

# Designing GANs: A Likelihood Ratio Approach

Kalliopi Basioti<sup>1</sup> George V. Moustakides<sup>2</sup>

## Abstract

We are interested in the design of generative adversarial networks. The training of these mathematical structures requires the definition of proper min-max optimization problems. We propose a simple methodology for constructing such problems assuring, at the same time, that they provide the correct answer. We give characteristic examples developed by our method, some of which can be recognized from other applications and some introduced for the first time. We compare various possibilities by applying them to well known datasets using neural networks of different configurations and sizes.

## 1. Introduction

The problem we are interested in can be summarized as follows: We are given two collections of training data  $\{Z_j\}$  and  $\{X_i\}$ . In the first set the samples follow the *origin* probability density  $h(Z)$  and in the second the *target* density  $f(X)$ . The target density  $f(X)$  is considered *unknown* while  $h(Z)$  can either be *known* with the possibility to produce samples  $Z_j$  every time it is necessary or *unknown* in which case we have a second fixed training set  $\{Z_j\}$ . Our goal is to design a deterministic transformation  $G(Z)$  so that the data  $\{Y_j\}$  produced by applying the transformation  $Y = G(Z)$  onto  $\{Z_j\}$  follow the target density  $f(Y)$ .

Of course one may wonder whether the proposed problem enjoys any solution, namely, whether there indeed exists a transformation  $G(Z)$  capable of transforming  $Z$  into  $Y$  with the former following the origin density  $h(Z)$  and the latter the target density  $f(Y)$ . The problem of transforming random vectors has been analyzed by (Box & Cox, 1964) where *existence* is shown under general conditions. Computing, however, the actual transformation is a completely different challenge with one of the possible solutions rely-

ing on adversarial approaches applied to neural networks.

The most well known usage of this result is, clearly, the possibility to generate synthetic data that follow the unknown target density  $f(X)$ . In this case  $h(Z)$  is selected to be simple (e.g. i.i.d. standard Gaussian or i.i.d. uniform) so that generating realizations from  $h(Z)$  is straightforward. As mentioned, the adversarial approach can be applied even if the origin density  $h(Z)$  is unknown provided that we have a dataset  $\{Z_j\}$  with data following the origin density. When, however,  $h(Z)$  is known and we can generate any number of realizations  $Z_j$ , it is expected, the adversarial approach, to identify the transformation  $G(Z)$  with higher accuracy due to the possibility of generating more training data.

It was (Goodfellow et al., 2016) that first introduced the idea of *adversarial* (min-max) optimization and demonstrated that it results in the determination of the desired transformation  $G(Z)$ . Alternative adversarial approaches by (Arjovsky et al., 2017; Binkowski et al., 2018) were subsequently suggested and shown to also deliver the correct transformation  $G(Z)$ . Finally, we must mention the work by (Nowozin et al., 2016) which is closely related to our results and for which, at the end of Section 2, we give details, emphasizing differences and similarities with our method. As in (Nowozin et al., 2016), we will show that our methods provides an *abundance* of adversarial problems that are capable of identifying the appropriate transformation  $G(Z)$ . Furthermore, we will also provide a *simple recipe* as to how we can successfully construct such problems.

Arguing along the same lines of the existing min-max formulations: We would like to optimally specify a vector transformation  $G(Z)$ , the *generator*, and a scalar function  $D(X)$ , the *discriminator*. To achieve this, for each combination  $\{G(Z), D(X)\}$  we define the cost function

$$\mathcal{J}(G, D) = E_f[\phi(D(X))] + E_h[\psi(D(G(Z)))] \quad (1)$$

where  $\phi(z), \psi(z)$  are two scalar functions of the scalar  $z$  and  $E_f[\cdot], E_h[\cdot]$  denote expectation with respect to the density  $f(X), h(Z)$  respectively. The optimum combination generator/discriminator is then identified by solving the following min-max problem

$$\min_{G(Z)} \max_{D(X)} \mathcal{J}(G, D) =$$

<sup>1</sup>Department of Computer Science, Rutgers University, New Brunswick, NJ, USA. <sup>2</sup>Department of Electrical and Computer Engineering, University of Patras, Patras, Greece.. Correspondence to: K. Basioti <kib21@scarletmail.rutgers.edu>, G. V. Moustakides <moustaki@upatras.gr>.

$$\min_{G(Z)} \max_{D(X)} \{E_f[\phi(D(X))] + E_h[\psi(D(G(Z)))]\}. \quad (2)$$

We must point out that our goal is not to solve (2), but rather *find a class of functions  $\phi(z), \psi(z)$  so that the transformation  $G(Z)$  that will come out of the solution of (2) is such that  $Y = G(Z)$  follows the target density  $f(Y)$  when  $Z$  follows the origin density  $h(Z)$ .*

If  $Z$  is random following  $h(Z)$  then  $Y = G(Z)$  is also random and we denote with  $g(Y)$  its corresponding probability density. Clearly, there exists a correspondence between transformations  $G(Z)$  and densities  $g(Y)$  when the density  $h(Z)$  of  $Z$  is fixed. Since we can write

$$E_h[\psi(D(G(Z)))] = E_g[\psi(D(Y))],$$

this allows us to argue that the min-max problem in (2) is equivalent to

$$\min_{g(Y)} \max_{D(X)} \{E_f[\phi(D(X))] + E_g[\psi(D(Y))]\}. \quad (3)$$

It is now possible to combine the two expectations by applying a change of measure and a change of variables and equivalently write (3) as follows

$$\min_{g(X)} \max_{D(X)} E_f[\phi(D(X)) + r(X)\psi(D(X))], \quad (4)$$

where  $r(X) = \frac{g(X)}{f(X)}$  denotes the corresponding likelihood ratio. Since  $f(X)$  is also fixed, there is again a correspondence between  $r(X)$  and  $g(X)$ , hence the previous min-max problem becomes equivalent to

$$\min_{r(X) \in \mathcal{R}_f} \max_{D(X)} E_f[\phi(D(X)) + r(X)\psi(D(X))]. \quad (5)$$

Here  $\mathcal{R}_f$  denotes the class of all likelihood ratios  $r(X)$  with respect to the density  $f(X)$ , namely, all the functions  $r(X)$  that satisfy

$$\mathcal{R}_f = \left\{ r(X) : r(X) \geq 0, \int r(X)f(X) dX = 1 \right\}. \quad (6)$$

Using these definitions, let us define the cost

$$J(r, D) = E_f[\phi(D(X)) + r(X)\psi(D(X))] \quad (7)$$

and, according to (5), we are interested in the following min-max problem

$$\min_{r(X) \in \mathcal{R}_f} \max_{D(X)} J(r, D). \quad (8)$$

As mentioned, our actual goal is not to solve the adversarial problem. Instead, we would like to properly *identify* pairs of functions  $\{\phi(z), \psi(z)\}$  so that (8) *accepts as solution the function  $r(X) = 1$* . Indeed, if  $r(X) = 1$  is the solution to (8), this means that  $g(X) = f(X)$  is the solution to (3) and, finally, that the optimum  $G(Z)$  obtained from (1) is such that  $Y = G(Z)$  follows  $g(Y) = f(Y)$  which, of course, is our original objective. Even though the min-max problem in (1) is what we attempt to solve, it is through (8) that we understand what its solution entails. In the next section we focus on (7), (8) and propose a simple design method

(recipe) for the two functions  $\phi(z), \psi(z)$  that assures that the solution of (8) is indeed  $r(X) = 1$ .

