Functional Space Analysis of Local GAN Convergence

Valentin Khrulkov [1] **Artem Babenko** [1 2] **Ivan Oseledets** [3]

Abstract

Recent work demonstrated the benefits of studying continuous-time dynamics governing the GAN training. However, this dynamics is analyzed in the model parameter space, which results in finite-dimensional dynamical systems. We propose a novel perspective where we study the local dynamics of adversarial training in the general functional space and show how it can be represented as a system of partial differential equations. Thus, the convergence properties can be inferred from the eigenvalues of the resulting differential operator. We show that these eigenvalues can be efficiently estimated from the target dataset before training. Our perspective reveals several insights on the practical tricks commonly used to stabilize GANs, such as gradient penalty, data augmentation, and advanced integration schemes. As an immediate practical benefit, we demonstrate how one can a priori select an optimal data augmentation strategy for a particular generation task.

1. Introduction

Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) allow for efficient learning of complicated probability distributions from samples. However, training such models is notoriously complicated: the dynamics can exhibit oscillatory behavior, and the convergence can be very slow. To alleviate this, a number of practical tricks, including the usage of data augmentation (Karras et al., 2020a; Zhao et al., 2020; Zhang et al., 2020) and optimization methods inspired by numerical integration schemes for ODEs (Qin et al., 2020) have been proposed. Nevertheless, the theory explaining the success of these methods is only starting to catch up with practice.

The standard way of training a GAN is to use simultaneous gradient descent. In Nagarajan & Kolter (2017) it was shown that under mild assumptions, this method is locally convergent. The authors Mescheder et al. (2018) demonstrated that when these assumptions are not satisfied, the usage of regularization methods such as the gradient penalty (Roth et al., 2017) or consensus optimization (Mescheder et al., 2017) is required to achieve convergence. In Balduzzi et al. (2018); Nie & Patel (2020); Liang & Stokes (2019) the local convergence is studied based on the eigenvalues of the Jacobian of the dynamics near the equilibrium. Such analysis involves the parameterization of the generator and the discriminator by neural networks and usually does not take into account the properties of the target distribution.

Our key idea is that this dynamics can be efficiently analyzed in the *functional space*. This is achieved by constructing a local quadratic approximation of the GAN training objective and writing down the dynamics as a system of partial differential equations (PDEs). The differential operator underlying this system has a remarkably simple form, and its spectrum can be analyzed in terms of the fundamental properties of the target distribution. Furthermore, we show that the convergence of the dynamics is determined by the *Poincaré constant* of this distribution. Intuitively, this constant describes the connectivity properties of a distribution; for instance, it is smaller for multimodal datasets with disconnected modes, where GAN convergence is known to be slower. This connection is practically important since the Poincaré constant of a dataset can be easily estimated a priori, and the larger it is, the better is the expected convergence of GAN. Thus, we can analyze common techniques that alter the train set in terms of their effect on the Poincaré constant. For instance, one can choose a proper augmentation strategy that increases the Poincaré constant the most. The main contributions of our paper are:

- We develop a linearized GAN training model in the functional space in the form of a PDE.
- We derive explicit formulas connecting the eigenfunctions and eigenvalues of the resulting PDE operator with the target distribution properties. This connection provides a theoretical justification for common GAN stabilization techniques.
- We describe an efficient, practical recipe for choosing optimal parameters for gradient penalty and data

[1] Yandex, Russia [2] National Research University Higher School of Economics, Moscow, Russia [3] Skolkovo Institute of Science and Technology, Moscow, Russia. Correspondence to: Valentin Khrulkov <khrulkov.v@gmail.com>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021. Copyright 2021 by the author(s).

augmentations for a particular dataset.

2. Second-order approximation of the GAN objective

We assume that data samples $\{X_i\}_{i=1}^N \in \mathbb{R}^d$ are produced by a target probability measure μ. The generator function $G(z)$ is a deterministic mapping from a latent space $\mathbb{R}^l$ to $\mathbb{R}^d$; z is sampled from a known probability measure μ_z. The discriminator D is a real-valued function on $\mathbb{R}^d$ which goal is to distinguish between real and fake samples. The GAN objective can be written as a min-max problem

$$\max_G \min_D f(D, G), \tag{1}$$

where

$$f(D, G) = \mathbb{E}_{x \sim \mu}\, \phi_1(D(x)) + \mathbb{E}_{z \sim \mu_z}\, \phi_2(D(G(z))), \tag{2}$$

and ϕ_1, ϕ_2 are some scalar convex twice differentiable functions. For example, for the LSGAN (Mao et al., 2017)

$$\phi_1(x) = \frac{1}{2}(x-1)^2, \quad \phi_2(x) = \frac{1}{2}(x+1)^2.$$

We analyze the behaviour of a GAN in a neighborhood of the Nash equilibrium corresponding to an optimal discriminator function D_* and an optimal generator function G_*. We make standard assumptions (Nagarajan & Kolter, 2017; Mescheder et al., 2018; Nie & Patel, 2020) that the optimal discriminator satisfies $D_*(x) = 0\ \forall x$ and the measure produced by G_* is equal to μ. Similarly, we also assume that $\phi_1''(0) + \phi_2''(0) \geq 0$ and $\phi_1'(0) = -\phi_2'(0) \neq 0$. In this setting we can recover many common GAN variations including the vanilla GAN (Goodfellow et al., 2014), WGAN (Arjovsky et al., 2017), LSGAN (Mao et al., 2017).

The standard way of solving the min-max problem is to represent D and G by some parametric models and solve it in the parameter space. The convergence of the resulting dynamics can be studied by linearizing it around the equilibrium point (Nagarajan & Kolter, 2017; Mescheder et al., 2018; Nie & Patel, 2020). This is equivalent to the construction of the quadratic approximation of $f(D, G)$ in the parameter space. In this paper, we do the quadratic approximation first, obtaining a new saddle point problem in a *functional* space that approximates the local dynamics of standard GAN training methods. This representation has a remarkably simple form, and its properties can be studied in detail.

Technical assumptions. We assume that μ is a measure with positive density $\rho > 0$ on $\mathbb{R}^d$, i.e., $d\mu = \rho(x)dx$. In practice, this can be (formally) achieved by smoothing a discrete data distribution with Gaussian noise. As the functional space, we consider $L_2(\mu) = L_2(\mu, \mathbb{R}^d)$ equipped with a natural weighted inner product:

$$\langle u_1, u_2 \rangle_\mu = \int u_1(x) u_2(x) d\mu.$$

We will also often utilize differential operators, which should be understood in the distributional sense. In these cases, the functions under study are assumed to be the elements of the corresponding Sobolev space. Specifically, we will utilize the weighted Sobolev spaces $H^1(\mu)$ and $H^2(\mu)$. These are spaces of generalized functions such that their distributional first order and second order derivatives respectively lie in $L_2(\mu)$.

