Towards Understanding Reinforcement Learning from Optimization Perspectives

Shaocong Ma

University of Utah

s.ma@utah.edu

November 12, 2021
Background
- Third-year Ph.D. student in EE at University of Utah.
- M.A. Degree and B.S. Degree in Statistics.

Research

- quantify everything
- as a guiding principle

Optimization

Reinforcement Learning

- motivation examples
- new challenges
1 Challenge 1: Non-Independent Data
   - Reduce the influence of data dependence
   - Classical optimization techniques on dependent data
   - Critical thinking: is data dependence always bad?

2 Challenge 2: Exploration-Exploitation Trade-Off
   - Quantify the error caused by lacking of exploration

3 Reference
Challenges from RL: Non-Independent Data

Dataset in Reinforcement Learning

The data point \((s_t, a_t, r_t, s_{t+1})\) in RL comes from a trajectory:

\[
 s_1, a_1, r_1, s_2, a_2, r_2, \ldots
\]
The data point \((s_t, a_t, r_t, s_{t+1})\) in RL comes from a trajectory:

\[ s_1, a_1, r_1, s_2, a_2, r_2, \ldots \]

\[ \{(s_i, a_i, r_i, s_{i+1})\} \text{ and } \{(s_j, a_j, r_j, s_{j+1})\} \text{ are non-independent!} \]
**Ultimate Goal of RL**  Find a strategy $\pi$ of selecting action to maximize the future return:

$$\max_\pi Q^\pi(s, a) := \mathbb{E}[\sum_{t=1}^{\infty} \gamma^t r_t | s, a]$$

**Deep Q-Learning (DQN) with Target Network** [DeepMind’13]

$$\theta_{k+1} \leftarrow \arg \min_{\theta} \mathbb{E}_{(s,a,r,s') \sim \mu} \| r + \gamma \max_{a'} Q_{\theta_k}(s', a') - Q_{\theta}(s, a) \|^2$$

An optimization problem!

where $\mu$ is the stat. dist. of the stochastic process $\{(s_t, a_t, r_t, s_{t+1})\}$. 
**Ultimate Goal of RL**  Find a strategy $\pi$ of selecting action to maximize the future return:

$$\max_{\pi} Q^\pi(s, a) := \mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^t r_t | s, a\right]$$

**Deep Q-Learning (DQN) with Target Network**  [DeepMind’13]

$$\theta_{k+1} \leftarrow \arg \min_{\theta} \mathbb{E}_{(s, a, r, s') \sim \mu} \left\| r + \gamma \max_{a'} Q_{\theta_k}(s', a') - Q_{\theta}(s, a) \right\|^2$$

An optimization problem!

where $\mu$ is the stat. dist. of the stochastic process $\{(s_t, a_t, r_t, s_{t+1})\}$.

**Key difference:** non-independent data
A general question  Solve the optimization problem

\[
\min_{\mathbf{x}} \mathbb{E}_{\xi \sim \mu} f(\mathbf{x}; \xi)
\]
given a stochastic process \( \{\xi_t\} \). How does it influence the optimization?

- RL applications: (double) Q-learning, Actor-Critic, PPO, and etc.

Existing work [Agarwal’12] With a high-probability,

\[
\mathbb{E}_{\xi \sim \mu} f(\bar{\mathbf{x}}_t; \xi) - \min_{\mathbf{x}} \mathbb{E}_{\xi \sim \mu} f(\mathbf{x}; \xi) \leq O\left(\frac{1}{\sqrt{t}}\right) + O\left(\frac{\sqrt{\tau}}{t} + \phi(\tau)\right),
\]

where \( \phi(\tau) := \sup_k \sup_{A \in \mathcal{F}_k} d_{TV}(\mathbb{P}(\xi_{\tau+k} \in \cdot | A), \mu) \).
1. **Challenge 1: Non-Independent Data**
   - Reduce the influence of data dependence
   - Classical optimization techniques on dependent data
   - Critical thinking: is data dependence always always bad?

2. **Challenge 2: Exploration-Exploitation Trade-Off**
   - Quantify the error caused by lacking of exploration

3. **Reference**
**Question** How can we reduce the influence of data dependence?

