

Monte-Carlo Gradient Estimation for Machine Learning

Mohamed, Rosca, Figurnov, Minh

Sean, Paco, Michael

University College London

Machine Learning Seminar Presentation
March 6, 2023

Presentation Overview

- 1 Introduction
- 2 Key Targets
- 3 Score Function Estimation
- 4 Pathwise Estimator
- 5 Measure-valued Estimator
- 6 Variance Reduction
- 7 Intuitive Case Study

Introduction

How do we compute the gradient of function expectations?

$$\mathcal{F}(\boldsymbol{\theta}) := \int d\mathbf{x} P(\mathbf{x}; \boldsymbol{\theta}) f(\mathbf{x}; \boldsymbol{\phi}) = \mathbb{E}_{P(\mathbf{x}; \boldsymbol{\theta})}[f(\mathbf{x}; \boldsymbol{\phi})] \quad (1)$$

$$\boldsymbol{\eta} := \nabla_{\boldsymbol{\theta}} \mathcal{F}(\boldsymbol{\theta}) = \nabla_{\boldsymbol{\theta}} \mathbb{E}_{P(\mathbf{x}; \boldsymbol{\theta})}[f(\mathbf{x}; \boldsymbol{\phi})] \quad (2)$$

We estimate it!

Key Targets

- **Consistency** ✓

The estimator, η , gets better with more samples.

- **Unbiasdness**

$\nabla \mathbb{E}_{P(x;\theta)}[\bar{\mathcal{F}}_N] = \nabla \mathbb{E}_{P(x;\theta)}[f(x; \phi)]$. This ensures that we have an accurate estimator.

- **Minimum Variance**

The more precise an estimate given a constant number of samples, N , the better.

- **Computational Efficiency**

Blazingly Fast!

Score Function Estimators

$$\boldsymbol{\eta} = \nabla_{\boldsymbol{\theta}} \mathbb{E}_{P(\mathbf{x}; \boldsymbol{\theta})} [f(\mathbf{x}; \phi)] \quad (3)$$

$$= \int P(\mathbf{x}; \boldsymbol{\theta}) \frac{\nabla_{\boldsymbol{\theta}} P(\mathbf{x}; \boldsymbol{\theta})}{P(\mathbf{x}; \boldsymbol{\theta})} f(\mathbf{x}; \phi) d\mathbf{x} \quad (4)$$

$$= \int P(\mathbf{x}; \boldsymbol{\theta}) f(\mathbf{x}; \phi) \nabla_{\boldsymbol{\theta}} \log(P(\mathbf{x}; \boldsymbol{\theta})) d\mathbf{x}, \quad (5)$$

$$\approx \bar{\boldsymbol{\eta}}_N = \frac{1}{N} \sum_{n=1}^N f(\hat{\mathbf{x}}^{(n)}; \phi) \nabla_{\boldsymbol{\theta}} \log(P(\hat{\mathbf{x}}^{(n)}; \boldsymbol{\theta})). \quad (6)$$

- **Idea:** Swap the order of integration and differentiation, then use the Monte-Carlo estimate.

Consideration of SFE

- **Unbiasedness** ✓: The estimator is unbiased in common ML settings.
- **Minimum Variance** ✗: The estimator is the highest variance for a constant number of samples compared with the other two.
- **Computational Efficiency** ✓: Certified Blazingly Fast!
 - $\mathcal{O}(N(D_\theta + L_{f(x;\phi)}))$
 - An estimate can be made with a single sample.

Further Considerations:

- Black Box score function allowed.
- Higher order differentials are conceptually simple but can be expensive.

Pathwise Gradient Estimator

$$\hat{x} \sim P(x; \theta) \quad \equiv \quad \hat{x} = g(\hat{\epsilon}, \theta), \quad \hat{\epsilon} \sim p(\epsilon) \quad (7)$$

- **Idea:** Reparameterise expectation using knowledge of the sampling path g and the base distribution $P(\epsilon)$ to swap order of derivative and integral

