Report on Machine Learning Part II (Q-learning)

Name: Imaculate Mosha

Student number: MSHIMA001

Date: 20th October 2015

Introduction

This report concerns the Practical Assignment on Artificial Intelligence using Q-learning(Reinforcement learning). As specified in the instructions.

* The provided code in CQLearningController.h and CQLearningController.cpp were modified to implemented the Q learning algorithm.
* One qtable was used for all sweepers, this took considerably less time to learn the correct behavior compared to having a qtable for each sweeper.
* The algorithm was tested with 20 Simulations for each environment after the sweepers learned the correct behaviour. The outcome measured were the Average number of mines collected per sweeper and the total number of deaths of sweepers per simulation. These numbers were averaged out over the 20 simulations.
* Data was analysed and conclusions were drawn.

Methodology

The Q table is implemented as a 2-Dimensional vector, where each inner vector is a vector of all actions(N,E,W,S) for each position( state).

The following methods of CQLearningController class were modified using the given pseudo code:

void CQLearningController::InitializeLearningAlgorithm(void)

{

Set the size of Q-Table and initalise all values to 0

}

double CQLearningController::R(uint x, uint y, uint sweeper\_no)

{

If the sweeper died(supermine or rock) return -100

Otherwise return -1 //(even with mines to allow exploration)

}

bool CQLearningController::Update(void){

For each sweeper do

observe current state (x,y)

select action (direction) with highest historic return (q-value). If there are multiple maximumum, choose randomly among them to avoid dominance of the default action (EAST)

Move the agent to the new state (based on the action selected)

For each sweeper do

observe new state

Update the Q(s,a) using function R(s,a)+ discountFactor \*max(Q(nest state, all actions))

}

Results

Environment 1 (40 mines, 10 super mines, 1 rock)

|  |  |  |
| --- | --- | --- |
| Iteration Number | Average Mines gathered per sweeper | Deaths |
| 16 | 1.267 | 0 |
| 17 | 1.267 | 0 |
| 18 | 1.267 | 0 |
| 19 | 1.267 | 0 |
| 20 | 1.267 | 0 |
| 21 | 1.267 | 0 |
| 22 | 1.267 | 0 |
| 23 | 1.267 | 0 |
| 24 | 1.267 | 0 |
| 25 | 1.267 | 0 |
| 26 | 1.267 | 0 |
| 27 | 1.267 | 0 |
| 28 | 1.267 | 0 |
| 29 | 1.267 | 0 |
| 30 | 1.267 | 0 |
| 31 | 1.267 | 0 |
| 32 | 1.267 | 0 |
| 33 | 1.267 | 0 |
| 34 | 1.267 | 0 |
| 35 | 1.267 | 0 |
| Average | 1.267 | 0 |

Average Mines Gathered over all sweepers per Iteration = 1.267 \*30 = 38 mines

The numbers stabilise (number of deaths start to becomes 0 sand number of mines gathered is maximum) at about iteration 20. This is the number of iterations needed to learn the behaviour.

Enviroment2 (25 mines, 25 super mines, 1 rock)

|  |  |  |
| --- | --- | --- |
| iterations | Average Mines gathered per sweeper | Deaths |
| 20 | 0.8 | 1 |
| 21 | 0.8 | 0 |
| 22 | 0.8 | 0 |
| 23 | 0.8 | 1 |
| 24 | 0.8 | 0 |
| 25 | 0.8 | 1 |
| 26 | 0.8 | 0 |
| 27 | 0.8 | 1 |
| 28 | 0.8 | 2 |
| 29 | 0.8 | 0 |
| 30 | 0.8 | 0 |
| 31 | 0.767 | 0 |
| 32 | 0.8 | 0 |
| 33 | 0.8 | 0 |
| 34 | 0.8 | 0 |
| 35 | 0.8 | 0 |
| 36 | 0.8 | 1 |
| 37 | 0.8 | 0 |
| 38 | 0.8 | 0 |
| 39 | 0.8 | 1 |
| ⌈Average | 0.79835 | 0.4 |

Average Mines Gathered over all sweepers per Iteration = 0.79835\*30 = 23.95 ~ 24 mines

The numbers stabilise (number of deaths start to becomes 0 sand number of mines gathered is maximum) at about iteration 15. This is the number of iterations needed to learn the behaviour.

Environment 3(10 mines, 40 super mines , 1 rock)

|  |  |  |
| --- | --- | --- |
| iterations | Average Mines gathered per sweeper | Deaths |
| 20 | 0.33 | 0 |
| 21 | 0.33 | 0 |
| 22 | 0.33 | 0 |
| 23 | 0.33 | 2 |
| 24 | 0.33 | 0 |
| 25 | 0.33 | 0 |
| 26 | 0.33 | 0 |
| 27 | 0.33 | 1 |
| 28 | 0.33 | 1 |
| 29 | 0.33 | 1 |
| 30 | 0.33 | 0 |
| 31 | 0.33 | 0 |
| 32 | 0.33 | 0 |
| 33 | 0.33 | 0 |
| 34 | 0.33 | 1 |
| 35 | 0.33 | 2 |
| 36 | 0.33 | 0 |
| 37 | 0.33 | 0 |
| 38 | 0.33 | 2 |
| 39 | 0.33 | 0 |
| Average | 0.33 | 0.5 |

Average Mines Gathered over all sweepers per Iteration = 0.33\*30 ~ 10 mines

The numbers stabilise (number of deaths start to becomes 0 sand number of mines gathered is maximum) at about iteration 13. This is the number of iterations needed to learn the behaviour.

Analysis

I plotted the graph of the average values of sweeper’s death and number of mines collected. for

The following graph was obtained.

From the graph, it is clear that the number of mines collected decreases from test environment 1 to environment 3 and the number of deaths increases accordingly. This is explained by the decrease in number of mines and increase in number of super mines from test environment 1 to 3. Moreover it was observed that rewarding positive value for collecting mines reduced exploration compared to a negative reward that is more than the reward for death. It is for this reason that a reward of -1 was given when there was no death. In addition to that with one Q-Table the number of iterations (~15 iterations) to the correct behaviour was considerably less than when each sweeper had its own Q-table (~ 170 iterations). This is explained in that with a shared table, all the sweepers have shared knowledge of the environment.