DEMYSTIFYING MMD GANS

Anonymous authors
Paper under double-blind review

ABSTRACT

We investigate the training and performance of generative adversarial networks using the Maximum Mean Discrepancy (MMD) as critic, termed MMD GANs. As our main theoretical contribution, we clarify the situation with bias in GAN loss functions raised by recent work: we show that gradient estimators used in the optimization process for both MMD GANs and Wasserstein GANs are unbiased, while gradients of the generator's theoretical loss are biased in both cases. We discuss the issue of kernel choice for the MMD critic, and characterize the kernel corresponding to the energy distance used for the Cramér GAN critic. Being an integral probability metric, the MMD benefits from training strategies recently developed for Wasserstein GANs. In experiments, the MMD GAN is able to employ a smaller critic network than the Wasserstein GAN, resulting in a simpler and faster-training algorithm with matching performance. We also propose an improved measure of GAN convergence, the *Kernel Inception Distance*, and show how to use it to dynamically adapt learning rates during GAN training.

1 INTRODUCTION

Generative Adversarial Networks (GANs; Goodfellow et al., 2014) provide a powerful method for general-purpose generative modeling of datasets. Given examples from some distribution, a GAN attempts to learn a *generator* function, which maps from some fixed noise distribution to samples that attempt to mimic a reference or target distribution. The generator is trained to trick a *discriminator*, or *critic*, which tries to distinguish between generated and target samples.

This alternative to standard maximum likelihood approaches for training generative models has brought about a rush of interest over the past several years. Likelihoods do not necessarily correspond well to sample quality (Theis et al., 2016), and GAN-type objectives focus much more on producing plausible samples, as illustrated particularly directly by Danihelka et al. (2017). This class of models has recently led to many impressive examples of image generation (e.g. Huang et al., 2017a;b; Jin et al., 2017; Zhu et al., 2017).

GANs are, however, notoriously tricky to train (Salimans et al., 2016). This might be understood in terms of the discriminator class. Goodfellow et al. (2014) showed that, when the discriminator is trained to optimality among a rich enough function class, the generator network attempts to minimize the Jensen-Shannon divergence between the generator and target distributions. This result has been extended to general f-divergences by Nowozin et al. (2016). According to Arjovsky & Bottou (2017), however, it is likely that both the GAN and reference probability measures are supported on manifolds within a larger space, as occurs for the set of images in the space of possible pixel values. These manifolds might not intersect at all, or at best might intersect on sets of measure zero. In this case, the Jensen-Shannon divergence is constant, and the KL and reverse-KL divergences are infinite, meaning that they provide no useful gradient for the generator to follow. This helps to explain some of the instability of GAN training.

The lack of sensitivity to distance, meaning that nearby but non-overlapping regions of high probability mass are not considered similar, is a long-recognized problem for KL divergence-based discrepancy measures (e.g. Gneiting & Raftery, 2007, Section 4.2). It is natural to address this problem using Integral Probability Metrics (IPMs; Müller, 1997): these measure the distance between probability measures via the largest discrepancy in expectation over a class of "well behaved" witness functions. Thus, IPMs are able to signal proximity in the probability mass of the generator and reference distributions. (Section 2 describes this framework in more detail.)

Arjovsky et al. (2017) proposed to use the Wasserstein distance between distributions as the discriminator, which is an integral probability metric constructed from the witness class of 1-Lipschitz functions. To implement the Wasserstein critic, Arjovsky et al. originally proposed weight clipping of the discriminator network, to enforce k-Lipschitz smoothness. Gulrajani et al. (2017) improved on this result by directly constraining the gradient of the discriminator network at points between the generator and reference samples. This new Wasserstein GAN implementation, called WGAN-GP, is more stable and easier to train.

A second integral probability metric used in GAN variants is the maximum mean discrepancy (MMD), for which the witness function class is a unit ball in a reproducing kernel Hilbert space (RKHS). Generative adversarial models based on minimizing the MMD were first considered by Li et al. (2015) and Dziugaite et al. (2015). These works optimized a generator to minimize the MMD with a fixed kernel, either using a generic kernel on image pixels or by modeling autoencoder representations instead of images directly. Sutherland et al. (2017) instead minimized the statistical power of an MMD-based test with a fixed kernel. Such approaches struggle with complex natural images, where pixel distances are of little value, and fixed representations can easily be tricked (as in the adversarial examples of Szegedy et al. (2014)).

Adversarial training of the MMD loss is thus the obvious choice. Here the kernel MMD is defined on the output of a convolutional network, which is trained adversarially. Recent notable work has made use of the IPM representation of the MMD to employ the same witness function regularization strategies as Arjovsky et al. (2017); Gulrajani et al. (2017), effectively corresponding to an additional constraint on the MMD function class. Without such constraints, the convolutional features are unstable and difficult to train (Sutherland et al., 2017). Li et al. (2017b) used the weight clipping strategy of Arjovsky et al., with additional constraints to encourage the kernel distribution embeddings to be injective.[1] In light of the observations by Gulrajani et al., however, we use a gradient constraint on the MMD witness function in the present work (see Sections 2.1 and 2.2).[2] Bellemare et al. (2017)'s method, the Cramér GAN, also used the gradient constraint strategy of Gulrajani et al. in their discriminator network. As we discuss in Section 2.3, the Cramér GAN discriminator is related to the energy distance, which is an instance of the MMD (Sejdinovic et al., 2013), and which can therefore use a gradient constraint on the witness function. Note, however, that there are important differences between the Cramér GAN critic and the energy distance, which make it more akin to the optimization of a scoring rule: we provide further details in Appendix A. Weight clipping and gradient constraints are not the only approaches possible: variance features (Mroueh et al., 2017) and constraints (Mroueh & Sercu, 2017) can work, as can other optimization strategies (Berthelot et al., 2017; Li et al., 2017a).

Given that both the Wasserstein distance and the MMD are integral probability metrics, it is of interest to consider how they differ when used in GAN training. Bellemare et al. (2017) pointed out a particular flaw of the Wasserstein distance as an objective: that gradients of both the duality-based estimator and of the alternative estimator of Genevay et al. (2017) are biased. Optimization over biased gradients can lead to incorrect parameter values, even in expectation; Bellemare et al. provide an explicit example where this occurs. As our main theoretical contribution, we show in Section 3 that the natural maximum mean discrepancy estimator has unbiased gradients, and therefore does not suffer from this problem.

There are additional empirical advantages of the MMD-GAN over the Wasserstein GAN. Certainly we would not expect the MMD on its own to perform well on raw image data, since these data lie on a low dimensional manifold embedded in a higher dimensional pixel space. Once the images are mapped through appropriately trained convolutional layers, however, they can follow a much simpler distribution with broader support across the mapped domain: a phenomenon also observed in autoencoders (Bengio et al., 2013). In this setting, the MMD with characteristic kernels (Sriperumbudur et al., 2010) shows strong discriminative peformance between distributions. To achieve comparable performance, a purely convolutional WGAN requires many more layers in the critic. In our experiments (Section 5), we find that MMD GANs achieve the same generator performance as WGAN-GPs with smaller discriminator networks, resulting in GANs with fewer parameters and

[1] When distribution embeddings are injective, the critic is guaranteed to be able to distinguish any two distributions, given an infinite number of samples.

[2] Li et al. also did this in a later revision of their paper, independent of this work.

easier training. Thus, the MMD-GAN discriminator can be understood as a hybrid model that plays to the strengths of both the initial convolutional mappings and the kernel layer that sits on top.

2 LOSSES AND WITNESS FUNCTIONS

We begin with a review of the MMD and relate it to the loss functions used by other GAN variants. Through its interpretation as an integral probability metric, we show that the gradient penalty of Gulrajani et al. (2017) applies to the MMD GAN.

2.1 MAXIMUM MEAN DISCREPANCY AND WITNESS FUNCTIONS

We consider a random variable X with probability measure $\mathbb{P}$, which we associate with the generator, and a second random variable Y with probability measure $\mathbb{Q}$, which we associate with the reference sample that we wish to learn. Our goal is to measure the distance from $\mathbb{P}$ to $\mathbb{Q}$ using samples drawn independently from each distribution.

The maximum mean discrepancy is a metric on probability measures (Gretton et al., 2012), which falls within the family of integral probability metrics (Müller, 1997); this family includes the Wasserstein and Kolmogorov metrics, but not for instance the KL or χ^2 divergences. Integral probability metrics make use of a class of *witness functions* to distinguish between $\mathbb{P}$ and $\mathbb{Q}$, choosing the function with the largest discrepancy in expectation over $\mathbb{P}, \mathbb{Q}$,

$$\mathcal{D}_{\mathcal{F}}(\mathbb{P}, \mathbb{Q}) = \sup_{f \in \mathcal{F}} \mathbb{E}_{\mathbb{P}}\, f(X) - \mathbb{E}_{\mathbb{Q}}\, f(Y). \tag{1}$$

The particular witness function class $\mathcal{F}$ determines the probability metric. For example, the Wasserstein-1 metric is defined using the 1-Lipschitz functions, the total variation by functions with absolute value bounded by 1, and the Kolmogorov metric using the functions of bounded variation 1. For more on this family of distances, see e.g. (Sriperumbudur et al., 2009b).

In this work, our witness function class $\mathcal{F}$ will be the unit ball in a reproducing kernel Hilbert space $\mathcal{H}$, with positive definite kernel $k(x, x')$. The key property of a reproducing kernel Hilbert space is the reproducing property: for all $f \in \mathcal{H}$, $f(x) = \langle f, k(x, \cdot)\rangle_{\mathcal{H}}$. We define the *mean embedding* of the probability measure $\mathbb{P}$ as the element $\mu_{\mathbb{P}} \in \mathcal{H}$ such that $\mathbb{E}_{\mathbb{P}}\, f(X) = \langle f, \mu_{\mathbb{P}}\rangle_{\mathcal{H}}$; it is given by $\mu_{\mathbb{P}} = \mathbb{E}_{X\sim\mathbb{P}}\, k(\cdot, X)$.[3]

The maximum mean discrepancy (MMD) is defined as (1) with $\mathcal{F}$ the unit ball in $\mathcal{H}$,

$$MMD(\mathbb{P}, \mathbb{Q}; \mathcal{H}) = \sup_{f \in \mathcal{H}, \|f\|_{\mathcal{H}} \leq 1} \mathbb{E}_{\mathbb{P}}\, f(X) - \mathbb{E}_{\mathbb{Q}}\, f(Y).$$

The witness function f^* that attains the supremum has a straightforward expression (Gretton et al., 2012, Section 2.3),

$$f^*(x) \propto \mathbb{E}_{\mathbb{P}}\, k(X, x) - \mathbb{E}_{\mathbb{Q}}\, k(Y, x). \tag{2}$$

Given samples $X = \{x_i\}_{i=1}^m$ drawn i.i.d. from $\mathbb{P}$, and $Y = \{y_j\}_{j=1}^n$ drawn i.i.d. from $\mathbb{Q}$, the empirical witness function is

$$\hat{f}(x) \propto \frac{1}{m}\sum_{i=1}^{m} k(x_i, x) - \frac{1}{n}\sum_{i=1}^{n} k(y_i, x), \tag{3}$$

and an unbiased estimator of the squared MMD is (Gretton et al., 2012, Lemma 6)

$$MMD_u^2(X, Y) = \frac{1}{m(m-1)}\sum_{i\neq j}^{m} k(x_i, x_j) + \frac{1}{n(n-1)}\sum_{i\neq j}^{n} k(y_i, y_j) - \frac{2}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n} k(x_i, y_j). \tag{4}$$

When the kernel is characteristic (Sriperumbudur et al., 2010; 2011), the embedding $\mu_{\mathbb{P}}$ is injective (i.e., associated uniquely with $\mathbb{P}$). Perhaps the best-known characteristic kernel is the exponentiated quadratic kernel, also known as the Gaussian RBF kernel,

$$k_\sigma^{rbf}(x, y) = \exp\left(-\frac{1}{2\sigma^2}\|x - y\|^2\right). \tag{5}$$

[3] This is well defined for Bochner-integrable kernels (Steinwart & Christmann, 2008, Definition A.5.20), for which $E_{\mathbb{P}}\, \|k(x, \cdot)\|_{\mathcal{H}} < \infty$ for the class of probability measures $\mathbb{P}$ being considered. For bounded kernels, the condition always holds, but for unbounded kernels, additional conditions on the moments might apply.

Both the kernel and its derivatives decay exponentially, however, causing significant problems in high dimensions, and especially when used in gradient-based representation learning. The rational quadratic kernel

$$k_\alpha^{rq}(x, y) = \left(1 + \frac{\|x - y\|^2}{2\alpha}\right)^{-\alpha} \tag{6}$$

with $\alpha > 0$ corresponds to a scaled mixture of exponentiated quadratic kernels, with a $\mathrm{Gamma}(\alpha, 1)$ prior on the inverse lengthscale (Rasmussen & Williams, 2006, Section 4.2). This kernel will be the mainstay of our experiments, as its tail behaviour is much superior to that of the exponentiated quadratic kernel; it is also characteristic.

2.2 Witness function and gradient penalties

The MMD has been a popular choice for the role of a critic in a GAN. This idea was proposed simultaneously by Dziugaite et al. (2015) and Li et al. (2015), with numerous recent follow-up works (Sutherland et al., 2017; Liu, 2017; Li et al., 2017b; Bellemare et al., 2017). As a key strategy in these recent works, the MMD of (4) is not computed directly on the samples; rather, the samples first pass through a mapping function h, generally a convolutional network. Note that we can think of this either as the MMD with kernel k on features $h(x)$, or simply as the MMD with kernel $\kappa(x, y) = k(h(x), h(y))$. The challenge is to learn the features h so as to maximize the MMD, without causing the critic to collapse to a trivial answer early in training.

Bearing in mind that the MMD is an integral probability metric, strategies developed for training the Wasserstein GAN critic can be directly adopted for training the MMD critic. Li et al. (2017b) employed the weight clipping approach of Arjovsky et al. (2017), though they motivated it using different considerations. Gulrajani et al. (2017) found a number of issues with weight clipping, however: it oversimplifies the loss functions given standard architectures, the gradient decays exponentially as we move up the network, and it seems to require the use of slower optimizers such as RMSProp rather than standard approaches such as Adam (Kingma & Ba, 2015).

It thus seems preferable to adopt Gulrajani et al.'s proposal of regularising the critic witness (3) by constraining its gradient norm to be nearly 1 along randomly chosen convex combinations of generator and reference points, $\alpha x_i + (1 - \alpha) y_j$ for $\alpha \sim \mathrm{Uniform}(0, 1)$. This was motivated by the observation that the Wasserstein witness satisfies this property (their Lemma 1), but perhaps its main benefit is one of regularization: if the critic function becomes too flat anywhere between the samples, the generator cannot easily follow its gradient. We will thus follow this approach, as did Bellemare et al. (2017), whose model we describe next.

2.3 The energy distance and associated MMD

Liu (2017) and Bellemare et al. (2017, Section 4) proposed to use the energy distance as the critic in an adversarial network. The energy distance (Székely & Rizzo, 2004; Lyons, 2013) is a measure of divergence between two probability measures, defined as

$$\mathcal{D}_e(\mathbb{P}, \mathbb{Q}) = -\frac{1}{2}\,\mathbb{E}_{\mathbb{P}}\,\rho(X, X') - \frac{1}{2}\,\mathbb{E}_{\mathbb{Q}}\,\rho(Y, Y') + \mathbb{E}_{\mathbb{P},\mathbb{Q}}\,\rho(X, Y), \tag{7}$$

where $\mathbb{E}_{\mathbb{P}}\,\rho(X, X')$ is an expectation over two independent samples from the generator $\mathbb{P}$ (likewise, Y and Y' are independent samples from the reference $\mathbb{Q}$), and $\rho(x, y)$ is a semimetric of negative type.[4] We will focus on the case $\rho_\beta(x, y) = \|x - y\|^\beta$ for $0 < \beta \leq 2$. When $\beta \leq 1$, ρ_β is a metric.

