Jordan Kravitz- jkravit2

CS 440 MP1 Experimentation

1) WorldWithoutThief

A) The first episode run has a negative reward (-8.0) while all other episodes run have a reward of 0.0. This is the case because once the robot has tried an action from a state the first time, the data Q[state][action] begins to be weighted towards the true value (as opposed to the initial value of 0 for all state-action combinations). In the first episode, the robot tries all possible actions that pass through the slippery zone. Due to the negative weight observed in the first iteration to cross the slippery boundary, on the second iteration (and all others following) the robot stays in the boundaries of the non-slippery area (Q value of 0) because it does not want to take any 'risk' (error) to go and explore the actions beyond the slippery region.

B) In this case, the robot successfully learns how to pick up and drop off packages with a nearly optimal policy. The episodes in this trial have a general trend of increasing in total reward, but there are exceptions to this rule quite commonly. The main difference between what occurred in this trial when compared to the previous trial, is that the error allowed the robot to navigate to actions it knew would be hazardous in return for the possibility of finding reward states later on (which it does). As this runs more episodes, these hazardous states can even become positive due to the fact that the robot learns they are on the path to a reward state.

C) The general trend in the case of a high element of randomness is to have episodes with negative rewards- however there are some episodes with positive rewards. This is the case because the robot arbitrarily picks a path half the time (depending on 'e'). Because of this, it becomes difficult to build a successful path even though it knows what the 'right' action most likely is. Due to these unsuccessful episodes, a garbage policy is created which just leads the robot into trouble. A great analogy to what is going on here is that if someone were to give you directions that were somewhat accurate and you followed them half the time and arbitrarily chose a direction the other half of the time- it becomes extremely difficult to get where you want to go.

2) WorldWithoutThief

A) The performance of the agent is very poor- it rarely has a positive episode and the simulated policy is not successful. Because there is no knowledge of the thief, there is a more limited set of states for the simulator to run and create an optimal policy. Due to this, it must blindly choose to move into a thief territory from a certain state regardless of where the thief is. This is an often unsuccessful endeavor (as evidenced by the episodes)- so it must either go through an extra slippery region (high probability of dropping packages) or thief ridden territory. Neither of which have a high probability of success which is why this is so unsuccessful.

B) The performance of the agent becomes quite good in this case- the later episodes are all positive and the simulated policy is very successful. Because there is now knowledge of the thief, the simulator uses these extra states to learn how to avoid the thief which is a trivial task in terms of state actions. Additionally, assuming the robot has properly learned how to avoid the

thief, the only “risk” it takes is going through the less slippery area when compared to the robots having no knowledge which must go through the less slippery area as well as either a thief zone or the higher slippery area.

C) The values that I achieved were .**055 for the best learning rate** and **.0095 as the error**. My approach was fairly straightforward- I did a binary search on each value individually to close in on roughly the correct value through a series of trials. Because they are not completely independent variables, I then manually tuned the values to result into what I believe is a near optimal tuning for these values.

3) When a single trial is plotted, the error is very apparent- every episode which veers off from the expected trajectory of the trial is very pronounced. When the trials are averaged together, the error is effectively averaged out. This is because the error in a certain trial can be both positive and negative error. The early episodes unsurprisingly have a higher slope than the later episodes in both cases. This is due to the fact that it is much easier to find a vastly better path than the last episode early on rather than at a later trial. What is pretty surprising to me is how early on a "pretty good" policy is constructed.