## 2. A Class of Functions $\phi(z), \psi(z)$

Suppose that  $\omega(r)$  is a *strictly increasing and (left and right) differentiable* scalar function of the nonnegative scalar  $r$ , i.e.  $r \in [0, \infty)$ . Denote with  $\mathcal{I}_\omega = \omega([0, \infty))$  the range of values of  $\omega(r)$  and let  $\omega^{-1}(z)$  be the inverse function of  $\omega(r)$  which is defined for  $z \in \mathcal{I}_\omega$ . Let  $\rho(z) > 0$  be a positive scalar function also defined for  $z \in \mathcal{I}_\omega$  then, using  $\omega(r)$  and  $\rho(z)$ , we propose the following pair  $\phi(z), \psi(z)$

$$\phi'(z) = -\omega^{-1}(z)\rho(z), \quad \psi'(z) = \rho(z), \quad (9)$$

where “ $'$ ” denotes derivative. Since  $\omega(r)$  and  $\rho(z)$  are arbitrary (provided they satisfy the strict increase and positivity constraint respectively), the class of pairs defined by (9) is very rich allowing for a multitude of choices. We show next that *any* such pair  $\{\phi(z), \psi(z)\}$  gives rise to a min-max problem, as in (8), that accepts  $r(X) = 1$  as its unique solution. We prove this claim in two steps. The first, involves a theorem where we consider a simplified version of the min-max problem.

**Theorem 1** *Let  $\omega(r), \phi(z), \psi(z)$  and  $\mathcal{I}_\omega$  be defined as above with the additional constraint  $\psi(\omega(1)) = 0$ . Fix  $r \geq 0$  and consider  $\phi(D) + r\psi(D)$  as a function of the scalar  $D$ . Then, for any  $D \in \mathcal{I}_\omega$ , we have that*

$$\phi(D) + r\psi(D) \leq \phi(\omega(r)) + r\psi(\omega(r)), \quad (10)$$

*with equality if and only if  $D = \omega(r)$ .*

*Consider next the minimization with respect to  $r$  of the maximal value in (10). It is true that*

$$\min_{r \geq 0} \{\phi(\omega(r)) + r\psi(\omega(r))\} = \phi(\omega(1)), \quad (11)$$

*with equality if and only if  $r = 1$ .*

**Proof** We note that the constraint  $\psi(\omega(1)) = 0$  does not affect the generality of our class of functions since from (9) we have that  $\psi(z)$ , after integration, is defined up to an arbitrary additive constant. We can always select this constant so that the constraint is satisfied. We would also like to emphasize that this constraint is needed only for the proof of this theorem and it is not necessary for the corresponding min-max problem defined in (8).

For fixed  $r$ , to find the maximum of  $\phi(D) + r\psi(D)$  we consider the derivative with respect to  $D$  which, using (9), takes the form

$$\phi'(D) + r\psi'(D) = (r - \omega^{-1}(D))\rho(D).$$

The strict increase of  $\omega(r)$  is inherited by its inverse function  $\omega^{-1}(z)$  which, combined with the positivity of  $\rho(z)$ , implies that the previous expression has the same sign as  $r - \omega^{-1}(D)$  or  $\omega(r) - D$ . Consequently  $D = \omega(r)$  is the

only critical point of  $\phi(D) + r\psi(D)$  which is a global maximum. Of course there are possibilities for extrema at the two end points of  $\mathcal{I}_\omega$  but they can only be (local) minima.

Let us now focus on the resulting function  $\phi(\omega(r)) + r\psi(\omega(r))$ . Taking its derivative with respect to  $r$  yields

$$\begin{aligned} \frac{d}{dr} \{ \phi(\omega(r)) + r\psi(\omega(r)) \} = \\ \{ \phi'(\omega(r)) + r\psi'(\omega(r)) \} \omega'(r) + \psi(\omega(r)) = \psi(\omega(r)), \end{aligned}$$

where the last equality is due to the specific definition of the two functions  $\phi(z), \psi(z)$  in (9). Since  $\psi'(z) = \rho(z) > 0$ , this implies that  $\psi(z)$  is strictly increasing, being also the integral of  $\rho(z)$  it is continuous in  $z$ . If we combine this property with the strict increase and continuity (as a result of left and right differentiability) of  $\omega(r)$  we conclude that  $\psi(\omega(r))$  is also strictly increasing and continuous in  $r$ . We recall that  $\psi(z)$  is selected to satisfy  $\psi(\omega(1)) = 0$ , consequently for  $r = 1$  the function  $\phi(\omega(r)) + r\psi(\omega(r))$  has a unique minimum which is global and no other critical points. Of course it can still exhibit extrema at  $r = 0$  and/or  $r \rightarrow \infty$  but they can only be (local) maxima. ■

A consequence of Theorem 1 is the next corollary, which constitutes the second and final step in proving that the adversarial problem defined in (8) has as unique solution the function  $r(X) = 1$ .