It has been shown (Sejdinovic et al., 2013, Lemma 12) that the energy distance is an instance of the maximum mean discrepancy, where the corresponding distance-induced kernel family for the distance (7) is

$$k_{\rho, z_0}^{dist}(x, y) = \frac{1}{2}\left[\rho(x, z_0) + \rho(y, z_0) - \rho(x, y)\right] \tag{8}$$

for any choice of $z_0 \in \mathcal{X}$ (often $z_0 = 0$ is chosen to simplify notation, which corresponds to a fractional Brownian motion kernel; Sejdinovic et al., 2013, Example 15). Note that the resulting

[4] ρ must satisfy the properties of a metric besides the triangle inequality, and for all $n \geq 2$, $x_1, \ldots, x_n \in \mathcal{X}$, and $a_1, \ldots, a_n \in \mathbb{R}$ with $\sum_i a_i = 0$, $\sum_{i=1}^n \sum_{j=1}^n a_i a_j \rho(x_i, x_j) \leq 0$.

kernel is *not* translation invariant. It is characteristic (when $\beta < 2$), and the MMD is well-defined for a class of distributions $\mathcal{P}$ that satisfy a moment condition (Sejdinovic et al., 2013, Remark 21), namely that there exists $z_0 \in \mathcal{X}$ such that

$$\int \rho(z, z_0) \, \mathrm{d}\mathbb{P}(z) < \infty \quad \forall \mathbb{P} \in \mathcal{P}.$$

To apply the regularization strategy of Gulrajani et al. (2017) in training the critic of an adversarial network, we need to compute the form taken by the witness function (2) given the kernel (8). Bearing in mind that

$$\mathbb{E}_{\mathbb{P}} \, k(x, X) = \rho(x, z_0) + \mathbb{E}_{\mathbb{P}} \, \rho(X, z_0) - \mathbb{E}_{\mathbb{P}} \, \rho(x, X),$$

where the second term of the above expression is constant, and substituting into (2), we have

$$\begin{aligned} f^*(x) &\propto \rho(x, z_0) - \mathbb{E}_{\mathbb{P}} \, \rho(x, X) - d(x, z_0) + \mathbb{E}_{\mathbb{Q}} \, \rho(x, Y) + C \\ &= \mathbb{E}_{\mathbb{Q}} \, \rho(x, Y) - \mathbb{E}_{\mathbb{P}} \, \rho(x, X) + C. \end{aligned}$$

This is in agreement with Bellemare et al.'s function f^* (their page 5), albeit via a different argument (note that the function g^* in their footnote 4 is not correct, as it is missing the constant terms).

We now turn to the divergence implemented by the critic in Bellemare et al.'s Algorithm 1, which is somewhat different from the energy distance (7). The Cramér GAN witness function is defined as

$$f_c(x) = \mathbb{E}_{\mathbb{P}} \, \rho(x, X) - \rho(x, 0), \tag{9}$$

which is regularized using Gulrajani et al.'s gradient constraint. The expected *surrogate loss* associated with this witness function, and used for the Cramér critic, is

$$\mathcal{D}_c(\mathbb{P}, \mathbb{Q}) = \mathbb{E}_{\mathbb{P}} \, \rho(X, X') + \mathbb{E}_{\mathbb{Q}} \, \rho(Y, 0) - \mathbb{E}_{\mathbb{P}} \, \rho(X, 0) - \mathbb{E}_{\mathbb{P},\mathbb{Q}} \, \rho(X', Y). \tag{10}$$

In brief, Y' in (7) is replaced by the origin: it is explained that this is necessary for instance in the conditional case, where two independent reference samples Y, Y' are not available. Unfortunately, following this change, it becomes straightforward to define $\mathbb{P}$ and $\mathbb{Q}$ which are different, yet have an expected $\mathcal{D}_c(\mathbb{P}, \mathbb{Q})$ loss of zero. For example, if $\mathbb{P}$ is a point mass at the origin in $\mathbb{R}$, and $\mathbb{Q}$ is a point mass a distance t from the origin, then $\mathbb{P} \neq \mathbb{Q}$ and yet $\mathcal{D}_c(\mathbb{P}, \mathbb{Q}) = 0$ because

$$\mathbb{E}_{\mathbb{P}} \, \rho(X, X') = \mathbb{E}_{\mathbb{P}} \, \rho(X, 0) = 0, \qquad \mathbb{E}_{\mathbb{P},\mathbb{Q}} \, \rho(X', Y) = \mathbb{E}_{\mathbb{Q}} \, \rho(Y, 0) = t.$$

Nevertheless, good empirical performance has been obtained in practice for the Cramér critic, both by Bellemare et al. (2017) and in our experiments of Section 5. Our Appendix A provides some insight into this behavior by considering the Cramér critic's relationship to the score function associated with the energy distance.

2.4 Other related models

Many other GAN variants fall into the framework of IPMs (e.g. Mroueh et al., 2017; Mroueh & Sercu, 2017; Berthelot et al., 2017). Notably, although Goodfellow et al. (2014) motivated GANs as estimating the Jensen-Shannon divergence, they can also be viewed as minimizing the IPM defined by the classifier family (Arora et al., 2017; Liu et al., 2017), thus motivating applying the gradient penalty to original GANs (Fedus et al., 2017). (Liu et al., 2017) in particular studies properties of these distances.

3 Gradient bias

The issue of biased gradients in GANs was brought to prominence by Bellemare et al. (2017, Section 3), who showed bias in the gradients of the empirical Wasserstein distance for finite sample sizes, and demonstrated cases where this bias could lead to serious problems in stochastic gradient descent, even in expectation. They then claimed that the energy distance used in the Cramér GAN critic does not suffer from these problems. We will now both formalize and clarify these results.

First, Bellemare et al.'s proof that the energy distance is unbiased was incomplete: the essential step in the reasoning, the exchange in the order of the expectations and the derivatives, is simply

assumed.[5] We thus show that, under very mild assumptions about the distributions in question, the form of the network, and the kernel, one can exchange the order of expectations and derivatives:

Theorem 1. *Let $G_\psi : \mathcal{Z} \to \mathcal{X}$ and $h_\theta : \mathcal{X} \to \mathbb{R}^d$ be deep networks, with parameters $\psi \in \mathbb{R}^{m_\psi}$ and $\theta \in \mathbb{R}^{m_\theta}$, of the form defined in Appendix C.1 and satisfying Assumptions **C** and **D**. This includes almost all feedforward networks used in practice, in particular covering convolutions, max pooling, and ReLU-like activations.*

Let $\mathbb{P}$ be a distribution on $\mathcal{X}$ such that $\mathbb{E}[\|X\|^2]$ exists, and likewise $\mathbb{Z}$ a distribution on $\mathcal{Z}$ such that $\mathbb{E}[\|Z\|^2]$ exists. $\mathbb{P}$ and $\mathbb{Z}$ need not have densities.

*Let $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ be a kernel function satisfying the growth assumption Assumption **E** for some $\alpha \in [1, 2]$. All kernels considered in this paper satisfy this assumption; see the discussion after Corollary 3.*

For μ-almost all $(\psi, \theta) \in \mathbb{R}^{m_\psi + m_\theta}$, where μ is the Lebesgue measure, the function

$$(\psi, \theta) \mapsto \mathbb{E}_{\substack{X \sim \mathbb{P} \\ Z \sim \mathbb{Z}}} \left[k(h_\theta(X), h_\theta(G_\psi(Z))) \right]$$

is differentiable at (ψ, θ), and moreover

$$\mathbb{E}_{\substack{X \sim \mathbb{P} \\ Z \sim \mathbb{Z}}} \left[\partial_{\psi,\theta} k(h_\theta(X), h_\theta(G_\psi(Z))) \right] = \partial_{\psi,\theta} \mathbb{E}_{\substack{X \sim \mathbb{P} \\ Z \sim \mathbb{Z}}} \left[k(h_\theta(X), h_\theta(G_\psi(Z))) \right].$$

Thus for μ-almost all (ψ, θ),

$$\mathbb{E}_{\substack{\mathbf{X} \sim \mathbb{P}^m \\ \mathbf{Z} \sim \mathbb{Z}^n}} \left[\partial_{\psi,\theta} k(h_\theta(\mathbf{X}), h_\theta(G_\psi(\mathbf{Z}))) \right] = \partial_{\psi,\theta} \left[MMD_u^2(h_\theta(\mathbb{P}), h_\theta(G_\psi(\mathbb{Z}))) \right]. \tag{11}$$

This result is shown in Appendix C, specifically as Corollary 3 to Theorem 4, which is a quite general result about interchanging expectations and derivatives of functions of deep networks. The proof is more complex than a typical proof that derivatives and integrals can be exchanged, due to the non-differentiability of ReLU-like functions used in deep networks.

But this is not the whole story. In WGANs, the generator attempts to minimize the loss function

$$\mathcal{W}(\mathbb{P}, \mathbb{Q}) = \sup_{f : \|f\|_L \leq 1} \mathbb{E}_{X \sim \mathbb{P}} f(X) - \mathbb{E}_{Y \sim \mathbb{Q}} f(Y), \tag{12}$$

based on an estimate $\hat{\mathcal{W}}(\mathbf{X}, \mathbf{Y})$: first critic parameters θ are estimated on a "training set" $\mathbf{X}^{tr}$, $\mathbf{Y}^{tr}$, and then the distance is estimated on the remaining "test set" $\mathbf{X}^{te}$, $\mathbf{Y}^{te}$ as

$$\frac{1}{m^{te}} \sum_{i=1}^{m^{te}} f_\theta(X_i) - \frac{1}{n^{te}} \sum_{j=1}^{n^{te}} f_\theta(Y_j). \tag{13}$$

(See Appendix B.4 for an explanation of the effective train-test split in training GANs.) Theorem 2, in Appendix B, show that this $\hat{\mathcal{W}}$ is biased (see also Appendix B.4). This further almost certainly implies that $\nabla \hat{\mathcal{W}}$ is biased as well, by Theorem 3. (Bellemare et al. rather showed gradient bias for a different estimator, $\nabla \mathcal{W}(\hat{\mathbb{P}}, \hat{\mathbb{Q}})$ for $\hat{\mathbb{P}}$ and $\hat{\mathbb{Q}}$ the empirical distributions.) Yet, for *fixed* θ, Corollary 1 shows that the estimator (13) is unbiased; it is only the procedure which first selects a θ and then evaluates (13) which is a biased estimator of (12).

The situation with MMD GANs, including energy distance-based GANs, is exactly analogous. We have (11): for almost all particular critic representations h_θ, the estimator of MMD^2 is unbiased. But the overall divergence the generator attempts to minimize is actually

$$\eta(\mathbb{P}, \mathbb{Q}) = \sup_\theta MMD^2 \left(h_\theta(\mathbb{P}), h_\theta(\mathbb{Q}) \right), \tag{14}$$

a distance previously studied by Sriperumbudur et al. (2009a) as well as Li et al. (2017b). An MMD GAN's effective estimator of $\hat{\eta}$ is also biased by Theorem 2 (see particularly Appendix B.5); by Theorem 3, its gradients are also almost certainly biased.

[5] Suppose $\hat{f}(\theta)$ is an unbiased estimator of a function $f(\theta)$, so that $\mathbb{E}\, \hat{f}(\theta) = f(\theta)$. Then, if we can exchange expectations and gradients, it is immediate that $\mathbb{E}\, \nabla \hat{f}(\theta) = \nabla\, \mathbb{E}\, \hat{f}(\theta) = \nabla f(\theta)$.

In both cases, the bias vanishes as the selection of θ becomes better; in particular, no bias is introduced by the use of a fixed (and potentially small) minibatch size, but rather by the optimization procedure and the total number of samples seen in training the discriminator.

Yet there is at least some sense in which MMD GANs might be considered "less biased" than WGANs. Optimizing the generator parameters of a WGAN while holding the critic parameters fixed is not sensible: consider, for example, $\mathbb{P}$ a point mass at the origin and $\mathbb{Q}$ a point mass at q. If $q > 0$, an optimal θ might correspond to $f(t) = t$; if we hold this f fixed, the optimal q is at $-\infty$, rather than at the truly optimal value of 0. But if we hold an MMD GAN's critic fixed and optimize the generator, we obtain the GMMN model of Li et al. (2015); Dziugaite et al. (2015), where bad solutions might seem as good as the optimum, but they can never seem arbitrarily *better*. Thus unbiased gradients of MMD_u^2 might somehow be more meaningful to the optimization process than unbiased gradients of (13); exploring and formalizing this intuition is an intriguing area for future work.

4 EVALUATION METRICS

One problem with comparing GAN models is that quantitative comparisons are difficult. Some insight can be gained by visually examining samples, but we also consider the following approaches to evaluate GAN methods.

Inception score This metric, proposed by Salimans et al. (2016), is based on the classification output $p(y \mid x)$ of the Inception model (Szegedy et al., 2016). Defined as $\exp\left(\mathbb{E}_x \operatorname{KL}(p(y \mid x) \parallel p(y))\right)$, it is highest when each image's predictive distribution has low entropy, but the marginal $p(y) = \mathbb{E}_x\, p(y \mid x)$ has high entropy. This score correlates somewhat with human judgement of sample quality on natural images, but it has some issues, especially when applied to domains which do not represent a variety of the types of classes in ImageNet. In particular, it knows nothing about the desired distribution for the model.

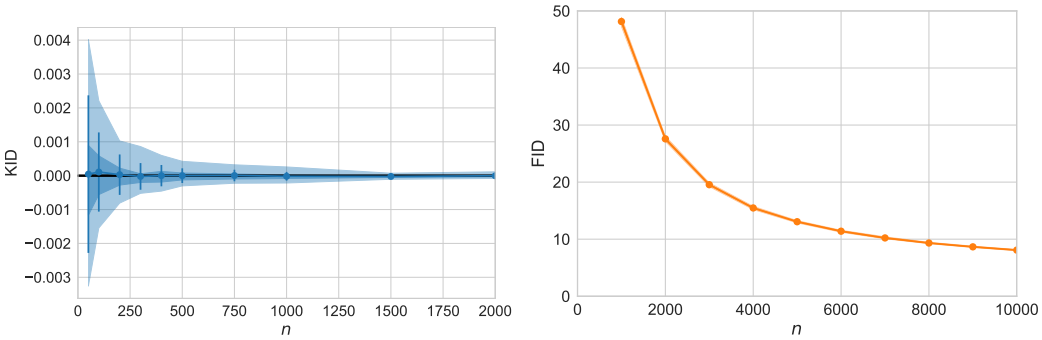
FID The *Fréchet Inception Distance*, proposed by Heusel et al. (2017), avoids some of the problems of Inception by measuring the similarity of the samples' representations in the Inception architecture (at the `pool3` layer) to those of samples from the target distribution. The FID fits a Gaussian distribution to the hidden activations for each distribution and then computes the Fréchet distance, also known as the Wasserstein-2 distance, between those Gaussians. Heusel et al. show that unlike the Inception score, the FID worsens monotonically as various types of artifacts are added to images. Note, however, that the estimator of FID is biased.[6]

KID We propose a metric similar to the FID, the *Kernel Inception Distance*, to be the squared MMD between Inception representations. We use a polynomial kernel, $k(x, y) = \left(\frac{1}{d} x^\intercal y + 1\right)^3$ where d is the representation dimension, to avoid correlations with the objective of MMD GANs as well as to avoid tuning any kernel parameters;[7] this can also be viewed as an MMD directly on input images with the kernel $K(x, y) = k(\phi(x), \phi(y))$, with ϕ the function mapping images to Inception representations. Compared to the FID, the KID has several advantages. First, it does not assume a parametric form for the distribution of activations. This is particularly sensible since the representations have ReLU activations, and therefore are not only never negative, but do not even have a density: about 2% of components in Inception representations are typically exactly zero. With the cubic kernel we use here, the KID compares skewness as well as the mean and variance. Also, unlike the FID, the KID has a simple unbiased estimator.

Figure 1 demonstrates the bias of the FID and the unbiasedness of the KID by comparing the CIFAR-10 train and test sets. The KID (Figure 1a) converges quickly to its presumed true value of 0; even for very small n, simple Monte Carlo estimates of the variance provide a reasonable measure of uncertainty. By contrast, the FID estimate (Figure 1b) does not behave so nicely: at $n = 2\,000$, when the KID estimator is essentially always 0, the FID estimator is still quite large. Even at $n = 10\,000$, the full size of the CIFAR test set, the FID still seems to be decreasing from its estimate of about 8.1 towards zero, showing the strong persistence of bias. This highlights that FID scores

[6] This is easily seen when the true FID is 0: here the estimator may be positive, but can never be negative.

[7] k is the default polynomial kernel in `scikit-learn` (Pedregosa et al., 2011).



(a) KID estimates are unbiased, and standard deviations shrink quickly even for small n.

(b) FID estimates exhibit strong bias for n even up to 10 000. All standard deviations are less than 0.5.

