

**BACKGROUND**

**Motivation**
- Function approximation is widely applied in various reinforcement learning algorithms.
- Can we design provably efficient Q-learning algorithms with function approximation?

**Previous Results**
- We have a good understanding of Q-learning in the tabular setting. Various efficient algorithms are known.
- Wen and Van Roy [3] proposed optimistic constraint propagation (OCP), which can deal with deterministic systems.
- Sample-efficient algorithms for “Linear MDPs” are known [4, 2], which further require assumptions on the transition model.

**Challenges**
- How to efficiently explore the state space, so that one can learn a good predictor (Q-function) that generalizes across states.
- Decide when to stop exploring, to avoid taking too many samples on similar distributions.
- Our idea: explicitly check the distribution shift.

**DISTRIBUTION SHIFT ERROR CHECKING ORACLE**

**Motivation**
- In RL, we often need to know whether a predictor learned from samples generated from one distribution $D_1$ can predict well on another distribution $D_2$.
- How to check distribution shift when function approximation is adopted?

**Distribution Shift Error Checking Oracle (DSEC)**
- $v = \max_{f_1, f_2 \in \mathcal{F}} \mathbb{E}_{s \sim D_2} [(f_1(s) - f_2(s))^2]$
- $s.t. \mathbb{E}_{s \sim D_1} [(f_1(s) - f_2(s))^2] + \Lambda(f_1, f_2) \leq \epsilon_1$.

- The oracle returns True if $v \geq \epsilon_2$ and False otherwise.
- Here $\mathcal{F}$ is the function class for Q-functions and $\Lambda$ is a regularizer to prevent pathological cases.

**Intuition**
- Let $f_2$ be the optimal Q-function and $f_1$ is a predictor we learned using samples generated from distribution $D_1$. We know $f_1$ has a small expected error on distribution $D_1$.
- $v$ is an upper bound on the expected error of $f_1$ on $D_2$, and thus we can use $v$ to test whether $f_1$ can predict well on $D_2$.

**ALGORITHM FOR LINEAR FUNCTION APPROXIMATION**

- Explore the state space level by level. At each level, use linear regression to learn the Q-function.
- At each level, change the exploration policy to consider all possible actions. Use DSEC to detect distribution shift. For linear functions, DSEC is equivalent to PCA.
- When distribution shift is detected, run the algorithm recursively to collect samples and learn the Q-function.
- Total number of recursions can be bounded using the ellipsoid potential lemma.

**OPEN PROBLEMS**

- Generalize DSEC and our algorithm to more general function classes including neural networks and kernel methods.
- Is the assumption that the optimal Q-function is linear sufficient for efficient reinforcement learning?
  - Our theoretical bound relies on assumptions including “gap” and “low variance”.
  - Are these assumptions necessary?
  - Yes for “agnostic” cases. See recent lower bounds [1].
- Practical versions of DSEC, and how to incorporate DSEC into practical RL algorithms.

**REFERENCES**


