Machine Learning: Reinforced Learning

Robert Horton*
University of Colorado Springs Colorado
rhorton2@uccs.edu

Gaberial Yeager†
University of Colorado Springs Colorado
gyeager@uccs.edu

September 23, 2021

Abstract

ABSTRACT HERE!!!!!

1. Introduction

If you walk into any casino, you are likely to see one of the most popular card games in the world. Blackjack was invented in France about 300 years ago where it was known as "Vingt et Un" which means "twenty-one". The objective is to get the sum of the value of your cards as close to 21 without going over 21. Going over 21 is know as a "break" which is a loss that gives the dealer a win. Its popularity is largely to the fact that is simple to learn. When playing blackjack there is a dealer and at least 1 player. Every card has a point value. Cards 2 through 10 are worth the corresponding number printed on them regardless of the color or suit. For example, 7 of hearts is simply worth 7 points. Face cards (jack, queen, king) are worth 10 points, and ace cards can be worth either 1 or 11 points, depending on whether or not the player "breaks" and goes over 21. A player is dealt two cards and may "hit" and receive another card or "stay" and refuse another card. There are a few cases that exist in casino blackjack that may be considered. In standard blackjack a player can split their hand up to 3 times, meaning the player can play up to 4 hands. An ace and a card valued at 10 points beats all other cards, even if the sum of the other cards is 21. The player places a wager at the beginning of the hand and may change their bet after two cards are dealt. The dealer has a advantage over the players because of the hole card that players must guess to play against. This advantage of the dealer becomes 0.5% if the player follows the basic blackjack strategy (Wong 1994). We will train an agent that will play blackjack according to policies it has learned during training. The agent will be trained with several different reinforced learning algorithms including Q-learning, Sarsa, and Temporal Difference will be compared to the optimal strategy and random plays (no strategy).

2. Background

Reinforcement learning, although considered a part of machine learning, is very different from the other types of machine learning. Two main categories that the different types of machine learning fall under are supervised learning[1] and unsupervised learning [2] [Sutton, R. S., & Barto, A. G. , 2018]. Although supervised learning can be used to speed along the process of machine learning this is not always an option when deploying a machine learning process into an unexplored environment. These types of implementations

*"Denim"
†"Gabe"

[1]Explanation of supervised learning
[2]Explanation of unsupervised learning

call upon reinforcement learning to perform an unsupervised learning process. Now when performing these unsupervised approaches of machine learning there are many different methods that have been used and studied over the years. Finding out which method is the most efficient and works best for training a machine to learn how to play blackjack will be the topic of this paper.

When considering which method of RL[3] to use when wanting to teach a machine how to play the game blackjack, there are many different options to consider. When considering the game blackjack the next cards that will be drawn are always unknown to the players and dealers. The hole card[4] is also a card that is not known to any player including the dealer, making this game an even more inherently stochastic model. Since bets are placed while playing this game, maximizing the profit on return for each game played is very important and is hard to do with the uncertainty of what the next card dealt will be and what hole card has been dealt to the dealer. This models the perfect Markov Decision Process where each game promises states and actions that are unknown and have not been explored yet.

When considering reinforcement learning it can be broken down further into two categories "On-Policy" or "Off-Policy" learning. On-Policy RL approaches are considered as methods that involve learning agents that learn from the value function denoted as v_π or q_π according to future possible states and future accumaltive rewards from moving through these possible states. These value functions for a given state at each step in time can be denoted as $v_\pi(s)$ or $q_\pi(s)$ for $s \in S$ where S is the set of all possible states. Off policy works when the learning agents learn by periodical taking actions that are not following the policy and rewards are not known to the agent.

[3]Reinforcement Learning

[4]Face down card that is dealt to dealer at the beginning of each game

Table 1: *Example table*

Blackjack RL Methods		
Name	On/Off-Policy	Greedy
Q Learning	Off-Policy	??
SARSA	On-Policy	??

Table 2: *Time Line*

Time Line 8/27-10/27	
Date	Task
09/20/2021	Semester Project Proposal Presentation
09/22/2021	Project Proposal Report
10/06/2021	Create Environment
10/17/2021	Training and Testing
10/20/2021	Midterm Presentations
10/27/2021	Midterm Project Report
10/27/2021	Midterm Project Demo

of the previous occupied state [GeeksforGeeks]. Above is a table that will be referred to when discussing what methods will be used for approaching these problems.

2.1. Q-Learning

One of the most well known and oldest methods listed is the "Q-Learning" approach. This approach is considered an off-policy TD control algorithm that can be defined as

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha\big[R_{t+1} + \gamma \binom{max}{a} Q(S_{t+1}, a) - Q(S_t, A_t)\big]$$

[1]

, where the steps taken are independent of what the next updated policy might be based upon that action. Depending upon the value of ϵ the frequency of random steps taken can vary and vary in how stochastic the model is. What is important to note is that even though

these models are stochastic the models are able quickly able to ensure a steady and step increase in profit after each game is played. Since this methods relies on deciding what actions to take based off of recent actions, the decision is completely independent of future rewards that might be gained by moving into the decided state, other then the current policy being used.

2.2. SARSA

The "SARSA" [5] is a slightly modified version of the Q -learning approach where the modification of the policy differs. Q-learning is considered an off policy where SARSA is an on policy [GeeksforGeeks]. The difference between the two, as iterated before, is that the action that is chosen to be taken is based off the accumulative reward off all future states that could possibly betaken. This approach does take a slightly more greedy approach but still involves exploration based off a given learning rate specified with in the model parameters.

3. Related Work

4. Methodology

A statement requiring citation [Figueredo and Wolf, 2009].

5. Conclusion

[5] State Action Reward State Action

References

[Sutton, R. S., & Barto, A. G. , 2018] Sutton, R. S., & Barto, A. G. (2018). Reinforcement Learning: An Introduction. Cambridge (Mass.): *The MIT Press.*, .

[GeeksforGeeks] "Sarsa Reinforcement Learning." *GeeksforGeeks*, 24 June 2021, www.geeksforgeeks.org/sarsa-reinforcement-learning/.

[Figueredo and Wolf, 2009] Figueredo, A. J. and Wolf, P. S. A. (2009). Assortative pairing and life history strategy - a cross-cultural study. *Human Nature*, 20:317–330.