Isabelle (Boo) Chalhoub - u0678302
Karla Kraiss - u0830999
August 29, 2017

Lab Report - A1

1. Introduction

In this lab we created an agent that explores a four by four grid of cells by either walking forward, turning right, or turning left. One of these actions is decided by the agent completely by random. Some of the cells in the grid are pits which halt the agent if it enters that cell. Not falling into a pit before the max number of steps is reached is a semblance of success that we are looking to measure. Another cell in the grid is gold which we look for later as another semblance of success for our trials.

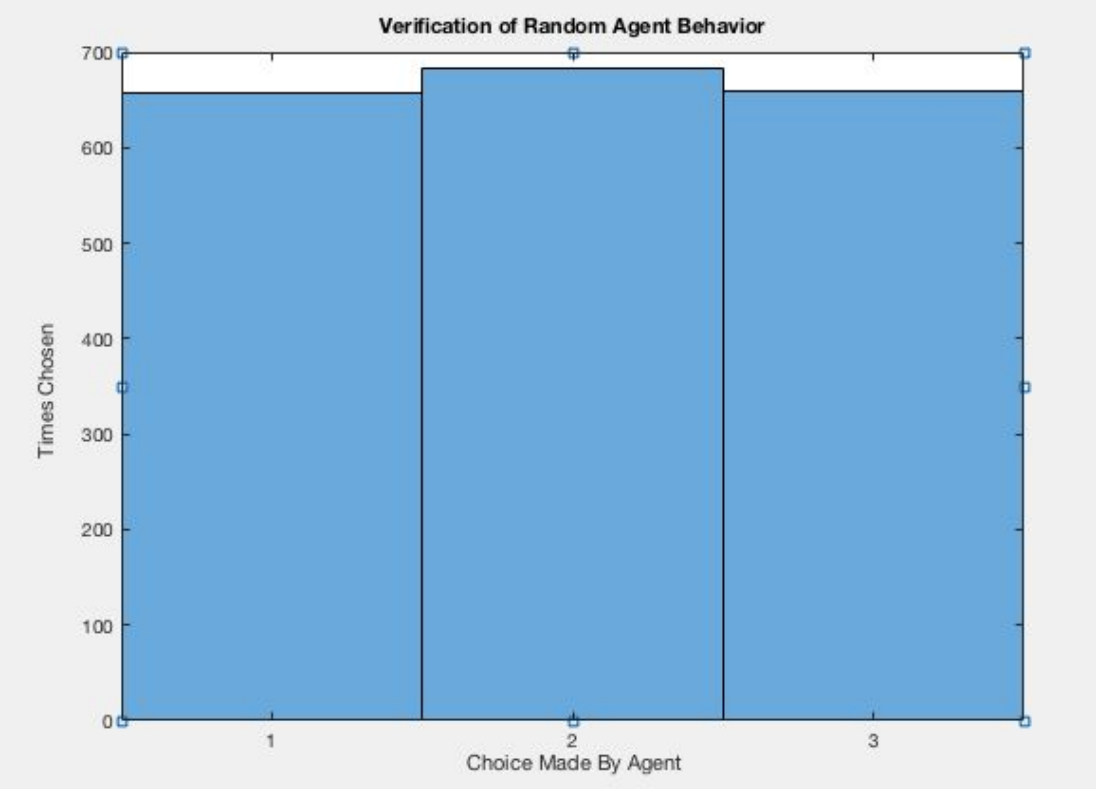
- On average, how successful is our random agent?
- Does varying the maximum number of steps the agent can take affect the probability of success?

2. Method

Agent: “action - ceil*((3)*rand(1));”
Driver: The method used for the driver is simply to generate a large number of samples and compute the mean, variance and confidence of the result with a maximum of 50 steps. Another way we computed the mean, variance, and confidence was to run 2000 trials for each increasing step allotment starting at 1 ending at 100. We store all of this data in a struct so we may calculate and plot each variable uniquely.

