Mixture-GAN

Jaspal Singh
Department of Computer Science
Yale University
New Haven, CT 06511
jaspal.singh@yale.edu

Alexander Tong
Department of Computer Science
Yale University
New Haven, CT 06511
alexander.tong@yale.edu

1 Introduction

Generative Adversarial Networks (GANs) have recently been proposed as an alternative to traditional probabilistic modeling [5]. Rather than directly learning a probabilistic model over the distribution, it is often enough to sample from that distribution. GANs train two deterministic function $\mathcal{G}$ that maps a simple and sample-able distribution to the data distribution, and $\mathcal{D}$ that discriminates between samples from the training data distribution and that of the distribution defined by the generator.

In a wide variety of settings GANs have been shown to generate realistic samples from the data distribution [4]. However, in many applications we do not want to sample from the training distribution, but a biased (or less biased) version of the training distribution. For example, it may be desirable to create a more fair data distribution for a downstream classification task. In the case of biological data, there may be some modes which we would like to focus on or ignore in analysis. In the standard GAN framework it is not a simple task to alter the distribution over modes, especially given the large variety of undesired training behavior such as mode-collapse, divergence and oscillations.

In this paper, we take a step in analyzing the behavior of GANs in the setting where we have unbalanced training data but want to sample in a balanced way. More specifically, we model our training data as generated from a biased mixture of distributions. The task in then to learn these distributions and their associated mixture parameters, which allows subsequent data generation with an arbitrary mixture. Our general method is to examine the simplest model in this setting that exhibits the challenges of learning on unbalanced data, then showing in what circumstances we can prove it converges and in what settings it empirically does not. Specifically, as in Li et al. [8], we study a model consisting of two one-dimensional univariate gaussians, but in addition we include a mixture parameter α that controls the mixture between the two gaussians. In our model the parameter $\alpha *$ used in the data generation models the amount of bias inherent in the data. We hypothesis that the convergence behavior of the model should be parameterized by apriori assumptions on the limits of bias in the data.

Next we motivate our model from three applications. First is to the literature of fairness. When given biased data, one direction of research is towards preprocessing steps that alter the training data such that downstream classification tasks produce fair results. One way of tackling such a task is using GANs to generate from a more fair distribution such as in FairGAN by Xu et al. (2018) [13]. In this line of work, the goal is to use a GAN as a function to learn a new dataset that will satisfy whatever measure of fairness for a downstream classification task. This task has mainly been approached from an empirical perspective, while models have been tested and show promising results there is no theoretical justification as to why these models perform well or in what circumstances. Here we aim to justify such work using our simple model.

A second application is towards biological data. Often in biology the state of interest is not the common state but the exceptional one. Given a sample of cells the behavior of common types are much better characterized than those of the rare types. In this setting it would be useful to learn a generative model that is able to generate a more balanced distribution allowing the study of rarer populations. Study of rare populations has traditionally been done by creating a large training dataset and sub-setting to the relevant population of interest. It would be significantly more economical if we were able to generate realistic samples from the underrepresented population of interest based on what information is contained in the over-represented population.

A third line of work approaches from the representation learning community. For a long time there has been interest in learning useful, however that is defined, representations of the data. One line of enquiry that is particularly relevant is work in the unsupervised dis-entanglement of features [3]. While there is no consensus on the definition of dis-entanglement, a high level goal is to learn representations that align with human semantic description of the important factors of variation within the dataset. GANs provide a natural framework to examine this problem. The task of unsupervised learning of a dis-entangled set of features can be described as learning a generator that is conditioned on the category such that when we perturb this condition, it generates the human expected result. With our model we are able to examine the properties of learning a disentangled representation from biased data.

(We hope) The main contributions of this paper are:

1. The specification of a simple model capturing learning the factors from a biased data distribution.
2. A theoretical examination of the convergence properties of our model.
3. An empirical demonstration of the behavior of our model across the amount of bias present in the data.

