Provably Good Batch Reinforcement Learning Without Great Exploration

Yao Liu¹, Adith Swaminathan², Alekh Agarwal², Emma Brunskill¹

1: Department of Computer Science, Stanford University, 2: Microsoft Research AI

Some slides are based on Emma’s talk at Simons institute.
Reinforcement Learning

Learning to make good decisions under uncertainty

Action

Feedback
(Reward, next state)
Why batch RL is (going to be) so impactful

• Large dataset is a key factor in the success of supervised learning. It can also enable RL to do more:

- Starcraft Replay (1M)
- Self-driving cars 1100h
- Robotics Grasping (1M)

Examples are from Aviral Kumar’s talk: Stabilizing Off-Policy RL
Why Batch RL is so necessary

- Good simulators or low cost to try out
- Learning from interactions
- Online RL

≠

- Hard to model and have high stakes
- Learning from logged data
- Batch RL
Why batch RL so necessary

- Offline/Batch learning to solve online interactive learning problem is actually common in practice:

  1. Independent: online learning/contextual bandit
  2. Dependent on the model: reinforcement learning

Context distribution shift
- Decrease of model performance!
Why batch RL so necessary

• Offline/Batch learning to solve online interactive learning problem is actually common in practice:

• Batch interactive learning provides formalism for these problems
Why batch RL is so hard/interesting

• Partial information:
  • RL v.s. supervised learning, RL v.s. imitation learning ...

• Distribution shift in both context and action
  • RL v.s. bandit

• Link to causality and counterfactual reasoning:
  • Data-driven RL (observational study) v.s. RL in simulators (RCT)
  • reward|action ? reward|do action ?
  • Need to be more robust w.r.t. failure of assumptions such as Markovian

• Gap between batch RL theory and in supervised learning theory
(Off-Policy) Batch Reinforcement Learning

How can we learn a good decision policy from static datasets?

• MDP: reward $r$, transition dynamics $P$
• Policy: maps from $s$ to (a distribution of) $a$
• Data draw from a static state-action distribution $\mu(s, a)$

\[(s, a, r, s') \sim \mu \times r \times P\]

• Goal: Find a policy $\pi$ to maximize:

\[V^\pi = \mathbb{E} \left[ \sum_{h=0}^{\infty} \gamma^t r_t \mid a_t \sim \pi(\cdot | s_t) \right]\]
Theoretical Guarantee of Batch RL

The learned policy should generalize from past datasets to the new (on-policy) context distribution?

• Validation → Off-policy evaluation
• Provable guarantee of the risk, which should be:
  • Account for finite samples
  • Be suitable for high-dimensional, continuous state spaces
  • Address function approximation
  • Ideally make minimal assumptions on hypothesis class and data distribution
Existing Guarantees of Batch RL

• Importance sampling based policy optimization [Thomas et al., 2015]
  • Error bound $\propto \text{Exp}(\text{horizon})$

• Kernel-based methods [Tosatto et al., 2020]
  • Error bound $\propto \text{Exp}(\text{dimension})$

• Marginalized importance sampling and policy gradient [Liu et al., 2019, Kallus & Uehara 2019, 2020]
  • Asymptotic efficiency & convergence guarantees

• Approximate dynamic programming (fitted value/policy iteration) [e.g. Munos 2003; Munos & Szepesvári 2008; Antos et al., 2008; Lazaric et al., 2012; Farahmand et al., 2009; Maillard et al., 2010; Le, Voloshin, Yue 2019; Chen & Jiang 2019; Xie & Jiang 2020]
  • Assumption 1: Realizability
  • Assumption 2: Low inherent Bellman error (Also strong, recently solved by Xie and Jiang)
  • Assumption 3: Distribution overlap between data and hypothesis
Concentrability Assumption

The third assumption is often called "concentrability":

$$C = \sup_{\pi \in \Pi, h} \left\| \frac{\eta_{\pi}^h(s, a)}{\mu(s, a)} \right\|_\infty < \infty$$