**Corollary 1** *If the functions  $\phi(z), \psi(z)$  satisfy (9) and  $\omega(r)$  is strictly increasing and left and right differentiable, then in the adversarial problem defined in (8) the maximizer is  $D(X) = \omega(r(X))$  and the minimizer is  $r(X) = 1$ , while the resulting min-max value is equal to*

$$\begin{aligned} \min_{r(X) \in \mathcal{R}_f} \max_{D(X)} E_f [\phi(D(X)) + r(X)\psi(D(X))] = \\ \phi(\omega(1)) + \psi(\omega(1)). \end{aligned} \quad (12)$$

**Proof** The proof is simple. First we observe that

$$\begin{aligned} E_f [\phi(D(X)) + r(X)\psi(D(X))] = \\ E_f [\phi(D(X)) + r(X)\tilde{\psi}(D(X))] + \psi(\omega(1)) \end{aligned} \quad (13)$$

with the last equality being true since  $E_f[r(X)] = 1$  and where  $\tilde{\psi}(z) = \psi(z) - \psi(\omega(1))$ . We start with the maximization problem. Since  $D(X)$  is a function of  $X$  we have

$$\begin{aligned} \max_{D(X)} E_f [\phi(D(X)) + r(X)\tilde{\psi}(D(X))] = \\ E_f \left[ \max_{D(X)} \{ \phi(D(X)) + r(X)\tilde{\psi}(D(X)) \} \right]. \end{aligned} \quad (14)$$

The maximization under the expectation can be performed for each fixed  $X$ . However, when we fix  $X$  then  $r(X)$  becomes a constant and the result of the maximization depends only on the actual value of  $r(X)$ . This suggests that we can limit ourselves to functions of the form  $D(X) =$

$D(r(X))$ . After this observation we can drop the dependence on  $X$  and perform, equivalently, the maximization

$$\max_D \{ \phi(D(r)) + r\tilde{\psi}(D(r)) \}$$

for each fixed  $r$ . The pair  $\{\phi(z), \tilde{\psi}(z)\}$  satisfies the assumptions of Theorem 1, therefore maximization is achieved for  $D(r) = \omega(r)$ . This implies that

$$\begin{aligned} \max_{D(X)} E_f [\phi(D(X)) + r(X)\psi(D(X))] = \\ E_f [\phi(\omega(r(X))) + r(X)\tilde{\psi}(\omega(r(X)))] + \psi(\omega(1)). \end{aligned}$$

We can now continue in a similar way for the minimization problem. Specifically

$$\begin{aligned} \min_{r(X) \in \mathcal{R}_f} \max_{D(X)} E_f [\phi(D(X)) + r(X)\tilde{\psi}(D(X))] = \\ \min_{r(X) \in \mathcal{R}_f} E_f [\phi(\omega(r(X))) + r(X)\tilde{\psi}(\omega(r(X)))] \geq \\ E_f \left[ \min_{r(X) \in \mathcal{R}_f} \{ \phi(\omega(r(X))) + r(X)\tilde{\psi}(\omega(r(X))) \} \right] \geq \\ E_f \left[ \min_r \{ \phi(\omega(r)) + r\tilde{\psi}(\omega(r)) \} \right] = \phi(\omega(1)), \end{aligned}$$

with the last inequality being true since the minimization that follows is unconstrained and the last equality being a consequence of Theorem 1. The final lower bound is clearly attained by  $r(X) = 1$ , which is also a legitimate solution of the constrained minimization, since  $r(X) = 1$  belongs to the class  $\mathcal{R}_f$  of likelihood ratios. Consequently  $r(X) = 1$  is the solution to the min-max problem. Returning to the original min-max setup with  $\psi(z)$  replacing  $\tilde{\psi}(z)$ , we can clearly see that it satisfies (12). This completes the proof. ■

**Remark 1** *The adversarial problem is defined with the help of the two functions  $\phi(z), \psi(z)$  which, according to (9), can be obtained by integrating the corresponding derivatives. However, this integration might not always be possible, analytically. As we will have the chance to confirm in Section 4, in an actual optimization algorithm (e.g. of gradient type) that solves (2), the exact form of  $\phi(z), \psi(z)$  is not necessary. Instead, what is required is their derivatives which are analytically available from (9).*

We must emphasize that there already exists the significant work by (Nowozin et al., 2016) that addresses a similar problem as our current work, namely the definition of a class of min-max optimizations that can be used to design the generator/discriminator pair. The class in (Nowozin et al., 2016) is defined in terms of a convex function  $f(r)$  which can be shown to correspond to the outcome of our maximization, namely the function  $\phi(\omega(r)) + r\psi(\omega(r))$ . This establishes a one-to-one correspondence between the two methods under the ideal (non data-driven) setup. However, we believe that, our approach enjoys certain significant advantages:

First, the definition of the two functions  $\phi(z), \psi(z)$  in Equ. (9) is straightforward while in (Nowozin et al., 2016) requires the solution of an optimization problem.

Second, in our case we have complete control over the result of the maximization problem that defines the discriminator. In other words we can decide what transformation  $\omega(r)$  of the likelihood ratio  $r$ , the discriminator must estimate. In (Nowozin et al., 2016) such flexibility does not exist.

Controlling the function we estimate with the discriminator plays a significant role in the implementation of our method. Indeed when we use a neural network to approximate the optimum discriminator, this affects the overall quality of the resulting generator/discriminator pair. We should also note that there are important applications in Statistics where one is interested in estimating only the transformation of the likelihood ratio, with the most common cases being the likelihood ratio itself, its logarithm (log-likelihood ratio), or the ratio  $\frac{r}{1+r}$  which plays the role of the posterior probability between two densities. In other words, there are applications where one is interested only in the “max” part of the min-max problem. In fact, in the next section we give examples of various choices of  $\omega(r)$  and mention problems where the discriminator function becomes the actual target and not the generator.

### 3. Examples

Let us now present characteristic cases for the  $\omega(r)$  function and give pairs  $\{\phi(z), \psi(z)\}$  that satisfy (9). As we proved, this implies that the corresponding adversarial problem in (8) accepts the desired solution  $r(X) = 1$ .

#### 3.1. Case $\omega(r) = r^\alpha$

For  $\omega(r) = r^\alpha, \alpha > 0$  we have that  $\omega^{-1}(z) = z^{\frac{1}{\alpha}}$  and  $\mathcal{I}_\omega = [0, \infty)$ . According to (9), for  $z \in [0, \infty)$  we must define

$$\phi'(z) = -z^{\frac{1}{\alpha}} \rho(z), \quad \psi'(z) = \rho(z). \quad (15)$$

The following examples can be shown to satisfy (15).

**A1)** If we select  $\rho(z) = z^\beta$ , with  $\beta \neq -1, -1 - \frac{1}{\alpha}$ , this yields  $\phi(z) = -\frac{z^{1+\frac{1}{\alpha}+\beta}}{1+\frac{1}{\alpha}+\beta}$  and  $\psi(z) = \frac{z^{1+\beta}}{1+\beta}$ . For  $\beta = -1$ ,  $\rho(z) = z^{-1}$ ,  $\phi(z) = -\alpha z^{\frac{1}{\alpha}}$ ,  $\psi(z) = \log z$ . For  $\beta = -1 - \frac{1}{\alpha}$ ,  $\rho(z) = z^{-1-\frac{1}{\alpha}}$ ,  $\phi(z) = -\log z$ ,  $\psi(z) = -\alpha z^{-\frac{1}{\alpha}}$ .