2 Related Work

Existing methods for data generation fall into a few categories. There are those which attempt to estimate the density for the underlying data distribution using either a parametric or non-parametric density estimators [10, 12]. More recently, methods such as Variational Autoencoders [6] and GANs [5] have been demonstrated to perform well on high dimensional data without explicitly modeling the data distribution. Here we aim for a hybrid of the two methods where we explicitly model the mixture terms for the data distribution, but leave the other sources of variation to a sample-able network architecture. While our model does not fully model the data distribution we hope to capture the data generation performance exhibited by network based methods, but the flexibility of parametric models for control of our data distribution.

The Krishnaswamy Lab is interested in sampling from the data geometry rather than the data distribution particularly in the setting of high dimensional biological datasets [9, 1]. SUGAR first learns a graph based representation of the data based on similarity between datapoints, then uses a resampling technique to generate samples evenly along the manifold [9]. More recent work by Amodio and Krishnaswamy (submitted to ICML 2019), uses a GAN to generate samples based on a resampling of an autocoder embedding space [1]. This work has primarily been empirically justified, however we are still searching for a more theoretically principled model.

Our proposed Mixture GAN model is largely inspired by the model considered by Li et al. (2018) [8]. In this work the high level objective is to study the GAN dynamics for simpler data distributions in a principled and provable manner. In particular the authors consider only the limited yet interesting setting of designing GANs for learning a Gaussian mixture with 2 Gaussians. The authors consider the class of generators:

$$\mathcal{G} = \left\{ \frac{1}{2}\mathcal{N}(\mu_1, 1) + \frac{1}{2}\mathcal{N}(\mu_2, 1) \mid \mu_1, \mu_2 \in \mathbb{R} \right\} \tag{1}$$

The class of discriminators $\mathcal{D}$ and the loss function are same as those considered in our proposed model (Section 5). The min-max optimization problem for finding the optimal GANs is hence formulated as:

$$\mu^* = \operatorname*{argmin}_{\mu} \max_{D} \mathbb{E}_{x \sim P}[D(x)] + \mathbb{E}_{x \sim G_{\mu^*}}[1 - D(x)] \tag{2}$$

where P is the data distribution, $D \in \mathcal{D}$, the generator is parameterized by $\mu \in \mathbb{R}^2$ and μ^* is the parameter of the optimal GAN. The authors study the convergence on this toy model using the first order and optimal discriminator approach (defined in Section 5). The authors experimentally observe that first order methods do not always converge and suffer from well studied GAN problems including vanishing gradient and mode collapse. The authors prove the convergence of the optimal discriminator approach under reasonable assumptions, stated in Theorem 1. To the best of our knowledge, this is the first work which aims at analyzing the GAN dynamics and proving their convergence for the simple yet non-trivial set of data distributions. However, the model considered in this work is too simple to be used for any known GAN applications.

Some of the recent works have focused on studying the notion of fairness in GANs [11, 13]. Given a data distribution of the form $D = \{X, Y, S\}$, where unprotected attributes $X \in \mathbb{R}^n$, the decision $Y \in \{0, 1\}$ and the protected attribute $S \in \{0, 1\}$. Fairness GANs are designed to learn the distribution D subject to some fairness constraint. To given an example, we discuss the FairGAN architecture and optimization problem proposed in Xu et al. (2018) [13]. The aim in this work is to propose a GAN which has an output distribution similar to the data distribution D while ensuring the statistical parity fairness condition is met, which is defined as:

$$Pr(y = 1|s = 1) = Pr(y = 1|s = 0) \tag{3}$$

i.e. the probability of the the predicted variable is independent of the protected attribute. The proposed GAN architecture is shown in Figure 1. As shown in the Figure, the generator G takes as input the noise z and the protected attribute s, and it outputs $(x, y) \in X \times Y$. The discriminator D_1 is the standard discriminator considered in any GAN, where its function is to differentiate between real and fake samples. It is given as input the protected attribute s along with an element of either the real distribution $P(x, y|s)$ or the output of the GAN $P_G(x, y|s)$. The second discriminator D_2 aims at predicting the protected attribute s given x, y from either the real data distribution or the GAN distribution. The optimization problem that must be solved by FairGAN can be stated as:

$$\min_{G} \max_{D_1, D_2} V(G, D_1, D_2) = V_1(G, D_1) + \lambda V_2(G, D_2) \tag{4}$$

where λ is a hyper parameter, and V_1, V_2 are defined as:

$$\begin{aligned} V_1(G, D_1) =& \mathbb{E}_{s \sim P(s), (x,y) \sim P(x,y|s)}[\log D_1(x, y, s)] + \mathbb{E}_{s \sim P(s), (x,y) \sim P_G(x,y|s)}[\log(1 - D_1(x, y, s))] \\ V_2(G, D_2) =& \mathbb{E}_{(x,y) \sim P(x,y|s=1)}[\log D_2(x, y)] + \mathbb{E}_{(x,y) \sim P_G(x,y|s=0)}[\log(1 - D_2(x, y))] \end{aligned}$$

The authors prove that solutions of the above optimization problem over all G, D_1, D_2 is achieved when $JSD(P(x, y, s)||P_G(x, y, s)) + JSD((P_G(x, y|s = 1)||P_G(x, y|s = 0)))$ is minimum, where JSD represents the Jensen-Shannon divergence between the two distributions. To the best of our knowledge, this is the first work which aims to study the notion of fairness in the context of GANs. A key drawback of this model is that it does not always ensure that the statistical parity fairness is met. Another drawback of this approach, and other GAN constructions in general, is that they do not give any convergence proofs for parameterized generator and discriminator algorithms like neural nets, which for GANs are the most commonly used primitives in practice.

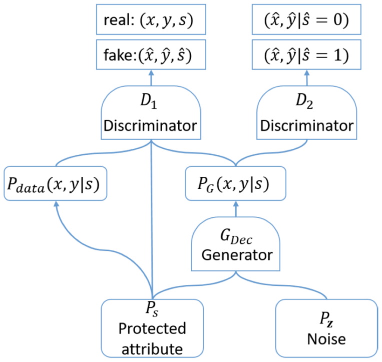

Figure 1: FairGAN architecture diagram copied from FairGAN paper [13]

3 Preliminaries

3.1 Generative Adversarial Networks (GANs)

GANs are generally described as a 2 player game, between a generator and a discriminator. Given a set of parameterized generators $\mathcal{G} = \{G_u, u \in \mathcal{U}\}$, a set of parameterized discriminators $\mathcal{D} = \{D_v, v \in \mathcal{V}\}$, a monotone function $m : \mathbb{R} \to \mathbb{R}$ and a data distribution P. The objective is to find a generator G_u given by:

$$\underset{u \in \mathcal{U}}{\operatorname{argmin}} \max_{v \in \mathcal{V}} \mathbb{E}_{x \sim P}[m(D_v(x))] + \mathbb{E}_{x \sim G_v}[m(1 - D_v(x))] \tag{5}$$

The aim of the discriminator is to distinguish between the samples of the real distribution P and the distribution of the generator G_v. On the other hand, the generator's aim is to fool the discriminator by producing samples that are indistinguishable to those drawn from the real data distribution.

The optimal generator parameters u^* is a solution to the Equation 5 is also represented be $G_{u^*} \sim P$, depicting that the two distributions are similar.

4 The proposed model - Mixture GAN

In this section we will introduce the notion of the mixture GANs, which are an extension to the generic GANs with some addition requirement.

Consider the GAN definition given in the Subsection 3.1, with the modification that the data distribution is of the form $P = \alpha P_0 + (1 - \alpha)P_1$, for some $\alpha \in (0, 1)$ and for some distributions P_0 and P_1. The objective of a mixture GAN is to find G_u such that the following two conditions hold:

1. $G_u \sim P$
2. There exist an *efficiently* computable function $\mathcal{A} : \mathcal{U} \to \mathcal{U} \times \mathcal{U}$ such that if $\mathcal{A}(u) = (u_0, u_1)$ and $G_u \sim P$, then $G_{u_0} \sim P_0$ and $G_{u_1} \sim P_1$.

