Project 1: Martingale

Mengya Ji
mji49@gatech.edu

Abstract—This project tests the Martingale betting strategy in American roulette using Monte Carlo simulation. The Martingale system doubles the bet after each loss until a win occurs, aiming for steady short-term gains. In Experiment 1, we ran 1000 episodes without a bankroll limit. Many players reached the profit target of +$80, but the average winnings were still negative because of the roulette house edge. In Experiment 2, we added a realistic $256 bankroll. With this limit, the chance of reaching +$80 dropped a lot, and the chance of losing the full bankroll (−$256) increased.

1 INTRODUCTION

This project investigates the Martingale betting strategy in the setting of American roulette, based on a strategy in which the gambler doubles the bet after every loss and resets to a $1 bet after a win.

2 EXPERIMENTS

Given the rules of American roulette, the probability of winning a single spin when wagering on black is 18/38, since there are 18 black pockets out of 38 total (the two green 0 and 00 count as losses).

2.1 UNLIMITED BANKROLL

In this case, the gambler is assumed to have unlimited funds and can always cover the next doubled bet, no matter how long the losing streak. Play continues until either total winnings reach +$80 or 1000 spins have been completed. This experiment shows the idealized Martingale, where financial constraints do not exist.

2.1.1 Plot of 10 episodes simulation

We ran the simple simulator for 10 episodes starting at $0. Most paths trend upward at first, but there are sharp downward dips caused by losing streaks and the bet-doubling rule. When an episode reaches +$80, the line flattens because we carry that value forward. Some dips extend beyond the y-axis limit (−256), which is expected under these bounds.

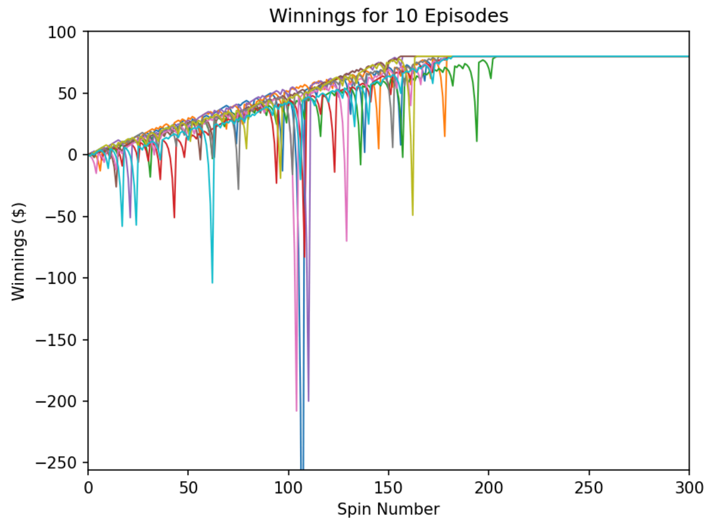

Figure 1—winnings for 10 episodes

2.1.2 Plot of Mean

We repeated the simulator for 1000 episodes and computed, for each spin, the mean winnings across episodes and the population standard deviation.

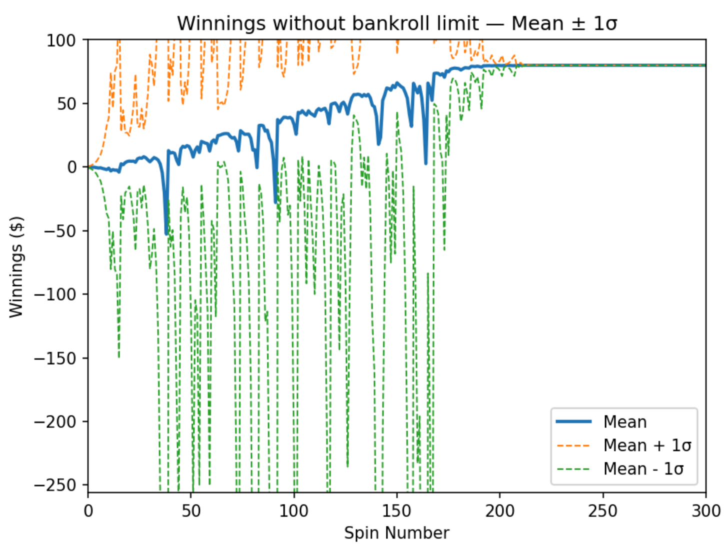

Figure 2 — mean ± 1σ for winnings without bankroll limit

Once an episode hits +$80, the line stays flat because we forward-fill that value. The upper band (mean + σ) sits near +$80 since many runs finish early. The lower band (mean − σ) drops fast because a few runs keep playing and hit long losing streaks. So the results are skewed: lots of early winners make things look good, but a small number of big losses pull the spread down.