**A2)** If we select  $\alpha = 1$ ,  $\rho(z) = \frac{1}{(1+z)z}$  then,  $\phi(z) = -\log(1+z)$  and  $\psi(z) = -\log(1+z^{-1})$ .

**A3)** If we select  $\alpha = 1$ ,  $\rho(z) = \frac{\tan^{-1}(z)}{z}$ , this yields  $\psi(z) = \int^z \rho(x)dx$  and  $\phi(z) = -z \tan^{-1}(z) + \frac{1}{2} \log(z^2 + 1)$ . This example corresponds to functions  $\phi(z), \psi(z)$  that are not both available in closed form. However they can still be used to define an optimization problem whose solu-

tion is numerically computable.

For the particular selection  $\omega(r) = r$  (corresponding to  $\alpha = 1$ ) we can show that the resulting cost is equivalent to the Bregman cost (Bregman, 1967). In fact there is a one-to-one correspondence between our  $\rho(z)$  function and the function that defines the Bregman cost. This correspondence however is lost once we switch to a different  $\alpha$  or a different  $\omega(r)$  function, suggesting that the proposed class of pairs  $\{\phi(z), \psi(z)\}$ , is far richer than the class induced by the Bregman cost.

We should mention that in A1) the selection  $\alpha = 1, \beta = 0$  is known as the *mean square error* criterion and if we apply *only the maximization problem* then this corresponds to a likelihood ratio estimation technique proposed in the literature (Sugiyama et al., 2010; 2013). Under the adversarial approach the cost takes the following interesting form

$$\begin{aligned} J(r, D) &= \mathbb{E}_f[-0.5D^2(X) + r(X)D(X)] \\ &= \frac{1}{2} \mathbb{E}_f[-(D(X) - r(X))^2 + (r(X) - 1)^2] + \frac{1}{2} \end{aligned}$$

where the equality is a consequence of  $r(X)$  being a likelihood ratio with respect to  $f(X)$ . As we can see, the maximization problem indeed yields  $D(X) = r(X)$  while the minimization that must follow, captures the desired solution  $r(X) = 1$ .

#### 3.2. Case $\omega(r) = \alpha^{-1} \log r$

For  $\omega(r) = \alpha^{-1} \log r, \alpha > 0$  we have  $\omega^{-1}(z) = e^{\alpha z}$  and  $\mathcal{I}_\omega = \mathbb{R}$ . As before  $\rho(z)$  must be strictly positive and, according to (9), for all real  $z$  we must define

$$\phi'(z) = -e^{\alpha z} \rho(z), \quad \psi'(z) = \rho(z). \quad (16)$$

We have the following examples that satisfy these equations.

**B1)** If  $\rho(z) = e^{-\beta z}$  with  $\beta \neq 0, \alpha$ , this produces  $\phi(z) = -\frac{e^{(\alpha-\beta)z}}{\alpha-\beta}$ ,  $\psi(z) = -\frac{e^{-\beta z}}{\beta}$ . If  $\beta = 0$  then  $\rho(z) = 1$ ,  $\phi(z) = -\frac{e^{\alpha z}}{\alpha}$ ,  $\psi(z) = z$ . If  $\beta = \alpha$  then  $\rho(z) = e^{-\alpha z}$ ,  $\phi(z) = -z$  and  $\psi(z) = -\frac{e^{-\alpha z}}{\alpha}$ .

**B2)** If  $\alpha = 1$ ,  $\rho(z) = \frac{1}{1+e^z}$  then,  $\phi(z) = -\log(1+e^z)$  and  $\psi(z) = -\log(1+e^{-z})$ .

**B3)** If  $\alpha = 1$ ,  $\rho(z) = \frac{1-e^{-z}}{z}$  this yields  $\phi(z) = \int^z \frac{1-e^x}{x} dx$  and  $\psi(z) = \int^z \frac{1-e^{-x}}{x} dx$ . The two functions  $\phi(z), \psi(z)$  can be written in terms of the Exponential integral or with the help of a power series expansion, but they do not enjoy any closed form expressions. On the other hand, their derivatives are simple and can be clearly used in a gradient type algorithm to numerically compute the solution of the corresponding optimization.

We would like to point out that the previous examples are presented for the first time and can be used either un-



der a min-max setting for the determination of the generator/discriminator pair or under a purely maximization setting for the direct estimation of the log-likelihood ratio function  $\log r(X)$ .

### 3.3. Case $\omega(r) = \frac{r}{r+1}$

When  $\omega(r) = \frac{r}{r+1}$  we have  $\omega^{-1}(z) = \frac{z}{1-z}$  and  $\mathcal{J}_\omega = [0, 1]$ . For  $\rho(z) > 0, z \in [0, 1]$  we must define the functions  $\phi(z), \psi(z)$  according to (9)

$$\phi'(z) = -\frac{z}{1-z}\rho(z), \quad \psi'(z) = \rho(z). \quad (17)$$

The next set of examples can be seen to satisfy (17).

**C1)** If we select  $\rho(z) = \frac{1}{z}$ , this yields  $\phi(z) = \log(1-z)$  and  $\psi(z) = \log z$ .

**C2)** Selecting  $\rho(z) = (1-z)^\alpha$ , with  $\alpha \neq 0, -1$ , yields  $\phi(z) = -\frac{1}{1+\alpha}(1-z)^{\alpha+1} + \frac{1}{\alpha}(1-z)^\alpha$  and  $\psi(z) = -\frac{1}{1+\alpha}(1-z)^{\alpha+1}$ . For  $\alpha = 0$ , we have  $\rho(z) = 1$  and  $\phi(z) = z + \log(1-z)$ ,  $\psi(z) = z$ , while for  $\alpha = -1$  we have  $\rho(z) = \frac{1}{1-z}$  and  $\phi(z) = -\log(1-z) - \frac{1}{1-z}$ ,  $\psi(z) = -\log(1-z)$ .

In C1) we recognize the functions used in the original article by (Goodfellow et al., 2016). C2) appears for the first time.

### 3.4. Case $\omega(r) = \text{sign}(\log r)$

This is a special case of  $\omega(r)$  with the corresponding function not being strictly increasing. It turns out that we can still come up with optimization problems, two of which are known and used in practice, by considering  $\omega(r)$  as a *limit* of a sequence of strictly increasing functions.