While the first objective of a mixture GAN is the same as the standard GAN, the second condition ensures that given a generator parameterized by u that learns the joint data distribution P, one can easily derive the parameters u_1, u_2 of generators that can sample from the individual components P_0 and P_1 of the data distribution.

Given the generators parameterized by u_1 and u_2, one can easily design a new generator with distribution $\beta G_{u_1} + (1 - \beta)G_{u_2}$ for any $\beta \in [0, 1]$, solving the high level objective described in the introduction. For example, for $\beta = 0.5$ one can generate samples uniformly from both the underlying unknown distributions P_0 and P_1.

5 A Mixture GAN for weighted Gaussian mixture model

It is not very obvious as to how to design mixture GANs as defined in the previous Section. An even greater challenge is to be able to study the GAN dynamics. Hence, in the current project our aim will be to study the design and analysis of mixture GANs for the simpler model where the data distribution P is a mixture of two univariate Gaussians with variance one and non-uniform mixture weights. This toy model is majorly inspired by Li et al. (2018) [8], where the authors study the GAN dynamics for the case where the data distribution is a mixture of two univariate Gaussians with variance one and uniform weights.

Generators Given the set of possible distributions of the real data, we consider the generator set $\mathcal{G}$ defined as:

$$\mathcal{G} = \Big\{ \alpha \mathcal{N}(\mu_1, 1) + (1 - \alpha)\mathcal{N}(\mu_2, 1) \mid \mu_1, \mu_2 \in \mathbb{R} \text{ and } \alpha \in [0, 1] \Big\} \tag{6}$$

Let $G_{\mu,\alpha}(x)$ denote the distribution in $\mathcal{G}$ with means μ_1 and μ_2 and weight parameter α.

Loss function While there are many loss functions that one can chose from, in this project we will mainly focus on the total variation distance defined as: For distribution P, Q,

$$d_{TV}(P, Q) = \max_{A} P(A) - Q(A) \tag{7}$$

where the maximum is taken over all measurable sets.

This exact loss function was also considered in [8] since it simplifies for the Gaussian mixture model to the following:

$$d_{TV}(G_{\mu,\alpha}, G_{\mu^*,\alpha^*}) = \max_{E=I_1 \cup I_2} G_{\mu,\alpha}(E) - G_{\mu^*,\alpha^*}(E) \tag{8}$$

where the maximum is taken over the union of two disjoint intervals $I_1, I_2 \subset \mathbb{R}$. While this simplification is not explicitly described in [8] it trivially follows from their Theorem A.2.

Discriminators The more simplified formulation of the loss function given in Equation 8 allows for the following very natural definition of the discriminator:

$$\mathbb{D} = \{\mathbb{I}_{l_1,r_1} + \mathbb{I}_{l_2,r_2} \mid l_1, r_1, l_2, r_2 \in \mathbb{R} \text{ and } l_1 \leq r_1 \leq l_2 \leq r_2\} \tag{9}$$

where $\mathbb{I}_{l,r}$ is the indicator function that outputs 1 on input in range $[l, r]$ and outputs 0 otherwise. A discriminator from this set with parameters l_1, r_1, l_2, r_2 is represented by $D_{l,r}$, where $l = (l_1, l_2)$ and $r = (r_1, r_2)$.