We start by finding the second-order approximation to the GAN objective near Nash equilibrium in the functional space. This process is analogous to the study of nonlinear (finite-dimensional) ODEs near an equilibrium point via linearization. We assume that G_* is invertible, i.e., for any $x \sim \mu$ there exists a unique $z \sim \mu_z$ such that $G_*(z) = x$. Let $(\delta D, \delta G)$ be the *functional variations* of the optimal discriminator and generator respectively. Informally, this means that δD and δG are sufficiently small perturbations of the equilibrium. Let $\alpha = \frac{1}{2}\left(\phi_1''(0) + \phi_2''(0)\right), \beta = \phi_2'(0)$ and $f_0 = f(D_*, G_*)$.

Theorem 1. *Let $D = D_* + \delta D = \delta D, G = G_* + \delta G$. Let us denote $u(x) = \delta D(x), v(x) = (\delta G)(G_*^{-1}(x))$. We assume $u \in H^2(\mu)$ and $v \in H^1(\mu)$. Then, (2) can be approximated as*

$$f(D, G) = f_0 + g(u, v) + R_3(u, v),$$

where

$$g(u, v) = \alpha \langle u, u \rangle_\mu + \beta \langle \nabla_x u, v \rangle_\mu,$$

and

$$R_3(u, v) = O\left(\left(\|u\|_{H^2(\mu)} + \|v\|_{H^1(\mu)}\right)^3\right)$$

Proof. See Appendix A for the proof. □

From Theorem 1 it follows that the local quadratic approximation to (2) is provided by $g(u, v)$.

3. Continuous gradient descent-ascent in the functional space

Now we need to solve the min-max problem of the form

$$\min_u \max_v g(u, v), \quad g(u, v) = \alpha \langle u, u \rangle_\mu + \beta \langle \nabla_x u, v \rangle_\mu. \tag{3}$$

We will refer to (3) as *linearized GAN problem* (lGAN). Since it is a local quadratic approximation of the original min-max problem, local behavior of GAN training methods can be understood on this task. The linearized GAN problem depends solely on the measure μ, and we will see what are the particular fundamental properties of this measure that determine the convergence of the gradient-based method for solving (3). The continuous descent-ascent flow in the functional space can be written as:

$$u_t = -\nabla_u g(u, v), v_t = \nabla_v g(u, v), \tag{4}$$

where $\nabla_u g$ and $\nabla_v g$ are Fréchet derivatives of the functional $g(\cdot, \cdot)$ with respect to u and v. It is important to define what is the meaning of (4), i.e., it should be understood in the weak sense. Select a test function $\hat{u}$ and take the scalar product of the first equation with it. We get

$$\langle u_t, \hat{u}\rangle_\mu = -\langle \nabla_u g(u, v), \hat{u}\rangle_\mu, \, \forall \hat{u} \in H^2(\mu) \tag{5}$$

Similar equation holds for v. The evaluation of the right-hand side is done by integration by parts, which leads to the system of the form

$$\begin{aligned} &\langle u_t, \hat{u}\rangle_\mu = -\alpha\langle u, \hat{u}\rangle_\mu - \beta\langle \nabla_x \hat{u}, v\rangle_\mu, \, \forall \hat{u} \in H^2(\mu). \\ &\langle v_t, \hat{v}\rangle_\mu = \beta\langle \nabla_x u, \hat{v}\rangle_\mu, \, \forall \hat{v} \in H^1(\mu). \end{aligned} \tag{6}$$

By making use of the density ρ, equations (6) can be rewritten in the strong form as

$$\begin{aligned} \rho u_t &= -\alpha\rho u + \beta\nabla_x \cdot (\rho v), \\ \rho v_t &= \rho\beta\nabla_x u. \end{aligned} \tag{7}$$

We are interested in the asymptotic behavior of u and v when t goes to infinity. It is completely described by the eigenvalues of the operator on the right-hand side of (6). We will say that (u_λ, v_λ) is an eigenfunction with an eigenvalue λ if it satisfies the following system of equations.

$$\begin{aligned} &\lambda\langle u_\lambda, \hat{u}\rangle_\mu = -\alpha\langle u_\lambda, \hat{u}\rangle_\mu - \beta\langle \nabla_x \hat{u}, v_\lambda\rangle_\mu, \, \forall \hat{u} \in H^2(\mu). \\ &\lambda\langle v_\lambda, \hat{v}\rangle_\mu = \beta\langle \nabla_x u_\lambda, \hat{v}\rangle_\mu, \forall \hat{v} \in H^1(\mu). \end{aligned} \tag{8}$$

4. Eigenfunctions and eigenvalues of the linearized operator

If (u_λ, v_λ) form an eigenfunction, then the solution of the time-dependent problem (6) with the initial conditions $(u(0) = u_\lambda, v(0) = v_\lambda)$ can be written as

$$u(t) = u_\lambda \exp(\lambda t), \quad v(t) = v_\lambda \exp(\lambda t). \tag{9}$$

We will see that $\{(u_\lambda, v_\lambda)\}_\lambda$ form a basis in our functional space and thus an arbitrary solution can be written as a series of terms (9). Thus, the real parts of the spectrum of the operator in (6) should be less than 0 in order for the system to have asymptotic stability. We will demonstrate that our system does not have eigenvalues with a positive real part but has a naturally interpretable kernel.

4.1. The kernel.

We find that the kernel has the following form.

Corollary 1. *Let (u_0, v_0) be an eigenfunction with $\lambda = 0$. Then,*

$$u_0 = C, \quad \langle \nabla_x \hat{u}, v_0\rangle_\mu = 0, \, \forall \hat{u} \in H^2(\mu), \tag{10}$$

or in the strong form:

$$u_0 = C, \nabla_x \cdot (\rho v_0) = 0. \tag{11}$$

Here C is a constant such that $C\alpha = 0$. I.e., for $\alpha \neq 0$ we get $C = 0$, and $C \in \mathbb{R}$ otherwise.

Proof. See Appendix A for the proof. □

This kernel has a straightforward interpretation. If we add a function v_0 that satisfies (10) to the G_*, the mapped measure will be the same up to second order terms. Indeed, consider (10) in the strong form. We obtain that $\nabla_x \cdot (\rho v_0) = 0$. Recall that the function $x + v_0(x) = x + \delta G(G_*^{-1}(x))$ maps a sample x to a sample from the synthetic density produced by $G_* + \delta G$, i.e. $v_0(x)$ can be considered as a *velocity* of each sample. If samples evolve with velocity $v_0(x)$, the differential equation for the density takes the form

$$\rho_t = \nabla_x \cdot (\rho v_0) = 0,$$

i.e., the density is invariant under such transformation. We will refer to the condition (10) as the *divergence-free* condition on v_0.

