A Block Coordinate Ascent Algorithm for Mean-Variance Optimization

Tengyang Xie*[1] Bo Liu*[2] Yangyang Xu[3] Mohammad Ghavamzadeh[4] Yinlam Chow[5] Daoming Lyu[2] Daesub Yoon[6] (* equal contribution)

[1]UMass Amherst [2]Auburn University [3]Rensselaer Polytechnic Institute [4]Facebook AI Research [5]Google DeepMind [6]ETRI

Challenges

Risk-sensitive Reinforcement Learning: How to ***manage risk*** (e.g., variance of the cumulative reward) for reinforcement learning?

Abstract:
- Computing an unbiased estimation of policy gradients with variance related risk criteria usually requires ***double sampling*** or ***multi-time-scale*** stochastic approximation algorithm.
- ***Sample complexity*** of existing methods is difficult to analyze.

Our approach: Mean-variance objective function based on its Legendre-Fenchel dual.

Problem Setup

Double sampling issue:
- Mean-variance optimization

$$\max_\theta \quad J(\theta) = \mathbb{E}_{\pi_\theta}[R] \quad \text{s.t.} \quad \mathrm{Var}_{\pi_\theta}(R) \leq \zeta$$

- Lagrangian relaxation procedure

$$\begin{aligned} J_\lambda(\theta) :=& \mathbb{E}_{\pi_\theta}[R] - \lambda\big(\mathrm{Var}_{\pi_\theta}(R) - \zeta\big) \\ =& J(\theta) - \lambda\big(M(\theta) - J(\theta)^2 - \zeta\big) \end{aligned}$$

- Computing stochastic gradient ($M(\theta) := \mathbb{E}_{\pi_\theta}[R^2]$)

$$\begin{aligned} \nabla_\theta J_\lambda(\theta_t) =& \nabla_\theta J(\theta_t) - \lambda \nabla_\theta \mathrm{Var}(R) \\ =& \nabla_\theta J(\theta_t) - \lambda\big(\nabla_\theta M(\theta) - 2J(\theta)\nabla_\theta J(\theta)\big) \end{aligned}$$

- We have ***no*** unbiased estimation of $J(\theta)\nabla_\theta J(\theta)$ with a single trajectory.

Multi-time-scale method[2]:

$$\begin{aligned} \theta_{k+1} =& \theta_k + \alpha_k\Big(R^k(\theta_k) - \lambda g'\left(\widetilde{V}_k - b\right)\left(\left(R^k(\theta_k)\right)^2 - 2R^k(\theta_k)\widetilde{J}_k\right)\Big) z^k(\theta_k) \\ \widetilde{J}_{k+1} =& \widetilde{J}_k + \beta_k\big(R^k(\theta_k) - \widetilde{J}_k\big) \\ \widetilde{V}_{k+1} =& \widetilde{V}_k + \beta_k\left(\left(R^k(\theta_k)\right)^2 - \widetilde{J}_k^2 - \widetilde{V}_k\right) \end{aligned}$$

Red terms make it impossible to analyze sample complexity by existing approaches[1].

Objective: Risk-sensitive reinforcement learning method with ***single-time-scale*** stepsize and provable sample complexity analysis.

Block Coordinate Reformulation

Reformulation using Legendre-Fenchel dual:

$$\begin{aligned} F_\lambda(\theta) :=& \left(J(\theta) + \frac{1}{2\lambda}\right)^2 - M(\theta) \\ =& \max_y \left(2y\Big(J(\theta) + \frac{1}{2\lambda}\Big) - y^2\right) - M(\theta) \end{aligned}$$

New optimization problem (standard nonconvex coordinate ascent problem):

$$\max_{\theta, y} \quad \hat{f}_\lambda(\theta, y) := 2y\Big(J(\theta) + \frac{1}{2\lambda}\Big) - y^2 - M(\theta).$$

Mean-Variance Policy Gradient (MVP)

Algorithm 1 Mean-Variance Policy Gradient (MVP)

