THE DIVERGENCES MINIMIZED BY NON-SATURATING GAN TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Interpreting generative adversarial network (GAN) training as approximate divergence minimization has been theoretically insightful, spurred discussion, and lead to theoretically and practically interesting extensions such as f-GANs and Wasserstein GANs. In this paper we show that the widely used "non-saturating" training scheme can also be interpreted in this way, specifically as minimizing a particular reverse KL-like f-divergence. We also develop a number of theoretical tools to help compare and classify f-divergences. We hope these results may help to clarify some of the theoretical discussion surrounding the divergence minimization view of GAN training.

1 INTRODUCTION

Generative adversarial networks (GANs) (Goodfellow et al., 2014) have enjoyed remarkable progress in recent years, producing images of striking fidelity, resolution and coherence (Karras et al., 2018; Miyato et al., 2018; Brock et al., 2018; Karras et al., 2019). There has been much progress in both theoretical and practical aspects of understanding and performing GAN training (Nowozin et al., 2016; Arjovsky & Bottou, 2017; Arjovsky et al., 2017; Mescheder et al., 2018; Gulrajani et al., 2017; Sønderby et al., 2017; Miyato et al., 2018; Karras et al., 2018; Brock et al., 2018; Karras et al., 2019).

One of the key considerations for GAN training is the scheme used to update the generator and critic. A rich avenue of developments has come from viewing GAN training as *divergence minimization*. Goodfellow et al. (2014) showed the conventional GAN training can be viewed as approximately minimizing the Jensen-Shannon divergence. f-GANs (Nowozin et al., 2016) approximately minimize f-divergences such as reverse KL in a principled way. Wasserstein GANs (Arjovsky et al., 2017) approximately minimize the Wasserstein metric, and combine solid theoretical underpinnings with strong practical results. Nevertheless a relatively unprincipled "non-saturating" scheme (Goodfellow et al., 2014) has continued to obtain groundbreaking results (Karras et al., 2019) and remains a state-of-the-art approach (Lucic et al., 2018).

The effect of the non-saturating scheme on training dynamics, and in particular whether it can be viewed as divergence minimization, has been source of discussion and some confusion since the original formulation of GAN training (Goodfellow et al., 2014). The main result of this paper is to show that the non-saturating scheme approximately minimizes the f-divergence $4\,\mathrm{KL}(\frac{1}{2}p + \frac{1}{2}q \,\|\, p)$, which we refer to as the *softened reverse KL divergence* (§6). This puts non-saturating training on a similar footing to Wasserstein GANs as a theoretically sound approach with strong empirical results. We also discuss how our results relate to previous attempts at this problem and attempt to clarify some of the confusion surrounding the divergence minimization view of non-saturating training.

In order to better understand the qualitative behavior of different divergences such as softened reverse KL, we develop several tools. We show how to write f-divergences in a symmetry-preserving way, allowing easy visual comparison of f-divergences in a way that reflects their qualitative properties (§7). We develop a rigorous formulation of *tail weight* which generalizes the notions of *mode-seeking* and *covering* behavior (§8). Using these tools we show that the softened reverse KL divergence is fairly similar to the reverse KL but very different to the Jensen-Shannon divergence approximately minimized by the original GAN training scheme.

2 PREVIOUS DISCUSSION OF NON-SATURATING GRADIENTS

The precise practical effect of the non-saturating scheme and whether it can be motivated in a principled way have been a source of discussion and some confusion. In this section we review previous attempts to view non-saturating gradients as a form of divergence minimization.

The original GAN paper claims that, compared to the saturating training scheme based on the Jensen-Shannon divergence, the non-saturating training scheme "results in the same fixed point of the dynamics of G and D but provides much stronger gradients early in learning." (Goodfellow et al., 2014, Section 3). It is true that the original and non-saturating generator gradients give the same final result in the non-parametric case where q is unrestricted, but this is fairly trivial since both gradients lead to $q = p$, as do all divergences. It is even true that the dynamics of training are essentially the same for the original and non-saturating gradients when $q \approx p$, but again this is fairly trivial since all f-divergences agree in this regime, as discussed in §3. However the "fixed point of the dynamics" is certainly not the same in the general case of parametric q (see §G for an empirical demonstration). Our results provide a precise way to view the relationship between saturating and non-saturating generator gradients: They are optimizing different f-divergences.

The original f-GAN paper presents a simple argument that the "non-saturating" training scheme has the same fixed points and that the original and non-saturating generator gradients have the same direction (Nowozin et al., 2016, Section 3.2)[1]. However this argument is erroneous. It is true that if $p \approx q$ then $(f^*)'(f'(u))$ is approximately 1 everywhere, and so the original and non-saturating generator gradients are approximately equal, but this is true of any f-divergence. There is no guarantee that the regime $p \approx q$ will ever be approached in the general case where q belongs to a parametric family, it is not the case that the original and non-saturating generator gradients point in approximately the same direction in general (see §G for an empirical demonstration). In fact, the non-saturating form of generator gradient can have completely different qualitative behavior. For example, we show that the non-saturating KL scheme in fact optimizes reverse KL.

A recent paper showed experimentally that the non-saturating generator gradient can successfully learn a distribution in a case where optimizing Jensen-Shannon divergence should fail, and used this to argue that perhaps it is not particularly helpful to view GANs as optimizing Jensen-Shannon divergence (Fedus et al., 2018). The divergence optimized in practice for parametric critics is not exactly the divergence which would be optimized by the theoretically optimal critic, and this distinction seems particularly important in the situation where p and q initially have non-overlapping support. However the fact that non-saturating training is not optimizing Jensen-Shannon is also highly relevant to this discussion, since the gradient in the limit of zero noise is zero for Jensen-Shannon but sizeable for softened reverse KL. Thus the success of non-saturating GAN training in practice may be as much due to its optimizing a different divergence as it is to using an inexact critic.

Arjovsky and Bottou correctly recognize that the non-saturating generator gradient results in approximately minimizing a different objective function and derive the function for classic GANs (Arjovsky & Bottou, 2017, Section 2.2.2). The objective function there is expressed as

$$\mathrm{KL}(q \,\|\, p) - 2\,\mathrm{JS}(p, q) \tag{1}$$

which is a slightly convoluted form of the expression $2\,\mathrm{KL}(\frac{1}{2}p + \frac{1}{2}q \,\|\, p)$ we derive below. The paper suggests the negative sign of the second term is "pushing for the distributions to be different, which seems like a fault in the update", whereas our expression for the divergence makes it clear that this is not an issue.

Poole et al. (2016) present a very similar view to that presented in this paper, including recognizing that the generator and critic may be trained to optimize different f-divergences and interpreting the classic non-saturating generator gradient as a hybrid scheme of this form where the generator gradient is based on a new f-divergence (Poole et al., 2016). However the f-divergence derived there is $f(u) = \log(1 + u^{-1})$, which differs from (50) by a factor of $u + 1$. We refer to this as the *improved generator objectives for GANs (IGOG)* divergence. It can be written as $D_f(p, q) = 2\,\mathrm{KL}(m \,\|\, p) + \mathrm{KL}(p \,\|\, m)$ where $m = \frac{1}{2}p + \frac{1}{2}q$. It has $f''(u) = u^{-2} - (1 + u)^{-2} = \frac{2u+1}{(1+u)^2 u^2}$, and has $(2, 0)$ tail weights. Figure 3 shows that this divergence is qualitatively quite similar to the softened reverse KL but is not identical. The source of the discrepancy between our results and theirs is matching the value instead of the gradient, and is described in detail in §A.