**Monotone Loss:** As a first approximation we propose  $\text{sign}(z) \approx \tanh(\frac{c}{2}z)$  where  $c > 0$  a parameter. We note that  $\lim_{c \rightarrow \infty} \tanh(\frac{c}{2}z) = \text{sign}(z)$ . Using this approximation we can write

$$\text{sign}(\log r) \approx \tanh\left(\frac{c}{2} \log r\right) = \frac{r^c - 1}{r^c + 1} = \omega(r). \quad (18)$$

As we mentioned, we have exact equality for  $c \rightarrow \infty$ . Let us perform our analysis by assuming that  $c$  is finite. We note that  $\omega^{-1}(z) = (\frac{1+z}{1-z})^{\frac{1}{c}}$  and  $\mathcal{J}_\omega = [-1, 1]$ . Consequently, if  $\rho(z) > 0$  for  $z \in [-1, 1]$ , we must define

$$\phi'(z) = -\left(\frac{1+z}{1-z}\right)^{\frac{1}{c}} \rho(z), \quad \psi'(z) = \rho(z). \quad (19)$$

**D1)** In (19) if we let  $c \rightarrow \infty$  in order to converge to the desired sign function, this yields  $\phi'(z) = -\rho(z)$  and  $\psi'(z) = \rho(z)$ . This suggests that  $\phi(z) = -\int^z \rho(x)dx$  is decreasing and  $\psi(z) = \int^z \rho(x)dx = -\phi(z)$  is increasing. In fact any strictly increasing function  $\psi(z)$  can be adopted provided we select  $\phi(z) = -\psi(z)$ .

There is a popular combination that falls under Case D1).

In particular, the selection  $\psi(z) = z = -\phi(z)$  known as Wasserstein GAN is proposed in (Arjovsky et al., 2017). We recall that in this case  $z \in [-1, 1]$ .

**Hinge Loss:** As a second approximation we use the expression  $\text{sign}(z) \approx \text{sign}(z)|z|^{\frac{1}{c}}$ ,  $c > 0$ , which is strictly increasing, continuous and converges to  $\text{sgn}(z)$  as  $c \rightarrow \infty$ . This suggests that

$$\text{sign}(\log r) \approx \text{sign}(\log r)|\log r|^{\frac{1}{c}} = \omega(r), \quad (20)$$

and  $\omega^{-1}(z) = e^{z^c}$ . Since  $\omega(r)$  can assume any real value we conclude that  $\mathcal{J}_\omega = \mathbb{R}$  which, clearly, differs from the previous approximation where we had  $\mathcal{J}_\omega = [-1, 1]$ . If  $\rho(z) > 0, z \in \mathbb{R}$  then, according to (9) we must define

$$\phi'(z) = -e^{z^c} \rho(z), \quad \psi'(z) = \rho(z). \quad (21)$$

We present the following case that leads to a very well known pair from a completely different application.

**D2)** Following (21), if we select  $\psi'(z) = \rho(z) = \{e^{-|z|^{\frac{1}{c}}} + \mathbb{1}_{\{z < -1\}}\} > 0$  then  $\phi'(z) = -e^{z^{\frac{1}{c}}} \{e^{-|z|^{\frac{1}{c}}} + \mathbb{1}_{\{z < -1\}}\}$ . If we now let  $c \rightarrow \infty$ , we obtain the limiting form for the derivatives which become  $\psi'(z) = -\mathbb{1}_{\{z < 1\}}$  and  $\phi'(z) = \mathbb{1}_{\{z > -1\}}$ . By integrating we arrive at  $\phi(z) = -\max\{1+z, 0\}$  and  $\psi(z) = -\max\{1-z, 0\}$ . The cost based on this particular pair is called the *hinge loss* (Tang, 2013) and it is very popular in binary classification where one is interested only in the maximization problem. The corresponding method is known to exhibit an overall performance which in practice is considered among the best (Rosasco et al., 2004; Janocha & Czarnecki, 2017). Here, as in (Zhao et al., 2017), we propose the hinge loss as a means to perform adversarial optimization for the design of the generator  $G(Z)$ .

This completes our presentation of examples. However, we must emphasize, that these are only a few illustrations of possible pairs  $\{\phi(z), \psi(z)\}$  one can construct. Indeed combining, as dictated by (9), any strictly increasing function  $\omega(r)$  with any positive function  $\rho(z)$  generates a legitimate pair  $\{\phi(z), \psi(z)\}$  and a corresponding min-max problem (8) that enjoys the desired solution  $r(X) = 1$ .

## 4. Data-Driven Setup and Neural Networks

Let us now consider the data-driven version of the problem. As mentioned, the target density  $f(X)$  is unknown. Instead we are given a collection of realizations  $\{X_i\}$  that follow  $f(X)$  and a second collection  $\{Z_j\}$  that follows the origin density  $h(Z)$ . These data constitute our *training set*. Regarding the second set  $\{Z_j\}$  it can either become available “on the fly” when  $h(Z)$  is known by generating realizations every time they are needed, or it can be considered fixed from the start exactly as  $\{X_i\}$ , if  $h(Z)$  is also unknown.

As we pointed out in Section 1, we are interested in de-

signing a generator  $G(Z)$  so that when we apply it onto the data  $Z_j$ , that is,  $Y_j = G(Z_j)$  the resulting  $Y_j$  will follow a density that matches the target density  $f(X)$ .

Since we are now considering the data-driven version of the problem, we are going to limit  $G(Z), D(X)$  to be the outputs of corresponding neural networks. Therefore the generator is replaced by  $G(Z, \theta)$  while the discriminator by  $D(X, \vartheta)$  where  $\theta, \vartheta$  summarize the parameters of the two neural networks. Of course instead of neural networks one could use any other parametric family, as SVMs, capable of efficiently approximating any nonlinear function.

Once we have selected our favorite  $\omega(r)$  and  $\rho(z)$  functions we can compute from (9) the functions  $\phi(z), \psi(z)$  that enter into the min-max problem defined in (2). This problem, after limiting the generator and discriminator to neural networks, can be rewritten as follows

$$\min_{\theta} \max_{\vartheta} \mathcal{J}(\theta, \vartheta) = \min_{\theta} \max_{\vartheta} \{E_f[\phi(D(X, \vartheta))] + E_h[\psi(D(G(Z, \theta), \vartheta))]\}. \quad (22)$$

If  $\theta_o, \vartheta_o$  are the corresponding optimum parameter values, and the structure of the two networks is sufficiently rich, we expect that  $G(Z, \theta_o), D(X, \vartheta_o)$  will approximate the optimum functions  $D(X), G(Z)$  of the ideal problem in (2) respectively. In particular for  $\theta_o$ , the generator  $G(Z, \theta_o)$ , whenever applied onto any  $Z_j$  that follows  $h(Z)$ , it will result in a  $Y_j = G(Z_j, \theta_o)$  that follows a density which is expected to be close to the target density  $f(Y)$ .

