Adversarial Variational Optimization of Non-Differentiable Simulators

Gilles Louppe[1] and **Kyle Cranmer**[1]

[1]*New York University*

Complex computer simulators are increasingly used across fields of science as generative models tying parameters of an underlying theory to experimental observations. Inference in this setup is often difficult, as simulators rarely admit a tractable density or likelihood function. We introduce Adversarial Variational Optimization (AVO), a likelihood-free inference algorithm for fitting a non-differentiable generative model or to perform empirical Bayes through variational inference. We adapt the training procedure of generative adversarial networks by replacing the differentiable generative network with a domain-specific simulator. We solve the resulting non-differentiable minimax problem by minimizing variational upper bounds of the two adversarial objectives. Effectively, the procedure results in learning an arbitrarily tight proposal distribution over simulator parameters, such that the corresponding marginal distribution of the generated data matches the observations. We present results of the method with simulators producing both discrete and continuous data.

I. INTRODUCTION

In many fields of science such as particle physics, epidemiology, and population genetics, computer simulators are used to describe complex data generation processes. These simulators relate observations $\mathbf{x}$ to the parameters $\boldsymbol{\theta}$ of an underlying theory or mechanistic model. In most cases, these simulators are specified as procedural implementations of forward, stochastic processes involving latent variables $\mathbf{z}$. Rarely do these simulators admit a tractable density (or likelihood) $p(\mathbf{x}|\boldsymbol{\theta})$. The prevalence and significance of this problem has motivated an active research effort in so-called *likelihood-free inference* algorithms such as Approximate Bayesian Computation (ABC) [1–6]. Often the simulation code involves control-flow that implies the dependence of $\mathbf{x}$ on $\mathbf{z}$ is non-differentiable, making inference even more difficult and precludes approaches such as Ref. [7].

In parallel, with the introduction of variational auto-encoders [8] and generative adversarial networks [9], there has been a vibrant research program around implicit generative models based on neural networks [10]. While these implicit generative networks also do not admit a tractable density, they are differentiable.

Generative models based on neural networks are highly parametrized and the model parameters have no obvious interpretation. In contrast, scientific simulators can be thought of as highly regularized generative models as they typically have relatively few parameters and they are endowed with some level of interpretation. In this setting, inference on the model parameters $\boldsymbol{\theta}$ is often of more interest than the latent variables $\mathbf{z}$.

In this note, we develop an unsupervised learning algorithm for the point estimation of the parameters $\boldsymbol{\theta}$ of a non-differentiable, implicit generative model. We adapt the adversarial training procedure of generative adversarial networks [9] by replacing the implicit generative network with a domain-based scientific simulator, and solve the resulting non-differentiable minimax problem by minimizing variational upper bounds [11, 12] of the adversarial objectives. The procedure results in learning an arbitrarily tight proposal distribution over simulator parameters, such that the corresponding marginal distribution of the generated data matches the observations.

II. PROBLEM STATEMENT

We consider a family of parametrized densities $p(\mathbf{x}|\boldsymbol{\theta})$ defined implicitly through the simulation of a stochastic generative process, where $\mathbf{x} \in \mathbb{R}^d$ is the data and $\boldsymbol{\theta}$ are the parameters of interest. The simulation may involve some complicated latent process where $\mathbf{z} \in \mathcal{Z}$ is a latent variable providing an external source of randomness. Unlike implicit generative models defined by neural networks, we do not assume $\mathbf{z}$ to be a fixed-size vector with a simple density. Instead, the dimension of $\mathbf{z}$ and the nature of its components (uniform, normal, discrete, continuous, etc.) are inherited from the control flow of the simulation code and may depend on $\boldsymbol{\theta}$ in some intricate way. Moreover, the dimension of $\mathbf{z}$ may be much larger than the dimension of $\mathbf{x}$. Thus we do not attempt to learn a recognition model $q(\mathbf{z}|\mathbf{x})$.

We assume that the stochastic generative process that defines $p(\mathbf{x}|\boldsymbol{\theta})$ is specified through a non-differentiable deterministic function $g(\cdot;\boldsymbol{\theta}) : \mathcal{Z} \to \mathbb{R}^d$. Operationally,

$$\mathbf{x} \sim p(\mathbf{x}|\boldsymbol{\theta}) \equiv \mathbf{z} \sim p(\mathbf{z}|\boldsymbol{\theta}), \mathbf{x} = g(\mathbf{z};\boldsymbol{\theta}) \tag{1}$$

such that the density $p(\mathbf{x}|\boldsymbol{\theta})$ can be written as

$$p(\mathbf{x}|\boldsymbol{\theta}) = \int_{\{\mathbf{z}:g(\mathbf{z};\boldsymbol{\theta})=\mathbf{x}\}} p(\mathbf{z}|\boldsymbol{\theta})\mu(d\mathbf{z}), \tag{2}$$

where μ is a probability measure.

Given some observed data $\{\mathbf{x}_i|i = 1,\dots,N\}$ drawn from the (unknown) true distribution $p_r(\mathbf{x})$, our goal is to estimate the parameters $\boldsymbol{\theta}^*$ that minimize the divergence between $p_r(\mathbf{x})$ and the implicit model $p(\mathbf{x}|\boldsymbol{\theta})$. That is,

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}\in\boldsymbol{\theta}} \rho(p_r(\mathbf{x}), p(\mathbf{x}|\boldsymbol{\theta})), \tag{3}$$

where ρ is some distance or divergence.

III. BACKGROUND

A. Generative adversarial networks

Generative adversarial networks (GANs) were first proposed by [9] as a way to build an implicit generative model capable of producing samples from random noise $\mathbf{z}$. More specifically, a generative model $g(\cdot;\boldsymbol{\theta})$ is pit against an adversarial classifier $d(\cdot;\boldsymbol{\phi}) : \mathbb{R}^d \to [0,1]$ with parameters $\boldsymbol{\phi}$ and whose antagonistic objective is to recognize real data $\mathbf{x}$ from generated data $\tilde{\mathbf{x}} = g(\mathbf{z};\boldsymbol{\theta})$. Both models g and d are trained simultaneously, in such a way that g learns to fool its adversary d (which happens when g produces samples comparable to the observed data), while d continuously adapts to changes in g. When d is trained to optimality before each parameter update of the generator, it can be shown that the original adversarial learning procedure [9] amounts to minimizing the Jensen-Shannon divergence $\mathrm{JSD}(p_r(\mathbf{x}) \parallel p(\mathbf{x}|\boldsymbol{\theta}))$ between $p_r(\mathbf{x})$ and $p(\mathbf{x}|\boldsymbol{\theta})$.

