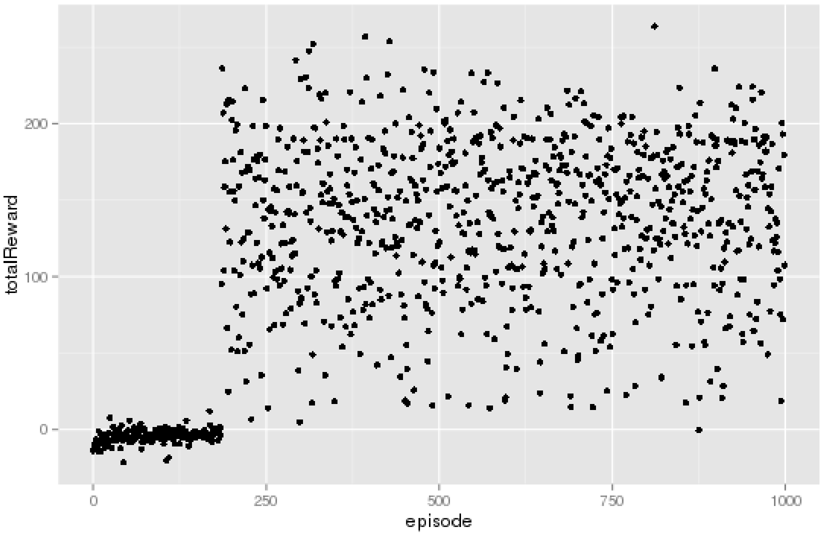
Jaime Mercado

1)

a) Agent's reward stays constant at zero. Agent refuses to cross column of slippery tiles. Similar to situation in lecture, if in a world with terminal state leading to two states, one with reward +1 and another with reward +100, both leading back to the terminal state, say our agent chooses randomly the state with reward +1, then returns to the terminal state and finishes an episode, then without exploration it will continue to pick state with known reward of +1, ignoring completely state with reward +100.

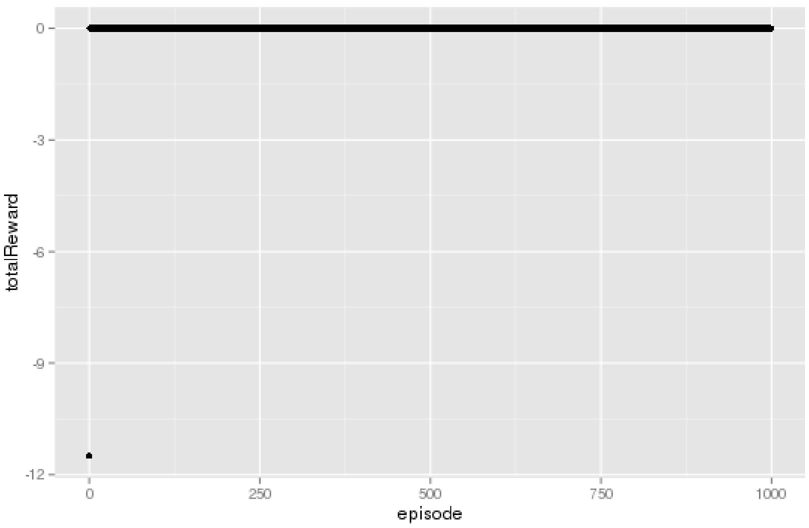
b) Now that the agent will take advantage of exploration via a small epsilon chance of selecting a random action, and not just exploiting the greedy action, the agent might stumble onto a slippery square, an action previously seen as non-optimal in part 1a and therefore completely ignored, thereby improving the estimate of a nongreedy's action value and ultimately produce a greater total reward in the long run. We can see the effect of randomly choosing an action with small probability epsilon in our graph (mp1_pt2_q1b_plot). The agent's total reward initially hovers around 0, sometimes receiving slightly negative total reward value and sometimes receiving a slightly positive reward value, and eventually after the agent explores enough of the world it will find and exploit a much more optimal policy, one that doesn't include completely avoiding the column of slippery squares.

