Homework 2

Zhengyang Ling(zl351), Yalin Liu(yl1255), Yujia Liu(yl1261)

March 22, 2021

3

(1)Representation:

We name our project as Automatic MineSweeper (AMS). The board in AMS is represented by a matrix with dimension dim and the number of mines n. the inferred information between cells is represented in the local methods solver(2.1),improved-agent(2.1 part1), jiekou(2.2 part2). -1 represent the bomb. -2 represent the flag. 0-8 for mines numbers around. 9 for the cell that not open yet.

(2)Inference:

We generate a matrix full of 9 represent the unknowen cell. When we decided to open the cell, we will send the coordinate of this cell to the answer and ask the value of this coordinate.We update the current state of knowledge based on a clue is to add the new information into the minesweeper matrix. Yes, AMS deduce everything it can from a given clue before continuing

(3)Decisions:

1.At first we will make decision base on the single cell. When the number of mines and flags around this cell is equal to the value of this cell, we will open the unopen cell arround this cell. If the number of unopen cells plus the number of mines and flags around this cell is equal to the value of this cell, we will flag these unopen cells. 2.1 that I mentioned in code and readme

2.If the traditional method(step1)cannot open all the cells, then we use improved-agent to solve for a more difficult circumstance which requires inference between two cells with numbers from 1 to 8 with unknown cells around them to decide whether some unknown cells are safe or not. In improved-agent, we should find two adjacent cells with number from 1 to 8 and decide whether some of the cells around them are numbers or mines. Therefore, the two cells satisfy two conditions: first, the two cells are adjacent; second, there are unknown cells around those two cells. Then, pass the coordinates of those two cells as parameters in the method logic-decide. In logic-decide, let g1 and g2 be the two coordinates indicated above. The value of g2 in minesweeper matrix should be greater or equal than that of g1, or the unknown cells around g2 should be more than those of g1. Let a1, a2, and ac be three lists that store the indexes of unknown cells only around g1, g2, and common area of g1 and g2. If the value of a1 is equal to the value of a2, then nothing can be inferred by those two coordinates, so return the matrix. Let m1, m2, and mc be the number of mines in areas a1, a2, and ac, respectively. Let s1, s2 be the suspected number of mines in area (a1+ac) and area (a2+ac). Let difference be the list of indexes that are in a2 but not in a1. If the length of list a1 is 0 and s1 is equal to s2, then it means that the unknown cells around g1 is totally included in those of g2, and the mines are restricted in the unknown areas around g1. Thus, the remaining cells in a2 (the indexes are stored in difference) must be safe, and we should open those cells. If s2−s1 is equal to the length of difference, then it means that the number of mines that are not in the common area are all in difference, so the indexes stored in the difference are all mines. Thus, we flag are positions in difference as mines. If this improved-agent still does not open all cells, it means that there are situations that we need to infer from more than two cells. 2.2 part 1 that I mentioned in code and readme

3. Then, for every open cell(not bomb or flags), we put the unopen cells aroud this cell and the number of bombs in these un open cells into one set. If any of these set is the subset of another this kind of set, we will divided the bigger set into two small set. And also use the number of bombs in the bigger set minus the number of bombs in the smaller set to get the number of bombs in the new set. If the number of bombs in the new set is 0, we will open all the cell in the new set. If the number of bombs in the new set is equal to the number of cell in the new set, we will mark all of these cells. 2.2 part 2 that I mentioned in code and readme

4. If we cannot get the cell to open from improve. We will calculate the probability of each cell of each set. If the probability that one cell have bomb is the least in all set and is less than the all unknowen bomb/all unopen cell, we will pick one cell randomly in this set. Else, we will pick randomly from all unopen cells.