As thoroughly explored in [13], GANs remain remarkably difficult to train because of vanishing gradients as d saturates, or because of unreliable updates when the training procedure is relaxed. As a remedy, Wasserstein GANs [14] reformulate the adversarial setup in order to minimize the Wasserstein-1 distance $W(p_r(\mathbf{x}), p(\mathbf{x}|\boldsymbol{\theta}))$ by replacing the adversarial classifier with a 1-Lipschitz adversarial critic $d(\cdot;\boldsymbol{\phi}) : \mathbb{R}^d \to \mathbb{R}$, and where

$$\begin{aligned} & W(p_r(\mathbf{x}), p(\mathbf{x}|\boldsymbol{\theta})) \\ &= \sup_{\boldsymbol{\phi}} \mathbb{E}_{\tilde{\mathbf{x}}\sim p(\mathbf{x}|\boldsymbol{\theta})}[d(\tilde{\mathbf{x}};\boldsymbol{\phi})] - \mathbb{E}_{\mathbf{x}\sim p_r(\mathbf{x})}[d(\mathbf{x};\boldsymbol{\phi})] \\ &\equiv \sup_{\boldsymbol{\phi}} \mathcal{L}_W. \end{aligned} \tag{4}$$

Under the WGAN-GP formulation of [15] for stabilizing the optimization procedure, training d and g results in alternating gradient updates on $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$ in order to respectively minimize

$$\mathcal{L}_d = \mathcal{L}_W + \lambda \mathbb{E}_{\hat{\mathbf{x}}\sim p(\hat{\mathbf{x}})}[(||\nabla_{\hat{\mathbf{x}}} d(\hat{\mathbf{x}};\boldsymbol{\phi})||_2 - 1)^2] \tag{5}$$

$$\mathcal{L}_g = -\mathcal{L}_W \tag{6}$$

where $\hat{\mathbf{x}} := \epsilon\mathbf{x} + (1-\epsilon)\tilde{\mathbf{x}}$, for $\epsilon \sim U[0,1]$, $\mathbf{x} \sim p_r(\mathbf{x})$ and $\tilde{\mathbf{x}} \sim p(\mathbf{x}|\boldsymbol{\theta})$.

B. Variational optimization

Variational optimization [12, 16] and evolution strategies [11] are general optimization techniques that can be used to form a differentiable bound on the optima of a non-differentiable function. Given a function f to minimize, these techniques are based on the simple fact that

$$\min_{\boldsymbol{\theta}\in\Theta} f(\boldsymbol{\theta}) \leq \mathbb{E}_{\boldsymbol{\theta}\sim q(\boldsymbol{\theta}|\boldsymbol{\psi})}[f(\boldsymbol{\theta})] = U(\boldsymbol{\psi}), \tag{7}$$

where $q(\boldsymbol{\theta}|\boldsymbol{\psi})$ is a proposal distribution with parameters $\boldsymbol{\psi}$ over input values $\boldsymbol{\theta}$. That is, the minimum of a set of function values is always less than or equal to any of their average. Provided that the proposal is flexible enough, the parameters $\boldsymbol{\psi}$ can be updated to place its mass arbitrarily tight around the optimum $\boldsymbol{\theta}^* = \min_{\boldsymbol{\theta}\in\Theta} f(\boldsymbol{\theta})$.

Under mild restrictions outlined in [12], the bound $U(\boldsymbol{\psi})$ is differentiable with respect to $\boldsymbol{\psi}$, and using the log-likelihood trick it can be rewritten as:

$$\begin{aligned} \nabla_{\boldsymbol{\psi}} U(\boldsymbol{\psi}) &= \nabla_{\boldsymbol{\psi}} \mathbb{E}_{\boldsymbol{\theta}\sim q(\boldsymbol{\theta}|\boldsymbol{\psi})}[f(\boldsymbol{\theta})] \\ &= \mathbb{E}_{\boldsymbol{\theta}\sim q(\boldsymbol{\theta}|\boldsymbol{\psi})}[f(\boldsymbol{\theta}) \nabla_{\boldsymbol{\psi}} \log q(\boldsymbol{\theta}|\boldsymbol{\psi})] \end{aligned} \tag{8}$$

Effectively, this means that provided that the score function $\nabla_{\boldsymbol{\psi}} \log q(\boldsymbol{\theta}|\boldsymbol{\psi})$ of the proposal is known and that one can evaluate $f(\boldsymbol{\theta})$ for any $\boldsymbol{\theta}$, then one can construct empirical estimates of Eqn. 8, which can in turn be used to minimize $U(\boldsymbol{\psi})$ with stochastic gradient descent. We will use Adam [17], but note the opportunity to incorporate the Natural Evolution Strategy algorithm [11] and variance reduction techniques [18].

IV. ADVERSARIAL VARIATIONAL OPTIMIZATION

The alternating stochastic gradient descent on $\mathcal{L}_d$ and $\mathcal{L}_g$ in GANs (Section III A) inherently assumes that the generator g is a differentiable function. In the setting where we are interested in estimating the parameters of a fixed non-differentiable simulator (Section II), rather than in learning the generative model itself, gradients $\nabla_{\boldsymbol{\theta}} g$ either do not exist or are inaccessible. As a result, gradients $\nabla_{\boldsymbol{\theta}} \mathcal{L}_g$ cannot be constructed and the optimization procedure cannot be carried out.

In this work, we propose to rely on variational optimization to minimize $\mathcal{L}_d$ and $\mathcal{L}_g$, thereby bypassing the non-differentiability of g. More specifically, we consider a proposal distribution $q(\boldsymbol{\theta}|\boldsymbol{\psi})$ over the parameters of g and $p(\mathbf{x}|\boldsymbol{\theta})$ and alternately minimize the variational upper bounds

$$U_d = \mathbb{E}_{\boldsymbol{\theta}\sim q(\boldsymbol{\theta}|\boldsymbol{\psi})}[\mathcal{L}_d] \tag{9}$$

$$U_g = \mathbb{E}_{\boldsymbol{\theta}\sim q(\boldsymbol{\theta}|\boldsymbol{\psi})}[\mathcal{L}_g] \tag{10}$$

respectively over $\boldsymbol{\phi}$ and $\boldsymbol{\psi}$. When updating $\boldsymbol{\phi}$, unbiased estimates of $\nabla_{\boldsymbol{\phi}} U_d$ can be obtained by directly evaluating the gradient of U_d over mini-batches of real and generated data, as is ordinarily done in stochastic gradient descent. When updating $\boldsymbol{\psi}$, $\nabla_{\boldsymbol{\psi}} U_g$ can be estimated as described in the previous section. That is,

