(α_D, α_G)-GANs: Addressing GAN Training Instabilities via Dual Objectives

Monica Welfert*, Kyle Otstot*, Gowtham R. Kurri†, and Lalitha Sankar*
*Arizona State University, USA {mwelfert,kotstot,lalithasankar}@asu.edu
†IIIT Hyderabad, India, {gowtham.kurri}@iiit.ac.in

2023 IEEE International Symposium on Information Theory (ISIT) | 978-1-6654-7554-9/23/$31.00 ©2023 IEEE | DOI: 10.1109/ISIT54713.2023.10206844

Abstract—**In an effort to address the training instabilities of GANs, we introduce a class of dual-objective GANs with different value functions (objectives) for the generator (G) and discriminator (D). In particular, we model each objective using α-loss, a tunable classification loss, to obtain (α_D, α_G)-GANs, parameterized by $(\alpha_D, \alpha_G) \in (0, \infty]^2$. For sufficiently large number of samples and capacities for G and D, we show that the resulting non-zero sum game simplifies to minimizing an f-divergence under appropriate conditions on (α_D, α_G). In the finite sample and capacity setting, we define estimation error to quantify the gap in the generator's performance relative to the optimal setting with infinite samples and obtain upper bounds on this error, showing it to be order optimal under certain conditions. Finally, we highlight the value of tuning (α_D, α_G) in alleviating training instabilities for the synthetic 2D Gaussian mixture ring and the Stacked MNIST datasets.**

I. Introduction

Generative adversarial networks (GANs) have become a crucial data-driven tool for generating synthetic data. GANs are generative models trained to produce samples from an unknown (real) distribution using a finite number of training data samples. They consist of two modules, a generator G and a discriminator D, parameterized by vectors $\theta \in \Theta \subset \mathbb{R}^{n_g}$ and $\omega \in \Omega \subset \mathbb{R}^{n_d}$, respectively, which play an adversarial game with each other. The generator G_θ maps noise $Z \sim P_Z$ to a data sample in $\mathcal{X}$ via the mapping $z \mapsto G_\theta(z)$ and aims to mimic data from the real distribution P_r. The discriminator D_ω takes as input $x \in \mathcal{X}$ and classifies it as real or generated by computing a score $D_\omega(x) \in [0, 1]$ which reflects the probability that x comes from P_r (real) as opposed to P_{G_θ} (synthetic). For a chosen value function $V(\theta, \omega)$, the adversarial game between G and D can be formulated as a zero-sum min-max problem given by

$$\inf_{\theta\in\Theta}\sup_{\omega\in\Omega} V(\theta,\omega). \tag{1}$$

Goodfellow *et al.* [1] introduce the vanilla GAN for which

$$V_{\text{VG}}(\theta,\omega)=\mathbb{E}_{X\sim P_r}[\log D_\omega(X)]+\mathbb{E}_{X\sim P_{G_\theta}}[\log(1-D_\omega(X))].$$

For this V_{VG}, they show that when the discriminator class $\{D_\omega\}_{\omega\in\Omega}$ is rich enough, (1) simplifies to minimizing the Jensen-Shannon divergence [2] between P_r and P_{G_θ}.

Various other GANs have been studied in the literature using different value functions, including f-divergence based GANs called f-GANs [3], IPM based GANs [4]–[6], etc. Observing that the discriminator is a classifier, recently, Kurri *et al.* [7], [8] show that the value function in (1) can be written using a class probability estimation (CPE) loss $\ell(y,\hat{y})$ whose inputs are the true label $y \in \{0,1\}$ and predictor $\hat{y} \in [0,1]$ (soft prediction of y) as

$$V(\theta,\omega)=\mathbb{E}_{X\sim P_r}[-\ell(1,D_\omega(X))]+\mathbb{E}_{X\sim P_{G_\theta}}[-\ell(0,D_\omega(X))].$$

Using this approach, they introduce α-GAN using the tunable CPE loss α-loss [9], [10], defined for $\alpha \in (0,\infty]$ as

$$\ell_\alpha(y,\hat{y}):=\frac{\alpha}{\alpha-1}\left(1-y\hat{y}^{\frac{\alpha-1}{\alpha}}-(1-y)(1-\hat{y})^{\frac{\alpha-1}{\alpha}}\right). \tag{2}$$

They show that the α-GAN formulation recovers various f-divergence based GANs including the Hellinger GAN [3] ($\alpha = 1/2$), the vanilla GAN [1] ($\alpha = 1$), and the Total Variation (TV) GAN [3] ($\alpha = \infty$). Further, for a large enough discriminator class, the min-max optimization for α-GAN in (1) simplifies to minimizing the Arimoto divergence [11], [12].

