A Latent Variational Framework for Stochastic Optimization

Philippe Casgrain [1,2]

[1]University of Toronto [2]Citadel LLC.

Introduction

- Stochastic optimization algorithms are important tools in machine learning, particularly in the optimization problems that arise from deep learning.
- Stochastic optimization algorithms overcome the computational hurdles of large scale optimization problems **by replacing the exact computation of the gradients with more easily computable statistical samples**.
- Although there is a wide variety of stochastic optimization algorithms (see e.g. [1, 2, 3, 5, 6, 8, 9, 12]), **there is no single theoretical interpretation** with which they can all be understood and compared.
- This paper **constructs a theoretical model for stochastic optimization** using a continuous-time model for these algorithms, similar to those used in [4, 7, 11].
- We use a variational interpretation of optimization, similar to the one used in [10] for deterministic optimization.

Stochastic Optimization

Objective

Minimize $f(x)$ over $x \in \mathbb{R}^d$ using only a stream of noisy gradients $\{g_{t_k}\}$

Canonical Example

Minimize a risk function f using a stream mini-batch gradients g:

$$f(x) = \frac{1}{|\mathfrak{N}|}\sum_{z\in\mathfrak{N}} \ell(x;z) \quad \text{and} \quad g_{t_k} = \frac{1}{|\mathfrak{N}_{t_k}^m|}\sum_{z\in\mathfrak{N}_{t_k}^m} \nabla_x \ell(x_{t_k};z) \tag{1}$$

where $\ell : \mathbb{R}^d \times \mathcal{Z} \to \mathbb{R}$, $\mathfrak{N} := \{z_i \in \mathcal{Z}\ ,\ i = 1,\dots,N\}$ is a set of training points, and $\mathfrak{N}_t^m \subseteq \mathfrak{N}$ are independent, uniformly sampled subsets.

Stochastic Optimization Algorithms

- A stochastic optimization algorithm is an iterative sequence designed to approximate $x^\star := \min_x f(x)$, without ever 'peeking' at f.
- Each step x_{t_k} is determined only by using past observed gradients. An algorithm is therefore defined by a measurable map

$$\{g_{t_{k'}}\}_{k'\le k} \longmapsto x_{t_k} \in \mathbb{R}^d \quad \text{for each } k \in \mathbb{N} \tag{2}$$

which defines each step.

- A stochastic optimization algorithm can therefore be any sequence $\{x_{t_k}\}_{k\in\mathbb{N}}$ which is progressively measurable with respect to $\{g_{t_k}\}_{k\in\mathbb{N}}$.

Continuous-Time Optimization

We can use a **continuous-time model** for optimization.

- We define an algorithm $X = (X_t)_{t\ge 0}$, $X_t \in \mathbb{R}^d$ as a differentiable, continuous-time process. We can interpret this model as an algorithm taking small steps $X_{t+\epsilon} \approx X_t + \epsilon \dot{X}_t$, over the short time intervals.
- As we run the algorithm, we collect observations from a noisy gradient process $(g_t)_{t\ge 0}$. This is a continuous-time interpretation of the r.h.s. of equation (1).
- We define the set of **admissible algorithms** $\mathcal{A}$ as

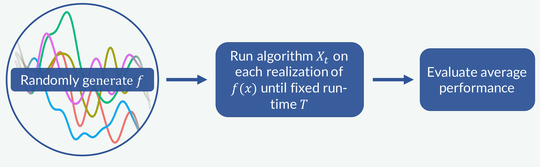
$$\mathcal{A} := \{X \text{ progressively measurable w.r.t } \mathcal{F}_t := \sigma\left((g_u)_{0\le u\le t}\right)\}\ .$$

This is the continuous-time version of the restriction (2).

- $X \in \mathcal{A}$ means that it may only use information from g, and cannot directly observe f nor ∇f.

The Variational Problem

- We evaluate the average performance of an algorithm $X \in \mathcal{A}$ over a randomly generated collection of optimization problems.
- Each problem is generated by a random function $f \mapsto C^2(\mathbb{R}^d)$.