Figure 1: Estimates of distances between the CIFAR-10 train and test sets. Each point is based on 100 samples, estimating with replacement; sampling without replacement, and/or using the full training set, gives similar results. Lines show means, error bars standard deviations, dark colored regions a $\frac{2}{3}$ coverage interval of the samples, light colored regions a 95% interval. Note the differing n axes.

can only be compared for specific values of n; even then, there is no particular reason to expect the bias to be the same between different pairs of distributions, so comparing estimates is quite risky. Moreover, Monte Carlo estimates of the variance are extremely small even when the estimate is very far from its asymptote, so it is difficult to judge the reliability of an estimate, and practitioners may be mislead by the very low variance into thinking that they have obtained the true value.

For models on MNIST, we replace the Inception featurization with features from a LeNet-like convolutional classifier[8] (LeCun et al., 1998), but otherwise compute the scores in the same way.

We also considered the diagnostic test of Arora & Zhang (2017), which estimates the approximate number of "distinct" images produced by a GAN. The amount of subjectivity in what constitutes a duplicate image, however, makes it hard to reliably compare models based on this diagnostic. Comparisons probably need to be performed both with a certain notion of duplication in mind and by a user who does not know which models are being compared, to avoid subconscious biases; we leave further exploration of this intriguing procedure to future work.

4.1 Learning rate adaptation

In supervised deep learning, it is common practice to dynamically reduce the learning rate of an optimizer when it has stopped improving the metric on a validation set. So far, this does not seem to be common in GAN-type models, so that learning rate schedules must be tuned by hand. We propose using an adaptive scheme, based on comparing the KID score for samples from a previous iteration to that from the current iteration.

To avoid setting an explicit threshold on the change in the numerical value of the score, we use a p-value obtained from the relative similarity test of Bounliphone et al. (2016): if the test does not indicate that our current model is closer to the validation set than the model from a certain number of iterations ago at a given significance level, we mark it as a failure; when a given number of failures occur in a row, we decrease the learning rate. Bounliphone et al.'s test is for the hypothesis $MMD(\mathbb{P}, \mathbb{Q}) < MMD(\mathbb{P}', \mathbb{Q})$, and since the KID can be viewed as an MMD on image inputs, we can apply it directly. (We use the slight corrections to the asymptotic distribution of the MMD estimator given by Sutherland et al. (2017) in this test.)

[8] `github.com/tensorflow/models/blob/master/tutorials/image/mnist/convolutional.py`

5 EXPERIMENTS

We compare the quality of samples generated by MMD-GAN using various kernels with samples obtained by WGAN-GP (Gulrajani et al., 2017) and Cramér GAN (Bellemare et al., 2017) on three standard benchmark datasets: the MNIST dataset of 28×28 handwritten digits[9], the CIFAR-10 dataset of 32×32 photos (Krizhevsky, 2009), and LSUN dataset of 64×64 pictures of bedrooms Yu et al. (2015).

For all of our experiments we used the DCGAN architecture (Radford et al., 2016) for both generator and critic. For MMD losses, we used only 16 top-layer neurons in the critic; more did not seem to improve performance, except for the *distance* kernel for which 256 neurons in the top layer was advantageous. As Bellemare et al. (2017) advise use of at least 256-dimensional critic output, this enabled exact comparison between Cramér GAN and energy distance MMD, which are directly related (Section 2.3). For the generator we used the standard number of convolutional filters (64 in the second-to-last layer); for the critic, we compared networks with 16 and 64 filters in the first convolutional layer.[10]

Models with smaller critics run considerably faster: on our systems, the 16-filter networks typically ran at about twice the speed of the 64-filter ones. Note that the critic size is far more important to training runtime than the generator size: we update the critic 5 times for each generator step, and moreover the critic network is run on two batches each time we use it, one from $\mathbb{P}$ and one from $\mathbb{Q}$. Given the same architecture, all models considered here run at about the same speed.

We evaluate several MMD GAN kernel functions in our experiments. The simplest is the linear kernel: $k^{dot}(x, y) = \langle x, y \rangle$, whose MMD corresponds to the distance between means (this is somewhat similar to the *feature matching* idea of Salimans et al., 2016). We also use the exponentiated quadratic (5) and rational quadratic (6) functions, with mixtures of lengthscales,

$$k^{rbf}(x, y) = \sum_{\sigma \in \Sigma} k^{rbf}_{\sigma}(x, y), \qquad k^{rq}(x, y) = \sum_{\alpha \in \mathcal{A}} k^{rq}_{\alpha}(x, y),$$

where $\Sigma = \{2, 5, 10, 20, 40, 80\}$, $\mathcal{A} = \{.2, .5, 1, 2, 5\}$. For the latter, however, we found it advantageous to add a linear kernel to the mixture, resulting in the mixed *RQ-dot* kernel $k^{rq*} = k^{rq} + k^{dot}$. Lastly we use the distance-induced kernel $k^{dist}_{\rho_1, 0}$ of (8), using the Euclidean distance ρ_1 so that the MMD is the energy distance. We also considered Cramér GANs, with the surrogate critic (10), and WGAN-GPs.

We also found it helpful to add an activation penalty to the critic representation network in certain MMD models. Otherwise the representations h_θ sometimes chose very large values, which for most kernels does not change the theoretical loss (defined only in terms of distances) but leads to floating-point precision issues. We use a combined L^2 penalty on activations across all critic layers, with a factor of 1 for RQ-dot kernel and 0.0001 for distance kernel.

Each model was trained with a batch size of 128, and 5 discriminator updates per generator update. For CIFAR-10 and LSUN we trained for $150\,000$ generator updates, while for MNIST we used $50\,000$. The initial learning rate was set to 10^{-4} and followed the adaptive scheme described in Section 4.1, with KID compared between the current model and the model $20\,000$ generator steps earlier ($5\,000$ for MNIST), every $2\,000$ steps (500 for MNIST). After 3 consecutive failures to improve, the learning rate was halved. This approach allowed us not to pick a different learning rate for each of the considered models, which would otherwise have been needed.

We scale the gradient penalty by 1, instead of the 10 recommended by Gulrajani et al. (2017) and Bellemare et al. (2017); we found this to usually work slightly better with MMD models. With the distance kernel, however, we scale the penalty by 10 to allow direct comparison with Cramér GAN.

MNIST After training for $50\,000$ generator iterations, all variants achieved reasonable results. Among MMD models, only the distance kernel saw an improvement with more neurons in the top layer. Table 1 shows the quantitative measures, computed on the basis of a LeNet model. All have

[9] `yann.lecun.com/exdb/mnist/`

[10] In the DCGAN architecture the number of filers doubles in each consecutive layer, so an f-filter critic has f, $2f$, $4f$ and $8f$ convolutional filters in layers 1-4, respectively.

achieved KIDs of essentially zero, and FIDs around the same as that of the test set, with Inception scores slightly lower. Figure 5, in Appendix D, shows samples.

Table 1: Mean (standard deviation) of score evaluations for the MNIST models.

	critic size				
loss	filters	top layer	Inception	FID	KID
rq	16	16	9.11 (0.01)	4.206 (0.05)	0.005 (0.004)
rbf	16	16	8.98 (0.02)	8.264 (0.02)	0.011 (0.006)
dot	16	16	8.86 (0.02)	6.245 (0.06)	0.006 (0.004)
dist	16	256	9.13 (.004)	6.179 (0.05)	0.005 (0.004)
Cramér GAN	16	256	9.25 (0.02)	3.385 (0.10)	0.006 (0.005)
WGAN-GP	16	1	9.12 (0.02)	6.915 (0.10)	0.009 (0.004)
test set	–	–	9.78 (0.02)	4.305 (0.16)	0.003 (0.003)

Figure 2, however, shows the evolution of our quantitative criteria throughout the training process for several models. This shows that the linear kernel *dot* and rbf kernel *rbf* are clearly worse than the other models at the beginning of the training process, but both make an improvement once learning rates are adapted to their needs. *rbf* however does not catch up fully with the other models. Examining samples during training, we observed that *rbf* more frequently produces extremely "blurry" outputs, which can persist for a substantial amount of time before eventually resolving. This makes sense, given the very fast gradient decay of the *rbf* kernel: when generator samples are extremely far away from the reference samples, slight improvements yield very little reward for the generator, and so bad samples can stay bad for a long time.

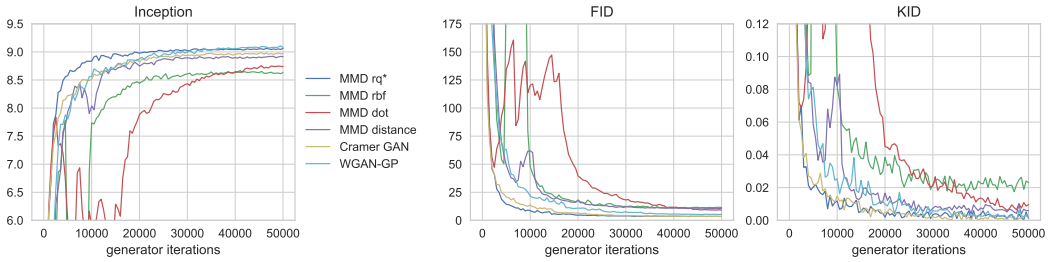


Figure 2: Score estimates over the learning process for MNIST training.

There is also some evidence that *dist*, and perhaps WGAN-GP, converge more slowly than *rq* and Cramér GAN. Given their otherwise similar properties, we thus recommend the use of *rq* kernels over *rbf* in MMD GANs.

CIFAR-10 Scores for various models trained on CIFAR-10 are shown in Table 2. The scores for *rq* with a small critic network approximately match those of WGAN-GP with a large critic network, at substantially reduced computational cost. With a small critic, WGAN-GP, Cramér GAN and the distance kernel all performed very poorly. Samples from these models are presented in Appendix D.

LSUN Bedrooms Table 3 presents scores for models trained on the LSUN Bedrooms dataset; samples from most of these models are shown in Figure 3. MMD GANs outperform Cramér and WGAN-GP for each critic size. Although results with smaller critic are worse than with large one for each considered model, MMDs still produce reasonably good samples, which certainly is not the case for WGAN-GP. Although Cramér GAN produces relatively good samples, the separate objects in these pictures often seem melted together and they are not as sharp as in MMD *rq** samples. With large critic, both Cramér GAN and MMD *rq** give good quality samples, many of which are hardly distinguishable from the test set by eye.

These results illustrate the benefits of using the MMD on deep convolutional feaures as a GAN critic. In this hybrid system, the initial convolutional layers map the generator and reference image distributions to a simpler representation, which is well suited to comparison via the MMD. The

Table 2: Mean (standard deviation) of score evaluations for the CIFAR-10 models.

loss	critic size filters	critic size top layer	Inception	FID	KID
rq	16	16	5.86 (0.06)	48.10 (0.16)	0.032 (0.001)
rq	64	16	6.51 (0.03)	39.90 (0.29)	0.027 (0.001)
dist	16	256	4.53 (0.03)	80.48 (0.19)	0.061 (0.001)
dist	64	256	6.39 (0.04)	40.25 (0.19)	0.028 (0.001)
Cramér GAN	16	256	4.67 (0.02)	74.93 (0.32)	0.060 (0.001)
Cramér GAN	64	256	6.39 (0.01)	40.27 (0.15)	0.028 (0.001)
WGAN-GP	16	1	3.15 (0.01)	147.09 (0.31)	0.116 (0.002)
WGAN-GP	64	1	6.53 (0.02)	37.52 (0.19)	0.026 (0.001)
test set	–	–	11.21 (0.13)	6.11 (0.05)	0.000 (0.000)

Table 3: Mean (standard deviation) of score evaluations for the LSUN models.

loss	critic size filters	critic size top layer	Inception	FID	KID
rq	16	16	3.12 (0.02)	279.88 (0.21)	0.297 (0.002)
rq	64	16	2.80 (0.02)	229.77 (0.26)	0.205 (0.002)
dist	16	256	3.41 (0.01)	299.69 (0.35)	0.316 (0.002)
dist	64	256	2.79 (0.01)	235.08 (0.24)	0.212 (0.002)
Cramér GAN	16	256	3.46 (0.02)	316.21 (0.15)	0.349 (0.002)
Cramér GAN	64	256	3.43 (0.02)	260.92 (0.50)	0.231 (0.002)
WGAN-GP	16	1	2.39 (0.01)	420.36 (0.28)	0.586 (0.003)
WGAN-GP	64	1	3.12 (0.01)	246.47 (0.40)	0.220 (0.002)

MMD in turn employs an infinite dimensional feature space to compare the outputs of these convolutional layers. By comparison, WGAN-GP requires a larger discriminator network to achieve similar performance. It is interesting to consider the question of kernel choice: the distance kernel and RQ kernel are both characteristic (Sriperumbudur et al., 2010), and neither suffers from the fast decay of the exponentiated quadratic kernel, yet the RQ kernel performs slightly better in our experiments. The relative merits of different kernel families for GAN training will be an interesting topic for further study.

REFERENCES

M. Arjovsky and L. Bottou. Towards principled methods for training generative adversarial networks. In *ICLR*, 2017. arXiv:1701.04862.

M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein generative adversarial networks. In *ICML*, 2017. arXiv:1701.07875.

S. Arora and Y. Zhang. Do GANs actually learn the distribution? an empirical study, 2017. arXiv:1706.08224.

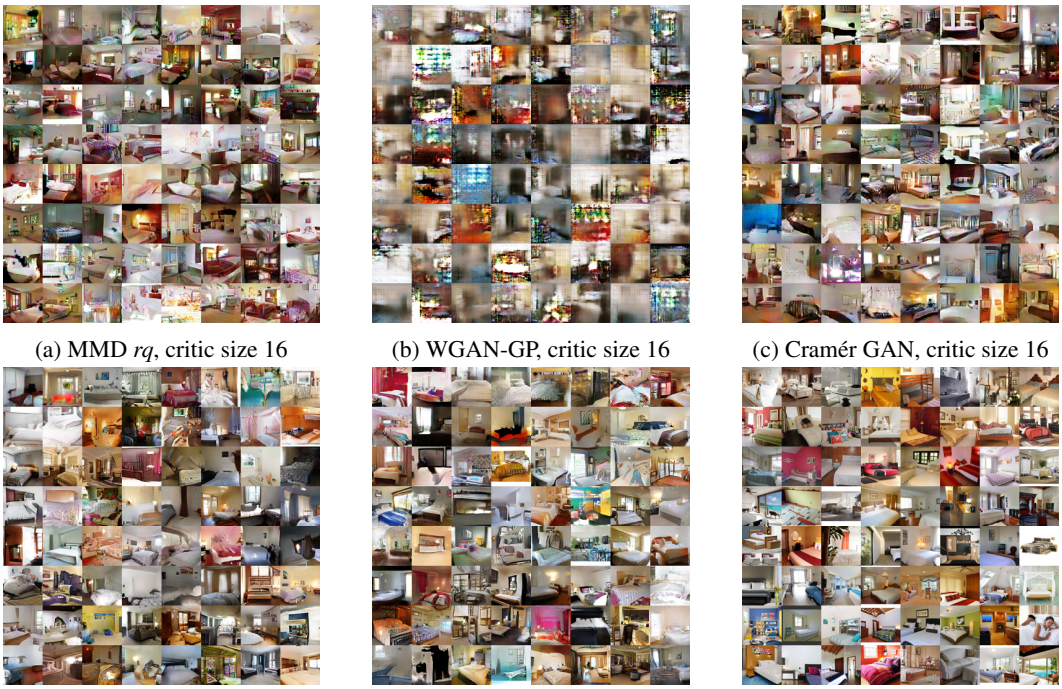
S. Arora, R. Ge, Y. Liang, T. Ma, and Y. Zhang. Generalization and equilibrium in generative adversarial nets (GANs). In *ICML*, 2017. arXiv:1703.00573.

M. G. Bellemare, I. Danihelka, W. Dabney, S. Mohamed, B. Lakshminarayanan, S. Hoyer, and R. Munos. The Cramer distance as a solution to biased Wasserstein gradients, 2017. arXiv:1705.10743.

Y. Bengio, G. Mesnil, Y. Dauphin, and S. Rifai. Better mixing via deep representations. In *ICML*, 2013. arXiv:1207.4404.

D. Berthelot, T. Schumm, and L. Metz. Began: Boundary equilibrium generative adversarial networks, 2017. arXiv:1703.10717.