With this assumption, the sample complexity of fitted Q iteration in Chen and Jiang 2019 is

$$O \left( \frac{C \ln |\mathcal{F}|}{\epsilon^2 (1 - \gamma)^2} \right)$$
Concentrability: a Very Strong Assumption

\[ C = \sup_{\pi \in \Pi, h} \left\| \eta_{h}^{\pi}(s, a) \right\|_{\infty} < \infty \]

• Naïve bound of \( C \) can be \( \exp(\text{horizon}) \)
• Often the data will cover most of desirable behaviors and will not cover every weird undesirable behavior:

Support of the Dataset

That also need to be included in the dataset!

What would happen if it doesn’t?
Challenges due to Insufficient Data Support

- When these three properties are combined, learning can diverge with the value estimates becoming unbounded. [Sutton and Barto 2018]
- The problem with concentrability assumption is not just for the theory, but also for algorithms.
  - Concentrability puts a strong restriction of distribution mismatch.
  - Otherwise -> deadly triad
Extrapolation Error in ADP Analysis

\[ Q(s, a) \leftarrow r(s, a) + \gamma \mathbb{E}_{s', a'} \left[ \max_{a'} Q(s', a') \right] \]

Estimating this part is hard, especially with off-policy data

- Maximization bias (in online Q learning):
  bias of \( \max_{a'} Q(s', a') \) due to the convexity of the max function.

- Extrapolation error (in off-policy Q learning): When \( (s', a_0) \) is not supported well for some \( a_0 \)
  Function approximation => \( Q(s', a_0) \) is out of control
  \( \max_{a'} Q(s', a') \) is out of control due to \( a_0 \)
Extrapolation Error in Practice

• Batch RL does not work well even with data generated from expert.
• $Q$ values explode up, likely due to the extrapolation error.

[Fujimoto et al., 2019]
Our Contribution

• We provide finite sample error bounds of variants of approximate policy iteration and value iteration algorithms that are *agnostic* to the concentrability assumption.
  • No assumption on $C$.
  • Instead you choose a threshold of the support level by data then the we can bound the error w.r.t. the best “supported” policy.

• We consider continuous spaces with function approximation.

• We also show a more practical (aka “deep RL”) version of our algorithm in some recent batch RL benchmarks.
Doing the Best with What We’ve Got

Simple ideas:
• restrict off policy optimization to those with overlap in data
• assume pessimistic outcomes for areas of state--action space with insufficient overlap/support

An interesting contrast:

Online RL:
Need to explore
Be optimistic in face of uncertainty

Batch RL:
Cannot explore
Be pessimistic in face of uncertainty?
Challenges to previous algorithm

Reasons why baselines fail:
- Many baselines focus on penalty/constraints that are based on $\text{dist}(\pi(a|s), \mu(a|s))$.
- In this example a sequence of large action conditional probabilities leads to a rare state.
- Due to finite samples, estimates of the reward of this rare state can be overestimated.

Success rate: $\#(\text{getting the optimal policy})/\#(\text{trials})$
Challenges to previous algorithms

Success rate: # (getting the optimal policy) / # (trials)

Reasons why baselines fail:

• SPIBB adds conservatism based on estimates of \( \mu \) & value of \( \mu \).
• In this example, the actions which is rare under \( \mu \) also have a stochastic transition and reward, thus the \( \mu \)'s value is overestimated.
Pessimistic Value Estimates

We modified the Bellman operators in the standard fitted Q/policy iteration.

• Filtration function:
  \[ \zeta(s, a; \hat{\mu}, b) = 1(\hat{\mu}(s, a) > b) \]

• Bellman operator:
  \[ T f(s, a) = r(s, a) + \gamma \mathbb{E}_{s'} \left[ \max_{a'} \zeta(s', a') f(s', a') \right] \]

  = 0 for \((s',a')\) with insufficient data.

  We assume \(r(s,a) \geq 0\). Therefore pessimistic estimate for such tuples.

  b can account for statistical uncertainty due to finite samples.
Marginalized Behavior Supported (MBI)

We modified the Bellman operators in the standard fitted Q/policy iteration.