While each of the abovementioned GANs have distinct advantages, they continue to suffer from one or more types of training instabilities, including vanishing/exploding gradients, mode collapse, and sensitivity to hyperparameter tuning. In [1], Goodfellow *et al.* note that the generator's objective in the vanilla GAN can *saturate* early in training (due to the use of the sigmoid activation) when D can easily distinguish between the real and synthetic samples, i.e., when the output of D is near zero for all synthetic samples, leading to vanishing gradients. Further, a confident D induces a steep gradient at samples close to the real data, thereby preventing G from learning such samples due to exploding gradients. To alleviate these, [1] proposes a *non-saturating* (NS) generator objective:

$$V_{\text{VG}}^{\text{NS}}(\theta,\omega)=\mathbb{E}_{X\sim P_{G_\theta}}[-\log D_\omega(X)]. \tag{3}$$

This NS version of the vanilla GAN may be viewed as involving different objective functions for the two players (in fact, with two versions of the $\alpha = 1$ CPE loss, i.e., log-loss, for D and G). However, it continues to suffer from mode collapse [13], [14]. While other dual-objective GANs have also been proposed (e.g., Least Squares GAN (LSGAN) [15], RényiGAN [16], NS f-GAN [3], hybrid f-GAN [17]), few have had success fully addressing training instabilities.

This work is supported in part by NSF grants CIF-1901243, CIF-1815361, CIF-2007688, CIF-2134256, SaTC-2031799, and SCH-2205080. Gowtham R. Kurri was with Arizona State University when this work was done.

978-1-6654-7554-9/23/$31.00 ©2023 IEEE

Recent results have shown that α-loss demonstrates desirable gradient behaviors for different α values [10]. It also assures learning robust classifiers that can reduce the confidence of D (a classifier) thereby allowing G to learn without gradient issues. To this end, we introduce a different α-loss objective for each player to address training instabilities. We propose a tunable dual-objective (α_D,α_G)-GAN, where the objective functions of D and G are written in terms of α-loss with parameters $\alpha_D \in (0,\infty]$ and $\alpha_G \in (0,\infty]$, respectively. Our key contributions are:

- For this non-zero sum game, we show that a Nash equilibrium exists. For appropriate (α_D,α_G) values, we derive the optimal strategies for D and G and prove that for the optimal D_{ω^*}, G minimizes an f-divergence and can therefore learn the real distribution P_r.
- Since α-GAN captures various GANs, including the vanilla GAN, it can potentially suffer from vanishing gradients due to a saturation effect. We address this by introducing a non-saturating version of the (α_D,α_G)-GAN and present its Nash equilibrium strategies for D and G.
- A natural question that arises is how to quantify the theoretical guarantees for dual-objective GANs, specifically for (α_D,α_G)-GANs, in terms of their estimation capabilities in the setting of limited capacity models and finite training samples. To this end, we define estimation error for (α_D,α_G)-GANs, present an upper bound on the error, and a matching lower bound under additional assumptions.
- Finally, we demonstrate empirically that tuning α_D and α_G significantly reduces vanishing and exploding gradients and alleviates mode collapse on a synthetic 2D-ring dataset. For the high-dimensional Stacked MNIST dataset, we show that our tunable approach is more robust in terms of mode coverage to the choice of GAN hyperparameters, including number of training epochs and learning rate, relative to both vanilla GAN and LSGAN.

II. Main Results

A. (α_D,α_G)-GAN

We first propose a dual-objective (α_D,α_G)-GAN with different objective functions for the generator and discriminator. In particular, the discriminator maximizes $V_{\alpha_D}(\theta,\omega)$ while the generator minimizes $V_{\alpha_G}(\theta,\omega)$, where

$$\begin{aligned}&V_\alpha(\theta,\omega)\\&=\mathbb{E}_{X\sim P_r}[-\ell_\alpha(1,D_\omega(X))]+\mathbb{E}_{X\sim P_{G_\theta}}[-\ell_\alpha(0,D_\omega(X))], \quad (4)\end{aligned}$$

for $\alpha=\alpha_D,\alpha_G\in(0,\infty]$. We recover the α-GAN [7], [8] value function when $\alpha_D=\alpha_G=\alpha$. The resulting (α_D,α_G)-GAN is given by

$$\sup_{\omega\in\Omega} V_{\alpha_D}(\theta,\omega) \qquad (5a)$$

$$\inf_{\theta\in\Theta} V_{\alpha_G}(\theta,\omega). \qquad (5b)$$

The following theorem presents the conditions under which the optimal generator learns the real distribution P_r when the discriminator set Ω is large enough.

Theorem 1. *For a fixed generator G_θ, the discriminator optimizing* (5a) *is given by*

$$D_{\omega^*}(x)=\frac{p_r(x)^{\alpha_D}}{p_r(x)^{\alpha_D}+p_{G_\theta}(x)^{\alpha_D}}, \qquad (6)$$

where p_r and p_{G_θ} are the corresponding densities of the distributions P_r and P_{G_θ}, respectively, with respect to a base measure dx (e.g., Lebesgue measure). For this D_{ω^*} and the function $f_{\alpha_D,\alpha_G}:\mathbb{R}_+\to\mathbb{R}$ defined as

$$f_{\alpha_D,\alpha_G}(u)=\frac{\alpha_G}{\alpha_G-1}\left(\frac{u^{\alpha_D\left(1-\frac{1}{\alpha_G}\right)+1}+1}{\left(u^{\alpha_D}+1\right)^{1-\frac{1}{\alpha_G}}}-2^{\frac{1}{\alpha_G}}\right), \qquad (7)$$

(5b) *simplifies to minimizing a non-negative symmetric f_{α_D,α_G}-divergence $D_{f_{\alpha_D,\alpha_G}}(\cdot||\cdot)$ as*

$$\inf_{\theta\in\Theta} D_{f_{\alpha_D,\alpha_G}}(P_r||P_{G_\theta})+\frac{\alpha_G}{\alpha_G-1}\left(2^{\frac{1}{\alpha_G}}-2\right), \qquad (8)$$

which is minimized iff $P_{G_\theta}=P_r$ for $(\alpha_D,\alpha_G)\in(0,\infty]^2$ such that $\left(\alpha_D\leq 1, \alpha_G>\frac{\alpha_D}{\alpha_D+1}\right)$ or $\left(\alpha_D>1, \frac{\alpha_D}{2}<\alpha_G\leq\alpha_D\right)$.

