Q learning

Mandatory assignment 05 - report by Frédéric Charon

This assignment is to implement a Q learning algorithm for a robot moving in between four rooms.

First, we must set up the R matrix and the Q matrix - both have the shape 6x6 since we have 6 different rooms. We fill the Q matrix with 0s and the R matrix with the rewards given for each state action pair. For each unvalid state action pair the reward is set to -1.

The key function is the update_q method, here we update the Q matrix for a state action pair according to the formula from the lecture: $Q(s,a) <- Q(s,a) + \alpha(R(s) + \Upsilon \max' Q(s',a') - Q(s,a))$. Alpha, the discount factor, is set to 0.9 and the learning rate gamma is set to 1.

In task 2 to 5 we update the Q matrix based on given state action pairs. We can see that task 2 and 4 takes the same pair but different results, this is because of the task 3 where we update the values for room 5 which is used to determine the new value when moving to this room.

Finally, we drop the robot in room 2 and want it to find a path to the final room 6, starting in room 2 we look at all the neighboring rooms and their Q values, we update them and got to the room with the highest value. We continue until we reach the final room. Here the path is from room 2 to room 5 and from there to room 6, which is indeed the shortest path for the robot.