- We define the performance functional $\mathcal{J}$ as

$$\mathcal{J}(X) = \mathbb{E}\Big[\overbrace{f(X_T)}^{\text{Average Performance of Algorithm}} + \underbrace{e^{-\delta_T}\int_0^T \mathcal{L}(t,X_t,\dot{X}_t)\,dt}_{\text{Path-Dependent Regularization Term}}\Big] \tag{3}$$

- $\mathcal{L}$ is the *Bregman Lagrangian*, $\mathcal{L}(t,X,\nu) := e^{\gamma_t}(e^{\alpha_t}D_h(X + e^{-\alpha_t}\nu, X) - e^{\beta_t}f(X))$, where $D_h(X,Y) = h(X) - h(Y) - \langle \nabla h(Y), Y - X\rangle$ is the Bregman divergence. The hyperparameters satisfy $\alpha,\beta,\delta \in C^1(\mathbb{R}^+)$ and $h \in C^2(\mathbb{R}^d)$ and strictly convex.
- Task: Determine the solution to the variational problem

$$X^\star = \arg\min_{X\in\mathcal{A}} \mathcal{J}(X)$$

Main Results

Theorem 1. (Solution to the Variational Problem) An algorithm X is a critical point of $\mathcal{J}$ if and only if the FBSDE

$$\begin{aligned} d\left(\frac{\partial\mathcal{L}}{\partial\nu}\right)_t &= \mathbb{E}\left[\left(\frac{\partial\mathcal{L}}{\partial X}\right)_t \middle| \mathcal{F}_t\right] dt + d\mathcal{M}_t \quad \forall t < T\,, \\ \left(\frac{\partial\mathcal{L}}{\partial\nu}\right)_T &= -e^{\delta_T}\,\mathbb{E}\left[\nabla f(X_T)\middle|\mathcal{F}_T\right] \end{aligned} \tag{4}$$

holds, where $\mathcal{M} = (\mathcal{M}_t)_{0\le t\le T}$ is an $\mathcal{F}_t$-adapted martingale.

Theorem 2. (Rate of Convergence) Assume that f is almost surely convex, $\dot{\gamma}_t = e^{\alpha_t}$ and $\dot{\beta}_t \le e^{\alpha_t}$. Moreover, assume that h is L-Lipschitz smooth and μ-strongly-convex. Define $x^\star$ to be a global minimum of f. If $x^\star$ exists almost surely, the optimizer defined by FBSDE (4) satisfies

$$\mathbb{E}\left[f(X_t) - f(x^\star)\right] = \left(e^{-\beta_t}\max\left\{1\,, e^{-2\gamma_t}\mathbb{E}\left[\,[\mathcal{M}]_t\,\right]\right\}\right)\ , \tag{5}$$

where $[\mathcal{M}]_t$ is the quadratic variation of the process $\mathcal{M}_t$.

Connection to Discrete Algorithms

We recover discrete algorithms with the following steps:

1. Specify a model for $(\nabla f(X_t))_{t\ge 0}$, and $(g_t)_{t\ge 0}$
2. Obtain (or approximate) a solution to the optimality equation (4)
3. Discretize solution over $\mathcal{T} = \{t_0 = 0\,,\ t_{k+1} = t_k + e^{-\alpha_{t_k}} : k \in \mathbb{N}\}$

Stochastic Mirror Descent & Stochastic Gradient Descent

- The Model:
  - Assume that $\nabla f(X_t) = \sigma_f W_t^f$ and $g_t = \nabla f(X_t) + \sigma_e W_t^e$.
  - $\sigma_f, \sigma_e > 0$ and $(W_t^e, W_t^f)_{t\ge 0}$ are independent Brownian motions.
  - This model assumes essentially no structure on gradients.
- **Result:** we obtain the update rule

$$X_{t_{k+1}} = \nabla h^*\left(\nabla h(X_{t_k}) - \tilde{\Phi}_{t_k} g_{t_k}\right)\,,$$

where $\tilde{\Phi}_t$ is a time-dependent learning rate.

