CS 188
Spring 2023

Regular Discussion 8 Solutions

1 Pacman with Feature-Based Q-Learning

We would like to use a Q-learning agent for Pacman, but the size of the state space for a large grid is too massive to hold in memory. To solve this, we will switch to feature-based representation of Pacman's state.

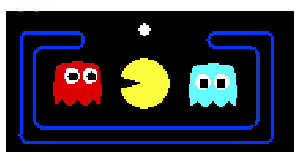
(a) We will have two features, F_g and F_p, defined as follows:

$$F_g(s,a) = A(s) + B(s,a) + C(s,a)$$
$$F_p(s,a) = D(s) + 2E(s,a)$$

where

$A(s)$ = number of ghosts within 1 step of state s
$B(s,a)$ = number of ghosts Pacman touches after taking action a from state s
$C(s,a)$ = number of ghosts within 1 step of the state Pacman ends up in after taking action a
$D(s)$ = number of food pellets within 1 step of state s
$E(s,a)$ = number of food pellets eaten after taking action a from state s

For this pacman board, the ghosts will always be stationary, and the action space is $\{left, right, up, down, stay\}$.



calculate the features for the actions $\in \{left, right, up, stay\}$

$$F_p(s, up) = 1 + 2(1) = 3$$
$$F_p(s, left) = 1 + 2(0) = 1$$
$$F_p(s, right) = 1 + 2(0) = 1$$
$$F_p(s, stay) = 1 + 2(0) = 1$$
$$F_g(s, up) = 2 + 0 + 0 = 2$$
$$F_g(s, left) = 2 + 1 + 1 = 4$$
$$F_g(s, right) = 2 + 1 + 1 = 4$$
$$F_g(s, stay) = 2 + 0 + 2 = 4$$

(b) After a few episodes of Q-learning, the weights are $w_g = -10$ and $w_p = 100$. Calculate the Q value for each action $\in \{left, right, up, stay\}$ from the current state shown in the figure.

$$\begin{aligned}
Q(s, up) &= w_p F_p(s, up) + w_g F_g(s, up) = 100(3) + (-10)(2) = 280 \\
Q(s, left) &= w_p F_p(s, left) + w_g F_g(s, left) = 100(1) + (-10)(4) = 60 \\
Q(s, right) &= w_p F_p(s, right) + w_g F_g(s, right) = 100(1) + (-10)(4) = 60 \\
Q(s, stay) &= w_p F_p(s, stay) + w_g F_g(s, stay) = 100(1) + (-10)(4) = 60
\end{aligned}$$

(c) We observe a transition that starts from the state above, s, takes action up, ends in state s' (the state with the food pellet above) and receives a reward $R(s, a, s') = 250$. The available actions from state s' are *down* and *stay*. Assuming a discount of $\gamma = 0.5$, calculate the new estimate of the Q value for s based on this episode.

$$\begin{aligned}
Q_{new}(s, a) &= R(s, a, s') + \gamma * \max_{a'} Q(s', a') \\
&= 250 + 0.5 * \max\{Q(s', down), Q(s', stay)\} \\
&= 250 + 0.5 * 0 \\
&= 250
\end{aligned}$$

where

$$\begin{aligned}
Q(s', down) &= w_p F_p(s, down) + w_g F_g(s, down) = 100(0) + (-10)(2) = -20 \\
Q(s', stay) &= w_p F_p(s, stay) + w_g F_g(s, stay) = 100(0) + (-10)(0) = 0
\end{aligned}$$

(d) With this new estimate and a learning rate (α) of 0.5, update the weights for each feature.

$$\begin{aligned}
w_p &= w_p + \alpha * (Q_{new}(s, a) - Q(s, a)) * F_p(s, a) = 100 + 0.5 * (250 - 280) * 3 = 55 \\
w_g &= w_g + \alpha * (Q_{new}(s, a) - Q(s, a)) * F_g(s, a) = -10 + 0.5 * (250 - 280) * 2 = -40
\end{aligned}$$

2 Q-learning

Consider the following gridworld (rewards shown on left, state names shown on right).

Rewards

+10	+1

State names

A	B
G1	G2

From state A, the possible actions are right($\rightarrow$) and down($\downarrow$). From state B, the possible actions are left($\leftarrow$) and down($\downarrow$). For a numbered state (G1, G2), the only action is to exit. Upon exiting from a numbered square we collect the reward specified by the number on the square and enter the end-of-game absorbing state X. We also know that the discount factor $\gamma = 1$, and in this MDP all actions are **deterministic** and always succeed.