Proof sketch. We substitute the optimal discriminator of (5a) into the objective function of (5b) and translate it into the form in (8) by finding the appropriate conditions on α_D and α_G for f_{α_D,α_G} to be a strictly convex function. Figure 1(a) illustrates the feasible (α_D,α_G)-region. A detailed proof can be found in [18, Appendix A].

Noting that α-GAN recovers various well-known GANs, including the vanilla GAN, which is prone to saturation, the (α_D,α_G)-GAN formulation using the generator objective function in (4) can similarly saturate early in training, causing vanishing gradients. We therefore propose the following NS alternative to the generator's objective in (4):

$$V^{\text{NS}}_{\alpha_G}(\theta,\omega)=\mathbb{E}_{X\sim P_{G_\theta}}[\ell_{\alpha_G}(1,D_\omega(X))], \qquad (9)$$

thereby replacing (5b) with

$$\inf_{\theta\in\Theta} V^{\text{NS}}_{\alpha_G}(\theta,\omega). \qquad (10)$$

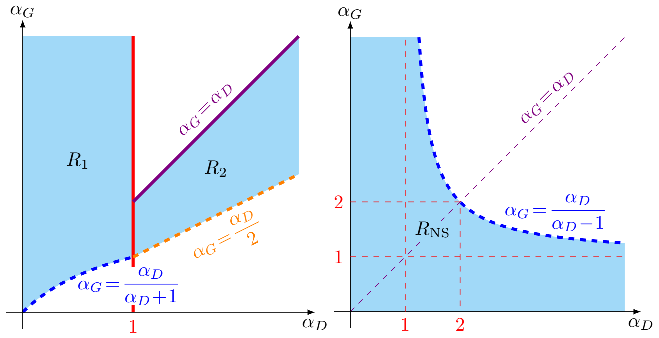
Comparing (5b) and (10), note that the additional expectation term over P_r in (4) results in (5b) simplifying to a symmetric divergence for D_{ω^*} in (6), whereas the single term in (9) will result in (10) simplifying to an asymmetric divergence. The optimal discriminator for this NS game remains the same as in (6). The following theorem provides the solution to (10) under the assumption that the optimal discriminator can be attained.

Theorem 2. *For the same D_{ω^*} in* (6) *and the function $f^{NS}_{\alpha_D,\alpha_G}:\mathbb{R}_+\to\mathbb{R}$ defined as*

$$f^{NS}_{\alpha_D,\alpha_G}(u)=\frac{\alpha_G}{\alpha_G-1}\left(2^{\frac{1}{\alpha_G}-1}-\frac{u^{\alpha_D\left(1-\frac{1}{\alpha_G}\right)}}{\left(u^{\alpha_D}+1\right)^{1-\frac{1}{\alpha_G}}}\right), \qquad (11)$$

(5b) *simplifies to minimizing a non-negative asymmetric $f^{NS}_{\alpha_D,\alpha_G}$-divergence $D_{f^{NS}_{\alpha_D,\alpha_G}}(\cdot||\cdot)$ as*

$$\inf_{\theta\in\Theta} D_{f^{NS}_{\alpha_D,\alpha_G}}(P_r||P_{G_\theta})+\frac{\alpha_G}{\alpha_G-1}\left(1-2^{\frac{1}{\alpha_G}-1}\right), \qquad (12)$$



(a) (b)

Fig. 1. (a) Plot of regions $R_1=\{(\alpha_D,\alpha_G)\in(0,\infty]^2\,|\,\alpha_D\leq 1,\alpha_G>\frac{\alpha_D}{\alpha_D+1}\}$ and $R_2=\{(\alpha_D,\alpha_G)\in(0,\infty]^2\,|\,\alpha_D>1,\frac{\alpha_D}{2}<\alpha_G\leq\alpha_D\}$ for which f_{α_D,α_G} is strictly convex. (b) Plot of region $R_{\text{NS}}=\{(\alpha_D,\alpha_G)\in(0,\infty]^2\,|\,\alpha_D+\alpha_G>\alpha_D\alpha_G\}$ for which $f^{\text{NS}}_{\alpha_D,\alpha_G}$ is strictly convex.

which is minimized iff $P_{G_\theta}=P_r$ for $(\alpha_D,\alpha_G)\in(0,\infty]^2$ such that $\alpha_D+\alpha_G>\alpha_G\alpha_D$.

