Math 381 HW 6 – Simulation of "no thanks"

Anthony, Aureliano

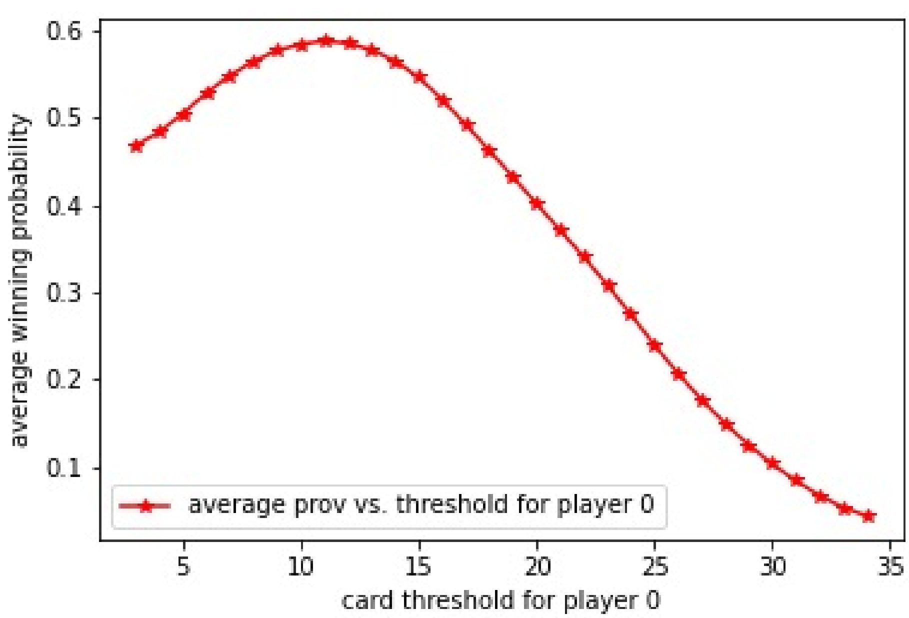
February 25, 2022

1 Introduction to Game *"No thanks"*

For this assignment, we are going to investigate different strategies for No Thanks! This game consist of 33 cards with numbers from 3 to 35, and each play is given 11 tokens, which players could use to decline a card later. Before the game start, someone shuffle the deck, then count off 24 cards face-down. The rest 9 cards won't be used during the game, and no player should look at them. The rule of game is on each turn, one card would be turned over and face up, the player either take the card with all the tokens on it, or pay a token to decline the card. After all 24 cards have been chosen, each player's score is the sum of values of all his single cards, plus that of the lowest card of each unbroken number sequence he has and subtracts the number of chips the player has. The player with lowest score wins the game. In the later sections, we will discuss and analyze 3 different game strategies as well as their effect on the probability to win the game with Monte Carlo Simulations.