2.1.3 Plot of Median

We then computed, for each spin, the median winnings across episodes and the population standard deviation.

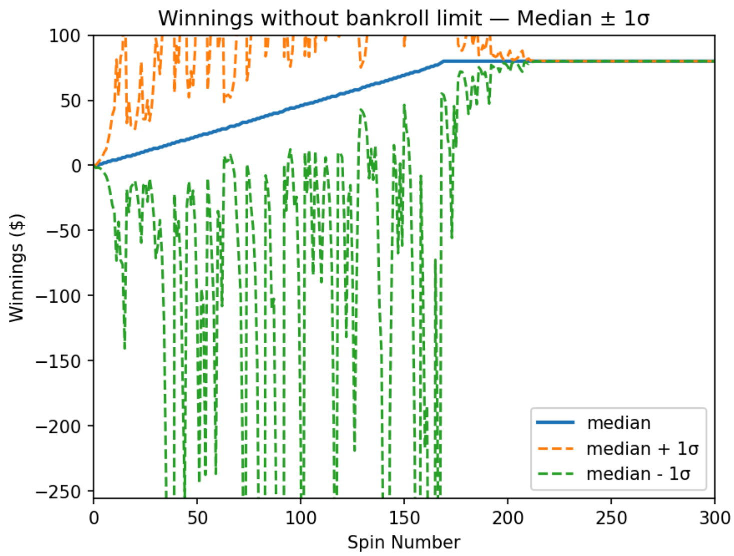

Figure 3 — median ± 1σ for winnings without bankroll limit

The median starts near $0 and rises as more episodes hit the +$80 target; once over half of the episodes have reached +$80, the median flattens at +$80 (forward-fill). The upper band (median + 1σ) quickly sits near the cap because many runs finish early. The lower band (median − 1σ) drops far below zero—those are the fewer runs that keep playing through long losing streaks. This shows a skewed distribution: capped upside, open downside during play. The spread widens with spins, then stabilizes once most runs have finished.

2.1.4 Analysis

In Experiment 1, based on the experiment results,we can calculate the estimated probability of winning exactly $80 within 1000 sequential bets. We estimate the probability of finishing exactly +$80 by checking the final winnings of each episode and taking the fraction equal to 80 (p_success = (df2.iloc[:, -1] == 80).mean()). Using 1,000 simulated episodes (forward-filled) this yielded p_success = 1.000 (1000/1000).

We can also calculate the expected value of winnings after 1000 sequential bets.

By definition, $E[X] = \sum_{x} x * P(X = x)$. In our simulation winning exactly $80 within 1000 sequential bets is 100%, therefore E[x] = 80 * 100% = 80. The expected value of winnings after 1000 sequential bets is $80.

Also, in Experiment 1, the standard deviation lines converge with one another to $80 as the number of sequential bets increases. This is because by the end, essentially all runs have hit +$80, so the distribution collapses to a point mass at 80, $\sigma \to 0$, and both bands equal the mean at about +$80.

2.2 Limited Bankroll

In this version, the gambler begins with a fixed bankroll of $256. The same rules apply as in Experiment 1, except that if total losses reach −$256, the gambler is bankrupt and the episode ends. An additional corner case is handled: if the required doubled bet is larger than the amount of money left, the gambler simply wagers the remaining bankroll. This experiment captures real-world risk by showing how often a player runs out of money before hitting the +$80 profit target.

2.2.1 Plot of Mean

We repeated the simulator for 1000 episodes and computed, for each spin, the mean winnings across episodes and the population standard deviation.

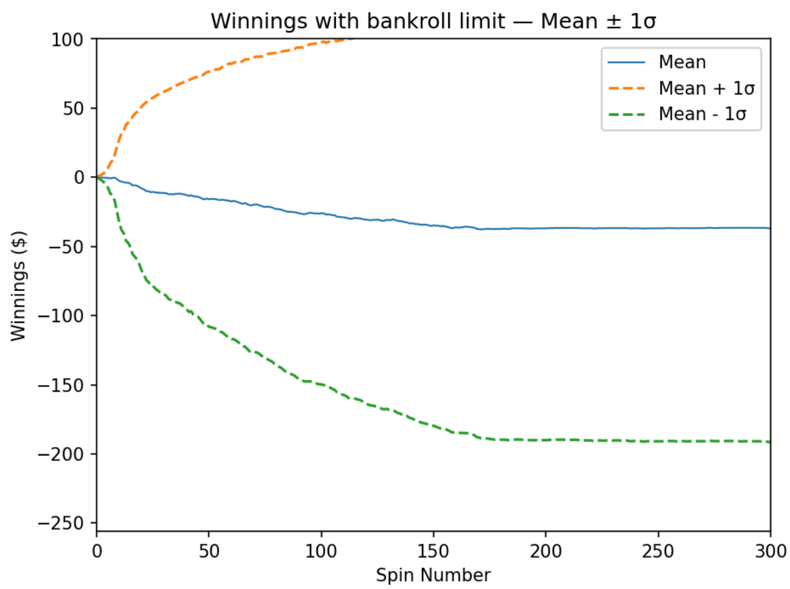

Figure 5 — mean ± 1σ for winnings with bankroll limit

With a bankroll cap, the mean drops below \$0 and then levels off. The upper band quickly climbs to about +\$80 and stays there. The lower band falls toward the −\$256 loss floor and then flatten.