**Answer** Just use a large batch size.

**Our work** [ICLR’22 - under review]

<table>
<thead>
<tr>
<th>Data dependence level</th>
<th>$\phi(k)$</th>
<th>SGD</th>
<th>Mini-batch SGD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geometric $\phi$-mixing (Weakly dependent)</td>
<td>$\exp(-k^\theta)$, $\theta &gt; 0$</td>
<td>$O(\epsilon^{-2}(\log \epsilon^{-1})^{\frac{2}{\theta}})$</td>
<td>$O(\epsilon^{-2})$</td>
</tr>
<tr>
<td>Fast algebraic $\phi$-mixing (Medium dependent)</td>
<td>$k^{-\theta}$, $\theta \geq 1$</td>
<td>$O(\epsilon^{-2-\frac{2}{\theta}})$</td>
<td>$\tilde{O}(\epsilon^{-2})$</td>
</tr>
<tr>
<td>Slow algebraic $\phi$-mixing (Highly dependent)</td>
<td>$k^{-\theta}$, $0 &lt; \theta &lt; 1$</td>
<td>$O(\epsilon^{-2-\frac{2}{\theta}})$</td>
<td>$O(\epsilon^{-1-\frac{1}{\theta}})$</td>
</tr>
</tbody>
</table>

How does this idea work?
- Reduce the variance:

  (single) \( \mathbb{E} \| f(x; \xi_t) - \mathbb{E}_{\xi \sim \mu} f(x; \xi) \|^2 \approx O(1) \)

  (mini-batch) \( \mathbb{E} \| \frac{1}{B} \sum_{i=1}^{B} f(x; \xi_{t+i}) - \mathbb{E}_{\xi \sim \mu} f(x; \xi) \|^2 \approx O\left(\frac{1}{B}\right) \)

- Reduce the bias:

  (single) \( \mathbb{E}_{\xi_\tau} f(x; \xi_\tau) - \mathbb{E}_{\xi \sim \mu} f(x; \xi) \approx \phi(\tau) \)

  (mini-batch) \( \frac{1}{B} \sum_{i=1}^{B} \mathbb{E}_{\xi_{\tau+i}} f(x; \xi_{\tau+i}) - \mathbb{E}_{\xi \sim \mu} f(x; \xi) \approx \frac{1}{B} \sum_{i=1}^{B} \phi(\tau + i) \)

- Put them back to [Agarwal’12]:

  \[
  \text{opt. error} \leq O\left(\frac{1}{\sqrt{tB}}\right) + O\left(\sqrt{\frac{\tau}{tB}} + \frac{1}{B} \sum_{i=1}^{B} \phi(i)\right). \\
  \quad \text{data dependence}
  \]
Many RL problems have highly dependent data!

- Markovian decision process admitting specific jump diffusion; e.g. financial market, self-driving car, and etc.
- Bad replay buffer; e.g.
  \[
  \{\xi_1\}, \{\xi_1, \xi_2\}, \{\xi_1, \xi_2, \xi_3\}, \ldots.
  \]
- Exploration with a updating policy.
1. Challenge 1: Non-Independent Data
   - Reduce the influence of data dependence
   - Classical optimization techniques on dependent data
   - Critical thinking: is data dependence always bad?

2. Challenge 2: Exploration-Exploitation Trade-Off
   - Quantify the error caused by lacking of exploration

3. Reference
**Question**  What is the influence of data dependence on those classical optimization techniques such as variance reduction?

**Answer**  The performance of variance reduction is reduced.

**Recap on Variance Reduction**

\[
\begin{align*}
\text{(SGD)} & \quad \nabla f(x; \xi) \\
\text{(SVRG)} & \quad \nabla f(x; \xi) - \nabla f(y; \xi) + \mathbb{E}_{\xi \sim \mu} \nabla f(y; \xi)
\end{align*}
\]

- For IID data, they are both unbiased while SVRG has lower variance when \( \|x - y\|^2 \) is small.
- For Markovian data, the bias may dominates the error term.
We apply the variance reduction technique to two existing gradient-based RL algorithms: TD learning with gradient correction (TDC) and Greedy-GQ algorithm.