mp1_pt2_q1b_plot

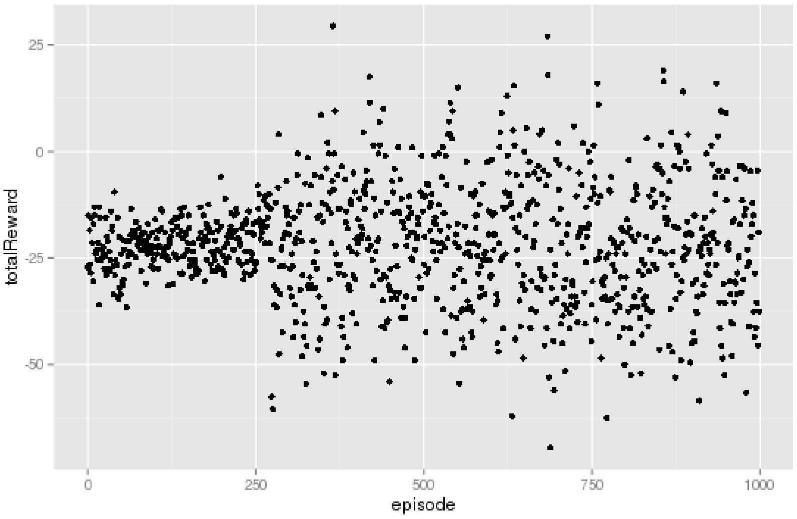


c) Because we've increase the probability of selecting a random action our agent now performs poorly. Exploration is necessary for improvement in our policy, as evident by comparison between graph (mp1_pt2_q1a_plot) and (mp1_pt2_q1b_plot). However, too much random exploration and we begin to introduce too much error in our Q values. Maybe the action randomly picked wasn't the best or maybe we didn't have the right Q value for the s', therefore updating the Q value for s is going to be problematic. We can mitigate this effect by lowering our "confidence", or learning rate (in this case rate=0.01), and see an improvement in total reward in later episodes in graph (mp1_pt2_q1c_plot_imp).

