The table above shows the relative performance of my agent in a round-robin tournament against several pre-defined agents.

The tournament opponents are listed below:

- Random: An agent that randomly chooses a move each turn.
- MM_Open: MinimaxPlayer agent using the open_move_score heuristic with search depth 3
- MM_Center: MinimaxPlayer agent using the center_score heuristic with search depth 3
- MM_Improved: MinimaxPlayer agent using the improved_score heuristic with search depth 3
- AB_Open: AlphaBetaPlayer using iterative deepening alpha-beta search and the open_move_score heuristic
- AB_Center: AlphaBetaPlayer using iterative deepening alpha-beta search and the center_score heuristic
- AB_Improved: AlphaBetaPlayer using iterative deepening alpha-beta search and the improved_score heuristic

Agent performance comparison after playing 25 matches:

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

Match #   Opponent    AB_Improved   AB_Custom   AB_Custom_2  AB_Custom_3
                       Won | Lost   Won | Lost   Won | Lost   Won | Lost
   1       Random      47  |   3    48  |   2    48  |   2    47  |   3
   2       MM_Open     40  |  10    38  |  12    39  |  11    30  |  20
   3      MM_Center    43  |   7    44  |   6    43  |   7    44  |   6
   4     MM_Improved   32  |  18    36  |  14    38  |  12    34  |  16
   5       AB_Open     26  |  24    27  |  23    26  |  24    26  |  24
   6      AB_Center    29  |  21    32  |  18    29  |  21    24  |  26
   7     AB_Improved   24  |  26    25  |  25    22  |  28    24  |  26
--------------------------------------------------------------------------
          Win Rate:      68.9%        71.4%        70.0%        65.4%
```

custom_score

This custom function simply returned the difference between # of moves available to my player vs opponent. This strategy:

- Did well against Random, MM_Open, MM_Center, AB_Open, AB_Center, winning by double digit margins
- Won 12x times vs MM_Improved
- Tied with AB_Improved, likely due to both agents being more sophisticated relative to other strategies by using iterative deepening alpha-beta search
- This strategy is computationally more expensive compared to others since we must go through all future moves for both players and take the difference.

This was the best strategy.

custom_score_2

This custom function simply increased the aggression as the end of the game was near. The idea is that I would allow less spaces for opponent to move (putting more weight to prevent winning).

This strategy:

- Did well against Random, MM_Center, AB_Open, AB_Center, winning by double digit margins
- Performed 25% worse vs MM_Open compared to other heuristic functions
- Won 16x times vs MM_Improved
- Lost 6x vs AB_Improved
- Switching strategies (mild to aggressive) might be a good idea, but it's not clear at which point it's most optimal to switch

This was the second best strategy.

custom_score_3

This custom function simply used the distance from current position differenced by number of opponent moves. The idea is that the further away you are from center, the more exploration is required leading to a lower probability of winning.

- Did well against Random, MM_Open, MM_Center, AB_Open, AB_Center, winning by double digit margins
- Won 18x times vs MM_Improved
- This heuristic is simple to compute
- Lost 2x vs AB_Improved

This strategy didn't seem to work (alt least when compared to others) since the center of the board doesn't necessarily translate to more options for my player to move.

Recommended Evaluation function is the **custom_score** above, which simply returned the difference between # of moves available to my player vs opponent.

Due to the following reasons:

1. win rate: this strategy tied with the most intelligent opponent agent
2. complexity: is relatively simple strategy, giving us a window into future moves available to players
3. this heuristic predicts the outcome of the game with 50% accuracy.
4. the other two custom score functions require switching strategies or penalization so further research needs to be done before i determine we see results better than AB_Imrproved