Heuristic Analysis

1. Heuristic Function

I implemented three functions in my game playing against the improved heuristic function.

Custom Score

This is naive variation of improved score. It calculates the difference between the number of own player moves and opponent player moves. It wights the number of opponent player moves with the factor 2.

Custom Score2

This function is the improved center_score in sample_players. It calculates the difference between own player distance and opponent distance from center.

Custom Score3:

This function calculate the distance between own player and opponent player.

2. Agent Information

- Random: Agent that randomly choose a move each turn
- MM_Null: Agent using fixed-depth mini-max search and the null_score function
- MM_Open: Agent using fixed-depth mini-max search and the open_move_score function
- MM_Improved: Agent using fixed-depth mini-max search and the improved_score function
- AB_Null: Agent using fixed-depth alpha-beta search and the null_score function
- AB_Open: Agent using fixed-depth alpha-beta search and the open_move_score function
- AB_Improved: Agent using fixed-depth alpha-beta search and the improved_score function

3. Results

We could evaluate our heuristic functions using tournament.py as below.

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

Match #	Opponent	AB_Improved		AB_Custom		AB_Custom_2		AB_Custom_3	
		Won	Lost	Won	Lost	Won	Lost	Won	Lost
1	Random	10	0	6	4	9	1	10	0
2	MM_Open	8	2	9	1	7	3	7	3
3	MM_Center	10	0	10	0	10	0	9	1
4	MM_Improved	8	2	9	1	9	1	8	2
5	AB_Open	5	5	5	5	4	6	4	6
6	AB_Center	5	5	7	3	6	4	5	5
7	AB_Improved	5	5	8	2	4	6	5	5

| | Win Rate: | 72.9% | | 77.1% | | 70.0% | | 68.6% | |

I focus on the results of match # 7 because this agent is the most sophisticated opponent. From this output, we consider the results of match #7, which was against the most sophisticated opponent. Custom_score heuristic performed best, followed by custom_2 and then followed by custom_3. In addition, custom_score heuristic is only one better than improved heuristic.

4. Heuristic function recommendation in my three functions

I recommend the custom_score heuristic based on playing results following reasons

1. It is relatively simple to implement. So any other persons try to use this heuristic and we could expand the n-player easily if we need.
2. If we would like to improve this function, we could tune using grid search for this function: A*own_moves - B*opp_moves
3. This function has the highest Win Rate which is 77.1% higher than improved heuristic. Also, this heuristic is the highest Win Rate from #2 to #7 opponent.