A Solvable High-Dimensional Model of GAN

Chuang Wang, Hong Hu and Yue M. Lu
John A. Paulson School of Engineering and Applied Sciences
Harvard University
33 Oxford Street, Cambridge, MA 02138, USA
{chuangwang,honghu}@g.harvard.edu, yuelu@seas.harvard.edu

Abstract

Despite the remarkable successes of generative adversarial networks (GANs) in many applications, theoretical understandings of their performance is still limited. In this paper, we present a simple shallow GAN model fed by high-dimensional input data. The dynamics of the training process of the proposed model can be exactly analyzed in the high-dimensional limit. In particular, by using the tool of scaling limits of stochastic processes, we show that the macroscopic quantities measuring the quality of the training process converge to a deterministic process that is characterized as the unique solution of a finite-dimensional ordinary differential equation (ODE). The proposed model is simple, but its training process already exhibits several different phases that can mimic the behaviors of more realistic GAN models used in practice. Specifically, depending on the choice of the learning rates, the training process can reach either a successful, a failed, or an oscillating phase. By studying the steady-state solutions of the limiting ODEs, we obtain a phase diagram that precisely characterizes the conditions under which each phase takes place. Although this work focuses on a simple GAN model, the analysis methods developed here might prove useful in the theoretical understanding of other variants of GANs with more advanced training algorithms.

1 Introduction

A generative adversarial network (GAN) [1] seeks to learn a high-dimensional probability distribution from samples. It consists of a generator and a discriminator. The generator produces fake data that try to fool the discriminator, whereas the discriminator aims to distinguish real samples from the fake ones. Training a GAN amounts to finding the best generator that can fool the most powerful discriminator, and this process is formulated as a MinMax game. In the past several years, GANs [1] and their variants [2, 3] have achieved remarkable successes in many applications, such as image super-resolution [4], image-to-image translation [5], and text-to-image generations [6]. While there have been numerous advances on the application front, considerably less is known about the underlying theory and conditions that can explain or guarantee the successful trainings of GANs.

The first theoretical analysis was given in the paper that originally proposed the GAN framework [1]. It is shown that the min-max game associated with the training process has a unique solution — with the generator precisely learning the real distribution and the discriminator completely fooled — if the number of training samples is infinite and if both the generator and the discriminator have unlimited capacities. Unfortunately, such idealized settings are far from reality. Moreover, convergence to such solution, if it indeed exists, is not guaranteed if one uses the standard stochastic gradient descent/ascent (SGDA) algorithm, which simultaneously optimizes the generator and discriminator.

Recently, it has been a very active area of research to study either the equilibrium properties [7–9] or the training dynamics [10–15] of various GAN models. An equilibrium analysis [7] revealed that a generator that uniformly draws $\mathcal{O}(n\log(n))$ real samples can fool a discriminator whose capacity is

Preprint. Work in progress.

bounded by n. It implies that the generator may collapse to a few modes if the discriminator is too weak. On the other hand, another line of work [2, 8] shows that if a discriminator is so powerful that it can perfectly distinguish real data from fake ones, the original GAN can suffer from the vanishing gradient problem during the training process. Adding artificial noise [16] or using Wasserstein GAN (WGAN) [2] can help to address this problem to some extent, as the growth of the objective function of WGAN is bounded by a linear function.

In this paper, we present a *high-dimensional* and *exactly solvable* model of GAN. Unlike previous work in the literature that studies GANs in various low-dimensional settings (see, *e.g.*, [10–13]), the model we propose here is high-dimensional: we assume that the ambient dimension n of the training samples is large and we study the limit as $n \to \infty$. Moreover, the number of parameters in the proposed GAN model, the number of training samples, and the number of iterations in the training process can all grow to infinity (in proportion to n). Our model is also exactly solvable: using the vanilla SGDA as the training algorithm, we obtain an asymptotically exact characterization of the dynamics of the training process. Specifically, our main technical contributions are twofold:

- We present an asymptotically exact theoretical analysis of the dynamics of the training process of the proposed GAN model. Our analysis is carried out on both the *macroscopic* and the *microscopic* levels, the precise definitions of which can be found in Section 3. The macroscopic state measures the overall performance of the training process, whereas the microscopic state contains all the detailed information. In the high-dimensional limit ($n \to \infty$), we show that the former converges to a deterministic process governed by an ordinary differential equation (ODE), whereas the latter stays stochastic, whose time-varying probability laws are characterized by a nonlinear partial differential equation (PDE).
- Despite the simplicity of the proposed model, we show that its training process can exhibit three markedly different phases, which mimic the behaviors of more realistic GAN models used in practice. Specifically, depending on the choice of the learning rates, the training process can reach either a successful, a failed, or an oscillating phase. By studying the stabilities of the fixed points of the limiting ODEs, we obtain a phase diagram that precisely characterizes the conditions under which each phase takes place.

Our work builds upon a general analysis framework [17] for studying the scaling limits of high-dimensional exchangeable stochastic processes with applications to online sparse PCA [18], ICA [19], subspace estimation [20] and nonlinear regression problems [17]. Similar techniques have also been used in the literature to study Monte Carlo methods [21], online perceptron learning [22, 23], and more recently, the supervised learning of two-layer neural networks [24].

Our analysis has connections to recent work (*e.g.* [10–13]) that uses stochastic approximation [25] to study the dynamics of GANs, but there are important distinctions. The analysis in [10–13] keeps the ambient dimension n fixed and studies the asymptotic limit as the step size tends to 0. Essentially, under this setting, the stochasticity in the gradient of the training process becomes negligible and thus the microscopic state (*i.e.*, the iterand of the training algorithm) converges to a deterministic process that is the solution of an ODE. The resulting ODE involves $\mathcal{O}(n)$ variables. In contrast, our analysis studies the limit as $n \to \infty$ but the limiting ODE for the macroscopic states only involves 5 variables (See Theorem 1.) This low-dimensional characterization makes our limiting results more practical to use. Moreover, in our analysis, the microscopic states remains stochastic (see Section 3.2), which more closely reflects the actual settings encountered in practice.