The proof mimics that of Theorem 1 and is detailed in [18, Appendix B]. Figure 1(b) illustrates the feasible (α_D,α_G)-region; in contrast to the saturating setting of Theorem 1, the NS setting constrains $\alpha\leq 2$ when $\alpha_D=\alpha_G=\alpha$. Nonetheless, we later show empirically in Section III-B that even tuning over this restricted set provides robustness against hyperparameter choices.

B. Estimation Error

Theorems 1 and 2 assume sufficiently large number of training samples and ample discriminator and generator capacity. However, in practice both the number of training samples and model capacity are usually limited. We consider a setting similar to prior works on generalization and estimation error for GANs (e.g., [8], [19]) with finite training samples $S_x=\{X_1,\ldots,X_n\}$ and $S_z=\{Z_1,\ldots,Z_m\}$ from P_r and P_Z, respectively, and with neural networks chosen as the discriminator and generator models. The sets of samples S_x and S_z induce the empirical real and generated distributions $\hat{P}_r$ and $\hat{P}_{G_\theta}$, respectively. A useful quantity to evaluate the performance of GANs in this setting is that of the estimation error, defined in [19] as the performance gap of the optimized value function when trained using only finite samples relative to the optimal when the statistics are known. Using this definition, [8] derived upper bounds on this error for α-GANs. However, such a definition requires a common value function for both discriminator and generator, and therefore, does not directly apply to the dual-objective setting we consider here.

Our definition relies on the observation that estimation error inherently captures the effectiveness of the generator (for a corresponding optimal discriminator model) in learning with limited samples. We formalize this intuition below.

Since (α_D,α_G)-GANs use different objective functions for the discriminator and generator, we start by defining the optimal discriminator ω^* for a generator model G_θ as

$$\omega^*(P_r,P_{G_\theta}):=\operatorname*{argmax}_{\omega\in\Omega} V_{\alpha_D}(\theta,\omega)\big|_{P_r,P_{G_\theta}}, \tag{13}$$

where the notation $|_{.,.}$ allows us to make explicit the distributions used in the value function. In keeping with the literature where the value function being minimized is referred to as the neural net (NN) distance (since D and G are modeled as neural networks) [8], [19], [20], we define the generator's NN distance $d_{\omega^*(P_r,P_{G_\theta})}$ as

$$d_{\omega^*(P_r,P_{G_\theta})}(P_r,P_{G_\theta}):=V_{\alpha_G}(\theta,\omega^*(P_r,P_{G_\theta}))\big|_{P_r,P_{G_\theta}}. \tag{14}$$

The resulting minimization for training the (α_D,α_G)-GAN using finite samples is

$$\inf_{\theta\in\Theta} d_{\omega^*(\hat{P}_r,\hat{P}_{G_\theta})}(\hat{P}_r,\hat{P}_{G_\theta}). \tag{15}$$

Denoting $\hat{\theta}^*$ as the minimizer of (15), we define the estimation error for (α_D,α_G)-GANs as

$$d_{\omega^*(P_r,P_{G_{\hat{\theta}^*}})}(P_r,P_{G_{\hat{\theta}^*}})-\inf_{\theta\in\Theta} d_{\omega^*(P_r,P_{G_\theta})}(P_r,P_{G_\theta}). \tag{16}$$

We use the same notation as in [8], detailed in the following for easy reference. For $x\in\mathcal{X}:=\{x\in\mathbb{R}^d:||x||_2\leq B_x\}$ and $z\in\mathcal{Z}:=\{z\in\mathbb{R}^p:||z||_2\leq B_z\}$, we model the discriminator and generator as k- and l-layer neural networks, respectively, with

$$D_\omega: x\mapsto\sigma\left(\mathbf{w}_k^\intercal r_{k-1}(\mathbf{W}_{d-1}r_{k-2}(\ldots r_1(\mathbf{W}_1(x)))\right) \tag{17}$$

$$G_\theta: z\mapsto\mathbf{V}_l s_{l-1}(\mathbf{V}_{l-1}s_{l-2}(\ldots s_1(\mathbf{V}_1 z))), \tag{18}$$