- Filtration function:
  \[ \zeta(s, a; \hat{\mu}, b) = 1(\hat{\mu}(s, a) > b) \]

- Bellman operator and Bellman evaluation operator:
  \[
  \mathcal{T} f(s, a) = r(s, a) + \gamma \mathbb{E}_{s', a'} \left[ \max_{a'} \zeta(s', a') f(s', a') \right] \\
  \mathcal{T}^\pi f(s, a) = r(s, a) + \gamma \mathbb{E}_{s', a' \sim \pi} \left[ \zeta(s', a') f(s', a') \right]
  \]
Previous Model-free Batch RL guarantees

• Assume \( \sup_{\pi \in \Pi, h} \left\| \frac{\eta^\pi_h(s,a)}{\mu(s,a)} \right\|_\infty \leq C \)

\[ V^* - V^\pi \leq O\left(\sqrt{\frac{C}{n}}\right) \]
Near Optimal in Well Supported Policy Class

• Assume $\sup_{\pi \in \Pi, h} \left\| \frac{\eta^\pi_h(s,a)}{\mu(s,a)} \right\|_\infty \leq C$

$V^* - V^\pi \leq O\left(\sqrt{\frac{C}{n}}\right)$

• Define $\Pi'$ to be all $\pi$ such that $\mathbb{E}_\pi \left[1(\hat{\mu}(s,a) < b)\right] \leq \epsilon$

$V^{\pi'} - V^\pi \leq O\left(\epsilon + \frac{1}{b\sqrt{n}}\right)$,

where $\pi'$ is the best policy in $\Pi'$
Assumptions

**Assumption 1** (Bounded densities). For any non-stationary policy \( \pi \) and \( h \geq 0 \), \( \eta_h^\pi(s, a) \leq U \).

**Assumption 2** (Density estimation error). With probability at least \( 1 - \delta \), \( \| \hat{\mu} - \mu \|_{TV} \leq \varepsilon_\mu \).

**Assumption 3** (Completeness under \( \tilde{T}^\pi \)). \( \forall \pi \in \Pi \), \( \max_{f \in \mathcal{F}} \min_{g \in \mathcal{F}} \| g - \tilde{T}^\pi f \|_{2, \mu}^2 \leq \varepsilon_\mathcal{F} \).

**Assumption 4** (\( \Pi \) Completeness). \( \forall f \in \mathcal{F} \), \( \min_{\pi \in \Pi} \| E_\pi [ \zeta \circ f(s, a)] - \max_a \zeta \circ f(s, a) \|_{1, \mu} \leq \varepsilon_\Pi \).
Error Bound

We bound the error w.r.t. the best policy in the following policy set:
{all policies such that Pr(μ(s, a) < b) ≤ ε}

Error bounds\(^1\): (Showing policy iteration algorithm as an example)

\[
O\left(\frac{UV_{\text{max}}}{(1 - \gamma)^3 b} \sqrt{\frac{\ln(|\mathcal{F}||\Pi|/\delta)}{n}}\right) + \frac{V_{\text{max}} \epsilon}{1 - \gamma}
\]

\(^1\): We omit some constant terms that is same as standard ADP analysis with function approximation.
Error Bound w.r.t. the Optimal Policy

- Assume $\sup_{\pi \in \Pi, h} \left\| \frac{\eta_{h}^\pi(s,a)}{\mu(s,a)} \right\|_\infty \leq C$

- Assume $\sup_{h} \left\| \frac{1\{\eta_{h}^\pi(s,a) > 0\}}{\mu(s,a)} \right\|_\infty < C$

$$V^* - V^\pi \leq O \left( \sqrt{\frac{C}{n}} \right)$$

Previous results

$$V^* - V^\pi \leq O \left( \frac{C}{\sqrt{n}} \right),$$

Our results
Experiment in Tabular Settings

• Discretized CartPole.
  • $|S| = 10^4$, $|A| = 2$
  • (The underlying dynamics and reward are still based on the continuous spaces)
## Experiment in MuJoCo domains