3. Verification of Program

Our agent picks a number between 1 and 3 inclusive at random. We use the MatLab rand() function with some tweaks and should verify that this is truly giving us a random choice. Below is a histogram of 2000 choices made by the agent in succession:



4. Data and Analysis

First we ran 2000 trials at 100 steps to get a baseline average. Next we ran 2000 trials for each incrementing step starting at 1 and going to 100. We noticed as the maximum number of steps increased, the average trials the explorer survived evened out (Figure 1). After about 50 maximum steps was passed the increase tapered off. The same is true for success rate as maximum steps increased (Figure 2).

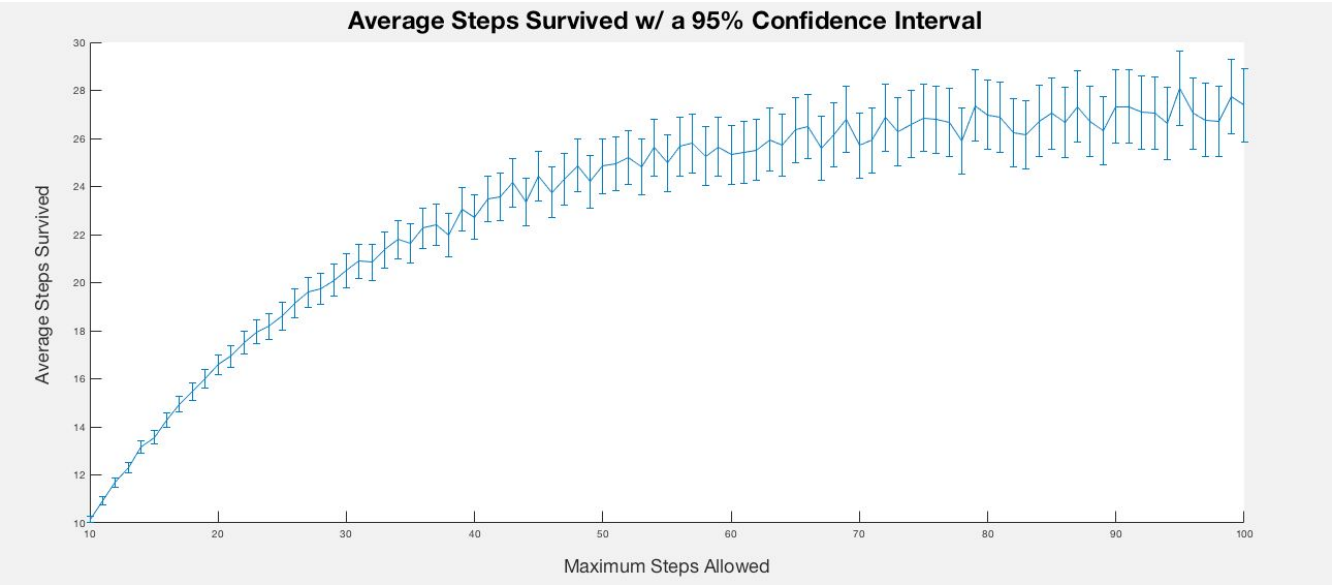


Figure 1

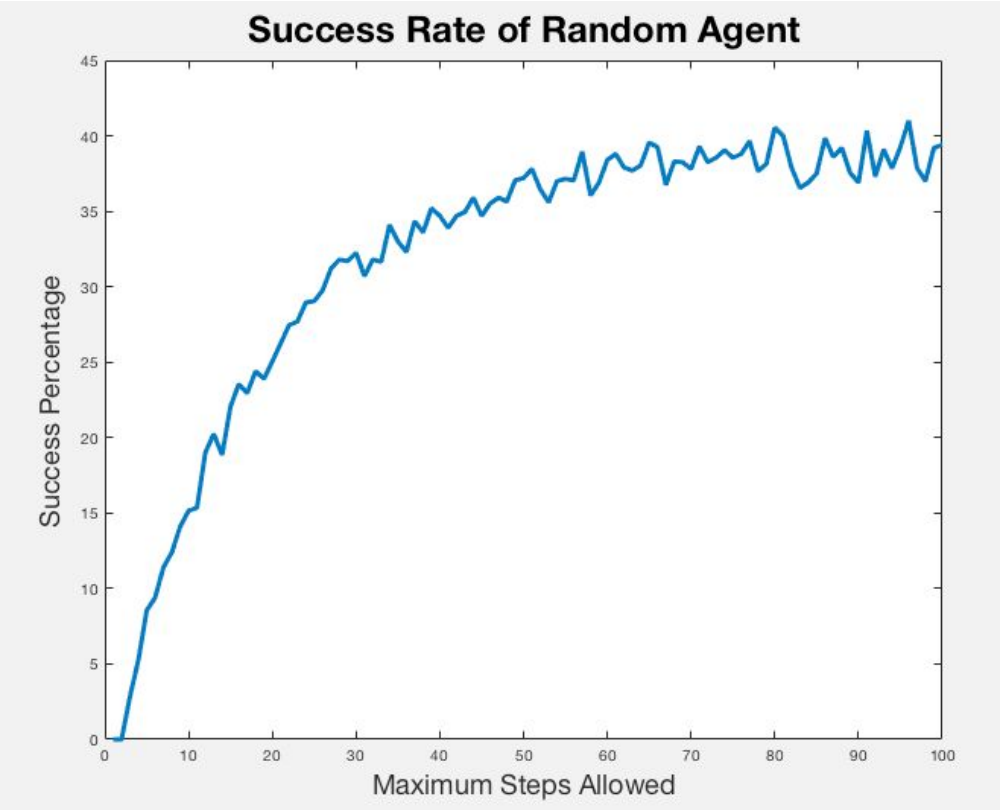


Figure 2

5. Interpretation

Increasing the maximum number of steps allowed to be taken by the agent made the survival rate much higher between a max value of 10 and 50. Once the agent is permitted to take 50 steps or more, the success rate changes much less. It seems to approach a limit of around 28 steps before the agent dies, no matter how many steps you permit it to take. This means that on a grid this size, with this number and placement of pits, the agent will not survive more than 30 steps when it is choosing its actions randomly, even with seemingly infinite steps. I would like to run these experiments on a differently formatted grid with the same number of pits to see if the placement has any bearing on this limit and how much it will change our results.