$$\nabla_{\boldsymbol{\psi}} U_g = \mathbb{E}_{\substack{\boldsymbol{\theta}\sim q(\boldsymbol{\theta}|\boldsymbol{\psi}), \\ \tilde{\mathbf{x}}\sim p(\mathbf{x}|\boldsymbol{\theta})}}[-d(\tilde{\mathbf{x}};\boldsymbol{\phi}) \nabla_{\boldsymbol{\psi}} \log q(\boldsymbol{\theta}|\boldsymbol{\psi})], \tag{11}$$

which we can approximate with mini-batches of generated data

$$\nabla_{\boldsymbol{\psi}} U_g \approx \frac{1}{M} \sum_{m=1}^{M} -d(g(\mathbf{z}_m;\boldsymbol{\theta}_m);\boldsymbol{\phi}) \nabla_{\boldsymbol{\psi}} \log q(\boldsymbol{\theta}_m|\boldsymbol{\psi}) \tag{12}$$

Algorithm 1 Adversarial variational optimization (AVO).

Inputs: observed data $\{\mathbf{x}_i \sim p_r(\mathbf{x})\}_{i=1}^N$, simulator g.
Outputs: proposal distribution $q(\boldsymbol{\theta}|\boldsymbol{\psi})$, such that $q(\mathbf{x}|\boldsymbol{\psi}) \approx p_r(\mathbf{x})$.
Hyper-parameters: The number n_{critic} of training iterations of d; the size M of a mini-batch; the gradient penalty coefficient λ; the entropy penalty coefficient γ.

$q(\boldsymbol{\theta}|\boldsymbol{\psi}) \leftarrow$ prior on $\boldsymbol{\theta}$ (with differentiable and known density)
while $\boldsymbol{\psi}$ has not converged **do**
  for $i = 1$ to n_{critic} **do** ▷ Update d
    Sample a mini-batch $\{\mathbf{x}_m \sim p_r(\mathbf{x}), \boldsymbol{\theta}_m \sim q(\boldsymbol{\theta}|\boldsymbol{\psi}), \mathbf{z}_m \sim p(\mathbf{z}|\boldsymbol{\theta}_m), \epsilon_m \sim U[0,1]\}_{m=1}^M$.
    for $m = 1$ to M **do**
      $\tilde{\mathbf{x}}_m \leftarrow g(\mathbf{z}_m; \boldsymbol{\theta}_m)$
      $\hat{\mathbf{x}}_m \leftarrow \epsilon_m \mathbf{x}_m + (1-\epsilon_m)\tilde{\mathbf{x}}_m$
      $U_d^{(m)} \leftarrow d(\tilde{\mathbf{x}}_m; \boldsymbol{\phi}) - d(\mathbf{x}_m; \boldsymbol{\phi}) + \lambda(||\nabla_{\hat{\mathbf{x}}_m} d(\hat{\mathbf{x}}_m; \boldsymbol{\phi})||_2 - 1)^2$
    end for
    $\boldsymbol{\phi} \leftarrow \text{Adam}(\nabla_{\boldsymbol{\phi}} \frac{1}{M} \sum_{m=1}^M U_d^{(m)})$
  end for
  Sample a mini-batch $\{\boldsymbol{\theta}_m \sim q(\boldsymbol{\theta}|\boldsymbol{\psi}), \mathbf{z}_m \sim p(\mathbf{z}|\boldsymbol{\theta}_m)\}_{m=1}^M$. ▷ Update $q(\boldsymbol{\theta}|\boldsymbol{\psi})$
  $\nabla_{\boldsymbol{\psi}} U_g \leftarrow \frac{1}{M} \sum_{m=1}^M -d(g(\mathbf{z}_m; \boldsymbol{\theta}_m)) \nabla_{\boldsymbol{\psi}} \log q_{\boldsymbol{\psi}}(\boldsymbol{\theta}_m)$
  $\nabla_{\boldsymbol{\psi}} H(q_{\boldsymbol{\psi}}) \leftarrow \frac{1}{M} \sum_{m=1}^M \nabla_{\boldsymbol{\psi}} q_{\boldsymbol{\psi}}(\boldsymbol{\theta}_m) \log q_{\boldsymbol{\psi}}(\boldsymbol{\theta}_m)$
  $\boldsymbol{\psi} \leftarrow \text{Adam}(\nabla_{\boldsymbol{\psi}} U_g + \gamma \nabla_{\boldsymbol{\psi}} H(q_{\boldsymbol{\psi}}))$
end while

for $\boldsymbol{\theta}_m \sim q(\boldsymbol{\theta}|\boldsymbol{\psi})$ and $\mathbf{z}_m \sim p(\mathbf{z}|\boldsymbol{\theta}_m)$. For completeness, Algorithm 1 outlines the proposed adversarial variational optimization (AVO) procedure, as built on top of WGAN-GP.

Algorithm 1 represents the simplest version of AVO; however, the variance of the noisy estimator of the gradients may be too large to be useful in many problems. Improvements are possible using variance reduction techniques such as Rao-Blackwellization, Control Variates, and quasi-Monte Carlo [18, 19]. Similarly, various fast algorithms for calculating the exact (entropically regularized) Wasserstein distance on empirical distributions are being developed [20–22]. Similarly, the algorithm could be adopted to use estimates to the KL distance [23] without the adversarial optimization for the critic. These variants may be particularly useful for small N.