W. Bounliphone, E. Belilovsky, M. B. Blaschko, I. Antonoglou, and A. Gretton. A test of relative similarity for model selection in generative models. In *ICLR*, 2016. arXiv:1511.04581.



(a) MMD *rq*, critic size 16 (b) WGAN-GP, critic size 16 (c) Cramér GAN, critic size 16

(d) MMD *rq*, critic size 64 (e) WGAN-GP, critic size 64 (f) Test set

Figure 3: Comparison of samples from various models and from the test set for the LSUN Bedroom database.

D.-A. Clevert, T. Unterthiner, and S. Hochreiter. Fast and accurate deep network learning by exponential linear units (ELUs). In *ICLR*, 2016. arXiv:1511.07289.

I. Danihelka, B. Lakshminarayanan, B. Uria, D. Wierstra, and P. Dayan. Comparison of maximum likelihood and GAN-based training of Real NVPs, 2017. arXiv:1705.05263.

G. K. Dziugaite, D. M. Roy, and Z. Ghahramani. Training generative neural networks via maximum mean discrepancy optimization. In *UAI*, 2015. arXiv:1505.03906.

W. Fedus, M. Rosca, B. Lakshminarayanan, A. M. Dai, S. Mohamed, and I. Goodfellow. Many paths to equilibrium: GANs do not need to decrease a divergence at every step, 2017. arXiv:1710.08446.

A. Genevay, G. Peyré, and M. Cuturi. Learning generative models with Sinkhorn divergences, 2017. arXiv:1706.00292.

T. Gneiting and A. E. Raftery. Strictly proper scoring rules, prediction, and estimation. *JASA*, 102 (477):359–378, 2007.

I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *NIPS*, 2014. arXiv:1406.2661.

A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. J. Smola. A kernel two-sample test. *JMLR*, 13, 2012.

I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville. Improved training of Wasserstein GANs. In *NIPS*, 2017. arXiv:1704.00028.

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. In *NIPS*, 2017. arXiv:1706.08500.

R. Huang, S. Zhang, T. Li, and R. He. Beyond face rotation: Global and local perception GAN for photorealistic and identity preserving frontal view synthesis, 2017a. arXiv:1704.04086.

X. Huang, Y. Li, O. Poursaeed, J. Hopcroft, and S. Belongie. Stacked generative adversarial networks. In *CVPR*, 2017b. arXiv:1612.04357.

Y. Jin, K. Zhang, M. Li, Y. Tian, H. Zhu, and Z. Fang. Towards the automatic anime characters creation with generative adversarial networks, 2017. arXiv:1708.05509.

Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. arXiv:1412.6980.

A. Klenke. *Probability Theory: A Comprehensive Course*. World Publishing Corporation, 2008.

A. Krizhevsky. Learning multiple layers of features from tiny images, 2009.

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998.

C. Li, D. Alvarez-Melis, K. Xu, S. Jegelka, and S. Sra. Distributional adversarial networks, 2017a. arXiv:1706.09549.

C.-L. Li, W.-C. Chang, Y. Cheng, Y. Yang, and B. Póczos. MMD GAN: Towards deeper understanding of moment matching network. In *NIPS*, 2017b. arXiv:1705.08584.

Y. Li, K. Swersky, and R. Zemel. Generative moment matching networks. In *ICML*, 2015. arXiv:1502.02761.

Lydia Tingruo Liu. On the two-sample statistic approach to generative adversarial networks. Master's thesis, University of Princeton Senior Thesis, April 2017. URL `http://arks.princeton.edu/ark:/88435/dsp0179408079v`.

S. Liu, O. Bousquet, and K. Chaudhuri. Approximation and convergence properties of generative adversarial learning. In *NIPS*, 2017. arXiv:1705.08991.

D. Lopez-Paz and M. Oquab. Revisiting classifier two-sample tests. In *ICLR*, 2017. arXiv:1610.06545.

R. Lyons. Distance covariance in metric spaces. *The Annals of Probability*, 41(5):3051–3696, 2013.

Boris Mityagin. The zero set of a real analytic function, 2015. arXiv:1512.07276.

Y. Mroueh and T. Sercu. Fisher GAN. In *NIPS*, 2017. arXiv:1705.09675.

Y. Mroueh, T. Sercu, and V. Goel. McGan: Mean and covariance feature matching GAN. In *ICML*, 2017. arXiv:1702.08398.

A. Müller. Integral probability metrics and their generating classes of functions. *Advances in Applied Probability*, 29(2):429–443, 1997.

S. Nowozin, B. Cseke, and R. Tomioka. f-GAN: Training generative neural samplers using variational divergence minimization. In *NIPS*, 2016. arXiv:1606.00709.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *JMLR*, 12:2825–2830, 2011.

A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *ICLR*, 2016. arXiv:1511.06434.

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA, 2006.

T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved techniques for training GANs. In *NIPS*, 2016. arXiv:1606.03498.

D. Sejdinovic, B. K. Sriperumbudur, A. Gretton, and K. Fukumizu. Equivalence of distance-based and RKHS-based statistics in hypothesis testing. *The Annals of Stastistics*, 41(5):2263–2291, 2013. arXiv:1207.6076.

B. K. Sriperumbudur, K. Fukumizu, A. Gretton, G. R. G. Lanckriet, and B. Schölkopf. Kernel choice and classifiability for RKHS embeddings of probability distributions. In *NIPS*, 2009a.

B. K. Sriperumbudur, K. Fukumizu, A. Gretton, B. Schölkopf, and G. R. G. Lanckriet. On integral probability metrics, phi-divergences and binary classification, 2009b. arXiv:0901.2698.

B. K. Sriperumbudur, A. Gretton, K. Fukumizu, G. R. G. Lanckriet, and B. Schölkopf. Hilbert space embeddings and metrics on probability measures. *JMLR*, 11:1517–1561, 2010. arXiv:0907.5309.

B. K. Sriperumbudur, K. Fukumizu, and G. R. G. Lanckriet. Universality, characteristic kernels and RKHS embedding of measures. *JMLR*, 12:2389–2410, 2011. arXiv:1003.0887.

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

I. Steinwart and A. Christmann. *Support Vector Machines*. Information Science and Statistics. Springer, 2008.

D. J. Sutherland, H.-Y. Tung, H. Strathmann, S. De, A. Ramdas, A. Smola, and A. Gretton. Generative models and model criticism via optimized maximum mean discrepancy. In *International Conference on Learning Representations*, 2017. arXiv:1611.04488.

C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. In *ICLR*, 2014. arXiv:1312.6199.

C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the Inception architecture for computer vision. In *CVPR*, 2016. arXiv:1512.00567.

G. Székely and M. Rizzo. Testing for equal distributions in high dimension. *InterStat*, 5, 2004.

L. Theis, A. van den Oord, and M. Bethge. A note on the evaluation of generative models. In *ICLR*, 2016. arXiv:1511.01844.

F. Yu, Y. Zhang, S. Song, A. Seff, and J. Xiao. LSUN: Construction of a large-scale image dataset using deep learning with humans in the loop, 2015. arXiv:1506.03365.

J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017. arXiv:1703.10593.

A SCORE FUNCTIONS, DIVERGENCES, AND THE CRAMÉR GAN

It is not immediately obvious how to interpret the surrogate loss (10). An insight comes from considering the *score function* associated with the energy distance, which we now briefly review (Gneiting & Raftery, 2007). A scoring rule is a function $S(\mathbb{P}, y)$, which is the loss incurred when a forecaster makes prediction $\mathbb{P}$, and the event y is observed. The *expected score* is the expectation under $\mathbb{Q}$ of the score,

$$S(\mathbb{P}, \mathbb{Q}) := \mathbb{E}_{Y \sim \mathbb{Q}} S(\mathbb{P}, Y).$$

If a score is *proper*, then the expected score obtained when $\mathbb{P} = \mathbb{Q}$ is greater or equal than the expected score for $\mathbb{P} \neq \mathbb{Q}$,

$$S(\mathbb{Q}, \mathbb{Q}) \geq S(\mathbb{P}, \mathbb{Q}).$$

A *strictly proper* scoring rule shows an equality only when $\mathbb{P}$ and $\mathbb{Q}$ agree. We can define a divergence measure based on this score,

$$\mathcal{D}_S(\mathbb{P}, \mathbb{Q}) = S(\mathbb{Q}, \mathbb{Q}) - S(\mathbb{P}, \mathbb{Q}). \tag{15}$$

Bearing in mind the definition of the divergence (15), it is easy to see (Gneiting & Raftery, 2007, eq. 22) that the energy distance (7) arises from the score function

$$S(\mathbb{P}, y) = \frac{1}{2}\,\mathbb{E}_{\mathbb{P}}\,\rho(X, X') - \mathbb{E}_{\mathbb{P}}\,\rho(X, y).$$

The interpretation is straightforward: the score of a reference sample y is determined by comparing its average distance to a generator sample with the average distance among independent generator samples, $\mathbb{E}_{\mathbb{P}}\,\rho(X, X')$.

As discussed earlier, the Cramér GAN critic does not use the energy distance (7) directly on the samples, but first maps the samples through a function h, for instance a convolutional network; this should be chosen to maximize the discriminative performance of the critic. Writing this mapping as h, we break the energy distance down as $\mathcal{D}_e(\mathbb{P}, \mathbb{Q}) = S(\mathbb{Q}, \mathbb{Q}) - S(\mathbb{P}, \mathbb{Q})$, where

$$S(\mathbb{Q}, \mathbb{Q}) = -\frac{1}{2}\,\mathbb{E}_{\mathbb{Q}}\,\rho(h(Y), h(Y')) \tag{16}$$

and

$$S(\mathbb{P}, \mathbb{Q}) = \frac{1}{2}\,\mathbb{E}_{\mathbb{P}}\,\rho(h(X), h(X')) - \mathbb{E}_{\mathbb{P},\mathbb{Q}}\,\rho(h(X), h(Y)). \tag{17}$$

When training the discriminator, the goal is to maximize the divergence by learning h, and so both (16) and (17) change: in other words, divergence maximization is not possible without two independent samples Y, Y' from the reference distribution $\mathbb{Q}$.

An alternative objective in light of the score interpretation, however, is to simply optimize the average score (17). In other words, we would find features h that make the average distance from generator to reference samples much larger than the average distance between pairs of generator samples. We no longer control the term encoding the "variability" due to $\mathbb{Q}$, $\mathbb{E}_{\mathbb{Q}}\,\rho(h(Y), h(Y'))$, which might therefore explode: for instance, h might cause $h(Y)$ to disperse broadly, and far from the support of $\mathbb{P}$, assuming sufficient flexibility to keep $\mathbb{E}_{\mathbb{P}}\,\rho(h(X), h(X'))$ under control. We can mitigate this by controlling the expected norm $\mathbb{E}_{\mathbb{Q}}\,\rho(h(Y), 0)$, which has the advantage of only requiring a single sample to compute. For example, we could maximize

$$-\frac{1}{2}\,\mathbb{E}_{\mathbb{P}}\,\rho(h(X), h(X')) + \mathbb{E}_{\mathbb{P},\mathbb{Q}}\,\rho(h(X), h(Y)) - \mathbb{E}_{\mathbb{Q}}\,\rho(h(Y), 0).$$

This resembles the Cramér GAN critic (10), but the generator-to-generator distance is scaled differently, and there is an additional term: $\mathbb{E}_{\mathbb{P}}\,\rho(h(X), 0)$ is being maximized, which is more difficult to interpret. An argument has been made (in personal communication with Bellemare et al.) that this last term is required if the function f_c in (9) is to be a witness of an integral probability metric (1), although the asymmetry of this witness in $\mathbb{P}$ vs $\mathbb{Q}$ needs to be analyzed further.

B Bias of generalized IPM estimators

We will now show that estimators of IPM-like distances based on data splitting are biased. Appendix B.1 defines a slight generalization of IPMs, used to analyze MMD GANs in the same framework as WGANs, and a class of estimators that are a natural model for the estimator used in GAN models. Appendix B.2 shows that such estimators are invariably biased in nontrivial cases; Appendix B.3 then demonstrates that any estimator with non-constant bias yields a biased gradient estimator. Appendices B.4 and B.5 demonstrate specific examples of this bias for the Wasserstein and maximized-MMD distances.

B.1 Generalized IPMs and data-splitting estimators

We will first define a slight generalization of IPMs: we will use this added generality to help analyze MMD GANs in Appendix B.5.

Definition 1 (Generalized IPM)**.** *Let $\mathcal{X}$ be some domain, with $\mathcal{M}$ a class of probability measures on $\mathcal{X}$.*[11] *Let $\mathcal{F}$ be some parameter set, and $J : \mathcal{F} \times \mathcal{M} \times \mathcal{M} \to \mathbb{R}$ an objective functional. The* generalized IPM $\mathcal{D} : \mathcal{M} \times \mathcal{M} \to \mathbb{R}$ *is then given by*

$$\mathcal{D}(\mathbb{P}, \mathbb{Q}) = \sup_{f \in \mathcal{F}} J(f, \mathbb{P}, \mathbb{Q}). \tag{18}$$

[11] All results in this section could be trivially extended to support $\mathbb{P}$ and $\mathbb{Q}$ over different domains $\mathcal{X}$ and $\mathcal{Y}$, if desired.

For example, if $\mathcal{F}$ is a class of functions $f : \mathcal{X} \to \mathbb{R}$, and J is given by

$$J_{IPM}(f, \mathbb{P}, \mathbb{Q}) = \mathbb{E}_{X\sim\mathbb{P}}\, f(X) - \mathbb{E}_{Y\sim\mathbb{Q}}\, f(Y), \tag{19}$$

then we obtain integral probability metrics (1). Given samples $\mathbf{X} \sim \mathbb{P}^m$ and $\mathbf{Y} \sim \mathbb{Q}^n$, let $\hat{\mathbb{P}}$ denote the empirical distribution of $\mathbf{X}$ (an equal mixture of point masses at each $X_i \in \mathbf{X}$), and similarly $\hat{\mathbb{Q}}$ for $\mathbf{Y}$. Then we have a simple estimator of (19) which is unbiased for fixed f:

$$\hat{J}_{IPM}(f, \mathbf{X}, \mathbf{Y}) = J_{IPM}(f, \hat{\mathbb{P}}, \hat{\mathbb{Q}}) = \frac{1}{m}\sum_{i=1}^{m} f(X_i) - \frac{1}{n}\sum_{j=1}^{n} f(Y_j).$$

Definition 2 (Data-splitting estimator of a generalized IPM)**.** *Consider the distance* (18)*, with objective J and parameter class $\mathcal{F}$. Suppose we observe iid samples $\mathbf{X} \sim \mathbb{P}^m$, $\mathbf{Y} \sim \mathbb{Q}^n$, for any two distributions $\mathbb{P}$, $\mathbb{Q}$. A* data-splitting estimator *is a function $\hat{\mathcal{D}}_{\mathcal{F}}(\mathbf{X}, \mathbf{Y})$ which first randomly splits the sample $\mathbf{X}$ into $\mathbf{X}^{tr}$, $\mathbf{X}^{te}$ and $\mathbf{Y}$ into $\mathbf{Y}^{tr}$, $\mathbf{Y}^{te}$, somehow selects a critic function $\hat{f}_{\mathbf{X}^{tr},\mathbf{Y}^{tr}} \in \mathcal{F}$ independently of $\mathbf{X}^{te}$, $\mathbf{Y}^{te}$, and then returns a result of the form*

$$\hat{\mathcal{D}}(\mathbf{X}, \mathbf{Y}) = \hat{J}(\hat{f}_{\mathbf{X}^{tr},\mathbf{Y}^{tr}}, \mathbf{X}^{te}, \mathbf{Y}^{te}),$$

where $\hat{J}(f, \mathcal{X}, \mathcal{Y})$ is an estimator of $J(f, \mathbb{P}, \mathbb{Q})$.

These estimators are defined by three components: the choice of relative sizes of the train-test split, the selection procedure for $\hat{f}_{\mathbf{X}^{tr},\mathbf{Y}^{tr}}$, and the estimator $\hat{J}$. The most obvious selection procedure is

$$\hat{f}_{\mathbf{X}^{tr},\mathbf{Y}^{tr}} \in \operatorname*{argmax}_{f\in\mathcal{F}} \hat{J}(f, \mathbf{X}^{tr}, \mathbf{Y}^{tr}), \tag{20}$$

though of course one could use regularization or other techniques to select a different $f \in \mathcal{F}$, and in practice one will use an approximate optimizer. Lopez-Paz & Oquab (2017) used an estimator of exactly this form in a two-sample testing setting.