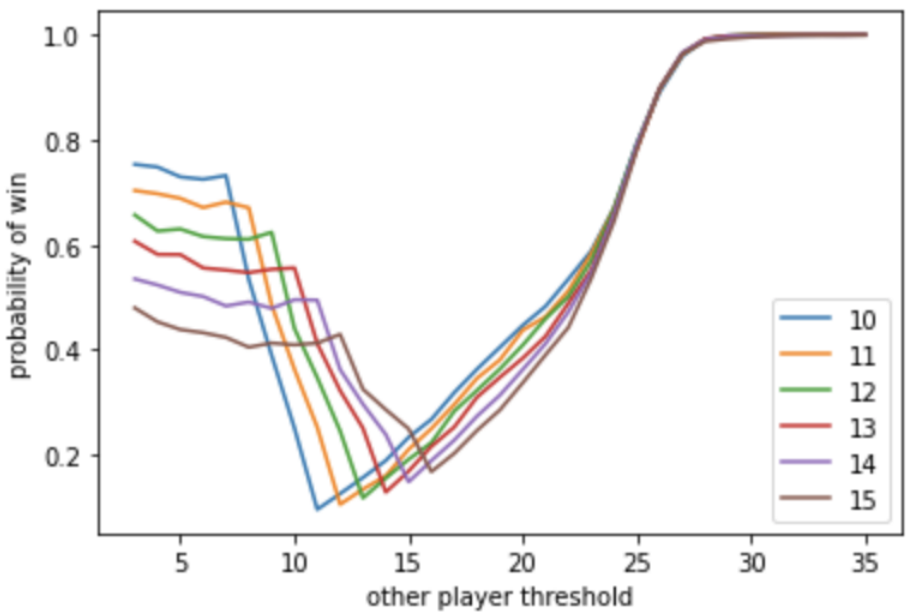
2 Strategy 0 : constant card threshold

We refer this strategy as the original strategy explained and implemented. That is, we will take the current card if the current card minus the number of tokens on the card is less than a constant threshold. We are going to determine the optimal base threshold strategy. Because we can't know other player's base strategy, we prefer to choose a value that has the highest average winning chance no matter what other players strategies are. We let player 1 to choose the base threshold strategy from 3 to 35 and play against other base thresholds from 3 to 35, and calculated the expected probability of winning.



From the graph, we can tell that the optimal threshold should be around 12 . So, we decide to take a closer look at the range 10 to 15. Below is the table of base threshold equal 10...15 against other base thresholds.

	5	10	15	20	25	30
10	[0.727, 0.741]	[0.244, 0.252]	[0.217, 0.230]	[0.442, 0.450]	[0.776, 0.787]	[0.9994, 0.9999]
11	[0.674,0.693]	[0.354,0.366]	[0.203, 0.217]	[0.424,0.438]	[0.775, 0.790]	[0.9993, 0.9998]
12	[0.619,0.632]	[0.442,0.456]	[0.185,0.195]	[0.404,0.419]	[0.783,0.796]	[0.9988 0.9995]
13	[0.559,0.582]	[0.550,0.572]	[0.166, 0.174]	[0.381, 0.398]	[0.781,0.791]	[0.9983 0.9993]
14	[0.496,0.516]	[0.478,0.497]	[0.142,0.156]	[0.356 0.367]	[0.776,0.794]	[0.99745 0.9983]
15	[0.435,0.449]	[0.402,0.417]	[0.238,0.260]	[0.322,0.338]	[0.775, 0.787]	[0.9958,0.9971]



The graph shows us that we can't find a universal winning base threshold. Every base threshold has its weakness, where its winning chance is smaller than 0.25. But here we choose 12 as the optimal base case since it has the highest average winning probability.

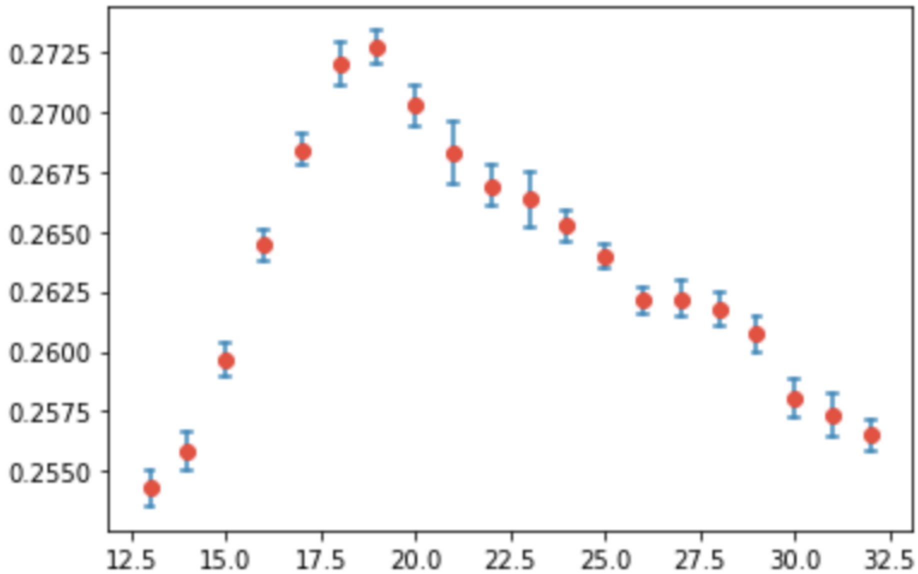
3 Strategy 1 : Passing to the next player if card is BIG

