Isolation Heuristic Review

In the project, we need to implement the min/max, alpha-beta pruning, and iterative deepening algorithms. I have implemented three different heuristics as required.

Heuristic Descriptions

In **Custom_score**

- Calculated percentage of legal_moves a player has vs. the number of remaining blank spaces.
- Opponent value is doubled to provide more aggressive play.

Custom_score_2

- Difference of legal moves multiplied by probability value whose sum equals to one.
- The opponent weight is .95 and own player has .05.
- This provides more conservative game play.
- I have used .95/.05 combo based on case study shared and it seems to have more consistent results.

Custom_score_3

- Calculated difference of legal moves.
- The opponents moves are doubled.
- Done this to make opponent player always had a commanding lead and my player can be more aggressive.

Results

All three implemented heuristics has win percentage more than 70% of the time. Adjusting time has negatively impact alpha-beta pruning techniques. I might need to calibrate weights or compare against neighboring moves differently.

Recommendation

I would recommend custom_score_2 because it had the consistent win rate. This heuristic also weighs the opponent's number of moves more heavily even if not equal. This possibly provides deeper searches prior to pruning potential leafs. I got idea of using .05 from the "Game Tree Searching by Min /Max Approximation" article

Game Results

```
(aind) Sunil-Sharma530-MPR00004:AIND-Isolation sunilsharma$ python tournament.py

This script evaluates the performance of the custom_score evaluation
function against a baseline agent using alpha-beta search and iterative
deepening (ID) called `AB_Improved`. The three `AB_Custom` agents use
ID and alpha-beta search with the custom_score functions defined in
game_agent.py.

                    **************************
                         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    10  |   0    10  |   0     9  |   1
    2       MM_Open      8  |   2    10  |   0     7  |   3     9  |   1
    3      MM_Center     9  |   1    10  |   0    10  |   0    10  |   0
    4     MM_Improved   10  |   0    10  |   0     8  |   2     9  |   1
    5       AB_Open      5  |   5     3  |   7     7  |   3     6  |   4
    6      AB_Center     4  |   6     6  |   4     4  |   6     7  |   3
    7     AB_Improved    6  |   4     5  |   5     7  |   3     6  |   4
--------------------------------------------------------------------------
            Win Rate:      74.3%        77.1%        75.7%        80.0%
```

Figure 1, Win rate of the three heuristic functions when running the default tournament script.

Figure 2, Win rate of the three heuristic functions when running the tournament script for 20 games and 150 milliseconds timeout.

Figure 3, Win rate of the three heuristic functions when running the tournament script for 20 games and 250 milliseconds timeout.