# COL 870 – Reinforcement Learning

## Assignment 1 – Achieve 31

### State Space

My state representation is of the form:

Where represents the set of possible states, represents number of total usable special cards, represents hard sum of player and represents set of possible dealer cards at start. represents the state in which player busts in the game as described in the problem statement. Now:

1. As the player can have at max 3 special cards each of black cards of 1, 2, 3:
2. We define hard sum as the sum of all cards of the player without considering any card as special. Hence, we add all black cards face values and subtract red cards face values to obtain the hard sum of the player. Now, for a no special cards the hard sum can vary from 0 to 31, otherwise the player would go bust if it exceeds 31 or goes below 0. For 1 special card, the hard sum can range from -10 to 31 as for {-10, … -1} the special card can be used as higher value and for {20, ... 31} can not be used as higher value. Similarly, for 2 and 3 cards the hard sum can vary from -20 to 31 and -30 to 31. Hence, hard sum has overall range:

Note that the states where hard sum is less than the value possible as per the number of special cards are considered as bust state

1. As the dealer can have cards 1 to 10 the dealer hand:

Among these states, the non-actionable states are those in which soft sum i.e. the maximum sum that can be formed by the cards (including the special ones) such that the sum is less than or equal to 31 is 31, or the player is in bust state. This is because when the soft sum is 31 it is obvious that the only reasonable option would be to stick and not hit. If the player sticks it can win or lead to a draw. If it sticks it will always go to bust state. The other case in which every possible combination of cards leads to sum of > 31 or < 0 is also non-actionable as player has lost (or draw match as per special case in the statement) in this state.

### Policy Evaluation

For the given simple policy of playing ‘hit’ until the player can reach a sum of 25 or more then playing ‘stick’, we evaluate using Monte Carlo and TD.

Policy evaluation results in terms of value function for:

* Update strategy: Monte Carlo
* Variant: First visit update
* Runs: 1
* Episodes per run: 1 million

|  |  |
| --- | --- |
|  |  |
|  |  |

Policy evaluation results in terms of value function for:

* Update strategy: Monte Carlo
* Variant: Every visit update
* Runs: 1
* Episodes per run: 1 million

|  |  |
| --- | --- |
|  |  |
|  |  |

We see that both graphs are similar in their trends. General trends and reasoning:

1. There is a drop at the value function at 25 soft sum for all special card range 0 to 3. This is because at this point for less than 25 cards, there is possibility of following policy and playing ‘hit’ to lead to drawing a black card with value >= 7 leading to bust. Moreover, the value function increases drastically after 25 because the policy now changes and the player ‘sticks’ leading to mostly winning the game as the soft sum is quite high (25 or more).
2. There are similar drop and peak at 15, 20 when there is 1 special card because based on whether the card is used with a higher value or not, the hard sum can be 15 or 25 to lead to the soft sum as 25. Similarly based on special card is used with higher value or not, the peak lies on 31 and 21.
3. As special cards increase, this patter or drop and sudden rise in value function repeats as many number of times as there are special cards with an offset of 10 based on how many special cards are being used with higher value.
4. There is also an increase in the value function as the dealer card value reduces as the chances of winning are higher if the existing dealer card is of low value. Though this increase is gradual still it is noticeable in all graphs.

Small differences in MC first visit and every visit:

1. The graph of the every-visit case has slightly higher variance than the first visit one probably because of higher number of samples with different number of occurrences of each sample.

Policy evaluation results in terms of value function for:

* Update strategy: Temporal Difference
* Initialized value for each state: 0
* k: 1
* Runs: 10
* Episodes per run: 100,000

|  |  |
| --- | --- |
|  |  |
|  |  |

Policy evaluation results in terms of value function for:

* Update strategy: Temporal Difference
* Initialized value for each state: 0
* k: 3
* Runs: 10
* Episodes per run: 100,000

|  |  |
| --- | --- |
|  |  |
|  |  |

Policy evaluation results in terms of value function for:

* Update strategy: Temporal Difference
* Initialized value for each state: 0
* k: 5
* Runs: 10
* Episodes per run: 100,000

|  |  |
| --- | --- |
|  |  |
|  |  |

Policy evaluation results in terms of value function for:

* Update strategy: Temporal Difference
* Initialized value for each state: 0
* k: 10
* Runs: 10
* Episodes per run: 100,000

|  |  |
| --- | --- |
|  |  |
|  |  |

Policy evaluation results in terms of value function for:

* Update strategy: Temporal Difference
* Initialized value for each state: 0
* k: 100
* Runs: 10
* Episodes per run: 100,000

|  |  |
| --- | --- |
|  |  |
|  |  |

Policy evaluation results in terms of value function for:

* Update strategy: Temporal Difference
* Initialized value for each state: 0
* k: 1000
* Runs: 10
* Episodes per run: 100,000

|  |  |
| --- | --- |
|  |  |
|  |  |

Most of these graphs are similar to the MC case with some differences:

1. The value functions are biased by the initialization. The graphs shown before have initialized value as 0 for all states.

* Decay is per episode
* 2b runs is episodes not convergence
* Discount kept 0.7