3.1 Strategy explanation

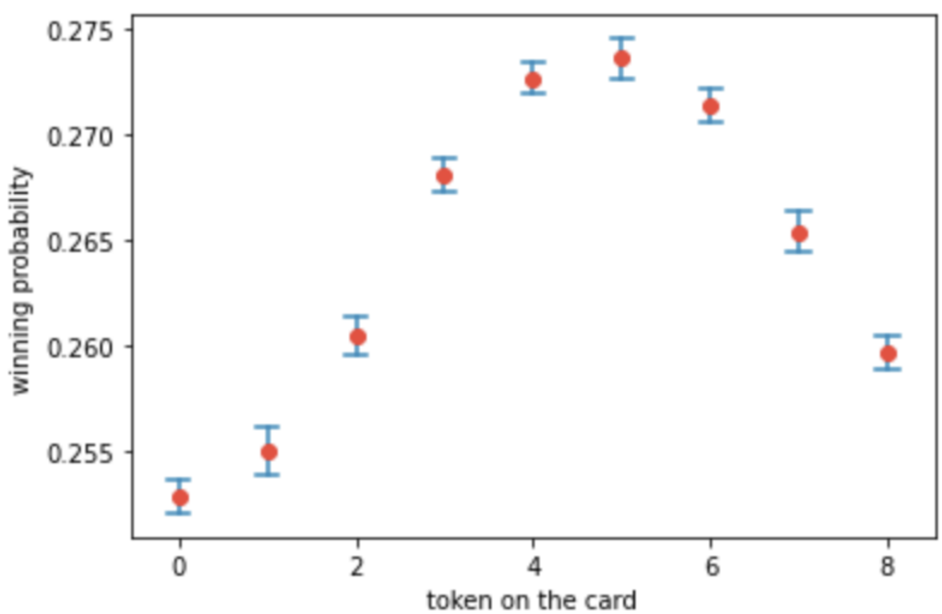
One approach to gain chips is to refuse a card that you could easily accept (since it would complete or add to one of your sequences), but that your opponent players would not desire (because it would get them too many points). Just send it around for another turn, it is possible that it would come back to you with more chips. This strategy adds more risks. We are risking tokens on the current card for more chips after loops it back.

3.2 Strategy implementation

To implement such strategy, we need to set a threshold value by which we can say if the given card is big or small. If the card has large value, we would take the strategy of passing it to the other players, and wish that it will come back after 1 loop with more tokens on it. However, if the card is small, and we have adjacent cards, we want to just take the card as what we normally did. Let's define our base strategy A, where each player pick the card when the value of card minus token is smaller or equal to 12, or the current card is adjacent to what player already have. We let three players to play strategy A. Then, we define strategy Bx as strategy A with the additional requirement about letting cards go one more round(we will try different token number) if the card value is larger than x, which could be any value between 13 to 35. We will let player one to play strategy B. Then, we let the four players to play against each other, and calculate and compare the probability of winning at the end.