As we optimize a WGAN-type model, we compute the loss (or its gradients) on a minibatch, while the current parameters of the critic are based only on data seen in previous iterations. We can view the current minibatch as $\mathbf{X}^{te}, \mathbf{Y}^{te}$, all previously-seen data as $\mathbf{X}^{tr}, \mathbf{Y}^{tr}$, and the current critic function as $\hat{f}_{\mathbf{X}^{tr},\mathbf{Y}^{tr}}$. Thus, at least in the first pass over the training set, WGAN-type approaches exactly fit the data-splitting form of Definition 2.

B.2 Estimator bias

These data-splitting estimators are, in general, biased downwards:

Theorem 2. *Consider a data-splitting estimator (Definition 2) of the generalized IPM $\mathcal{D}$ (Definition 1) based on an unbiased estimator $\hat{J}$ of J: for any fixed $f \in \mathcal{F}$,*

$$\mathbb{E}_{\substack{\mathbf{X}\sim\mathbb{P}^m \\ \mathbf{Y}\sim\mathbb{Q}^n}} \left[\hat{J}(f, \mathbf{X}, \mathbf{Y})\right] = J(f, \mathbb{P}, \mathbb{Q}).$$

Then either the selection procedure is almost surely perfect,

$$\Pr\left(J\left(\hat{f}_{\mathbf{X}^{tr},\mathbf{Y}^{tr}}, \mathbb{P}, \mathbb{Q}\right) = \mathcal{D}(\mathbb{P}, \mathbb{Q})\right) = 1, \tag{21}$$

or else the estimator has a downward bias:

$$\mathbb{E}\,\hat{\mathcal{D}}\left(\mathbf{X}, \mathbf{Y}\right) < \mathcal{D}\left(\mathbb{P}, \mathbb{Q}\right). \tag{22}$$

Proof of Theorem 2. Since $\mathbf{X}^{tr}, \mathbf{Y}^{tr}$ are independent of $\mathbf{X}^{te}, \mathbf{Y}^{te}$,

$$\mathbb{E}\,\hat{\mathcal{D}}(\mathbf{X}, \mathbf{Y}) = \mathbb{E}_{\substack{\mathbf{X}^{tr},\mathbf{Y}^{tr} \\ \mathbf{X}^{te},\mathbf{Y}^{te}}} \left[\hat{J}(\hat{f}_{\mathbf{X}^{tr},\mathbf{Y}^{tr}}, \mathbf{X}^{te}, \mathbf{Y}^{te})\right] = \mathbb{E}_{\mathbf{X}^{tr},\mathbf{Y}^{tr}} \left[J(\hat{f}_{\mathbf{X}^{tr},\mathbf{Y}^{tr}}, \mathbb{P}, \mathbb{Q})\right].$$

Define the suboptimality of $\hat{f}_{\mathbf{X}^{tr},\mathbf{Y}^{tr}}$ as

$$\varepsilon := \mathcal{D}(\mathbb{P}, \mathbb{Q}) - J\left(\hat{f}_{\mathbf{X}^{tr},\mathbf{Y}^{tr}}, \mathbb{P}, \mathbb{Q}\right),$$

so that $\mathbb{E}\,\hat{\mathcal{D}}(\mathbf{X}, \mathbf{Y}) = \mathcal{D}(\mathbb{P}, \mathbb{Q}) - \mathbb{E}[\varepsilon]$. Note that $\varepsilon \geq 0$, since $\mathcal{D}(\mathbb{P}, \mathbb{Q}) = \sup_{f \in \mathcal{F}} J(f, \mathbb{P}, \mathbb{Q})$ and so for any $f \in \mathcal{F}$ we have

$$J(f, \mathbb{P}, \mathbb{Q}) \leq \mathcal{D}(\mathbb{P}, \mathbb{Q}).$$

Thus, either $\Pr(\varepsilon = 0) = 1$, in which case (21) holds, or else $\mathbb{E}[\varepsilon] > 0$, giving (22). □

Note that many estimators of IPMs do not actually perform this data splitting procedure, instead estimating $\mathcal{D}(\mathbb{P}, \mathbb{Q})$ with the distance between empirical distributions $\mathcal{D}(\hat{\mathbb{P}}, \hat{\mathbb{Q}})$. The standard biased estimator of the MMD (Gretton et al., 2012, Equation 5), the IPM estimators of Sriperumbudur et al. (2012), and the empirical Wasserstein estimator studied by Bellemare et al. (2017) are all of this form. Considering $\mathbb{P} = \mathbb{Q}$, so that $\mathcal{D}(\mathbb{P}, \mathbb{Q}) = 0$, but $\hat{\mathbb{P}} \neq \hat{\mathbb{Q}}$, so that $\mathcal{D}(\hat{\mathbb{P}}, \hat{\mathbb{Q}}) > 0$, makes it clear that these estimators at least sometimes have an upwards bias.

Theorem 2 also makes clear that as $\hat{f}_{\mathbf{X}^{tr}, \mathbf{Y}^{tr}}$ converges to its optimum, the bias of $\hat{\mathcal{D}}$ should vanish (as in Bellemare et al., 2017, Theorem 3). Moreover, in the GAN setting the minibatch size only directly determines $\mathbf{X}^{te}$, $\mathbf{Y}^{te}$, which do not contribute to this bias; bias is due rather to the training procedure and the number of samples seen through the training process. As long as $\hat{f}_{\mathbf{X}^{tr}, \mathbf{Y}^{tr}}$ is not optimal, however, the estimator will remain biased.

B.3 Gradient estimator bias

We will now show that biased estimators, except for estimators with a constant bias, must also have biased gradients.

Assume that, as in the GAN setting, $\mathbb{Q}$ is given by a generator network G_ψ with parameter ψ and inputs $Z \sim \mathbb{Z}$, so that $Y = G_\psi(Z) \sim \mathbb{Q}_\psi$. The generalized IPM of (18) is now a function of ψ, which we will denote as

$$\mathcal{D}(\psi) := \mathcal{D}(\mathbb{P}, \mathbb{Q}_\psi).$$

Consider an estimator $\hat{\mathcal{D}}(\psi)$ of $\mathcal{D}(\psi)$. Theorem 3 shows that when $\hat{\mathcal{D}}(\psi)$ and $\mathcal{D}(\psi)$ are differentiable, the gradient $\nabla_\psi \hat{\mathcal{D}}(\psi)$ is an unbiased estimator for $\nabla_\psi \mathcal{D}(\psi)$ only if the bias of $\hat{\mathcal{D}}(\psi)$ doesn't depend on ψ. This is exceedingly unlikely to happen for the biased estimator $\hat{\mathcal{D}}(W)$ defined in Theorem 2.

Theorem 3. *Let $\mathcal{D} : \Psi \to \mathbb{R}$ be a function on a parameter space $\Psi \subseteq \mathbb{R}^d$, with a random estimator $\hat{\mathcal{D}} : \Psi \to \mathbb{R}$ which is almost surely differentiable. Suppose that $\hat{\mathcal{D}}$ has unbiased gradients:*

$$\mathbb{E}[\nabla_\psi \hat{\mathcal{D}}(\psi)] = \nabla_\psi \mathcal{D}(\psi).$$

Then, for each connected component of Ψ,

$$\mathbb{E}\,\hat{\mathcal{D}}(\psi) = D(\psi) + \text{const},$$

where the constant can vary only across distinct connected components.

Proof. Let ψ_1 and ψ_2 be an arbitrary pair of parameter values in Ψ, connected by some smooth path $r : [0, 1] \to \Psi$ with $r(0) = \psi_1$, $r(1) = \psi_2$. For example, if Ψ is convex, then paths of the form $r(t) = t\psi_1 + (1 - t)\psi_2$ are sufficient. Using Fubini's theorem and standard results about path integrals, we have that

$$\begin{aligned}
\mathbb{E}[\hat{\mathcal{D}}(\psi_1) - \hat{\mathcal{D}}(\psi_2)] &= \mathbb{E}\left[\int_0^1 \left(\nabla \hat{\mathcal{D}}(r(t))\right) \cdot r'(t)\,\mathrm{d}t\right] \\
&= \int_0^1 \mathbb{E}\left[\nabla \hat{\mathcal{D}}(r(t))\right] \cdot r'(t)\,\mathrm{d}t \\
&= \int_0^1 \left(\nabla \mathcal{D}(r(t))\right) \cdot r'(t)\,\mathrm{d}t \\
&= \mathcal{D}(\psi_1) - \mathcal{D}(\psi_2).
\end{aligned}$$

This implies that $\mathbb{E}[\hat{\mathcal{D}}(\psi)] = D(\psi) + \text{const}$ for all ψ in the same connected component of Ψ. □

B.4 WGANs

Theorem 2 holds for the original WGANs, whose critic functions are exactly L-Lipschitz, considering $\mathcal{F}$ as the set of L-Lipschitz functions so that $\mathcal{D}_{\mathcal{F}}$ is L times the Wasserstein distance. It also holds for either WGANs or WGAN-GPs with $\mathcal{F}$ the actual set of functions attainable by the critic architecture, so that $\mathcal{D}$ is "neural network distance" of Arora et al. (2017) or the "adversarial divergence" of Liu et al. (2017).

It should be obvious that for nontrivial distributions $\mathbb{P}$ and $\mathbb{Q}$ and reasonable selection criteria for $\hat{f}_{\mathbf{X}^{tr},\mathbf{Y}^{tr}}$, (21) does not hold, and thus (22) does (so that the estimate is biased downwards).

Example For an explicit demonstration, consider the Wasserstein case, $\mathcal{F}$ the set of 1-Lipschitz functions, with $\mathbb{P} = \mathcal{N}(1,1)$ and $\mathbb{Q} = \mathcal{N}(0,1)$. Here $\mathcal{D}_{\mathcal{F}}(\mathbb{P},\mathbb{Q}) = 1$; the only critic functions $f \in \mathcal{F}$ which achieve this are $f(t) = t + C$ for $C \in \mathbb{R}$.

If we observe only one training pair $X^{tr} \sim \mathbb{P}$ and $Y^{tr} \sim \mathbb{Q}$, when $X^{tr} > Y^{tr}$, $f_1(t) = t$ is a maximizer of (20), leading to the expected estimate $J(f_1,\mathbb{P},\mathbb{Q}) = 1$. But with probability $\Phi\left(-1/\sqrt{2}\right) \approx 0.24$ it happens that $X^{tr} < Y^{tr}$. In such cases, (20) could give e.g. $f_{-1}(t) = -t$, giving the expected response $J(f_{-1},\mathbb{P},\mathbb{Q}) = -1$; the overall expected estimate of the estimator using this critic selection procedure is then $\mathbb{E}\,\hat{\mathcal{D}}_{\mathcal{F}}(\mathbf{X},\mathbf{Y}) = \left(1 - \Phi\left(-1/\sqrt{2}\right)\right) - \Phi\left(-1/\sqrt{2}\right) \approx 0.52$.

The only way to achieve $\mathbb{E}\,\hat{\mathcal{D}}_{\mathcal{F}}(\mathbf{X},\mathbf{Y}) = 1$ would be a "stubborn" selection procedure which chooses $f_1 + C$ no matter the given inputs. This would have the correct output $\mathbb{E}\,\hat{\mathcal{D}}_{\mathcal{F}}(\mathbf{X},\mathbf{Y}) = 1$ for this $(\mathbb{P},\mathbb{Q})$ pair. Applying this same procedure to $\mathbb{P} = \mathcal{N}(-1,1)$ and $\mathbb{Q} = \mathcal{N}(0,1)$, however, would then give $\mathbb{E}\,\hat{\mathcal{D}}_{\mathcal{F}}(\mathbf{X},\mathbf{Y}) = -1$, when it should also be 1.

B.5 Maximal MMD estimator

Recall the distance $\eta(\mathbb{P},\mathbb{Q}) = \sup_\theta MMD^2(h_\theta(\mathbb{P}), h_\theta(\mathbb{Q}))$ defined by (14). MMD GANs can be viewed as estimating η according to the scheme of Theorem 2, with $\mathcal{F}$ the set of possible parameters θ, $J(\theta,\mathbb{P},\mathbb{Q}) = MMD^2(h_\theta(\mathbb{P}), h_\theta(\mathbb{Q}))$, and $\hat{J}(\theta,\mathbf{X},\mathbf{Y}) = MMD_u^2(h_\theta(\mathbf{X}), h_\theta(\mathbf{Y}))$. Clearly our optimization scheme for θ does not almost surely yield perfect answers, and so again we have

$$\mathbb{E}\,\hat{\eta}(\mathbf{X},\mathbf{Y}) < \eta(\mathbb{P},\mathbb{Q}).$$

As $m^{tr}, n^{tr} \to \infty$, as for Wasserstein it should be the case that $\hat{\eta} \to \eta$. This is shown for certain kernels, along with the rate of convergence, by Sriperumbudur et al. (2009a, Section 4).

Example For a particular case, consider

$$\mathbb{P} = \mathcal{N}\left(\begin{bmatrix}1\\0\end{bmatrix}, I\right), \qquad \mathbb{Q} = \mathcal{N}\left(\begin{bmatrix}0\\0\end{bmatrix}, I\right),$$

with $h_\theta : \mathbb{R}^2 \to \mathbb{R}$ given by $h_\theta(x) = \theta^\mathsf{T} x$, $\|\theta\| = 1$, so that h_θ chooses a one-dimensional projection of the two-dimensional data. Then use the linear kernel k^{dot}, so that the MMD is simply the difference in means between projected features:

$$MMD^2(h_\theta(\mathbb{P}), h_\theta(\mathbb{Q})) = \|\mathbb{E}\,\theta^\mathsf{T} X - \mathbb{E}\,\theta^\mathsf{T} Y\|^2 = \theta_1^2,$$

and

$$\mathbb{E}\,\hat{\eta}(\mathbf{X},\mathbf{Y}) = \mathbb{E}_{\mathbf{X}^{tr},\mathbf{Y}^{tr}}\left[\left(\hat{\theta}_{\mathbf{X}^{tr},\mathbf{Y}^{tr}}\right)_1^2\right].$$

Clearly $\eta(\mathbb{P},\mathbb{Q}) = 1$, which is obtained by $\theta \in \{(-1,0),(1,0)\}$; any other valid θ will yield a strictly smaller value of $MMD^2(h_\theta(\mathbb{P}), h_\theta(\mathbb{Q}))$.

The MMD GAN estimator of η, if the optimum is achieved, uses

$$\begin{aligned}\hat{\theta}_{\mathbf{X}^{tr},\mathbf{Y}^{tr}} &= \operatorname*{argmax}_{\theta} MMD_u^2(\mathbf{X}^{tr},\mathbf{Y}^{tr}) \\ &= \operatorname*{argmax}_{\theta} \frac{1}{m^{tr}(m^{tr}-1)}\sum_{i\neq j}^{m^{tr}}(\theta^\mathsf{T} X_i^{tr})(X_j^{tr\,\mathsf{T}}\theta) + \frac{1}{n^{tr}(n^{tr}-1)}\sum_{i\neq j}^{n^{tr}}(\theta^\mathsf{T} Y_i^{tr})(Y_j^{tr\,\mathsf{T}}\theta) \\ &\qquad - \frac{2}{m^{tr}n^{tr}}\sum_{i=1}^{m^{tr}}\sum_{j=1}^{n^{tr}}(\theta^\mathsf{T} X_i^{tr})(Y_j^{tr\,\mathsf{T}}\theta) \\ &= \operatorname*{argmax}_{\theta} \theta^\mathsf{T} A_{\mathbf{X}^{tr},\mathbf{Y}^{tr}}\theta,\end{aligned}$$

where $A_{\mathbf{X}^{tr},\mathbf{Y}^{tr}} \in \mathbb{R}^{2\times 2}$. $\hat{\theta}_{\mathbf{X}^{tr},\mathbf{Y}^{tr}}$ is then the normalized first eigenvector of $A_{\mathbf{X}^{tr},\mathbf{Y}^{tr}}$. For Gaussian $\mathbf{X}^{tr}$, $\mathbf{Y}^{tr}$ with finite m^{tr}, n^{tr}, $\hat{\theta}_{\mathbf{X}^{tr},\mathbf{Y}^{tr}}$ is a continuous random variable and so does not almost surely lie in $\{(-1,0),(1,0)\}$; thus by Theorem 2, $\mathbb{E}\,\hat{\eta}(\mathbf{X},\mathbf{Y}) < \eta(\mathbb{P},\mathbb{Q}) = 1$. (A numerical simulation for the former gives a value around 0.6 when $m^{tr} = n^{tr} = 2$.)

