TÉCNICO LISBOA

Planeamento, Aprendizagem e Decisão Inteligente

MECD e MMAC

Relatório do Trabalho Prático 4 – Parte Teórica

Autores:

Guilherme Lopes (105319) guilherme.n.lopes@tecnico.ulisboa.pt
Leonardo Brito (105257) leonardo.amado.brito@tecnico.ulisboa.pt

Grupo 50

2022/2023 – 2º Semester, P3

Índice

1 Exercício 1

1.1 Alínea (a)

Para obtermos o *Q-learning update*, temos que usar a seguinte fórmula:

$$Q^{(t+1)}(x_t, a_t) = Q^{(t)}(x_t, a_t) + \alpha \left(c_t + \gamma \min_{a' \in \mathcal{A}} Q^{(t)}(x_{t+1}, a') - Q^{(t)}(x_t, a_t) \right) \tag{1}$$

O $\alpha = 0.1$ e o $\gamma = 0.9$. Também temos a informação que,

$Q^{(t)}_{(E,1,0,1)} = [2.8\ 2.8\ 2.8\ 2.8\ 2.54\ 2.0]$
$Q^{(t)}_{(F,1,0,1)} = [2.8\ 2.8\ 2.95\ 2.0\ 3.14\ 2.8]$

Cuja cada posição dos vetores é correspondente às respetivas ações:
$\mathcal{A}$ = { DG, CG, U, D, L, R }

Onde,

- DG - Deitar o lixo fora;
- CG - Recolher o lixo;
- U - Ir para cima;
- D - Ir para baixo;
- L - Ir para a esquerda;
- R - Ir para a direita;

São dadas as seguintes informações sobre a transição no passo t:
x_t = (E, 1, 0, 1) a_t = R $c_t = 0.2$ x_{t+1} = (F, 1, 0, 1)

Logo, os resultados serão:

$$Q^{(t+1)}((E, 1, 0, 1), R) = 2.0 + 0.1\,(0.2 + 0.9 \times 2.0 - 2.0) = 2.0. \tag{2}$$

1.2 Alínea (b)

O *SARSA update* é dado pela fórmula:

$$Q^{(t+1)}(x_t, a_t) = Q^{(t)}(x_t, a_t) + \alpha \left(c_t + \gamma \min_{a' \in \mathcal{A}} Q^{(t)}(x_{t+1}, a_{t+1}) - Q^{(t)}(x_t, a_t) \right) \tag{3}$$

Tal como na alínea anterior (a), são dadas as seguintes informações sobre a transição no passo t:
x_t = (E, 1, 0, 1) a_t = R $c_t = 0.2$ x_{t+1} = (F, 1, 0, 1) a_t = R

Logo, considerando que o $\alpha = 0.1$ e o $\gamma = 0.9$, o resultado será:

$$Q^{(t+1)}((E, 1, 0, 1), R) = 2.0 + 0.1\,(0.2 + 0.9 \times 2.8 - 2.0) = 2.072. \tag{4}$$

1.3 Alínea (c)

Um exemplo de um algoritmo de aprendizagem *on-policy* é o da pergunta 1(b) (SARSA) e um exemplo de um algoritmo de aprendizagem *off-policy* é o utilizado na pergunta 1(a).

Um algoritmo de aprendizagem *on-policy* aprende os valores associados à política utilizada para mostrar o MDP. Isso ocorre porque os algoritmos *on-policy* atualizam os valores de Q para refletir a escolha da política atual. Isso significa que o valor de Q para um par de estado-ação específico é atualizado usando a recompensa imediata recebida pelo agente e o valor de Q para o próximo estado e ação escolhida pela política atual. Ou seja, quando um agente segue uma política durante a amostragem, os valores de Q aprendidos por um algoritmo *on-policy* refletem a utilidade da política atual em termos de recompensas futuras esperadas. Na pergunta 1(b), isto pode ser comprovado na dependência da ação em *t+1*. Um algoritmo de aprendizagem *off-policy*, por outro lado, aprende o valor associado a alguma política alvo e não necessariamente a utilizada para a amostragem do MDP, ou seja, aprende os valores Q para a política ótima independentemente da política de aprendizagem. Na pergunta 1(a), isto é visível da ação a' que minimiza (x_{t+1}, a').