

# Adversarial Variational Optimization of Non-Differentiable Simulators

Gilles Louppe<sup>1</sup> and Kyle Cranmer<sup>1</sup>

<sup>1</sup>New York University

In this note, ... [GL: todo.]

## I. INTRODUCTION

[GL: Prescribed vs. implicit. See case of non-diff models in Balaji et al.]

## II. PROBLEM STATEMENT

We consider a family of parameterized densities  $p_\theta(\mathbf{x})$  defined implicitly through the simulation of a stochastic generative process, where  $\mathbf{x} \in \mathbb{R}^d$  is the data and  $\theta$  are the parameters of interest. The simulation may involve some complicated latent process, such that

$$p_\theta(\mathbf{x}) = \int p_\theta(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z} \quad (1)$$

where  $\mathbf{z} \in \mathbb{R}^m$  is a latent variable providing an external source of randomness.

We assume that we already have an accurate simulation of the stochastic generative process that defines  $p_\theta(\mathbf{x}|\mathbf{z})$ , as specified through a deterministic function  $g(\cdot; \theta) : \mathbb{R}^m \rightarrow \mathbb{R}^d$ . That is

$$p_\theta(\mathbf{x}) = \frac{\partial}{\partial x_1} \dots \frac{\partial}{\partial x_d} \int_{\{\mathbf{z}: g(\mathbf{z}; \theta) \leq \mathbf{x}\}} p(\mathbf{z})d\mathbf{z}. \quad (2)$$

The simulator  $g$  is assumed to be a non-invertible function, that can only be used to generate data in forward mode. For this reason, evaluating the integral in Eqn. 2 is intractable. Importantly, and as increasingly found in science, we consider the additional constraint that  $g$  is a non-differentiable model, e.g. when specified as a computer program.

Given some observed data  $\{\mathbf{x}_i | i = 1, \dots, N\}$  drawn from the (unknown) true distribution  $p_r$ , our goal is the inference of the parameters of interest  $\theta^*$  that minimize the divergence between  $p_r$  and the modeled data distribution  $p_\theta$  induced by  $g(\cdot; \theta)$  over  $\mathbf{z}$ . That is,

$$\theta^* = \arg \min_{\theta} \rho(p_r, p_\theta), \quad (3)$$

where  $\rho$  is some distance or divergence.

## III. BACKGROUND

### A. Generative adversarial networks

Generative adversarial networks (GANs) were first proposed by [4] as a way to build an implicit generative model capable of producing samples from random

noise  $\mathbf{z}$ . More specifically, a generative model  $g(\cdot; \theta)$  is pit against an adversarial classifier  $d(\cdot; \phi) : \mathbb{R}^d \rightarrow [0, 1]$  with parameters  $\phi$  and whose antagonistic objective is to recognize real data  $\mathbf{x}$  from generated data  $g(\mathbf{z}; \theta)$ . Both models  $g$  and  $d$  are trained simultaneously, in such a way that  $g$  learns to maximally confuse 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 amounts to minimizing the Jensen-Shannon divergence  $\text{JSD}(p_r \parallel p_\theta)$  between  $p_r$  and  $p_\theta$ .

As thoroughly explored in [1], 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 [2] reformulate the adversarial setup in order to minimize the Wasserstein-1 distance  $W(p_r, p_\theta)$  by replacing the adversarial classifier with a 1-Lipschitz adversarial critic  $d(\cdot; \phi) : \mathbb{R}^d \rightarrow \mathbb{R}$ . Under the WGAN-GP formulation of [5] for stabilizing the optimization procedure, training  $d$  and  $g$  results in alternating gradient updates on  $\phi$  and  $\theta$  in order to respectively minimize

$$\begin{aligned} \mathcal{L}_d &= \mathbb{E}_{\tilde{\mathbf{x}} \sim p_\theta} [d(\tilde{\mathbf{x}}; \phi)] - \mathbb{E}_{\mathbf{x} \sim p_r} [d(\mathbf{x}; \phi)] \\ &\quad + \lambda \mathbb{E}_{\tilde{\mathbf{x}} \sim p_\theta} [(\|\nabla_{\tilde{\mathbf{x}}} d(\tilde{\mathbf{x}}; \phi)\|_2 - 1)^2] \end{aligned} \quad (4)$$

$$\mathcal{L}_g = -\mathbb{E}_{\tilde{\mathbf{x}} \sim p_\theta} [d(\tilde{\mathbf{x}}; \phi)] \quad (5)$$

