# Heuristic Analysis

For the heuristic analysis we will evaluate 3 heuristics in the custom\_scores and comment about the ID\_Improved.

## Three heuristics

The heuristics are:

### Difference between number of player’s movements minus number of opponent’s movements

Here it is the code:

return (float(len(game.get\_legal\_moves(player))) - len(game.get\_legal\_moves(game.get\_opponent(player))))

This is the simplest heuristic we analysed. The results were 65.00% student against 57.86% ID\_Improved. In the end evaluating this heuristic against the ID\_Improved, they are really calculating the same thing.

The difference in performance is due to the Student code to be faster being able to evaluate deeper positions.

### Average difference between number of player’s movements minus number of opponent’s movements for one move ahead

Code:

score = 0

for move in game.get\_legal\_moves(player):

game\_copy = game.forecast\_move(move)

score += float(len(game\_copy.get\_legal\_moves(player))) - len(game\_copy.get\_legal\_moves(game.get\_opponent(player)))

division = (len(game.get\_legal\_moves(player)) if len(game.get\_legal\_moves(player)) != 0 else 0.001)

return score / division

The performance here was 73.57% Student and 67.14% ID\_Improved. When we go one movement ahead we are seeing more into the future than the ID\_Improved, so the performance is better.

### Average difference between number of player’s movements minus number of opponent’s movements for two moves ahead

Code:

score = 0

division = 0

for move in game.get\_legal\_moves(player):

game\_copy = game.forecast\_move(move)

for second\_move in game\_copy.get\_legal\_moves(game.get\_opponent(player)):

game\_second\_copy = game\_copy.forecast\_move(second\_move)

score += (

float(len(game\_second\_copy.get\_legal\_moves(player))) -

len(game\_second\_copy.get\_legal\_moves(game.get\_opponent(player))))

division += 1

division = (division if division != 0 else 0.001)

return score / division

The performance here was 48.57% Student and 65.71% ID\_Improved. When we increased the code the time to run it became too high, in the end we couldn’t perform better than ID\_Improved. We lost too much time performing the custom\_score and were not seeing deeper enough in the end.

## Conclusion

In the end the Average difference for one move ahead was the heuristic choosen. It has a balance between looking into the future and having a good performance for the competition.