Problem statement Consider the data distribution P equal to $\alpha^*\mathcal{N}(\mu_1^*, 1) + (1 - \alpha^*)\mathcal{N}(\mu_2^*, 1)$, for some $\alpha^* \in [0, 1]$ and $\mu_1^*, \mu_2^* \in \mathbb{R}$. The objective is to compute a mixture GAN that satisfies the two requirements defined in Section 4:

1. Design an efficient algorithm to solve for the generator G_{μ^*,α^*} from the following min-max optimization problem:

$$\begin{aligned} \mu^*, \alpha^* &= \operatorname*{argmin}_{\mu,\alpha} \max_{l,r} L(\mu, \alpha, l, r) \\ &= \operatorname*{argmin}_{\mu,\alpha} \max_{l,r} \mathbb{E}_{x\sim P}[D_{l,r}(x)] + \mathbb{E}_{x\sim G_{\mu^*,\alpha^*}}[1 - D_{l,r}(x)] \end{aligned} \tag{10}$$

2. There is a natural formulation of the algorithm $\mathcal{A}$ for this model in order to ensure the second requirement is met:

$$\mathcal{A}(\mu, \alpha) = ((\mu, 1), (\mu, 0)) \tag{11}$$

   Note that if $G_{\mu,\alpha}$ correctly learns the distribution P, then the generators $G_{\mu,1}$ and $G_{\mu,0}$ correctly learn the distribution P_0 and P_1. In other words if $G_{\mu,\alpha} \sim P$, then $G_{\mu,1} \sim P_0$ and $G_{\mu,0} \sim P_1$.

To solve the above described problem we will aim to analyze the performance of two classes of algorithm based on first order dynamics and optimal discriminator dynamics, both of which are discussed ahead.

First order dynamics In this approach we solve the min-max optimization problem defined in Equation 10 by iteratively applying gradient descent. We minimize the function along the parameters l, r followed by maximizing along the parameters μ, α. Formally, the first order dynamics are specified by the following equations:

$$\begin{aligned} &\text{Given starting points } \mu^{(0)}, \alpha^{(0)}, l^{(0)}, r^{(0)} \text{ and step sizes } \eta_g, \eta_d : \\ &\mu^{(t+1)} = \mu^{(t)} - \eta_g \nabla_\mu L(\mu^{(t)}, \alpha^{(t)}, l^{(t)}, r^{(t)}) \\ &\alpha^{(t+1)} = \alpha^{(t)} - \eta_g \nabla_\alpha L(\mu^{(t)}, \alpha^{(t)}, l^{(t)}, r^{(t)}) \\ &l^{(t+1)} = l^{(t)} - \eta_d \nabla_l L(\mu^{(t)}, \alpha^{(t)}, l^{(t)}, r^{(t)}) \\ &r^{(t+1)} = r^{(t)} - \eta_d \nabla_r L(\mu^{(t)}, \alpha^{(t)}, l^{(t)}, r^{(t)}) \end{aligned}$$

Optimal discriminator dynamics In this approach we perform gradient descent on the function over the parameters of the generator (for solving the "min" component of the problem) while assuming the optimal discriminator at each step (i.e. solving the "max" component of the problem optimally). More formally the optimal discriminator dynamics can be given by:

$$\begin{aligned}
&\text{Given starting points } \mu^{(0)}, \alpha^{(0)} \text{ and step size } \eta_g : \\
&l^{(t)}, r^{(t)} = \underset{l,r}{\operatorname{argmax}}\, L(\mu^{(t)}, \alpha^{(t)}, l, r) \\
&\mu^{(t+1)} = \mu^{(t)} - \eta_g \nabla_\mu L(\mu^{(t)}, \alpha^{(t)}, l^{(t)}, r^{(t)}) \\
&\alpha^{(t+1)} = \alpha^{(t)} - \eta_g \nabla_\alpha L(\mu^{(t)}, \alpha^{(t)}, l^{(t)}, r^{(t)})
\end{aligned}$$

6 Discussion

Our efforts will be in two directions, an empirical analysis and a theoretical analysis. First, we will describe our empirical efforts.

