Saishnu Ramesh Kumar
Harry Atulbhai Patel
Rong Jun Leong
Chester Lee

CSCI 154 - Project 2 Report (Blackjack)

Motivation:

This project on the Blackjack game is a great opportunity for us to understand the application of Monte Carlo simulation in a real-world scenario. The project allows the evaluation of different policies for playing blackjack and analyzing their effectiveness through win rates, providing insights into decision-making under uncertainty. By simulating and analyzing the complex system of blackjack, we can apply theoretical concepts and computational techniques to a real-world problem and gain practical experience in solving complex problems using computational tools. Additionally, this project offers hands-on experience in Monte Carlo simulation methods, which are widely used in the scientific fields, thus equipping us with skills applicable to various domains while developing a deeper understanding of blackjack and the strategies for increasing the likelihood of winning.

Problem Statement:

The objective of this project is to evaluate the success rates of five distinct Blackjack strategies under two game settings: single deck and infinite deck. The study aims to analyze the winning percentages of different strategies to determine their effectiveness in increasing the probability of winning in the game of Blackjack. By comparing the results of the different strategies, we can identify the optimal approach to maximize the player's success rate in the game.

Related Work and Background Material:

In preparation for this project, we first studied the rules of the blackjack game and some strategies that can be used to increase the win rate of a game. We also explored the Monte Carlo simulation methods and decided to use Python programming language to code this project. The following are the main points regarding the rules of playing Blackjack:

- At the start of the game, each player receives two cards, and one of the dealer's cards is hidden until the end of the round.

- Face cards (Jack, Queen, King) are worth 10 points, and Aces can be worth either 1 or 11 points, depending on the player's choice.
- During the game, players have three options: hit, stick, or burst.
    1. Hit means drawing another card from the deck to increase the player's hand value.
    2. Stick means holding the current hand value and ending the player's turn.
    3. Burst happens when the player's hand value exceeds 21, resulting in an automatic loss, regardless of the dealer's hand value.
- If a player's initial hand consists of an Ace and a card with a value of 10 points (10, Jack, Queen, or King), they have Blackjack and win the round.

Approach:

For this blackjack project, we chose to use the Monte Carlo simulation approach to approximate the efficiency of five different policies under two versions of the game. In our simulation, there are two versions where a single deck and an infinite deck are used respectively. In a single deck version, we simulated the game by shuffling a deck of 52 cards after every game, whereas in an infinite deck version, we simulated the game by randomly drawing cards from a deck with equal probability. The various policies used by the player in this simulation are as follows:

1. Stick >= Soft 17: This strategy involves sticking when the player's hand value is 17 or greater, including when an Ace is being used as an 11 (soft hand).
2. Stick >= Hard 17: This is similar to the previous strategy but applies only to hard hands (without an Ace).
3. Always stick: This strategy involves always sticking regardless of the hand value, which is a conservative approach that avoids the risk of going bust.
4. Stick >= 17, if dealer < 7 and player > 11, stick. Else hit.
5. Stick >= 17, if dealer < 7 and player > 14, stick. Else hit.

Experiment set-up (including Metrics, Benchmarks, Data collection):

We chose to run the simulation for 1,000,000 (one million) iterations or games to collect precise and accurate results. The more the number of iterations run, the more accurate the result it gave. The device used to run the simulation is MSI GF63, Intel i5, with Windows 11. The IDE used is Python IDLE and the script is written in Python language. In our script, the player needs to provide several inputs as follows:

1. The player is asked to select a policy to play the game.
2. The player is asked to choose between a single deck or infinite deck to use during the game.
3. The player is asked to specify the number of iterations or games they want to play.
4. At the end of the simulation, the script will display several statistics, including the number of wins, losses, and ties recorded during the game, as well as the winning percentage of the chosen strategy. The script also shows the total amount of time taken to run the experiment.

Results:

The total number of games run is 1,000,000 for each policy in different versions of decks.

Policies

1. Stick >= 17
2. Stick >= Hard 17
3. Always Stick
4. Stick >= 17, if dealer < 7 and player > 11, stick. Else hit.
5. Stick >= 17, if dealer < 7 and player > 14, stick. Else hit.

Policies	**Infinite Deck**				**Single Deck**			
	Wins	**Losses**	**Ties**	**Win Rate (%)**	**Wins**	**Losses**	**Ties**	**Win Rate (%)**
1	420688	489301	90011	**46.23**	421442	491595	86963	**46.16**
2	415313	498619	86068	**45.44**	415640	501563	82797	**45.32**
3	388142	565429	46429	**40.70**	389940	566376	43684	**40.78**
4	437273	482252	80475	**47.55**	439635	482152	78213	**47.69**

5	431399	482585	86016	**47.20**	434272	482317	83411	**47.38**

Conclusion and Contributions:

In general, the single-deck version and the infinite-deck version have a very minimal difference in their win rate. Policy 4 and Policy 5 have a better chance of winning than the other policies for both deck types. Policy 3 had the lowest winning percentage for both deck types. Overall, a player is more likely to lose a game than win a game.

From this simulation project, we have learned to apply the Monte Carlo method in solving real-world problems, especially in analyzing the probability of uncertainty. We believe that the player's win rate in a Blackjack game can hardly reach the 50 percent mark as Blackjack is a house-favor game. However, we still hope to implement the policy following the strategy chart found online to determine the win rate of the best strategy. More strategies such as various methods of card counting can also affect the win rate of the player.

Feedback: Based on the feedback provided, it appears that the presentation slides were generally well-organized and effectively conveyed information without excessive text. Despite there being a lack of feedback on potential changes that could be made to improve the presentation, it may have been beneficial on our part to provide more details on the approach taken to develop the program and the properties that were implemented.