**MIE567 Assignment 1**

Jade Khiev <SN>

Nikita Kochnev <SN>

Padmanie Maulkhan <SN>

Kwan Hon Benjamin Leung 999613556

Deterministic Gridworld

Part A

2. The state space is represented as s(x)(y), which refers to the x-th row and y-th column beginning with the top left. Thus, the set of feasible state spaces are:   
   {s11, s13, s14, s15, s21, s23, s24, s25, s31, s32, s33, s34, s35, s41, s45, s51, s52, s53, s54, s55}

In this problem, we begin at s51, and our goal is to reach s13.

The action space is {U, D, L, R}, which represents the action of going up, down, left, and right respectively.

1. The transition probability matrix is dependent on the action taken:

This is the set of actions that result in not moving, because they would have moved into a blocked square:



This is the set of actions that result in moving to a different (i.e. non-blocked) square:

|  |  |  |  |
| --- | --- | --- | --- |
| Origin state (s) | Action (a) | Destination state (s') | P(s'|s,a) |
| s11 | D | s21 | 1 |
| s14 | D | s24 | 1 |
| L | s13 | 1 |
| R | s15 | 1 |
| s15 | D | s25 | 1 |
| L | s14 | 1 |
| s21 | U | s11 | 1 |
| D | s31 | 1 |
| s23 | U | s13 | 1 |
| D | s33 | 1 |
| R | s24 | 1 |
| s24 | U | s14 | 1 |
| D | s34 | 1 |
| L | s23 | 1 |
| R | s25 | 1 |
| s25 | U | s15 | 1 |
| D | s35 | 1 |
| L | s24 | 1 |
| s31 | U | s21 | 1 |
| D | s41 | 1 |
| R | s32 | 1 |
| s32 | L | s31 | 1 |
| R | s33 | 1 |
| s33 | U | s23 | 1 |
| L | s32 | 1 |
| R | s34 | 1 |
| s34 | U | s24 | 1 |
| L | s33 | 1 |
| R | s35 | 1 |
| s35 | U | s25 | 1 |
| D | s45 | 1 |
| L | s34 | 1 |
| s41 | U | s31 | 1 |
| D | s51 | 1 |
| s45 | U | s35 | 1 |
| D | s55 | 1 |
| s51 | U | s41 | 1 |
| R | s52 | 1 |
| s52 | L | s51 | 1 |
| R | s53 | 1 |
| s53 | L | s52 | 1 |
| R | s54 | 1 |
| s54 | L | s53 | 1 |
| R | s55 | 1 |
| s55 | U | s45 | 1 |
| L | s54 | 1 |

1. We define the Reward function as the following:

for all other

We define a high reward (+25) for reaching the goal state of s13 because we want to be incentivized to go there.

We assign a cost of -1 to any regular feasible move so as to disincentivize taking too many moves.

1. See Gridworld.py

Part B – Value Iteration

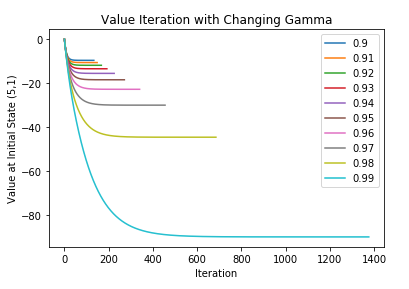


Figure 1. Initial state values against each iteration plotted for varying values of gamma.

Table 1. Policy obtained for each state (rows) at each value of gamma (columns).



Part C – Policy Iteration

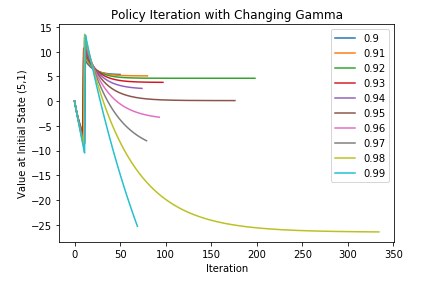
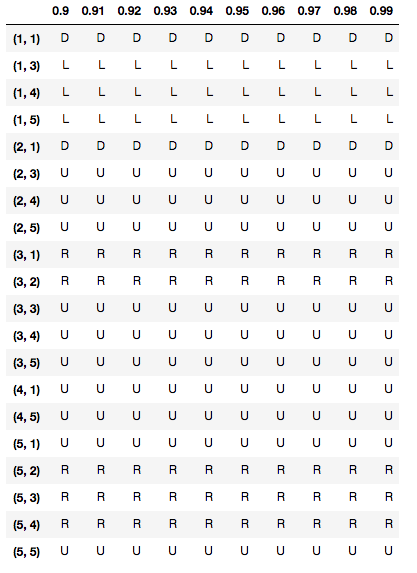


Figure 2. Initial state values against each iteration plotted for varying values of gamma.

Table 2. Policy obtained for each state (rows) at each value of gamma (columns).



Windy Gridworld

Part A

1. The state space is represented as s(x)(y), which refers to the x-th row and y-th column beginning with the top left. Thus, the set of feasible state spaces are:   
   {s11, s13, s14, s15, s21, s23, s24, s25, s31, s32, s33, s34, s35, s41, s45, s51, s52, s53, s54, s55}

In this problem, we begin at s51, and our goal is to reach s13.

The action space is {U, D, L, R}, which represents the action of going up, down, left, and right respectively.

*(nothing has changed here)*

1. The transition actions are still the same, except instead of a deterministic movement, the likelihood of achieving the intended movement is now . As such, the transition probability matrices are still as follows.

We assume that attempting to move to a blocked square will still fail with probability 1, so the transition probability matrix for such moves remains the same.