The rest of the paper is organized as follows. We present the proposed GAN model and the associated training algorithm in Section 2. Our main results are presented in Section 3, where we show that the macroscopic and microscopic dynamics of the training process converge to their respective limiting processes that are characterized by an ODE and PDE, respectively. In Section 4, we analyze the stationary solutions of the limiting ODEs and present a phase diagram that precisely characterizes the long-term behaviors of the training process. We conclude in Section 5.

2 Formulations

In this section, we introduce the proposed GAN model and specify the associated training algorithm. We also present two concrete examples of the proposed model.

Model for the real data. In order to establish the theoretical analysis, we first impose a model for the probability distribution from which we draw our real data samples. In this work, we assume that the real data $\boldsymbol{y}_k \in \mathbb{R}^n$, $k = 0, 1, \ldots$ are drawn according to the following generative model:

$$\boldsymbol{y}_k = \mathcal{G}(c_k, \boldsymbol{a}_k; \boldsymbol{u}) \stackrel{\text{def}}{=} \boldsymbol{u} c_k + \boldsymbol{a}_k, \tag{1}$$

where $\boldsymbol{u} \in \mathbb{R}^n$ is a deterministic unknown feature vector, c_k is a random variable drawn from an unknown distribution P_c, and $\boldsymbol{a}_k$ is an n-dimensional random vector acting as the background noise. Without loss of generality, we assume $\|\boldsymbol{u}\|^2 = 1$. Note that this generative model, referred to as the spiked covariance model [26] in the literature, is commonly used in the theoretical study of principal component analysis.

The GAN model The GAN we are going to analyze is defined as follows. We assume that the generator $\mathcal{G}$ has the same linear structure as the real data model (1) given above:

$$\widetilde{\boldsymbol{y}}_k = \mathcal{G}(\widetilde{c}_k, \widetilde{\boldsymbol{a}}_k; \widetilde{\boldsymbol{w}}), \tag{2}$$

but the parameters are different. Here, $\widetilde{\boldsymbol{y}}_k$ denotes a fake sample produced by the generator; $\widetilde{\boldsymbol{a}}_k$ is an n-dimensional random vector similar to $\boldsymbol{a}_k$ in (1); the random variable $\widetilde{c}_k$ is drawn from a distribution $P_{\widetilde{c}}$ that is not necessarily the same as P_c; and the vector $\widetilde{\boldsymbol{w}}$ represents the parameters of the generator. (In an ideal case in which the generator learns the underlying true probability distribution perfectly, we have $\widetilde{\boldsymbol{w}} = \boldsymbol{u}$.) Throughout the paper, we follow the notational convention that all the symbols that are decorated with a tilde (*e.g.*, $\widetilde{\boldsymbol{y}}_k$, $\widetilde{c}_k$, $\widetilde{\boldsymbol{a}}_k$, $\widetilde{\boldsymbol{w}}$) denote quantities associated with the generator.

We define the discriminator $\mathcal{D}$ of our GAN model as

$$\mathcal{D}(\boldsymbol{x}; \boldsymbol{w}) \stackrel{\text{def}}{=} \widehat{D}(\boldsymbol{x}^\top \boldsymbol{w}).$$

Here, $\boldsymbol{x}$ is an input vector, which can be either the real data $\boldsymbol{y}_k$ from (1) or the fake one $\widetilde{\boldsymbol{y}}_k$ from (2); $\widehat{D} : \mathbb{R} \mapsto \mathbb{R}$ can be any function; and the vector $\boldsymbol{w} \in \mathbb{R}^n$ represents the parameters associated with the discriminator.

The training algorithm. The proposed GAN model has two parameter vectors $\boldsymbol{w}$ and $\widetilde{\boldsymbol{w}}$ to be learned from the data. The training process is formulated as the following MinMax problem

$$\min_{\widetilde{\boldsymbol{w}}} \max_{\boldsymbol{w}} \mathbb{E}_{\boldsymbol{y} \sim P(\boldsymbol{y}; \boldsymbol{u})} \mathbb{E}_{\widetilde{\boldsymbol{y}} \sim \widetilde{P}(\widetilde{\boldsymbol{y}}, \widetilde{\boldsymbol{w}})} L(\boldsymbol{y}, \widetilde{\boldsymbol{y}}; \boldsymbol{w}) - \tfrac{\lambda}{2} H(\|\boldsymbol{w}\|^2) + \tfrac{\lambda}{2} H(\|\widetilde{\boldsymbol{w}}\|^2), \tag{3}$$

where $L(\boldsymbol{y}, \widetilde{\boldsymbol{y}}; \boldsymbol{w}) \stackrel{\text{def}}{=} F(\widehat{D}(\boldsymbol{y}^\top \boldsymbol{w})) - \widetilde{F}(\widehat{D}(\widetilde{\boldsymbol{y}}^\top \boldsymbol{w}))$ is the main cost function, with $F(\cdot)$ and $\widetilde{F}(\cdot)$ being two functions that quantify the performance of the discriminator; $H(\cdot)$ is a regularization term introduced to control the magnitude of the parameters $\boldsymbol{w}$ and $\widetilde{\boldsymbol{w}}$; $\lambda > 0$ is a constant; and $P(\boldsymbol{y}; \boldsymbol{u})$ and $\widetilde{P}(\widetilde{\boldsymbol{y}}; \widetilde{\boldsymbol{w}})$ represent the distributions of the real data $\boldsymbol{y}$ and the fake data $\widetilde{\boldsymbol{y}}$ as specified by (1) and (2), respectively.