**Our work** [NeurIPS'20]

<table>
<thead>
<tr>
<th></th>
<th>TDC</th>
<th>VR-TDC</th>
</tr>
</thead>
<tbody>
<tr>
<td>IID</td>
<td>$\tilde{O}(\epsilon^{-1})$</td>
<td>$\tilde{O}(\epsilon^{-\frac{3}{5}})$</td>
</tr>
<tr>
<td>Markovian</td>
<td>$\tilde{O}(\epsilon^{-1})$</td>
<td>$\tilde{O}(\epsilon^{-1})$</td>
</tr>
</tbody>
</table>

**Our work** [ICLR'21]

<table>
<thead>
<tr>
<th></th>
<th>Greedy-GQ</th>
<th>VR-Greedy-GQ</th>
<th>SVRG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Markovian</td>
<td>$\tilde{O}(\epsilon^{-3})$</td>
<td>$\tilde{O}(\epsilon^{-2})$</td>
<td>-</td>
</tr>
<tr>
<td>IID</td>
<td>-</td>
<td>-</td>
<td>$O(\epsilon^{-\frac{5}{3}})$</td>
</tr>
</tbody>
</table>
Outline

1. **Challenge 1: Non-Independent Data**
   - Reduce the influence of data dependence
   - Classical optimization techniques on dependent data
   - Critical thinking: is data dependence always bad?

2. **Challenge 2: Exploration-Exploitation Trade-Off**
   - Quantify the error caused by lacking of exploration

3. **Reference**
**Question** Does the data dependence always make the algorithm perform worse?

**Answer** No. Sometimes, the dependence makes it better!

**Our work** [ICML’20]

- The empirical risk minimization problem:

  \[
  \min_x \frac{1}{n} \sum_{i=1}^n \ell_i(x).
  \]

- We show that **sampling with reshuffle is better than IID sampling.**
1. **Challenge 1: Non-Independent Data**
   - Reduce the influence of data dependence
   - Classical optimization techniques on dependent data
   - Critical thinking: is data dependence always bad?

2. **Challenge 2: Exploration-Exploitation Trade-Off**
   - Quantify the error caused by lacking of exploration

3. **Reference**
**Question** How can we theoretically understand Exploration-Exploitation trade-off?

**Answer** We need to quantify the error caused by lacking of exploration.

**Our work** [ICML’22 - To be submitted]

- Given the off-line data $D$, what is the best performance achieved by Q-learning?
- Bound the gap to optimal value function:

$$(1 - \gamma) \mathbb{E}_{s \sim \mu_0} [V^*(s) - V^{\pi(K)}(s)]$$

$$\leq \frac{2}{1-\gamma} \sqrt{C \cdot (\epsilon_{\text{approx}} + \frac{1}{|D|}) + 2\gamma^K \| Q^* - Q^{(0)} \|_{2,\mathbb{I}}}$$

**Standard error of off-line Q-learning**

$$+ M \cdot \sum_{k=0}^{K-1} \gamma^k \sqrt{\nu_{K-k}(D^c)} + M \cdot \sum_{k=0}^{K-1} \gamma^k \sqrt{\nu^*_{K-k}(D^c)}.$$
Greedy policy defined by a Q-function:

\[
\pi(a|s) = \begin{cases} 
1 & a = \arg\max_{a \in \mathcal{A}} Q(s, a) \\
0 & \text{o.w.}
\end{cases}
\]

\(\pi^{(k)}\) is the greedy policy defined by the Q-function at \(k\)-th iteration.

State visitation measure of a policy \(\pi\):

\[
d^\pi := (1 - \gamma) \mathbb{E} \sum_{i=0}^{\infty} \gamma^i 1(s_t = s)
\]

where \(\{s_t\}\) is generated via the policy \(\pi\). And

\[
\nu_k := d^{\pi^{(k)}} \otimes \pi^{(k)}
\]

is the greedy-policy state-action visitation measure;

\[
\nu_k^* := d^{\pi^{(k)}} \otimes \pi^*
\]

is the optimal policy state-action visitation measure.
Exploration error:

\[ \epsilon_{\text{exploration}} = \sum_{k=0}^{K-1} \gamma^k \sqrt{\nu_{K-k}(D^c)} + \sum_{k=0}^{K-1} \gamma^k \sqrt{\nu^*_{K-k}(D^c)}. \]

- More efficient exploration strategy:
  - For each episode, it suffices to explore all possible state-action pairs generated by the target greedy policy AND one-step action taken by optimal policy.
  - Optimal exploration strategy: One-step Monte Carlo Tree Search.

- More reasonable replay buffer design:
  - All state-action pairs generated by greedy-policy are important. Don’t delete them until the next epoch.

...