[1] Only in the final NIPS version of the paper, not the arxiv preprint.

3 THE FAMILY OF F-DIVERGENCES

We start by reviewing the definition of an f-divergence (Ali & Silvey, 1966) and establishing some basic properties. These properties are described in more detail in §B. Throughout the paper we use the convention that p is the "true" distribution and q is a model intended to approximate p.

Given a strictly convex twice continuously differentiable function $f : \mathbb{R}_{>0} \to \mathbb{R}$ with $f(1) = 0$, the f-divergence between probability distributions with densities[2] p and q over $\mathbb{R}^K$ is defined as:[3]

$$D_f(p, q) = \int q(x) f\left(\frac{p(x)}{q(x)}\right) \mathrm{d}x \tag{2}$$

f-divergences satisfy several mathematical properties. Firstly D_f is linear in f. Secondly $D_f(p, q) \geq 0$ for all distributions p and q with equality iff $p = q$. This justifies referring to D_f as a divergence. D_f is completely determined by f''. As we will see, the algebraic form of f'' is often simpler than that of f. All f-divergences agree up to an overall scale factor on the divergence between nearby distributions: If $p \approx q$ then $D_f(p, q) \approx f''(1) \operatorname{KL}(p \,\|\, q)$ (see §B). This can also be seen in Figure 2, where all f-divergences approximately overlap near zero. If $f'(1) = 0$ and $f''(1) = 1$ then we say f is in *canonical form*. We can always find such an f by appropriately scaling D_f. Using canonical form removes a superficial difference in scaling between different f-divergences, making them easier to compare, e.g. in Figure 2.

The definition (2) appears to be quite asymmetric in how it treats p and q, but it obeys a particular symmetry (Reid & Williamson, 2011). Let $f_{\mathrm{R}}(u) = uf(u^{-1})$. Then $f'_{\mathrm{R}}(u) = f(u^{-1}) - u^{-1}f'(u^{-1})$ and so $f''_{\mathrm{R}}(u) = u^{-3}f''(u^{-1})$. It is easy to verify that $D_{f_{\mathrm{R}}}(p, q) = D_f(q, p)$. With $A = \{x : q(x) > p(x)\}$ and $B = \{x : q(x) < p(x)\}$, we have

$$D_f(p, q) = \int_A q(x) f\left(\frac{p(x)}{q(x)}\right) \mathrm{d}x + \int_B p(x) f_{\mathrm{R}}\left(\frac{q(x)}{p(x)}\right) \mathrm{d}x \tag{3}$$

This is more explicitly symmetric than (2) in the role of p and q. We refer to A as the set of *left mismatches* ($q > p$), and B as the set of *right mismatches* ($q < p$). At each point in A, the two distributions p and q are somewhat mismatched, and the penalty paid for this mismatch in terms of the overall divergence D_f is governed by the behavior of $f(u)$ for $0 < u < 1$ (the "left" of the graph of f). Similarly the penalty paid for right mismatches is governed by $f(u)$ for $u > 1$. Note from (3) that a left mismatch can only be heavily penalized if the point is plausible under q, i.e. $q(x)$ is not tiny. Similarly a right mismatch can only be heavily penalized for points plausible under p.

4 VARIATIONAL DIVERGENCE ESTIMATION

f-GANs are based on an elegant way to estimate the f-divergence between two distributions given only samples from the two distributions (Nguyen et al., 2010). In this section we review this approach to *variational divergence estimation*. See §E for details on how our derivation and notation relates to that of Nowozin et al. (2016).

There is an elegant variational bound on the f-divergence $D_f(p, q)$ between two densities p and q. Since f is strictly convex, its graph lies at or above any of its tangent lines and only touches in one place. That is, for $k, u > 0$,

$$f(k) \geq f(u) + (k - u)f'(u) = kf'(u) - \big[uf'(u) - f(u)\big] \tag{4}$$

with equality iff $k = u$. This inequality is illustrated in the appendix in Figure 4. Substituting $p(x)/q(x)$ for k and $u(x)$ for u, for any continuously differentiable function $u : \mathbb{R}^K \to \mathbb{R}_{>0}$ we obtain

$$D_f(p, q) \geq \int p(x) f'(u(x)) \,\mathrm{d}x - \int q(x) \left[u(x)f'(u(x)) - f(u(x))\right] \mathrm{d}x \tag{5}$$

[2]Most results also hold for "discrete" probability distributions. The only difference is that the reparameterization trick can no longer be used to reduce variance of the finite sample approximations.

[3]For simplicity, we assume the probability distributions are suitably nice, e.g. absolutely continuous with respect to the Lebesgue measure on $\mathbb{R}^K$, $p(x), q(x) > 0$ for $x \in \mathbb{R}^K$, and p and q continuously differentiable.

with equality iff $u = u^*$, where $u^*(x) = p(x)/q(x)$. The function u is referred to as the *critic*. It will be helpful to have a concise notation for this bound. Writing $u(x) = \exp(d(x))$ without loss of generality, for any continuously differentiable function $d : \mathbb{R}^K \to \mathbb{R}$, we have

$$D_f(p, q) \geq E_f(p, q, d) \tag{6}$$

with equality iff $d = d^*$, where

$$E_f(p, q, d) = \int p(x) a_f(d(x)) \, \mathrm{d}x - \int q(x) b_f(d(x)) \, \mathrm{d}x \tag{7}$$

$$a_f(d) = f'(\exp(d)) \tag{8}$$

$$b_f(d) = \exp(d) f'(\exp(d)) - f(\exp(d)) \tag{9}$$

$$d^*(x) = \log p(x) - \log q(x) \tag{10}$$

Note that both a_f and b_f are linear in f. Their derivatives $a'_f(\log u) = u f''(u)$ and $b'_f(\log u) = u^2 f''(u)$ depend on f only through f''.

The bound (6) leads naturally to *variational divergence estimation*. The f-divergence between p and q can be estimated by maximizing E_f with respect to d (Nguyen et al., 2010). Conveniently E_f is expressed in terms of expectations and may be approximately computed and maximized with respect to d using only samples from p and q. If we parameterize d as a neural net d_ν with parameters ν then we can approximate the divergence by maximizing $E_f(p, q, d_\nu)$ with respect to ν. This does not compute the exact divergence because there is no guarantee that the optimal function d^* lies in the family $\{d_\nu : \nu\}$ of functions representable by the neural net, but we hope that for sufficiently flexible neural nets the approximation will be close.

Here we briefly summarize the three main f-divergences we consider. The Kullback-Leibler (KL) divergence $\mathrm{KL}(p \,\|\, q)$ has $f''(u) = u^{-1}$, $a_f(d) = d$ and $b_f(d) = \exp d - 1$. It has $(1, 2)$ tail weights and is left-bounded and right-unbounded. The reverse KL divergence $\mathrm{KL}(q \,\|\, p)$ has $f''(u) = u^{-2}$, $a_f(d) = \exp(-d) - 1$ and $b_f(d) = d$. It has $(2, 1)$ tail weights and is left-unbounded and right-bounded. Finally the canonicalized Jensen-Shannon divergence $4\,\mathrm{JS}(p, q) = 2\,\mathrm{KL}(p \,\|\, m) + 2\,\mathrm{KL}(q \,\|\, m)$, where $m = \frac{1}{2}p + \frac{1}{2}q$, has $f''(u) = \frac{2}{u(u+1)}$, $a_f(d) = 2\log\sigma(d) + 2\log 2$ and $b_f(d) = -2\log\sigma(-d) - 2\log 2$. It has $(1, 1)$ tail weights and is bounded. See §D for details.