4.2. Non-zero spectrum.

4.2.1. Weighted Laplace operator

We now address non-zero eigenvalues of our operator. It will be convenient to utilize the *weighted Laplace operator* Δ_μ which is defined (in the weak form) as follows.

$$\langle \Delta_\mu w, \hat{w}\rangle_\mu := -\langle \nabla_x w, \nabla_x \hat{w}\rangle_\mu, \, \forall \hat{w} \in H^2(\mu). \tag{12}$$

In the strong form Δ_μ takes the following form:

$$\Delta_\mu w = \frac{1}{\rho}\nabla_x \cdot (\rho\nabla_x w), \tag{13}$$

which results in the standard Laplacian in the case of the standard Lebesgue measure on $\mathbb{R}^d$. This operator commonly appears in the study of the diffusion processes (Coifman & Lafon, 2006) and weighted heat equations (Grigoryan, 2009).

Spectrum of the weighted Laplacian. In what follows we will use eigenvalues and eigenfunctions of Δ_μ. Firstly, this a self-adjoint non-positive definite operator and under mild assumptions on μ (Cianchi et al., 2011; Grigoryan, 2009), e.g., when ρ decreases sufficiently fast, it has a discrete spectrum. Let us study it in more detail. Consider the eigenvalue problem for $-\Delta_\mu$ written in the weak form.

$$\langle \nabla_x \hat{w}, \nabla_x w_\xi \rangle_\mu = \xi \langle \hat{w}, w_\xi \rangle_\mu \quad \forall \hat{w} \in H^2(\mu). \tag{14}$$

We note that $w_0 \equiv 1$ is the eigenfunction with $\xi = 0$; thus, due to self-adjointness of Δ_μ for every other eigenfunction w_ξ we have $\langle 1, w_\xi \rangle_\mu = 0$, i.e., it has zero mean with respect to μ.

The Poincaré constant of μ. Let us consider the smallest *non-zero* eigenvalue $\xi_{\min}$ of $-\Delta_\mu$. We obtain that the following inequality holds.

$$\langle \nabla_x w, \nabla_x w \rangle_\mu \geq \xi_{\min} \langle w, w \rangle_\mu, \langle 1, w \rangle_\mu = 0, \forall w \in H^2(\mu), \tag{15}$$

and the exact minimizer of this inequality is achieved by $w_{\xi_{\min}}$ (Grigoryan, 2009). The value of $\xi_{\min}$ (sometimes its inverse) is called the *Poincaré constant* of the measure μ and often appears in Sobolev-type inequalities (Adams & Fournier, 2003). The exact values of $\xi_{\min}$ for a given measure are almost never known analytically. To make use of our results in practical settings, we propose a simple neural network based approach for estimation of $\xi_{\min}$. We discuss it in Section 6.

What is it exactly? We can provide an intuitive meaning of $\xi_{\min}$ by drawing an analogy with graph Laplacians. In this case, the second smallest eigenvalue of the graph Laplacian, called the Fiedler value (Fiedler, 1973), reflects how well connected the overall graph is. Thus, the Poincaré constant is in a way a continuous analog of this constant, reflecting the "connectivity" of a measure. This is the property that we would expect to impact the GAN convergence, as based on rich empirical evidence, the datasets that are more "disconnected" (such as ImageNet) are very challenging to model. We provide experimental evidence in support of this intuitive explanation in Section 7.

4.2.2. Spectrum of lGAN

We are now ready to fully describe the spectrum of our lGAN model. Let $\{\xi_i\}_{i=1}^\infty$ be the non-zero spectrum of $-\Delta_\mu$ and $\{w_{\xi_i}\}_{i=1}^\infty$ be the set of the corresponding eigenfunctions. Recall from Theorem 1 that the constants α and β are defined purely in terms of functions ϕ_1 and ϕ_2. We now state our main result. For simplicity we assume that all the eigenvalues are of multiplicity one.

Theorem 2. *The non-zero spectrum of* (8) *is described as follows.*

- *The eigenvalues are given by $\{\lambda_i^\pm\}_{i=1}^\infty$ where $\lambda_i^\pm$ are roots of the quadratic equation:*

$$\lambda^2 + \alpha\lambda + \beta^2 \xi_i = 0. \tag{16}$$

- *The corresponding eigenfunctions are written in terms eigenfunctions of $-\Delta_\mu$ as follows.*

$$(u_{\lambda_i^\pm}, v_{\lambda_i^\pm}) = (w_{\xi_i}, \frac{\beta}{\lambda_i^\pm} \nabla_x w_{\xi_i}). \tag{17}$$

Proof. See Appendix A for the proof. □

Theorem 3. *Let (u_λ, v_λ) be the eigenfunction with eigenvalue λ (see* (8)*) and $\lambda \neq 0$. Then u_λ is an eigenfunction of the* weighted Laplace operator $-\Delta_\mu$, *defined as the solution of the eigenvalue problem*

$$\langle \nabla_x \hat{u}, \nabla_x u \rangle_\mu = \xi \langle u, \hat{u} \rangle_\mu, \forall \hat{u} \in H^2(\mu), \tag{18}$$

and $v_\lambda = \frac{\beta}{\lambda} \nabla_x u_\lambda$. The eigenvalue λ satisfies the quadratic equation

$$\lambda^2 + \lambda\alpha + \beta^2 \xi = 0, \tag{19}$$

Proof. See Appendix A for the proof. □

With the explicit formulas for eigenvalues and eigenvectors at hand, we are now ready to analyze the convergence of the problem (6).

5. Convergence

The following theorem expresses the solution in terms of the eigenfunctions and also provides the convergence estimates.

Theorem 4. *Let $u_0 \in H^2(\mu), v_0 \in H^1(\mu)$ and $\int u_0 d\mu = c_0$. Then, these functions can be written as*

$$u_0 = c_0 + \sum_{k=1}^\infty (c_k^+ + c_k^-) w_{\xi_k},$$

$$v_0 = \widetilde{v}_0 + \nabla_x V_0, \; V_0 = \sum_{k=1}^\infty \left(c_k^+ \frac{\beta}{\lambda_i^+} + c_k^- \frac{\beta}{\lambda_i^-} \right) w_{\xi_k},$$

and $\widetilde{v}_0$ is divergence-free, i.e. $\langle \nabla_x \hat{u}, \widetilde{v}_0 \rangle_\mu = 0$. The coefficients c_k^+ and c_k^- can be obtained as the solution of the linear systems:

$$\begin{pmatrix} 1 & 1 \\ \frac{\beta}{\lambda_i^+} & \frac{\beta}{\lambda_i^-} \end{pmatrix} \begin{pmatrix} c_k^+ \\ c_k^- \end{pmatrix} = \begin{pmatrix} \langle u_0, w_{\xi_k} \rangle_\mu \\ \langle V_0, w_{\xi_k} \rangle_\mu \end{pmatrix} \tag{20}$$

