Class Tutorial 9

1. Approximate Greedy Policy
In previous tutorial we analyzed a TD(0) policy-evaluation scheme. Generally, we would like to perform an improvement relatively to the evaluated policy value. Prove the following proposition (which is a more basic version of Theorem 11.1 from lecture notes).

a. Proposition: Let \( v^* \) be the value of the optimal policy, \( \hat{v}^* \) be an estimator of its value s.t \( |v^* - \hat{v}^*|_\infty \leq \epsilon \). Then, the Greedy policy w.r.t. \( \hat{v}^* \), \( \pi_G \), satisfies

\[
|v_{\pi_G} - v^*|_\infty \leq \frac{2\gamma \epsilon}{1 - \gamma}
\]

Solution
a. We use the fixed point properties of \( v^* \), the fact that \( T^{\pi_G} \hat{v}^* = T \hat{v}^* \), and the fact that \( T^\pi, T \) are \( \gamma \) contractions in the max norm (all discussed in lectures).

\[
|v_{\pi_G} - v^*|_\infty \leq |v_{\pi_G} - T \hat{v}^*|_\infty + |T \hat{v}^* - v^*|_\infty \\
= |T^{\pi_G} v_{\pi_G} - T^{\pi_G} \hat{v}^*|_\infty + |T \hat{v}^* - T v^*|_\infty \\
\leq \gamma |v_{\pi_G} - \hat{v}^*|_\infty + \gamma |\hat{v}^* - v^*|_\infty \\
\leq \gamma |v_{\pi_G} - v^*|_\infty + \gamma |v^* - \hat{v}^*|_\infty + \gamma |\hat{v}^* - v^*|_\infty \\
\leq \gamma |v_{\pi_G} - v^*|_\infty + 2\gamma \epsilon
\]

By moving the first term in the RHS to the LHS and dividing by \( 1 - \gamma \) we conclude the proof.

2. Least Squares Temporal Difference (LSTD)
In the previous tutorial we have seen that the online TD(0) converges to a solution of the linear equation

\[
Aw = b.
\]

Now we will propose a batch algorithm that find a solution to the same equation. We are given a sequence of \( N \) state pairs \( \{s_i, s'_i\}_{i=1}^N \), where \( s_i \sim d \), and \( s'_i \sim P^\pi (s | s_i) \).

a. Suggest estimators for \( A \) and \( b \) from the data \( \{s_i, s'_i\}_{i=1}^N \).

b. Suggest a batch algorithm for finding \( w \).

Solution
a. Recall that
\[ b = \Phi^\top Dr = \sum_s d(s)r(s)\phi(s) \approx \sum_{i=1}^N r(s_i)\phi(s_i) \]

And

\[ A = \gamma\Phi^\top DP\Phi - \Phi^\top D\Phi \]

Therefore we similarly have

\[ \Phi^\top D\Phi = \sum_s d(s)\phi(s)\phi^\top(s) \approx \sum_{i=1}^N \phi(s_i)\phi^\top(s_i) \]

And

\[ \Phi^\top DP\Phi = \sum_{i,s'} d(s)P^\pi(s' | s)\phi(s)\phi(s') \approx \sum_{i=1}^N \phi(s_i)\phi^\top(s_i') \]

b. Given the data, we first form the estimates \( \hat{A}, \hat{b} \) using the estimators described above:

\[ \hat{A} = \sum_{i=1}^N \phi(s_i)\left(\gamma\phi^\top(s_i') - \phi^\top(s_i)\right) \]

\[ \hat{b} = \sum_{i=1}^N r(s_i)\phi(s_i) \]

We then solve the linear equation:

\[ w = \hat{A}^{-1}\hat{b} \]

3. Least Squares Policy Iteration (LSPI)

In the previous question we explored batch policy evaluation with function approximation. We now propose a batch algorithm for *policy improvement* with function approximation.

Similar to evaluating the value function \( V^\pi(s) \), we can also evaluate the state-action value function \( Q^\pi(s,a) \). We approximate \( Q^\pi(s) \) using linear function approximation, i.e.,

\[ \tilde{Q}^\pi(s,a) = \phi(s,a)^\top w \]

where \( \phi(s,a) \) are state-action features. We assume that the data is a sequence of \( N \) state-action-next state pairs \( \{s_i,a_i,s_i'\}_{i=1}^N \).

a. For a known policy \( \pi \), extend the LSTD algorithm to evaluating the weights for \( \tilde{Q}^\pi(s,a) \).

b. For a given weight vector \( w \), what is the greedy policy w.r.t. \( \tilde{Q}^\pi(s,a) = \phi(s,a)^\top w \)?
c. Show that LSTD can be used to evaluate the weights for $\tilde{Q}_{\pi_{\text{greedy}}}(s, a)$ of the greedy policy w.r.t. some $w$.

d. Suggest an algorithm that interleaves the policy evaluation of LSTD and policy improvement using the greedy policy.

Solution

a. Note that we can define an 'augmented' state space $\mathcal{S} = \{s, a\}$, and perform LSTD on the augmented space:

$$\hat{A} = \sum_{i=1}^{N} \phi(s_i, a_i) \left( \gamma \phi^\top(s_i', \pi(s_i')) - \phi^\top(s_i, a_i) \right)$$

$$\hat{b} = \sum_{i=1}^{N} r(s_i, a_i) \phi(s_i, a_i)$$

$$w = \hat{A}^{-1} \hat{b}$$

b. The greedy policy is given by

$$\pi_{\text{greedy}}(s; w) = \arg\max_a \phi(s, a)^\top w$$

c. The only change we need to make is:

$$\hat{A} = \sum_{i=1}^{N} \phi(s_i, a_i) \left( \gamma \phi^\top(s_i', \pi_{\text{greedy}}(s_i'; w)) - \phi^\top(s_i, a_i) \right)$$

d. The Least-Squares Policy Iteration (LSPI) works iteratively, as follows:

start with some arbitrary $w_0$

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

$$\hat{A} = \sum_{i=1}^{N} \phi(s_i, a_i) \left( \gamma \phi^\top(s_i', \pi_{\text{greedy}}(s_i'; w_i)) - \phi^\top(s_i, a_i) \right)$$

$$\hat{b} = \sum_{i=1}^{N} r(s_i, a_i) \phi(s_i, a_i)$$

$$w_{i+1} = \hat{A}^{-1} \hat{b}$$