The percentage of times the agent found gold also reached a limit of around a 40% success rate, no matter the maximum number of steps allowed beyond 50. Not only does this mean that 40% of the time the agent was able to find gold, the agent was able to find the gold before dying 40% of the time because of how our system is structured. This is much higher than I expected and my guess is because the agent starts at [1,1] and only has to reach [2,2] in order to get the gold. My guess is that if we placed the gold farther away on the board this percentage would decrease.

6. Critique

Simulating a random agent within Matlab was a large concept we learned through writing the driver code. We also determined that having a larger number of trials would even out our data set to a more uniform result. A better way to set up this experiment would be to pose a (null) hypothesis, that our agent is truly random, and then apply standard statistical methods to reject or accept the hypothesis.

7. Log

Boo's Log: (Odd Sections)

I spent about half an hour programming alone and familiarizing myself with Matlab. In this time I wrote the one line of agent code and stubbed out what our driver function needed to do. Karla and I pair programmed for about 3 hours finishing up the driver and pulling all of the statistics. I spent around 1.5 hours on my half of the lab report.

Karla's Log: (Even Sections)

Boo and I coded the driver functionality together pair programming style. I navigated the entire time while she drove and looked up methods we could use or how to use Matlab to implement our solution. This process took about 3 hours when incorporating the work we did to generate our plots. Afterwards I spent around an hour and a half on my lab report. I also educated myself on general Matlab functionality for roughly half an hour.