A. Empirical Bayes through Adversarial Variational Inference

The variational objectives 9-10 have the effect of replacing the modeled data distribution of Eqn. 1 with the marginal distribution parametrized by $\boldsymbol{\psi}$

$$q(\mathbf{x}|\boldsymbol{\psi}) = \int p(\mathbf{x}|\boldsymbol{\theta}) q(\boldsymbol{\theta}|\boldsymbol{\psi}) d\boldsymbol{\theta}. \tag{13}$$

We can think of $p(\mathbf{x}|\boldsymbol{\psi})$ as a *variational program* as described in [24], though more complicated than a simple reparameterization of normally distributed noise $\mathbf{z}$ through a differentiable function $g(\mathbf{z}, \boldsymbol{\psi})$. In our case, the variational program is a marginalized, non-differentiable simulator. Its density is intractable; nevertheless, it can generate samples for $\mathbf{x}$ and is differentiable with respect to $\boldsymbol{\psi}$ and $\boldsymbol{\phi}$. Operationally, we sample from this marginal model via

$$\mathbf{x} \sim q(\mathbf{x}|\boldsymbol{\psi}) \equiv \boldsymbol{\theta} \sim q(\boldsymbol{\theta}|\boldsymbol{\psi}), \mathbf{z} \sim p(\mathbf{z}|\boldsymbol{\theta}), \mathbf{x} = g(\mathbf{z}; \boldsymbol{\theta}). \tag{14}$$

Typically in variational inference one starts with a tractable prior $p(\boldsymbol{\theta})$ and a tractable likelihood $p(\mathbf{x}|\boldsymbol{\theta})$, thus fixing the posterior $p(\boldsymbol{\theta}|\mathbf{x})$. After optimizing $q(\boldsymbol{\theta}|\boldsymbol{\psi})$ to approximate the posterior, one would criticize the model through the the posterior predictive distribution as advocated by Box [25]. Effectively, we are optimizing variational posterior $q(\boldsymbol{\theta}|\psi)$ by directly minimizing the divergence of the corresponding posterior predictive distribution $q(\mathbf{x}|\boldsymbol{\psi})$ and the data distribution $p_r(\mathbf{x})$ without evaluating the intractable likelihood $p(\mathbf{x}|\boldsymbol{\theta})$ or specifying a prior. Consequently, the proposed algorithm is a form of variational inference through model criticism [25].

We can also view the optimization through the lens of empirical Bayes (EB), where one specifies a family of priors and $p(\boldsymbol{\theta}|\boldsymbol{\psi})$ and then estimates $\boldsymbol{\psi}^*$ using the data. Traditionally, we would have access to the likelihood and optimize with respect to the family of posteriors $p(\boldsymbol{\theta}|\mathbf{x}, \boldsymbol{\psi}) = p(\mathbf{x}|\boldsymbol{\theta}) p(\boldsymbol{\theta}|\boldsymbol{\psi}^*) / \int p(\mathbf{x}|\boldsymbol{\theta}) p(\boldsymbol{\theta}|\boldsymbol{\psi}^*) d\boldsymbol{\theta}$ as an intermediate step to computing $\boldsymbol{\psi}^*$. In our case, the likelihood is intractable and we have not explicitly specified a prior family. However, the procedure corresponds to empirical Bayes if we identify $q(\boldsymbol{\theta}|\boldsymbol{\psi}^*) = p(\boldsymbol{\theta}|\mathbf{x}, \boldsymbol{\psi})$. Thus we call our method AVO-EB and note that it is applicable even outside of the likelihood-free context.

[KC: Clarify hierarchical models and iid data x. Factor of N/M for AVO-EB?]

B. Point estimates

When ρ is the Wasserstein distance, $\boldsymbol{\theta}^*$ is referred to as the minimum Wasserstein estimator (MWE). When the model is well specified, $\boldsymbol{\theta}^*$ coincides with the true data-generating parameter; however, if the model is misspecified, $\boldsymbol{\theta}^*$ is typically different from the maximum likelihood estimator (MLE), which is the minimizer of $KL(p_r(\mathbf{x})|p(\mathbf{x}|\boldsymbol{\theta}))$ [26]. In order to more effectively target point estimates $\boldsymbol{\theta}^*$, we augment Eqn. 10 with a regularization term corresponding to the differential entropy H of the proposal distribution $q(\boldsymbol{\theta}|\boldsymbol{\psi})$. That is,

$$U_g = \mathbb{E}_{\boldsymbol{\theta}\sim q(\boldsymbol{\theta}|\boldsymbol{\psi})}[\mathcal{L}_g] + \gamma H(q(\boldsymbol{\theta}|\boldsymbol{\psi})) \tag{15}$$

where $\gamma \in \mathbb{R}^+$ is a hyper-parameter controlling the trade-off between the generator objective and the tightness of the proposal distribution. We refer to the algorithm with $\gamma > 0$ as AVO-MWE. For small values of γ, proposal distributions with large entropy are not penalized, which results in learning a smeared variation of the original simulator corresponding to the posterior predictive model. On the other hand, for large values of γ, the procedure is encouraged to fit a proposal distribution with low entropy, which has the effect of concentrating its density tightly around one or a few $\boldsymbol{\theta}$ values.

Very large penalties may eventually make the optimization unstable, as the variance of $\nabla_{\boldsymbol{\psi}} \log q(\boldsymbol{\theta}_m|\boldsymbol{\psi})$ typically increases as the entropy of the proposal decreases.

V. EXPERIMENTS

A. Univariate discrete data

As a first illustrative experiment, we evaluate inference for a discrete Poisson distribution with unknown mean λ. We artificially consider the distribution as a parametrized simulator, from which we can only generate data.

The observed data is sampled from a Poisson with mean $\lambda^* = 7$. Algorithm 1 is run for 300 epochs with mini-batches of size $M = 64$ and the following configuration. For the critic d, we use a 3-layer MLP with 10 hidden nodes per layer and ReLU activations. At each epoch, Adam is run for $n_{\text{critic}} = 100$ iterations with a step size $\alpha = 0.01$, decay rates $\beta_1 = \beta_2 = 0.5$ and its inner first and second moment vectors reset at each outer epoch in order to avoid building momentum in staled directions. For estimating λ^*, we parameterize $\boldsymbol{\theta}$ as $\log(\lambda)$ and use a univariate Gaussian proposal distribution $q(\boldsymbol{\theta}|\boldsymbol{\psi})$ initialized with a mean at $\log(5)$ and unit variance. At each epoch, parameters $\boldsymbol{\psi}$ are updated by taking one Adam step, with $\alpha = 0.01$ and $\beta_1 = \beta_2 = 0.5$. The gradient penalty coefficient is set to $\lambda = 0.025$, and the entropy penalty is evaluated at both $\gamma = 0$ and $\gamma = 5$.

