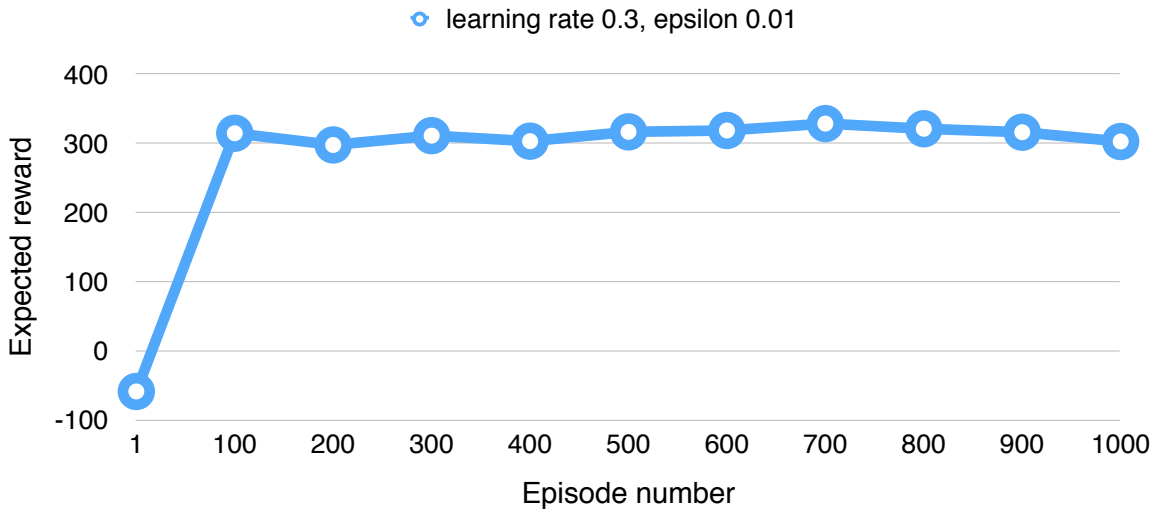
Vassil Mladenov
CS 440
MP1 Part 2

1. In WorldWithoutThief
   (a) With epsilon at 0, the agent gets stuck. It moved upward and then stopped in the upper left hand corner. This is because the path towards the goal is filled with negative rewards (the wall of slippery spaces) and the lack of randomness in exploring causes the robot to stay on the left hand side of the board, presumably so as to incur minimal losses.
   (b) With epsilon at 0.1, the behavior is more ideal. It changed from simulation to simulation. Once, it refused to move, the other time it got stuck after a single delivery. However, usually most runs had the robot delivering packages to both customers, always avoiding the double-wide set of slippery patches until it had delivered both packages. The successful behavior is caused by the randomness factor being set reasonably low, with the only issues arising from the fact that the randomness is constant and not decaying.
   (c) With the high epsilon, the behavior was much less predictable. The robot would frequently get stuck and not reach its goal. This is to be expected, given that the robot is equally likely to choose a random action and to follow it's best path.
2. In WorldWithThief
   (a) The agent does not achieve the goal and gets stuck in the top left. The epsilon may be too low for the robot to adequately survey its surroundings given that it is unaware of the thief. It has no opportunity to learn the thief's movements because it is unaware.
   (b) The performance is much better. The robot expertly avoids the thief and delivers packages properly. The robot learns the thief's movements and can see where it must go to avoid getting robbed. This allows for maximum
   (c) Out of all my trials, I found an optimal learning rate of 0.3 and an epsilon of 0.01 for when the robot is aware of the thief. This was the rate that caused the robot to consistently avoid the thief across simulations. It learned quickly and didn't have too much randomness taking it in directions that would cause the robot to be penalized.
3. Table Data (x-axis not constant to show initial progress)



I observe a sharp increase in expected reward initially and then an fairly level expectation for the remainder of the simulation.