AI Report

Pacman in a harder world

For our second assignment, I created an MDP solver to allow Pacman to win games. I did this by representing the game as several different states each with a reward reflecting either a positive aspect (e.g. food) or a negative aspect (e.g. ghosts).

I first wanted to create a grid that would represent the game-world and also allow me to fill it with expected utility values. I did this by getting the coordinates of the top right corner using the API, and then generating a list from (0,0) up until that point.

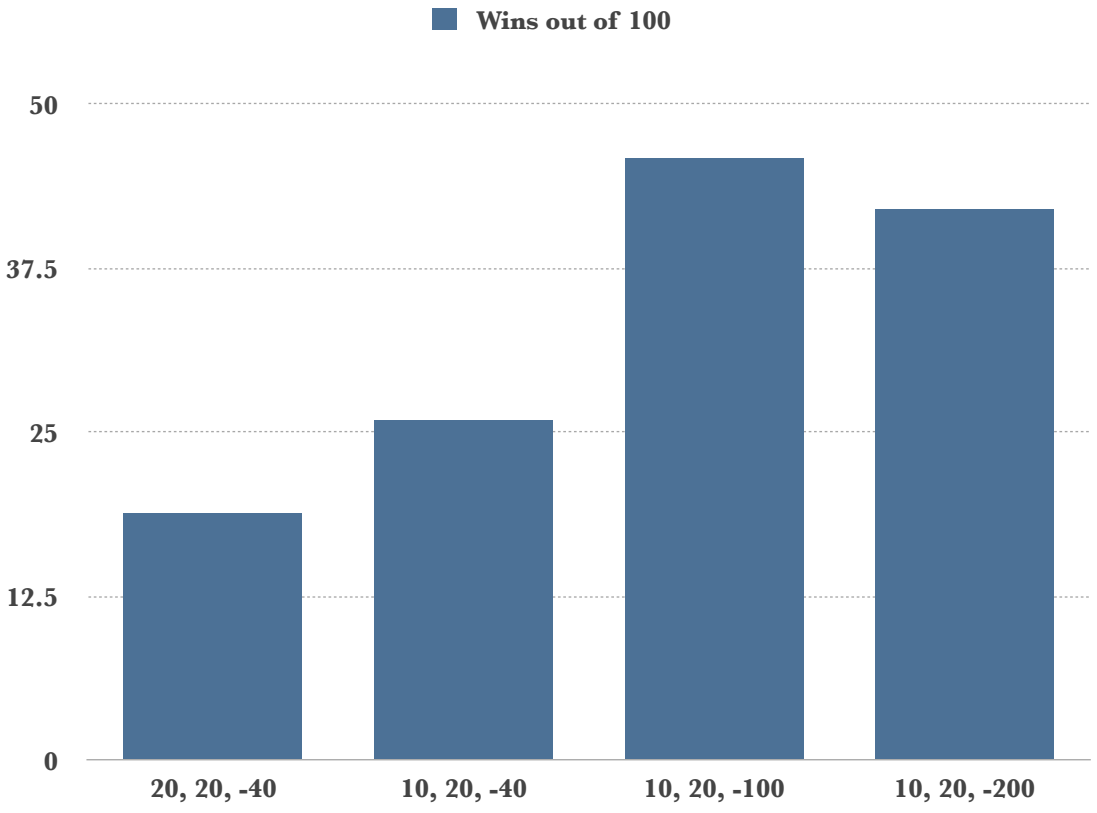
I then set my initial rewards for every location on the grid. Walls have a value of None as they are not accessible to Pacman and thus are not involved in the value iteration that will occur later. Food and capsules are given values of 10 and 20 respectively. The ghosts are given a value of -100, and tiles surrounding them are also given a negative reward (of -100, -30, and -20 depending on the distance from the ghost). This is necessary for success as the ghosts are not stationary and would otherwise run into an unsuspecting pacman.

Next I perform value iteration on the map until the values converge or until 100 iterations, whichever occurs first. While I found that the values usually converge within 60-70 iterations on the mediumClassic map, sometimes it could continue for a while so adding a limit helped to reduce the running time.

