Variance Reduced Policy Evaluation with Smooth Function Approximation

Hoi-To Wai (CUHK), Mingyi Hong (UMN), Zhuoran Yang (Princeton), Zhaoran Wang (Northwestern), Kexin Tang (UMN)

Motivation
- Policy evaluation (PE) evaluates the value function of average reward at a state, given a policy.
- For large state space, nonlinear (and smooth) function approximation is widely used, e.g., neural net.

Aim: Theoretical study of an efficient algorithm for policy evaluation with nonlinear function approx.

Problem Formulation

Discounted MDP: \((S, \mathcal{A}, P, R, \gamma)\)
- \(S\) – state space, \(\mathcal{A}\) – action space.
- \(P^a: S \times S \to \mathbb{R}\), Markov kernel for state transition under action \(a \in \mathcal{A}\).
- \(R(s, a)\) – reward at state \(s\) and under action \(a\).
- \(\gamma \in (0, 1)\) – discount factor.

Policy: \(\pi\) is a conditional probability \(\pi(a|s)\) of choosing action \(a\) under state \(s\).

Goal: given a policy \(\pi\), learn the value function \(V^\pi(s) = \mathbb{E}_\pi \left[ \sum_{k=0}^{\infty} \gamma^k R(s_k, a_k) \right]\) with \(s_0 = s, a_0 \sim \pi(\cdot|s_0)\), \(P^a(s_{k+1} \mid s_k, a_k) = \pi(a_k \mid s_k)\).

PE can be solved by Bellman eq. \(\Rightarrow V^\pi(s) = T^\pi V^\pi(s)\), where for any measurable function \(f : S \to \mathbb{R}\), \(T^\pi f(s) = \mathbb{E}[R(s, a) + \gamma P^a f(s) | a \sim \pi(\cdot|s)]\).

Challenges:
- The state space \(S\) is large (can be infinite).
- State transition probability is unknown.

Remedy: nonlinear function approximation:
- Replace \(V^\pi(s)\) by a parameterized function \(V_\theta(s)\).
- E.g., \(\theta \in \mathbb{R}^d \subset \mathbb{R}^d\) are the weights of a NN.
- Objective: find \(\theta\) to minimize \(J(\theta) := \frac{1}{2} \mathbb{E} \left[ (T^\pi V_\theta(s) - V_\theta(s))^2 \right]\)

Projected Bellman Error Minimization as Primal-dual Optimization

- The function \(V_{\theta}(s) := (\nabla_\theta V_\theta(s) + \nabla_\theta H_{\theta}(s))\) and Hessian \(H_{\theta}(s) := \nabla_\theta^2 V_\theta(s)\).
- Evaluating unbiased stochastic gradient of \(J(\theta)\) is hard, sampling from \(P^a(\cdot)|s_k\) and forming \(C_{\gamma,k}\).
- Define \(C_{\gamma,k} := \mathbb{E}_{P^a(\cdot)|s_k}[(\nabla_\theta V_\theta(s) + \nabla_\theta H_{\theta}(s))|s_k]\), the loss function \(J(\theta)\) admits a Fenchel’s dual reformulation [1]:

\[
J(\theta) = \frac{1}{2} \mathbb{E}_{P^a(\cdot)|s_k} \left[ (T^\pi V_\theta(s) - V_\theta(s))^2 \right] + \frac{1}{2} \mathbb{E}_{P^a(\cdot)|s_k} \left[ (T^\pi V_\theta(s) - V_\theta(s))^2 \right] + \frac{1}{2} \mathbb{E}_{P^a(\cdot)|s_k} \left[ (T^\pi V_\theta(s) - V_\theta(s))^2 \right]
\]