Input: Stepsizes $\{\beta_t^\theta\}$ and $\{\beta_t^y\}$, and number of iterations N
  Option I: $\{\beta_t^\theta\}$ and $\{\beta_t^y\}$ satisfy the Robbins-Monro condition
  Option II: β_t^θ and β_t^y are set to be constants
for episode $t = 1, \dots, N$ **do**
  Generate the initial state $s_1 \sim P_0$
  while $s_k \neq s^*$ **do**
    Take the action $a_k \sim \pi_{\theta_t}(a|s_k)$ and observe the reward r_k and next state s_{k+1}
  end while
  Update the parameters

$$\begin{aligned} R_t =& \sum_{k=1}^{\tau_t} r_k \\ \omega_t(\theta_t) =& \sum_{k=1}^{\tau_t} \nabla_\theta \ln \pi_{\theta_t}(a_k|s_k) \\ y_{t+1} =& y_t + \beta_t^y\left(2R_t + \frac{1}{\lambda} - 2y_t\right) \\ \theta_{t+1} =& \theta_t + \beta_t^\theta\left(2y_{t+1}R_t - (R_t)^2\right)\omega_t(\theta_t) \end{aligned}$$

end for
Output $\bar{x}_N$:
  Option I: Set $\bar{x}_N = x_N = [\theta_N, y_N]^\top$
  Option II: Set $\bar{x}_N = x_z = [\theta_z, y_z]^\top$, where z is uniformly drawn from $\{1, 2, \dots, N\}$

Theoretical Analysis of MVP

Finite sample analysis:

Theorem. *Let the output of the MVP Algorithm be $\bar{x}_N$ following Option II. If $\{\beta_t^\theta\}$, $\{\beta_t^y\}$ are constants and satisfies $2\beta_t^{\min} > L(\beta_t^{\max})^2$ for $t = 1, \cdots, N$, we have*

$$\mathbb{E}\left[\|\nabla \hat{f}_\lambda(\bar{x}_N)\|_2^2\right] \leq \frac{\hat{f}_\lambda^* - \hat{f}_\lambda(x_1) + N(\beta_t^{\max})^2 C}{N(\beta_t^{\min} - \frac{L}{2}(\beta_t^{\max})^2)}$$

where $\hat{f}_\lambda^* = \max_x \hat{f}_\lambda(x)$.

Corollary. *The convergence rate of the MVP algorithm with constant stepsizes $\mathcal{O}(1/\sqrt{N})$ implies that the sample complexity $N = \mathcal{O}(1/\varepsilon^2)$ in order to find ε-stationary solution.*

Asymptotic convergence:

Theorem. *Let $\{x_t = (\theta_t, y_t)\}$ be the sequence of the outputs generated by MVP algorithm with Option I. If $\{\beta_t^\theta\}$ and $\{\beta_t^y\}$ are time-diminishing real positive sequences satisfying the Robbins-Monro condition, i.e., $\sum_{t=1}^\infty \beta_t^\theta = \infty$, $\sum_{t=1}^\infty (\beta_t^\theta)^2 < \infty$, $\sum_{t=1}^\infty \beta_t^y = \infty$, and $\sum_{t=1}^\infty (\beta_t^y)^2 < \infty$, then MVP Algorithm will converge such that*