With this expansion, the solution to (6) *is*

$$\begin{aligned} u(t) &= c_0 e^{-\alpha t} + \sum_{k=1}^\infty w_{\xi_k} \left(c_k^+ e^{\lambda_k^+ t} + c_k^- e^{\lambda_k^- t} \right), \\ v(t) &= \widetilde{v}_0 + \nabla_x V(t), \\ V(t) &= \sum_{k=1}^\infty w_{\xi_k} \left(c_k^+ \frac{\beta}{\lambda_i^+} e^{\lambda_k^+ t} + c_k^- \frac{\beta}{\lambda_i^-} e^{\lambda_k^- t} \right). \end{aligned} \tag{21}$$

For $\alpha > 0$ the norms of $u(t)$ and $V(t)$ can be estimated as

$$\|u(t)\|_\mu \leq 2\|u_0\|_\mu e^{\eta t}, \|V(t)\|_\mu \leq C\|V_0\|_\mu e^{\eta t},$$

where $\eta = \mathrm{Re}\Big(\frac{-\alpha+\sqrt{\alpha^2-4\beta^2\xi_{\min}}}{2}\Big) < 0$ *is the maximal real part of the eigenvalues.*

Proof. See Appendix A for the proof. □

Discussion. Theorem 4 provides the exact representation of the solution in terms of the eigenfunctions of Δ_μ. The eigenvalues are given as the solution of the quadratic equation (19), thus the spectral properties of $-\Delta_\mu$ completely determine the dynamics of the convergence. Specifically, we observe that the speed of convergence is determined by the value of $\xi_{\min}$, i.e., the lowest non-zero mode of the weighted Laplacian.

There are two distinct cases. If the first eigenvalue of $-\Delta_\mu$ satisfies $\alpha^2 - 4\beta^2\xi_{\min} \leq 0$, (i.e., $\xi_{\min}$ is large enough), then all the eigenvalues $\lambda_i^\pm$ are complex, and their real part is equal to $-\frac{\alpha}{2}$. Ideally, we would have $\alpha^2 - 4\beta^2\xi_{\min} = 0$, which would provide us with the optimal convergence speed and no oscillations for the highest mode. Consider now the case of a small $\xi_{\min} \ll \frac{\alpha^2}{4\beta^2}$. In this case, η will be close to 0, which would result in a slow convergence rate. These observations can be used to explain the success of various practical GAN training methods, as we show in Section 7.

In the case is $\alpha = 0$, which holds, for instance, for WGAN, we obtain the well-known purely oscillatory behavior (Nagarajan & Kolter, 2017). We also note that for $\alpha > 0$, the average value of the discriminator exponentially decays to 0. This resembles the convergence plots of state-of-the-art GANs, see, e.g., Karras et al. (2020a, Figure 6b).

In Appendix B we discuss a concrete application of Theorem 4 to the case of $\mu \sim \mathcal{N}(0, 1)$. The spectrum of $-\Delta_\mu$ can be found explicitly and is given by $\mathbb{Z}_{\geq 0}$.

What about neural networks? The discussion so far focuses on the functional spaces. In reality, these functions are approximated by neural networks, and the dynamics is written in the parameters θ_D and θ_G of these networks. Local convergence analysis of such dynamics is possible in these parameters (Mescheder et al., 2018; Nie & Patel, 2020); however, such analysis involves eigenvalues of the Jacobian of the loss. It is not easy to connect these properties to the fundamental properties of the measure. The functional space analysis shows this connection. The approximation by neural networks (or by any other parametric representation) can be thought of as a *spatial discretization* of PDEs. If the number of parameters increases, the eigenvalues of the discretized problem should approximate the eigenvalues of the infinite-dimensional problem. It is worth noting that different discretizations (for example, different neural network architectures) may lead to different properties. One can compare such discretizations by looking at the quality of approximation of the eigenvalues of the weighted Laplace operator (see Section 6 for the algorithmic details). Also, such techniques used in practice as Jacobian regularization (Karras et al., 2020b) can be considered as methods for choosing a better-conditioned discretization. Finally, for future research, we would like to mention that the lGAN problem is similar to the saddle point problems (Benzi et al., 2005) appearing in mathematical modeling of fluid problems. For example, for robust discretization, one has to choose discretization spaces of u ('pressure'), and v ('velocity') such that they satisfy the famous LBB (Ladyzhenskaya-Babuška-Brezzi) condition (Boffi et al., 2013; Ladyzhenskaya, 1969) in order to get good convergence properties. This task requires systematic study, and we leave it for future research.

6. Effects of common practices on the asymptotic convergence

Regularization. Several studies have shown that if we penalize the norm of the gradient of the GAN objective with respect to the discriminator, it improves the convergence (Gulrajani et al., 2017; Mescheder et al., 2018). In our case, it results in an additional term $\frac{\gamma}{2}E_\mu\|\nabla_D f(D,G)\|^2$. After linearization we obtain a regularized loss $\hat{g}(u, v)$.

$$\max_v \min_u \hat{g}(u,v), \quad \hat{g}(u,v) = g(u,v) + \frac{\gamma}{2}\,\mathbb{E}_\mu\,\|\nabla_x u\|^2.$$

New eigenvalue problem has the form

$$\lambda\langle u_\lambda, \hat{u}\rangle_\mu = -\alpha\langle u_\lambda, \hat{u}\rangle_\mu - \beta\langle\nabla_x\hat{u}, v_\lambda\rangle_\mu - \gamma\langle\nabla_x u, \nabla_x\hat{u}\rangle,$$
$$\lambda\langle v_\lambda, \hat{v}\rangle_\mu = \beta\langle\nabla_x u_\lambda, \hat{v}\rangle_\mu, \forall\hat{v} \in H^1(\mu),\ \hat{u} \in H^2(\mu).$$

The kernel stays the same, and moreover, the u_λ component is also the same, which can be seen again by using $\hat{v} = \nabla_x\hat{u}$ in the second equation. The only difference is the connection between eigenvalues of the weighted Laplace operator and eigenvalues of the linearized problem, which changes as follows:

$$\lambda^2 + (\alpha + \gamma\xi_i)\lambda + \beta^2\xi_i = 0,$$

yielding

$$\lambda_i^\pm = \frac{-(\alpha+\gamma\xi_i) \pm \sqrt{(\alpha+\gamma\xi_i)^2 - 4\beta^2\xi_i}}{2}. \qquad (22)$$

Optimal parameters. One of the most interesting conclusions of our analysis is that the convergence is determined by the connection between α, β, γ and ξ. If we fix ϕ_1, ϕ_2 in advance, we can not control α and β and convergence for a particular dataset will be determined by $\xi_{\min}$. I.e., for one dataset the loss may work well, but for another, it may fail.