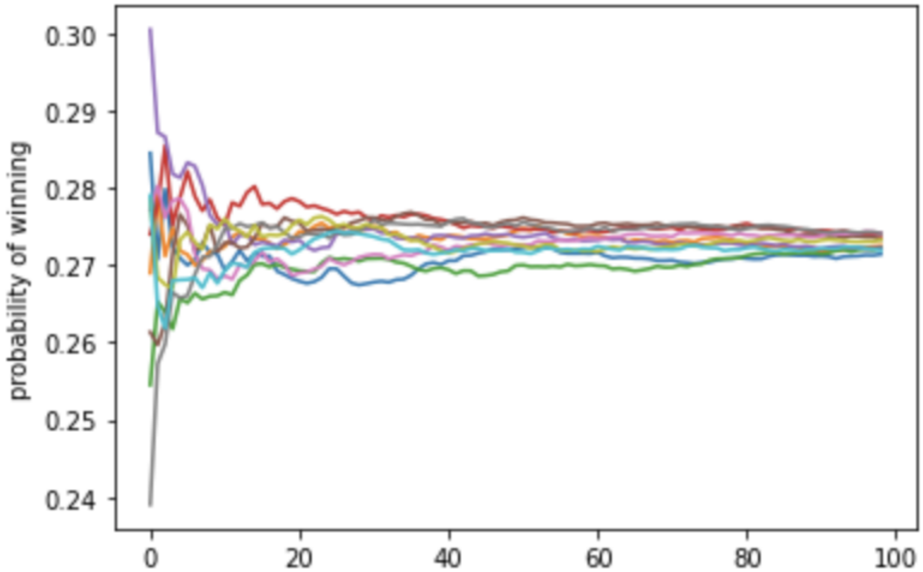
Above is a comparison between A and Bx strategies for x=13, 16, ...35.The y-axis for above graph is the probability of winning, and x-axis is the big value threshold. From the graph, even though there are some overlaps between confidence interval, we can tell that, in general, choosing 19 as large value threshold is the optimal strategy, though 18 is possibly as good as 19. We couldn't make the confidence interval smaller due to constraint of computation power.
Furthermore, we investigate the threshold on the number of tokens on the card.

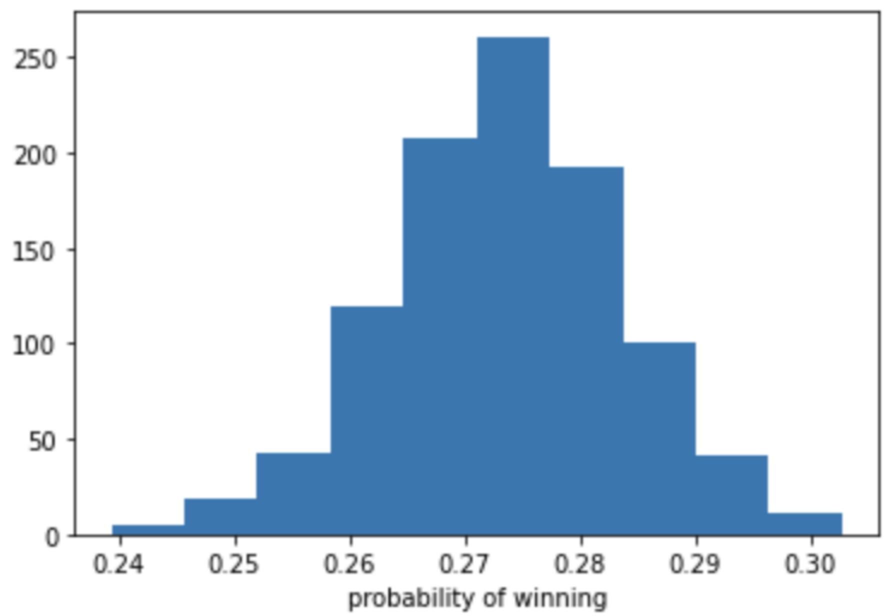


This graph shows us that when we choose 19 as the large value threshold, it has highest chance of winning if we pick up the card when the number of tokens is at least 5, though we possibly have same winning chance if we pick the card when it has at least 4 tokens on it.

3.3 Plot & Analysis

We do 1000 runs of 100000 simulated games that yield 100000 estimates of the winning probability of this strategy. In addition, The central limit theorem says that these estimates should be close to normally distributed. To verify that, we plot a histogram of these estimates.





From the first graph, we see that the probability of winning do converge as the number of games increase.

From the histogram, we can see they are approximately normally distributed. Furthermore, I run the simulation 10 times with 10000 games each. The values are 0.270, 0.271, 0.273, 0.273, 0.275, 0.275, 0.275, 0.275, 0.277, 0.277. We could use the fact that normally distributed tells symmetric about the mean, then we can estimate the probability that all 10 estimates are one side of the true probability as $\frac{2}{2^{10}} \approx 0.0019$ Thus, there is a approximate 99.8 percent probability that the true probability is between 0.270 and 0.277. The lower bound of the confident interval is larger than 0.25, which suggest that this strategy does increase the probability of winning.