Experimental Design We are especially interested in how the model behaves in the first order and optimal discriminator models in relation to the parameter α^*. In relatively unbiased datasets we can expect experiments run by Li et al. (2018) to well capture the convergence behavior. However, with more extreme values of α^* we expect the probability of convergence to be lower in both models. Since we are focusing on the behavior of the model with respect to α^* we plan to run experiments that explore the behavior over training time of models with differing levels of α^*. Thus we have two main experiments. The first examining the effect of α on convergence behavior. This would be similar to Figure 1 of the Li et al. (2018) paper. We would look for a point at which dynamics become unstable after a large variance from the unbiased distribution of α^*. The second would be similar to Figure 2 of the Li et al. (2018) paper showing a heatmap of success probability over initial condition parameters $\mu_1^{(0)}, \mu_2^{(0)}$ and $\alpha^{(0)}$. We would like to observe the effect of α on the success probability and the dynamics of training given its importance in broader applications of fairness, biology, and dis-entanglement.

Theoretical Design We hope to expand the work by Li et al. (2018) to the biased data setting. We would like to alter the convergence theorems provided in their work to cover the parameter α^* covering the bias in the data. We restate the main theorem from Li et al. (2018) here for convenience (theorem 4.1 in their paper) [8]:

Theorem 1 (Li et al., 2018) *Fix $\delta > 0$ sufficiently small and $C > 0$. Let $\mu^* \in \mathbb{R}^2$ so that $|\mu_i^*| \leq C$, and $|\mu_1^* - \mu_2^*| \geq \delta$. Then for all initial points $|\widehat{\mu}_1^{(0)} - \widehat{\mu}_2^{(0)}| \geq \delta$, if we let $\eta = \operatorname{poly}(1/\delta, e^{-C^2})$ and $T = \operatorname{poly}(1/\delta, e^{-C^2})$, then if $\widehat{\mu}^{(T)}$ is specified by the optimal discriminator dynamics, we have $d_{TV}(G_{\mu^*}, G_{\widehat{\mu}^{(T)}}) \leq \delta$.*

In English, if our data means μ^* are far enough apart, and also within a reasonable range of values then under the optimal discriminator dynamics model we can bound the total variation distance between our learned distribution at time T, $G_{\widehat{\mu}^{(T)}}$ and the data distribution. This is a very non-trivial proof given the non-convexity and non-smoothness of the objective function.

Our desired theorem should be similar to theirs, but our bounds should be as functions of α. Thus we expect to prove a theorem of the following form:

Theorem 2 *Fix $\delta > 0$ sufficiently small, and $B, C > 0$. Let f be a function such that $f : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$. Let $\mu^* \in \mathbb{R}^2$ so that $|\mu_i^*| \leq C$, and $|\mu_1^* - \mu_2^*| \geq \delta$. Let $\alpha^* \in [0, 1]$ so that $\alpha^*(1 - \alpha^*) > B$. Then for all initial points $|\widehat{\mu}_1^{(0)} - \widehat{\mu}_2^{(0)}| \geq \delta$ and $\widehat{\alpha}^{(0)}(1 - \widehat{\alpha}^{(0)}) > B$, if we let $\eta = \operatorname{poly}(1/\delta, 1/B, e^{-C^2})$ and $T = \operatorname{poly}(1/\delta, 1/B, e^{-C^2})$, then if $\widehat{\mu}^{(T)}$ is specified by the optimal discriminator dynamics, we have $d_{TV}(G_{\mu^*, \alpha^*}, G_{\widehat{\mu}^{(T)}, \widehat{\alpha}^{(T)}}) \leq f(\delta, B)$.*

We expect that f should be a linear function of δ and are unsure of its relationship to B. Since B represents the unbiasedness of the data, it seems reasonable to expect that the convergence may be better behaved on datasets with large B than those with small B. At this time we can't offer any expectation on how B should relate to η and T so we have stuck in a $\operatorname{poly}(1/B)$ as a placeholder.

The main challenge in our theoretical work is understanding the dynamics of the training procedure with respect to our new parameter α, since α is in some sense a different type of parameter than μ. Hence, the analysis provided in Li et al. (2018) may not suffice for this new type of parameter.

