epsilon$ approximated SOSP for $g(A)$. Next we show such a point also yield an $\epsilon$ approximated first-order stationary point of the reparametrized function $\tilde{g}(Z) \equiv g(A), \forall Z = AA^\top$.

### 6.3. Relation on Approximate Optimality

In this section, we establish the relationship between $\tilde{g}$ and $g$. We present the general form of our target Problem 1:

$$\min_{A \in \mathbb{R}^{d \times k}} g(A) \equiv \tilde{g}(AA^\top) \tag{4}$$
$$\text{s.t. } \text{Tr}(A^\top X_i A) = y_i, X_i \in \mathbb{S}, y_i \in \mathbb{R}, i = 1, \cdots, n.$$

Similar to the previous section, the stationary property might not be obvious on the original problem. Instead, we could look at the re-parametrized version as:

$$\min_{Z \in \mathbb{S}} \quad \tilde{g}(Z) \tag{5}$$
$$\text{s.t.} \quad \text{Tr}(X_i Z) = y_i, X_i \in \mathbb{S}, y_i \in \mathbb{R}, i = 1, \cdots, n,$$
$$Z \succeq 0,$$

**Definition 1.** *A matrix $A \in \mathbb{R}^{d \times k}$ is called an $\epsilon$-approximate second-order stationary point ($\epsilon$-SOSP) of Eqn. (4) if there exists a vector $\lambda$ such that:*

$$\begin{cases} \text{Tr}(A^\top X_i A) = y_i, i \in [n] \\ \|(\nabla_Z \tilde{g}(AA^\top) - \sum_{i=1}^n \lambda_i X_i)\tilde{\boldsymbol{a}}_j\| \leq \epsilon\|\tilde{\boldsymbol{a}}_j\|, \\ \qquad (\{\tilde{\boldsymbol{a}}_j\}_j \text{ span the column space of } A) \\ \text{Tr}(B^\top D\nabla_A \mathcal{L}(A, \lambda)[B]) \geq -\epsilon\|B\|^2, \\ \qquad \forall B \text{ s.t. } \text{Tr}(B^\top X_i A) = 0 \end{cases}$$

*Here $\mathcal{L}(A, \lambda)$ is the Lagrangian form $\tilde{g}(AA^\top) - \sum_{i=1}^n \lambda_i(\text{Tr}(A^\top X_i A) - y_i)$.*

Specifically, when $\epsilon = 0$ the above definition is exactly the second-order KKT condition for optimizing (4). Next we present the approximate first-order KKT condition for (5):

**Definition 2.** *A symmetric matrix $Z \in \mathbb{S}^n$ is an $\epsilon$-approximate first order stationary point of function (5) ($\epsilon$-FOSP) if and only if there exist a vector $\sigma \in \mathbb{R}^m$ and a symmetric matrix $S \in \mathbb{S}$ such that the following holds:*

$$\begin{cases} \text{Tr}(X_i Z) = y_i, i \in [n] \\ Z \succeq 0, \\ S \succeq -\epsilon I, \\ \|S\tilde{\boldsymbol{a}}_j\| \leq \epsilon\|\tilde{\boldsymbol{a}}_j\|, \\ \qquad (\{\tilde{\boldsymbol{a}}_j\}_j \text{ span the column space of } Z) \\ S = \nabla_Z \tilde{g}(Z) - \sum_{i=1}^n \sigma_i X_i. \end{cases}$$

**Lemma 6.** *Let latent dimension $k = d$. For an $\epsilon$-SOSP of function (4) with $A$ and $\lambda$, it infers an $\epsilon$-FOSP of function (5) with $Z, \sigma$ and $S$ that satisfies: $Z = AA^\top, \sigma = \lambda$ and $S = \nabla_Z \tilde{g}(AA^\top) - \sum_i \lambda_i X_i$.*

Now it remains to show an $\epsilon$-FOSP of $\tilde{g}(Z)$ indeed yields a good approximation for the ground truth parameter matrix.