4 Strategy 2 : Gambling on filling the gap later

4.1 Strategy explanation

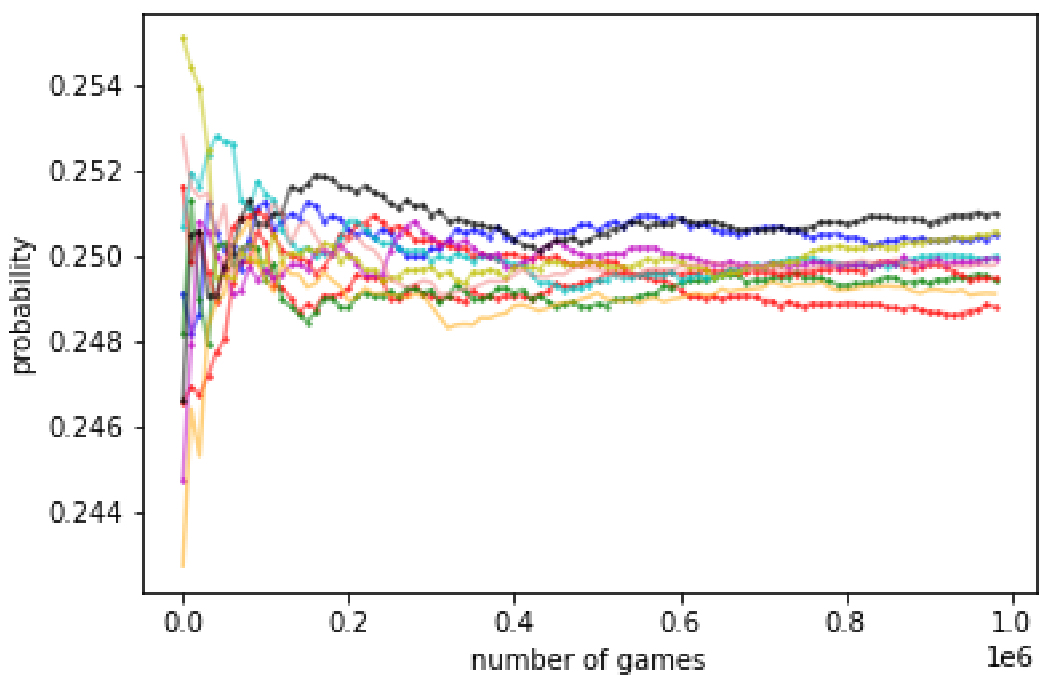
Suppose we have already got a 30 in my cards list, and the current card is 28. The normal approach would be not to take it because it is not adjacent with 30. (By adjacent, I mean the number is either greater than or smaller than the card I have by 1.) However, if we know that there is a possibility that we will get 29 later before the game ends, we can fill the gap, and we will have a better score out of it. So our second strategy is to take the card if the current card number is ± 2 to any card I have in my list. We take the card, and we hope that there will be an adjacent card coming later in the game.

4.2 Strategy implementation

To implement this strategy, for one thing, we need to keep track of all the cards that have already shown so that we can have an idea that if there will be adjacent card coming later. If the adjacent card has already been shown in the game, we know that there would be no more of it, so we will just keep the same approach – either take it if card number minus tokens on it is smaller than threshold or pass it to the next player by placing one token on it. However, if the adjacent card is not shown yet, and we know that it is possible to have it later in the game. So we will take the card, and wish that we will have adjacent card later. In our example, We let all four players to pick the card when the value of card minus token is smaller or equal to 12, or the current card is adjacent to what player already have. In addition, we let one player to choose the card if the value of current card plus 2 equals to the card he already have and the value of current card plus 1(gap) has not showed yet. Then, we let the four players to play against each other, and calculate and compare the probability of winning at the end.