(4)Performance: AMS makes great decision. It is a better player than any peopele in our group.It does not surprise us. After careful consideration, we will find that AMS's decision is correct and almost optimal. And AMS is much faster than our human brain. It can make that decision basing on (3)Decisions that I mentioned. For possibility part. If the probability that one cell have bomb is the least in all set and is less than the all unknowen bomb/all unopen cell, we will pick one cell randomly in this set. Else, we will pick randomly from all unopen cells.
The graph make sense.AMS become "hard" when q>=0.8. AMS beat simple algorithm when p>0.1. The simple algorithm is almost same with AMS when q<=0.1 because we do not need the improve part at all, we can judge all the possibilities directly based on the a single grid(that is simple alg) without going into 2.2part. When q>=0.2, almost all the time we run AMS, AMS able to work out things that the basic agent cannot.

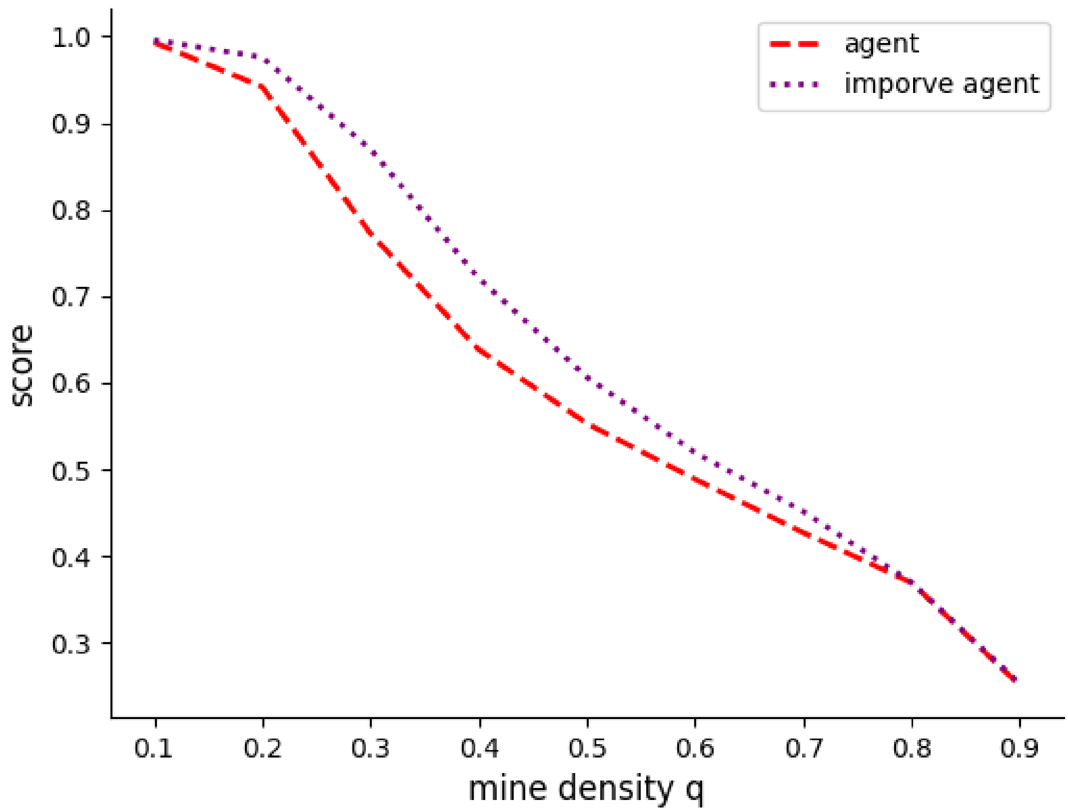

Figure 1: performance:common agent vs improve(without global and better decision) 100 tests/0.1q

