A-NICE-MC: Adversarial Training for MCMC

Jiaming Song, Shengjia Zhao and Stefano Ermon

Computer Science Department, Stanford University

Introduction

Setting: Learn the **transition operator** for a Markov chain.

Goals:
1. Match **its stationary distribution** to a **target distribution**.
2. Learn a operator for **efficient MCMC** inference.

In the case of **general Markov chains**, we propose a novel **adversarial training procedure** that

- avoids sampling from the stationary distribution directly
- capable of reaching the target distribution asymptotically

The learned Markov chain:

- can start from **random noise**
- can be trained in a **likelihood free** way
- **does not** require detailed balance

In the case of **MCMC**, we propose **A-NICE-MC**, which combines **the guarantees of MCMC** and **the expressiveness of neural networks**.

- We introduce **a NICE proposal** to increase the acceptance rate for neural network proposals
- A bootstrap procedure is used for **end-to-end training**
- Training with a pairwise discriminator can **increase ESS**.

Problem Setup

We consider two settings:

- We have **direct samples** from the target distribution. (Markov chains)
- We do not have direct access to samples, but have **an analytical expression** of the energy function for the target. (Markov Chain Monte Carlo)

Paper, Code and Website

Code for reproducing the experiments are available at https://github.com/ermongroup/a-nice-mc

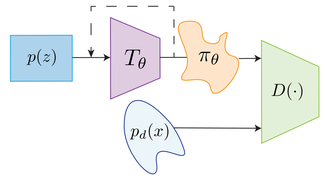
The full-length paper is at https://arxiv.org/abs/1706.07561

Website: http://tsong.me/a-nice-mc

Adversarial Training for Markov Chains

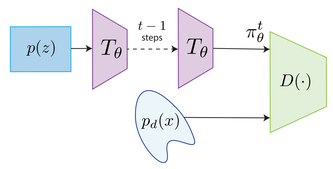
For a Markov chain, it is difficult to compute the marginal likelihood but easy to sample from it, so we consider **adversarial training**, which is **a likelihood-free method** that trains deep generative models using a two player game between a **discriminator** D and a **generator** G.

This cannot be done directly for π_θ since it would require running the chain until convergence, resulting in optimization difficulties.

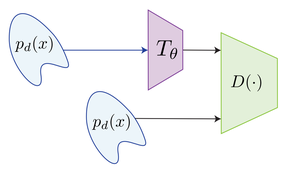


We tackle this problem by splitting the objective into two parts.

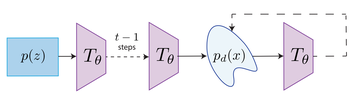
- From a noise distribution, the model should reach the data distribution in T steps.
  (when outside the data distribution, move closer)



- From a data distribution, the model should remain in the data distribution after 1 step.
  (when within the data distribution, remain there)



The resulting generative process is illustrated as follows:



Adversarial Training for MCMC

To design an efficient transition operator for MCMC need to consider the following three aspects:

Satisfying detailed balance, which requires the usage of Metropolis-Hastings (**non-differentiable**).

Training without direct samples, which requires obtaining samples through a **bootstrap procedure**.

High Effective Sample Size (ESS), which requires samples to have **low autocorrelation**.

A NICE Proposal for Detailed Balance

We consider a type of neural network called NICE[1].
NICE learns a **reversible**, **volume preserving** bijection.

A NICE Proposal for $p(x)$ with **any NICE network** $f(x, v)$, where v is an auxiliary variable:

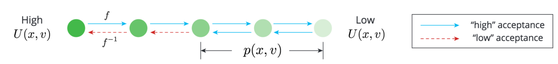
- Randomly sample $v \sim p(v)$
- For probability 0.5, take a "forward step":
$$x', v' = f(x, v)$$
- For probability 0.5, take a "backward step":
$$x', v' = f^{-1}(x, v)$$

We can prove that $p(x', v'|x, v) = p(x, v|x', v')$ for all x, v, x', v'.

[1] Non-linear Independent Components Estimation, Dinh et al.

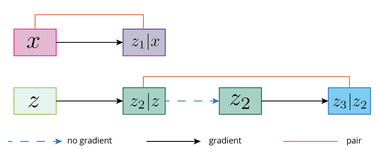
Training with Bootstrap

The transition is non-differentiable due to Metropolis-Hastings. Instead of using score based gradient estimators, we can **simply train the forward mapping** $f(x, v)$ such that it generates the stationary distribution.



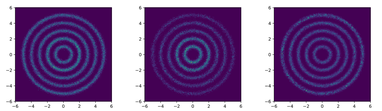
Increase ESS with Pairwise Discriminator

We can increase ESS during training with a pairwise discriminator, which would decrease autocorrelation between samples.



Experiments on Energy Functions

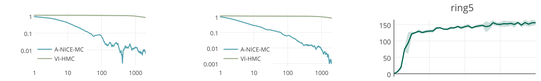
When the energy functions have multiple distinct modes, it becomes difficult for HMC to move efficiently across modes. A-NICE-MC can overcome this easily through the NICE proposal. We use an example with 5 distinct ring shaped modes.



(a) True (b) HMC (c) A-NICE-MC

Figure: Scatter plots for samples.

A-NICE-MC estimates the distribution better than HMC.



(a) $\mathbb{E}[\sqrt{x_1^2 + x_2^2}]$ (b) $\mathrm{Std}[\sqrt{x_1^2 + x_2^2}]$ (c) ESS w/ Training

Figure: (a-b) Mean absolute estimation error of statistics w/ # of samples. (c) ESS w/ # of Training Iterations.

Experiments on Logistic Regression

The posterior for logistic regression have a single mode, which makes HMC a strong baseline.

Although HMC can have higher ESS by carefully tuning the step size, A-NICE-MC outperforms HMC by ESS/s since A-NICE-MC proposal is cheap.

Table: ESS for Bayesian logistic regression tasks.

ESS	A-NICE-MC	HMC
german	926.49	**2178.00**
heart	1251.16	**5000.00**
australian	1015.75	**1345.82**

Table: ESS per second for Bayesian logistic regression tasks.

ESS/s	A-NICE-MC	HMC
german	**1289.03**	216.17
heart	**3204.00**	1005.03
australian	**1857.37**	289.11

Conclusion

We present an efficient adversarial training framework for learning the transition operator to perform Markov Chain Monte Carlo. The trained operator have desired properties, such as higher ESS and ESS/s than traditional approaches such as HMC. Our hope is that these ideas will allow us to bridge the gap between MCMC and neural network function approximators, similarly to what "black-box techniques" did in the context of variational inference.