4.3 Plot & Analysis

We can simulate the game, and plot the probability of 10 runs as below :



From the first plot, we can see that the probability converges to 0.25, which is the same probability as if we do not use any strategy for the game. Thus, we can conclude that the strategy to take the "gap card" does not increase the probability of winning.
From the second plot, we can see that the distribution of simulating 10 runs with 10000 games for each run is normal with mean centered around 0.24. This result is consistent with central limit theorem. We can calculate the confidence interval in this way : Since the results from 10 runs are 0.24791, 0.249581666667, 0.2492075,0.25055,0.2508375,0.24733, 0.251861666667, 0.249748333333,0.2495675, 0.251145. based on the same argument as in strategy 1, we can conclude that 99.8% confident that the true probability lies within the interval [0.24733, 0.251861666667].

To further test if this strategy can improve the winning probability beyond 0.25, we need to have a stricter confidence interval. Thus, we run the simulation 10 times but each time we simulate more games – instead of 10000, we changed to 100,000. The confidence interval derived from this longer simulation is : [0.24908350000000137, 0.250177916666668]. This indeed gives us a bit stricter interval than the previous interval. However, since this interval still includes 0.25, we still cannot conclude if this strategy is better than 0.25 or not. If we keep increasing the number of simulation to larger value, we found that the larger simulation does not yield a better result. Thus, the only conclusion out of this strategy is being inconclusive. Even though we did not derive a interval that is either above 0.25 or lower, we can confidently say that this strategy will not improve the winning probability too much nor decreasing it too much.
If we compare this strategy with respect to strategy 1, then we can find that the confidence interval is [0.242618333333, 0.246526666667]. This interval indicates that the probability of winning is less than 0.25 if we are competing with strategy 1.

Besides of comparing the strategy 2 with respect to strategy 0, we can also compare it with strategy 1. With 10 runs of simulations, we find that the 99.8% confidence interval is : [0.241605, 0.24594]. This interval of probability is slightly lower than that when we run strategy 2 with reference from strategy 0 even though the difference is very subtle. This slight decrease of winning probability can be explained by that we are referring with strategy 1 which is a slightly better strategy than 0. If all other 3 players use this slight better strategy, it make sense that we lost some winning probability.

5 Strategy 3 : non-constant card threshold

5.1 Strategy explanation

By the rule of the game, we know that we are forced to take the current card if we do not have any tokens left. If the current card is large, and meanwhile, we do not have an adjacent card, we are in a very terrible situation since taking the card would significantly lower our score. Thus, to avoid such terrible situations happening, we want to ensure that we will avoid spending all of our tokens. In other words, we want to increase the card threshold if we have fewer tokens left. With larger card threshold, we are more likely to take the card instead of spending another token for it.

5.2 Strategy implementation

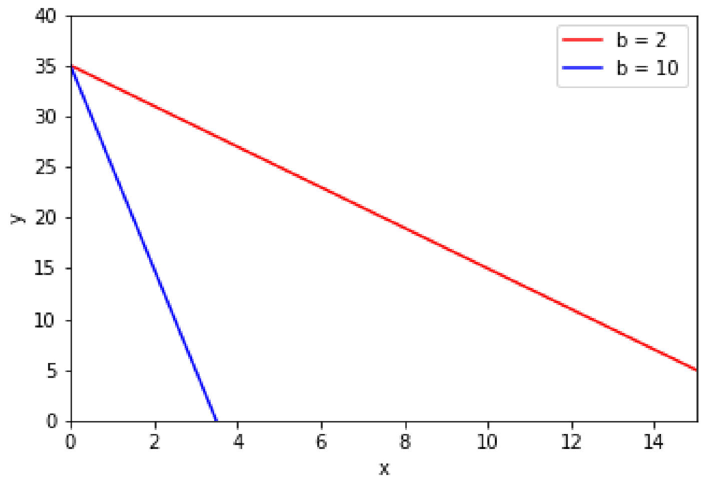
As explained above, we want to avoid spending all of our tokens before the game ends. One way to avoid it is to set the card threshold non-constant. In the original code, we set a constant value of card threshold, for instance, 20 for me and 25 for other players. To implement this strategy, we want to set the card threshold larger when tokens left is fewer. So we came up with a linear functions to achieve this :
$card\ threshold = 35 - bx$ for various positive b
where x is the number of tokens left for all the functions above.