We consider a standard training algorithm that uses the vanilla stochastic gradient descent/ascent (SGDA) to seek a solution of (3). To simplify the theoretical analysis, we consider an online (*i.e.*, streaming) setting where each data sample $\boldsymbol{y}_k$ is used only once. At step k, the model parameters $\boldsymbol{w}_k$ and $\widetilde{\boldsymbol{w}}_k$ are updated using a new real sample $\boldsymbol{y}_k$ and two fake samples $\widetilde{\boldsymbol{y}}_{2k}$ and $\widetilde{\boldsymbol{y}}_{2k+1}$, according to

$$\begin{aligned} \boldsymbol{w}_{k+1} &= \boldsymbol{w}_k + \tfrac{\tau}{n} \big[\nabla_{\boldsymbol{w}_k} L(\boldsymbol{y}_k, \widetilde{\boldsymbol{y}}_{2k}; \boldsymbol{w}_k) - \lambda H'(\|\boldsymbol{w}_k\|^2) \boldsymbol{w}_k \big] \\ \widetilde{\boldsymbol{w}}_{k+1} &= \widetilde{\boldsymbol{w}}_k - \tfrac{\widetilde{\tau}}{n} \big[\nabla_{\widetilde{\boldsymbol{w}}_k} L\big(\boldsymbol{y}_k, G(\widetilde{c}_{2k+1}, \widetilde{\boldsymbol{a}}_{2k+1}, \widetilde{\boldsymbol{w}}_k); \boldsymbol{w}_k\big) + \lambda H'(\|\widetilde{\boldsymbol{w}}_k\|^2) \widetilde{\boldsymbol{w}}_k \big], \end{aligned} \tag{4}$$

where $\widetilde{c}_{2k+1}, \widetilde{\boldsymbol{a}}_{2k+1}$ are random variables that generates the fake sample $\widetilde{\boldsymbol{y}}_{2k+1}$ according to (2), and $H'(\cdot)$ denotes the derivative of $H(\cdot)$. The two parameters τ and $\widetilde{\tau}$ in the above expressions control the learning rates of the discriminator and the generator, respectively. In (4), we only consider a single-step update for $\boldsymbol{w}_k$, and thus this is a special case of Algorithm 1 in [1] with the batch-size m set to 1. We note that the analysis presented in this paper can be naturally extended to the mini-batch case where m is a finite number.

Example 1. Following the original GAN model proposed in [1], we can set the two functions F and $\widetilde{F}$ in (3) to $F(x) = \log(x)$ and $\widetilde{F}(x) = -\log(1 - x)$, respectively. We also define a discriminator that mimics a single-layer neural network with a unique mode and a constant bias -1. Specifically, we let $\widehat{D}(x) = \phi\big(\psi(x - 1) - 1\big)$ in (3), where $\phi(x) = (1 + e^{-x})^{-1}$ and $\psi(x) = \log(1 + e^x)$ are the sigmoid and the rectifier functions, respectively. The regularization function is chosen to be $H(x) = x$.

Example 2. We also consider an example for the Wasserstein GAN [2], where we define $F(\widehat{D}(x)) = \widetilde{F}(\widehat{D}(x)) = x^2/2$, and $H(x) = \log\cosh(x-1)$. Furthermore, by setting the regularization parameter $\lambda \to \infty$, the original problem (3) becomes a constrained MinMax problem

$$\min_{\|\widetilde{\boldsymbol{w}}\|=1} \max_{\|\boldsymbol{w}\|=1} \mathbb{E}_{\boldsymbol{y}\sim P(\boldsymbol{y};\boldsymbol{u})} \mathbb{E}_{\widetilde{\boldsymbol{y}}\sim \widetilde{P}(\widetilde{\boldsymbol{y}},\widetilde{\boldsymbol{w}})} \left[(\boldsymbol{y}^\top \boldsymbol{w})^2 - (\widetilde{\boldsymbol{y}}^\top \boldsymbol{w})^2 \right].$$

We will focus on this special case when we investigate the steady-state behaviors of the training algorithm in Section 4.

As a preview, Figure 1 illustrates our analysis for the training process under the settings stated in Example 1. (The results for Example 2 are similar.) Despite of the simplicity of the proposed GAN model, Figure 1 shows that its training process can exhibit very complicated patterns. Depending on the choice of the learning rates τ and $\widetilde{\tau}$, the process will fall into one of three possible phases: success, failure or oscillating. In all these cases, our asymptotic analysis (to be presented in the next section) can accurately predict the dynamics of the algorithms.

3 Dynamics of the GAN

Definition 1. Let $\boldsymbol{X}_k \stackrel{\text{def}}{=} [\boldsymbol{u}, \widetilde{\boldsymbol{w}}_k, \boldsymbol{w}_k] \in \mathbb{R}^{n\times 3}$. We call $\boldsymbol{X}_k$ the *microscopic state* of the training process at iteration step k.

The microscopic state $\boldsymbol{X}_k$ contains all the information about the training process. In fact, the sequence $\{\boldsymbol{X}_k\}_{k=0,1,2,\ldots}$ forms a Markov chain on $\mathbb{R}^{n\times 3}$. This can be easily verified from the update rule of $\boldsymbol{X}_k$ as defined in (4), in which the real data $\boldsymbol{y}_k$ and fake data $\widetilde{\boldsymbol{y}}_k$ are drawn according to (1) and (2) respectively. The Markov chain is driven by the initial state $\boldsymbol{X}_0$ and the sequence of random variables $\{(c_k, \boldsymbol{a}_k, \widetilde{c}_{2k}, \widetilde{\boldsymbol{a}}_{2k}, \widetilde{c}_{2k+1}, \widetilde{\boldsymbol{a}}_{2k+1})\}_{k=0,1,2,\ldots}$.

Definition 2. We define $\boldsymbol{M}_k \stackrel{\text{def}}{=} \boldsymbol{X}_k^\top \boldsymbol{X}_k$ as the *macroscopic state* of the Markov chain $\boldsymbol{X}_k$ at step k.

By construction, $\boldsymbol{M}_k = \begin{bmatrix} 1 & \widetilde{q}_k & q_k \\ \widetilde{q}_k & \widetilde{z}_k & r_k \\ q_k & r_k & z_k \end{bmatrix}$ is a 3×3 matrix. Due to symmetry, the macroscopic state $\boldsymbol{M}_k$ can be equivalently represented by a 5-dimensional vector $\boldsymbol{m}_k \stackrel{\text{def}}{=} \begin{bmatrix} \widetilde{q}_k & \widetilde{z}_k & q_k & z_k & r_k \end{bmatrix}^\top$.