C Proof of unbiased gradients

We now proceed to prove Theorem 1, and indeed its generalization Theorem 4.

Exchanging the gradient and the expectation can often be guaranteed using a standard result in measure theory (see Proposition 1), as a corollary of the Dominated Convergence theorem (Proposition 2). This result, however, requires the property Proposition 1.(ii): for almost all inputs X, the mapping is differentiable on the entirety of a neighborhood around θ. This order of quantifiers is important: it allows the use of the mean value theorem to control the average rate of change of the function, and the result then follows from Proposition 2.

For a neural network with the ReLU activation function, however, this assumption doesn't hold in general. For instance, if $\theta = (\theta_1,\theta_2) \in \mathbb{R}^2$ with $\theta_2 \neq 0$ and $X \in \mathbb{R}$, one can consider this very simple function: $h_\theta(X) = \max(0, \theta_1 + \theta_2 X)$. For any fixed value of θ, the function $h_\theta(X)$ is differentiable in θ for all X in $\mathbb{R}$ except for $X_\theta = -\theta_1/\theta_2$. However, if we consider a ball of possible θ values $B(\theta, r)$, the function is not differentiable on the set $\{-\theta_1'/\theta_2' \in \mathbb{R} \mid \theta' \in B(\theta, r)\}$, which can have positive measure for many possible distributions for X.

In Theorem 4, we provide a proof of Theorem 1 for all parameter values outside of a "bad set" $\Theta_{\mathbb{P}}$, without relying on Proposition 1.(ii). This can be done using Lemma 1, which takes advantage of the particular structure of neural networks to control the average rate of change without using the mean value theorem. Dominated convergence (Proposition 2) can then be applied directly.

We also show in Proposition 3 that the set $\Theta_{\mathbb{P}}$, of parameter values where Theorem 4 might not hold, has zero Lebesgue measure. This relies on the standard Fubini theorem (Klenke, 2008, Theorem 14.16) and Lemma 3, which ensures that the network $\theta \mapsto h_\theta(X)$ is differentiable for almost all parameter values θ when X is fixed. Although Lemma 3 might at first sight seem obvious, it requires some technical considerations in topology and differential geometry.

Proposition 1 (Differentiation Lemma (e.g. Klenke, 2008, Theorem 6.28))**.** *Let V be a nontrivial open set in $\mathbb{R}^m$ and let $\mathbb{P}$ be a probability distribution on $\mathbb{R}^d$. Define a map $h : \mathbb{R}^d \times V \mapsto \mathbb{R}^n$ with the following properties:*

(i) For any $\theta \in V$, $\mathbb{E}_{\mathbb{P}}[\|h_\theta(X)\|] < \infty$.

(ii) For $\mathbb{P}$-almost all $X \in \mathbb{R}^d$, the map $V \to \mathbb{R}^n$, $\theta \mapsto h_\theta(X)$ is differentiable.

(iii) There exists a $\mathbb{P}$-integrable function $g : \mathbb{R}^d \mapsto \mathbb{R}$ such that $\|\partial_\theta h_\theta(X)\| \leq g(X)$ for all $\theta \in V$.

Then, for any $\theta \in V$, $\mathbb{E}_{\mathbb{P}}[\|\partial_\theta h_\theta(X)\|] < \infty$ and the function $\theta \mapsto \mathbb{E}_{\mathbb{P}}[h_\theta(X)]$ is differentiable with differential:

$$\partial_\theta \mapsto \mathbb{E}_{\mathbb{P}}[h_\theta(X)] = \mathbb{E}_{\mathbb{P}}[\partial_\theta h_\theta(X)].$$

Proposition 2 (Dominated Convergence Theorem (e.g. Klenke, 2008, Corollary 6.26))**.** *Let $\mathbb{P}$ be a probability distribution on $\mathbb{R}^d$ and f a measurable function. Let $(f_n)_{n\in\mathbb{N}}$ be a sequence of of integrable functions such that for $\mathbb{P}$-almost all $X \in \mathbb{R}^d$, $f_n(X) \to f(X)$ as n goes to ∞. Assume that there is a dominating function g: $\|f_n(X)\| \leq g(X)$ for $\mathbb{P}$-almost all $X \in \mathbb{R}^d$ for all $n \in \mathbb{N}$, and $\mathbb{E}_{\mathbb{P}}[g(X)] < \infty$. Then f is $\mathbb{P}$-integrable, and $\mathbb{E}_{\mathbb{P}}[f_n(X)] \to \mathbb{E}_{\mathbb{P}}[f(X)]$ as n goes to ∞.*

C.1 Network definition

We would like to consider general feed-forward networks with a directed acyclic computation graph G. Here, G consists of $L+1$ nodes, with a root node $i = 0$ and a leaf node $i = L$. We denote by $\pi(i)$ the set of parent nodes of i. The nodes are sorted according to a topological order: if j is a parent node of i, then $j < i$. Each node i for $i > 0$ computes a function f_i, which outputs a vector in $\mathbb{R}^{d_i}$ based on its input in $\mathbb{R}^{d_{\pi(i)}}$, the concatenation of the outputs of each layer in $\pi(i)$. Here $d_{\pi(i)} = \sum_{j\in\pi(i)} d_j$, and $d_0 = d > 0$. We define the feed-forward network that factorizes according the graph G and with functions f_i recursively:

$$\begin{aligned} h^0 &= X \\ h^i &= f_i(h^{\pi(i)}) \qquad \forall\, 0 < i \leq L \end{aligned}$$

$h^{\pi(i)}$ is the concatenation of the vectors h^j for $j \in \pi(i)$. The functions f_i can be of two types:

- *Affine transform (Linear Module)*:

$$f_i(Y) = g_i(W^i) \begin{bmatrix} Y \\ 1 \end{bmatrix}$$

  W^i is a vector of dimension m_i. The function $g_i : \mathbb{R}^{m_i} \to \mathbb{R}^{d_i \times (d_{\pi(i)}+1)}$ is a known linear operator on the weights W^i, which can account for convolutions and similar linear operations. We will sometimes use $\widetilde{Y}$ to denote the augmented vector $\begin{bmatrix} Y \\ 1 \end{bmatrix}$, which accounts for bias terms.

- *Non-linear*: These f_i have no learnable weights. f_i can potentially be non-differentiable, such as max pooling, ReLU, and so on. Some conditions on f_i will be required (see Assumption **D**); the usual functions used in practice satisfy these conditions.

Denote by C the set of nodes i such that f_i is non-linear. θ is the concatenation of parameters of all linear modules: $\theta = (W^k)_{k\in C^c}$, where C^c is the complement of C in $[L] = \{1, ..., L\}$. Call the total number of parameters $m = \sum_{i\in C^c} m_i$, so that $\theta \in \mathbb{R}^m$. The feature vector of the network corresponds to the output of the last node L and will be denoted $h_\theta := h_\theta^L \in \mathbb{R}^{d_L}$. The subscript θ stands for the parameters of the network.

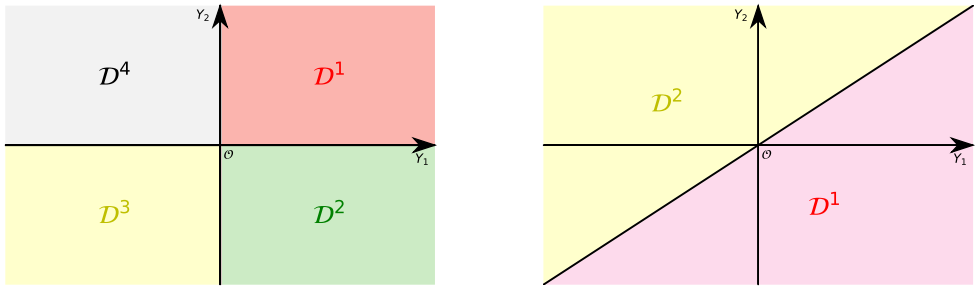
Domain of analyticity For a given node $i \in C$, we assume the domain of analiticity $\mathcal{D}_i$ for the function f_i to be the union of K_i disjoint sets:

$$\mathcal{D}_i = \bigcup_{k=1}^{K_i} \mathcal{D}_i^k.$$

To control the topology of these sets, we assume that each set $\mathcal{D}_i^k$ is defined by $S_{i,k}$ real analytic functions $G_{i,k,s} : \mathbb{R}^{d_\pi} \to \mathbb{R}$:

$$\mathcal{D}_i^k := \left\{ Y \in \mathbb{R}^{d_{\pi(i)}} \mid \forall s \in [S_{i,k}],\, G_{i,k,s}(Y) > 0 \right\}.$$

We assume that $\bigcup_{k=1}^{K_i} \bar{\mathcal{D}}_i^k = \mathbb{R}^{d_{\pi(i)}}$ and that the $\mathcal{D}_i^k$ are disjoint. We are interested in the case where f_i is real analytic on each one of the domains $\mathcal{D}_i^k$, but might be non-differentiable at their borders (see Assumption **D**). This includes most of the functions used for deep learning, including ReLU, Leaky ReLU, max-pooling, and so on.



(a) Domains of analyticity for the ReLU function in $\mathbb{R}^2$. (b) Domains of analyticity for max-pooling in $\mathbb{R}^2$.

Figure 4: Illustrations of the domains of analyticity for two functions used in deep learning. a represents the domains of analyticity for the ReLU function in $\mathbb{R}^2$. Each of the four domains $\mathcal{D}^1$, $\mathcal{D}^2$, $\mathcal{D}^3$ and $\mathcal{D}^4$ can be described by two inequalities of the form $G_{k,1}(Y) > 0$ and $G_{k,2}(Y) > 0$, where k is the index of each domain. Here, $G_{k,1}(Y) = \pm Y_1$ and $G_{k,2}(Y) = \pm Y_2$ which are analytic functions. Figure 4b represents the domains of analyticity for max-pooling function in $\mathbb{R}^2$. $\mathcal{D}^1$ and $\mathcal{D}^2$ are each one described by one inequality of the form $G_{k,1}(Y) > 0$ with $G_{1,1}(Y) = Y_1 - Y_2$ and $G_{2,1}(Y) = Y_2 - Y_1$ which are both analytic functions.

For example, if f_i computes the ReLU activation function on two inputs, then we have $K_i = 4$, with each $\mathcal{D}_i^k$ corresponding to a quadrant of the real plane (see Figure 4a). These quadrants might each be defined by $S_{i,k} = 2$ inequalities of the form $G_{i,k,1}(Y) > 0$ and $G_{i,k,2}(Y) > 0$, where $G_{i,k,s}(Y) = \pm Y_1$ and $G_{i,k,s}(Y) = \pm Y_2$ are analytic. Moreover, on each of these domains f_i coincides with an analytic function:

$$f_i(Y) = \begin{cases} (Y_1, Y_2); & Y \in \mathcal{D}_i^1 \\ (Y_1, 0); & Y \in \mathcal{D}_i^2 \\ (0, 0); & Y \in \mathcal{D}_i^3 \\ (0, Y_2); & Y \in \mathcal{D}_i^4 \end{cases} .$$

Another example is when f_i computes max-pooling on two inputs. In that case we have $K_i = 2$, and each domain $\mathcal{D}_i^k$ corresponds to a half plane (see Figure 4b). Each domain is defined by one inequality $G_{i,k,1}(Y) > 0$ with $G_{i,1,1}(Y) = Y_1 - Y_2$ and $G_{i,2,1}(Y) = Y_2 - Y_1$. Again, $G_{i,k,1}$ are analytic functions and f_i coincides with an analytic function on each of the domains:

$$f_i(Y) = \begin{cases} Y_1; & Y \in \mathcal{D}_i^1 \\ Y_2; & Y \in \mathcal{D}_i^2 \end{cases} .$$

When f_i is analytic on the whole space, $\mathcal{D}_i = \mathbb{R}^{d_{\pi(i)}}$, we can choose $K_i = 1$ to get $\mathcal{D}_i^1 = \mathbb{R}^{d_{\pi(i)}}$, which can be defined by a single function ($S_{i,k} = 1$) of $G_{i,1,1}(Y) = 1$. This case corresponds to most of the differentiable functions used in deep learning, such as the softmax, sigmoid, hyperbolic tangent, and batch normalization functions.

Index sets For a given node i, the following sets will be used extensively:

$$P := \left\{(i,k,s) \in \mathbb{N}^3 \mid i \in \{0\} \cup [L], k \in [K_i], s \in [S_{i,k}]\right\}$$

$$\neg i = \{(j,k,s) \in P \mid j \text{ is an ancestor of } i \text{ or } j = i\}$$

$$\neg\pi(i) = \bigcup_{j \in \pi(i)} \neg j = \{(j,k,s) \in P \mid j \text{ is an ancestor of } i\}$$

$$\partial i = \{(i,u,v) \in P\}$$

Note that $\partial i \subseteq \neg i$, and that $\neg i = \partial i \cup \neg \pi(i)$.

Furthermore, if $a(i)$ is the set of ancestors of node i, we define a backward trajectory starting from node i as an element q of the form:

$$q := (j, k_j)_{j \in a(i) \cup \{i\}}$$

where k_j are integers in $[K_j]$. We call $T(i)$ the set of such trajectories for node i.

Parameter sets When $X \in \mathbb{R}^d$ is fixed, to simplify notations we will often use h_θ^i to denote $h_\theta^i(X)$. For $p \in P$ of the form $p = (i, k, s)$, we introduce:

$$S^p = \{\theta \in \mathbb{R}^m \mid G_{i,k,s}(h_\theta^{\pi(i)}) = 0\}.$$

We also denote by ∂S^p the boundary of the set S^p. If Q is a subset of P, we use the following notation for convenience:

$$S^Q := \bigcup_{q \in Q} S^q, \qquad \partial S^Q := \bigcup_{q \in Q} \partial S^q.$$

For a given $\theta_0 \in \mathbb{R}^m$, we define the set of input vectors $X \in \mathbb{R}^d$ such that h_{θ_0} is not differentiable:

$$\mathcal{N}(\theta_0) = \Big\{X \in \mathbb{R}^d \mid \theta \mapsto h_\theta(X) \text{ is not differentiable at } \theta_0\Big\}.$$

Consider a random variable X in the input space $\mathbb{R}^d$, following the distribution $\mathbb{P}$.

For a given distribution $\mathbb{P}$, we introduce the following set of "critical" parameters:

$$\Theta_{\mathbb{P}} = \Big\{\theta \mid \mathbb{P}(\mathcal{N}(\theta)) > 0\Big\}.$$

This is the set of parameters θ where the network is not differentiable for a non-negligible set of datasets X.

Finally, for a given $X \in \mathbb{R}^d$, we define the set of parameters for which the network is not differentiable:

$$\Theta_X = \Big\{\theta_0 \in \mathbb{R}^m \mid \theta \mapsto h_\theta(X) \text{ is non-differentiable in } \theta_0\Big\}.$$

Top-level function Also define a function on the feature space $K : \mathbb{R}^{d_L} \to \mathbb{R}$.

This function might simply be $K(U) = U$, as in Corollaries 1 and 2. But it also allows us to represent the kernel function of an MMD GAN in Corollary 3: here we take X to be the two inputs to the kernel stacked together, apply the network to each of the two inputs with the same parameters in parallel, and then compute the kernel value between the two representations with K.

C.2 Assumptions

We will need the following assumptions at various points, where $\alpha \geq 1$:

- **A** (Moments) $\mathbb{E}_{\mathbb{P}}[\|X\|^\alpha] < \infty$.
- **B** The function K is continuously differentiable, and satisfies the following growth conditions where C_0 and C_1 are constants:

$$|K(U)| \leq C_0(\|U\|^\alpha + 1)$$
$$\|\nabla K(U)\| \leq C_1(\|U\|^{\alpha-1} + 1).$$

- **C** (Lipschitz nonlinear layers) For each $i \in C$, f_i is M-Lipschitz.
- **D** (Analytic pieces) For each $i \in C$, there are K_i functions $(f_i^k)_{k \in [K_i]}$ that are real analytic on $\mathbb{R}^{d_{\pi(i)}}$ which agree with f_i on the closure of their domains:

$$f_i(Y) = f_i^k(Y) \qquad \forall Y \in \bar{\mathcal{D}}_i^k.$$

Note that Assumption **B** is satisfied by the function $K(U) = U$, used in Corollaries 1 and 2, with $\alpha = 1$. It is also satisfied by the functions of MMD GAN with each of the kernels we consider in this work; see Corollary 3.