Consider the following episodes:

Episode 1 ($E1$)

s	a	s'	r
A	$\downarrow$	$G1$	0
$G1$	exit	X	10

Episode 2 ($E2$)

s	a	s'	r
B	$\downarrow$	$G2$	0
$G2$	exit	X	1

Episode 3 ($E3$)

s	a	s'	r
A	$\rightarrow$	B	0
B	$\downarrow$	$G2$	0
$G2$	exit	X	1

Episode 4 ($E4$)

s	a	s'	r
B	$\leftarrow$	A	0
A	$\downarrow$	$G1$	0
$G1$	exit	X	10

(a) Consider using temporal-difference learning to learn $V(s)$. When running TD-learning, all values are initialized to zero.
For which sequences of episodes, if repeated infinitely often, does $V(s)$ converge to $V^*(s)$ for all states s?

(Assume appropriate learning rates such that all values converge.)
Write the correct sequence under "Other" if no correct sequences of episodes are listed.

☐ $E1, E2, E3, E4$ ☐ $E1, E2, E1, E2$ ☐ $E1, E2, E3, E1$ ■ $E4, E4, E4, E4$
☐ $E4, E3, E2, E1$ ☐ $E3, E4, E3, E4$ ☐ $E1, E2, E4, E1$

■ Other See explanation below

TD learning learns the value of the executed policy, which is $V^\pi(s)$. Therefore for $V^\pi(s)$ to converge to $V^*(s)$, it is necessary that the executing policy $\pi(s) = \pi^*(s)$.

Because there is no discounting since $\gamma = 1$, the optimal deterministic policy is $\pi^*(A) = \downarrow$ and $\pi^*(B) = \leftarrow$ ($\pi^*(G1)$ and $\pi^*(G2)$ are trivially exit because that is the only available action). Therefore episodes $E1$ and $E4$ act according to $\pi^*(s)$ while episodes $E2$ and $E3$ are sampled from a suboptimal policy.

From the above, TD learning using episode $E4$ (and optionally $E1$) will converge to $V^\pi(s) = V^*(s)$ for states A, B, $G1$. However, then we never visit $G2$, so $V(G2)$ will never converge. If we add either episode $E2$ or $E3$ to ensure that $V(G2)$ converges, then we are executing a suboptimal policy, which will then cause $V(B)$ to not converge. Therefore none of the listed sequences will learn a value function $V^\pi(s)$ that converges to $V^*(s)$ for all states s. An example of a correct sequence would be $E2, E4, E4, E4, \ldots$; sampling

$E2$ first with the learning rate $\alpha = 1$ ensures $V^\pi(G2) = V^*(G2)$, and then executing $E4$ infinitely after ensures the values for states A, B, and $G1$ converge to the optimal values.

We also accepted the answer such that the value function $V(s)$ converges to $V^*(s)$ for states A and B (ignoring $G1$ and $G2$). TD learning using only episode $E4$ (and optionally $E1$) will converge to $V^\pi(s) = V^*(s)$ for states A and B, therefore the only correct listed option is $E4, E4, E4, E4$.

(b) Consider using Q-learning to learn $Q(s, a)$. When running Q-learning, all values are initialized to zero. For which sequences of episodes, if repeated infinitely often, does $Q(s, a)$ converge to $Q^*(s, a)$ for all state-action pairs (s, a)

(Assume appropriate learning rates such that all Q-values converge.)
Write the correct sequence under "Other" if no correct sequences of episodes are listed.

- ■ $E1, E2, E3, E4$
- ☐ $E1, E2, E1, E2$
- ☐ $E1, E2, E3, E1$
- ☐ $E4, E4, E4, E4$
- ■ $E4, E3, E2, E1$
- ■ $E3, E4, E3, E4$
- ☐ $E1, E2, E4, E1$
- ☐ Other ________

For $Q(s, a)$ to converge, we must visit all state action pairs for non-zero $Q^*(s, a)$ infinitely often. Therefore we must take the exit action in states $G1$ and $G2$, must take the down and right action in state A, and must take the left and down action in state B. Therefore the answers must include $E3$ and $E4$.