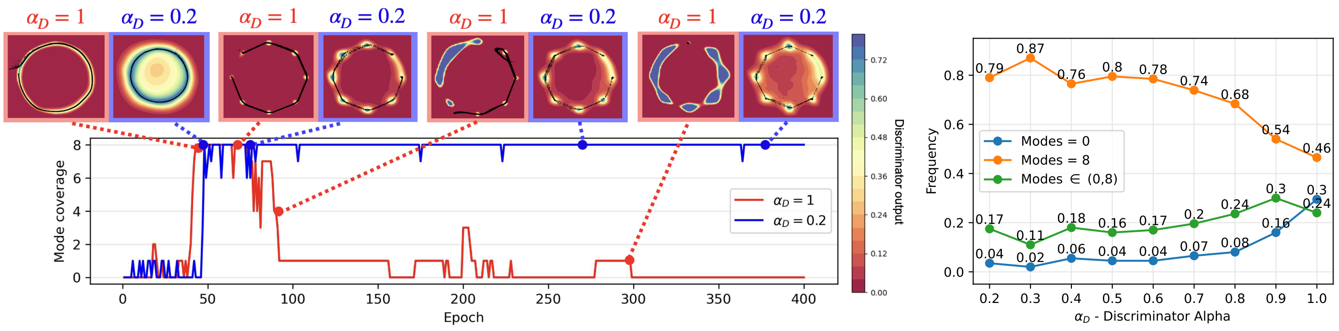
where (i) $\mathbf{w}_k$ is a parameter vector of the output layer; (ii) for $i\in[1:k-1]$ and $j\in[1:l]$, $\mathbf{W}_i$ and $\mathbf{V}_j$ are parameter matrices; (iii) $r_i(\cdot)$ and $s_j(\cdot)$ are entry-wise activation functions of layers i and j, respectively, i.e., for $\mathbf{a}\in\mathbb{R}^t$, $r_i(\mathbf{a})=[r_i(a_1),\ldots,r_i(a_t)]$ and $s_i(\mathbf{a})=[s_i(a_1),\ldots,s_i(a_t)]$; and (iv) $\sigma(\cdot)$ is the sigmoid function given by $\sigma(p)=1/(1+\mathrm{e}^{-p})$. We assume that each $r_i(\cdot)$ and $s_j(\cdot)$ are R_i- and S_j-Lipschitz, respectively, and also that they are positive homogeneous, i.e., $r_i(\lambda p)=\lambda r_i(p)$ and $s_j(\lambda p)=\lambda s_j(p)$, for any $\lambda\geq 0$ and $p\in\mathbb{R}$. Finally, as is common in such analysis [19], [21]–[23], we assume that the Frobenius norms of the parameter matrices are bounded, i.e., $||\mathbf{W}_i||_F\leq M_i$, $i\in[1:k-1]$, $||\mathbf{w}_k||_2\leq M_k$, and $||\mathbf{V}_j||_F\leq N_j$, $j\in[1:l]$. We now present an upper bound on (16) in the following theorem.

Theorem 3. *In the setting described above, with probability at least $1-2\delta$ over the randomness of training samples $S_x=\{X_i\}_{i=1}^n$ and $S_z=\{Z_j\}_{j=1}^m$, we have*

$$\begin{aligned} & d_{\omega^*(P_r,P_{G_{\hat{\theta}^*}})}(P_r,P_{G_{\hat{\theta}^*}})-\inf_{\theta\in\Theta} d_{\omega^*(P_r,P_{G_\theta})}(P_r,P_{G_\theta}) \\ & \leq\frac{4C_{Q_x}(\alpha_G)B_xU_\omega\sqrt{3k}}{\sqrt{n}}+\frac{4C_{Q_z}(\alpha_G)U_\omega U_\theta B_z\sqrt{3(k+l-1)}}{\sqrt{m}} \\ & \quad+U_\omega\sqrt{\log\frac{1}{\delta}}\left(\frac{4C_{Q_x}(\alpha_G)B_x}{\sqrt{2n}}+\frac{4C_{Q_z}(\alpha_G)B_zU_\theta}{\sqrt{2m}}\right), \end{aligned} \tag{19}$$

where the parameters $U_\omega:=M_k\prod_{i=1}^{k-1}(M_iR_i)$ and $U_\theta:=N_l\prod_{j=1}^{l-1}(N_jS_j)$, $Q_x:=U_\omega B_x$, $Q_z:=U_\omega U_\theta B_z$, and

$$C_h(\alpha):=\begin{cases}\sigma(h)\sigma(-h)^{\frac{\alpha-1}{\alpha}}, & \alpha\in(0,1]\\ \left(\frac{\alpha-1}{2\alpha-1}\right)^{\frac{\alpha-1}{\alpha}}\frac{\alpha}{2\alpha-1}, & \alpha\in(1,\infty).\end{cases} \tag{20}$$



(a)

(b)

Fig. 2. (a) Plot of mode coverage over epochs for (α_D,α_G)-GAN training with the **saturating** objectives in (5). Fixing $\alpha_G=1$, we compare $\alpha_D=1$ (vanilla GAN) with $\alpha_D=0.2$. Placed above this plot are 2D visuals of the generated samples (in black) at different epochs; these show that both GANs successfully capture the ring-like structure, but the vanilla GAN fails to maintain the ring over time. We illustrate the discriminator output in the same visual as a heat map to show that the $\alpha_D=1$ discriminator exhibits more confident predictions (tending to 0 or 1), which in turn subjects G to vanishing and exploding gradients when its objective $\log(1-D)$ saturates as $D\to 0$ and diverges as $D\to 1$, respectively. This combination tends to repel the generated data when it approaches the real data, thus freezing any significant weight update in the future. In contrast, the less confident predictions of the $(0.2,1)$-GAN create a smooth landscape for the generated output to descend towards the real data. (b) Plot of success and failure rates over 200 seeds for a range of α_D values with $\alpha_G=1$ for the **saturating** (α_D,α_G)-GAN on the 2D-ring, which underscores the stability of $(\alpha_D<1,\alpha_G)$-GANs relative to vanilla GAN.

The proof is similar to that of [8, Theorem 3] (and also [19, Theorem 1]). We observe that (19) does not depend on α_D, an artifact of the proof techniques used, and is therefore most likely not the tightest bound possible. See [18, Appendix C] for proof details.

