POLITEX: Regret Bounds for Policy Iteration Using Expert Prediction

Yasin Abbasi-Yadkori 1, Peter L. Bartlett 2, Kush Bhatia 2, Nevena Lazić 3, Csaba Szepesvári 4, Gellért Weisz 4

1 Adobe, 2 Berkeley, 3 Google, 4 DeepMind
Setting and notation

- Markov decision process (MDP)
  - observed states $x \in S$
  - discrete actions $a \in \{1, \ldots, A\}$
  - unknown costs $c(x, a)$
  - unknown transition dynamics $P(x_{t+1}|x_t, a_t)$
- Average cost of a policy $\pi(a|x)$:

$$\lambda_\pi = \mathbb{E}\left[ \lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} c(x_t^\pi, a_t^\pi) \right]$$

A1 Unichain: MDP states form a single recurrent class under any policy.

A2 Uniform mixing: $\| (\nu' - \nu_\pi) H \|_1 \leq \exp(-1/\kappa) \| \nu' - \nu_\pi \|$, where $\nu_\pi(x, a)$ is the steady-state distribution of $\pi$, and $H(x, a, (x', a')) = P(x'|x, a)\pi(a'|x')$.

$\{ (x_t^\pi, a_t^\pi) \}_{t=1,2,\ldots}$ denotes the state-action sequence when following $\pi$. 
Policy iteration

**Input:** phase length $\tau > 0$, initial state $x_0$

Set $\hat{Q}_0(x, a) = 0$, $\pi_0(a|x) = 1/A \ \forall x, a$

for $i := 0, 1, 2, \ldots$, do

Policy evaluation:

- Execute $\pi_i$ for $\tau$ time steps and collect data.
- Compute the action-value estimate $\hat{Q}_i(x, a)$.

Policy improvement:

\[ \pi_{i+1}(\cdot|x) = \arg\min_{u \in \Delta} \langle u, \hat{Q}_i(x, \cdot) \rangle \]

end for

\[ Q_\pi(x, a) = c(x, a) - \lambda_\pi + \mathbb{E} \left[ \lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} c(x_t^\pi, a_t^\pi) \right] \]

$\hat{Q}_i$ is an approximation of $Q_{\pi_i}$ (e.g. linear or neural network)
Policy iteration using expert advice (POLITEX)

**Input:** phase length $\tau > 0$, initial state $x_0$

Set $\hat{Q}_0(x, a) = 0$, $\pi_0(a|x) = 1/A \ \forall x, a$

**for** $i := 0, 1, 2, \ldots$, **do**

Policy evaluation:
- Execute $\pi_i$ for $\tau$ time steps and collect data.
- Compute the action-value estimate $\hat{Q}_i(x, a)$.

Policy improvement:

$$
\pi_{i+1}(\cdot|x) = \arg\min_{u \in \Delta} \langle u, \sum_{j=0}^{i} \hat{Q}_j(x, \cdot) \rangle - \eta^{-1} \mathcal{H}(u)
$$

$$
\propto \exp \left( -\eta \sum_{j=0}^{i} \hat{Q}_j(x, \cdot) \right)
$$

**end for**
For $\hat{Q}_i$ estimated from $\tau$ transitions, we require

$$\hat{Q}_i \in [b, b + Q_{\text{max}}] \quad \text{and} \quad \|Q_{\pi_i} - \hat{Q}_i\|_{\nu_{\pi_i}} = \epsilon_0 + O(1/\sqrt{\tau}),$$

where $\epsilon_0$ is the approximation error. Satisfied e.g. by LSPE (Bertsekas & Ioffe, 1996) under a "feature excitation" assumption on the policies.

Then the regret of Politex w.r.t a reference policy $\pi^*$, defined as $R_T = \sum_{t=1}^{T} c(x_t, a_t) - c(x_t^*, a_t^*)$, is of the order

$$R_T = \tilde{O}(T^{3/4} + \epsilon_0 T).$$

Regret bound does not scale in the size of the underlying MDP.

Unlike existing policy iteration results (for discounted MDPs), does not depend on the concentrability coefficient.

Easy to implement - no confidence bounds required.
Experiments

**Politex + LSPE on Queueing networks**

- 4-queue
  - LSPI
  - POLITEX
  - RLSVI

- 8-queue

**Politex + neural nets on Ms Pacman**

- Reward per episode
  - epsilon-greedy
  - POLITEX
  - DQN

---

Nevena Lazić (Google)
Related work


  MDP-E uses an experts algorithm in each state $x$ with losses $Q(x, a)$. Politex is similar, but learns the action-value function from data.


  Similar approach applied to the control of LQ systems.


  Asymptotic convergence analysis of average-cost LSPE, here adapted to finite-sample analysis for learning Q functions.


  Similar algorithms based on heuristics.