Dice Game Report

VALUE ITERATION ALGORITHM AND IT'S IMPLEMENTATION

My agent uses a value iteration algorithm to determine an optimal policy. For each pass of updates, the value iteration algorithm typically has time complexity of $O(S^2A)$, where S is number of states and A is number of actions, since it must update utilities for each state (first S) by calculating, for each action (A), the value of each outcome (second S). For the dice game, S is (number of sides)^(number of dice) whereas A is equal to the number of dice. My implementation improves speed but not worst-case time complexity by vectorising updating calculations of the Bellman equation and caching the results of the get_next_states methods. The number of passes required to reach an error ε representing the maximum of the differences between the algorithm's utilities and the optimal utilities is $N = \log(2R_{max}/\varepsilon(1 - \gamma))/\log(1/\gamma)$, where R_{max} is the maximum reward and γ is the discount factor (Russel and Norvig 2002, p. 655). Given N increases rapidly as γ approaches 1, I introduced a second stopping condition to my algorithm based on whether the bellman update on any given iteration is sufficiently small.

HYPERPARAMETER OPTIMISATION AND PERFORMANCE

I optimised my three hyperparameters – the discount factor and the two stopping conditions – with an iterative grid search, which initiated multiple games and used the agent to obtain an average score for the combination of parameters used. The iterative grid search started with a wide range, and then reduced the range, centred around the optimal combination from the previous iteration. I incorporated the tradeoff between optimal score and calculation time by setting limits to how small the two stopping conditions could be, but let gamma range from 0 to 1, relying on the second stopping condition to act as a break if a gamma close to 1 led to onerous calculation time. The optimisation of all three parameters was conducted for the dice game with default initialisation conditions, but optimisation over wide ranging conditions would have taken an impractically long time to calculate. Cursory exploration of optimal conditions for different game initialisations suggested that the optimal gamma could vary more, and made more of an impact, than the two stopping conditions, especially for extreme penalty values or potential rewards from dice rolls – extreme penalty values with low potential rewards tended to favour lower gammas. I therefore updated the iterative grid search to focus only on optimising gamma and ran it over 1500 randomly generated dice game initialisations, recording the optimal gamma, the penalty, and the sum-product of the dice sides with their biases, multiplied by number of dice (proxying potential reward). I ran logit, linear, and random forest regressions with these two latter features to predict the optimal gamma, with the best of these (random forest) achieving a mean square error of 0.05 on the validation set. I intended to pass the regressor into MyAgent to rapidly adapt its gamma value based on the initialisation it was given. However, a closer inspection of optimal gamma values revealed they were heavily skewed to values close to one, and an agent with a gamma close to one often beat one that relied on the trained regressor, leading me to set gamma close to 1.

FURTHER WORK

My investigations into the optimal gamma given different initialisations invites theoretical and empirical work. Theoretical work could focus on why optimal gamma values cluster around 1 and the link between different reward structures and the optimal gamma, whereas empirical work could be adapted so that the feature proxying the potential reward of a dice roll incorporates the impact of dice flips when one or more dice are the same, potentially leading to better regressor performance. The impact on time complexity of parallelising the updating of values for each state would also be a fruitful area of research.

References

Russell, S., and Norvig, P., 2002. *Artificial Intelligence: A Modern Approach.*Third Edition.