Since we want to isolate the effectiveness of this strategy without the help of holding different card threshold for different players, we will investigate the effect of this strategy versus the strategy which is that all players share the same constant card threshold. Thus, all of the analysis below of strategy 3 will have the reference of strategy 0(which means that other players will use a constant card threshold of 12) to figure out the best value of b to maximize the winning probability. After finding the best slope, we can compare the optimal linear function strategy with respect with the previous strategies, and see which one is better.

5.3 Plot & Analysis

5.3.1 Linear threshold function

We defined the linear threshold as $35 - bx$ where b is any positive value. With larger the value of b, we can expect to have a steeper straight line. For example, we can plot the function $35 - bx$ with b = 2 and b = 10:



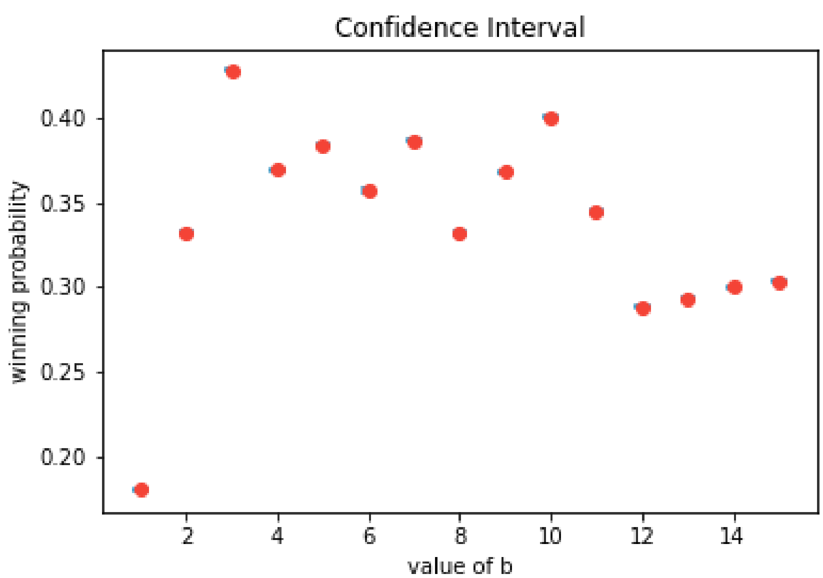
By using the linear function with $b = 10$ as above, we yield the following simulation results :

By running the simulation 10 times, we calculated the 99.8%confidence interval as [0.39894833333333335 , 0.4023133333333332]. We can see that the lower bound of the confidence interval is higher than 0.25. Therefore, we can conclude that the strategy of using linear function with b = 10 can improve the winning probability.

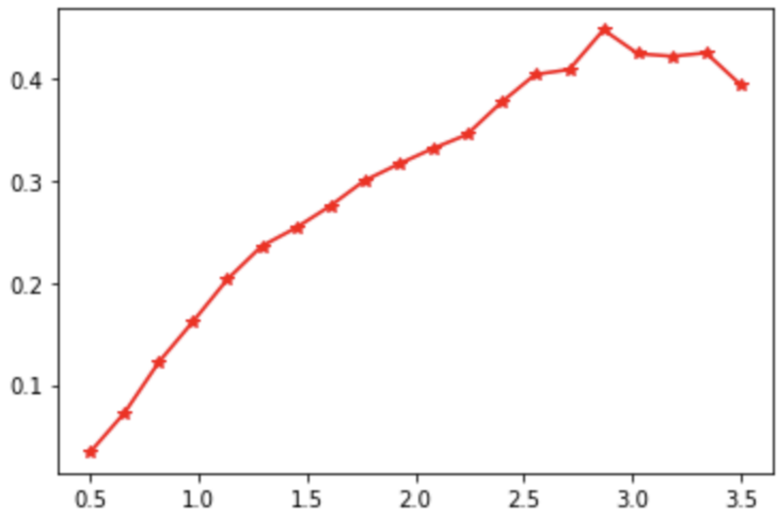
By varying the value of b from 10 to 2, we can again repeat the simulation, and yield the following result :