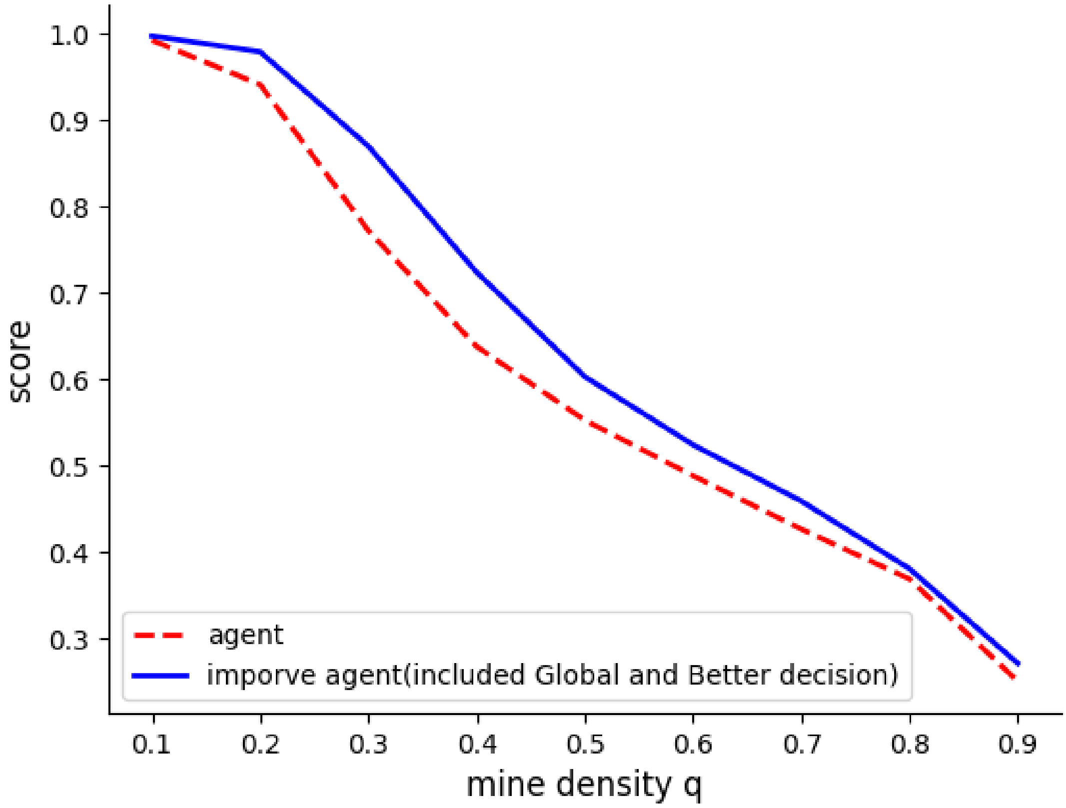

Figure 2: performance:common agent vs improve included global and better decision(AMS) 100 tests/0.1q

(5)Efficiency:

If there exists a situation that we must guess among some unknown cells, our algorithm may not choose the best cell to open. We just performed a simple calculation of the probability and will not use complicated formulas to always choose the best grid. I think this is an implementation constraint. If I can improve my algorithm, I may write another function that has a calculation to open the best unknown cell which we can get as much information as possible or which the probability of being a mine is as small as possible.
One of implementation constraints is that we will generate many new lists for every steps openning the cell. It can be improve by repeat use the list that created before. And because of the memory that python can used is limit, it cannot generate the gif for mineswapping(if dimension is too large)(but the other function is normal).

4 bonus:

global information When AMS has marked all mines, AMS will open all the remaining grids and end the mission early. When AMS opens all safety grids (total number of grids - number of mines = number of safety grids) AMS will mark all the remaining grids as flags and end the mission early which get higher score. When p>=0.8, global information make score higher frequently. When p>= 0.9 global information make score higher greatly.Especially the matrix of 20*20 with 399 mines or 10*10 with 99mines. It works really well.