5 Variational divergence minimization

f-GANs (Nowozin et al., 2016) generalize classic GANs to allow approximately minimizing any f-divergence. In this section we briefly review and discuss the f-GAN formulation.

Consider the task of estimating a probabilistic model from data using an f-divergence. Here p is the true distribution and the goal is to minimize $l(\lambda) = D_f(p, q_\lambda)$ with respect to λ, where $\lambda \mapsto q_\lambda$ is a parametric family of densities over $\mathbb{R}^K$. We refer to q_λ as the *generator*. For implicit generative models such as typical GAN generators, the distribution q_λ is the result of a deterministic transform $\overline{x}_\lambda(z)$ of a stochastic latent variable z. However we do not need to assume this specific form for most of our discussion.

We first note that the variational divergence bound E_f satisfies a convenient gradient matching property. This is not made explicit in the original f-GAN paper. Denote the optimal d given p and q_λ by d^*_λ. We saw above that $D_f(p, q_\lambda)$ and $E_f(p, q_\lambda, d)$ match values at $d = d^*_\lambda$. They also match gradients: From the definitions of D_f and E_f we can verify that they have the same gradient with respect to the generator parameters λ:

$$\frac{\partial}{\partial\lambda} D_f(p, q_\lambda) = \frac{\partial}{\partial\lambda} E_f(p, q_\lambda, d)\bigg|_{d=d^*_\lambda} = -\int \left[\frac{\partial}{\partial\lambda} q_\lambda(x)\right] b_f(d^*_\lambda(x)) \, \mathrm{d}x \tag{11}$$

We can minimize $D_f(p, q_\lambda)$ using *variational divergence minimization*, maximizing $E_f(p, q_\lambda, d_\nu)$ with respect to ν while minimizing it with respect to λ. Adversarial optimization such as this lies at

the heart of all flavors of GAN training. Define $\overline{\lambda}$ and $\overline{\nu}$ as

$$\overline{\lambda} = -\frac{\partial}{\partial \lambda} E_f(p, q_\lambda, d_\nu) = \int \left[\frac{\partial}{\partial \lambda} q_\lambda(x)\right] b_f(d_\nu(x)) \, \mathrm{d}x \tag{12}$$

$$\overline{\nu} = \frac{\partial}{\partial \nu} E_f(p, q_\lambda, d_\nu) \tag{13}$$

To perform the adversarial optimization, we can feed $\overline{\lambda}$ and $\overline{\nu}$ (or in practice, stochastic approximation to them) as the gradients into any gradient-based optimizer designed for minimization, e.g. stochastic gradient descent or ADAM. There is a simple generalization of the above training procedure, which is to base the generator gradients on E_f but the critic gradients on E_g for a possibly different function g (Poole et al., 2016, Section 2.3). We refer to this as using *hybrid* (f, g) gradients. This also approximately minimizes D_f. See §F for more details.

When training classic GANs in practice, an alternative *non-saturating* loss is used as the basis for the generator gradient, and is found to perform much better in practice (Goodfellow et al., 2014). This issue has been discussed in detail previously, so we just give a summary here and discuss in more detail in §F. Early on in training, the generator and data distribution are typically not well matched, with samples from p being very unlikely under q and vice versa. This means most of the probability mass of p and q is in regions where d has large magnitude, corresponding to the positive and negative tails in Figure 2 and (19). In this regime Jensen-Shannon has very flat gradient, and it is not too surprising that this might lead to optimization issues. Similar concerns do not apply to other f-divergences such as KL or reverse KL, but an alternative "non-saturating" generator gradient has still been suggested for use in f-GANs (Nowozin et al., 2016). For both GANs and f-GANs the specific change is to replace b_f by a_f in the definition of $\overline{\lambda}$ in (12). We are not aware of a particular motivation for this procedure in the case of f-GANs other than that it yields the traditional non-saturating GAN scheme in the case of Jensen-Shannon.

6 Effect of non-saturating gradients

We now discuss the effect of the non-saturating generator gradient on training. We show that, for an optimal critic, the non-saturating generator gradient is the gradient of a globally coherent objective function, that this objective function is an f-divergence, and that this f-divergence is not the same as the one optimized by using the original "saturating" gradient. We explicitly derive the divergences optimized by the "non-saturating" KL, reverse KL and Jensen-Shannon training schemes.

We first establish our main result: "Non-saturating" training based on g is precisely equivalent to a hybrid (f, g) scheme for some f. Consider the f-divergence D_f defined by

$$f''(u) = u^{-1} g''(u) \tag{14}$$

This is a valid f-divergence since $f''(u) > 0$. It is straightforward to verify that $b'_f = a'_g$, so $b_f = a_g + k$, where the constant $k \in \mathbb{R}$ does not affect the (reparameterized) gradients. Since the non-saturating gradient uses b instead of a in the definition of its generator gradient $\overline{\lambda}$, an original generator gradient using f is the same as a non-saturating generator gradient using g. Since the critic gradient is still based on g, the overall scheme is a hybrid (f, g) one, and so approximately minimizes D_f.

We now explicitly compute the corresponding f for some common choices of g. It is easy to show that if D_g has (R, S) tails then D_f has $(R+1, S-1)$ tails, so the divergence effectively optimized by non-saturating training penalizes left mismatches more strongly and right mismatches less strongly than the original divergence. For the KL divergence, $g''(u) = u^{-1}$, so $f''(u) = u^{-2}$. We already saw in §4 that this is the reverse KL divergence. Thus "non-saturating" training based on the KL divergence is a hybrid (reverse KL, KL) scheme, and so in fact approximately minimizes the reverse KL. This equivalence also follows directly from the equality of KL's a_g to reverse KL's b_f. For the reverse KL divergence, $g''(u) = u^{-2}$, so $f''(u) = u^{-3}$. The corresponding f may be obtained by integrating twice, choosing constants of integration such that f is canonical. We show in §D that D_f is the canonicalized Pearson χ^2 divergence. It has $(3, 0)$ tail weights and is left-unbounded and right-bounded. Thus "non-saturating" training based on the reverse KL divergence is a hybrid ($\frac{1}{2}\chi^2$, reverse KL) scheme, and so approximately minimizes the Pearson χ^2 divergence. For the canonicalized Jensen-Shannon divergence, $g''(u) = \frac{2}{u(u+1)}$, so $f''(u) = \frac{2}{u^2(u+1)}$. The corresponding f may

again be obtained by integrating twice, choosing constants of integration such that f is canonical. We show in §D that this corresponds to $D_f(p,q) = 4\,\mathrm{KL}(\frac{1}{2}p + \frac{1}{2}q \,\|\, p)$. This divergence does not have an existing name as far as we are aware. In this paper we have termed it the *softened reverse KL (SRKL) divergence* (see §C for details on this terminology). It has $a_f(d) = -2\exp(-d) - 2\log\sigma(d)$ and $b_f(d) = -2\log\sigma(d)$. It has $(2,0)$ tail weights and is left-unbounded and right-bounded. Thus the non-saturating training scheme described by Goodfellow et al. (2014) is a hybrid (SRKL, JS) scheme, and so approximately minimizes the softened reverse KL.

Having derived our main result that the typical non-saturating GAN training scheme effectively optimizes the softened reverse KL divergence, we focus on understanding the qualitative properties of this divergence. We do this by developing some analytic tools applicable to any f-divergence.

