Dylan Barth
AIND Term 1
Isolation Project

Heuristic Analysis

Introduction

The goal of isolation is to force the opponent into a situation where they have no more available moves. Aside from the already implemented heuristics that were useful (improved score and center score), potentially useful heuristics that I came up with included:

- distance to the opponent,
- number of overlapping legal moves with the opponent,
- improved center score (scoring highly if our player is toward the center and the opponent toward the edges),
- improved score plus next move (i.e. looking into the future to explore the number of moves from each of the current legal moves).

Two additional ideas I had were:

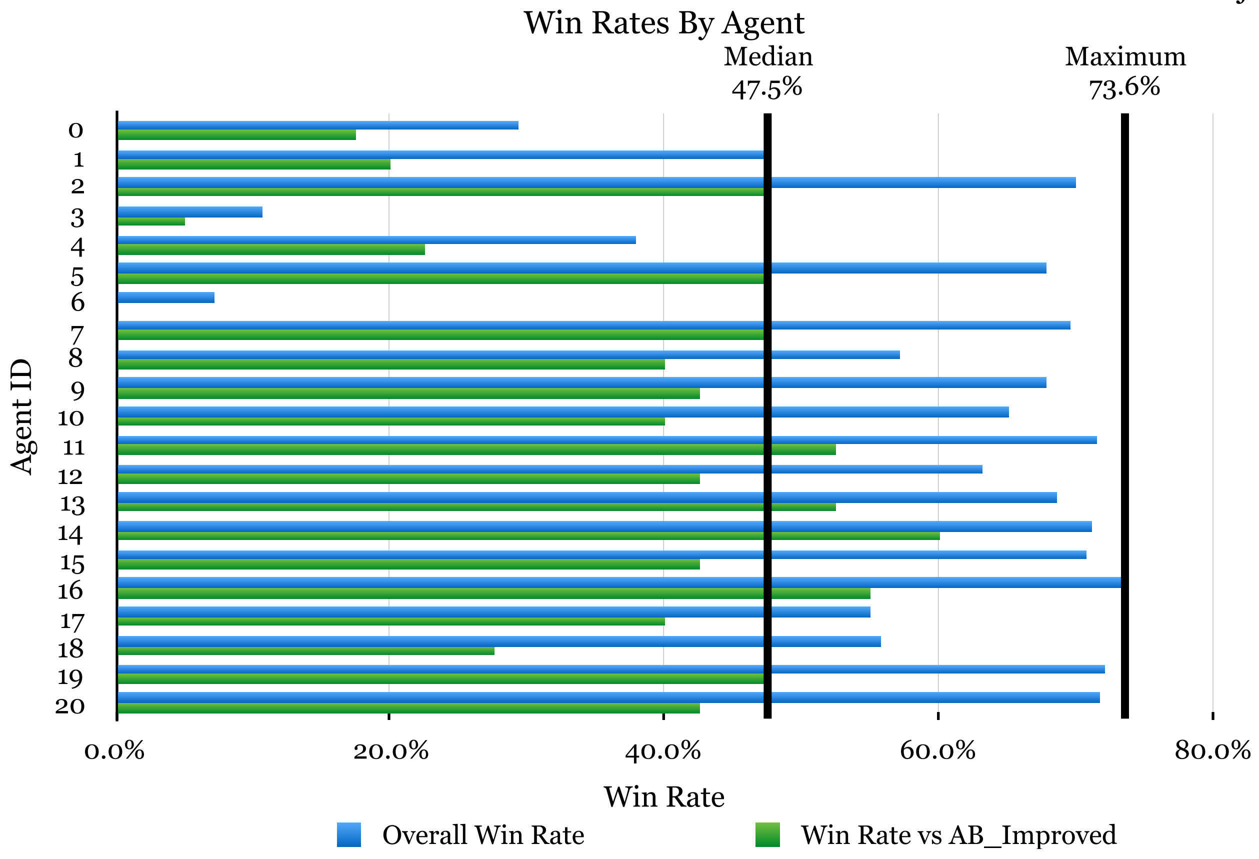
- Combining heuristics, for example: improved_score plus center_score.
- Changing heuristics based on how far along the game is eg have an endgame strategy. My thinking was that this might work well in combination with more computationally expensive heuristics such as improved score plus next move.

Heuristic Evaluation

In order to evaluate each heuristic, I created pre_tournament.py to include an agent that used each of my heuristics. Due to the number of heuristics, I've listed them all out below, with an ID and description. The results from pre_tournament.py are in the table below.

ID	Heuristic	Description
0	distance_to_opponent	Euclidean distance to opponent.
1	overlap_with_opponent_moves	Number of legal moves intersecting with opponent legal moves.
2	num_player_moves	Number of legal moves for the player.
3	num_opponent_moves	Number of legal moves for the opponent.
4	distance_to_center	Euclidean distance to center of board.
5	next_round_improved_score_for_player	Number of legal moves for the player in the next round of play (looking on move into the future).
6	next_round_improved_score_for_opponent	Number of legal moves for the player in the next round of play (looking on move into the future).
7	improved_score	Number of legal moves for player minus legal moves for opponent.
8	improved_center_score	Euclidean distance to center of board for player minus euclidean distance to center of board for opponent.