We have some hope in that for a general n-dimensional gaussian mixture model the problem is a convex (although not closed form) function of the data even under unknown mixture parameters. Thus our model is still learn-able by a different and convex optimization.

Finally, we list a set of other potential lines of enquiry.

1. Generalization from 2 to n gaussians. Our simple model captures the behavior of GANs on two simple populations. Many more datasets can be captured by more than 2 gaussians in some space. Thus we would like to generalize our results to n models. We have hope for this task as the Discriminator and Generator architectures in Li et al. (2018) generalize well to a larger number of gaussians. For two gaussians the Discriminator is a function of two intervals. It can be shown from Theorem A.2 that the number of intervals as a function of the number of gaussians in the 1D dataset scale as $I(n) = 2(n-1)$.
2. Generalization to other distance functions beyond total variation. In practice, state of the art GAN architectures are not trained with the total variation loss [7]. They are trained with losses based on 1-Wasserstein distance, Jensen-Shannon, or KL divergence. While TV divergence is very simplistic under this model it would be excellent to relate the simple model more closely to state of the art applications.
3. Expanding what we learn from our simple model to a more natural application. Ideally, we can use the relationship among parameters learned from our theoretical and experimental work on this toy example to inform a more practical model. Thus we hope to show a disentanglement or fairness application on a real-world dataset.
4. Evaluation of fixed point behavior under this model. This has been done for more complex models but is the line of theoretical work in GANs that has received more attention. We would follow work like that of Arora et al. (2017) who study the conditions under which equilibria exist in GAN training [2].

We would especially appreciate any feedback you have concerning the priority of tasks in within the scope of this project and/or any other items you would find interesting.

References

[1] Matthew Amodio and Smita Krishnaswamy. Generating and Aligning from Data Geometries with Generative Adversarial Networks. *arXiv:1901.08177 [cs, stat]*, January 2019.

[2] Sanjeev Arora, Rong Ge, Yingyu Liang, Tengyu Ma, and Yi Zhang. Generalization and Equilibrium in Generative Adversarial Nets (GANs). *arXiv:1703.00573 [cs, stat]*, March 2017.

[3] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets. *arXiv:1606.03657 [cs, stat]*, June 2016.

[4] Ian Goodfellow. NIPS 2016 Tutorial: Generative Adversarial Networks. *arXiv:1701.00160 [cs]*, December 2016.

[5] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Networks. *arXiv:1406.2661 [cs, stat]*, June 2014.

[6] Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In *arXiv:1312.6114 [Cs, Stat]*, December 2013.

[7] Karol Kurach, Mario Lucic, Xiaohua Zhai, Marcin Michalski, and Sylvain Gelly. The GAN Landscape: Losses, Architectures, Regularization, and Normalization. *arXiv:1807.04720 [cs, stat]*, July 2018.

[8] Jerry Li, Aleksander Madry, John Peebles, and Ludwig Schmidt. On the Limitations of First-Order Approximation in GAN Dynamics. In *Proceedings of the 35th International Conference on Machine Learning*, Stockholm, Sweeden, 2018.

[9] Ofir Lindenbaum, Jay S. Stanley III, Guy Wolf, and Smita Krishnaswamy. Geometry-Based Data Generation. *arXiv:1802.04927 [cs]*, February 2018.

[10] Carl Edward Rasmussen. The Infinite Gaussian Mixture Model. In *NeurIPs*, page 7, 2000.

[11] Prasanna Sattigeri, Samuel C. Hoffman, Vijil Chenthamarakshan, and Kush R. Varshney. Fairness GAN. *arXiv:1805.09910 [cs, stat]*, May 2018.

[12] David W. Scott. *Kernel Density Estimators*, chapter 6, pages 137–216. John Wiley & Sons, Ltd, 2015.

[13] Depeng Xu, Shuhan Yuan, Lu Zhang, and Xintao Wu. FairGAN: Fairness-aware Generative Adversarial Networks. *arXiv:1805.11202 [cs, stat]*, May 2018.