ISE-537
Homework 5
Lucas (Deuce) Palmer

Problem #1 *(code in Problem1-ValueIteration.ipynb)*

Question: Please explain how to fill in the blanks?

The blanks can be filled by analyzing the ‘rest’ actions and the ‘work’ actions for each node on the graph. For example, if we rest at 3, there is a 40% chance of going to 2 and a 60% chance of staying at 3 [$0.4V(2) + 0.6V(3)$]. If we instead work at 3, there is a 50% chance of going to 4 and a 50% chance of staying at 3 [$0.5V(4) + 0.5V(3)$]. The reward at 3 is -1, so we will subtract 1 from both equations. The same exact methodology can be applied to node 4, and since node 6 is a terminating state, its function equals its reward.

By applying the methodology explained above…

$V(3) = \max\{-1 + 0.4V(2) + 0.6V(3); -1 + 0.5V(4) + 0.5V(3)\}$
$V(4) = \max\{-10 + 0.9V(6) + 0.1V(4); -10 + V(7)\}$
$V(6) = 100$

Question: Please implement the algorithm to solve (1)

i. 25 iterations

ii. $V(1) = 88.29$
$V(2) = 88.31$
$V(3) = 86.89$
$V(4) = 88.89$
$V(5) = -10$
$V(6) = 100$
$V(7) = -1000$

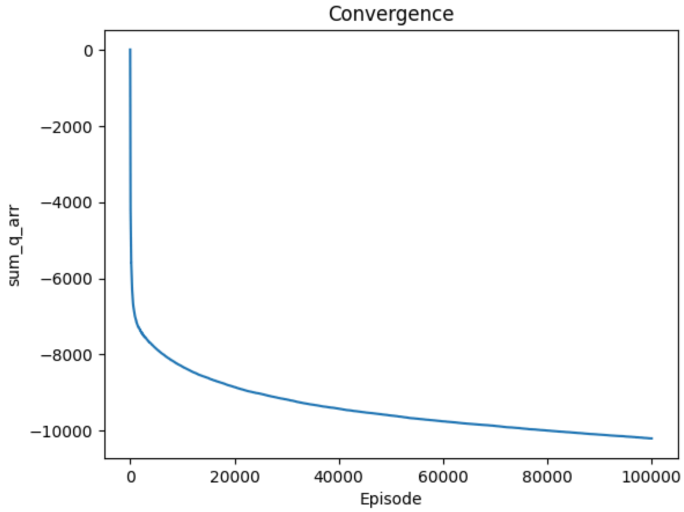
iii. State 1 => Rest
State 2 => Work
State 3 => Work
State 4 => Rest

The out-put greedy policy above describes the actions that should be taken at each non-terminal state in order to maximize rewards.

Problem #2 *(code in Problem2-SmartCab.ipynb)*

(a) i. q_table[1, 1] = -2.363
q_table[51, 3] = -2.486

ii.



As the rate of change of the sum_q_arr values is beginning to slow very rapidly, it can be inferred that the algorithm has shown evidence of convergence. If the graph were to continue this pattern, the sum_q_arr values may decrease slightly more for episodes beyond 100,000. Therefore, it could also be argued that the algorithm has not quite reached complete convergence.

(b) Average timesteps per episode: 12.89
Average penalties per episode: 0.0

The design of the evaluation model is to keep track of the number of epochs and penalties per episode. Specifically, this model is only concerned with penalties of -10, which indicates that the cab driver has picked up or dropped off a passenger in the wrong location as described in the original Python notebook ("10 point penalty for illegal pick-up and drop-off actions"). These actions cause large penalties that could substantially decrease the performance of the algorithm. Therefore, it is easy to understand why the strategy of the evaluation model would place special emphasis on these specific penalties.