A simple stochastic gradient algorithm that can solve the min-max optimization in (22) is the following

$$\vartheta_t = \vartheta_{t-1} + \mu \{ \phi'(D(X_t, \vartheta_{t-1})) \nabla_{\vartheta} D(X_t, \vartheta_{t-1}) + \psi'(D(Y_t, \vartheta_{t-1})) \nabla_{\vartheta} D(Y_t, \vartheta_{t-1}) \}, \quad (23)$$

corresponding to the maximization problem and

$$\theta_t = \theta_{t-1} - \mu \psi'(D(Y_t, \vartheta_{t-1})) (\mathcal{J}_{\theta} G(Z_t, \theta_{t-1}))^T \nabla_X D(Y_t, \vartheta_{t-1}), \quad (24)$$

for the minimization. Here  $\mathcal{J}_{\theta} G(Z, \theta)$  denotes the Jacobian of  $G(Z, \theta)$  with respect to  $\theta$ ,  $Y_t = G(Z_t, \theta_{t-1})$  and  $\mu$  is the learning rate of the two updates. With  $X_t$  we denote a training sample from the collection  $\{X_i\}$  while with  $Z_t$  either a sample from  $\{Z_t\}$  when  $h(Z)$  is unknown or a new realization following  $h(Z)$  if the latter is known. If the collection of training data is exhausted after applying the iterations several times, then we simply reuse them.

With (23), (24) we confirm Remark 1, namely that in an optimization algorithm we do not necessarily need the functions  $\phi(z), \psi(z)$  explicitly, but only their derivatives.

It has also been observed (Goodfellow et al., 2016; Arjovsky et al., 2017) that in order for the optimization algorithm to converge properly, for each iteration of the min-

imization problem we must perform *several* iterations of the maximization problem (common practice suggests at least *five* iterations of the maximization problem for each iteration of the minimization).

**Remark 2** When replacing  $D(X), G(Z)$  with neural networks we must take special care of the corresponding outputs. Basically, we must guarantee that they are of the correct form. This is particularly important in the case of the scalar output  $D(X, \vartheta)$  of the discriminator. We recall that the optimum discriminator is  $D(X) = \omega(r(X))$ . This implies that we need to assure that  $D(X, \vartheta)$  takes values in  $\mathcal{I}_{\omega}$  (the range of  $\omega(r)$ ). Consequently, we must apply the proper nonlinearity in the output of the discriminator that will guarantee this fact.

## 5. Experiments

We implemented most of the examples mentioned in Section 3 using the datasets MNIST, CelebA and CIFAR-10. Before presenting our results, we would like to give details about the following pairs  $\{\phi(z), \psi(z)\}$  that exhibited the best overall performance in our experiments:

**Exponential:** From Example B1),  $\alpha = 1, \beta = 0.5$ , yields  $\phi(z) = -e^{0.5z}$ ,  $\psi(z) = -e^{-0.5z}$ , while  $\mathcal{I}_{\omega} = \mathbb{R}$ . No nonlinearity is needed in the discriminator output.

**B1b:** From Example B1), for  $\alpha = \beta = 1$  we obtain  $\phi(z) = -z$  and  $\psi(z) = -e^{-z}$  with  $\mathcal{I}_{\omega} = \mathbb{R}$ . No nonlinearity is needed in the discriminator output.

**B2:** Example B2), with  $\phi(z) = -\log(1 + e^z)$  and  $\psi(z) = -\log(1 + e^{-z})$  and  $\mathcal{I}_{\omega} = \mathbb{R}$ . No nonlinearity is needed in the discriminator output.

**Cross entropy:** This is the classical method proposed in (Goodfellow et al., 2016) corresponding to Example C1) with  $\phi(z) = \log(1 - z)$ ,  $\psi(z) = \log z$  and  $\mathcal{I}_{\omega} = [0, 1]$ . To the discriminator output we apply the sigmoid function.

**Wasserstein:** We are in Example D1) with  $\phi(z) = z = -\psi(z)$  and  $\mathcal{I}_{\omega} = [-1, 1]$ . To limit the output of the discriminator we use the function  $\tanh(z)$ .

**Hinge:** From Example D2),  $\phi(z) = -\max\{1 + z, 0\}$ ,  $\psi(z) = -\max\{1 - z, 0\}$  and  $\mathcal{I}_{\omega} = \mathbb{R}$ . No nonlinearity is needed in the discriminator output.

For each dataset we present the best five methods in terms of convergence rate and quality of synthetic results produced by the generator.

We recall that GANs are notorious for their nonrobust behavior (Bengio, 2012; Creswell et al., 2018; Mescheder et al., 2017). For the stabilization of the training process, we used the gradient-penalty methodology described in (Gulrajani et al., 2017) which was generalized to a class of Lipschitz GANs in (Zhou et al., 2019).

For the generator, we used a four-layer neural network where the first layer is linear and the remaining deconvolutional; with ReLU activation functions between the layers except the final layer where we used a sigmoid function since the output is an image with pixel values in the range  $[0, 1]$ . The generator input is a standard i.i.d. normal vector with dimension 64 for MNIST and 128 for CelebA and CIFAR-10.

For the discriminator, we used a four-layer neural network with three convolutional layers followed by a linear layer. We applied Leaky ReLUs between the layers except for the final layer where we adopted proper functions based on the range  $\mathcal{J}_\omega$ . For the training of the two neural networks we applied the Adam algorithm (Kingma et al., 2015) with  $\beta_1 = 0.5$ ,  $\beta_2 = 0.9$ , learning rate  $10^{-4}$  and batch size 50 for MNIST and 64 for CelebA and CIFAR-10. For all datasets the training lasted 200000 iterations.

The first set of experiments involves training with MNIST. Table 1 summarizes the top five attained Frechet Inception

MNIST	FID	KID
B1b	2.049	$6.248 \cdot 10^{-4}$
B2	2.089	$6.133 \cdot 10^{-4}$
Cross entropy	2.105	$6.434 \cdot 10^{-4}$
Exponential	2.065	$6.084 \cdot 10^{-4}$
Wasserstein	2.078	$6.924 \cdot 10^{-4}$

Table 1. Best scores obtained by different objective functions during training with 200000 iterations using MNIST.