For the other movement attempts, the transition probability matrix changes to the following:

|  |  |  |  |
| --- | --- | --- | --- |
| Origin state (s) | Action (a) | Destination state (s') | P(s'|s,a) |
| s11 | D | s21 | p |
| D | s11 | 1-p |
| s14 | D | s24 | p |
| D | s14 | 1-p |
| L | s13 | p |
| L | s14 | 1-p |
| R | s15 | p |
| R | s14 | 1-p |
| s15 | D | s25 | p |
| D | s15 | 1-p |
| L | s14 | p |
| L | s15 | 1-p |
| s21 | U | s11 | p |
| U | s21 | 1-p |
| D | s31 | p |
| D | s21 | 1-p |
| s23 | U | s13 | p |
| U | s23 | 1-p |
| D | s33 | p |
| D | s23 | 1-p |
| R | s24 | p |
| R | s23 | 1-p |
| s24 | U | s14 | p |
| U | s24 | 1-p |
| D | s34 | p |
| D | s24 | 1-p |
| L | s23 | p |
| L | s24 | 1-p |
| R | s25 | p |
| R | s24 | 1-p |
| s25 | U | s15 | p |
| U | s25 | 1-p |
| D | s35 | p |
| D | s25 | 1-p |
| L | s24 | p |
| L | s25 | 1-p |
| s31 | U | s21 | p |
| U | s31 | 1-p |
| D | s41 | p |
| D | s31 | 1-p |
| R | s32 | p |
| R | s31 | 1-p |
| s32 | L | s31 | p |
| L | s32 | 1-p |
| R | s33 | p |
| R | s32 | 1-p |
| s33 | U | s23 | p |
| U | s33 | 1-p |
| L | s32 | p |
| L | s33 | 1-p |
| R | s34 | p |
| R | s33 | 1-p |
| s34 | U | s24 | p |
| U | s34 | 1-p |
| L | s33 | p |
| L | s34 | 1-p |
| R | s35 | p |
| R | s34 | 1-p |
| s35 | U | s25 | p |
| U | s35 | 1-p |
| D | s45 | p |
| D | s35 | 1-p |
| L | s34 | p |
| L | s35 | 1-p |
| s41 | U | s31 | p |
| U | s41 | 1-p |
| D | s51 | p |
| D | s41 | 1-p |
| s45 | U | s35 | p |
| U | s45 | 1-p |
| D | s55 | p |
| D | s45 | 1-p |
| s51 | U | s41 | p |
| U | s51 | 1-p |
| R | s52 | p |
| R | s51 | 1-p |
| s52 | L | s51 | p |
| L | s52 | 1-p |
| R | s53 | p |
| R | s52 | 1-p |
| s53 | L | s52 | p |
| L | s53 | 1-p |
| R | s54 | p |
| R | s53 | 1-p |
| s54 | L | s53 | p |
| L | s54 | 1-p |
| R | s55 | p |
| R | s54 | 1-p |
| s55 | U | s45 | p |
| U | s55 | 1-p |
| L | s54 | p |
| L | s55 | 1-p |

1. We define the Reward function as the following:

*(nothing changes here)*

1. <need a new V(s51) function with probabilities
2. See WindyGridworld.py

Part B – Value Iteration (ValueIteration.py)

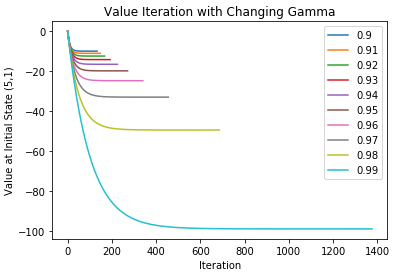


Figure 3. Initial state values for each iteration plotted for varying values of gamma.

Table 3. Policy obtained for each state (rows) at each value of gamma (columns).



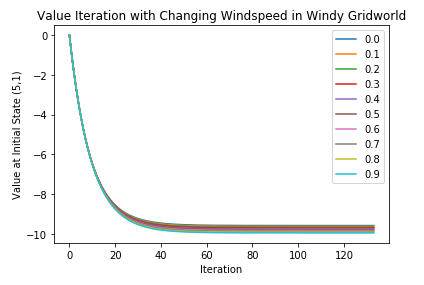


Figure 4. Initial state values for each iteration plotted for varying values of gamma.

Table 4. Policy obtained for each state (rows) at each value of gamma (columns).



Part C – Policy Iteration (PolicyIteration.py)

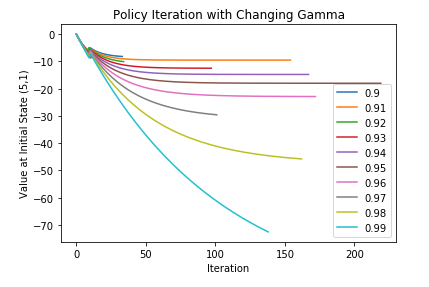


Figure 5. Using gamma of 0.90, initial state values for each iteration plotted for varying values of gamma

Table 5. Policy obtained for each state (rows) at each value of gamma (columns).



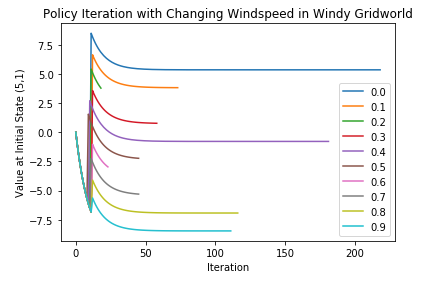


Figure 6. Using gamma of 0.90, initial state values for each iteration plotted for varying values of gamma

Table 6. Policy obtained for each state (rows) at each value of gamma (columns).