When $\alpha_D=\alpha_G=\infty$, (8) reduces to the total variation distance (up to a constant) [7, Theorem 2], and (14) simplifies to the loss-inclusive NN distance $d^{\ell}_{\mathcal{F}_{nn}}(\cdot,\cdot)$ defined in [8, eq. (13)] with $\phi(\cdot)=-\ell_\alpha(1,\cdot)$ and $\psi(\cdot)=-\ell_\alpha(0,\cdot)$ for $\alpha=\infty$. We consider a slightly modified version of this quantity with an added constant to ensure nonnegativity (more details in [18, Appendix D]). For brevity, we henceforth denote this as $d^{\ell_\infty}_{\mathcal{F}_{nn}}(\cdot,\cdot)$. As in [19], suppose the generator's class $\{G_\theta\}_{\theta\in\Theta}$ is rich enough such that the generator G_θ can learn the real distribution P_r and that the number m of training samples in S_z scales faster than the number n of samples in S_x[1]. Then $\inf_{\theta\in\Theta} d^{\ell_\infty}_{\mathcal{F}_{nn}}(P_r,P_{G_\theta})=0$, so the estimation error simplifies to the single term $d^{\ell_\infty}_{\mathcal{F}_{nn}}(P_r,P_{G_{\hat{\theta}^*}})$. Furthermore, the upper bound in (19) reduces to $O(c/\sqrt{n})$ for some constant c (note that, in (20), $C_h(\infty)=1/4$). In addition to the above assumptions, also assume the activation functions r_i for $i\in[1:k-1]$ are either strictly increasing or ReLU. For the above setting, we derive a matching min-max lower bound (up to a constant multiple) on the estimation error.

Theorem 4. *For the setting above, let $\hat{P}_n$ be an estimator of P_r learned using the training samples $S_x=\{X_i\}_{i=1}^n$. Then,*

$$\inf_{\hat{P}_n}\sup_{P_r\in\mathcal{P}(\mathcal{X})}\mathbb{P}\left\{d^{\ell_\infty}_{\mathcal{F}_{nn}}(\hat{P}_n,P_r)\geq\frac{C(\mathcal{P}(\mathcal{X}))}{\sqrt{n}}\right\}>0.24,$$

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

$$C(\mathcal{P}(\mathcal{X}))=\frac{\log(2)}{20}\Big[\sigma(M_k r_{k-1}(\ldots r_1(M_1B_x)) \\ -\sigma(M_k r_{k-1}(\ldots r_1(-M_1B_x))\Big]. \quad (21)$$

Proof sketch. We prove that $d^{\ell_\infty}_{\mathcal{F}_{nn}}$ is a semi-metric. The remainder of the proof is similar to that of [19, Theorem 2]. A detailed proof is in [18, Appendix D].

III. Illustration of Results

In this section, we compare (α_D,α_G)-GAN to two state-of-the-art GANs, namely the vanilla GAN (i.e., the $(1,1)$-GAN) and LSGAN [15], on two datasets: (i) a synthetic dataset generated by a two-dimensional, ring-shaped Gaussian mixture distribution (2D-ring) [24] and (ii) the Stacked MNIST image dataset [25]. For each dataset and different GAN objectives, we report several metrics that encapsulate the stability of GAN training over hundreds of random seeds. This allows us to clearly showcase the potential for tuning (α_D,α_G) to obtain stable and robust solutions for image generation.

A. 2D Gaussian Mixture Ring

The 2D-ring is an oft-used synthetic dataset for evaluating GANs. We draw samples from a mixture of 8 equal-prior Gaussian distributions, indexed $i\in\{1,2,\ldots,8\}$ with a mean of $(\cos(2\pi i/8),\ \sin(2\pi i/8))$ and variance 10^{-4}. We generate 50,000 training and 25,000 testing samples; additionally, we generate the same number of 2D latent Gaussian noise vectors.

Both the D and G networks have 4 fully-connected layers with 200 and 400 units, respectively. We train for 400 epochs with a batch size of 128, and optimize with Adam [26] and a learning rate of 10^{-4} for both models. We consider three distinct settings that differ in the objective functions as: **(i)** (α_D,α_G)-GAN in (5); **(ii)** NS (α_D,α_G)-GAN's in (5a), (10); **(iii)** LSGAN with the 0-1 binary coding scheme (see [18, Appendix E] for details).

For every setting listed above, we train our models on the 2D-ring dataset for 200 random state seeds, where each seed contains different weight initializations for D and G. Ideally, a stable method will reflect similar performance across randomized initializations and also over training epochs; thus, we explore how GAN training performance for each setting varies

[1] Since the noise distribution P_Z is known, one can generate an arbitrarily large number m of noise samples.

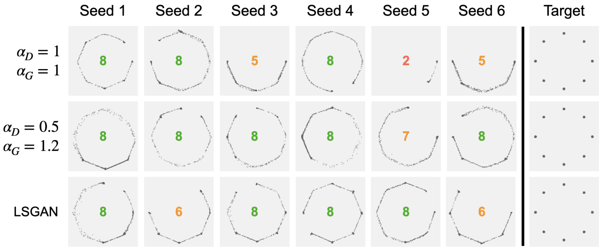
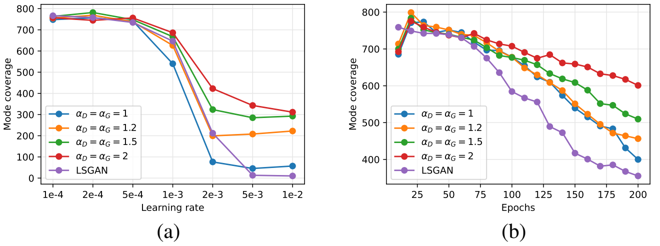


Fig. 3. Generated samples from two (α_D,α_G)-GANs trained with the **NS** objectives in (5a), (10), as well as the LSGAN. We provide 6 seeds to illustrate the stability in performance for each GAN across multiple runs.