**Lemma 7.** *If $Z$ is an $\epsilon$-FOSP of function (5), then $\|Z - Z^*\|_F \leq \mathcal{O}(\epsilon)$. Here $Z^* = A^*(A^*)^\top$ is the optimal solution for function (5).*

Together with the previous arguments, we finally achieve our main theorem on connecting the recovery guarantees with the sample complexity and batch size[3]:

**Theorem 4.** *For arbitrary $\delta < 1, \epsilon$, given small enough learning rate $\eta < 1/poly(d, 1/\epsilon, \log(1/\delta))$, let sample size $n \geq \tilde{\Theta}(d^5/\epsilon^2 \log^2(1/\delta))$, batch size $m \geq \mathcal{O}(d^5/\epsilon)$, for large enough $T = poly(1/\eta, 1/\epsilon, d, \log(1/\delta))$, the output of Algorithm 1 satisfies*

$$\|A^{(T)}(A^{(T)})^\top - Z^*\|_F \leq \mathcal{O}(\epsilon),$$

*with probability $1 - \delta$, under Assumptions 2 & 3 and $k = d$.*

Therefore we have shown that with finite samples of $poly(d, 1/\epsilon)$, we are able to learn the generating distribution with error measured in the parameter space, using stochastic gradient descent ascent. This echos the empirical success of training WGAN. Meanwhile, notice our error bound matches the lower bound on dependence of $1/\epsilon$, as suggested in (Wu et al., 2019).

## 7. Experiments

In this section, we provide simple experimental results to validate the performance of stochastic gradient descent ascent and provide experimental support for our theory.

---

[3]The exact error bound comes from the fact that when diagonal terms of $AA^\top$ are fixed, $\|A\|_2 = \mathcal{O}(\sqrt{d})$.
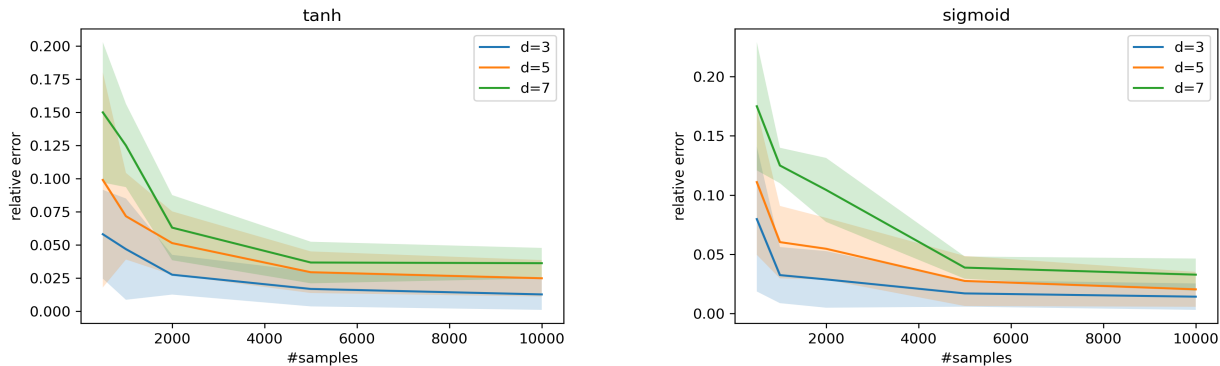
Figure 1: Recovery error ($\|AA^\top - Z^*\|_F$) with different observed sample sizes $n$ and output dimension $d$.



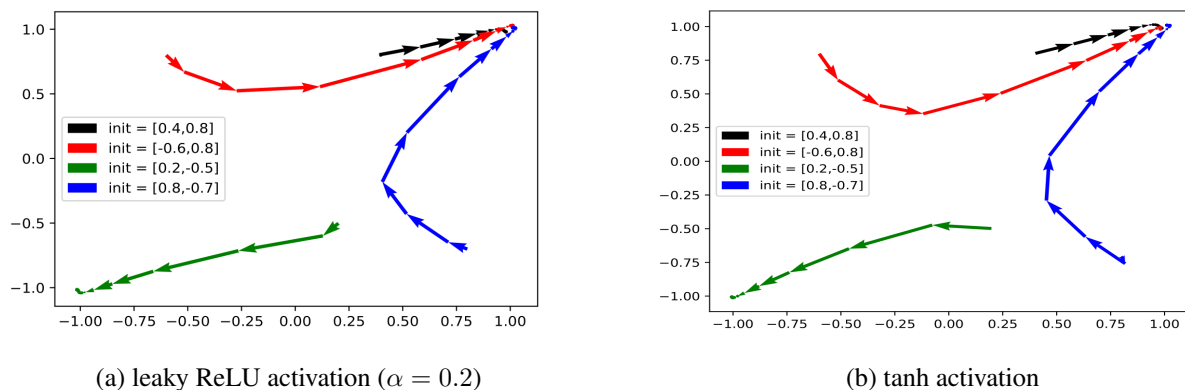(a) leaky ReLU activation ($\alpha = 0.2$)

(b) tanh activation

Figure 2: Comparisons of different performance with leakyReLU and tanh activations. Same color starts from the same starting point. For both cases, parameters always converge to true covariance matrix. Each arrow indicates the progress of 500 iteration steps.

