CS4804 Homework 3

Homework must be submitted electronically following the instructions on the course homepage. Make sure to explain you reasoning or show your derivations. You will lose points for unjustified answers, even if they are correct.

Written Problems

1. The programming homework represents Markov decision processes (MDPs) with rewards determined by the state and action $R(s,a)$, while the MDPs we studied in class reward states regardless of actions $R(s)$.

   (a) (2 points) Write the variation of the Bellman equation for describing the utility $U(s)$ of a particular state when rewards are granted for state-action pairs $R(s,a)$. The equation should reflect that the utility $U(s)$ is, as before, defined as the expected discounted reward if the agent performs optimally starting from state s.

   (b) (2 points) Write the variation of the Q-learning update when an agent is in state s, takes action a, transitions to state s', and receives reward $R(s,a)$. This equation should express how to update the value of $Q(s,a)$. Hint: it should be very similar to the Q-learning update with state rewards.

2. Section 21.4 in R+N discusses more compact, approximate parameterizations for utility functions, using a linear combination of *basis feature functions* of the state:

$$\hat{U}^{\pi}_{\boldsymbol{\theta}}(s) := \sum_{i=1}^{D} \theta_i f_i(s).$$

   For this linear approximation, the TD-learning update for each parameter θ_i, when transitioning from s to s', is

$$\theta_i \leftarrow \theta_i + \alpha \left(R(s) + \gamma \hat{U}^{\pi}_{\boldsymbol{\theta}}(s') - \hat{U}^{\pi}_{\boldsymbol{\theta}}(s) \right) f_i(s)$$

   (a) (6 points) For any finite, discrete state space, describe a basis of feature functions for which this linear approximation is exactly TD-learning. Show how the approximate TD-learning update maps exactly to the full TD update

$$U^{\pi}(s) \leftarrow U^{\pi}(s) + \alpha(R(s) + \gamma U^{\pi}(s') - U^{\pi}(s))$$

   (b) (5 points) If the discrete state space is the 5×5 grid world below, describe how you can use a much more compact set of feature functions to represent states, and discuss how this compact form enables generalization to states your TD-learning agent has never visited.

-1	-1	-1	-1	-1
-1				-1
-1		+10		-1
-1				-1
-1	-1	-1	-1	-1