Artificial Intelligence/Intelligent Agents 2022/2023

Homework2 Adaptive Search

Step 1:Implement possible variations on $\alpha\beta MinMax_L$ search that would speed up computation:

a) $\alpha\beta MinMax_{best\ l}$:explore only subtrees of most promising nodes (i.e. best k nodes according the H_l evaluation, l may go from 0 to L, obviously the larger it is the better it performs but the less speed-up is obtained). Experiment on some small values of l to set it up to a value that suits your computing resources…

b) $\alpha\beta MinMax_{consistent}$:explore only subtrees of nodes such that their evaluations $[H0,H1,\ldots,H_l]$ have smallest variance

i.e. *b1*) k nodes whose evaluations at increasing levels has least variance, or

b2) nodes whose evaluations at increasing levels has variance smaller than a fixed threshold

c) $\alpha\beta MinMax_{improving}$:explore only subtrees of nodes such that their evaluations $[H0,H1,\ldots,H_l]$ is increasing(decreasing)

ci) strictly increasing may not be a good choice, you may want to experiment on different implementations of the increasing/decreasing condition…

Step 2: -Use your version of $MinMax_{speed\ up}$ that you developed in step 1 to create a training set for learning the H_L evaluation of states, given $h_0,\ldots h_k$ static evaluation/observation of the state.

Populate the TS by playing games…

-Train you regressor (you can import library functions…) to obtain a $_{predictiveL}MinMax1$(that is $_{speed\ up}MinMax$ at level 1 that uses the prediction as static evaluation)

-Evaluate performance of your $_{predictiveL}MinMax1$ against your $_{speed\ up}MinMax_L$, and against $_{speed\ up}MinMax_{L/2}$

-Evaluate performance of your $_{predictiveL}MinMaxL$ ((MinMax at level L that uses the prediction as static evaluation) against your $_{speed\ up}MinMax_L$, *and against* $_{speed\ up}MinMax_{L/2}$