Each element of $\boldsymbol{M}_k$ has a clear geometric meaning. The diagonal entries $\widetilde{z}_k$ and z_k are the squared norm of $\widetilde{\boldsymbol{w}}_k$ and $\boldsymbol{w}_k$, respectively. The cosines of the angles among the three vectors $\boldsymbol{u}$, $\widetilde{\boldsymbol{w}}_k$ and $\boldsymbol{w}_k$ are specified by $\cos(\angle(\boldsymbol{u}, \widetilde{\boldsymbol{w}}_k)) = \widetilde{q}_k/\sqrt{\widetilde{z}_k}$, $\cos(\angle(\boldsymbol{u}, \boldsymbol{w}_k)) = q_k/\sqrt{z_k}$ and $\cos(\angle(\widetilde{\boldsymbol{w}}_k, \boldsymbol{w}_k)) = r_k/\sqrt{\widetilde{z}_k z_k}$.

In what follows, we investigate the dynamics of the training algorithm (4) at both the macroscopic and the microscopic levels. At the macroscopic level, by examining the cosines of the angles, we study how closely the model parameters $\widetilde{\boldsymbol{w}}_k$, $\boldsymbol{w}_k$ associated with the generator and discriminator can align with the ground truth feature vector $\boldsymbol{u}$. At the microscopic level, we study how the elements in the vectors $\widetilde{\boldsymbol{w}}_k$ and $\boldsymbol{w}_k$ evolve as a stochastic process. As our analysis will reveal, the mechanisms behind the two levels are different: the macroscopic dynamics is asymptotically deterministic whereas the microscopic dynamics stays stochastic even as $n \to \infty$.

3.1 Macroscopic dynamics

We first study the asymptotic dynamics of the macroscopic state $\boldsymbol{m}_k$. Our theoretical analysis is carried out under the following assumptions.

- (A.1) Both of the sequences of $c_k \sim P_c$ and $\widetilde{c}_k \sim P_{\widetilde{c}}$ for $k = 0, 1, \ldots$ are i.i.d. random variables with bounded moments of all orders. In addition, $\{c_k\}$ is independent of $\{\widetilde{c}_k\}$.
- (A.2) The sequences $\{\boldsymbol{a}_k\}$ and $\{\widetilde{\boldsymbol{a}}_k\}$ for $k = 0, 1, \ldots$ are both independent Gaussian vectors with zero mean and covariance matrix $\boldsymbol{I}_n$. Moreover, $\{\boldsymbol{a}_k\}$, $\{\widetilde{\boldsymbol{a}}_k\}$ are independent of $\{c_k\}$ and $\{\widetilde{c}_k\}$.
- (A.3) The first-order derivative $H'(x)$ and the derivatives up to fourth order of the functions $F(\widehat{D}(x))$ and $\widetilde{F}(\widehat{D}(x))$ exist and they are also uniformly bounded.

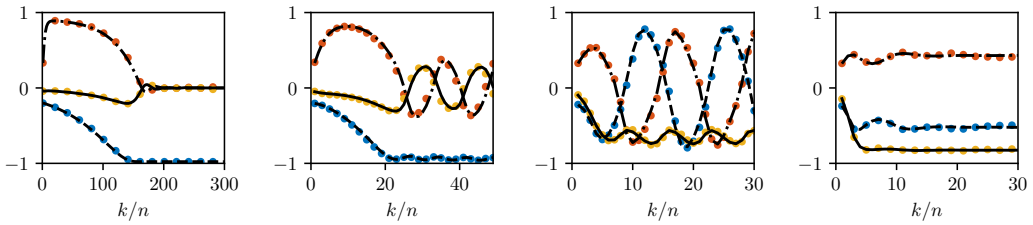


Figure 1: Macroscopic dynamics of the GAN: The colored dots show the experimental results of a single trial of the training algorithm as described in (4) and the black curves under the dots are theoretical predictions given by Theorem 1. The red, blue and yellow dots represent $\cos(\angle(\boldsymbol{u}, \boldsymbol{w}_k))$, $\cos(\angle(\boldsymbol{u}, \widetilde{\boldsymbol{w}}_k))$, $\cos(\angle(\boldsymbol{w}_k, \widetilde{\boldsymbol{w}}_k))$ respectively. We set a fixed learning rate $\tau = 1$ for the discriminator. The four figures from left to right correspond to setting the generator's learning rate to $\widetilde{\tau} = 0.1$. 0.5, 1.5 and 2, respectively. When $\widetilde{\tau} = 0.1$, the training process ends up in a state where $\mathcal{G}$ wins and $\mathcal{D}$ completely loses. When $\widetilde{\tau} = 0.5$, it reaches a state oscillating around the state of the first experiment ($\widetilde{\tau} = 0.1$). When $\widetilde{\tau} = 1$, it reaches a different type of oscillatory state where $\mathcal{G}$ and $\mathcal{D}$ are highly correlated. When $\widetilde{\tau} = 1.5$, the training process reaches a stationary state in which both $\mathcal{G}$ and $\mathcal{D}$ have some correlations with the true feature vector.

(A.4) Let u_i, $w_{0,i}$ and $\widetilde{w}_{0,i}$ denote the ith elements of $\boldsymbol{u}$, $\boldsymbol{w}_0$ and $\widetilde{\boldsymbol{w}}_0$, respectively. For $i = 1, 2, \ldots, n$, we have $\mathbb{E}\,(u_i^4 + w_{0,i}^4 + \widetilde{w}_{0,i}^4) \le C/n^2$, where C is a constant that does not depend on n.

(A.5) The initial macroscopic state $\boldsymbol{m}_0$ satisfies $\mathbb{E}\,\|\boldsymbol{m}_0 - \boldsymbol{m}_0^*\| \le C/\sqrt{n}$, where $\boldsymbol{m}_0^*$ is a deterministic vector.