An alternative approach is to optimize over the hyperparameters of the loss and regularizer taking the Poincaré constant into account. Expression (22) gives a direct way to do that. The eigenvalue with the maximal real part corresponds to $\xi = \xi_{\min}$ and is given by the formula

$$\lambda_{\max} = \frac{-(\alpha + \gamma\xi_{\min}) + \sqrt{(\alpha + \gamma\xi_{\min})^2 - 4\beta^2\xi_{\min}}}{2}.$$

The maximum is obtained if the discriminant is equal to 0, which gives

$$\alpha + \gamma\xi_{\min} = 2|\beta|\sqrt{\xi_{\min}}. \tag{23}$$

Another desirable property is the absence of oscillations. This is only possible if the quadratic function under the square root is non-negative. Since it is equal to zero at $\xi = \xi_{\min}$, it is necessary and sufficient for it to have non-negative derivative:

$$2\gamma(\alpha + \gamma\xi_{\min}) \geq 4\beta^2 \to \gamma \geq \frac{|\beta|}{\sqrt{\xi_{\min}}}. \tag{24}$$

Simple analysis gives the following conditions for the parameters α, β, γ such that we have optimal local convergence rate and all the eigenvalues of the linearized operator are real:

$$\begin{gathered}\alpha + \gamma\xi_{\min} = 2|\beta|\sqrt{\xi_{\min}}, \\ \frac{|\beta|}{\sqrt{\xi_{\min}}} \leq \gamma \leq \frac{2|\beta|}{\sqrt{\xi_{\min}}},\ 0 \leq \alpha \leq \frac{|\beta|}{\sqrt{\xi_{\min}}}.\end{gathered} \tag{25}$$

Discrete approximation of the time dynamics. The dynamics (6) can be written in the operator form as

$$w_t = \mathcal{L}w,\ w(0) = w_0,\ w = \begin{bmatrix} u \\ v \end{bmatrix}. \tag{26}$$

The usage of gradient descent optimization for this problem is equivalent to the forward Euler scheme:

$$w_{k+1} = w_k + \tau\mathcal{L}w_k, \tag{27}$$

and the eigenvalues of the discretized operator are equal to

$$\lambda_i^e = 1 + \tau\lambda_i, \tag{28}$$

where λ are given by Theorem 3. Since for all ξ_i such that $\alpha^2 - 4\beta^2\xi_i < 0$ the eigenvalue λ_i is complex, the modulus of the corresponding λ_i^e is

$$|\lambda_i^e|^2 = \left(1 - \frac{\tau\alpha}{2}\right)^2 + \tau^2\left(4\beta^2\xi_i - \alpha^2\right). \tag{29}$$

Since Δ_μ is an unbounded operator under mild assumptions on μ there exists an eigenvalue ξ_i of it that makes $|\lambda_i^e| > 1$, i.e. the Euler scheme is *absolutely unstable*. On the discrete level, when the functions are approximated by a neural network, we might be working in a subspace that does not contain eigenvalues that are too large, and the method might actually converge; but this is very problem–specific. As has been noted by Qin et al. (2020) the usage of more advanced time integration schemes leads to competitive GAN training results even without such functional constraints as spectral normalization (Miyato et al., 2018). If the parameters are selected in the range (25), then there are no complex eigenvalues, and we can make Euler scheme convergent. This requires regularization. Another important research direction is the development of more suitable time discretization schemes. The system (6) belongs to the class of *hyperbolic problems*, and special time discretization schemes have to be used. One class of such methods is total variation diminishing (TVD) schemes (Gottlieb & Shu, 1998). Note, that one of the methods considered in Qin et al. (2020) is the Heun method (Süli & Mayers, 2003) which is a second-order TVD Runge-Kutta scheme, which confirms its empirical efficiency. Thus, our analysis provides the theoretical foundation for the results of Qin et al. (2020).

Data augmentation. One of the most successful practical tricks significantly improving GAN convergence have been the usage of data augmentation (Karras et al., 2020a; Zhang et al., 2020; Zhao et al., 2020). In our framework, the usage of data augmentations is reflected in the shift of $\xi_{\min}$. Based on the intuition discussed earlier, higher values of $\xi_{\min}$ correspond to more "connected" distributions and allow for faster convergence. This is exactly what is intuitively achieved by data augmentation: we fill the "holes" in our dataset with new samples and make it more connected. We describe our experiments confirming this idea in Section 7.

Variational estimation of $\xi_{\min}$. For practical analysis of GAN convergence, we need to estimate the value of the Poincaré constant $\xi_{\min}$ for a given dataset. This can be performed in the standard supervised learning manner. Recall from the definition that $\xi_{\min}$ is the minimizer of the following optimization problem (commonly called the Rayleigh quotient).

$$\begin{gathered}\min \frac{\langle \nabla_x w, \nabla_x w \rangle_\mu}{\langle w, w \rangle_\mu} \\ \text{s.t. } w \in H^2(\mu),\ \langle 1, w \rangle_\mu = 0.\end{gathered} \tag{30}$$

Let us rewrite this optimization problem in a more convenient form. Note that the constraint $\langle 1, w \rangle_\mu = 0$ simply means that the $\mathbb{E}_{x\sim\mu}\, w = 0$. Since subtraction of a constant does not affect the gradient, we can rewrite (30) as

$$\begin{gathered}\min \frac{\mathbb{E}_{x\sim\mu}\,\|\nabla_x w(x)\|^2}{\mathrm{Var}_{x\sim\mu} w(x)} \\ \text{s.t } w \in H^2(\mu),\end{gathered} \tag{31}$$

where $\mathrm{Var}_{x \sim \mu}(w)$ denotes the variance of w with respect to μ. In practice we can parameterize w with a neural network, and perform optimization of (31) in a stochastic manner by replacing expectations over μ with their empirical counterparts (over a batch of inputs). Due to the scale invariance of the Rayleigh quotient, we employ spectral normalization (Miyato et al., 2018) as an additional regularization; this also falls in line with commonly used discriminator architectures. To summarize, we utilize a neural network $w(x;\theta)$ and for a dataset X we perform stochastic optimization of the loss given by (31). I.e, the loss on a single batch $X_b = \{X_i\}_{i=1}^N$ is given by

$$\mathcal{L}(\theta) = \frac{\frac{1}{N}\sum_{i=1}^N \|\nabla_x w(X_i;\theta)\|^2}{\frac{1}{N}\sum_{i=1}^N f(X_i)^2 - (\frac{1}{N}\sum_{i=1}^N w(X_i))^2}, \quad (32)$$

and, correspondingly, $\xi_{\min} \approx \min_\theta \mathcal{L}(\theta)$. Minimization of this loss function can be performed with standard optimizers such as SGD or Adam (Kingma & Ba, 2015). Note that the value of $\xi_{\min}$ obtained via a neural network approximation as an *upper bound* on the actual value. In principle, we might obtain arbitrarily small estimations while increasing the network's expressive power. In Section 7 we show that the obtained $\xi_{\min}$ estimate in fact converges to a certain (non-zero) value when we increase the number of parameters.