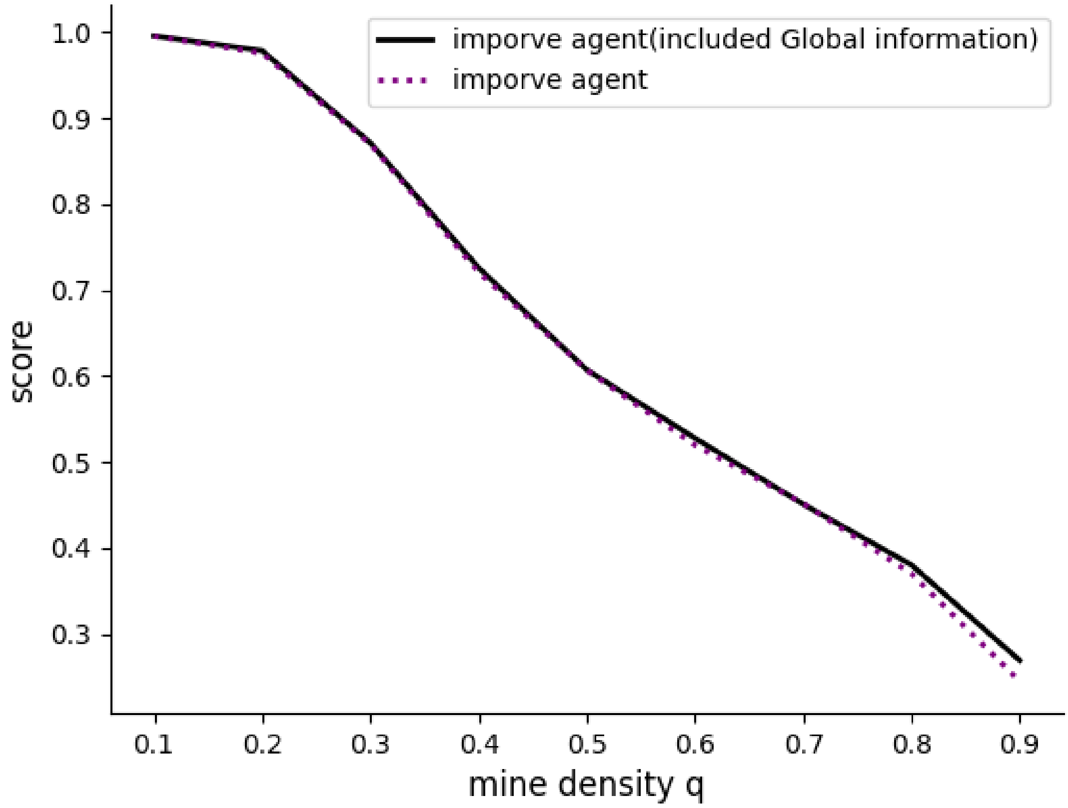

Figure 3: improve vs improve included globla information

better decision The major part have been written in the forth paragraph of decision part. In there, I will included some justification. I use this part only when there is a cell which has probability that is less than the all unknowen bomb/all unopen cell, because we know the cell have a huge advantage(I mean the cell compare to the other cell will be more likely reveal imformation) only if the probability that is less than the all unknowen bomb/all unopen cell. If it doesn't have a huge advantage, I think randomly pick will be better. Because pick randomly in all unopen may reveal more information, the unopen cell that around the open cell may reveal less information, which will cause we have to pick in all unopen cells randomly. So compare to have to pick randomly in all unopen cell, I think that it will be better to pick in all unopen cell if we don't know where have a huge advantage.