7 Pushforwards and symmetry-preserving divergence plots

While f-divergences unify many divergences, just plotting the function f is often not informative. The symmetric relationship between divergences such as KL and reverse KL is obfuscated, and f may grow quickly even when the divergence is well-behaved. In this section we develop a straightforward and intuitive way to compare f-divergences visually through a *symmetry-preserving divergence plot*. Our perspective also allows a simple summary of the prevalence of mismatches between p and q, through a *pushforward plot*.

Firstly note that for $x \sim q(x)$, $p(x)/q(x)$ is a random variable with some distribution. In fact, since (2) is the expected value of some function of this random variable, $D_f(p,q)$ must depend only on the one-dimensional distribution of this random variable and not on the detailed distribution of p and q in space. Formally the distribution of this random variable may be described as the *pushforward measure* of q through the function $u^*(x) = p(x)/q(x)$. To obtain more intuitive plots, we will work in terms of $d^*(x) = \log p(x) - \log q(x)$ instead of u^*. We denote the density of the pushforward of q through d^* by $\tilde{q}_{d^*}(d)$. Rewriting the expectation in (2), we obtain

$$D_f(p,q) = \int \tilde{q}_{d^*}(d) f(\exp d)\,\mathrm{d}d \tag{15}$$

As above we can write this more symmetrically. Define

$$s_f(d) = \begin{cases} f(\exp d), & d < 0 \\ f_{\mathrm{R}}\left(\exp(-d)\right), & d > 0 \end{cases} \tag{16}$$

By considering expectations of an arbitrary function of d expressed in x-space and d-space, we can show that

$$\tilde{q}_{d^*}(d) = \tilde{p}_{d^*}(d)\exp(-d) \tag{17}$$

Thus, using (3) and (17), we can write the f-divergence as

$$D_f(p,q) = \int_{-\infty}^{0} \tilde{q}_{d^*}(d) s_f(d)\,\mathrm{d}d + \int_{0}^{\infty} \tilde{p}_{d^*}(d) s_f(d)\,\mathrm{d}d \tag{18}$$

$$= \int_{-\infty}^{\infty} \max\left\{\tilde{p}_{d^*}(d), \tilde{q}_{d^*}(d)\right\} s_f(d)\,\mathrm{d}d \tag{19}$$

An f-divergence $D_f(p,q)$ involves an interaction between the distributions p, q and the function f, and (19) nicely decomposes this interaction in terms of something that only depends on p and q (the pushforwards) and something that only depends on f (the function s_f), connected via a one-dimensional integral. By plotting s_f and imagining integrating against various pushforwards, we can see the properties of different f-divergences in a very direct way. By plotting the pushforwards, we can get a feel for what types of mismatch between p and q are present in multidimensional x-space, and understand at a glance how badly these mismatches would be penalized for a given f-divergence.

Examples of pushforwards for the simple case where p and q are multidimensional Gaussians with common covariance are shown in Figure 1. In this case the pushforwards $\tilde{q}_{d^*}$ and $\tilde{p}_{d^*}$ are themselves one-dimensional Gaussians (since d^* is linear), with densities $\mathcal{N}(-\frac{1}{2}\sigma^2, \sigma^2)$ and $\mathcal{N}(\frac{1}{2}\sigma^2, \sigma^2)$ respectively, for some σ (this follows from (17)). Examples of s_f for various f-divergences are shown in Figure 2. We refer to s_f as a *symmetry-preserving* representation of f. Note that as long as f is in

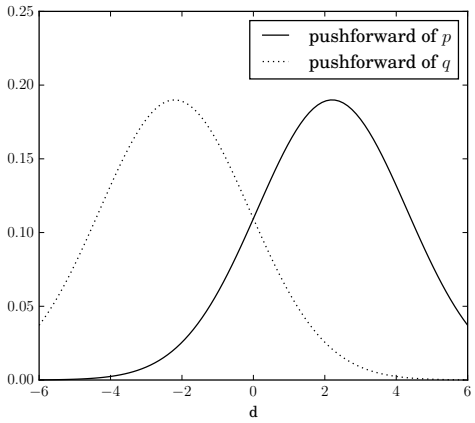

Figure 1: Plots of the pushforward densities $\tilde{p}_{d^*}(d)$ and $\tilde{q}_{d^*}(d)$ for the case where p and q are multidimensional Gaussians with common covariance. The f-divergence for a given f may be obtained by integrating these pushforwards against s_f in Figure 2 using (19).

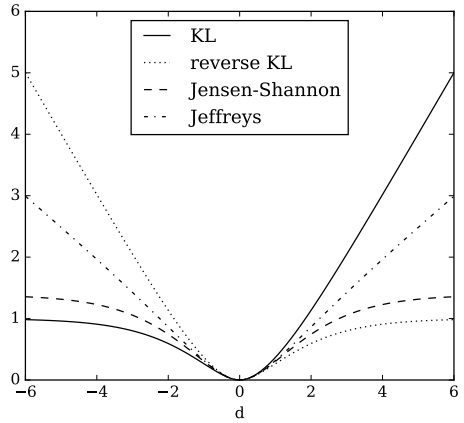

Figure 2: Plots of $s_f(d)$ for various f-divergences. The f-divergence for a given p and q may be obtained by integrating s_f against the pushforwards of p and q such as those shown in Figure 1 using (19). Symmetries such as that between KL and reverse KL are evident.

canonical form, s_f is twice continuously differentiable at zero. Figure 2 directly expresses several facts about divergences. It shows that left mismatches (regions of space where $q(x) > p(x)$, corresponding to $d < 0$) are penalized by reverse KL much more severely than right mismatches (regions of space where $q(x) > p(x)$, corresponding to $d > 0$). The symmetry between KL and reverse KL is evident. We see that Jensen-Shannon and the Jeffreys divergence (the average of KL and reverse KL) are both symmetric in how they penalize left and right mismatches, but differ greatly in how much they penalize small versus large mismatches.

Applying the tools developed in this section to analyze the non-saturating variant of GAN training, Figure 3 shows the symmetry-preserving representation $s_f(d)$ for the Jensen-Shannon and softened reverse KL divergences, as well as the reverse KL for comparison. The qualitative behavior of softened reverse KL is quite similar to reverse KL. As discussed in §C, softening has the potential to make large right mismatches much less severely penalized, thus making the divergence more mode-seeking. Here softening increases the slope of the left tail and changes the right tail behavior slightly, but these changes are relatively minor modifications. The Jensen-Shannon is extremely different to the reverse KL and softened reverse KL.

8 Classification of f-divergence tails

In this section we introduce a classification scheme for f-divergences in terms of their behavior for large left and right mismatches. While different f-divergences differ in details, this classification determines many aspects of their qualitative behavior.

First we define the notion of tail weight and examine some of its consequences. If $f''(u) \sim Cu^{-R}$ as $u \to 0$ for $C > 0$ and $f''(u) \sim Du^{S-3}$ as $u \to \infty$ for $D > 0$ then we say that D_f has (Cu^{-R}, Du^{S-3}) tails and (R, S) tail weights. Here we have used the notation $g(u) \sim h(u)$ as $u \to a$ to mean $g(u)/h(u) \to 1$ as $u \to a$. Note that, since $f''_{\mathrm{R}}(u) = u^{-3} f''(u^{-1})$, f having a u^{S-3} right tail is equivalent to f_{R} having a u^{-S} left tail. Thus tail weights interact simply with symmetry: If D_f has (R, S) tail weights then $D_{f_{\mathrm{R}}}$ has (S, R) tail weights. Intuitively, the left tail weight R determines how strongly large left mismatches are penalized compared to small mismatches (which are penalized the same amount by every canonical f-divergence), whereas the right tail weight S determines how strongly large right mismatches are penalized compared to small mismatches.