ID	Heuristic	Description
9	improved_score_plus_center_mod2	improved_score plus center score divided in two
10	improved_score_plus_center	improved_score plus center score
11	improved_score_minus_center	improved_score minus center score
12	improved_score_plus_distance_to_opponent	improved_score plus distance_to_opponent
13	improved_score_minus_distance_to_opponent	improved_score minus distance_to_opponent
14	improved_score_plus_overlap_with_opponent	improved_score plus overlap_with_opponent
15	improved_score_minus_overlap_with_opponent	improved_score minus overlap_with_opponent
16	improved_score_plus_improved_center	improved_score plus improved_center_score
17	improved_score_minus_improved_center	improved_score minus improved_center_score
18	center_then_improved_score	Start game with center score strategy, and then switch to improved score in the end game.
19	improved_with_endgame_strategy	Start game with improved score strategy and then in the endgame switch to improved_score plus next_round_improved_score_for_player.
20	improved_with_improved_endgame_strategy	Start game with improved score strategy and then in the endgame switch to improved_score plus next_round_improved_score_for_player minus next_round_improved_score_for_opponent



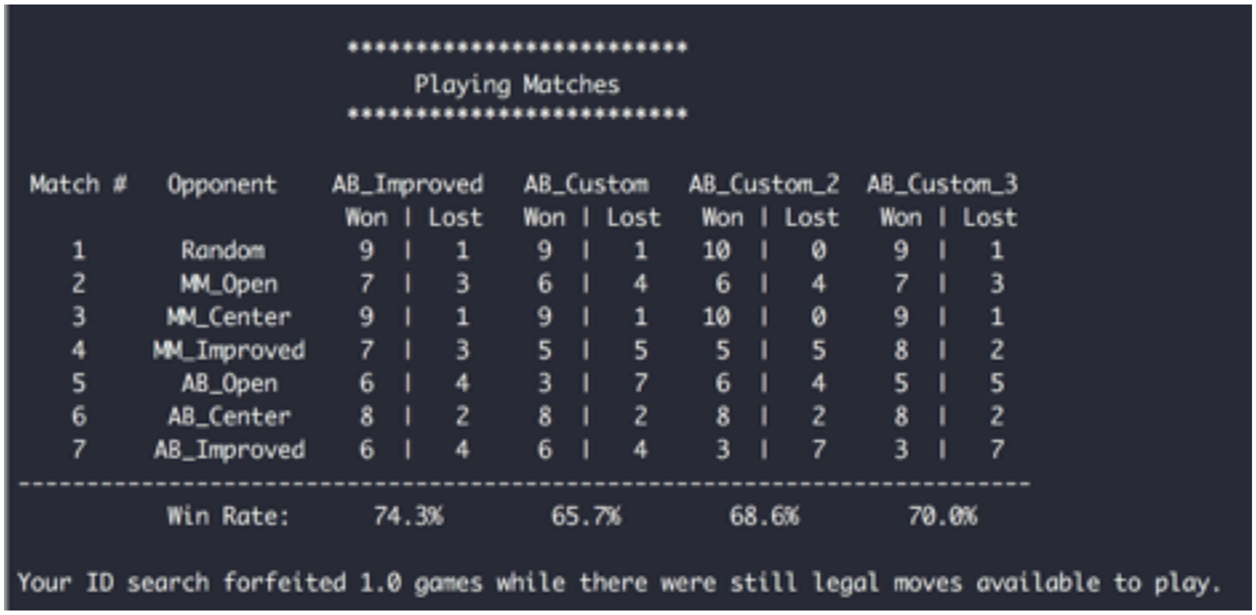
The results from the pre_tournament.py script are charted above. Each agent played 20 matches per opponent with 150ms allowed per move. The chart shows the overall win rate in blue, with the win rate against AB-Improved (the strongest opponent) in green. Only a few of the heuristics performed consistently better than the AB_Improved agent:

- 11 (improved_score_minus_center)
- 13 (improved_score_minus_distance_to_opponent)
- 14 (improved_score_plus_overlap_with_opponent)
- 16 (improved_score_plus_improved_center)

I was disappointed that my endgame strategies did not do better because my intuition was that they would be stronger heuristics. Given more time, I would explore these further, perhaps tweaking how the heuristics evaluates whether or not we are in the endgame stage (at present this is based on percentage of blank spaces left), and seeing what is computationally feasible given the amount of board space left.

Tournament Results

From the results above, agents 14, 16, and 11 were chosen to compete in the tournament. 13 was eliminated, because although it performed as well as 11 against AB_Improved, it had a lower win-rate overall.



```
                   *************************
                        Playing Matches
                   *************************

 Match #   Opponent    AB_Improved   AB_Custom    AB_Custom_2  AB_Custom_3
                        Won | Lost   Won | Lost   Won | Lost   Won | Lost
    1       Random       9  |  1      9  |  1     10  |  0      9  |  1
    2       MM_Open      7  |  3      6  |  4      6  |  4      7  |  3
    3      MM_Center     9  |  1      9  |  1     10  |  0      9  |  1
    4     MM_Improved    7  |  3      5  |  5      5  |  5      8  |  2
    5       AB_Open      6  |  4      3  |  7      6  |  4      5  |  5
    6      AB_Center     8  |  2      8  |  2      8  |  2      8  |  2
    7     AB_Improved    6  |  4      6  |  4      3  |  7      3  |  7
--------------------------------------------------------------------------
           Win Rate:      74.3%        65.7%        68.6%        70.0%

Your ID search forfeited 1.0 games while there were still legal moves available to play.
```

Recommendation: Agent 14 (improved_score_plus_overlap_with_opponent)
This agent performed the best and most consistently in the round-robin, although it failed to differentiate itself in the tournament.py. However, the tournament played fewer matches (5 vs 20 in the pre-tournament rounds) so I decided to treat the pre-tournament results as closer to the true performance metrics. Despite poor performance in the tournament, improved_score_plus_overlap_with_opponent heuristic would be my recommendation for the custom metric because in the pre-tournament, it's win rate was high against all opponents, it performed the best against the most challenging agent AB_Improved, and in the tournament, it was the only custom agent with a positive win rate against AB_Improved.