mp1_pt2_q1a_plot



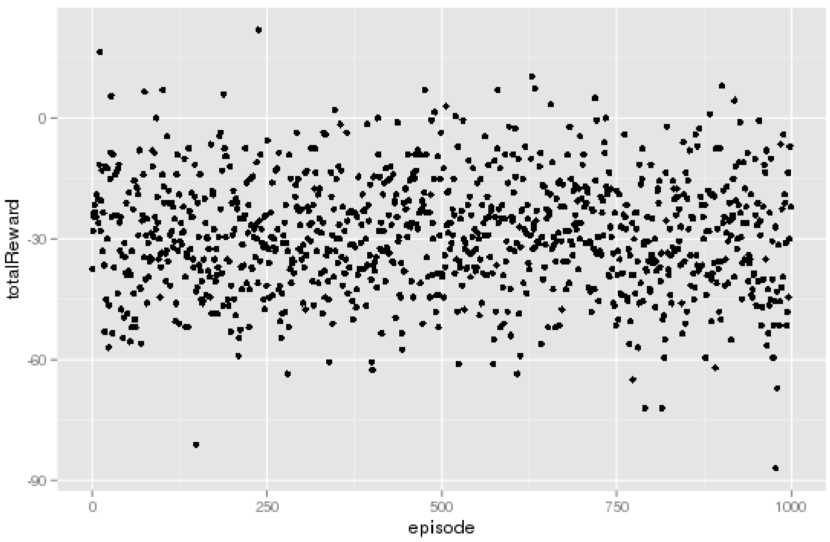
mp1_pt2_q1c_plot_imp



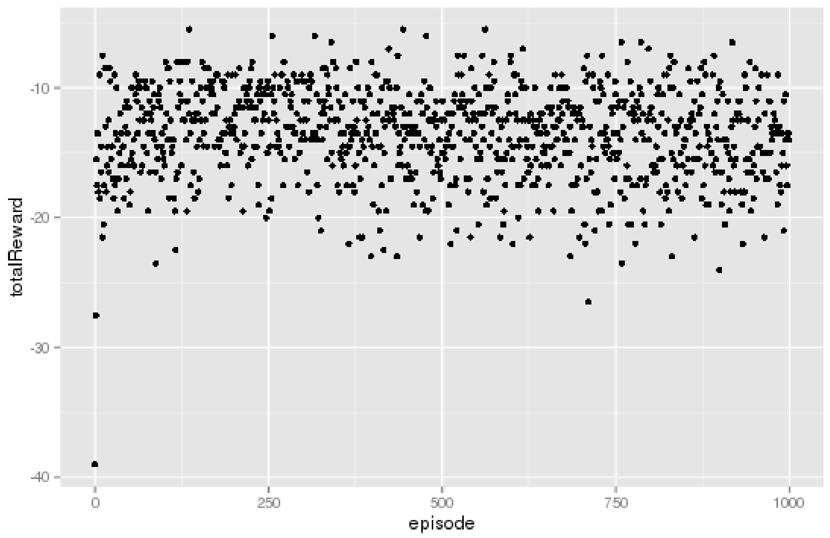
2)

a) In worldwiththief where the agent isn't aware of the thief's location we see a similar behavior to that of our observations in 1c. With openings in the wall of slippery squares between customers the agent will prefer travelling through these openings since they provide a path with a better reward because there is no chance of falling and breaking the packages. However, because the agent has no awareness of the thief, these openings, at (3, 2) and (2, 3), shoehorn the agent into a path that has a chance of incurring a negative reward on the agent. It’s this uncertainty that produces errors in the estimated Q values, similar to the increased randomness of action selected in 1c. This similarity behavior can be seen in the similar shape of graphs (mp1_pt2_q1c_plot) and (mp1_pt2_q2a_plot).