Some f-divergences such as Jensen-Shannon are bounded, while others such as KL are unbounded, and it is useful to have a characterization of when boundedness occurs. We say D_f is bounded if

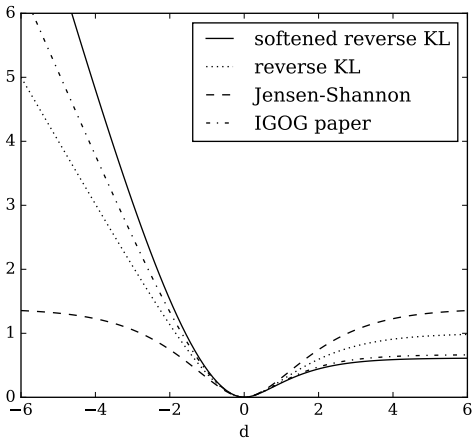

Figure 3: Plots of $s_f(d)$ for various reverse KL-like f-divergences. Softened reverse KL is the divergence effectively minimized by non-saturating GAN training. IGOG is the divergence derived by Poole et al. (2016).

divergence	tail weight	boundedness	
	(left, right)	(left, right)	overall
KL	$(1, 2)$	$(0, \infty)$	∞
RKL	$(2, 1)$	$(\infty, 0)$	∞
Jensen-Shannon	$(1, 1)$	$(0, 0)$	0
Jeffreys	$(2, 2)$	(∞, ∞)	∞
Pearson χ^2	$(3, 0)$	$(\infty, 0)$	∞
softened RKL	$(2, 0)$	$(\infty, 0)$	∞
IGOG	$(2, 0)$	$(\infty, 0)$	∞

Table 1: Tail weights and boundedness for the f-divergences considered in this paper. For boundedness, 0 denotes bounded and ∞ denotes unbounded. A divergence is bounded if and only if left and right tail weights are both less than 2.

there is an $M \in \mathbb{R}$ such that $D_f(p, q) \leq M$ for all densities p and q. We say f is *left-bounded* if f is bounded on $(0, 1)$, and *right-bounded* if f_{R} is bounded on $(0, 1)$, or equivalently if $f(u)/u$ is bounded on $u > 1$. From (3) it is easy to see that if f is left-bounded and right-bounded then D_f is bounded. The converse is also true: If f is left-unbounded or right-unbounded then we can find p and q with arbitrarily large divergence $D_f(p, q)$. This can be seen for example by partitioning $\mathbb{R}^K$ into two sets A and B and considering densities p and q which are constant on A and constant on B, or strictly speaking smooth approximations thereof. Tail weight determines boundedness. It can be checked by integrating and bounding that a divergence with (R, S) tail weights is left-bounded iff $R < 2$ and right-bounded iff $S < 2$. Thus D_f is bounded iff $R, S < 2$. The tail weights and boundedness properties of various f-divergences considered in this paper are summarized in Table 1. Boundedness properties can also be seen in Figure 2. Left and right boundedness of f is trivially equivalent to left and right boundedness of s_f. Thus we can see that reverse KL is left unbounded but right bounded, for example. The unbounded tails in this plot are all asymptotically linear in d.

Tail weights provide an extension of the typical classification of divergences as *mode-seeking* or *covering* (Bishop, 2006, Section 10.1.2). Models trained with reverse KL tend to have distributions which are more compact than the true distribution, sometimes only successfully modeling certain modes (density peaks) of a multi-modal true distribution. Models trained with KL tend to have distributions which are less compact than the true distribution, "covering" the true distribution entirely even if it means putting density in regions which are very unlikely under the true distribution (Bishop, 2006, Figure 10.3). However there are important qualitative aspects of divergence behavior that are not captured by these labels. For example, Jensen-Shannon is neither mode-seeking nor covering: It would be more accurate to say that a model trained using Jensen-Shannon tries to match very closely when it matches, but doesn't worry overly about large mismatches in either direction. The Jeffreys divergence is also symmetric and so neither mode-seeking nor covering, but has very different behavior from Jensen-Shannon. Tail weights capture these distinctions in a straightforward but precise way.

Tail weights and boundedness provide an extremely concise way to see the qualitative effect of using the non-saturating variant of GAN training. The softened reverse KL divergence effectively optimized by conventional non-saturating GAN training has tail weights (2, 0), and so is unbounded, is likely to have strong gradients starting from a random initialization where large mismatches are present, and penalizes left mismatches strongly but tolerates large right mismatches and so is mode-seeking. In contrast the Jensen-Shannon divergence effectively optimized by saturating GAN training has tail weights (1, 1), and so is bounded, is likely to have weak gradients in the presence of large mismatches, and tolerates large left and right mismatches.

REFERENCES

Syed Mumtaz Ali and Samuel D Silvey. A general class of coefficients of divergence of one distribution from another. *Journal of the Royal Statistical Society: Series B (Methodological)*, 28(1): 131–142, 1966.

Martin Arjovsky and Lon Bottou. Towards principled methods for training generative adversarial networks. In *Proc. ICLR*, 2017.

Martin Arjovsky, Soumith Chintala, and Lon Bottou. Wasserstein generative adversarial networks. In *Proc. ICML*, pp. 214–223, 2017.

David Balduzzi, Sebastien Racaniere, James Martens, Jakob Foerster, Karl Tuyls, and Thore Graepel. The mechanics of n-player differentiable games. In *Proc. ICML*, 2018.

Christopher M Bishop. *Pattern recognition and machine learning*. Springer, 2006.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *Proc. ICLR*, 2018.

Sung-Hyuk Cha. Comprehensive survey on distance/similarity measures between probability density functions. *International Journal of Mathematical Models and Methods in Applied Sciences*, 2007.

William Fedus, Mihaela Rosca, Balaji Lakshminarayanan, Andrew M. Dai, Shakir Mohamed, and Ian Goodfellow. Many paths to equilibrium: GANs do not need to decrease a divergence at every step. In *Proc. ICLR*, 2018.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pp. 2672–2680, 2014.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of Wasserstein GANs. In *Advances in Neural Information Processing Systems*, pp. 5767–5777, 2017.

Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. In *Proc. ICLR*, 2018.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4401–4410, 2019.

Mario Lucic, Karol Kurach, Marcin Michalski, Sylvain Gelly, and Olivier Bousquet. Are GANs created equal? A large-scale study. In *Advances in neural information processing systems*, pp. 700–709, 2018.

Lars Mescheder, Sebastian Nowozin, and Andreas Geiger. The numerics of GANs. In *Advances in Neural Information Processing Systems*, pp. 1825–1835, 2017.

Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for GANs do actually converge? In *Proc. ICML*, 2018.

Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *Proc. ICLR*, 2018.

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

XuanLong Nguyen, Martin J Wainwright, and Michael I Jordan. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 56(11):5847–5861, 2010.

Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-GAN: Training generative neural samplers using variational divergence minimization. In *Advances in Neural Information Processing Systems*, pp. 271–279, 2016.

Wei Peng, Yuhong Dai, Hui Zhang, and Lizhi Cheng. Training GANs with centripetal acceleration. *arXiv preprint arXiv:1902.08949*, 2019.

Ben Poole, Alexander A Alemi, Jascha Sohl-Dickstein, and Anelia Angelova. Improved generator objectives for GANs. In *Proc. NIPS Workshop on Adversarial Training*, 2016.

Mark D Reid and Robert C Williamson. Information, divergence and risk for binary experiments. *Journal of Machine Learning Research*, 12(Mar):731–817, 2011.