The top left plot in Figure 1 illustrates the resulting proposal distributions $q(\boldsymbol{\theta}|\boldsymbol{\psi})$ after AVO. For both $\gamma = 0$ and $\gamma = 5$, the proposal distributions correctly concentrate their density around the true parameter value $\log(\lambda^*) = 1.94$. Under the effect of the positive entropy penalty $H(q(\boldsymbol{\theta}|\boldsymbol{\psi}))$, the proposal distribution for $\gamma = 5$ concentrates its mass more tightly, yielding in this case a more precise inference. The top right plot compares the model distributions to the true distribution. As theoretically expected from adversarial training, we see that the resulting distributions closely match the true distribution, with in this case visually slightly better results for the penalized model. The bottom plot of Figure 1 shows empirical estimates of $-U_d$ with respect to the epoch number. For both $\gamma = 0$ and $\gamma = 5$, the curves quickly fall towards 0, which indicates that $\mathbb{E}_{\tilde{\mathbf{x}}\sim p(\mathbf{x}|\boldsymbol{\theta})}[d(\tilde{\mathbf{x}};\boldsymbol{\phi})] \approx \mathbb{E}_{\mathbf{x}\sim p_r(\mathbf{x})}[d(\mathbf{x};\boldsymbol{\phi})]$ and that the critic cannot distinguish between true and model data. This confirms that adversarial variational optimization works despite the discreteness of the data and lack of access to the density $p(\mathbf{x}|\boldsymbol{\theta})$ or its gradient.

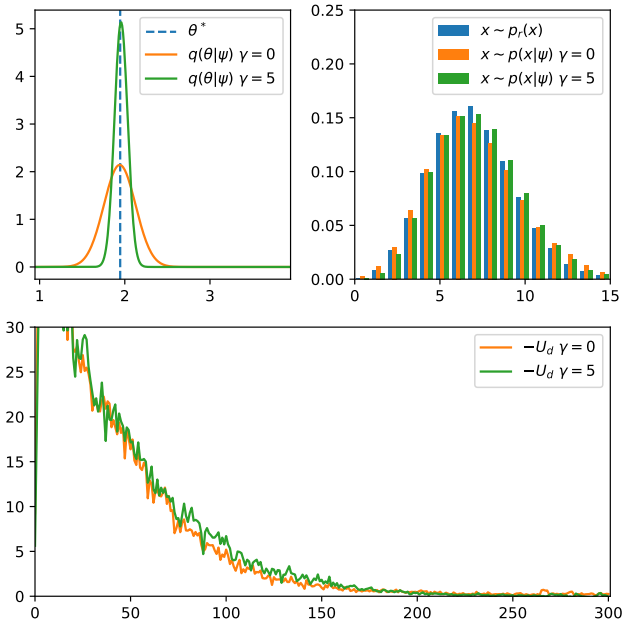

FIG. 1. Discrete Poisson model with unknown mean. (*Top left*) Proposal distributions $q(\boldsymbol{\theta}|\boldsymbol{\psi})$ after adversarial variational optimization. For both $\gamma = 0$ and $\gamma = 5$, the distributions correctly concentrate their density around the true value $\log(\lambda^*)$. Penalizing the entropy of the proposal distribution ($\gamma = 5$) results in a tighter density. (*Top right*) Model distributions $q(\mathbf{x}|\boldsymbol{\psi})$ after training. This plot shows that the resulting parameterizations of the simulator closely reproduce the true distribution. (*Bottom*) Empirical estimates of the variational upper bound U_d as optimization progresses.

B. Multidimensional continuous data

As a second toy example, we consider a generator producing 5-dimensional continuous data, as originally specified in Section 4.2. of [6]. More specifically, we consider the following generative process:

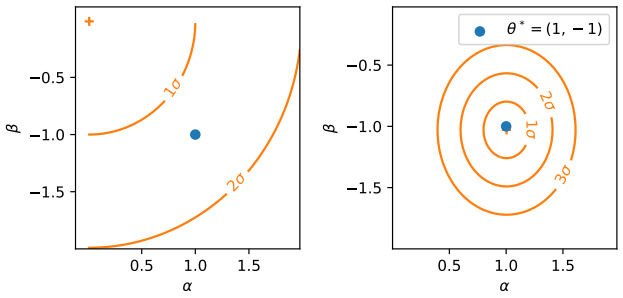

FIG. 2. Multidimensional continuous data. (*Left*) Density $q(\boldsymbol{\theta}|\boldsymbol{\psi})$ at the beginning of the procedure, for a proposal distribution initialized with zero mean and unit variance. (*Right*) Density $q(\boldsymbol{\theta}|\boldsymbol{\psi})$ after adversarial variational optimization ($\gamma = 10$). The proposal density correctly converges towards a distribution whose density concentrates around $\boldsymbol{\theta}^* = (1, -1)$.

- $\mathbf{z} = (z_0, z_1, z_2, z_3, z_4)$, such that $z_0 \sim \mathcal{N}(\mu = \alpha, \sigma = 1)$, $z_1 \sim \mathcal{N}(\mu = \beta, \sigma = 3)$, $z_2 \sim \text{Mixture}[\frac{1}{2}\mathcal{N}(\mu = -2, \sigma = 1), \frac{1}{2}\mathcal{N}(\mu = 2, \sigma = 0.5)]$, $z_3 \sim \text{Exponential}(\lambda = 3)$, and $z_4 \sim \text{Exponential}(\lambda = 0.5)$;
- $\mathbf{x} = R\mathbf{z}$, where R is a fixed semi-positive definite 5×5 matrix defining a fixed projection of $\mathbf{z}$ into the observed space.

Again, the AVO algorithm does not have access to the density or its gradient, only samples from the generative model We consider observed data generated at the nominal values $\boldsymbol{\theta}^* = (\alpha^* = 1, \beta^* = -1)$. The simulator parameters are modeled with a factored (mean field) Gaussian variational distribution $q(\boldsymbol{\theta}|\boldsymbol{\psi}) = q(\alpha|\boldsymbol{\psi})q(\beta|\boldsymbol{\psi})$, where each component was initialized with zero mean and unit variance. Hyper-parameters are set to $M = 64$, $n_{\text{critic}} = 100$, $\lambda = 0.025$, $\gamma = 10$ and Adam configured with $\alpha = 0.01$, $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The network architecture of the critic is the same as in the previous example.

Starting with a proposal distribution $q(\boldsymbol{\theta}|\boldsymbol{\psi})$ largely spread over the parameter space, as illustrated in the left plot of Figure 2, AVO quickly converges towards a variational distribution whose density concentrates around the nominal values $\boldsymbol{\theta}^*$, as shown in the right plot of Figure 2. Overall, this example further illustrates and confirms the ability of adversarial variational optimization for inference with multiple parameters and multidimensional data, where reliable approximations of $p(\mathbf{x}|\boldsymbol{\theta})$ in a traditional MLE setting would otherwise be difficult to construct.