7. Experiments

Experimental setup. Our experiments are organized as follows. We start by numerically investigating the value of $\xi_{\min}$ and the impact of formulas from Section 6 on the convergence on synthetic datasets. We then study the more practical CIFAR-10 (Krizhevsky et al., 2009) dataset. Firstly, we show the correlation between the $\xi_{\min}$ obtained for various augmented versions of the dataset and FID values obtained for the corresponding GAN. We then show that a similar correlation holds when we perform *instance selection*, a recently proposed technique shown to improve GAN convergence (DeVries et al., 2020). In Appendix C we provide additional visualizations. For the synthetic datasets, our experiments were performed in JAX (Bradbury et al., 2018) using the ODE-GAN code available at GitHub[1]. For CIFAR-10 experiments we utilized PyTorch (Paszke et al., 2019). Importantly, we work with *standard* GAN models and validate theoretical findings predicted by lGAN.

Gaussian Mixture. In order to study the effect of $\xi_{\min}$ and the choice of α and γ on the training procedure, we set up a simple one-dimensional test: a mixture of two normal distributions $\mathcal{N}(0,1)$ and $\mathcal{N}(M,1)$, where M is the separation parameter. Intuitively, the larger the M, the smaller is $\xi_{\min}$, since it reduces connectivity of our measure. We also verify it numerically, as shown in Figure 1 (top). In this case, we utilize a simple two-layer MLP with spectral normalization on top of linear layers. We observe that indeed the value of $\xi_{\min}$ decays rapidly with the increase of the separation parameter M.

To experiment with GAN convergence in this toy setting, we setup two MLP models for D and G, and train the LSGAN-like model with

$$\phi_1(x) = \frac{\alpha}{2}x^2 + \beta x,\ \phi_2(x) = -\beta x,$$

and gradient penalty with factor γ. We train the model using the ODE-GAN approach with the Heun method. We experiment with two moderately separated mixtures, namely with $M = 4$ and $M = 4$. Respective $\xi_{\min}$ values obtained by the numerical simulation described above are $\xi_{\min} = 0.25$ and $\xi_{\min} = 0.124$. We consider five options for the values of (α, β, γ). We start with the baseline WGAN corresponding to $(0, -1, 0)$. We consider two options for the value of γ: the optimal one $\gamma = {}^{2|\beta|}/_{\sqrt{\xi_{\min}}}$ predicted by the theory, and a sub-optimal $\gamma = {}^{|\beta|}/_{\sqrt{\xi_{\min}}}$. With the latter value we obtain $\mathrm{Im}\ \lambda_{\max} \sim |\beta|\sqrt{\xi_{\min}}$. We also consider two LSGAN variants. First one is the default version with the parameters $(0.25, -0.5, 0)$. For the second version we fix $\beta = 1$ and choose the optimal α and γ according to (25). We measure performance of a GAN model via the Earth Mover's Distance (EMD) between the Gaussians fitted on real and synthetic data. This approach resembles the commonly used Frechét Inception Distance (FID) metric for GAN evaluation. Results are visualized on Figure 1. We observe that the resulting convergence plots match our theoretical predictions. In particular, we see that when the parameters are chosen in the optimal way, methods converge more rapidly and stably. On the other hand, when γ is not tuned or is sub-optimal, the convergence is more oscillatory, and GANs struggle to converge. Note that even for the optimal γ for methods with $\alpha > 0$, we still observe some oscillations. This may be a result of a noise induced by neural function approximation or stochasticity of training.

Toy data augmentation. In this set of experiments, we verify if the effects of data augmentations on GAN convergence can be predicted by evaluating $\xi_{\min}$ for the correspondingly augmented dataset. We perform it is follows: consider

$$\mu = \frac{1}{2}\mathcal{N}(-3,1) + \frac{1}{2}\mathcal{N}(3,1) \quad (33)$$

and its corresponding augmented version:

$$\mu_p = \Big(\frac{1}{2} - \frac{p}{4}\Big)\mathcal{N}(-3,1) + \frac{p}{2}\mathcal{N}(0,1) + \Big(\frac{1}{2} - \frac{p}{4}\Big)\mathcal{N}(3,1). \quad (34)$$

By increasing the value of p, we improve the connectivity of the measure and correspondingly increase the value $\xi_{\min}$ of

[1] https://github.com/deepmind/deepmind-research/tree/master/ode_gan

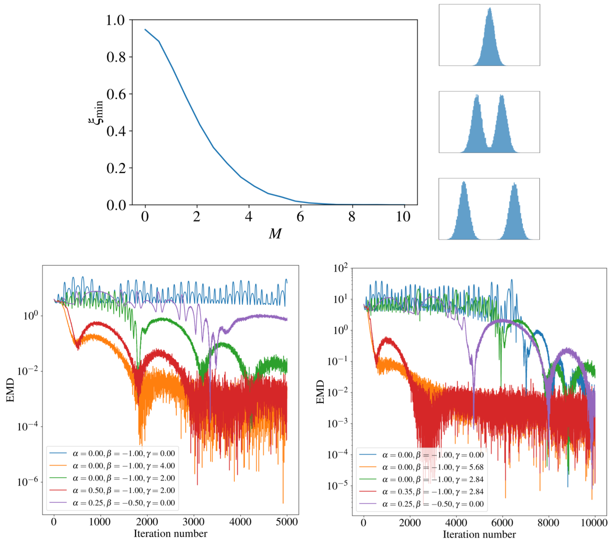

Figure 1. Numerical experiments with GAN convergence on a mixture of two univariate Gaussians. *(Top)* Estimated value of $\xi_{\min}$ vs the mixture separating parameter M. *(Bottom)* Convergence plots of GANs with different loss functions and regularization strengths. Methods with parameters selected optimally according to theory present better convergence.

μ_p. In Figure 2 we provide visualizations and estimated values of $\xi_{\min}$ for $p \in \{0, 0.2, 0.4\}$. For these values, we train a standard LSGAN with the Heun method, and visualize the convergence plots (specifically, we plot $\|D(t)\|^2_{\mu_p}$). We observe that increasing $\xi_{\min}$ in these cases indeed corresponds to better convergence.

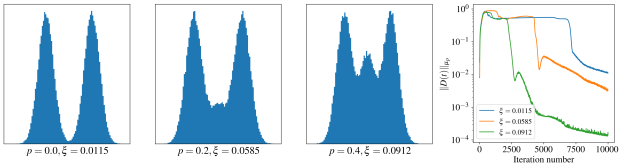

Figure 2. A toy data augmentation example: increasing connectivity corresponds to a better convergence.