Casper Kaae Sønderby, Jose Caballero, Lucas Theis, Wenzhe Shi, and Ferenc Huszár. Amortised MAP inference for image super-resolution. In *Proc. ICLR*, 2017.

A Further comparison to previous work

As mentioned in §2, Poole et al. (2016) present a very similar view to that presented in this paper, including recognizing that the generator and critic may be trained to optimize different f-divergences and interpreting the classic non-saturating generator gradient as a hybrid scheme of this form where the generator gradient is based on a new f-divergence (Poole et al., 2016). We now discuss the discrepancy between our result and theirs.

In the language of the present paper, Poole et al. (2016, Equation (8)) define the approximation

$$D_f(p,q) = \int q(x) f(p(x)/q(x))\,\mathrm{d}x \approx \tilde{E}_f(p,q,d) = \int q(x) f(\exp(d(x)))\,\mathrm{d}x \tag{20}$$

and show that the gradients of $\tilde{E}_f$ for this particular f match the non-saturating GAN gradients. This is a valid approximation of the value, since $D_f(p,q) = \tilde{E}_f(p,q,d^*)$ for the optimal critic $d(x) = d^*(x) = \log p(x) - \log q(x)$. However the gradients are not the same: The partial derivative of the left side of (20) with respect to the parameters of q involves two terms, one for each occurrence of $q(x)$ in the integrand, and the partial derivative of the right side only includes one of these. Thus it is not the case that optimizing $\tilde{E}_f$ using gradient descent (while continually keeping the critical optimal) optimizes D_f.

B Properties of f-divergences

In this section we go into more detail about some of the properties of f-divergences which were briefly covered in the main text.

Firstly note that D_f is linear f, that is $D_{f+g} = D_f + D_g$ and $D_{kf} = kD_f$ where $k > 0$. If $f(1) = 0$ and $g(1) = 0$ then $(f+g)(1) = 0$ and $(kf)(1) = 0$, so D_{f+g} and D_{kf} are valid f-divergences. Secondly note that adding an affine term to $f(u)$ does not affect D_f: If $g(u) = f(u) + k - ku$ for $k \in \mathbb{R}$ then $D_g = D_f$. Any affine term added must be of the form $k - ku$ in order to respect the $f(1) = 0$ constraint. Thus the second derivative f'' determines the divergence completely. This property is also true of the various bounds and finite sample approximations[4] derived in this paper, so we may legitimately consider f'' rather than f as the essential quantity of interest for a given divergence. Working with f'' has the added advantage that for many common f-divergences f'' has a simpler algebraic form than f. For any densities p and q we have $D_f(p,q) \geq 0$ with equality iff $p = q$, as can be seen by plugging the constant function $u(x) = 1$ into (5). If $f'(1) = 0$ and $f''(1) = 1$ then we say f is in *canonical form*. We can put any f in canonical form by scaling and adding a suitable affine term, and this corresponds to a scaling of D_f. Each f-divergence has a unique canonical form.

[4] As long as the reparameterization trick is used to compute finite sample approximations, as is standard practice. If a simpler finite sample approximation such as naive REINFORCE is used then k affects the variance of the generator gradient.

Different f-divergences may behave very differently when p and q are far apart but are essentially identical when $q \approx p$. In fact the divergence between nearby distributions belonging to some family is given by $f''(1)$ times the Fisher metric of the family. Specifically

$$D_f\left(q_\lambda, q_{\lambda+\varepsilon v}\right) = \tfrac{1}{2}\varepsilon^2 f''(1) v^\mathsf{T} F(\lambda) v + O(\varepsilon^3) \tag{21}$$

where $\varepsilon \in \mathbb{R}$, $v \in \mathbb{R}^K$, and $F(\lambda) = \sum_x q_\lambda(x)(\frac{\partial}{\partial\lambda}\log q_\lambda(x))(\frac{\partial}{\partial\lambda}\log q_\lambda(x))^\mathsf{T}$ is the Fisher information matrix for the parametric family of distributions specified by q_λ. Alternatively this may be stated in the non-parametric form

$$D_f\left(q, q+\varepsilon v\right) = \tfrac{1}{2}\varepsilon^2 f''(1) \int \frac{\big(v(x)\big)^2}{q(x)}\,\mathrm{d}x + O(\varepsilon^3) \tag{22}$$

where $v : \mathbb{R}^K \to \mathbb{R}$ satisfies $\int v(x)\,\mathrm{d}x = 0$. Informally we may state this as:

$$D_f(p, q) \approx \tfrac{1}{2} f''(1) \int \frac{\big(p(x)-q(x)\big)^2}{p(x)}\,\mathrm{d}x \tag{23}$$

Thus all f-divergences agree up to a constant factor on the divergence between two nearby distributions, and they are all just scaled versions of the Fisher metric in this regime. This can also be seen in Figure 2, where all f-divergences approximately overlap near zero.

C Divergence symmetrization and softening

We can apply some simple operations to a divergence to obtain another divergence. In this section we consider the effect of reversing, symmetrizing and *softening* operations on f-divergences. Many common f-divergences can be obtained from others in this way, and this provides a unified way of concisely describing many f-divergences based on KL, for example.

Consider applying an operation to a divergence $D(p, q)$ to obtain another divergence $\tilde{D}(p, q)$. We already saw the reversing operation $\tilde{D}(p, q) = D(q, p)$ in §3. If D is an f-divergence with function $f(u)$ then D_R is an f-divergence with function $f_\mathrm{R}(u) = uf(u^{-1})$. In this case $f''_\mathrm{R}(u) = u^{-3}f''(u^{-1})$. Symmetrization means $\tilde{D}(p, q) = \frac{1}{2}D(p, q) + \frac{1}{2}D(q, p)$. If D is an f-divergence then $f \mapsto \frac{1}{2}f + \frac{1}{2}f_\mathrm{R}$ enacts symmetrization. Finally *(q-)softening* refers to replacing q with $m = \frac{1}{2}p + \frac{1}{2}q$, i.e. $\tilde{D}(p, q) = 4D(p, m)$. If D is an f-divergence with function f then setting the new $f(u)$ to be $2(1+u)f(\frac{2u}{1+u})$ enacts softening. In this case the new $f''(u)$ is $\frac{8}{(1+u)^3}f''(\frac{2u}{1+u})$. The factor of 4 above is to ensure that the divergence remains canonical after softening, i.e. $f''(1) = 1$. Softening has the potential to make large right mismatches much less severely penalized, since in regions of space where $p(x)/q(x)$ was large because $p(x)$ was moderate and $q(x)$ was tiny, $p(x)/m(x)$ is now approximately 2, so a large right mismatch is only penalized by the softened divergence as much as a moderate right mismatch is penalized by the original divergence. This is reflected in the tail weights: It is easy to show using the tools we have developed above that if the original divergence has (R, S) tail weights then the softened divergence has $(R, 0)$ tail weights.

Many f-divergences can be written concisely as a series of these operations. For example reverse KL is Reverse(KL), Jeffreys is Symmetrize(KL), the canonicalized K-divergence $4\,\mathrm{KL}(p \,\|\, m)$ (Cha, 2007) is Soften(KL) and canonicalized Jensen-Shannon is Symmetrize(Soften(KL)). In this terminology, the main claim of this paper is that the non-saturating procedure for GAN training is in fact effectively minimizing the softened reverse KL divergence $4\,\mathrm{KL}(m \,\|\, p)$ given by Soften(Reverse(KL)).

D Detailed expressions for various f-divergences

In this section we give more details of the f-divergences considered in §4 and §6. The expressions for D_f and E_f are obtained by plugging the chosen f into (2) and (7) respectively.