(a) (b)

Fig. 4. Mode coverage vs. (a) varied learning rates with fixed epoch number $(=50)$ and (b) varied epoch numbers with fixed learning rate $(=5\times10^{-4})$ for different GANs, underscoring the vanilla GAN's hyperparameter sensitivity.

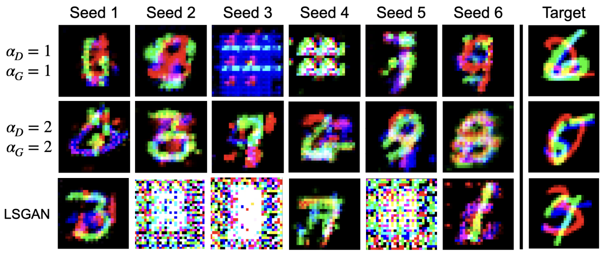


Fig. 5. Generated Stacked MNIST samples from three GANs over 6 seeds when trained for 200 epochs with a learning rate of 5×10^{-4}.

across seeds and epochs. Our primary performance metric is *mode coverage*, defined as the number of Gaussians (0-8) that contain a generated sample within 3 standard deviations of its mean. A score of 8 conveys successful training, while a score of 0 conveys a significant GAN failure; on the other hand, a score in between 0 and 8 may be indicative of common GAN issues, such as mode collapse or failure to converge.

For the saturating setting, the improvement in stability of the $(0.2,1)$-GAN relative to the vanilla GAN is illustrated in Fig. 2 as detailed in the caption. In fact, vanilla GAN completely fails to converge to the true distribution 30% of the time while succeeding only 46% of the time. In contrast, the (α_D,α_G)-GAN with $\alpha_D<1$ learns a more stable G due to a less confident D (see also Fig. 2(a)). For example, the $(0.3,1)$-GAN success and failure rates improve to 87% and 2%, respectively. Finally, for the NS setting in Fig. 3, we find that tuning α_D and α_G yields more consistently stable outcomes than vanilla and LSGANs. Mode coverage rates over 200 seeds for saturating (Tables I and II) and NS (Table III) are in [18, Appendix E].

B. Stacked MNIST

The Stacked MNIST dataset is an enhancement of MNIST [27] as it contains images of size $3\times28\times28$, where each RGB channel is a 28×28 image randomly sampled from MNIST. Stacked MNIST is a popular choice for image generation since its use of 3 channels allows for a total of $10^3=1000$ modes, as opposed to the 10 modes (digits) in MNIST, which makes the latter much easier for GANs to learn. We generate 100,000 training samples, 25,000 testing samples, and the same number of 100-dimension latent Gaussian noise vectors.

We use the DCGAN architecture [28] for training, which uses deep convolutional neural networks (CNN) for both D and G (details in Tables IV, V [18, Appendix E]). As in other works, we focus solely on the NS setting using appropriate objective functions for vanilla GAN, (α_D,α_G)-GAN, and LSGAN. We compute the mode coverage of each trial by feeding each generated sample to a 1000-mode CNN classifier. The classifier is obtained by pretraining on MNIST to achieve 99.5% validation accuracy. We also consider a range of settings for two key hyperparameters: the number of epochs and learning rate for Adam optimization. Each combination of objective function, number of epochs, and learning rate is trained for 100 seeds; this allows us to report the *mean mode coverage*. We also report the mean Fréchet Inception Distance (FID)[2].

In Fig. 4(a) and 4(b), we empirically demonstrate the dependence of mode coverage on learning rate and number of epochs, respectively (FID plots are in [18, Appendix E-C]). Achieving robustness to hyperparameter initialization is highly desirable in the unsupervised GAN setting as the choices that facilitate steady model convergence are not easily determined without prior mode knowledge. Observing the mode coverage of different (α_D,α_G)-GANs, we find that as the learning rate or training time increases, the performance of both vanilla GAN and LSGAN deteriorates faster than a GAN with $\alpha_D=\alpha_G>1$ (see [18, Appendix E] for additional details that motivate this choice). Finally, as shown in Fig. 5, we observe that the outputs of (α_D,α_G)-GAN are more consistent and accurate across multiple seeds, relative to LSGAN and vanilla GAN.

IV. Concluding Remarks

We have introduced a dual-objective GAN formulation, focusing in particular on using α-loss for both players' objectives. Our results highlight the value of tuning α in alleviating training instabilities and enhancing robustness to learning rates and training epochs, hyperparameters whose optimal values are generally not known *a priori*. Generalization guarantees of (α_D,α_G)-GANs is a natural extension to study. An equally important problem is to evaluate if our observations hold more broadly, including, when the training data is noisy [30].