The class of networks we consider here, under Assumptions **C** and **D**, includes the vast majority of deep networks used in practice. For example, most variants of the ReLU activation function as well as max pooling are 1-Lipschitz, and can be decomposed into linear functions f_i^k on each $\bar{\mathcal{D}}_i^k$, as discussed previously. The softmax function as well as most differentiable activation functions, including the logistic sigmoid, $\tanh$, and the ELU (Clevert et al., 2016), also satisfy Assumptions **C** and **D**.

C.3 Main results

We first state the main result, which implies Theorem 1 via Corollaries 1 to 3. The proof depends on various intermediate results which will be established afterwards.

Theorem 4. *Under Assumptions **A** to **D**, for μ-almost all $\theta_0 \in \mathbb{R}^m$ the function $\theta \mapsto \mathbb{E}_{\mathbb{P}}[K(h_\theta(X))]$ is differentiable at θ_0, and*

$$\partial_\theta \, \mathbb{E}_{\mathbb{P}}[K(h_\theta(X))] = \mathbb{E}_{\mathbb{P}}[\partial_\theta \, K(h_\theta(X))],$$

where μ is the Lebesgue measure.

Proof. Let θ_0 be in $\Theta_{\mathbb{P}}^c$, i.e.: θ_0 is such that $h_\theta(X)$ is differentiable in θ_0 for $\mathbb{P}$-almost all X. Consider a sequence $(\theta_n)_{n\in\mathbb{N}}$ that converges to θ_0; there is then an $R > 0$ such that $\|\theta_n - \theta_0\| < R$ for all $n \in \mathbb{N}$. Letting X be in $\mathbb{R}^d$, Lemma 2 gives that

$$|K(h_{\theta_n}(X)) - K(h_{\theta_0}(X))| \leq F(X)\|\theta_n - \theta_0\|$$

with $\mathbb{E}_{\mathbb{P}}[F(X)] < \infty$. It also follows that:

$$|\partial_\theta k(h_{\theta_0}(X))| \leq F(X)$$

for $\mathbb{P}$-almost all $X \in \mathbb{R}^d$. The sequence $\mathcal{M}_n(X)$ defined by:

$$\mathcal{M}_n(X) = \frac{1}{\|\theta_n - \theta_0\|}|K(h_{\theta_n}(X)) - K(h_{\theta_0}(X)) - \partial_\theta \, K(h_{\theta_0}(X))(\theta_n - \theta_0)|$$

converges point-wise to 0 and is bounded by the integrable function $2F(X)$, therefore by the dominated convergence theorem (Proposition 2) it follows that

$$\mathbb{E}_{\mathbb{P}}[\mathcal{M}_n(X)] \to 0.$$

Using Lemma 1 and Assumption **B** to dominate the sequence $K(h_{\theta_n}(X))$, we get

$$\mathbb{E}_{\mathbb{P}}[K(h_{\theta_n}(X))] \to \mathbb{E}_{\mathbb{P},\mathbb{Q}}[K(h_{\theta_0}(X))].$$

Finally we define the sequence

$$\mathcal{R}_n = \frac{1}{\|\theta_n - \theta_0\|}\left|\mathbb{E}_{\mathbb{P},\mathbb{Q}}[K(h_{\theta_n}(X))] - \mathbb{E}_{\mathbb{P},\mathbb{Q}}[K(h_{\theta_0}(X))] - \mathbb{E}_{\mathbb{P},\mathbb{Q}}[\partial_\theta \, K(h_{\theta_n}(X))](\theta_0 - \theta_0)\right|,$$

which is upper-bounded by $\mathbb{E}_{\mathbb{P}}[\mathcal{M}_n(X)]$ and therefore converges to 0. We have shown that for $\theta_0 \in \Theta_{\mathbb{P}}^c$:

$$\partial_\theta \, \mathbb{E}_{\mathbb{P}}[K(h_\theta(X))] = \mathbb{E}_{\mathbb{P}}[\partial_\theta \, K(h_\theta(X))].$$

But by Proposition 3, $\mu(\Theta_{\mathbb{P}}) = 0$, and so we conclude that this holds for μ-almost all θ_0 in $\mathbb{R}^m$. □

These corollaries of Theorem 4 apply it to specific GAN architectures. Here we use the distribution $\mathbb{Z}$ to represent the noise distribution.

Corollary 1 (WGANs)**.** *Let $\mathbb{P}$ and $\mathbb{Z}$ be two distributions, on $\mathcal{X}$ and $\mathcal{Z}$ respectively, each satisfying Assumption **A** for $\alpha = 1$. Let $G_\psi : \mathcal{Z} \to \mathcal{X}$ be a generator network and $D_\theta : \mathcal{X} \to \mathbb{R}$ a critic network, each satisfying Assumptions **C** and **D**. Then, for μ-almost all (θ, ψ), we have that*

$$\mathbb{E}_{\substack{\mathbf{X}\sim\mathbb{P}^m \\ \mathbf{Z}\sim\mathbb{Z}^n}}\left[\partial_{\theta,\psi}\left[\frac{1}{m}\sum_{i=1}^m D_\theta(X_i) - \frac{1}{n}\sum_{j=1}^n D_\theta(G_\psi(Z_j))\right]\right] = \partial_{\theta,\psi}\left[\mathbb{E}_{X\sim\mathbb{P}} D_\theta(X) - \mathbb{E}_{Z\sim\mathbb{Z}} D_\theta(G_\psi(Z))\right].$$

Proof. By linearity, we only need the following two results:

$$\mathbb{E}_{X\sim\mathbb{P}}\,\partial_\theta D_\theta(X_i) = \partial_\theta\,\mathbb{E}_{X\sim\mathbb{P}}\,D_\theta(X) \quad \text{and} \quad \mathbb{E}_{Z\sim\mathbb{Z}}\,\partial_{\theta,\psi} D_\theta(G_\psi(Z_j)) = \partial_{\theta,\psi}\,\mathbb{E}_{Z\sim\mathbb{Z}}\,D_\theta(G_\psi(Z)).$$

The first follows immediately from Theorem 4, using the function $K(U) = U$ (which clearly satisfies Assumption **B** for $\alpha = 1$). The latter does as well by considering that the augmented network $h_{(\theta,\psi)}(Z) = D_\theta(G_\psi(Z))$ still satisifes the conditions of Theorem 4. □

Corollary 2 (Original GANs)**.** *Let $\mathbb{P}$ and $\mathbb{Z}$ be two distributions, on $\mathcal{X}$ and $\mathcal{Z}$ respectively, each satisfying Assumption **A** for $\alpha = 1$. Let $G_\psi : \mathcal{Z} \to \mathcal{X}$ be a generator network, and $D_\theta : \mathcal{X} \to \mathbb{R}$ a discriminator network, each satisfying Assumptions **C** and **D**. Further assume that the output of D is almost surely bounded: there is some $\gamma > 0$ such that for μ-almost all (θ, ψ),*

$$\Pr_{X\sim\mathbb{P}}(\gamma \le D_\theta(X) \le 1-\gamma) = 1 \qquad \textit{and} \qquad \Pr_{Z\sim\mathbb{Z}}(\gamma \le D_\theta(G_\psi(Z)) \le 1-\gamma) = 1.$$

Then we have the following:

$$\begin{aligned}
\mathbb{E}_{X\sim\mathbb{P}}\left[\partial_\theta\left[\log D_\theta(X)\right]\right] &= \partial_\theta\left[\mathbb{E}_{X\sim\mathbb{P}}\left[\log D_\theta(X)\right]\right] \\
\mathbb{E}_{Z\sim\mathbb{Z}}\left[\partial_{\theta,\psi}\left[\log\left(1 - D_\theta(G_\psi(Z))\right)\right]\right] &= \partial_{\theta,\psi}\left[\mathbb{E}_{Z\sim\mathbb{Z}}\left[\log\left(1 - D_\theta(G_\psi(Z))\right)\right]\right] \\
\mathbb{E}_{Z\sim\mathbb{Z}}\left[\partial_{\theta,\psi}\left[\log D_\theta(G_\psi(Z))\right]\right] &= \partial_{\theta,\psi}\left[\mathbb{E}_{Z\sim\mathbb{Z}}\left[\log D_\theta(G_\psi(Z))\right]\right].
\end{aligned}$$

Thus, by linearity, gradients of all the loss functions given in Goodfellow et al. (2014, Section 3) are unbiased.

Proof. The $\log$ function is real analytic and $(1/\gamma)$-Lipschitz on $(\gamma, 1-\gamma)$. The claim therefore follows from Theorem 4, using the networks $\log \circ D_\theta$, $\log \circ [x \mapsto (1-x)] \circ D_\theta \circ G_\psi$, and $\log \circ D_\theta \circ G_\psi$ with $K(U) = U$. □

The following assumption about a kernel k implies Assumption **B** when used as a top-level function K:

E Suppose k is a kernel such that there are constants C_0, C_1 where

$$\begin{aligned}
|k(U,V)| &\le C_0\left(\left(\|U\|^2 + \|V\|^2\right)^{\alpha/2} + 1\right) \\
\|\nabla_{U,V} k(U,V)\| &\le C_1\left(\left(\|U\|^2 + \|V\|^2\right)^{(\alpha-1)/2} + 1\right).
\end{aligned}$$

Corollary 3 (MMD GANs)**.** *Let $\mathbb{P}$ and $\mathbb{Z}$ be two distributions, on $\mathcal{X}$ and $\mathcal{Z}$ respectively, each satisfying Assumption **A** for some $\alpha \ge 1$. Let k be a kernel satisfying Assumption **E**. Let $G_\psi : \mathcal{Z} \to \mathcal{X}$ be a generator network and $D_\theta : \mathcal{X} \to \mathbb{R}$ a critic representation network each satisfying Assumptions **C** and **D**. Then*

$$\mathbb{E}_{\substack{\mathbf{X}\sim\mathbb{P}^m \\ \mathbf{Z}\sim\mathbb{Z}^n}}\left[\partial_{\theta,\psi}\, MMD_u^2(D_\theta(\mathbf{X}), D_\theta(G_\psi(\mathbf{Z})))\right] = \partial_{\theta,\psi}\, MMD^2(D_\theta(\mathbb{P}), D_\theta(G_\psi(\mathbb{Z}))).$$

Proof. Consider the following augmented networks:

$$\begin{aligned}
h^{(1)}_{(\theta,\psi)}(X, Z) &= (D_\theta(X), D_\theta(G_\psi(Z))) \\
h^{(2)}_{(\theta,\psi)}(Z, Z') &= (D_\theta(G_\psi(Z)), D_\theta(G_\psi(Z))) \\
h^{(3)}_{\theta}(X, X') &= (D_\theta(X), D_\theta(X)).
\end{aligned}$$

$h^{(1)}$ has inputs distributed as $\mathbb{P} \times \mathbb{Z}$, which satisfies Assumption **A** with the same α as $\mathbb{P}$ and $\mathbb{Z}$, and $h^{(1)}$ satisfies Assumptions **C** and **D**. The same is true of $h^{(2)}$ and $h^{(3)}$. Moreover, the function

$$K\left(\begin{bmatrix} U \\ V \end{bmatrix}\right) = k(U, V)$$

satisfies Assumption **B**. Thus Theorem 4 applies to each of $h^{(1)}$, $h^{(2)}$, and $h^{(3)}$. Considering the form of MMD_u^2 (4), the result follows by linearity and the fact that MMD_u^2 is unbiased (Gretton et al., 2012, Lemma 6). □

Each of the kernels considered in this paper satisfies Assumption **E** with α at most 2:

- $k^{dot}(x,y) = \langle x, y\rangle$ works with $\alpha = 2$, $C_0 = 1$, $C_1 = 1$.
- k^{rbf}_{σ} of (5) works with $\alpha = 2$, $C_0 = 1$, $C_1 = \sqrt{2}\sigma^{-2}$.
- $k^{rq}_{\alpha'}$ of (6) works with $\alpha = 2$, $C_0 = 1$, $C_1 = \sqrt{2}$.
- $k^{dist}_{\rho_\beta,0}$ of (8), using $\rho_\beta(x,y) = \|x-y\|^\beta$ with $1 \leq \beta \leq 2$, works with $\alpha = \beta$, $C_0 = 3$, $C_1 = 4\beta$.

Since the existence of a moment implies the existence of all lower-order moments by Jensen's inequality, this finalizes the proof of Theorem 1.

C.4 Bounds on network growth

The following lemmas were used in the proof of Theorem 4. We start by stating a result on the growth and Lipschitz properties of the network.

Lemma 1. *Under Assumption **C**, there exist continuous functions $\theta \mapsto b(\theta), a(\theta)$ and $(\theta,\theta') \mapsto (\alpha(\theta,\theta'), \beta(\theta,\theta'))$ such that:*

$$\|h_\theta(X)\| \leq b(\theta) + a(\theta)\|X\|$$
$$\|h_\theta(X) - h_{\theta'}(X)\| \leq \|\theta-\theta'\|\Big(\beta(\theta,\theta') + \alpha(\theta,\theta')\|X\|\Big)$$

for all X in $\mathbb{R}^d$ and all θ,θ' in $\mathbb{R}^m$.

Proof. Let X in $\mathbb{R}^d$ and θ,θ' in $\mathbb{R}^m$. We proceed by recursion on the nodes of the network. For $i=0$ the inequalities hold trivially $b_0 = 0$, $a_0 = 1$, $\beta_0 = 0$ and $\alpha_0 = 0$. Assume now that:

$$\|h^{\pi(i)}_\theta(X)\| \leq b_{\pi(i)}(\theta) + a_{\pi(i)}(\theta)\|X\|$$
$$\|h^{\pi(i)}_\theta(X) - h^{\pi(i)}_{\theta'}(X)\| \leq \|\theta-\theta'\|\big(\beta_{\pi(i)}(\theta,\theta') + \alpha_{\pi(i)}(\theta,\theta')\|X\|\big)$$

where $a_{\pi(i)}(\theta)$, $b_{\pi(i)}(\theta)$, $\alpha_{\pi(i)}(\theta,\theta')$ and $\beta_{\pi(i)}(\theta,\theta')$ are continuous functions. If i is a linear layer then:

$$\begin{aligned}\|h^i_\theta\| &\leq \|g_i(W^i)\widetilde{h}^{\pi(i)}_\theta)| \\ &\leq \|g_i\|\|W^i\|\Big(\|h^{\pi(i)}_\theta\| + 1\Big) \\ &\leq b_i(\theta) + a_i(\theta)\|X\|\end{aligned}$$

with $a_i(\theta) = \|g_i\|\|W^i\|a_{\pi(i)}(\theta)$ and $b_i(\theta) = \|g_i\|\|W^i\|b_{\pi(i)}(\theta)$. Moreover, we have that:

$$\begin{aligned}\|h^i_\theta(X) - h^i_{\theta'}(X)\| &= \|g_i(W_i)\widetilde{h}^{\pi(i)}_\theta(X) - g_i(W'_i)\widetilde{h}^{\pi(i)}_{\theta'}(X)\| \\ &\leq \|g_i(W_i)\big(\widetilde{h}^{\pi(i)}_{\theta'}(X) - \widetilde{h}^{\pi(i)}_\theta(X)\big)\| + \|\big(g_i(W_i - W'_i)\big)\widetilde{h}^{\pi(i)}_{\theta'}(X)\| \\ &\leq \|g_i\|\big(\|W_i\|\|h^{\pi(i)}_{\theta'}(X) - h^{\pi(i)}_\theta(X)\| + \|W_i - W'_i\|(\|h^{\pi(i)}_{\theta'}(X)\| + 1)\big) \\ &\leq \|\theta-\theta'\|\big(\beta_i(\theta,\theta') + \alpha_i(\theta,\theta')\|X\|\big)\end{aligned}$$

with:

$$\alpha_i(\theta,\theta') = \|g_i\|\big((\|W_i\| + \|W'_i\|)\alpha_{\pi(i)}(\theta,\theta') + (a_{\pi(i)}(\theta) + a_{\pi(i)}(\theta'))\big)$$
$$\beta_i(\theta,\theta') = \|g_i\|\big((\|W_i\| + \|W'_i\|)\beta_{\pi(i)}(\theta,\theta') + (b_{\pi(i)}(\theta) + b_{\pi(i)}(\theta')) + 1\big).$$

When i is not a linear layer, then by Assumption **C** f_i is $M-$Lipschitz, we get directly the functions by recursion: $\alpha_i = M\alpha_{\pi(i)}$, $\beta_i = M\beta_{\pi(i)}$, $a_i = Ma_{\pi(i)}$ and $b_i = Mb_{\pi(i)}$. □

Lemma 2. *Let R be a positive constant and $\theta \in \mathbb{R}^m$. Under Assumptions **A** to **C**, the following hold for all $\theta' \in B(\theta,R)$ and all X in $\mathbb{R}^d$:*

$$|\,K(h_{\theta'}(X)) - K(h_\theta(X))| \leq F(X)\|\theta-\theta'\|$$

with $\mathbb{E}_{\mathbb{P}}[F(X)] < \infty$.