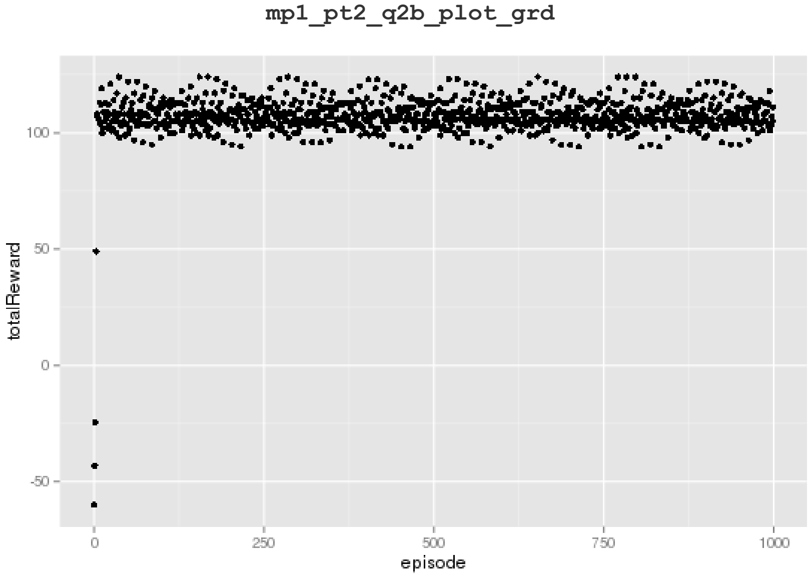
mp1_pt2_q1c_plot

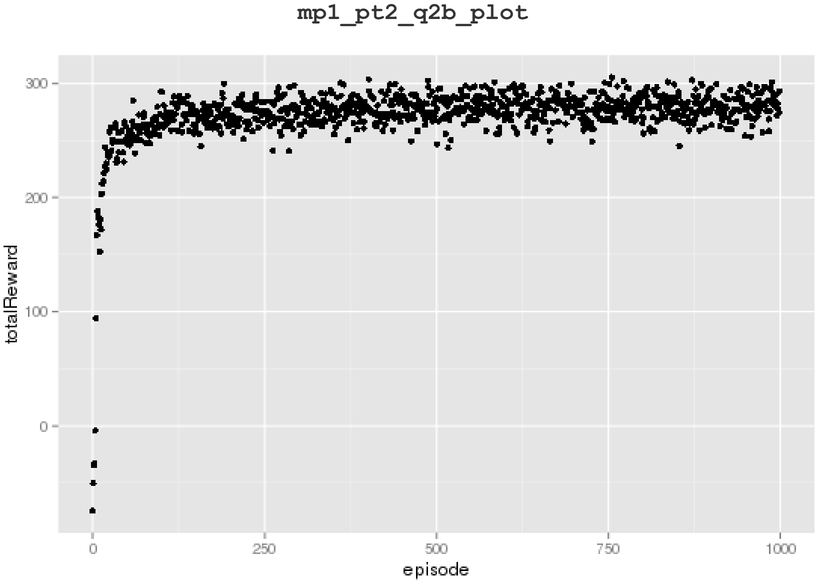


mp1_pt2_q2a_plot



b) In worldwiththief where the agent is aware of the thief's location the agent is basically left to run unobstructed. There are holes in the walls of slippery squares between customers and now the agent even knows of the thief's location, allowing it to avoid both obstacle and properly update the Q values of the world's states. The small epsilon probability of selecting a random action may still cause the agent to behave non-optimal. However, this is required for learning, otherwise we end up with a situation as described in 1a where upon finding an "assumed" optimal policy the agent will continue to select that same found greedy action, as demonstrated in graph (mp1_pt2_q2b_plot_grd). Furthermore, this small epsilon probability of selecting a random action only causes a small fluctuation in total reward per episode, as demonstrated by graph (mp1_pt2_q2b_plot).

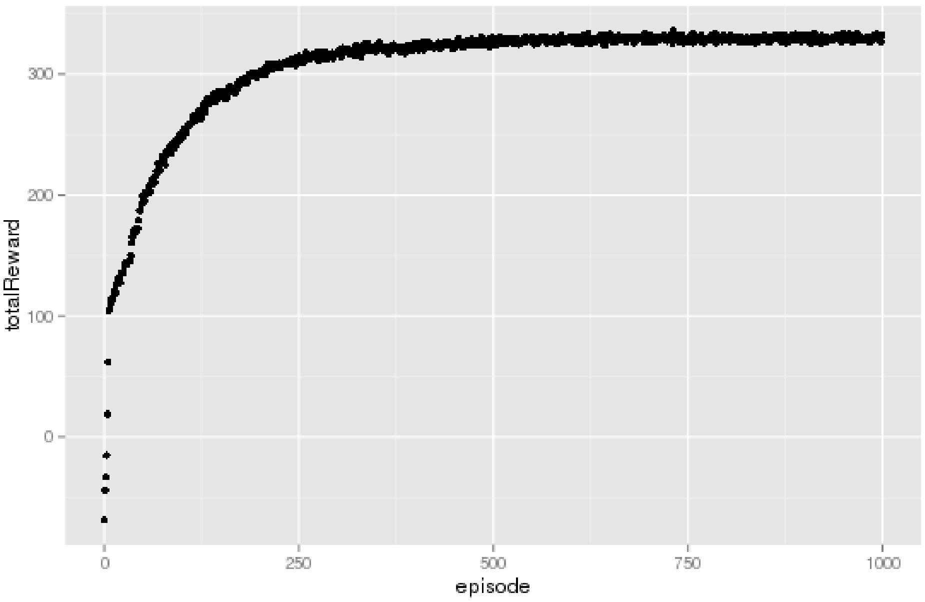




c) From our previous observations we have seen that choosing a random action every now and then promotes exploration and can therefore increases our total reward in the long run. How often we choose a random action can speed up the learning of our agent but too much and it will cause the total reward to vary greatly between episodes. Therefore a small epsilon probability of 0.01 was chosen to allow the agent a slow but stable increase in total reward over the long run. We've also seen how reducing the learning rate can decrease the amount of error we introduce in our Q value estimations and therefore a learning rate of 0.1 was selected.

3) With our selection from 2c we can see that our graph (mp1_pt2_q3_plot) is steadily converging to an optimal total reward.

mp1_pt2_q3_plot


totalReward
episode