C. Electron–positron annihilation

As a more realistic example, we now consider a (simplified) simulator from particle physics for electron–positron

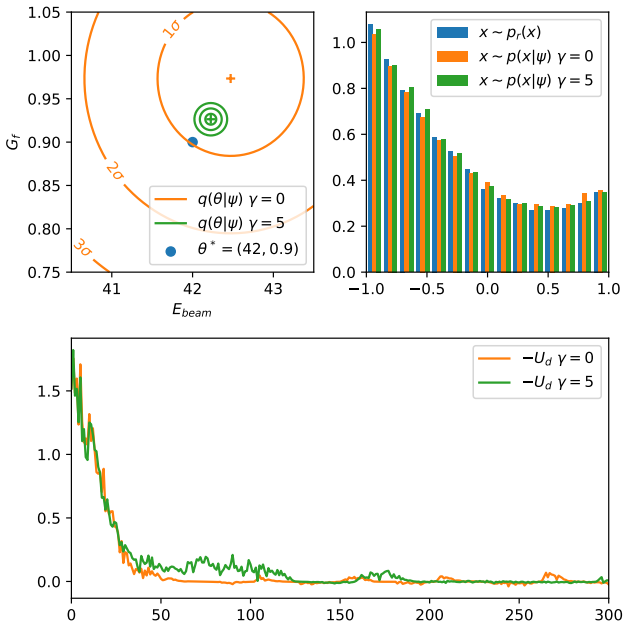

FIG. 3. Electron–positron annihilation. (*Top left*) Proposal distributions $q(\boldsymbol{\theta}|\boldsymbol{\psi})$ after adversarial variational optimization. The density of the penalized distribution ($\gamma = 5$) is here highly concentrated, resulting in the green mass near $\boldsymbol{\theta}^*$. (*Top right*) Model distributions $q(\mathbf{x}|\boldsymbol{\psi})$ after training. Despite the differences between their respective proposal distributions, both models closely match the observed data. (*Bottom*) Empirical estimates of the variational upper bound U_d as optimization progresses.

collisions resulting in muon–antimuon pairs ($e^+e^- \to \mu^+\mu^-$). The simulator approximates the distribution of observed measurements $\mathbf{x} = \cos(A) \in [-1, 1]$, where A is the polar angle of the outgoing muon with respect to the originally incoming electron. Neglecting measurement uncertainty induced from the particle detectors, this random variable is approximately distributed as

$$p(\mathbf{x}|E_{\text{beam}}, G_f) = \frac{1}{Z}\left[(1 + \mathbf{x}^2) + c(E_{\text{beam}}, G_f)\mathbf{x}\right] \quad (16)$$

where Z is a known normalization constant and c is an asymmetry coefficient function. Due to the linear term in the expression, the density $p(\mathbf{x}|E_{\text{beam}}, G_f)$ exhibits a so-called *forward-backward* asymmetry. Its size depends on the values of the parameters E_{beam} (the beam energy) and G_f (the Fermi constant) through the coefficient function c.

A typical physics simulator for this process includes a more precise treatment of the quantum mechanical $e^+e^- \to \mu^+\mu^-$ scattering using `MadGraph` [27], ionization of matter in the detector due to the passage of the out-going $\mu^+\mu^-$ particles using `GEANT4` [28], electronic noise and other details of the sensors that measure the ionization signal, and the deterministic algorithms that estimate the polar angle A based on the sensor readouts. The simulation of this process is highly non-trivial as is

the space of latent variables $\mathcal{Z}$.

In this example, we consider observed data generated with the simplified generator of Eqn. 16 using $\boldsymbol{\theta}^* = (E^*_{\text{beam}} = 42, G^*_f = 0.9)$. The simulator parameters are modeled with a factored (mean field) Gaussian variational distribution $q(\boldsymbol{\theta}|\boldsymbol{\psi}) = q(E_{\text{beam}}|\boldsymbol{\psi})q(G_f|\boldsymbol{\psi})$, where each component is respectively initialized with mean 45 and 1 and variance 1 and 0.01. Hyper-parameters are set to $M = 64$, $n_{\text{critic}} = 100$, $\lambda = 0.0025$ and Adam configured with $\alpha = 0.01$, $\beta_1 = 0.9$ and $\beta_2 = 0.999$. As with the first example, we compare entropy penalties $\gamma = 0$ and $\gamma = 5$.

The top left plot in Figure 3 illustrates the resulting proposal distributions $q(\boldsymbol{\theta}|\boldsymbol{\psi})$ for $\gamma = 0$ and $\gamma = 5$ after AVO. We see that the distributions both arrive in the neighborhood of $\boldsymbol{\theta}^*$, with a density more highly concentrated for $\gamma = 5$ than for $\gamma = 0$. Despite these differences and the relative distance with $\boldsymbol{\theta}^*$, both models closely match the observed data, as shown in the top right plot of Figure 3, with again slightly better results for the entropy penalized model. This suggests either a relatively flat landscape around $\boldsymbol{\theta}^*$ or that the observed data can in this case also be reproduced with the posterior predictive distribution $q(\mathbf{x}|\boldsymbol{\psi})$. Finally, the bottom plot of Figure 3 shows that for both $\gamma = 0$ and $\gamma = 5$ the variational upper bound $-U_d$ quickly fall towards 0, which indicates convergence towards a distribution that the critic cannot distinguish from the true distribution.

VI. RELATED WORK

This work sits at the intersection of several lines of research related to likelihood-free inference, approximate Bayesian computation (ABC), implicit generative networks, and variational inference. Viewed from the literature around implicit generative models based on neural networks, the proposed method can be considered as a direct adaptation of generative adversarial networks [9, 13, 14] to non-differentiable simulators using variational optimization [12, 16].

Adversarial Learned Inference (ALI) [29], Bidirectional GANs (BiGANs) [30], α-GAN [31], Adversarial Variational Bayes (AVB) [32], and the `PC-Adv` algorithm of [33] are a recent extension to GANs that add an inference network to the generative model. Each of these assume a tractable density $p(\mathbf{x}|\boldsymbol{\theta})$ that is differentiable with respect to $\boldsymbol{\theta}$, which is not satisfied in the likelihood-free setting. Our lifting of the non-differentiable simulator $p(\mathbf{x}|\boldsymbol{\theta})$ to the variational program $q(\mathbf{x}|\boldsymbol{\psi})$ provides the ability to differentiate expectations with respect to $\boldsymbol{\psi}$ as in Eqn 8; however, the density $q(\mathbf{x}|\boldsymbol{\psi})$ is still intractable. Moreover, we do not attempt to define a recognition model $q(\mathbf{z}|\boldsymbol{\theta}, \mathbf{x})$ as the latent space $\mathcal{Z}$ of many real-world simulators is complicated and not amenable to neural a recognition model.