Theorem 1. *Fix $T > 0$. It holds under Assumptions (A.1)–(A.5) that*

$$\max_{0 \le k \le nT} \mathbb{E}\,\|\boldsymbol{m}_k - \boldsymbol{m}(\tfrac{k}{n})\| \le \frac{C(T)}{\sqrt{n}},$$

where $C(T)$ is a constant that depends on T but not on n, and $\boldsymbol{m}(t)$ is a deterministic function that is the unique solution of the following ODE:

$$\tfrac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{m}(t) = \boldsymbol{g}(\boldsymbol{m}(t)), \text{ with the initial condition } \boldsymbol{m}(0) = \boldsymbol{m}_0^*. \tag{5}$$

The complete proof and the detailed expression of $\boldsymbol{g}(\cdot)$ can be found in the Supplementary Materials [27]. This theorem implies that for each $k = \lfloor tn \rfloor$ for some $t \in [0, T]$, the macroscopic state $\boldsymbol{m}_k$ converges in probability to a deterministic number $\boldsymbol{m}(t)$, and the convergence rate is $\mathcal{O}(1/\sqrt{n})$.

Numerical verification. We verify the asymptotic prediction given by the ODE (5) via numerical simulations under the settings stated in Example 1. The dimension is $n = 10,000$. After testing different combinations of the learning rates τ and $\widetilde{\tau}$, we have observed at least four nontrivial dynamical patterns. The results are shown in Figure 1. In all these experiments, our theoretical predictions match the actual trajectories of the macroscopic states very well. We also run the same experiments for WGAN as described in Example 2 and have observed similar dynamical patterns. In Section 4, we will present a detailed analysis to quantify the conditions under which each distinctive dynamical pattern emerges.

3.2 Microscopic dynamics

In this section, we study how the elements in $\boldsymbol{X}_k = [\boldsymbol{u}, \widetilde{\boldsymbol{w}}_k, \boldsymbol{w}_k]$ evolve during the training process. Unlike the macroscopic state $\boldsymbol{m}_k$, which only has 5 degrees of freedom, the matrix $\boldsymbol{X}_k$ contains $n \times 3$ elements. One should not expect a precise prediction of each element. Instead, we study the evolution of the *empirical measure* of the microscopic states, which is defined as

$$\mu_k(U, \widetilde{W}, W) \overset{\text{def}}{=} \tfrac{1}{n}\textstyle\sum_{i=1}^n \delta(U - \sqrt{n}u_i, \widetilde{W} - \sqrt{n}\widetilde{w}_{k,i}, W - \sqrt{n}w_{k,i}),$$

where $\delta(\cdot, \cdot, \cdot)$ is a 3-D Dirac measure, and u_i, $\widetilde{w}_{k,i}$ and $w_{k,i}$ denote the ith element of the vectors $\boldsymbol{u}$, $\widetilde{\boldsymbol{w}}_k$ and $\boldsymbol{w}_k$ respectively. The scaling factor $\sqrt{n}$ in the Dirac measures is introduced because u_i,

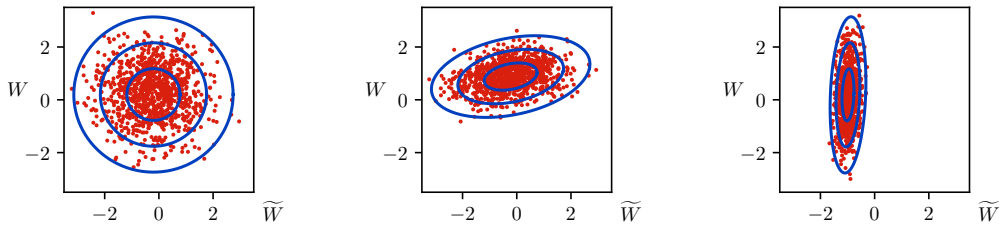


Figure 2: The evolution of the microscopic states. From left to right, we consider $t = 0$, 10, and 150. For each fixed t, the red points in the corresponding figure represent the values of $(\sqrt{n}\widetilde{w}_{k,i}, \sqrt{n}w_{k,i})$ for $i = 1, 2, \ldots, n$, where $k = \lfloor nt \rfloor$. The blue ellipses illustrate the contours corresponding to one, two, and three standard deviations of the 2-D Gaussian distribution given in (7).

$\widetilde{w}_{k,i}$ and $w_{k,i}$ are $\mathcal{O}(1/\sqrt{n})$ quantities. We also embed the discrete-time measure-valued stochastic process μ_k into a continuous-time process by defining $\mu_t^{(n)} \overset{\text{def}}{=} \mu_k(U, \widetilde{W}, W)$ with $k = \lfloor nt \rfloor$.

Following the general technical approach presented in [17, 19], we can show that the empirical measure μ_k converges weakly to a deterministic measure-valued process. Formally, under the same assumptions as Theorem 1, the sequence of measure-valued process $\{\{\mu_t^{(n)}\}_{t\in[0,T]}\}_n$ converges weakly to a deterministic process $\{\mu_t\}_{t\in[0,T]}$, which is the unique solution of the following PDE (given in its weak form): for any bounded test function $\varphi(U, \widetilde{W}, W) \in \mathcal{C}^3$,

$$\begin{aligned}\tfrac{\mathrm{d}}{\mathrm{d}t}\langle \mu_t, \varphi(U, \widetilde{W}, W)\rangle = \tfrac{\tau^2}{2} V(q_t, r_t, z_t)\langle \mu_t, \tfrac{\partial^2}{\partial \widetilde{W}^2}\varphi\rangle + \langle \mu_t, \widetilde{\tau}\big(\widetilde{G}(r_t, z_t)W + \widetilde{J}(\widetilde{z}_t)\widetilde{W}\big)\tfrac{\partial}{\partial \widetilde{W}}\varphi\rangle \\ + \big\langle \mu_t, \tau\big(G(q_t, z_t)U - \widetilde{G}(r_t, z_t)\widetilde{W} + J(q_t, r_t, z_t)W\big)\tfrac{\partial}{\partial W}\varphi\big\rangle,\end{aligned} \tag{6}$$

where $[\widetilde{q}_t\ \widetilde{z}_t\ q_t\ z_t\ r_t] = \boldsymbol{m}(t)$ is the solution of the ODE in Theorem 1, and $\langle \mu_t, \cdot\rangle$ denotes the expectation with respect to the measure μ_t. The definitions of the functions V, G, $\widetilde{G}$, J, and $\widetilde{J}$, and the formal derivation of (6) are presented in the Supplementary Materials [27]. We refer readers to [17] for a general framework for rigorously establishing the above scaling limit.

