Heuristic Analysis

In game_agent.py, I implemented three custom_score functions:

1. custom_score(): number of my legal moves minus number of opponent legal moves minus distance from my location to center board.
2. custom_score_2(): the square of distance from my location to opponent location.
3. custom_score_3(): number of my legal moves minus 1.5 times of number of opponent legal moves.

The result shows in the picture:

```
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     8  |   2    10  |   0
2       MM_Open      7  |   3     8  |   2     8  |   2     8  |   2
3      MM_Center     8  |   2     9  |   1     7  |   3    10  |   0
4     MM_Improved    6  |   4     7  |   3     7  |   3     7  |   3
5       AB_Open      5  |   5     7  |   3     7  |   3     6  |   4
6      AB_Center     6  |   4     7  |   3     6  |   4     5  |   5
7     AB_Improved    4  |   6     6  |   4     3  |   7     4  |   6
--------------------------------------------------------------------
Win Rate:      65.7%        77.1%        65.7%        71.4%
```

From the result we can see all of my custom score functions performing not worse than AB_Improved agent. The custom_score_2() has the same win rate with AB_Improved agent, which is 65.7%. The custom_score_3() has 71.4% win rate, which is better than AB_Improved agent. And custom_score() performs best, which is 77.1%.

I recommend custom_score(). First of all, it passes the test and has the best win rate. Second, it wins more games than all of seven tournament agents which indicates that it performs stable. And finally this evaluation function considers more features of a game state, which performs better reasonably.