From the point of view of likelihood-free inference, where non-differentiable simulators are the norm, our contributions are threefold. First is the process of lifting expectations with respect to the non-differentiable simulator $\mathbb{E}_{\tilde{\mathbf{x}}\sim p(\mathbf{x}|\boldsymbol{\theta})}$ to a differentiable expectations with respect to the variational program $\mathbb{E}_{\tilde{\mathbf{x}}\sim q(\mathbf{x}|\boldsymbol{\psi})}$. Secondly, is the introduction of a novel form of variational inference that works in a likelihood-free setting. Thirdly, the AVI-MC algorithm can be viewed as a form of empirical Bayes where the prior is optimized based on the data.

Perhaps the closest to our work is [26], which uses Wasserstein distance both to find point estimates $\boldsymbol{\theta}^*$ and as a part of a rejection sampler in an ABC-like setup (as opposed to variational inference). They emphasize that this approach allows one to eliminate the summary statistics typically used in ABC and calculate the Wasserstein distance explicitly, without making use of the Kantorovich-Rubinstein duality and of a critic d. For high-dimensional data, they note that this is computationally expensive and introduce an approximation based on projection of the data onto Hilbert space-filling curves. Their Remark 5.1 points out that [22] proposed an approximation of the gradient of an entropy-regularized Wasserstein distance, which uses a similar duality. They note that "Unfortunately, it is not applicable in the setting of purely generative models, as it involves point-wise evaluations of the derivative of the log-likelihood." Thus, our contribution is to provide gradients of an approximate MWE by taking expectations with the variational program $q(\mathbf{x}|\boldsymbol{\psi})$. This paired with the critic Kantorovich-Rubinstein dual formulation of the Wasserstein distance allows us to work in high dimensions without summary statistics and optimize $\boldsymbol{\psi}$ with stochastic gradient descent. Additionally, our our AVO-EB procedure avoids the inefficiencies of their Wasserstein rejection sampler incurred from using the ABC-likelihood.

As reviewed in [10], likelihood-free inference is intimately tied to a class of algorithms that can be framed as density estimation-by-comparison. In most cases, these inference algorithms are formulated as an iterative two-step process where the model distribution is first compared to the true data distribution and then updated to make it more comparable to the latter. Relevant work in this direction includes [6, 31, 34–37].

This work has also many connections to work on variational inference, in which the goal is to optimize the recognition model $q(\mathbf{z}, \boldsymbol{\theta}|\boldsymbol{\psi})$ so that it is close to the true posterior $p(\mathbf{z}, \boldsymbol{\theta}|\mathbf{x})$. There have been efforts to extend variational inference to intractable likelihoods; however, they require restrictive assumptions. In [19], the authors consider Variational Bayes with an Intractable Likelihood (VBIL). In that approach "the only requirement is that the intractable likelihood can be estimated unbiasedly." In the case of simulators, they propose to use the ABC-likelihood with an ϵ kernel. The ABC likelihood is only unbiased as $\epsilon > 0$, thus this method inherits the drawbacks of the ABC-likelihood including the choice of summary statistics and the inefficiency in evaluating the ABC likelihood for high-dimensional data and small ϵ.

In [38], the authors introduce Hamiltonian ABC which

applies in the likelihood-free setting, and they estimate gradients with respect to $\boldsymbol{\theta}$ through finite differences from multiple forward passes of the simulator with variance reduction strategies based on controlling the source of randomness used for the latent variable $\mathbf{z}$.

Lastly, we make a connection to Operator Variational Inference (OPVI) [24], which is a generalization of variational inference formulated as the following optimization problem

$$\boldsymbol{\lambda}^* = \inf_{\boldsymbol{\lambda}} \sup_{\boldsymbol{\phi}} \mathbb{E}_{\mathbf{z}\sim q(\mathbf{z}|\boldsymbol{\lambda})}[(O^{p,q} f_{\boldsymbol{\phi}})] \,. \tag{17}$$

In traditional VI with the KL distance, this corresponds to $(O^{p,q} f) = \log q(\mathbf{z}) - \log p(\mathbf{z}|\mathbf{x}) \, \forall f \in \mathcal{F}$. AVO can be cast into a similar form with expectations over $\mathbf{x}$ instead of $\mathbf{z}$ and

$$\boldsymbol{\psi}^* = \inf_{\boldsymbol{\psi}} \sup_{\boldsymbol{\phi}} \mathbb{E}_{\mathbf{x}\sim q(\mathbf{x}|\boldsymbol{\psi})}[(O^{p_r, p_{\boldsymbol{\psi}}} d_{\boldsymbol{\phi}})] \tag{18}$$

$$= \inf_{\boldsymbol{\psi}} \sup_{\boldsymbol{\phi}} \mathbb{E}_{\mathbf{x}\sim p_r(\mathbf{x})}[d(\mathbf{x})] - \mathbb{E}_{\mathbf{x}\sim q(\mathbf{x}|\boldsymbol{\psi})}[d(\mathbf{x})] \,. \tag{19}$$

Rewriting Eqn. 18 as Eqn. 19 is possible through importance sampling, corresponding to an implicit form of the operator

$$(O^{p,q} d) = \left(\frac{p_r(\mathbf{x})}{q(\mathbf{x}|\boldsymbol{\psi})} - 1 \right) d(\mathbf{x}) \,, \tag{20}$$

which reinforces the link to density ratio estimation and inference with implicit models.

VII. SUMMARY

In this note, we develop likelihood-free inference algorithms for non-differentiable, implicit generative models. The algorithms combine adversarial training with variational optimization to minimize variational upper bounds on the otherwise non-differentiable adversarial objectives. The AVO-EB algorithm enables empirical Bayesian through variational inference in the likelihood free setting. This approach does not incur the inefficiencies of an ABC-like rejection sampler. The AVO-MWE algorithm provides point estimates for the generative model, which asymptotically correspond to the data generating parameter when the model is well-specified. Both algorithms work on continuous or discrete observed data.

Preliminary results on toy problems with discrete and continuous data validate the proposed method. While the obtained results are encouraging, the complete validation of the method remains to be carried out in real conditions on a full fledged scientific simulator – which we plan to achieve for a next version of this work. In terms of method, several components need to be further investigated. First, we need to better study the interplay between the entropy penalty and the adversarial objectives. Second, we should better understand the dynamics of the optimization procedure, in particular when combined with momentum-based optimizers like Adam. Third, we need to consider whether less noisy estimates of the gradients $\nabla_{\boldsymbol{\psi}} U_g$ can be computed.

