

Millie N White

Blackjack Simulation

3/12/2021

Even though my husband is a poker dealer, I do not gamble, so I was intrigued by this project.

From my internet search, there seemed to be a lot of interest in strategies of increasing the odds of winning at blackjack against the house or casino.

My main interest in this project was determining if the number of games played increased the chances of winning the game.

How to play blackjack

The game starts with the player or players making bets and the dealer then deals themselves and each player 2 cards.

Even though both cards of the players are visible, the dealers second card is hidden.

Cards with 2 to 10 on their face are worth the value of the number on the face of the card.

King, queen and jack cards are worth 10, except for the Ace, which is worth 1 or 11.

For Aces, the player picks the value of 1 or 11, depending on whether the total hand would go over 21

The objective of blackjack is to beat the dealer's hand without going over 21.

If you get 21 points exactly on the deal, that is called a "blackjack."

For example, a King combined with an Ace is 21 and is Blackjack

There are 4 decisions a player can make after the initial hand has been dealt.

The player can hit, stand, double or split

Hit — draw an additional card; the player can hit as many times as needed

Stand — end your turn

Double — increase the bet amount by the initial bet, get one additional card, and end the turn

Split — If two cards have the same value, the player has an option to split the hand to have two hands.

My Python Blackjack game setup and game rules

For the purposes of this project, stand is the only decision in the blackjack game and there is only one player in the game.

The initial game will keep track of one player playing against the house.

There is betting in blackjack, but for the purposes of this project let us assume we are playing during a zombie apocalypse and money has no value and the satisfaction of the win is all the player is after.

The objective is to simulate how many wins and losses the player has playing against the dealer.

I will be using python functions for this simulation.

The following are the functions in my program and their use :

1. **main** function which has the bulk of the sequence of the program
2. **deal** function which deals cards to the players
3. **hit** function adds 1 card to player or dealer hand.
4. **sum_of_hand** function which totals up player and dealer hand.
6. **determine_winner** which determines the winner of the game, player or dealer.

Running the simulation

Two cards each, called a hand, are dealt to the dealer and the player using the deal function.

The game uses the sum_of_hand function to check if there is a winner.

If there is no winner, the game assumes the player stands and then checks if the dealer's hand totals less than 21, and if it does, he is dealt another card using the hit function.

I ran the simulation with 1 deck which had approximately 13 games, 10 decks, 50 decks and 100 decks.

The two global functions **number_of_games_won** and **number_of_games_lost** and **number_of_games_tied**, keep track of how many games are won, lost and tied.

The following table shows results obtained using my program to simulate my version of blackjack rounded off to 2 decimal places.

Number of decks	Number of games	Percentage of games won, lost and tied
1	13	tied 0 lost 6 won 7 percentage of games won 53.85 percentage of games lost 46.15 percentage of games tied 0.0
1	13	tied 0 lost 7 won 6 percentage of games won 46.15 percentage of games lost 53.85 percentage of games tied 0.0
10	127	tied 0 lost 54 won 73 percentage of games won 57.48 percentage of games lost 42.52 percentage of games tied 0.0
10	127	tied 1 lost 66 won 61 percentage of games won 47.66 percentage of games lost 51.56 percentage of games tied 0.78
50	638	tied 1 lost 326 won 311 percentage of games won 48.75 percentage of games lost 51.10 percentage of games tied 0.16
50	635	tied 1 lost 296 won 338 percentage of games won 53.23 percentage of games lost 46.61 percentage of games tied 0.16
100	1264	tied 4 lost 623 won 637 percentage of games won 50.40 percentage of games lost 49.29 percentage of games tied 0.32
100	1266	tied 3 lost 597 won 666 percentage of games won 52.61 percentage of games lost 47.16 percentage of games tied 0.24 77

Conclusion

The following table shows the averages of games lost, won and tied based on the number of rounds of blackjack.

Average number of games	Won	Lost	Tied
13	50	50	0
127	52.57	47.04	0.39
636	50.99	48.85	0.16
1265	51.50	48.23	0.28

It seems that if player options are limited to stand, the chances of winning are 50-50 with 13 games of blackjack, with a slightly better chance of winning as the number of games increase to 127.

There is a slight decrease as the games increase to 636 and a slight decrease as the games increase to 1265.

The percentage loss decreases slightly, and the percentage of ties increase slightly as the number of games increase.

In conclusion, I will not be taking up blackjack any time soon.

I enjoyed learning to simulate it, however, since my chances of winning are only slightly above fifty percent, I will save myself and my money and pick up another hobby.

References

Blackjack Simulation: Is There a Way to Beat Casinos?

<https://py.plainenglish.io/blackjack-simulation-is-there-a-way-to-beat-casinos-e23421ea9b28>

Let's Play Blackjack (with Python)

<https://towardsdatascience.com/lets-play-blackjack-with-python-913ec66c732f>