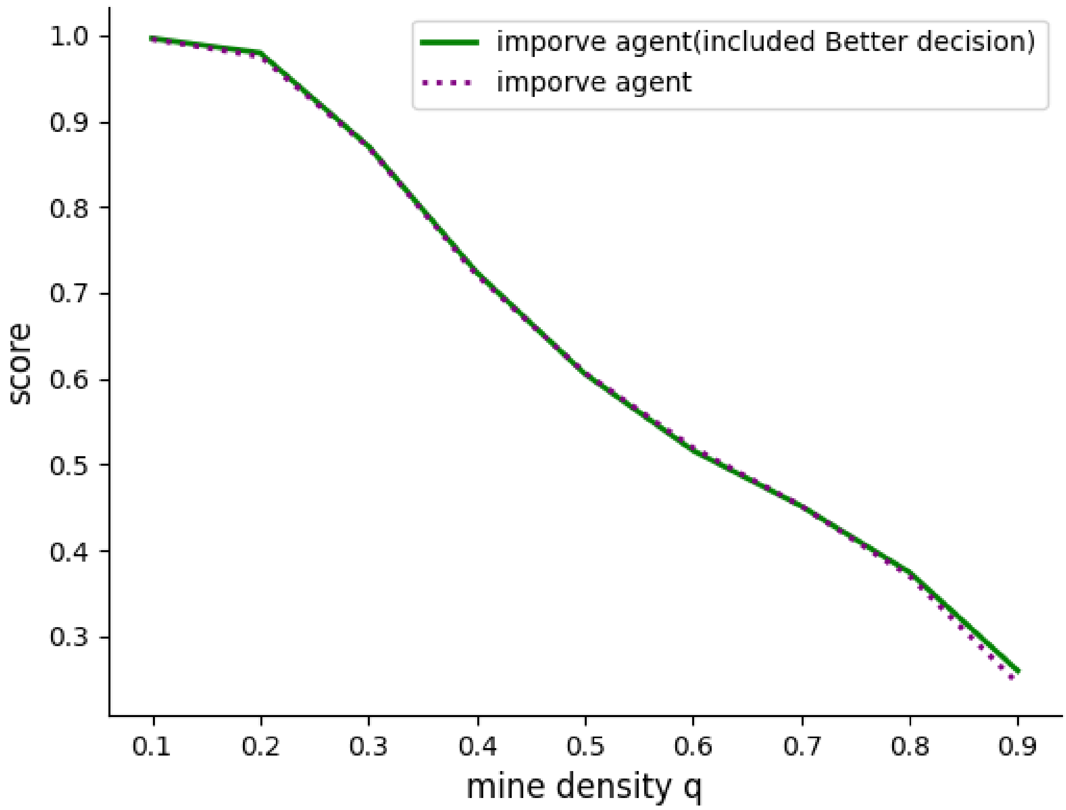

Figure 4: improve vs improve included better decision 100 tests/0.1q

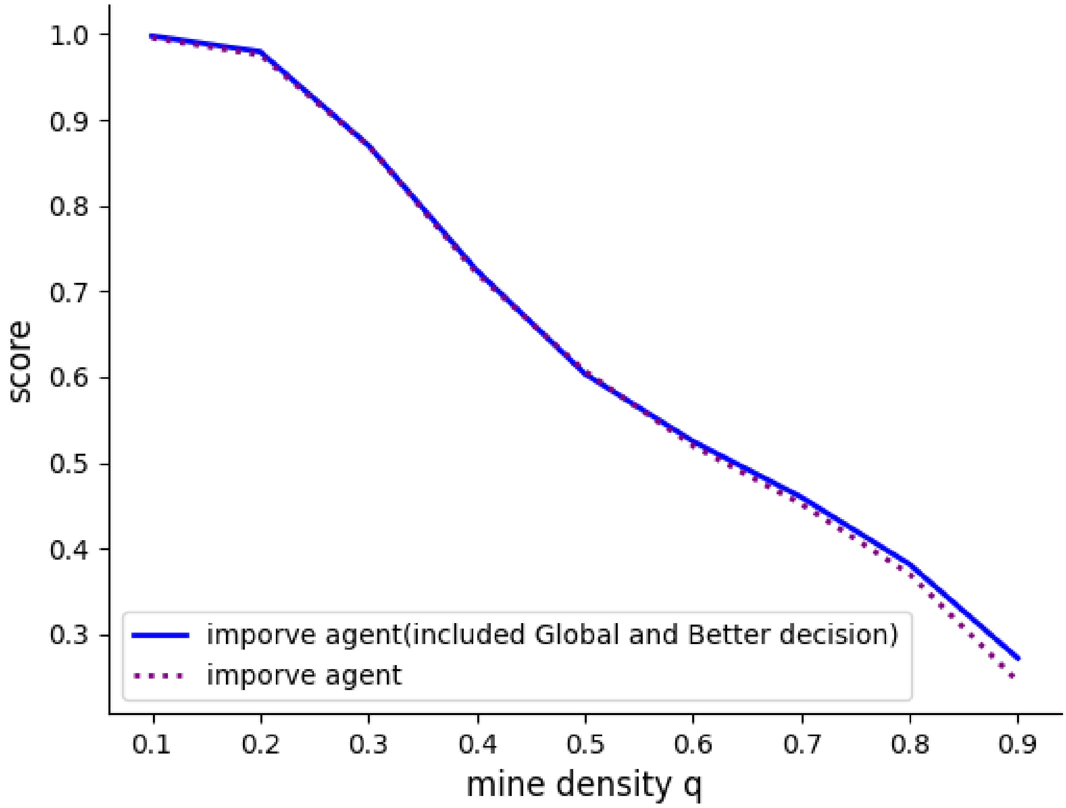

Figure 5: improve vs improve included