- This algorithm corresponds exactly to stochastic mirror descent
- The special case of $h(x) = \frac{1}{2}\|x\|^2$ gives stochastic gradient descent.
- We can interpret result as showing that gradient descent implicitly assumes SGD/mirror descent are optimal when gradients are structureless..

Connection to Discrete Algorithms - *Continued*

Kalman Gradient Descent & Stochastic Momentum Descent

- The Model:
  - Assume that $\nabla f(X_t) = b^\intercal y_t$ where $y_t \in \mathbb{R}^k$ evolves as

$$dy_t = -A y_t\,dt + B dW_t$$

  - Noisy gradients are observed according to

$$dg_t = \nabla f(X_t)\,dt + \sigma dB_t$$

  - $b \in \mathbb{R}^{k\times d}$, $A, B, \sigma \in \mathbb{R}^{k\times k}$ are nonnegative-definite
  - $(W_t, B_t)_{t\ge 0}$ are indep. Brownian Motions of size k and d.
- This model generates the update rule

$$X_{t_{k+1}} = \nabla h^*\left(\nabla h(X_{t_k}) - b^\intercal \tilde{\Phi}_{t_k}\hat{y}_{t_k}\right)\,,$$

where $\hat{y}_t$ is the Kalman filter $b^\intercal\hat{y}_t = \mathbb{E}\left[\nabla f(X_t) \mid \{g_{t_{k'}}\}_{k'\le k}\right]$, and $\tilde{\Phi}_t \in \mathbb{R}^{d\times k}$ is a deterministic learning rate.

- When $h(x) = \frac{1}{2}\|x\|^2$, we recover the Kalman Gradient Descent algorithm from [9].
- Letting $k \to \infty$ with $h(x) = \frac{1}{2}\|x\|^2$, we find that the asymptotic update rule takes the form

$$X_{t_{k+1}} = \Psi_{t_k}^{(0)} X_{t_k} + \Psi_{t_k}^{(1)} g_{t_k}$$

where $\Psi_{t_k}^{(0)}, \Psi_{t_k}^{(1)}$ are time-dependent matrices.

- This rule corresponds exactly to stochastic momentum descent.
- This shows that Kalman Gradient Descent and Stochastic Momentum Descent are in fact related algorithms.
- Thse algorithms are optimal when gradients are expected to decay exponentially in time and stochastic gradient noise is IID.

References

[1] A. Beck and M. Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.

[2] J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011.

[3] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[4] W. Krichene, A. Bayen, and P. L. Bartlett. Accelerated mirror descent in continuous and discrete time. In *Advances in neural information processing systems*, pages 2845–2853, 2015.

[5] A. S. Nemirovsky and D. B. Yudin. Problem complexity and method efficiency in optimization. 1983.

[6] Y. Nesterov. A method for unconstrained convex minimization problem with the rate of convergence o (1/k^ 2). In *Doklady AN USSR*, volume 269, pages 543–547, 1983.

[7] M. Raginsky and J. Bouvrie. Continuous-time stochastic mirror descent on a network: Variance reduction, consensus, convergence. In *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, pages 6793–6800. IEEE, 2012.

[8] H. Robbins and S. Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.

[9] J. Vuckovic. Kalman gradient descent: Adaptive variance reduction in stochastic optimization. *arXiv preprint arXiv:1810.12273*, 2018.

[10] A. Wibisono, A. C. Wilson, and M. I. Jordan. A variational perspective on accelerated methods in optimization. *Proceedings of the National Academy of Sciences*, 113(47):E7351–E7358, 2016.

[11] P. Xu, T. Wang, and Q. Gu. Continuous and discrete-time accelerated stochastic mirror descent for strongly convex functions. In *International Conference on Machine Learning*, pages 5488–5497, 2018.

[12] M. D. Zeiler. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012.