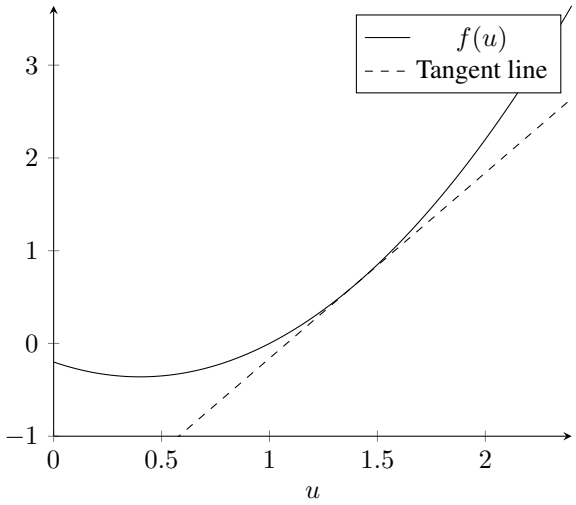

Figure 4: A strictly convex function $f : \mathbb{R}_{>0} \to \mathbb{R}$ and a tangent line. The variational bound used by f-GANs is based on the fact that a strictly convex function f lies at or above its tangent lines.

The KL divergence satisfies:

$$f(u) = u \log u \tag{24}$$
$$f''(u) = u^{-1} \tag{25}$$
$$D_f(p, q) = \mathrm{KL}(p \,\|\, q) = \int p(x) \log \frac{p(x)}{q(x)} \,\mathrm{d}x \tag{26}$$
$$E_f(p, q, d) = 1 + \int p(x) d(x) \,\mathrm{d}x - \int q(x) \exp(d(x)) \,\mathrm{d}x \tag{27}$$
$$a_f(d) = d \tag{28}$$
$$b_f(d) = \exp(d) - 1 \tag{29}$$

The KL divergence has (u^{-1}, u^{-1}) tails, $(1, 2)$ tail weights, and is left-bounded and right-unbounded.

The reverse KL divergence satisfies:

$$f(u) = -\log u \tag{30}$$
$$f''(u) = u^{-2} \tag{31}$$
$$D_f(p, q) = \mathrm{KL}(q \,\|\, p) = \int q(x) \log \frac{q(x)}{p(x)} \,\mathrm{d}x \tag{32}$$
$$E_f(p, q, d) = 1 - \int p(x) \exp(-d(x)) \,\mathrm{d}x - \int q(x) d(x) \,\mathrm{d}x \tag{33}$$
$$a_f(d) = 1 - \exp(-d) \tag{34}$$
$$b_f(d) = d \tag{35}$$

The reverse KL divergence has (u^{-2}, u^{-2}) tails, $(2, 1)$ tail weights, and is left-unbounded and right-bounded.

The Jensen-Shannon divergence $\mathrm{JS}(p, q)$ has $f''(1) = 1/4$ and so is not canonical. In most of the paper we therefore consider the canonicalized Jensen-Shannon divergence $4\,\mathrm{JS}(p, q)$. This satisfies:

$$f(u) = 2u \log u - 2(u+1)\log(u+1) + 4\log 2 \tag{36}$$

$$f''(u) = \frac{2}{u(u+1)} \tag{37}$$

$$D_f(p, q) = 4\,\mathrm{JS}(p, q) \tag{38}$$

$$= 2\,\mathrm{KL}(p \,\|\, \tfrac{1}{2}p + \tfrac{1}{2}q) + 2\,\mathrm{KL}(q \,\|\, \tfrac{1}{2}p + \tfrac{1}{2}q) \tag{39}$$

$$= 4\log 2 + 2\int p(x) \log \frac{p(x)}{p(x) + q(x)}\,\mathrm{d}x + 2\int q(x) \log \frac{q(x)}{p(x) + q(x)}\,\mathrm{d}x \tag{40}$$

$$E_f(p, q, d) = 4\log 2 + 2\int p(x) \log \sigma(d(x))\,\mathrm{d}x + 2\int q(x) \log \sigma(-d(x))\,\mathrm{d}x \tag{41}$$

$$a_f(d) = 2\log \sigma(d) + 2\log 2 \tag{42}$$

$$b_f(d) = -2\log \sigma(-d) - 2\log 2 \tag{43}$$

The canonicalized Jensen-Shannon divergence has $(2u^{-1}, 2u^{-2})$ tails, $(1, 1)$ tail weights, and is both left-bounded and right-bounded and so bounded overall.

The *Pearson χ^2 (or Kagan) divergence* has $f''(1) = 2$ as so is not canonical. The canonicalized Pearson χ^2 divergence satisfies:

$$f(u) = \frac{(u-1)^2}{2u} \tag{44}$$

$$f''(u) = u^{-3} \tag{45}$$

$$D_f(p, q) = \tfrac{1}{2}\int \frac{\big(q(x) - p(x)\big)^2}{p(x)}\,\mathrm{d}x \tag{46}$$

$$E_f(p, q, d) = -\tfrac{1}{2} - \tfrac{1}{2}\int p(x) \exp\big(-2d(x)\big)\,\mathrm{d}x + \int q(x) \exp\big(-d(x)\big)\,\mathrm{d}x \tag{47}$$

$$a_f(d) = \tfrac{1}{2} - \tfrac{1}{2}\exp(-2d) \tag{48}$$

$$b_f(d) = 1 - \exp(-d) \tag{49}$$

The canonicalized Pearson χ^2 divergence has (u^{-3}, u^{-3}) tails, $(3, 0)$ tail weights, and is left-unbounded and right-bounded. The expression for f here corrects a swapped definition in the original f-GAN paper[5] (according to the definitions of the Pearson and Neyman divergences given in the paper, the expression given for the Pearson f is actually the Neyman f and vice versa) (Nowozin et al., 2016). In §6 we discussed the equality of the non-saturating reverse KL generator gradient to the conventional canonicalized Pearson χ^2 generator gradient. This can be seen from (14), as we did in §6, or directly by noting that a_f for the reverse KL divergence is equal to b_f for the Pearson χ^2 divergence.

The softened reverse KL divergence satisfies:

$$f(u) = 2(u+1)\log \frac{u+1}{u} - 4\log 2 \tag{50}$$

$$f''(u) = \frac{2}{u^2(u+1)} \tag{51}$$

$$D_f(p, q) = 4\,\mathrm{KL}(\tfrac{1}{2}p + \tfrac{1}{2}q \,\|\, p) \tag{52}$$

$$E_f(p, q, d) = 2 - 4\log 2 + 2\int p(x) \Big[-\exp\big(-d(x)\big) - \log \sigma\big(d(x)\big)\Big]\,\mathrm{d}x - 2\int q(x) \log \sigma\big(d(x)\big)\,\mathrm{d}x \tag{53}$$

$$a_f(d) = 2\exp(-d) - 2\log \sigma(d) - 2 - 2\log 2$$

$$b_f(d) = 2\log \sigma(d) + 2\log 2$$

[5] In the the arxiv preprint, not the final NIPS version of the paper.

The SRKL divergence has $(2u^{-2}, 2u^{-3})$ tails, $(2, 0)$ tail weights, and is left-unbounded and right-bounded. In §6 we discussed the equality of the non-saturating canonicalized Jensen-Shannon generator gradient to the conventional softened reverse KL generator gradient. This can be seen from (14), as we did in §6, or directly by noting that a_f for the canonicalized Jensen-Shannon divergence is equal to b_f for the softened reverse KL divergence.