Data augmentation of CIFAR-10. We now analyze a practical case of the CIFAR-10 dataset. We consider a number of augmentations commonly utilized in data augmentation pipelines for GANs (see Figure 3). In particular, these augmentations include both spatial (translations, zooming) and visual augmentations (color adjustment). Folklore knowledge is that the first type is generally helpful, while the second is not. In our framework, the positive effect of augmentations can be understood as improving the "connectivity" of a dataset, which can be quantitatively measured by evaluating $\xi_{\min}$ of the augmented version of a dataset.

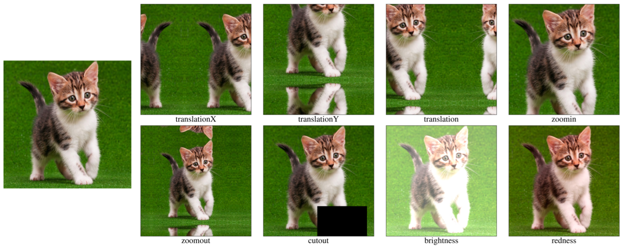

Figure 3. Image augmentations commonly used in GAN training.

Increased value of $\xi_{\min}$ both intuitively and theoretically (as supported by Theorem 3) corresponds to better convergence and better FID scores. Our experiments are based on a thorough analysis of the impact of augmentations on the quality of a GAN trained on CIFAR-10 provided in Zhao et al. (2020). Specifically, the authors select a *single* augmentation from a predefined set and train several types of GANs by augmenting real and fake images. The strength of the augmentation is controlled by a single parameter λ, and the authors provide the obtained FID for a number of its possible values (e.g., we may vary how strongly we zoom an image); we refer the reader to Zhao et al. (2020) for specific details on each augmentation. We selected a large portion of augmentations studied in this paper and replicated its data augmentation setup. For each augmentation and each strength value, we minimize the Rayleigh quotient with a neural network mimicking the SNDCGAN discriminator from Zhao et al. (2020). We train it on the train part of the dataset by applying the respective augmentation to each image with probability one. Note that the actual augmentation strength is randomly sampled from the range $[0, \lambda]$, so we cover the entire distribution relatively well. For training, we use the batch size of 256 and Adam optimizer with a learning rate 10^{-4} (results are not sensitive to these parameters). We measure the actual value of $\xi_{\min}$ by sweeping across the augmented test set 5 times to better cover the distribution and aggregate the results across all 50000 (augmented) samples. For the reference values, we choose the FID scores obtained by the SNDCGAN model with Balanced Consistency Regularization (bCR) compiled from Zhao et al. (2020, Figure 3). This model achieves better generation quality so our theory is more reliable in this case. Our results are provided at Figure 4. For convenience, we normalize the obtained values by dividing it by $\xi_{\min}$ of the non-augmented dataset (approximately equal to 0.0042 by our estimation). We observe that FID scores in many cases indeed follow the behavior of $\xi_{\min}$. For instance, for `zoomin` and `cutout`, the estimated $\xi_{\min}$ values predict that there is an intermediate optimal augmentation strength, which is matched by the practice. For `translation`, we observe that stronger augmentations worsen the connectivity of the dataset, which results in worse FID scores. We can also observe a significant drop in $\xi_{\min}$ for color-based augmentations, confirming their practical inefficiency. We note, in some cases (e.g., `zoomout`), we do not observe a direct correspondence. This may be a result of auxiliary effects not covered by our theory or of a discrepancy between ours and the authors' implementation.

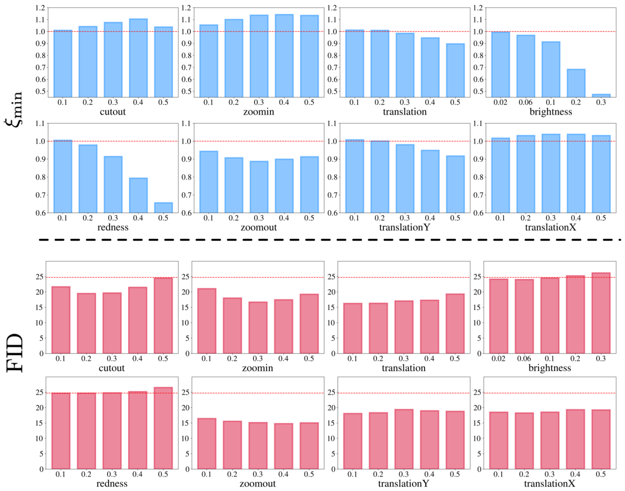

Figure 4. *(Top)* The values of $\xi_{\min}$ for augmented versions of CIFAR-10 for different values of the augmentation strength. We normalize them dividing by $\xi_{\min} \approx 0.0042$ of the non-augmented version of the dataset. *(Bottom)* The FID values of a SNDCGAN (BCR) trained on the corresponding dataset. The values are taken from Zhao et al. (2020). For $\xi_{\min}$ a higher value is "better" in terms of connectivity of dataset and theoretical convergence. For FID smaller values are better. In most cases we observe a negative rank correlation between $\xi_{\min}$ and FID.

Instance selection. Another possible way to manipulate the data distribution is to remove non-typical samples, which can confuse the generator (DeVries et al., 2020). This can be performed by fitting a Gaussian model on feature vectors of a dataset produced by some pretrained model (in our experiments, we used Inception v3 (Szegedy et al., 2016)) and keeping only samples with high likelihood under this model. This procedure results in better FID scores of trained GANs, as shown in DeVries et al. (2020). In our framework, this can be understood as another way to improve the dataset's connectivity since by removing outliers, we reduce the number of gaps in the data. Experiments in the aforementioned paper were conducted on the resized ImageNet dataset; however, we hypothesize that this phenomenon may also be understood on the conceptually similar CIFAR-10 dataset. We perform an experiment confirming this effect quantitatively. We follow the steps described above, and for each value of $\psi \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$ evaluate $\xi_{\min}$ of the dataset obtained by removing samples in the bottom ψ-quantile of the likelihood. Our results are provided in Table 1. We observe that for higher values of the truncation parameter, we indeed obtain better dataset connectivity, confirming practical findings of DeVries et al. (2020).

Table 1. Normalized differences in $\xi_{\min}$ for CIFAR-10 with instance selection performed according to the procedure outlined in DeVries et al. (2020) with respect to the truncation (lower) quantile ψ. $\Delta\xi_{\min}$ is computed as $({}^{\xi_{\min}(\psi)}/_{\xi_{\min}(0)} - 1) \times 10^2$ for easier comparison. We observe that stronger truncation increases the value of $\xi_{\min}$, confirming better empirical GAN convergence.