Proof. We will first prove the following inequality:

$$|\,K(U) - K(V)| \leq C_1\big((\|U - V\| + \|V\|)^{\alpha-1}\big)\|U - V\|$$

for all U and V in $\mathbb{R}^L$.

Let t be in $[0, 1]$ and define the function f by

$$f(t) = K(tU + (1 - t)V)$$

Then $f(0) = K(V)$ and $f(1) = K(U)$. Moreover, f is differentiable and its derivative is given by:

$$f'(t) = \nabla\, K((tU + (1 - t)V))(U - V)$$

Using Assumption **B** one has that:

$$\begin{aligned}
|f'(t)| &= \big\|\nabla\, K((tU + (1 - t)V)(U - V)\big\| \\
&\leq \|\nabla\, K(tU + (1 - t)V)\|\|(U - V)\| \\
&\leq C_1\big(\|t(U - V) + V\|^{\alpha-1}\big)\|U - V\| \\
&\leq C_1\big((\|U - V\| + \|V\|)^{\alpha-1}\big)\|U - V\|.
\end{aligned}$$

The conclusion follows using the mean value theorem. Now choosing $U = h_{\theta'}(X)$ and $V = h_\theta(X)$ one gets the following:

$$|k(h_{\theta'}(X)) - k(h_\theta(X))| \leq C_1((\|h_{\theta'}(X) - h_\theta(X)\| + \|h_\theta(X)\|)^{\alpha-1})\|h_{\theta'}(X) - h_\theta(X)\|$$

Under Assumption **C**, it follows by Lemma 1 that:

$$\begin{aligned}
\|h_{\theta'}(X) - h_\theta(X)\| &\leq \Big(\beta(\theta, \theta') + \alpha(\theta, \theta')\|X\|\Big) \\
\|h_\theta(X)\| &\leq b(\theta) + a(\theta)\|X\|
\end{aligned}$$

The functions a, b, α, β defined in Lemma 1 are continuous, and hence all bounded on the ball $B(\theta, R)$; choose $D > 0$ to be a bound on all of these functions. It comes after some algebra:

$$|k(h_{\theta'}(X)) - k(h_\theta(X))| \leq C_1 D^\alpha (R + 1)^\alpha (1 + \|X\|)^\alpha \|\theta' - \theta\|$$

Set $F(X) = C_1 D^\alpha (R + 1)^\alpha (1 + \|X\|)^\alpha$. Since $\alpha \geq 1$, $t \mapsto (1 + t^{1/\alpha})^\alpha$ is concave on $t \geq 0$, and so we have that

$$\mathbb{E}\left[(1 + \|U\|)^\alpha\right] \leq \Big(1 + \mathbb{E}\left[\|U\|^\alpha\right]^{1/\alpha}\Big)^\alpha < \infty$$

via Jensen's inequality and Assumption **A**. Thus $F(X)$ is integrable.

□

C.5 Critical parameters have zero measure

The last result required for the proof of Theorem 4 is Proposition 3.

Proposition 3. *Under Assumption **D**, the set $\Theta_{\mathbb{P}}$ has 0 Lebesgue measure for any distribution $\mathbb{P}$.*

Proof. We will first consider the following two sets:

$$\begin{aligned}
D &= \Big\{(\theta, X) \in \mathbb{R}^m \times \mathbb{R}^d \mid \theta \in \Theta_{\mathbb{P}} \textit{ and } X \in \mathcal{N}(\theta)\Big\} \\
Q &= \Big\{(\theta, X) \in \mathbb{R}^m \times \mathbb{R}^d \mid \theta \in \Theta_X\Big\}.
\end{aligned}$$

Note that we have the inclusion $D \subseteq Q$. We endow the two sets with the product measure $\nu := \mu \times \mathbb{P}$, where μ is the Lebesgue measure. Therefore $\nu(D) \leq \nu(Q)$. On one hand by Fubini's theorem, we know that:

$$\begin{aligned}
\nu(Q) &= \int_{\mathbb{R}^d} \int_{\Theta_X} d\mu(\theta) d\mathbb{P}(X) \\
&= \int_{\mathbb{R}^d} \mu(\Theta_X) d\mathbb{P}(X).
\end{aligned}$$

By Lemma 3, we have that $\mu(\Theta_X) = 0$; therefore $\nu(Q) = 0$ and hence $\nu(D) = 0$. On the other hand, we use again Fubini's theorem for $\nu(D)$ to write:

$$\begin{aligned}\nu(D) &= \int_{\Theta_{\mathbb{P}}} \int_{\mathcal{N}(\theta)} d\mathbb{P}(X) d\mu(\theta) \\ &= \int_{\Theta_{\mathbb{P}}} \mathbb{P}(\mathcal{N}(\theta)) d\mu(\theta) = 0.\end{aligned}$$

If $\mu(\Theta_{\mathbb{P}}) > 0$, it would imply that $\mathbb{P}(\mathcal{N}(\theta)) = 0$ for almost all $\theta \in \Theta_{\mathbb{P}}$, which contradicts the definition of the set $\Theta_{\mathbb{P}}$. We conclude that $\mu(\Theta_{\mathbb{P}}) = 0$. □

Lemma 3. *Under Assumption **D**, for any X in $\mathbb{R}^d$, the set Θ_X has 0 Lebesgue measure: $\mu(\Theta_X) = 0$.*

Proof. We first need to prove that $\Theta_X \subseteq \partial S^P$. Let θ_0 be in Θ_X, then by Assumption **D**, it follows that $\theta_0 \in S^P$. Assume for the sake of contradiction that $\theta_0 \notin \partial S^P$. Applying Lemma 5 to the output node L, one directly shows that there exist $q \in T(L)$ and $\eta > 0$ such that $h_\theta = h_\theta^L = f^q(\theta)$ for all $\theta \in B(\theta_0, \eta)$, where f^q is a real analytic function. This would mean that the network is differentiable at θ_0, which contradicts the fact that $\theta \in \Theta_X$. One can then conclude that $\Theta_X \subseteq \partial S^P$, hence, by Lemma 4, it comes that $\mu(\Theta_X) = 0$. □

Lemma 4. *Under Assumption **D***

$$\mu(\partial S^P) = 0.$$

Proof. We will proceed by recursion. For $i = 0$ we trivially have $\partial S^{\neg 0} = \emptyset$, thus $\mu(\partial S^{\neg 0}) = 0$. Assume now that

$$\mu(\partial S^{\neg \pi(i)}) = 0.$$

For some $s = (p, q)$ with $p \in \partial i$ and $q \in T(i)$, define the set

$$\mathcal{M}_s = \{\theta \in \mathbb{R}^m \mid G_p(f^q(\theta)) = 0\}$$

where f^q is a real analytic function defined in Lemma 5 which locally agrees with $h_\theta^{\pi(i)}$.

We will prove now that for any θ in $\partial S^{\partial i} \setminus \partial S^{\neg \pi(i)}$ there exists $s \in \partial i \times T(i)$ such that $\theta \in \mathcal{M}_s$ and $\mu(\mathcal{M}_s) = 0$. We proceed by contradiction. Let $\theta \in \partial S^{\partial i} \setminus \partial S^{\neg \pi(i)}$, then there exists $\eta > 0$ such that $B(\theta, \eta) \subseteq \mathbb{R}^m \setminus \partial S^{\neg \pi(i)}$. By Lemma 5, there exists q in $T(i)$ such that

$$h_{\theta'}^{\pi(i)} = f^q(\theta') \qquad \forall \theta' \in B(\theta, \eta). \tag{23}$$

Moreover, since $\theta \in \partial S^{\partial i}$ there exists $p \in \partial i$ such that $G_p(h_\theta^{\pi(i)}) = 0$. This means that for $s = (p, q)$ we have $\theta \in \mathcal{M}_s$. If $\mu(\mathcal{M}_s) > 0$ then by Lemma 6 $\mathcal{M}_s = \mathbb{R}^m$, hence we would have $B(\theta, r) \subseteq \mathcal{M}_s$. By Equation (23) it follows that $B(\theta, \eta) \subseteq S^{\partial i}$. This contradicts the fact that θ is in $\partial S^{\partial i}$.

We have shown that $\partial S^{\partial i} \setminus \partial S^{\neg \pi(i)} \subseteq \bigcup_{s \in A} \mathcal{M}_s$, where the sets $\mathcal{M}_s$ have zero Lebesgue measure and $A \subseteq P \times \bigcup_{j=0}^{L} T(j)$ is finite. This implies:

$$\mu(\partial S^{\partial i} \setminus \partial S^{\neg \pi(i)}) \leq \sum_{s \in A} \mu(\mathcal{M}_s) = 0.$$

Using the recursion assumption $\mu(\partial S^{\neg \pi(i)}) = 0$, one concludes that $\mu(\partial S^{\neg i}) = 0$. Hence for the last node L, recalling that $\neg L = P$ one gets $\mu(\partial S^P) = 0$.

□

Lemma 5. *Let i be a node in the graph. Under Assumption **D**, if $\theta \in \mathbb{R}^m \setminus \partial S^{\neg i}$, then there exists $\eta > 0$ and $q \in T(i)$ such that $h_\theta^i = f^q(\theta)$ on the ball $B(\theta, \eta)$, where f^q is real analytic in $\mathbb{R}^m$.*

Proof. We proceed by recursion on the nodes of the network. If $i = 0$, we trivially have $h^0_\theta = X$ which is real analytic on $\mathbb{R}^m$. Assume the result for $\neg\pi(i)$ and let $\theta \in \mathbb{R}^m \setminus \partial S^{\neg i}$. In particular $\theta \in \mathbb{R}^m \setminus \partial S^{\neg\pi(p)}$. By the recursion assumption, we get:

$$(\exists \eta > 0)(\exists q \in T(i))(\forall \theta' \in B(\theta, \eta)) \qquad h^{\pi(i)}_{\theta'} = f^q(\theta') \tag{24}$$

with f^q real analytic in $\mathbb{R}^m$.

If $\theta \notin S^{\partial i}$, then for η small enough $B(\theta, \eta)$ is included in the complement of $S^{\partial i}$, therefore by Assumption **D**, there exists k such that $h^i_{\theta'} = f^k_i(h^{\pi(i)}_{\theta'})$ for all $\theta' \in B(\theta, \eta)$ where f^k_i is real analytic. By (24) we have:

$$h^i_{\theta'} = f^k_i(f^q(\theta')). \tag{25}$$

If $\theta \in S^{\partial i}$, then noting that $\theta \notin \partial S^{\partial i}$, it follows that for η small enough $B(\theta, \eta) \subseteq S^{\partial i}$. Denote by A the set of indices $s \in \partial i$ such that $\theta \in S^s$, therefore $\theta \in \bigcap_{s \in A} S^s$ and $\theta \notin \bigcup_{s \in A^c} S^s$. We will show that for η small enough $B(\theta, \eta) \subseteq \bigcap_{s \in A} S^s$. By contradiction, there exists a sequence θ_n and indices s_n such that $s_n \in A^c$ and θ_n converges towards θ. s_n has a constant subsequence, so we might assume without loss of generality that $s_n = s_0$ for some $s_0 \in A^c$. Since S^{s_0} is a closed set by continuity of the network, it follows that $\theta \in S^{s_0}$ by taking the limit. This contradicts the fact that $\theta \notin \bigcup_{s \in A^c} S^s$. Hence, for η small enough, $B(\theta, \eta) \subseteq \bigcap_{s \in A} S^s$. Again, by Assumption **D** there exists k satisfying (25).

By setting $f^{q_0} = f^k_i(f^q)$ with $q_0 = ((i, k) \oplus q)$, where $\oplus$ denotes concatenation, it finally follows that $h^i_{\theta'} = f^{q_0}(\theta')$ for all θ' in $B(\theta, \eta)$ and f^{q_0} is real analytic on $\mathbb{R}^m$, which concludes the proof. □

Lemma 6. *Let $\theta \mapsto F(\theta) : \mathbb{R}^m \to \mathbb{R}$ be a real analytic function on $\mathbb{R}^m$ and define the set:*

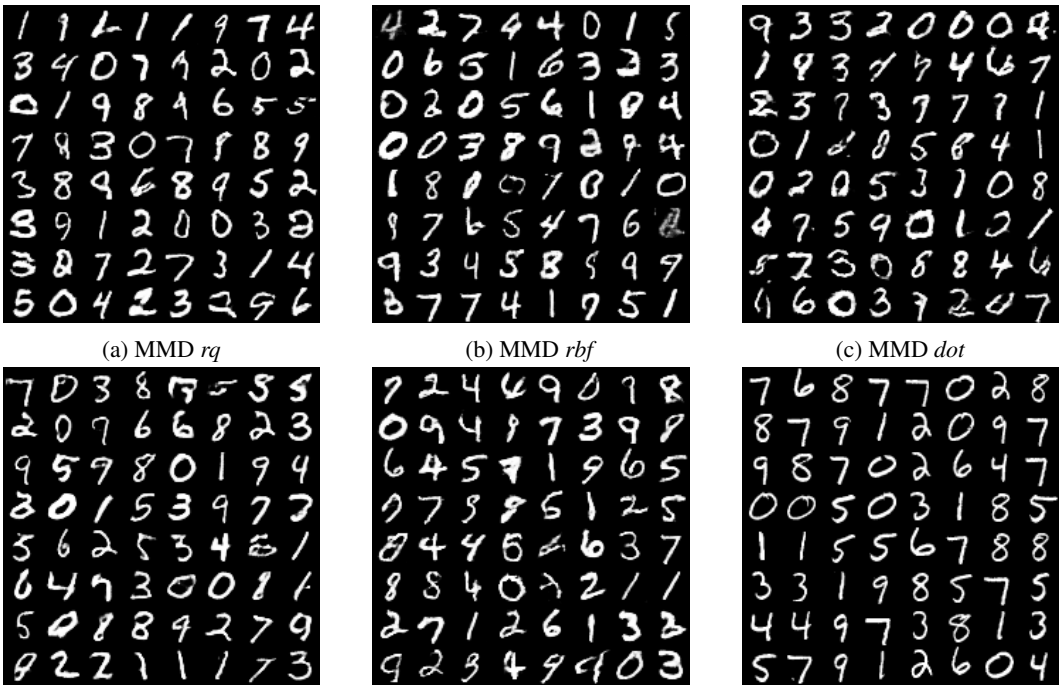
$$\mathcal{M} := \{\theta \in \mathbb{R}^m | F(\theta) = 0\}.$$

Then either $\mu(\mathcal{M}) = 0$ or F is identically zero.

Proof. For a proof of this result, see Proposition 0 of Mityagin (2015). □

D MODEL SAMPLES

Samples from some of the MNIST and CIFAR-10 models are shown in Figure 5 and Figure 6 respectively.



(a) MMD *rq* (b) MMD *rbf* (c) MMD *dot*

(d) Cramer GAN (e) WGAN-GP (f) MNIST test set

Figure 5: Samples from the models listed in Table 1. Rational-quadratic and Gaussian kernels obtain retain sample quality despite reduced discriminator complexity. Each of these models generates good quality samples with the standard DCGAN discriminator (critic size 64).

(a) MMD *rq**, critic size 16 (b) WGAN-GP, critic size 16 (c) Cramér GAN, critic size 16

(d) MMD *rq**, critic size 64 (e) Cramér GAN, critic size 64 (f) Test set

Figure 6: Comparison of samples from various models, as well as true samples from the test set. WGAN-GP samples with critic size 16 are quite bad. Cramér GAN samples with critic size 16 are more appealing to the eye, but seem to have pixel-level issues. Large-critic Cramér and MMD *rq** GAN are of similar quality.