Nonconvex Stochastic Optimization

Saeed Ghadimi

ORFE

ORFE 544

February 21, 2019

Motivation and background

- Problem of interest:

$$\min_{x \in X} f(x).$$

- Assume that $f : \mathbb{R}^d \to \mathbb{R}$ is continuously differentiable and $X \subset \mathbb{R}^d$ is a closed convex set.
- Assume that exact information of f is not available. It can be given as

$$f(x) = \mathbb{E}[F(x, \xi)].$$

- Random vector ξ has distributions P supported on $\Omega \subseteq \mathbb{R}^{\bar{d}}$.

Motivation and background

Why Nonconvex Stochastic Optimization?

- Machine learning: nonconvex loss function or regularization

$$f(x) = \int_{\Omega} l(x, \xi) dP(\xi) + r(x)$$

Motivation and background

Why Nonconvex Stochastic Optimization?

- Machine learning: nonconvex loss function or regularization

$$f(x) = \int_{\Omega} l(x, \xi) dP(\xi) + r(x)$$

- Text analytics
  - A word to vector method to convert the text into numerical features
    - Skip-Gram
    - Continuous bag of words
  - A support vector machine (SVM)/neural network model to learn from the features

Motivation and background

Why Nonconvex Stochastic Optimization?

- Machine learning
- Endogenous uncertainty

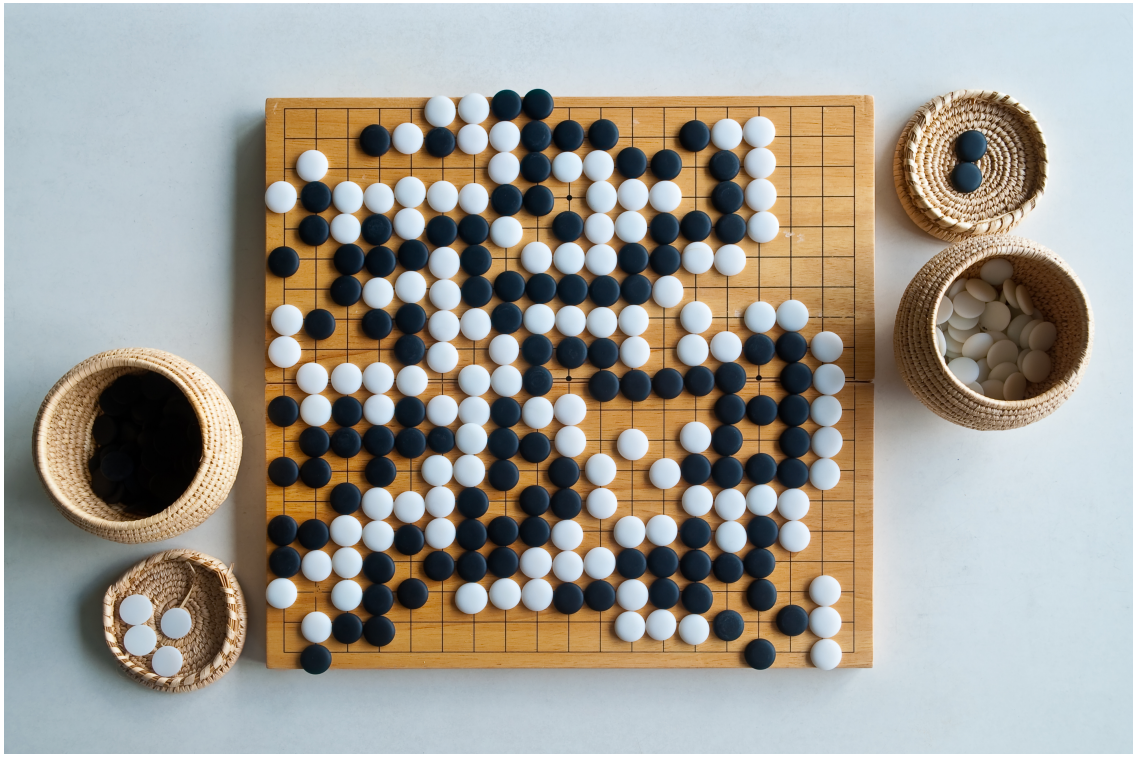
$$f(x) = \int_{\Omega(x)} F(x, \xi) dP_x(\xi).$$

- Nested stochastic optimization
- Zeroth-order stochastic optimization
  - Hyperparameter tuning in machine learning
  - Simulation-based optimization

Motivation and background

Hyperparameter tuning in machine learning

- Google DeepMind has recently used to improve performance of the AlphaGo to beat human professionals.



Motivation and background

Simulation-based optimization

- Noisy function evaluations are available through a simulation process.
- Each simulation can be very expensive.
- Many inventory-related problems in supply chains, finance, energy storage fit into this setting.

Randomized Stochastic Gradient

Basic setting for stochastic optimization:

- Objective function f is equipped with a stochastic first-order oracle ($\mathcal{SFO}$).

Assumption

Given any $x \in X$, the $\mathcal{SFO}$ outputs a *stochastic gradient* $G(x, \xi)$ such that

a) $\mathbb{E}[G(x, \xi)] = \nabla f(x)$,

b) $\mathbb{E}\left[\|G(x, \xi) - \nabla f(x)\|^2\right] \leq \sigma^2$.