We focus on Algorithm 1 that targets to recover the parameter matrix. We conduct a thorough empirical studies on three joint factors that might affect the performance: the number of observed samples $m$ (we set $n = m$ as in general GAN training algorithms), the different choices of activation function $\phi$, and the output dimension $d$.

In Figure 1 we plot the relative error for parameter estimation decrease over the increasing sample complexity. We fix the hidden dimension $k = 2$, and vary the output dimension over $\{3, 5, 7\}$ and sample complexity over $\{500, 1000, 2000, 5000, 10000\}$. Reported values are averaged from 20 runs and we show the standard deviation with the corresponding colored shadow. Clearly the recovery error decreases with higher sample complexity and smaller output dimension. From the experimental results, we can see that our algorithm always achieves global convergence to the ground truth generators from any random initialization point.

To visually demonstrate the learning process, we also in-

clude a simple comparison for different $\phi$: i.e. leaky ReLU and tanh activations, when $k = 1$ and $d = 2$. We set the ground truth covariance matrix to be $[1, 1; 1, 1]$, and therefore a valid result should be $[1, 1]$ or $[-1, -1]$. From Figure 2 we could see that for both leaky ReLU and tanh, the stochastic gradient descent ascent performs similarly with exact recovery of the ground truth parameters.

## 8. Conclusion

We analyze the convergence of stochastic gradient descent ascent for Wasserstein GAN on learning a single layer generator network. We show that stochastic gradient descent ascent algorithm attains the global min-max point, and provably recovers the parameters of the network with $\epsilon$ absolute error measured in Frobenius norm, from $\tilde{\Theta}(d^5/\epsilon^2)$ i.i.d samples.

# Acknowledgements

# References

Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein generative adversarial networks. In *International conference on machine learning*, pp. 214–223, 2017.

Arora, S., Ge, R., Liang, Y., Ma, T., and Zhang, Y. Generalization and equilibrium in generative adversarial nets (GANs). In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 224–232. JMLR. org, 2017.

Arora, S., Risteski, A., and Zhang, Y. Do GANs learn the distribution? some theory and empirics. 2018.

Bai, Y., Ma, T., and Risteski, A. Approximability of discriminators implies diversity in GANs. *arXiv preprint arXiv:1806.10586*, 2018.

Bora, A., Jalal, A., Price, E., and Dimakis, A. G. Compressed sensing using generative models. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 537–546. JMLR. org, 2017.

Daniely, A., Frostig, R., and Singer, Y. Toward deeper understanding of neural networks: The power of initialization and a dual view on expressivity. In *Advances In Neural Information Processing Systems*, pp. 2253–2261, 2016.

Danskin, J. M. *The theory of max-min and its application to weapons allocation problems*, volume 5. Springer Science & Business Media, 2012.

Daskalakis, C. and Panageas, I. Last-iterate convergence: Zero-sum games and constrained min-max optimization. *arXiv preprint arXiv:1807.04252*, 2018a.

Daskalakis, C. and Panageas, I. The limit points of (optimistic) gradient descent in min-max optimization. In *Advances in Neural Information Processing Systems*, pp. 9236–9246, 2018b.

Daskalakis, C., Ilyas, A., Syrgkanis, V., and Zeng, H. Training gans with optimism. *arXiv preprint arXiv:1711.00141*, 2017.

Daskalakis, C., Gouleakis, T., Tzamos, C., and Zampetakis, M. Efficient statistics, in high dimensions, from truncated samples. In *the 59th IEEE Annual Symposium on Foundations of Computer Science (FOCS)*, 2018.

Dumoulin, V., Belghazi, I., Poole, B., Mastropietro, O., Lamb, A., Arjovsky, M., and Courville, A. Adversarially learned inference. *arXiv preprint arXiv:1606.00704*, 2016.