Finally I created a method to move pacman from his current location to the adjacent square with the highest value. This all culminates in pacman constantly moving around the map to areas of higher expected utility.

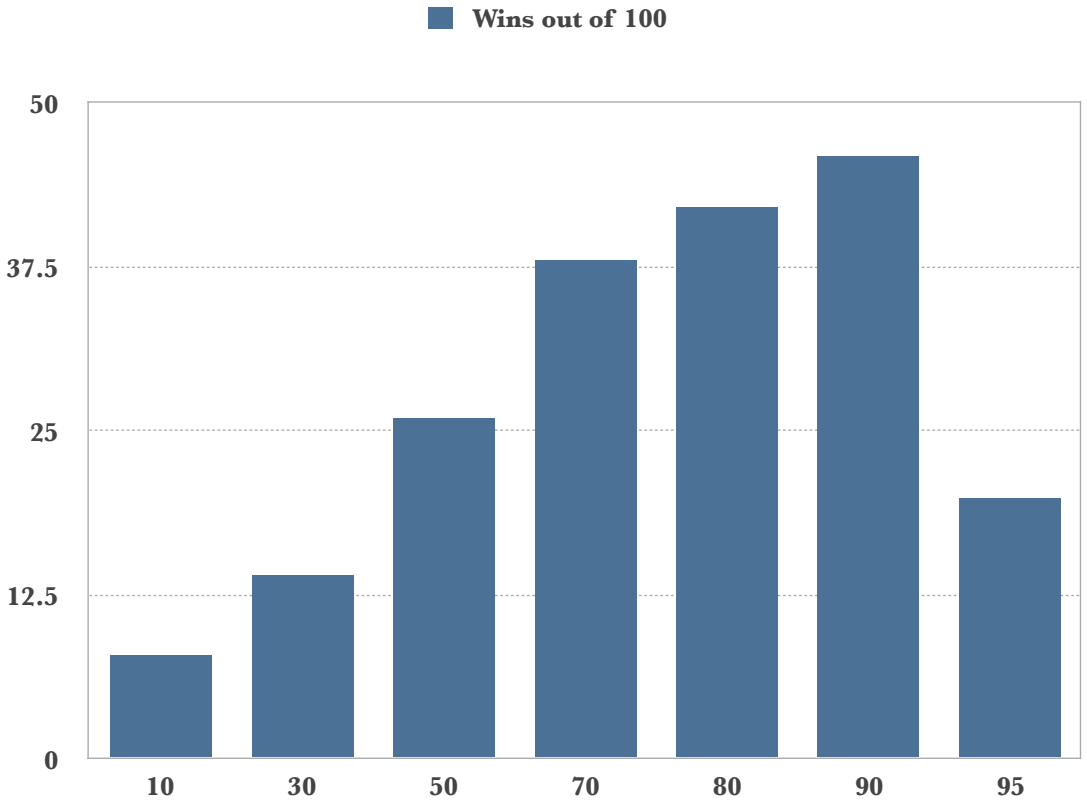
My results are satisfactory for the current assignment. On the mediumClassic map I have a win rate of 46% and on the smallGrid map a win rate of 63%. To achieve this I had to find optimal values for the rewards as well as for the gamma used in the Bellman equation.

The graph below details my results with different initial reward values on the mediumClassic map. The x axis shows the values for food, capsules, and ghosts respectively.



The data gave me some insight into the value of collecting capsules over food, as when capsules are given an initial preference the winrate increases. I assume this is due to the fact that pacman cannot die for a significant period of time after consuming a capsule. It also showed me that the ghosts need a low enough reward value to deter pacman from coming too close.

Changing the gamma value in the Bellman equation also produces different policies, with a lower value putting more emphasis on short term gains while a higher value does the opposite. I tested various values and the results for the mediumClassic map can be seen below.



From the results you can see that in this game it is important to incentivise long term gains. Interestingly when the gamma increases too much the win rate drops immensely.

Ideally I would like to further test all of these variables with different values to find the global optimum, but the results above give me confidence that I am at least in the same ballpark.