E f-GAN NOTATION

The original f-GAN paper (Nowozin et al., 2016) phrases the results presented in §4 in terms of the Legendre transform f^* of f. The two descriptions are equivalent, as can be seen by setting $T(x) = f'(u(x))$ and using the result $f^*(f'(u)) = uf'(u) - f(u)$. We find our description helpful since it avoids having to explicitly match the domain of f^*, ensures the optimal d is the same for all f-divergences, and because the Legendre transform is complicated for one of the divergences we consider. An "output activation" was used in the original f-GAN paper to adapt the output d of the neural net to the domain of f^*. This is equal to $f'(\exp(d))$, up to irrelevant additive constants, for all the divergences we consider, and so our description also matches the original description in this respect.

F MORE VARIATIONAL DIVERGENCE MINIMIZATION

The gradient matching property shows that performing very many critic updates followed by a single generator update is a sensible learning strategy which, assuming the critic is sufficiently flexible and amenable to optimization, essentially performs very slow gradient-based optimization on the true divergence D_f with respect to λ. However in practice performing a few critic updates for each generator update, or simultaneous generator and critic updates, performs well, and it is easy to see that these approaches at least have the correct fixed points in terms of Nash equilibria of E_f and optima of D_f, subject as always to the assumption that the critic is sufficiently richly parameterized. Convergence properties of these schemes are investigated much more thoroughly elsewhere, for example (Nagarajan & Kolter, 2017; Gulrajani et al., 2017; Mescheder et al., 2017; 2018; Balduzzi et al., 2018; Peng et al., 2019), and are not the main focus here.

A similar discussion applies to hybrid schemes. Subject as always to the assumption of a richly parameterized critic, if we perform very many critic updates for each generator update, then the d used to compute the generator gradient will still be close to d^*, and so the generator gradient will be close to the gradient of D_f, even though the path d took to approach d^* was governed by g rather than f. The fixed points of the two gradients are also still correct, and so it seems reasonable to again use more general update schemes and we might hope for similar convergence results (not analyzed here).

For an implicit generative model $x_\lambda(z)$ where $z \sim \mathbb{P}(z)$, we have

$$E_f(p, q_\lambda, d) \stackrel{c}{=} -\int \mathbb{P}(z) b_f\left(d_\nu(\overline{x}_\lambda(z))\right) \mathrm{d}z \tag{54}$$

Thus there is a $b'_f(d)$ factor in the generator gradient, and in fact this is the only way the choice of f-divergence affects the generator gradient. For reverse KL, $b'_f(d) = 1$, allowing the gradients from the other factors to pass freely. Most of the contribution to the initial gradient for reverse KL is likely to come from regions in space with large negative d due to the $\mathbb{P}(z)$ factor. For canonicalized Jensen-Shannon, $b'_f(d) = 2\sigma(d)$, which tends to zero exponentially quickly as $d \to -\infty$ and tends to 2 as $d \to \infty$. Regions of space with large positive d have a tiny contribution to the gradient due to the $\mathbb{P}(z)$ factor, while regions with large negative d are exponentially suppressed by $b'_f(d)$. Based on these considerations it might be tempting to conclude that left-unboundedness is the most important factor in being able to learn from a random initialization. A divergence with left tail weight R has $b'_f(d) \sim \exp(-d(R-2))$ so $R \geq 2$ ensures that $b'_f(d)$ does not decay exponentially as $d \to -\infty$. However the case of KL shows that right-unboundedness is also capable of allowing learning. For KL, $b'_f(d) = \exp d$, and the situation is complicated, since it exponentially magnifies gradients from regions with large positive d, which are extremely unlikely under $\mathbb{P}(z)$. We know the overall gradient can sometimes be a reasonable learning signal, since training models such as a multivariate

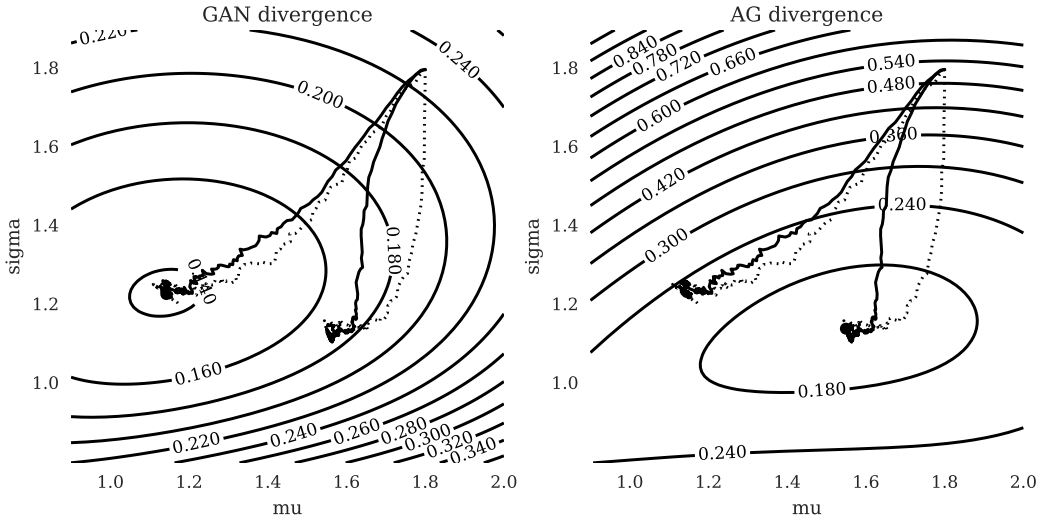

Figure 5: Comparing training using the saturating and non-saturating GAN generator gradients on a toy problem. The true distribution p is a mixture of two 1D Gaussians and the model distribution q is a single Gaussian. Contour plots show the Jensen-Shannon (JS) divergence (left), and softened reverse KL divergence $4\,\mathrm{KL}(\frac{1}{2}p + \frac{1}{2}q \,\|\, p)$ (right) as a function of model parameters. Lines show the progression of SGD-based JS training based on the original, saturating gradient and based on the non-saturating gradient (solid for learned critic; dotted for optimal critic). The original scheme converges to the JS divergence minimum. The non-saturating scheme, which by the results of this paper is equivalent to a hybrid (SRKL, JS) scheme, converges to the SRKL divergence minimum as expected.

Gaussian using KL divergence works well. However even if the expected gradient allows learning, the stochastic approximation obtained by sampling from q is likely to have extremely large variance.

The saturation issue is sometimes presented as being specific to the loss E_f used for classic GAN training, but the gradient matching property presented in §5 shows it is fundamental to the Jensen-Shannon divergence. The more critic updates we perform initially, the more saturated d is on samples from q, and the more closely the gradient of E_f with respect to λ approximates the gradient of the true divergence D_f.

The typical fix to the saturation issue is to use the *non-saturating* generator gradient

$$\overline{\lambda} = \int \left[\frac{\partial}{\partial \lambda} q_\lambda(x)\right] \log \sigma\left(d(x)\right) \mathrm{d}x = \int \mathbb{P}(z) \left[\frac{\partial}{\partial \lambda} \log \sigma\left(d(\overline{x}_\lambda(z))\right)\right] \mathrm{d}x \tag{55}$$

Since the gradient of $\log \sigma(d)$ tends to 1 as d tends to $-\infty$, the gradient used for training is now larger.

G Experimental validation of mathematical result

In order to validate our mathematical conclusions we conducted a simple experiment. Training behavior using the original and non-saturating gradients on a toy problem is shown in Figure 5. We see that the two cases minimize different divergences, as expected based on the theoretical arguments presented above.