Feizi, S., Farnia, F., Ginart, T., and Tse, D. Understanding GANs: the LQG setting. *arXiv preprint arXiv:1710.10793*, 2017.

Ge, R., Huang, F., Jin, C., and Yuan, Y. Escaping from saddle points—online stochastic gradient for tensor decomposition. In *Conference on Learning Theory*, pp. 797–842, 2015.

Ge, R., Lee, J. D., and Ma, T. Learning one-hidden-layer neural networks with landscape design. *arXiv preprint arXiv:1711.00501*, 2017.

Gidel, G., Hemmat, R. A., Pezeshki, M., Priol, R. L., Huang, G., Lacoste-Julien, S., and Mitliagkas, I. Negative momentum for improved game dynamics. In *the 22nd International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2019.

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

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. C. Improved training of wasserstein gans. In *Advances in neural information processing systems*, pp. 5767–5777, 2017.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, pp. 6626–6637, 2017.

Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1125–1134, 2017.

Jin, C., Netrapalli, P., and Jordan, M. I. Minmax optimization: Stable limit points of gradient descent ascent are locally optimal. *arXiv preprint arXiv:1902.00618*, 2019.

Journée, M., Bach, F., Absil, P.-A., and Sepulchre, R. Low-rank optimization for semidefinite convex problems. *arXiv preprint arXiv:0807.4423*, 2008.

Karras, T., Aila, T., Laine, S., and Lehtinen, J. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

Korpelevich, G. The extragradient method for finding saddle points and other problems. *Matecon*, 12:747–756, 1976.

Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4681–4690, 2017.

Lee, J. D., Simchowitz, M., Jordan, M. I., and Recht, B. Gradient descent only converges to minimizers. In *Conference on learning theory*, pp. 1246–1257, 2016.

Lee, J. D., Panageas, I., Piliouras, G., Simchowitz, M., Jordan, M. I., and Recht, B. First-order methods almost always avoid strict saddle points. *Mathematical programming*, 176(1-2):311–337, 2019.

Lei, Q., Yen, I. E.-H., Wu, C.-y., Dhillon, I. S., and Ravikumar, P. Doubly greedy primal-dual coordinate descent for sparse empirical risk minimization. In *International Conference on Machine Learning*, pp. 2034–2042, 2017.

Lei, Q., Zhuo, J., Caramanis, C., Dhillon, I. S., and Dimakis, A. G. Primal-dual block generalized frank-wolfe. In *Advances in Neural Information Processing Systems*, pp. 13866–13875, 2019.

Lei, Q., Nagarajan, S. G., Panageas, I., and Wang, X. Last iterate convergence in no-regret learning: constrained min-max optimization for convex-concave landscapes. *arXiv preprint arXiv:2002.06768*, 2020.

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

Liang, T. and Stokes, J. Interaction matters: A note on non-asymptotic local convergence of generative adversarial networks. In *the 22nd International Conference on Artificial Intelligence and Statistics ( AISTATS)*, 2019.

Lin, T., Jin, C., and Jordan, M. I. On gradient descent ascent for nonconvex-concave minimax problems. *arXiv preprint arXiv:1906.00331*, 2019.

Mescheder, L., Nowozin, S., and Geiger, A. The numerics of GANs. In *Advances in Neural Information Processing Systems*, pp. 1825–1835, 2017.

Mescheder, L., Geiger, A., and Nowozin, S. Which training methods for GANs do actually converge? *arXiv preprint arXiv:1801.04406*, 2018.

Mokhtari, A., Ozdaglar, A., and Pattathil, S. A unified analysis of extra-gradient and optimistic gradient methods for saddle point problems: Proximal point approach. *arXiv preprint arXiv:1901.08511*, 2019.

Nagarajan, V. and Kolter, J. Z. Gradient descent GAN optimization is locally stable. In *Advances in Neural Information Processing Systems*, pp. 5585–5595, 2017.

Tzeng, E., Hoffman, J., Saenko, K., and Darrell, T. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7167–7176, 2017.

Vershynin, R. Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*, 2010.

Wu, S., Dimakis, A. G., and Sanghavi, S. Learning distributions generated by one-layer ReLU networks. *arXiv preprint arXiv:1909.01812*, 2019.

Zhang, P., Liu, Q., Zhou, D., Xu, T., and He, X. On the discrimination-generalization tradeoff in GANs. 2018.