Distances (FID) (Heusel et al., 2017) and Kernel Inception Distances (KID) (Binkowski et al., 2018) by the various methods. In Figure 1 we present examples of generated synthetic numerals by the corresponding methods. We observe that for this particular dataset the designed GANs have comparable performance with the Wasserstein and Cross Entropy exhibiting the smallest and B1b and Exponential the highest scores.

The second set of experiments involves the CelebA dataset. Table 2 reports the best observed FID, KID scores of the

CelebA	FID	KID
B2	1.186	$1.166 \cdot 10^{-4}$
Cross entropy	1.199	$1.135 \cdot 10^{-4}$
Exponential	1.218	$1.139 \cdot 10^{-4}$
Hinge	1.227	$1.060 \cdot 10^{-4}$
Wasserstein	1.193	$1.062 \cdot 10^{-4}$

Table 2. Best scores obtained by different objective functions during training with 200000 iterations using CelebA.

competing methods, while Figures 2, 3 depict the evolution of the corresponding scores with the number of iterations during training. From these figures we observe that both scores of the Hinge method exhibit a high variability while B2, Cross entropy, Exponential and Wasserstein

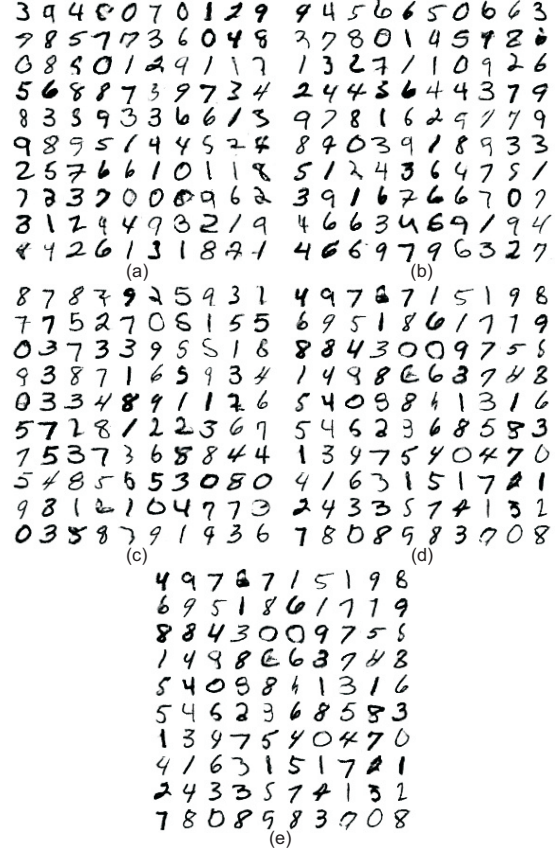


Figure 1. Numerals generated by (a) B1b, (b) B2, (c) Cross Entropy, (d) Exponential, (e) Wasserstein. Training with MNIST.

have comparable behavior. In Figure 4 we have examples of synthetic faces generated by each method.

The third and last set of experiments involves the far more challenging CIFAR-10 dataset. Table 3 summarizes the best FID, KID scores, while Figures 5, 6 capture their evolution during training; finally, Figure 7 hosts examples of corresponding synthetic images. Interestingly, from Figure 5, we distinguish for this dataset, two performance groups. We can see that B1b and Wasserstein have a visibly better performance than the second group which includes the Cross Entropy, Exponential and B2.

CIFAR-10	FID	KID
B1b	36.998	$1.309 \cdot 10^{-4}$
B2	40.281	$5.864 \cdot 10^{-4}$
Cross entropy	38.788	$4.617 \cdot 10^{-4}$
Exponential	38.041	$5.369 \cdot 10^{-4}$
Wasserstein	37.242	$1.619 \cdot 10^{-4}$

Table 3. Best scores obtained by different objective functions during training with 200000 iterations using CIFAR-10.

## Acknowledgement

This work was supported by the US National Science Foundation, Grant CIF 1513373, through Rutgers University.



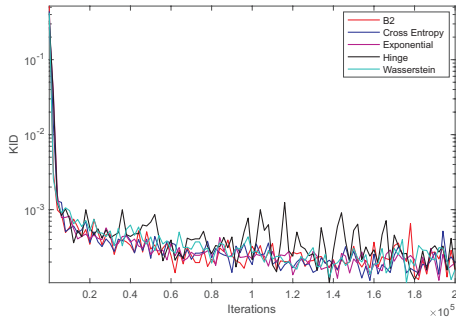


Figure 2. Evolution of the KID score during training with CelebA.

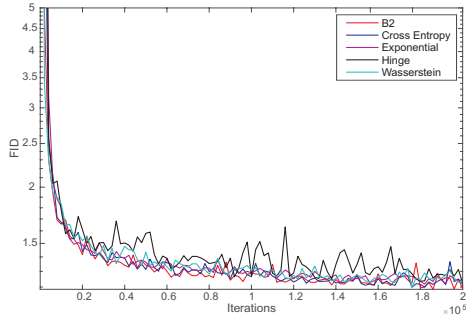


Figure 3. Evolution of the FID score during training with CelebA.

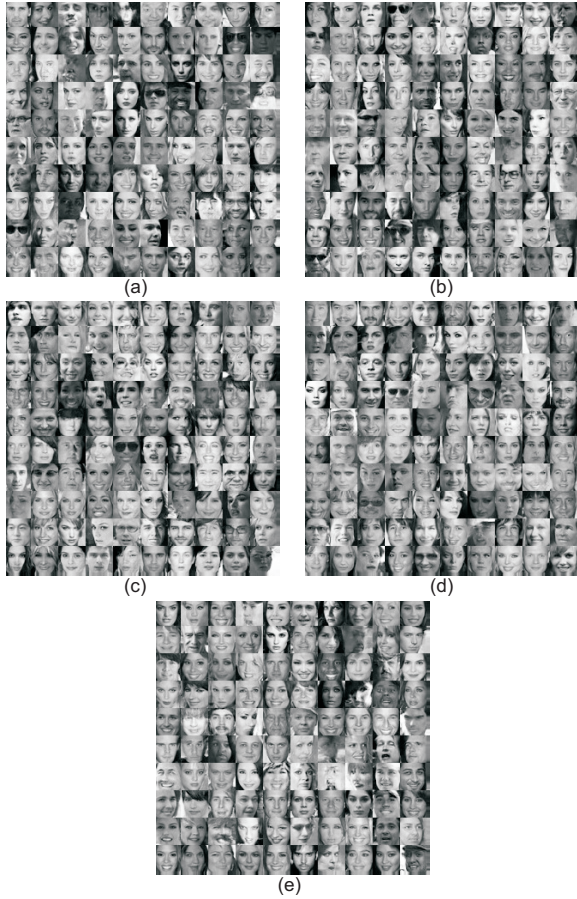


Figure 4. Synthetic images generated by (a) B2, (b) Cross entropy, (c) Exponential, (d) Hinge, (e) Wasserstein. Training with CelebA.

