CS440 MP 1 Pt 2 Brian Xue/bxue2

1a. Basically entirely 0s, except episode one which is usually some negative number. This makes sense, since after failing test 1, the policy will make it less likely for the robot to repeat its original steps, and make it more likely to start off by walking into a wall. And since there's no randomness, it'll never stop walking into the wall once the policy tells it to.

b. The first 20ish episodes are either negative or low positive numbers. This makes sense, since the robot doesn't learned a clear policy yet. After that, the reward is usually in the low 3-digits (between 100 and 300), with the occasional dip back down into the 2 and 1 digit range, meaning it's mostly figured out its policy and is helped/messed up every once in a while due to the randomness.

c. At episilon = 0.5, the reward was consistently negative, usually between -10 and -40. This is probably because the randomness is so high. With a 50% chance to go in the wrong direction, the robot is going to end up slipping a lot more often since it takes a lot more steps to reach any of its destinations. When testing at epsilon = 0.8, the rewards were even lower, usually in between the -70 and – 95 range, which makes sense given the randomness is even higher. And when testing at epsilon = 0.3, the rewards, while smaller than epsilon = 0.1, were at least in the positive double digits.

Interestingly though, the rewards when epsilon = 1.0 (complete randomness) managed to be better than episilon = 0.8. I assume this would be because a completely random path isn't as likely to slip due to running into walls and being less likely to go back and re-stock on packages.

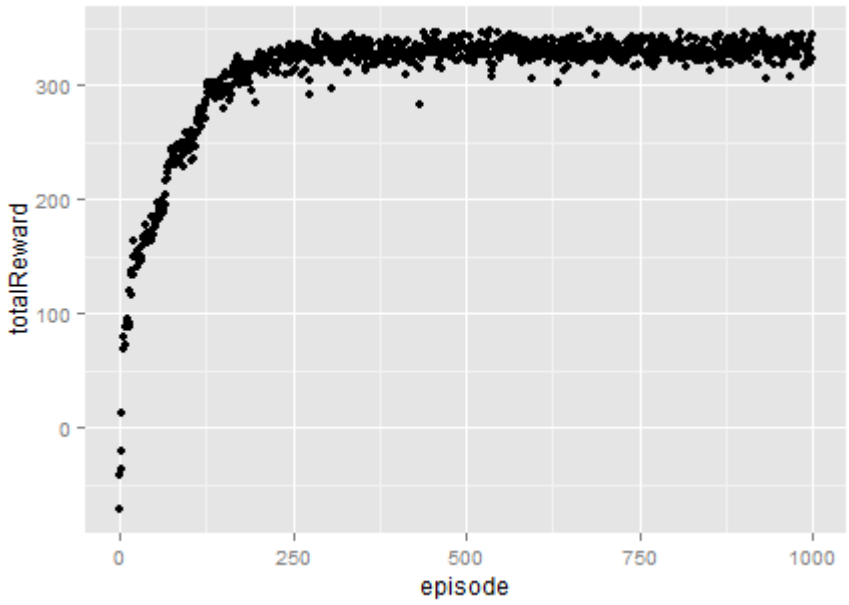
2a. At epsilon = 0.05 and thief location off, the reward is consistently in the low negatives, between -5 and -20. This is probably due to the fact that the thief can't be seen. The robot's policy is always going to be outdated since the thief is constantly moving, and since the robot is going to change its policies after running into the thief, it's never going to settle on one "correct" policy either.

b. With the thief's location known, the reward is significantly higher, starting low but quickly ramping into the 200s, where it stabilizes and slowly climbs up to around 270-300. This massive difference in reward is due to the fact that the robot knows where the thief is, and has built up policies accounting for its location. Presumably, the policy chooses the robot's direction based off not just its own position but the robbers (there's significantly more numbers in the policy.txt compared to the previous trials). With the chance of running into the thief significantly reduced, the robot is much more likely to be successful with its deliveries, hence the high scores.

c. After raising/lowering both epsilon and the learning rate/alpha, I've decided to stick with epsilon = 0.01 and learning rate = 0.15, where rewards average between 330 – 350. Epsilon seems to be better when it's lower, presumably because the randomness is really only helpful when the robot is still learning, and isn' really wanted after the optimal path has been found. The learning rate really didn't have quite as much of an effect. Presumably this is because we're using a high amount of episodes, and speeding up/slowing down the learning process doesn't matter as much in the long run.

3. For the most part, the graph starts with a steep upwards slope and then suddenly plateaus between episodes 100 and 250. After that, there's a slight upwards slope still if you only look at the average, but its pretty faint, as the robot has basically stopped learning anything really useful from that point on and progress drastically slows down.

Epsilon = 0.01, Learning rate = 0.15, 1000 episodes



Epsilon = 0.01, Learning rate = 0.15, 10000 episodes