2.2.2 Plot of Median

We then computed, for each spin, the median winnings across episodes and the population standard deviation.

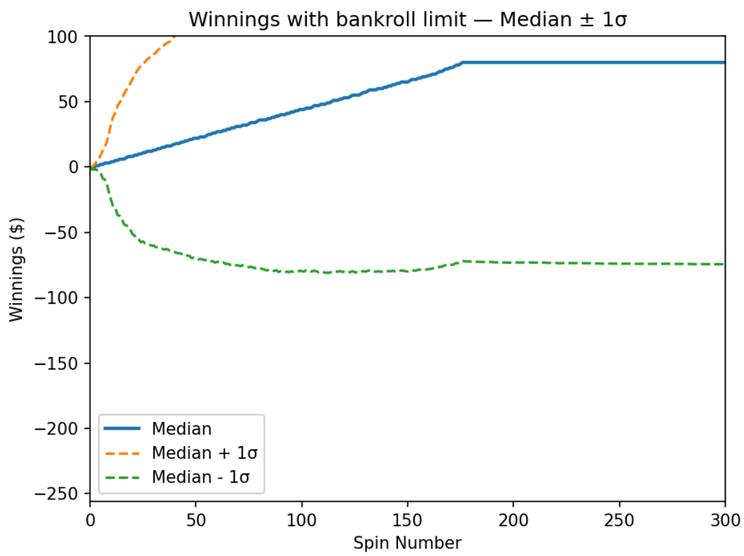

Figure 5 — median ± 1σ for winnings with bankroll limit

The median climbs from \$0 and reaches +\$80 then stays flat due to forward-fill. The upper band (median + 1σ) keeps on the +\$80 cap because many runs finish early. The lower band (median − 1σ) drops to negative and then levels off.

2.2.3 Analysis

In Experiment 2, based on the experiment results,we can calculate the estimated probability of winning exactly \$80 within 1000 sequential bets with the bankroll limits. We estimate the probability of finishing exactly +\$80 by checking the final winnings of each episode and taking the fraction equal to 80 (p_success = (df3.iloc[:, -1] == 80).mean()). Using 1,000 simulated episodes (forward-filled) this yielded p_success = 0.637.

We can also calculate the expected value of winnings after 1000 sequential bets.

By definition, $E[X] = \sum_{x} x * P(X = x)$. In our simulation, each of the results has an equal probability of 1/1000, therefore $E[X] = \sum_{x} x/1000$, which is the mean of the final winnings. We can use the code (ev_exp2 = df3.iloc[:, -1].mean()) to generate the expected values, which yields: -41.403.

In Experiment 2, the upper standard deviation line (mean + stdev) stabilize at a maximum, and lower standard deviation line (mean – stdev) stabilize at a minimum value. This happens because with a bankroll cap and forward-fill there are two absorbing outcomes: +$80 or −$256. As more episodes finish at one of these, the distribution of winnings stops changing, so the mean and the standard deviation become time-invariant, and the bands stabilize. They do not converge, they remain separated at the maximum and minimum.

3 CONCLUSIONS & ANALYSIS

3.1 Conclusions

Our simulations show that Martingale only looks good when money is unlimited. In Experiment 1, almost every run eventually reaches +$80. In the realistic case (Experiment 2) with a $256 bankroll, many runs still hit +$80, but enough go bust at −$256 that the average final result is negative. The bands flatten between the two endpoints and do not converge. Overall, Martingale produces frequent small wins but is not profitable on average once real-world limits are included.

3.2 Further Analysis: benefits of using expected values

There are lots of benefits of using expected values when conducting experiments instead of simply using the result of one specific random episode. Using expected value means averaging results over many episodes instead of judging by one lucky or unlucky run. This reduces noise and makes findings stable and reproducible, reflects the whole outcome distribution by weighting each result by its probability, and gives a fair single number to compare strategies. It also exposes cases where frequent small wins are outweighed by rare big losses, and it pairs naturally with uncertainty measures like standard deviation or confidence intervals. In short, EV shows what typically happens—not just what happened once.

4 REFERENCES

1. Wikipedia (accessed 2025), Roulette
2. Wikipedia (accessed 2025), Expected Value