<table>
<thead>
<tr>
<th>D4RL datasets name</th>
<th>MBS-BCQ</th>
<th>MBS-BEAR</th>
<th>BCQ</th>
<th>BEAR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hopper-medium</td>
<td>75.2</td>
<td>33.1</td>
<td>54.5</td>
<td>47.6</td>
</tr>
<tr>
<td>HalfCheetah-medium</td>
<td>38.4</td>
<td>39.7</td>
<td>40.7</td>
<td>38.6</td>
</tr>
<tr>
<td>Walker2d-medium</td>
<td>68.1</td>
<td>75.4</td>
<td>53.1</td>
<td>33.2</td>
</tr>
</tbody>
</table>

- The distribution $\mu(s)$ and $\mu(a|s)$ are approximated by two learned VAEs.

Notice that our algorithm is the only one with meaningful theoretical guarantees in function approximation settings. (In this case $C$ is very likely to be $\infty$.)
Experiment in MuJoCo domains

<table>
<thead>
<tr>
<th>D4RL datasets name</th>
<th>MBS-BCQ</th>
<th>MBS-BEAR</th>
<th>BCQ</th>
<th>BEAR</th>
<th>BEAR - a recent new implementation</th>
<th>MOPO</th>
<th>CQL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hopper-medium</td>
<td>75.2</td>
<td>33.1</td>
<td>54.5</td>
<td>47.6</td>
<td>52.1</td>
<td>26.5</td>
<td>58.0</td>
</tr>
<tr>
<td>HalfCheetah-medium</td>
<td>38.4</td>
<td>39.7</td>
<td>40.7</td>
<td>38.6</td>
<td>41.7</td>
<td>40.2</td>
<td>44.4</td>
</tr>
<tr>
<td>Walker2d-medium</td>
<td>68.1</td>
<td><strong>75.4</strong></td>
<td>53.1</td>
<td>33.2</td>
<td>59.1</td>
<td>14.0</td>
<td><strong>79.2</strong></td>
</tr>
</tbody>
</table>

Notice that our algorithm is the only one with meaningful theoretical guarantees in function approximation settings. (In this case C is very likely to be $\infty$.)
Discussion & Future Work

• If we are not looking for computationally tractable algorithm, is there a better information-theoretical error bounds?

• A conjecture: Assume \( \left\| \frac{\eta_{h^*}(s,a)}{\mu(s,a)} \right\|_\infty \leq C \)

\[
V^* - V^\pi \leq O\left(\sqrt{\frac{C}{n}}\right)
\]

or even better?
Discussion & Future Work

An interesting contrast:

Online RL:  
Need to explore  
Be optimistic in face of uncertainty

Batch RL:  
Cannot explore  
Be pessimistic in face of uncertainty?

• Is pessimism the best thing to do for batch RL?
• In some sense, our algorithm is similar to the R-max algorithm in the optimism case.
• \textit{Lower} confidence bound algorithm (in contrast to UCB)?
Discussion & Future Work

• A spectrum of algorithm:

<table>
<thead>
<tr>
<th>Imitation Learning</th>
<th>Naïve Batch RL</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Exploitation”</td>
<td>“Exploration”</td>
</tr>
<tr>
<td>Suitable for expert data</td>
<td>Suitable for exploratory data</td>
</tr>
<tr>
<td>High bias when assumption failed</td>
<td>High variance when assumption failed</td>
</tr>
</tbody>
</table>

• Does there exist a continuous spectrum of algorithms to fill the gap?
• Is there a way to adaptively choose parameter/algorithm based on the type of datasets?
Conclusion

• We provide finite sample error bound that is *agnostic* to the concentrability assumption.
  • No assumption on C.
  • The error bound is w.r.t. the best “supported” policy.
  • Works for continuous spaces with function approximation.

• Algorithm intuition: pessimism in face of the lack of support

• We also show a more practical (aka “deep RL”) version of our algorithm in some recent batch RL benchmarks.