ACKNOWLEDGMENTS

We would like to thank Lukas Heinrich for helpful comments regarding the electron–positron annihilation simulation.

GL and KL are both supported through NSF ACI-1450310, additionally KC is supported through PHY-1505463 and PHY-1205376.

[1] Mark A Beaumont, Wenyang Zhang, and David J Balding. Approximate bayesian computation in population genetics. *Genetics*, 162(4):2025–2035, 2002.

[2] Paul Marjoram, John Molitor, Vincent Plagnol, and Simon Tavaré. Markov chain monte carlo without likelihoods. *Proceedings of the National Academy of Sciences*, 100(26):15324–15328, 2003.

[3] Scott A Sisson, Yanan Fan, and Mark M Tanaka. Sequential monte carlo without likelihoods. *Proceedings of the National Academy of Sciences*, 104(6):1760–1765, 2007.

[4] Scott A Sisson and Yanan Fan. *Likelihood-free MCMC*. Chapman & Hall/CRC, New York.[839], 2011.

[5] Jean-Michel Marin, Pierre Pudlo, Christian P Robert, and Robin J Ryder. Approximate bayesian computational methods. *Statistics and Computing*, pages 1–14, 2012.

[6] Kyle Cranmer, Juan Pavez, and Gilles Louppe. Approximating likelihood ratios with calibrated discriminative classifiers. *arXiv preprint arXiv:1506.02169*, 2015, 1506.02169.

[7] M. M. Graham and A. J. Storkey. Asymptotically exact inference in differentiable generative models. *ArXiv e-prints*, May 2016, 1605.07826.

[8] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. *CoRR*, abs/1312.6114, 2013.

[9] 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*, pages 2672–2680, 2014.

[10] S. Mohamed and B. Lakshminarayanan. Learning in Implicit Generative Models. *ArXiv e-prints*, October 2016, 1610.03483.

[11] D. Wierstra, T. Schaul, T. Glasmachers, Y. Sun, and J. Schmidhuber. Natural Evolution Strategies. *ArXiv e-prints*, June 2011, 1106.4487.

[12] J. Staines and D. Barber. Variational Optimization. *ArXiv e-prints*, December 2012, 1212.4507.

[13] M. Arjovsky and L. Bottou. Towards Principled Methods for Training Generative Adversarial Networks. *ArXiv e-prints*, January 2017, 1701.04862.

[14] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein GAN. *ArXiv e-prints*, January 2017, 1701.07875.

[15] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville. Improved Training of Wasserstein GANs. *ArXiv e-prints*, March 2017, 1704.00028.

[16] J Staines and D Barber. Optimization by variational bounding. In *ESANN 2013 proceedings, 21st European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, pages 473–478, 2013.

[17] D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. *ArXiv e-prints*, December 2014, 1412.6980.

[18] Rajesh Ranganath, Sean Gerrish, and David Blei. Black box variational inference. In *Artificial Intelligence and Statistics*, pages 814–822, 2014.

[19] Minh-Ngoc Tran, David J Nott, and Robert Kohn. Variational bayes with intractable likelihood. *Journal of Computational and Graphical Statistics*, (just-accepted), 2017, 1503.08621.

[20] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in neural information processing systems*, pages 2292–2300, 2013.

[21] Aude Genevay, Marco Cuturi, Gabriel Peyré, and Francis Bach. Stochastic optimization for large-scale optimal transport. In *Advances in Neural Information Processing Systems*, pages 3440–3448, 2016.

[22] Grégoire Montavon, Klaus-Robert Müller, and Marco Cuturi. Wasserstein training of restricted boltzmann machines. In *Advances in Neural Information Processing Systems*, pages 3718–3726, 2016.

[23] Fernando Pérez-Cruz. Estimation of information theoretic measures for continuous random variables. In *Advances in neural information processing systems*, pages 1257–1264, 2009.

[24] R. Ranganath, J. Altosaar, D. Tran, and D. M. Blei. Operator Variational Inference. *ArXiv e-prints*, October 2016, 1610.09033.

[25] George EP Box. Sampling and bayes' inference in scientific modelling and robustness. *Journal of the Royal Statistical Society. Series A (General)*, pages 383–430, 1980.

[26] Espen Bernton, Pierre E Jacob, Mathieu Gerber, and Christian P Robert. Inference in generative models using the wasserstein distance. *arXiv preprint arXiv:1701.05146*, 2017.

[27] Johan Alwall, Michel Herquet, Fabio Maltoni, Olivier Mattelaer, and Tim Stelzer. MadGraph 5 : Going Beyond. *JHEP*, 06:128, 2011, 1106.0522.

[28] S. Agostinelli et al. GEANT4: A Simulation toolkit. *Nucl. Instrum. Meth.*, A506:250–303, 2003.

[29] Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Alex Lamb, Martin Arjovsky, Olivier Mastropietro, and Aaron Courville. Adversarially learned inference. *arXiv preprint arXiv:1606.00704*, 2016.

[30] Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. *arXiv preprint arXiv:1605.09782*, 2016.

[31] Mihaela Rosca, Balaji Lakshminarayanan, David Warde-Farley, and Shakir Mohamed. Variational approaches for auto-encoding generative adversarial networks. *arXiv preprint arXiv:1706.04987*, 2017.

[32] Lars M. Mescheder, Sebastian Nowozin, and Andreas Geiger. Adversarial variational bayes: Unifying variational autoencoders and generative adversarial networks. *CoRR*, abs/1701.04722, 2017.

[33] F. Huszár. Variational Inference using Implicit Distributions. *ArXiv e-prints*, February 2017, 1702.08235.

[34] Michael U Gutmann and Aapo Hyvärinen. Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. *Journal of Machine Learning Research*, 13(Feb):307–361, 2012.

[35] K Cranmer, J Pavez, G Louppe, and WK Brooks. Experiments using machine learning to approximate likelihood ratios for mixture models. In *Journal of Physics: Conference Series*, volume 762, page 012034. IOP Publishing, 2016.

[36] R. Dutta, J. Corander, S. Kaski, and M. U. Gutmann. Likelihood-free inference by ratio estimation. *ArXiv e-prints*, November 2016, 1611.10242.

[37] Michael U Gutmann, Ritabrata Dutta, Samuel Kaski, and Jukka Corander. Likelihood-free inference via classification. *Statistics and Computing*, pages 1–15, 2017.

[38] Edward Meeds, Robert Leenders, and Max Welling. Hamiltonian abc. *arXiv preprint arXiv:1503.01916*, 2015.