Pathwise Gradient Estimator

$$\eta = \nabla_{\boldsymbol{\theta}} \mathbb{E}_{P(\mathbf{x}; \boldsymbol{\theta})} [f(\mathbf{x}; \phi)] \quad (8)$$

$$= \nabla_{\boldsymbol{\theta}} \int P(\boldsymbol{\epsilon}) f(g(\boldsymbol{\epsilon}; \boldsymbol{\theta})) d\boldsymbol{\epsilon} \quad (9)$$

$$= \mathbb{E}_{P(\boldsymbol{\epsilon})} [\nabla_{\boldsymbol{\theta}} f(g(\boldsymbol{\epsilon}; \boldsymbol{\theta}))] \quad (10)$$

$$\approx \bar{\eta}_N = \frac{1}{N} \sum_{n=1}^N \nabla_{\boldsymbol{\theta}} f(g(\hat{\boldsymbol{\epsilon}}^{(n)}; \boldsymbol{\theta})); \quad \hat{\boldsymbol{\epsilon}}^{(n)} \sim P(\boldsymbol{\epsilon}) \quad (11)$$

- **Idea:** Reparameterise expectation using knowledge of the sampling path g and the base distribution $P(\boldsymbol{\epsilon})$ to swap order of derivative and integral

Consideration of Pathwise Estimators

- **Unbiasedness** ✗

Function needs to be differentiable

- **Minimum Variance** ✓

Variance bounds are independent of parameter dimensionality

- **Computational Efficiency** ✓

Same computational cost as SFE

- $\mathcal{O}(N(D_\theta + L_{f(x;\phi)}))$
- An estimate can be made with a single sample.

Measure-valued Gradient Estimator

For D dimensional parameters θ , we can write the gradient for the i th parameter θ_i as

$$\eta_i = \nabla_{\theta_i} \mathbb{E}_{P(\mathbf{x}; \theta)} [f(\mathbf{x}; \phi)] \quad (12)$$

$$= \int \nabla_{\theta_i} P(\mathbf{x}; \theta) f(\mathbf{x}; \phi) d\mathbf{x} \quad (13)$$

$$= c_{\theta_i} (\mathbb{E}_{p_i^+(\mathbf{x}; \theta)} [f(\mathbf{x}; \phi)] - \mathbb{E}_{p_i^-(\mathbf{x}; \theta)} [f(\mathbf{x}; \phi)]) \quad (14)$$

$$\approx \bar{\eta}_{i,N} = \frac{c_{\theta_i}}{N} \left(\sum_{n=1}^N f(\dot{\mathbf{x}}^{(n)}) - f(\ddot{\mathbf{x}}^{(n)}) \right); \quad (15)$$

$$\dot{\mathbf{x}}^{(n)} \sim p_i^+(\mathbf{x}; \theta), \ddot{\mathbf{x}}^{(n)} \sim p_i^-(\mathbf{x}; \theta)$$

- **Idea:** Use properties of signed measures to compute gradient by decomposing it into a weighted difference of two expectations

Consideration of Measure-valued Estimators

- **Unbiasedness** ✓: The estimator is unbiased in common ML settings
- **Minimum Variance** ✓: Variance depends on choice of decomposition so can be chosen to be low
- **Computational Efficiency** ✗: Expensive
 - $\mathcal{O}(2ND_{\theta}L_f(\mathbf{x};\phi))$
 - Typically not preferred in a high-dimensional setting

Variance Reduction

We introduce three common methods of variance reduction:

- Large-samples
- Coupling
- Control variates

Variance Reduction

Method 1: Large-samples

Large-samples is the simplest method, which means increasing the sample size, N , if we can.

- Variance shrinks in the order of $\mathcal{O}(\frac{1}{N})$
- Computational cost increases linearly in N
- It serves as a baseline method for other more complex methods

Variance Reduction

Method 2: Coupling

Coupling is designed for estimators that take the form of the difference between two expectations:

$$\eta = \mathbb{E}_{p_1(x)}[f(x)] - \mathbb{E}_{p_2(x)}[f(x)] \quad (16)$$

$$\mathbb{V}_{p_{12}(x_1, x_2)}[\bar{\eta}_{cpl}] = \mathbb{V}_{p_{12}(x_1, x_2)}[f(x_1) - f(x_2)] \quad (17)$$

$$= \mathbb{V}_{p_1(x)}[f(x_1)] + \mathbb{V}_{p_2(x)}[f(x_2)] - 2\text{Cov}_{p_{12}(x_1, x_2)}[f(x_1), f(x_2)] \quad (18)$$

Coupling ensures that the blue term is positive by picking a **Common random number** and using it as a seed when we sample x_1 and x_2 .