Randomized Stochastic Gradient

Unconstrained problems:

Algorithm 1 Randomized Stochastic Gradient(RSG) Method

Input: Initial point x_1, iteration limit N, stepsizes $\{\gamma_k\}_{k\geq 1}$ and probability mass function (PMF) $P_R(\cdot)$ supported on $\{1,\ldots,N\}$.

0. Generate a random integer R according to the PMF P_R.
1. For $k = 1,\ldots,R$:

   Call the stochastic first-order oracle for computing $G(x_k,\xi_k)$ and set

$$x_{k+1} = x_k - \gamma_k G(x_k,\xi_k).$$

Output x_R.

Randomized Stochastic Gradient

- An ϵ-optimal point $\bar{x} \in \mathbb{R}^d : \mathbb{E}[f(\bar{x}) - f(x^*)] \leq \epsilon$.
- No rate of convergence can be provided to find an ϵ-optimal point, when f is nonconvex.
- An ϵ-stationary point $\bar{x} \in \mathbb{R}^d : \mathbb{E}[\|\nabla f(\bar{x})\|] \leq \epsilon$.

Theorem

a) An ϵ-stationary point of the problem (in expectation) can be found while the total number of calls to the $\mathcal{SFO}$ is bounded by $\mathcal{O}\left(\frac{1}{\epsilon^4}\right)$.

b) Recall that this bound is in the order of $\mathcal{O}\left(\frac{1}{\epsilon^2}\right)$ to find an ϵ-optimal point of the problem, when f is convex.

Randomized Stochastic Gradient

Algorithm 2 Two-phase RSG (2-RSG) Method

Input: Initial point x_1, iteration limit N, sample size T, and confidence level $\Lambda \in (0, 1)$.

0. Set $S = \lceil \log 2/\Lambda \rceil$.
1. **Optimization phase:**
   For $s = 1, \ldots, S$

   Call the RSG method with input x_1, iteration limit N, stepsizes $\{\gamma_k\}$ and probability mass function P_R same as in RSG method. Let $\bar{x}_s$ be the output of this procedure.
2. **Post-optimization phase:**
   Choose a solution $\bar{x}^*$ from the candidate list $\{\bar{x}_1, \ldots, \bar{x}_S\}$ such that

$$\|g(\bar{x}^*)\| = \min_{s=1,\ldots,S} \|g(\bar{x}_s)\|, \quad g(\bar{x}_s) := \frac{1}{T} \sum_{k=1}^{T} G(\bar{x}_s, \xi_k).$$

Randomized Stochastic Gradient

- An (ϵ, Λ)-stationary point $\bar{x} \in \mathbb{R}^n : \text{Prob}\{\|\nabla f(\bar{x})\| \leq \epsilon\} \geq 1 - \Lambda$ for some $\Lambda \in (0, 1)$.

Theorem

Under a light-tail assumption on the stochastic gradients, one can show that an (ϵ, Λ)-stationary point of the problem can be found while the total number of calls to the $\mathcal{SFO}$ in the 2-RSG method is bounded by $\mathcal{O}\left(\frac{\log(1/\Lambda)}{\epsilon^4}\right)$.

Randomized Stochastic Gradient

Constrained and composite problems:

- Problem of interest: $\min_{x \in X} \{\phi(x) = f(x) + \mathcal{X}(x)\}$.
- The regularization term $\mathcal{X}(\cdot)$ is convex and possibly nonsmooth.
- Size of the gradient is no longer a good notion of stationary points.
- A new termination criterion "gradient mapping" is employed.
- It plays the gradient role for constrained/nonsmooth problems.

Randomized Stochastic Gradient

Constrained/composite problems:

Algorithm 3 Randomized Stochastic Projected Gradient(RSPG) Method

Input: Initial point $x_1 \in X$, iteration limit N, stepsizes $\{\gamma_k\}_{k\geq 1}$, the positive integer $\{m_k\}_{k\geq 1}$, and probability mass function (PMF) $P_R(\cdot)$ supported on $\{1,\ldots,N\}$.

0. Generate a random integer R according to the PMF P_R.
1. For $k = 1,\ldots,R$:

   Call the $\mathcal{SFO}$ m_k times to obtain $G(x_k,\xi_{k,i})$, $i = 1,\ldots,m_k$, set $G_k = \frac{1}{m_k}\sum_{i=1}^{m_k} G(x_k,\xi_{k,i})$, and compute

   $$x_{k+1} = \arg\min_{u\in X}\left\{\langle G_k, u\rangle + \frac{1}{2\gamma_k}\|u,-x_k\|^2 + \mathcal{X}(u)\right\}.$$

Output x_R.

Randomized Stochastic Gradient

- A two-phase variant of the RSPG method can be also designed.
- Sample complexities of the RSPG and 2-RSPG methods are similar to their counterparts for unconstrained problems.
- If the feasible set X is bounded, one can design another two-phase variant of the RSPG, called 2-RSPG-V method.
- The only difference between the 2-RSPG-V and 2-RSPG methods is that in the optimization phase of the former, the S runs are not independent. In particular, the output of each run is the starting point of the next run.
- While the sample complexity of both methods are the same, the practical performance of the 2-RSPG-V method seems to better than that of the 2-RSPG method.