$$\lim_{t \to \infty} \mathbb{E}[\|\nabla \hat{f}_\lambda(x_t)\|_2] = 0.$$

Finite-Sample Analysis of Nonconvex Block Stochastic Gradient (BSG) Algorithms

MVP algorithm belongs to the family of nonconvex BSG algorithm

objective function: $\min_{x \in \mathbb{R}^n} f(x) = \mathbb{E}_\xi[F(x, \xi)]$

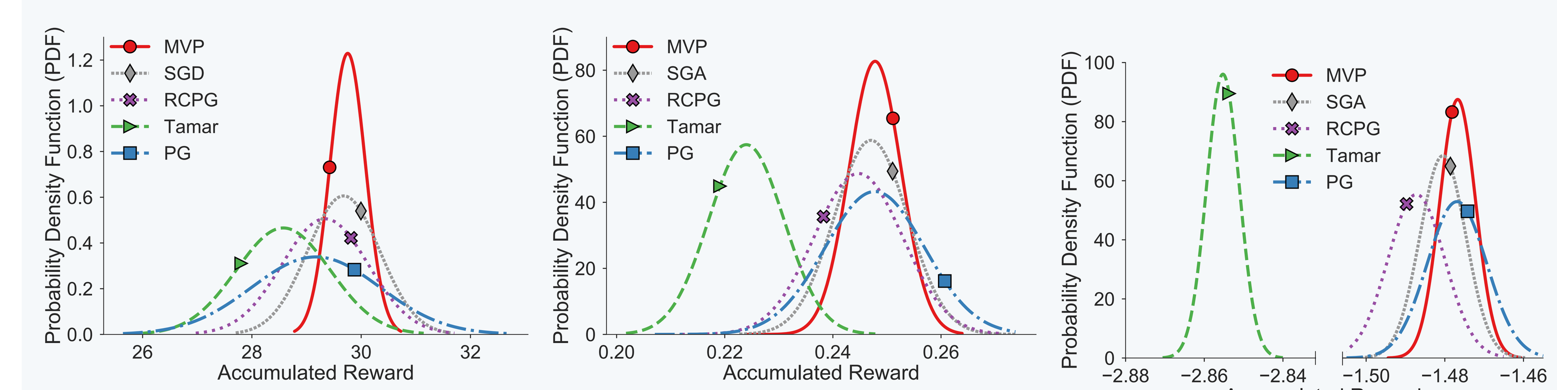
Theorem. *Let the output of the nonconvex BSG algorithm be $\bar{x}_N = x_z$. If stepsizes satisfy $2\beta_t^{\min} > L(\beta_t^{\max})^2$ for $t = 1, \cdots, N$, then we have*

$$\mathbb{E}\left[\|\nabla f(\bar{x}_N)\|_2^2\right] \leq \frac{f(x_1) - f^* + \sum_{t=1}^N (\beta_t^{\max})^2 C_t}{\sum_{t=1}^N (\beta_t^{\min} - \frac{L}{2}(\beta_t^{\max})^2)},$$

where $f^* = \min_x f(x)$. $C_t = (1 - \frac{L}{2}\beta_t^{\max})\sum_{i=1}^b L\sqrt{\sum_{j<i}(G^2 + \sigma^2)} + b\left(AG + \frac{L}{2}\sigma^2\right)$, where G is the gradient bound, L is the Lipschitz constants, σ is the variance bound.

Corollary. *The convergence rate of the nonconvex BSG algorithm with constant stepsizes $\mathcal{O}(1/\sqrt{N})$ implies that the sample complexity $N = \mathcal{O}(1/\varepsilon^2)$ in order to find ε-stationary solution.*

Experimental Study



(a) Portfolio management domain (b) American-style option domain (c) Optimal stopping domain

Figure: Empirical results of the distributions of the return (cumulative rewards) random variable. Note that markers only indicate different methods.

	Portfolio Management		American-style Option		Optimal Stopping	
	Mean	Std	Mean	Std	Mean	Std
MVP	**29.754**	**0.325**	**0.2478**	**0.00482**	**-1.4767**	0.00456
PG	29.170	1.177	0.2477	0.00922	-1.4769	0.00754
Tamar	28.575	0.857	0.2240	0.00694	-2.8553	**0.00415**
SGA	29.679	0.658	0.2470	0.00679	-1.4805	0.00583
RCPG	29.340	0.789	0.2447	0.00819	-1.4872	0.00721

Table: Performance Comparison among Algorithms

References

[1] Gal Dalal, Gugan Thoppe, Balázs Szörényi, and Shie Mannor. Finite sample analysis of two-timescale stochastic approximation with applications to reinforcement learning. In *Proceedings of the 31st Conference On Learning Theory*, pages 1199--1233. PMLR, 06--09 Jul 2018.

[2] Aviv Tamar, Dotan Di Castro, and Shie Mannor. Policy gradients with variance related risk criteria. In *Proceedings of the twenty-ninth international conference on machine learning*, pages 387--396, 2012.