Isaiah King

Homework 3: On-Policy Monte-Carlo Learning

We observe that given enough time, an on-policy MC agent can learn close to optimal policies for this environment. However, the amount of time it takes to find useful policies is far too great. For models using more than 1000 episodes, it takes close to 1 minute to complete, which increases geometrically at least with episode length. We theorize one possible reason for this: the memory needed to hold accumulated rewards is O(N\*S) where N is number of episodes, and S is the number of states. S is constant, so the special complexity simplifies to O(N), which appears linear. However, after a certain point, the amount of data becomes so large it no longer fits into memory. The amount of data becomes so large that basically every episode causes episode\_length cache misses just to access the list of rewards. Compound that with the nontrivial amount of time it takes to calculate the mean of an increasingly long array of integers, and the computational and especially memory complexity of this method quickly explodes.

However, as stated previously, the results found by this method, at least with lower epsilon values, approach the optimal policies when enough episodes are run. The efficiency of the path found seems to correlate strongly with the number of episodes, and have a lower, but not non-zero correlation with the length of episodes.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Length of best path with epsilon 0.02 | | Episode Length | | | |
| 10 | 100 | 1000 | 10000 |
| Number of Episodes | 10 | INF | INF | INF | INF |
| 100 | INF | INF | INF | INF |
| 1000 | INF | INF | INF | INF |
| 10000 | INF | 58 | 42 | INF |
| 100000 | INF | 55 | 46 | 55 |
|  |  |  |  |  |  |
| Length of best path with epsilon 0.25 | | Episode Length | | | |
| 10 | 100 | 1000 | 10000 |
| Number of Episodes | 10 | INF | INF | INF | INF |
| 100 | INF | INF | 695 | INF |
| 1000 | INF | INF | 216 | 271 |
| 10000 | INF | 66 | 95 | INF |
|  |  |  |  |  |  |
| Length of best path with epsilon 0.50 | | Episode Length | | | |
| 10 | 100 | 1000 | 10000 |
| Number of Episodes | 10 | 770 | INF | INF | 860 |
| 100 | INF | INF | 596 | 421 |
| 1000 | 585 | 853 | 224 | 320 |
| 10000 | INF | 106 | 248 | 378 |

Upon a more granular trial of number of episodes and lengths with the most efficient epsilon, 0.02, the data suggests that after a certain point, more and longer episodes do not improve the model. The below table seems to suggest that a local optimal configuration of the parameters exists somewhere around 70,000 episodes of length 400 when using an epsilon value of 0.02

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Length of best path with epsilon 0.02 | | Episode Length | | | |
| 100 | 400 | 700 | 1000 |
| Number of Episodes | 10000 | 40 | 44 | 154 | 593 |
| 40000 | 50 | 41 | 42 | 56 |
| 70000 | 43 | 40 | 48 | 48 |
| 100000 | 46 | 48 | 54 | 56 |

We presume that a TDL method is better suited for this problem. Even with fewer than 400 states, the memory complexity required to generate useful models was too great and resulted in runtimes greater than 30 minutes on a single CPU. If TDL is used, the need to maintain a full list of every time a state has been visited is abated, and we presume runtime will improve as memory use decreases.