Numerical verification. We verify the predictions given by the PDE (6) using a special choice of the target feature vector $\boldsymbol{u}$ whose elements are all 1's. We also set the initial condition $P_0(\widetilde{W}, W|U = 1)$ to be a Gaussian distribution. In this case, the PDE (6) admits a particularly simple analytical solution: at any time t, the solution $P_t(\widetilde{W}, W|U = 1)$ is a Gaussian distribution whose mean and covariance matrix are given by

$$\mathbb{E}_{P_t(\widetilde{W},W|U=1)}\begin{bmatrix}\widetilde{W}\\ W\end{bmatrix} = \begin{bmatrix}\widetilde{q}_t\\ q_t\end{bmatrix} \quad \text{and} \quad \mathbb{E}_{P_t(\widetilde{W},W|U=1)}\begin{bmatrix}\widetilde{W}\\ W\end{bmatrix}\begin{bmatrix}\widetilde{W} & W\end{bmatrix} = \begin{bmatrix}\widetilde{z}_t & r_t\\ r_t & z_t\end{bmatrix}. \tag{7}$$

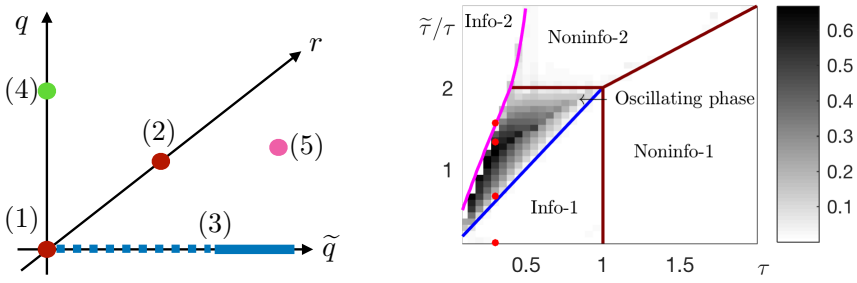
Figure 2 overlays the contours of the probability distribution $P_t(\widetilde{W}, W|U = 1)$ at different times t over the point clouds of the actual experiment data $(\sqrt{n}w_{k,i}, \sqrt{n}\widetilde{w}_{k,i})$. We can see that the theoretical prediction given by (6) has excellent agreement with simulation results.

4 Local Stability Analysis and Phase Diagram

In this section, we study how the learning rates τ and $\widetilde{\tau}$ affect the performance of the training algorithm. In what follows, we focus on the WGAN model as described in Example 2, but the phenomena and the conclusions we reach can be generalized to other cases.

In order to further reduce the degrees of freedom of the ODE (5), we let the regularization parameter $\lambda \to \infty$. In this case, the vectors $\widetilde{\boldsymbol{w}}_k, \boldsymbol{w}_k$ are always normalized and thus $z_k = \widetilde{z}_k = 1$. The macroscopic state is then described by only three scalars q_k, $\widetilde{q}_k$ and r_k. Correspondingly, the ODE in Theorem 1 reduces to a simpler form:

$$\begin{cases}\frac{d}{dt}\widetilde{q}_t &= \widetilde{\tau}\widetilde{\sigma}^2 r_t\,(q_t - r_t\widetilde{q}_t)\\ \frac{d}{dt}q_t &= \tau\big[\sigma^2 - \tau - \sigma^2\big(1+\tfrac{\tau}{2}\big)q_t^2\big]q_t - \tau\widetilde{\sigma}^2\big[\widetilde{q}_t + \big(\tfrac{\tau}{2}-1\big)r_t q_t\big]r_t\\ \frac{d}{dt}r_t &= \tau\sigma^2\widetilde{q}_t q_t + r\big(\widetilde{\sigma}^2(\widetilde{\tau}-\tau) - \tau^2\big) + r_t^3\widetilde{\sigma}^2\big(\tau - \widetilde{\tau} - \tfrac{\tau^2}{2}\big) - r_t q_t^2\tau\sigma^2\big(1+\tfrac{\tau}{2}\big),\end{cases} \tag{8}$$



(a) (b)

Figure 3: (a) The locations of the five types of fixed points of the ODE (8). The properties of these fixed points are listed in Table 1. (b) The phase diagram for the stationary state of the ODE (8). The colored lines illustrate the theoretical prediction of the boundaries between the different phases. Simulations results for a single numerical experiment with $\sigma = \widetilde{\sigma} = 1$ are also shown to illustrate the oscillating phase: Each grey square represents the value of $\frac{1}{200}\int_{800}^{1000}[(q_t - \langle q_t\rangle)^2 + (\widetilde{q}_t - \langle \widetilde{q}_t\rangle)^2 + (r_t - \langle r_t\rangle)^2]\,\mathrm{d}t$ where $\langle q_t\rangle = \frac{1}{200}\int_{800}^{1000} q_t\,\mathrm{d}t$, and $\langle \widetilde{q}_t\rangle$ and $\langle r_t\rangle$ are defined similarly. Note that the above quantity measures the variation (over time) of the training process as it approaches steady states. We see that the variation is indeed nonzero in the oscillating phase (see also the middle two figures in Figure 1), whereas the variation is close to zero in all other phases.

Table 1: List of the fixed points of the ODE (8) when $\sigma = \widetilde{\sigma}$.