[2] FID is an unsupervised similarity metric between the real and generated feature distributions extracted by InceptionNet-V3 [29].

References

[1] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, 2014, p. 2672–2680.

[2] J. Lin, "Divergence measures based on the Shannon entropy," *IEEE Transactions on Information Theory*, vol. 37, no. 1, pp. 145–151, 1991.

[3] S. Nowozin, B. Cseke, and R. Tomioka, "f-GAN: Training generative neural samplers using variational divergence minimization," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, 2016, p. 271–279.

[4] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proceedings of the 34th International Conference on Machine Learning*, vol. 70, 2017, pp. 214–223.

[5] B. K. Sriperumbudur, K. Fukumizu, A. Gretton, B. Schölkopf, and G. R. Lanckriet, "On the empirical estimation of integral probability metrics," *Electronic Journal of Statistics*, vol. 6, pp. 1550–1599, 2012.

[6] T. Liang, "How well generative adversarial networks learn distributions," *arXiv preprint arXiv:1811.03179*, 2018.

[7] G. R. Kurri, T. Sypherd, and L. Sankar, "Realizing GANs via a tunable loss function," in *IEEE Information Theory Workshop (ITW)*, 2021, pp. 1–6.

[8] G. R. Kurri, M. Welfert, T. Sypherd, and L. Sankar, "α-GAN: Convergence and estimation guarantees," in *IEEE International Symposium on Information Theory (ISIT)*, 2022, pp. 276–281.

[9] T. Sypherd, M. Diaz, L. Sankar, and P. Kairouz, "A tunable loss function for binary classification," in *IEEE International Symposium on Information Theory*, 2019, pp. 2479–2483.

[10] T. Sypherd, M. Diaz, J. K. Cava, G. Dasarathy, P. Kairouz, and L. Sankar, "A tunable loss function for robust classification: Calibration, landscape, and generalization," *IEEE Transactions on Information Theory*, vol. 68, no. 9, pp. 6021–6051, 2022.

[11] F. Österreicher, "On a class of perimeter-type distances of probability distributions," *Kybernetika*, vol. 32, no. 4, pp. 389–393, 1996.

[12] F. Liese and I. Vajda, "On divergences and informations in statistics and information theory," *IEEE Transactions on Information Theory*, vol. 52, no. 10, pp. 4394–4412, 2006.

[13] M. Arjovsky and L. Bottou, "Towards principled methods for training generative adversarial networks," *arXiv preprint arXiv:1701.04862*, 2017.

[14] M. Wiatrak, S. V. Albrecht, and A. Nystrom, "Stabilizing generative adversarial networks: A survey," *arXiv preprint arXiv:1910.00927*, 2019.

[15] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley, "Least squares generative adversarial networks," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.

[16] H. Bhatia, W. Paul, F. Alajaji, B. Gharesifard, and P. Burlina, "Least kth-order and Rényi generative adversarial networks," *Neural Computation*, vol. 33, no. 9, pp. 2473–2510, 2021.

[17] B. Poole, A. A. Alemi, J. Sohl-Dickstein, and A. Angelova, "Improved generator objectives for gans," *arXiv preprint arXiv:1612.02780*, 2016.

[18] M. Welfert, K. Otstot, G. R. Kurri, and L. Sankar, "(α_D,α_G)-GANs: Addressing GAN training instabilities via dual objectives," *arXiv preprint arXiv:2302.14320*, 2023.

[19] K. Ji, Y. Zhou, and Y. Liang, "Understanding estimation and generalization error of generative adversarial networks," *IEEE Transactions on Information Theory*, vol. 67, no. 5, pp. 3114–3129, 2021.

[20] S. Arora, R. Ge, Y. Liang, T. Ma, and Y. Zhang, "Generalization and equilibrium in generative adversarial nets (GANs)," in *Proceedings of the 34th International Conference on Machine Learning*, vol. 70, 2017, pp. 224–232.

[21] B. Neyshabur, R. Tomioka, and N. Srebro, "Norm-based capacity control in neural networks," in *Conference on Learning Theory*. PMLR, 2015, pp. 1376–1401.

[22] T. Salimans and D. P. Kingma, "Weight normalization: A simple reparameterization to accelerate training of deep neural networks," *Advances in neural information processing systems*, vol. 29, pp. 901–909, 2016.

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

[24] A. Srivastava, L. Valkov, C. Russell, M. U. Gutmann, and C. Sutton, "VEEGAN: Reducing mode collapse in GANs using implicit variational learning," in *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[25] Z. Lin, A. Khetan, G. Fanti, and S. Oh, "PacGAN: The power of two samples in generative adversarial networks," *IEEE Journal on Selected Areas in Information Theory*, vol. 1, no. 1, pp. 324–335, 2020.

[26] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[27] L. Deng, "The MNIST database of handwritten digit images for machine learning research," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 141–142, 2012.

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

[29] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, G. Klambauer, and S. Hochreiter, "GANs trained by a two time-scale update rule converge to a Nash equilibrium," *arXiv preprint arXiv:1706.08500*, 2017.

[30] S. Nietert, Z. Goldfeld, and R. Cummings, "Outlier-robust optimal transport: Duality, structure, and statistical analysis," in *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, 2022, pp. 11 691–11 719.