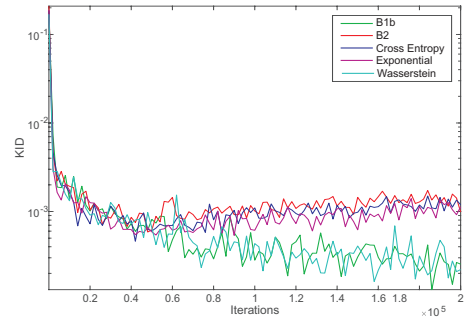


Figure 5. Evolution of KID score during training with CIFAR-10.

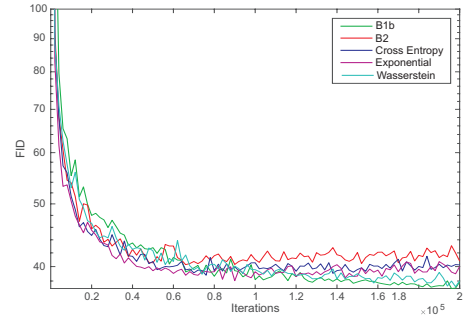


Figure 6. Evolution of FID score during training with CIFAR-10.

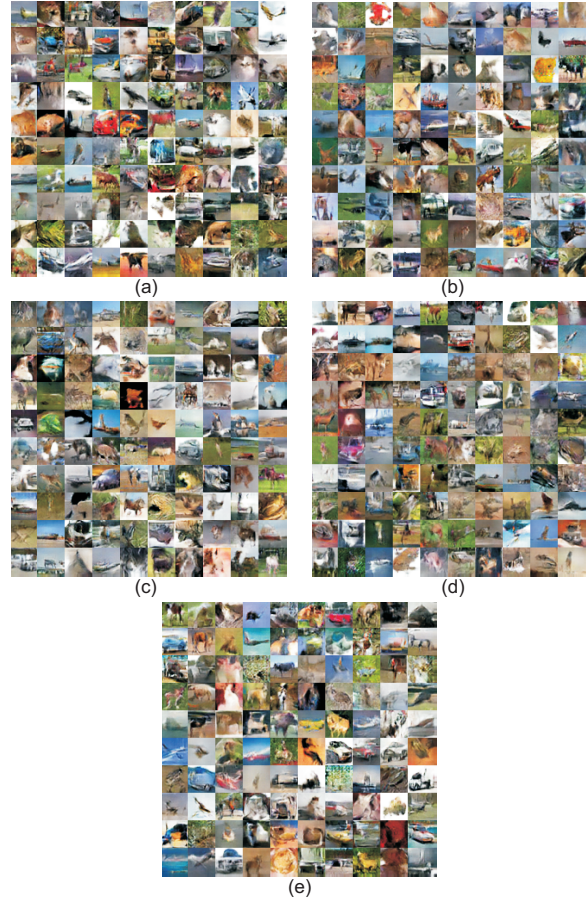


Figure 7. Synthetic images generated by (a) B1b, (b) B2, (c) Cross Entropy, (d) Exponential, (e) Wasserstein. Training with CIFAR-10.



## References

- Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein generative adversarial networks. *Proceedings of Machine Learning Research (PMLR 2017)*, pp. 214–223, 2017.
- Bengio, Y. Practical recommendations for gradient-based training of deep architectures. *arXiv:1206.5533*, 2012.
- Binkowski, M., Sutherland, D. J., Arbel, M., and Gretton, A. Demystifying MMD GANs. *Proceedings International Conference on Learning Representations*, 2018.
- Bregman, L. M. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR Computational Mathematics and Mathematical Physics*, 7:200–217, 1967.
- Box, G. E. P., and Cox, D. R. An analysis of transformations. *Journal of the Royal Statistical Society. Series B*, 26(2):211–252, 1964.
- Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., and Bharath, A. A. Generative adversarial networks: An overview *IEEE Signal Processing Magazine*, 35(1):53–65, January 2018.
- Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. *arXiv: 1406.2661*, 2014.
- Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. Improved training of Wasserstein GANs. *arXiv: 1704.00028*, 2017.
- Heusel, M., Ramsauer, H., Unterthiner, T., and Nessler, B. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. *Proceedings Advances Neural Information Processing Systems Conference*, 2017.
- Janocha, K., and Czarnecki, W. M. On loss functions for deep neural networks in classification. *arXiv:1702.05659*, 2017.
- Mescheder, L. M., Nowozin, S. and Geiger A. The numerics of GANs. *Proceedings Advances Neural Information Processing Systems Conference*, 2017.
- Moulin, P., and Veeravalli, V. V. *Statistical Inference for Engineers and Data Scientists*. Cambridge, New York, 2019.
- Kingma, D. P., and Ba J. L. Adam: A method for Stochastic Optimization. *Proceedings International Conference on Learning Representations*, 2015.
- Nowozin, S., Cseke, B., and Tomioka, R. f-GAN: Training generative neural samplers using variational divergence minimization. *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pp. 271–279, 2016.
- Rosasco, L., De Vito, E., Caponnetto, A., Piana, M., and Verri, A. Are loss functions all the same? *Journal Neural Computation*, 16(5):1063–1076, 2004.
- Sugiyama, M., Suzuki, T., and Kanamori, T. Density ratio estimation: A comprehensive review. *RIMS Kokyuroku*, 1703: 10–31, 2010.
- Sugiyama, M., Suzuki, T., and Kanamori, T. *Density Ratio Estimation in Machine Learning*. Cambridge, New York, 2013.
- Tang, Y. Deep learning using linear support vector machines. *arXiv:1306.0239*, 2013.
- Zhao, J., Mathieu, M., and LeCun Y. Energy-Based Generative Adversarial Networks. *Proceedings International Conference on Learning Representations*, 2015.
- Zhou, Z., Liang, J., Song, Y., Yu, L., Wang, H., Zhang, W., Yu, Y., and Zhang, Z. Lipschitz generative adversarial nets. *Proceedings of the 36th International Conference on Machine Learning*, pp. 7584–7593, 2019.