Type	Location	Existence	Stable Region	Intuitive Interpretation
1	$\widetilde{q} = q = 0$ $r = 0$	always	$\tau > \sigma^2$, $\frac{\widetilde{\tau}}{\tau} < \frac{\tau+\sigma^2}{\sigma^2}$	Both $\mathcal{G}$ and $\mathcal{D}$ fail, and they are uncorrelated
2	$\widetilde{q} = q = 0$ $r = \pm r^* \neq 0$	$\frac{\widetilde{\tau}}{\tau} \geq \frac{\tau+\sigma^2}{\sigma^2}$ or $\frac{\widetilde{\tau}}{\tau} \leq 1 - \frac{\tau}{2}$	$\max\{2, \frac{\tau+\sigma^2}{\sigma^2}\} \leq \frac{\widetilde{\tau}}{\tau} \leq g(\tau)$	Both $\mathcal{G}$ and $\mathcal{D}$ fail, and they are correlated
3	$q = r = 0$ $\lvert\widetilde{q}\rvert \in (0, 1]$	always	$\lvert\widetilde{q}\rvert = 1$ is stable if $\frac{\widetilde{\tau}}{\tau} \leq \min\{\frac{2\tau}{\sigma^2}, \max\{\frac{\tau^2\sigma^{-2}}{\lvert\tau-\sigma^2\rvert}, 4\}\}$	$\mathcal{G}$ wins and $\mathcal{D}$ loses
4	$\widetilde{q} = r = 0$ $q = \pm p^* \neq 0$	always	always unstable	$\mathcal{G}$ loses and $\mathcal{D}$ wins
5	None of $\widetilde{q}$, q or r is zero	not always, at most 8 fixed points	can be computed numerically	Both $\mathcal{G}$ and $\mathcal{D}$ are informative

where σ^2 and $\widetilde{\sigma}^2$ are the variance of the distributions P_c and $P_{\widetilde{c}}$, respectively. The details of the reduction from the general ODE (5) to (8) is presented in the Supplementary Materials [27].

In what follows, we compute the fixed points of the ODE (8), defined as the solutions of the equations $\frac{\mathrm{d}}{\mathrm{d}t}\widetilde{q}_t = \frac{\mathrm{d}}{\mathrm{d}t}q_t = \frac{\mathrm{d}}{\mathrm{d}t}r_t = 0$, and investigate their local stabilities. For simplicity, we only present the result when $\sigma = \widetilde{\sigma}$. Even in this simplest case, there are 5 types of fixed points, the locations of which are visualized in the 3-dimensional space $(\widetilde{q}, q, r)$ shown in Figure 3(a). Each type of the fixed points has an intuitive meaning in terms of the two-player game between $\mathcal{G}$ and $\mathcal{D}$. We list the detailed information in Table 1, in which we also define a function $g(\tau) = \begin{cases}[1 + (\frac{\sigma^2}{2} - \frac{\sigma^2}{\tau})^{-1}]^{-1}, & \text{if } \tau \leq \frac{2\sigma^2}{\sigma^2+2} \\ +\infty, & \text{otherwise}\end{cases}$, which is needed in specifying the stable region for each type of fixed points.

Phase diagram. By analyzing the local stabilities of these fixed points, we obtain the phase diagram as shown in Figure 3(b). (The details can be found in the Supplementary Materials [27].) In particular, three major phases are identified under different settings of the learning rates τ and $\widetilde{\tau}$.

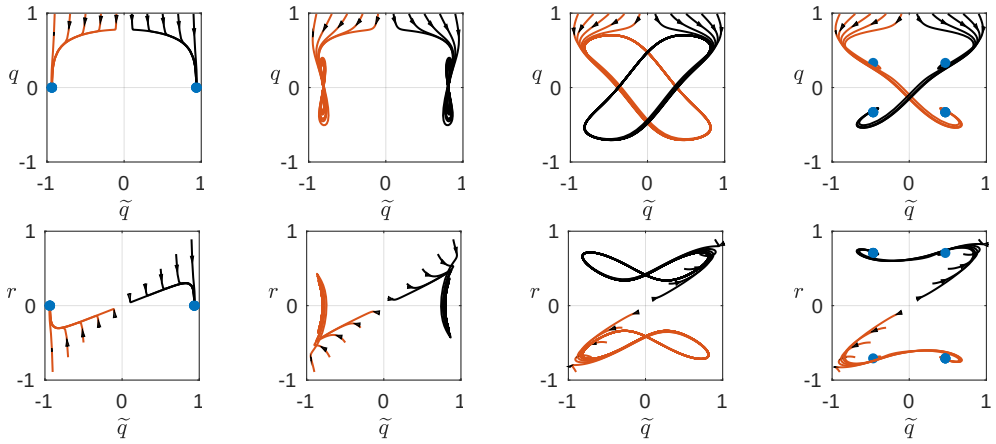


Figure 4: Phase portraits in the informative phase and the oscillating phase as projected onto the $\widetilde{q}$–q plane (top row) and the $\widetilde{q}$–r plane (bottom row). We set $\tau = 0.3$ and increase $\widetilde{\tau} = 0.03,\ 0.2,\ 0.4,\ 0.47$ from left to right. (These parameter settings are marked by the four red dots in the phase diagram in Figure 3(b).) The first column shows a case in the phase of info-1, where a subset of type (3) fixed points are stable. The second and third columns are in the oscillating phase. And the last column is in info-2, where the fixed points of type-5 are stable. The blue dots in the figures show the stable fixed points.

Noninformative phase: We say that the ODE (8) is in a noninformative phase if either a type-1 or type-2 fixed point in Table 1 is stable. In this case, $\widetilde{q}_t$ can be trapped at zero, which indicates that the generator's parameter vector $\widetilde{\boldsymbol{w}}$ has no correlation with the true feature vector $\boldsymbol{u}$. In Figure 4(b), the region labeled as noninfo-1 is the stable region for the type-1 fixed point, and noninfo-2 is the stable region for the type-2 fixed point. The two regions have no overlap. However, we note that in noninfo-1, the type-3 fixed points can also be stable, in which case the stationary point of the ODE is determined by the initial condition.

Informative phase: We say that the ODE (8) is in an informative phase if neither type-1 nor type-2 fixed point is stable, and if at least one fixed point of type-3 and type-5 is stable. In this case, it is guaranteed that $\widetilde{q}$ is nonzero, indicating that the generator can achieve non-vanishing correlation with the real feature vector. In addition, the stable regions for the type-3 and type-5 fixed points are disjoint. They are shown in Figure 4(b) as info-1 and info-2, respectively. The difference between the two region is that, in info-1, q is exactly 0 indicating that the discriminator is completely fooled, whereas in info-2, q is nonzero.