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

### B. Variational optimization

Following [6], variational optimization (VO) (also known as the search gradient algorithm [7]) is a general optimization technique that can be used to form a differentiable bound on the optima of a non-differentiable function. Given a function  $f$  to minimize, VO is based on the simple fact that

$$\min_{\mathbf{c} \in \mathcal{C}} f(\mathbf{c}) \leq \mathbb{E}_{\mathbf{c} \sim q_\psi(\mathbf{c})} [f(\mathbf{c})] = U(\psi), \quad (6)$$

where  $q_\psi$  is a proposal distribution with parameters  $\psi$  over input values  $\mathbf{c}$ . 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  $\psi$  can be updated to place its mass arbitrarily tight around the optimum  $\mathbf{c}^* = \min_{\mathbf{c} \in \mathcal{C}} f(\mathbf{c})$ .

Under mild restrictions outlined in [6], the bound  $U(\psi)$  is differentiable, and using the log-likelihood trick it comes:

$$\begin{aligned}
 \nabla_{\psi} U(\psi) &= \nabla_{\psi} \int f(\mathbf{c}) q_{\psi}(\mathbf{c}) d\mathbf{c} \\
 &= \int f(\mathbf{c}) \nabla_{\psi} q_{\psi}(\mathbf{c}) d\mathbf{c} \\
 &= \int [f(\mathbf{c}) \nabla_{\psi} \log q_{\psi}(\mathbf{c})] q_{\psi}(\mathbf{c}) d\mathbf{c} \\
 &= \mathbb{E}_{\mathbf{c} \sim q_{\psi}(\mathbf{c})} [f(\mathbf{c}) \nabla_{\psi} \log q_{\psi}(\mathbf{c})] \quad (7)
 \end{aligned}$$

Effectively, this means that provided that the score function  $\nabla_{\psi} \log q_{\psi}(\mathbf{c})$  of the proposal is known and that one can evaluate  $f(\mathbf{c})$  for any  $\mathbf{c}$ , then one can construct empirical estimates of Eqn. 7, which can in turn be used to perform stochastic gradient descent (or a variant thereof) in order to minimize  $U(\psi)$ .

#### IV. ADVERSARIAL VARIATIONAL OPTIMIZATION

#### V. EXPERIMENTS

##### A. Toy problem

##### B. Physics example

#### VI. RELATED WORKS

[GL: Implicit generative models.] [GL: ABC.] [GL: carl [3].] [GL: Wood's papers.] [GL: CMA-ES.]

#### VII. SUMMARY

#### ACKNOWLEDGMENTS

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

- 
- [1] ARJOVSKY, M., AND BOTTOU, L. Towards Principled Methods for Training Generative Adversarial Networks. *ArXiv e-prints* (Jan. 2017).
  - [2] ARJOVSKY, M., CHINTALA, S., AND BOTTOU, L. Wasserstein GAN. *ArXiv e-prints* (Jan. 2017).
  - [3] CRANMER, K., PAVEZ, J., AND LOUPPE, G. Approximating likelihood ratios with calibrated discriminative classifiers. *arXiv preprint arXiv:1506.02169* (2015).
  - [4] GOODFELLOW, I., POUGET-ABADIE, J., MIRZA, M., XU, B., WARDE-FARLEY, D., OZAIR, S., COURVILLE, A., AND BENGIO, Y. Generative adversarial nets. In *Advances in Neural Information Processing Systems* (2014), pp. 2672–2680.
  - [5] GULRAJANI, I., AHMED, F., ARJOVSKY, M., DUMOULIN, V., AND COURVILLE, A. Improved Training of Wasserstein GANs. *ArXiv e-prints* (Mar. 2017).
  - [6] STAINES, J., AND BARBER, D. Variational Optimization. *ArXiv e-prints* (Dec. 2012).
  - [7] WIERSTRA, D., SCHAUL, T., GLASMACHERS, T., SUN, Y., AND SCHMIDHUBER, J. Natural Evolution Strategies. *ArXiv e-prints* (June 2011).