1 Learning in Gridworld

Consider the example gridworld that we looked at in lecture. We would like to use TD learning and q-learning to find the values of these states.

Suppose that we have the following observed transitions:
(B, East, C, 2), (C, South, E, 4), (C, East, A, 6), (B, East, C, 2)

The initial value of each state is 0. Assume that $\gamma = 1$ and $\alpha = 0.5$.

(a) What are the learned values from TD learning after all four observations?

(b) What are the learned Q-values from Q-learning after all four observations?
Q2. MDPs and RL: Go Bears!

Cal’s Football team is playing against UCLA for the big homecoming game Saturday night. With a lot of losses in the season so far, Cal needs to switch up their strategy to get any hope of winning this game.

Luckily, the Quarterback (Joe) is a star student in CS188 and has decided to model the game as a Markov Decision Process. There are only two states – the Play state (shown as the field in the diagram) and the Win State. Although the connectivity of the states is known, the probabilities for each are not.

There are no actions available from the Win state – the game simply ends.

From the Play state there are three actions: Run, Pass, and HailMary. The connectivity of each action to the two states is shown above.

Reward Values:

<table>
<thead>
<tr>
<th>State</th>
<th>Action</th>
<th>State'</th>
<th>R(s,a,s')</th>
</tr>
</thead>
<tbody>
<tr>
<td>Play</td>
<td>Run</td>
<td>Play</td>
<td>2</td>
</tr>
<tr>
<td>Play</td>
<td>Pass</td>
<td>Play</td>
<td>4</td>
</tr>
<tr>
<td>Play</td>
<td>Pass</td>
<td>Win</td>
<td>10</td>
</tr>
<tr>
<td>Play</td>
<td>Hail Mary</td>
<td>Play</td>
<td>0</td>
</tr>
<tr>
<td>Play</td>
<td>Hail Mary</td>
<td>Win</td>
<td>100</td>
</tr>
</tbody>
</table>

(a) Learning Values Joe wants to learn the value of the play state so he can estimate the outcome of the game. He uses a discount factor of 0.5 for all questions below.

(i) Joe first uses temporal difference value learning to learn the value of the play state. After initializing his beliefs to 0, he sees two episodes while in tape review. With a learning rate \( \alpha \) of 0.5 what value of the state play does he learn?
\begin{tabular}{|c|c|c|}
\hline
State & Action & State' \\
\hline
Play & Run & Play \\
Play & Hail Mary & Play \\
\hline
\end{tabular}

\[ V(\text{play}) = \]

(ii) Coach Tedford decides to give Joe a fixed policy instead:

\[ \pi(s) = \text{Run} \]

What value for the state \textit{play} would Joe calculate if he ran value iteration until convergence? Keep in mind that \( \sum_{n=0}^{\infty} \left(\frac{1}{2}\right)^n = 2 - \left(\frac{1}{2}\right)^n = 1 + 0.5 + 0.25 + 0.125 + ... \)

\[ V^\pi(\text{play}) = \]

(b) \textbf{Game Time} Joe watches the next lecture video from class and now wants to use Q-learning to compute his optimal strategy.

(i) First Joe uses temporal difference Q-learning to learn the values of the Q nodes. He sees three episodes during the first quarter:

\begin{tabular}{|c|c|c|}
\hline
State & Action & State' \\
\hline
Play & Run & Play \\
Play & Hail Mary & Play \\
Play & Pass & Win \\
\hline
\end{tabular}

Update the Q node values after processing each episode (in order). Use a learning rate of 0.5 and a discount rate of 0.5.

\begin{tabular}{|c|c|c|}
\hline
State & Action & \( Q(s,a) \) \\
\hline
Play & Run & \\
Play & Hail Mary & \\
Play & Pass & \\
\hline
\end{tabular}

(c) Q learning is going well, but it’s taking too much time. Thankfully Oski shows up with some special information – he has watched so many games that he know’s the true transition probabilities! Here they are:
<table>
<thead>
<tr>
<th>State</th>
<th>Action</th>
<th>State'</th>
<th>( R(s,a,s') )</th>
<th>( T(s,a,s') )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Play</td>
<td>Run</td>
<td>Play</td>
<td>2</td>
<td>1.0</td>
</tr>
<tr>
<td>Play</td>
<td>Pass</td>
<td>Play</td>
<td>4</td>
<td>0.5</td>
</tr>
<tr>
<td>Play</td>
<td>Pass</td>
<td>Win</td>
<td>10</td>
<td>0.5</td>
</tr>
<tr>
<td>Play</td>
<td>Hail Mary</td>
<td>Play</td>
<td>0</td>
<td>0.9</td>
</tr>
<tr>
<td>Play</td>
<td>Hail Mary</td>
<td>Win</td>
<td>100</td>
<td>0.1</td>
</tr>
</tbody>
</table>

(i) Now with these probabilities, what is the optimal policy when there is one time step left? The value?

\[
\pi_{k=1}(\text{play}) = \]
\[
V_{k=1}(\text{play}) =
\]

(ii) For two time steps left, what is the optimal policy with discount factor 0.5? Hint: you can use your value above to aid in this computation.

\[
\pi_{k=2}(\text{play}) = \]
\[
V_{k=2}(\text{play}) = \]