Oscillating phase: We say that the ODE (8) is in an oscillating phase if none of the fixed points in Table 1 is stable. In this phase, limiting cycles emerge and the system will oscillate on these cycles indefinitely. Moreover, we found two types of limiting cycles.

To further illustrate the phase transitions, we draw 4 phase portraits in Figure 4 corresponding to different choices of the step sizes. The figures in the middle two columns show the two types of limiting cycles that can emerge in the oscillating phase.

5 Conclusion

We present a simple high-dimensional model for GAN with an exactly analyzable training process. Using the tool of scaling limits of stochastic processes, we show that the macroscopic state associated with the training process converges to a deterministic process characterized as the unique solution of an ODE, whereas the microscopic state remains stochastic and can be described by a limiting PDE. Although our analysis is carried out in the asymptotic setting, numerical experiments show that our theoretical predictions can accurately capture the actual performance of the training algorithm at moderate dimensions. Our analysis also reveals several different phases of the training process that highly depend on the choice of the learning rates. Despite its simplicity, the proposed model of GAN provides valuable insights that might prove useful in the study of more realistic models and more involved training algorithms.

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 System*, 2014, pp. 2672–2680.

[2] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein Generative Adversarial Networks," *Proceedings of The 34th International Conference on Machine Learning*, pp. 1–32, 2017.

[3] M. Lucic, K. Kurach, M. Michalski, S. Gelly, and O. Bousquet, "Are GANs Created Equal? A Large-Scale Study," *arXiv preprint arXiv:1711.10337*, 2017.

[4] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4681–4690.

[5] 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*, 2017, pp. 1125–1134.

[6] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative Adversarial Text to Image Synthesis," *33rd International Conference on Machine Learning*, pp. 1060–1069, 2016.

[7] S. Arora, R. Ge, Y. Liang, and Y. Zhang, "Generalization and Equilibrium in Generative Adversarial Nets," in *International Conference on Machine Learning*, 2017, pp. 224–232.

[8] M. Arjovsky and L. Bottou, "Towards Principled Methods for Training Generative Adversarial Networks," *arXiv preprint arXiv:1701.04862*, 2017.

[9] S. Feizi, C. Suh, F. Xia, and D. Tse, "Understanding GANs: the LQG Setting," *arXiv preprint arXiv:1710.10793*, 2017.

[10] L. Mescheder, S. Nowozin, and A. Geiger, "The numerics of GANs," in *Advances in Neural Information Processing Systems*, 2017, pp. 1823–1833.

[11] V. Nagarajan and J. Z. Kolter, "Gradient descent GAN optimization is locally stable," in *Advances in Neural Information and Processing Systems*, 2017, pp. 5591–5600.

[12] K. Roth, A. Lucchi, S. Nowozin, and T. Hofmann, "Stabilizing Training of Generative Adversarial Networks through Regularization," in *Advances in Neural Information Processing Systems*, 2017, pp. 2015–2025.

[13] L. Mescheder, A. Geiger, and S. Nowozin, "Which Training Methods for GANs do actually Converge?" *arXiv preprint arXiv:1801.04406*, 2018.

[14] 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*, 2017, pp. 6629–6640.

[15] J. Li, A. Madry, J. Peebles, and L. Schmidt, "Towards Understanding the Dynamics of Generative Adversarial Networks," *arXiv preprint arXiv:1706.09884*, 2017.

[16] C. K. Sønderby, J. Caballero, L. Theis, W. Shi, and F. Huszár, "Amortised MAP Inference for Image Super-resolution," *arXiv preprint arXiv:1610.04490*, 2016.

[17] C. Wang, J. Mattingly, and Y. M. Lu, "Scaling Limit: Exact and Tractable Analysis of Online Learning Algorithms with Applications to Regularized Regression and PCA," *arXiv preprint arXiv:1712.04332*, 2017.

[18] C. Wang and Y. M. Lu, "Online Learning for Sparse PCA in High Dimensions: Exact Dynamics and Phase Transitions," in *Information Theory Workshop (ITW), 2016 IEEE*, 2016, pp. 186–190.

[19] ——, "The Scaling Limit of High-Dimensional Online Independent Component Analysis," in *Advances in Neural Information Processing Systems*, 2017, pp. 6641–6650.

[20] C. Wang, Y. C. Eldar, and Y. M. Lu, "Subspace Estimation from Incomplete Observations: A High-Dimensional Analysis," *arXiv preprint arXiv:1805.06834*, 2018.

[21] G. O. Roberts, A. Gelman, and W. R. Gilks, "Weak convergence and optimal scaling of random walk Metropolis algorithms," *Annals of Applied Probability*, vol. 7, no. 1, pp. 110–120, 1997.

[22] D. Saad and S. A. Solla, "Exact Solution for On-Line Learning in Multilayer Neural Networks," *Phys. Rev. Lett.*, vol. 74, no. 21, pp. 4337–4340, 1995.

[23] M. Biehl and H. Schwarze, "Learning by on-line gradient descent," *Journal of Physics A*, vol. 28, no. 3, pp. 643–656, 1995.

[24] S. Mei, A. Montanari, and P.-M. Nguyen, "A Mean Field View of the Landscape of Two-Layers Neural Networks," *arXiv preprint*, p. arXiv:1804.06561, 2018.

[25] H. Kushner and G. G. Yin, *Stochastic approximation and recursive algorithms and applications*. Springer Science & Business Media, 2003, vol. 35.

[26] I. Johnstone and A. Lu, "On consistency and sparsity for principal components analysis in high dimensions," *Journal of the American Statistical Association*, vol. 104, no. 486, pp. 682–693, 2009.

[27] C. Wang, H. Hu, and Y. M. Lu, "Supplementary materials: A solvable high-dimensional model of GAN," 2018. [Online]. Available: https://lu.seas.harvard.edu/files/yuelu/files/gan-a-solvable-model-supplementary-materials.pdf