Batch RL setting: observe a trajectory of state-action pairs \((s_1, a_1, s_2, a_2, \ldots, s_m, a_m, s_{m+1})\) generated from \(\pi\),

\[
\min_{\theta \in \Theta} \frac{1}{m} \sum_{k=1}^{m} L(\theta, s_k, w) = \min_{\theta \in \Theta} \frac{1}{m} \sum_{k=1}^{m} \left( \mathbb{E}[R(s_k, a_k) + \gamma V_\theta(s_{k+1}) - V_\theta(s_k)] \right)
\]

Nonconvex Primal-Dual Gradient with Variance Reduction (nPD-VR) Algorithm

- Directly optimizing the finite-sum problem has high complexity \(\Rightarrow\) SGD is fast but slow convergence...
- Philosophy: balance between complexity and speed of convergence \(\Rightarrow\) variance reduction via SAGA [2],

for \(k \geq 1\) do

Select \(i_k, j_k \in [1, \ldots, m]\) uniformly and independently.

Primal-dual gradient update through

\[
\theta^{k+1} = \theta^k - \beta \left( C_{\gamma,k} + \nabla_\theta L_i(\theta, s_k) - \nabla_\theta L_j(\theta, s_k) \right)
\]

Update stored variables as:

\[
\theta_{i_k}^{k+1} = \theta_{i_k}^k, \quad \theta_{j_k}^{k+1} = \theta_{j_k}^k
\]

Main Steps of Proof
- Bound primal-dual updates’ progress on the objective value \(L(\theta^k, w^k)\).
- By carefully controlling the step size, we show

\[
\Omega\left( \min(\alpha, \beta) \right) \sum_{k=0}^{K-1} \mathbb{E}\left[ \left| L(\theta^k, w^k) \right| \right] \leq \mathbb{E}\left[ \left| L(\theta^0, w^0) \right| \right] + \mathbb{E}\left[ \left\| \nabla_\theta L(\theta^0, w^0) \right\| \right]
\]

Involves new technique in controlling the error due to SAGA.
- Green term \(\leq \sum_{k=0}^{K-1} \mathbb{E}\left[ \left| \theta^{k+1} - \theta^k \right| \right] \).
- Selecting the right step size ensures the RHS of (A) is \(O(1)\).
- Using \(K \sim \mathcal{O}(1)\) finishes the proof.

Preliminary Experiments

- Setting: mountaincar dataset \(w = m = 5000\).
- Nonlinear function \(V_\theta(s)\) is parameterized as 2-layer neural network with \(n\) neurons.
- Set constraints as \(\theta \in [0, 1]^d\) and \(w \in [0, 100]^d\).
- Step sizes are \(\alpha = 2 \times 10^{-3}, \beta = 3 \times 10^{-4}\).

Primal-dual SAGA –
- A primal-dual version of non-convex SAGA in [2].
- Update \(w, \theta\) indices \(i_k, j_k\) to ensure unbiasedness.
- \(O(d^2)\) FLOPS per iteration (reduced to \(O(d)\) w/ approx).

Challenges of analysis —
- One-sided non-convexity.
- Algorithm is non-monotone.

Assumptions —
- Strong concavity for \(L\) w.r.t. \(w\).
- Lipschitz cts. gradient for \(L\).
- \((\theta^k, w^k)\) is bounded.

Theorem 1. Choosing step sizes \(\alpha, \beta = O(1/m)\). Let \(K\) be uniformly picked from \([1, \ldots, K]\). It holds that

\[
\mathbb{E}\left[ \left\| \nabla_\theta L(\theta^K, w^K) \right\|^2 \right] \leq \mathbb{E}\left[ \left\| \nabla_\theta L(\theta^0, w^0) \right\|^2 \right] + \frac{K}{m} \mathbb{E}\left[ \left\| \nabla_\theta L(\theta^0, w^0) \right\|^2 \right]
\]

(n = 50 neurons)

(n = 100 neurons)

Compared to plain SGD, nPD-VR converges to a stationary point with less no. of epochs.

Future work – mini-batch design to speed up convergence, improve analysis with projection of \(w\), etc.

References:

(n = 100 neurons)