global and better decision 100 tests/0.1q

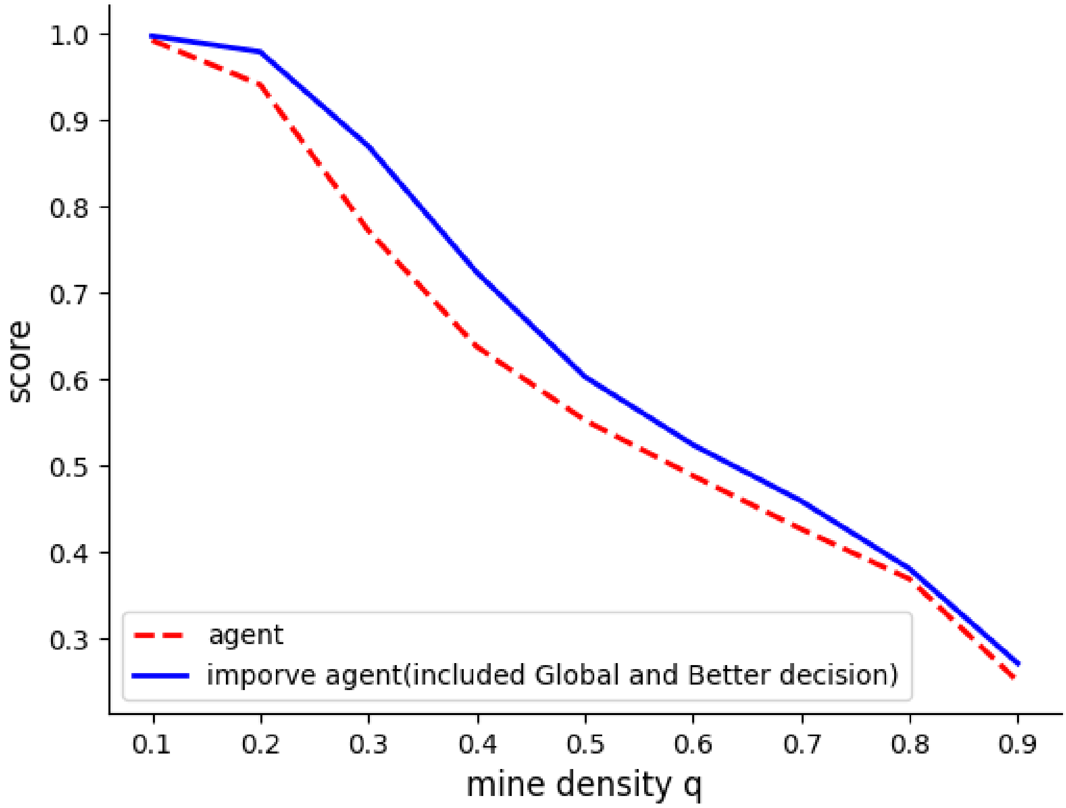

Figure 6: common agent vs improve included global and better decision(AMS) 100 tests/0.1q

combine together