Variance Reduction

Method 3: Control Variates

Control Variates is a general method. It reduces variance by constructing a **substitute function**, $\tilde{f}(x)$ with properties:

- $\mathbb{E}(\tilde{f}(x)) = \mathbb{E}(f(x))$
- $\mathbb{V}(\tilde{f}(x)) < \mathbb{V}(f(x))$

Design a function $h(x)$ with known $\mathbb{E}_{p(x;\theta)}(h(x))$, then we have:

$$\tilde{f}(x) = f(x) - \beta \left[\underbrace{h(x)}_{\text{control variate}} - \mathbb{E}_{p(x;\theta)}[h(x)] \right] \quad (19)$$

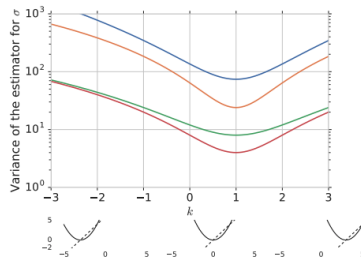
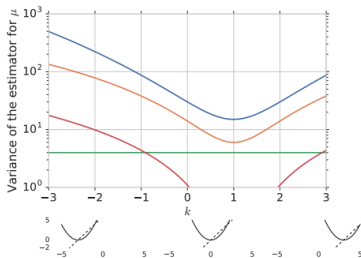
$$\frac{\mathbb{V}[\tilde{f}(x)]}{\mathbb{V}[f(x)]} = \frac{\mathbb{V}\left[f(x) - \beta \left(h(x) - \mathbb{E}_{p(x;\theta)}[h(x)]\right)\right]}{\mathbb{V}[f(x)]} = 1 - \frac{\text{Cov}^2(f, h)}{\mathbb{V}[f]\mathbb{V}[h]} \quad (20)$$

Intuitive Case Study

Quadratic Cost function

$$\eta = \nabla_{\theta} \underbrace{\int \mathcal{N}(x | \mu, \sigma^2) f(x; k) dx}_{\text{Gaussian}}; \quad f \in \left\{ (x - k)^2, \exp\{-kx^2\}, \cos(kx) \right\}$$

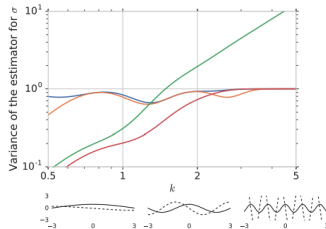
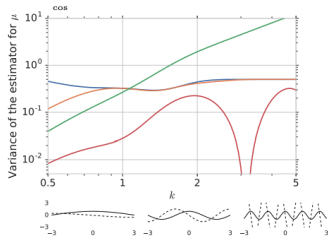
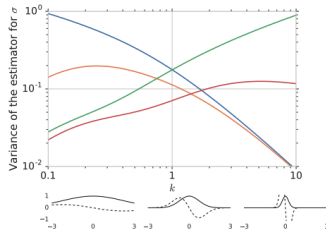
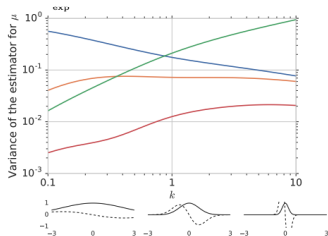
— Score function
 — Score function + variance reduction
 — Pathwise
 — Measure-valued + variance reduction
— Value of the cost
- - - Derivative of the cost



Intuitive Case Study

Exponential and Cosine cost functions

— Score function
 — Score function + variance reduction
 — Pathwise
 — Measure-valued + variance reduction
— Value of the cost
- - - Derivative of the cost



Conclusion

The big takeaway here is that the lack of universal ranking is a general property of gradient estimators.

This table may be helpful in terms of which method to choose:

-	Score Function	Pathwise	Measure-valued
Consistency	✓	✓	✓
Unbiasedness	✓	✗	✓
min. Var.	✗	✓	✓
cmpt. eff.	✓	✓	✗

Variance reduction is always recommended since it indeed reduces variance for any method we choose.