ψ	0	0.1	0.2	0.3	0.4	0.5
$\Delta\xi_{\min} \uparrow$	0	0.95	−0.04	1.24	**2.90**	2.74

Limitations of variational estimation. We now analyze whether the obtained approximations of $\xi_{\min}$ are reliable in the sense that they will not get arbitrarily small when the networks of higher capacity are used. To verify the stability of the obtained $\xi_{\min}$ estimate, we perform a sweep across the width (i.e., the number of filters in convolutions of the ResNet-18 model) of the network. We evaluate 19 values of width starting with 32 with the increment 8. The obtained $\xi_{\min}$ values with respect to the number of parameters are visualized in Figure 5. We observe that the obtained value indeed converges when we increase the depth; based on the plot of the relative error we may note that the convergence is roughly linear in terms of parameter count ($R^2 = 0.87$).

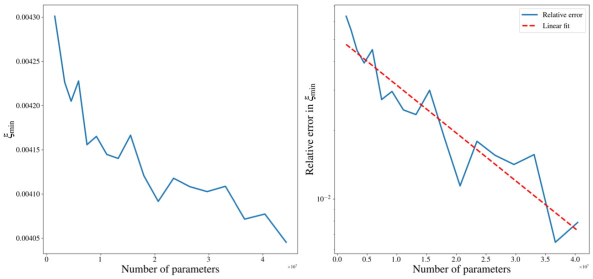

Figure 5. *(Left)* Convergence of the variational $\xi_{\min}$ estimate with respect to the number of parameters in the network. Parameters are given in multiples of 10^7. *(Right)* Relative error of the obtained estimated with respect to the value obtained for width=176.

8. Conclusion

We presented a novel framework for a theoretical understanding of GAN training by analyzing the local convergence in the functional space. Namely, we represent the GAN dynamics as a system of partial differential equations and analyze the spectrum of the corresponding differential operator, which determines the dynamics convergence. As the main result, we show how the spectrum depends on the properties of the target distribution, in particular, on its Poincaré constant. Our perspective provides a new understanding of established GAN tricks, such as gradient penalty or dataset augmentation. For practitioners, our paper develops an efficient method that allows to choose optimal augmentations for a particular dataset.

References

Adams, R. A. and Fournier, J. J. *Sobolev spaces*. Elsevier, 2003.

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

Balduzzi, D., Racaniere, S., Martens, J., Foerster, J., Tuyls, K., and Graepel, T. The mechanics of n-player differentiable games. In *International Conference on Machine Learning*, pp. 354–363. PMLR, 2018.

Benzi, M., Golub, G. H., Liesen, J., et al. Numerical solution of saddle point problems. *Acta numerica*, 14(1):1–137, 2005.

Boffi, D., Brezzi, F., Fortin, M., et al. *Mixed finite element methods and applications*, volume 44. Springer, 2013.

Bradbury, J., Frostig, R., Hawkins, P., Johnson, M. J., Leary, C., Maclaurin, D., Necula, G., Paszke, A., VanderPlas, J., Wanderman-Milne, S., and Zhang, Q. JAX: composable transformations of Python+NumPy programs, 2018. URL http://github.com/google/jax.

Cianchi, A., Maz'ya, V., et al. On the discreteness of the spectrum of the Laplacian on noncompact Riemannian manifolds. *Journal of differential geometry*, 87(3):469–492, 2011.

Coifman, R. R. and Lafon, S. Diffusion maps. *Applied and computational harmonic analysis*, 21(1):5–30, 2006.

DeVries, T., Drozdzal, M., and Taylor, G. W. Instance Selection for GANs. *Advances in Neural Information Processing Systems*, 2020.

Dieudonné, J. *Foundations of modern analysis*. Read Books Ltd, 1969.

Fiedler, M. Algebraic connectivity of graphs. *Czechoslovak mathematical journal*, 23(2):298–305, 1973.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative Adversarial Nets. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems*, volume 27, pp. 2672–2680. Curran Associates, Inc., 2014.

Gottlieb, S. and Shu, C.-W. Total variation diminishing Runge-Kutta schemes. *Mathematics of computation*, 67 (221):73–85, 1998.

Griffiths, D. J. Introduction to electrodynamics, 2005.

Grigoryan, A. *Heat kernel and analysis on manifolds*, volume 47. American Mathematical Soc., 2009.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. C. Improved Training of Wasserstein GANs. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30, pp. 5767–5777. Curran Associates, Inc., 2017.

Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J., and Aila, T. Training generative adversarial networks with limited data. *Advances in Neural Information Processing Systems*, 2020a.

Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., and Aila, T. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8110–8119, 2020b.

Kingma, D. P. and Ba, J. Adam: A Method for Stochastic Optimization. In *ICLR (Poster)*, 2015. URL http://arxiv.org/abs/1412.6980.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

Ladyzhenskaya, O. A. *The mathematical theory of viscous incompressible flow*, volume 2. Gordon and Breach New York, 1969.

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*, pp. 907–915. PMLR, 2019.

Mao, X., Li, Q., Xie, H., Lau, R. Y., Wang, Z., and Paul Smolley, S. Least squares generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 2794–2802, 2017.

Mescheder, L., Nowozin, S., and Geiger, A. The Numerics of GANs. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30, pp. 1825–1835. Curran Associates, Inc., 2017.

Mescheder, L., Geiger, A., and Nowozin, S. Which Training Methods for GANs do actually Converge? In *International Conference on Machine learning (ICML)*, pp. 3481–3490. PMLR, 2018.

Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y. Spectral Normalization for Generative Adversarial Networks. In *International Conference on Learning Representations*,

2018. URL https://openreview.net/forum?id=B1QRgziT-.

Nagarajan, V. and Kolter, J. Z. Gradient descent GAN optimization is locally stable. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 5591–5600, 2017.

Nie, W. and Patel, A. B. Towards a better understanding and regularization of GAN training dynamics. In *Uncertainty in Artificial Intelligence*, pp. 281–291. PMLR, 2020.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An Imperative Style, High-Performance Deep Learning Library. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. Curran Associates, Inc., 2019.

Qin, C., Wu, Y., Springenberg, J. T., Brock, A., Donahue, J., Lillicrap, T. P., and Kohli, P. Training Generative Adversarial Networks by Solving Ordinary Differential Equations. *Advances in Neural Information Processing Systems*, 2020.

Roth, K., Lucchi, A., Nowozin, S., and Hofmann, T. Stabilizing Training of Generative Adversarial Networks through Regularization. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30, pp. 2018–2028. Curran Associates, Inc., 2017.

Süli, E. and Mayers, D. F. *An introduction to numerical analysis*. Cambridge university press, 2003.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826, 2016.

Zhang, H., Zhang, Z., Odena, A., and Lee, H. Consistency Regularization for Generative Adversarial Networks. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=S1lxKlSKPH.

Zhao, Z., Zhang, Z., Chen, T., Singh, S., and Zhang, H. Image augmentations for GAN training. *arXiv preprint arXiv:2006.02595*, 2020.