Same as previous step, we can calculate the confidence interval as : 0.3308416666666665 , 0.3351866666666663. Since this interval is also higher than 0.25, we can conclude that this strategy can improve the winning probability. However, comparing with the linear function with b = 10, the probability from b =2 function is lower than the probability from b = 10 function.

After having a general idea of how does the value of b affecting the probability, we can investigate all of the options of b from b = 1 to b = 15, and find the best value of b that maximize the winning probability.
We iterated through the value of b from b = 1 to b = 15, and plot the winning probability with 99.8% confidence interval as following :



We make a more detailed investigation for b ranging from 0.5 to 3.5, and here is the plot:



From the plots above, we can see that the max winning probability is when b = 3, corresponding to a mean probability of 0.428 and a 99.8% confidence interval of [0.426476 , 0.4291216]. Therefore, we can conclude that within the range from 1 to 15, the optimal slope for linear function is 3, and then we can compare this optimal linear function with previous strategies, and see which is better.

5.3.2 Summary and comparison of non-constant card threshold functions

So we can compare the optimal linear function with previous strategies. We implemented by setting that player 0 will use strategy 3 with optimal linear function, and other players will use the previous strategies. We run this simulation 10 times, and we can have the 99.8% confidence interval. Comparison of the results :

Comparison of linear function with b = 3 vs. previous strategies	
Threshold function	99.8% confidence interval
versus Strategy 0	[0.4264766 ,0.4291216]
versus Strategy 1	[0.39284, 0.397813]
versus Strategy 2	[0.42472583,0.4283]

This result makes sense. If we are competing with strategy1 which is a better strategy than strategy 0 as shown in previous section, we lost some winning probability. But if we compete with strategy 2 which is an equally-good strategy as strategy 0, then we have approximately same winning probability.

6 Conclusion & Suggestions for winning

In summary, we first found that the optimal card threshold to maximize the average winning probability is 12 and we call this strategy as strategy0. Then we come up with 3 new strategies, and we compare each strategy with reference with the strategy 0.
The first strategy we come up with is to set another threshold. If the current card value is larger than the threshold, then we want to pass it another round so that we could have more tokens on the card. The confidence interval we found for this strategy is [0.270,0.277].
Then we come with another strategy – instead of just taking the adjacent cards, we also want to take the cards that gap one with any cards we had. In reference with strategy 0, we found that this strategy gives us a confidence interval of winning [.249, 0.250]. If we compare it with strategy 1, we found confidence interval of winning [0.2426, 0.246].
Then the third strategy we come up with is to set a non-constant card threshold function. The function we chose is a linear function in the form of $y = 35 - b*$tokens we had. By varying the value of b, we found that the optimal b that can give us the best winning probability is b= 3, and the confidence interval with reference to strategy is [0.426 ,0.429]. With reference to strategy 1, we have confidence interval of [0.392, 0.397]. With reference to strategy 3, we have confidence interval of[0.424,0.428].

Thus, by such comparison, we found that the strategy 3 is the best strategy among all of 3, then is the strategy 1, and the strategy 2 does not yield much difference from no strategy scenario.

Therefore, we suggest to use strategy 3 with a linear card threshold function with slope = 3. This strategy can boost the winning probability to around 42%, which is way higher than 0.25 winning probability.

7 Appendix :

We have attached out Python code in the Github repo, please checkout the following repo :
Our Github repo – branch HW 6
Note that we changed the structure of original code for the simulation by making it more Object-oriented. We write different strategies code to different files, and we only import those files when we want to use that strategy in the simulation. We also split the initialization of the game as well as score calculations into separate files.