Project 1 : Martingale

ML4T FALL 2024

Huida David Shi

hshi320@gatech.edu

1.1 Estimated probability of winning $80 within 1000 sequential bets [question 1]

Let's estimate the chance of not winning $80. We have to lose 921 times out of 1000 spins, but the percentage chance to lose is estimated only at 52.63% per spin. With an infinite bankroll you can lose streak for a very long time and still win it all back on the next round.

% chance of losing 921 times

$$\left(\frac{20}{38}\right)^{921} = 0$$

The calculation is denoted as (20/38)^921 which is practically equal to 0%. That means there is almost a 100% chance to win $80 in 1000 spins.

1.1.1 *FIGURE 1*

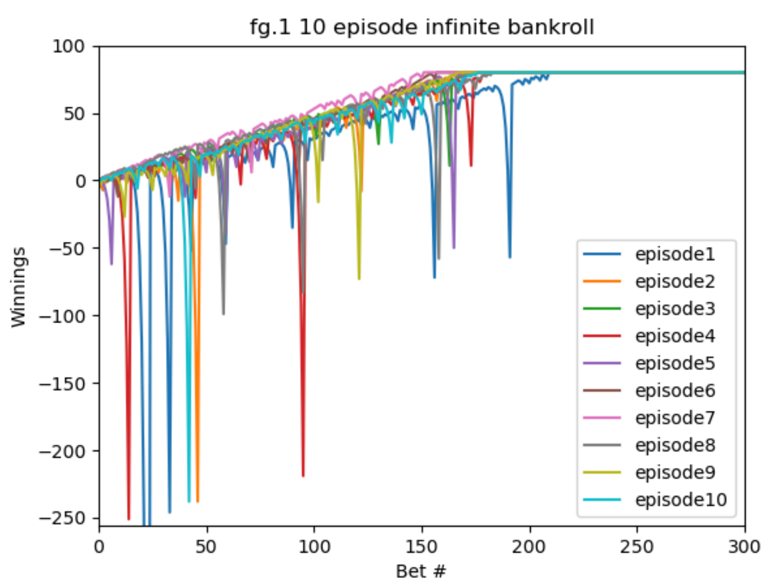


Figure shows a simulation of 10 episodes, target of $80 is reached at around 180-220 spins.

1 episode = 1000 spins

1.2 Estimated expected value of winnings with infinite bankroll [question 2]

The estimated expected value of winnings is $80. In figure 1-2 the average winnings converge to our win limit (80) in less than 250 spins. Expected value is denoted as

$E(X) = \Sigma (x * P(x)) = \mu$ $1.00 * 80 + 0.00 * y = 80$

y is any value not reaching 80 by the end of episode

When there are enough spins the expected value converges with the mean.

1.2.1 *FIGURE 2*

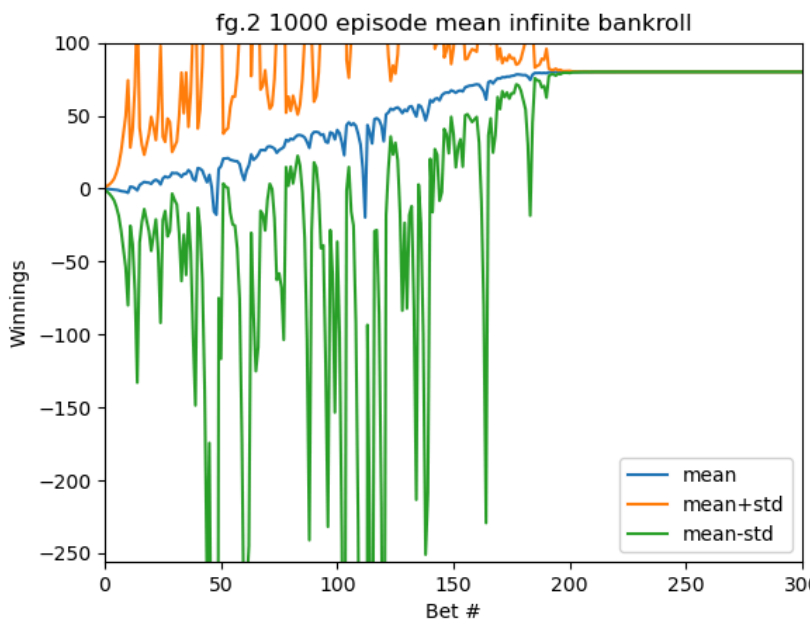


Figure shows the average winnings of a simulation of 1000 episodes. Graph shows convergence at $80 which is the win limit we set.

1.3 Upper lower maximum minimum standard deviation lines [question 3]

The upper and lower standard deviation lines converge and stabilize when we reach the win limit. In Figure 2 the standard deviation stabilization is hard to determine due to the plot axis limits and low number of bets. In Figure 6, the win limit is increased to $200 and the graph is zoomed out. We can see the standard deviation at the 400th spin mark is still very volatile. There is no visible pattern to conclude having more spins will stabilize the standard deviation.

However the standard deviation does converge when hitting the limit because the simulation stops. Since the martingale method is structured to regain all previous losses, increasing the number of bets will make each individual bet less sensitive, ultimately showing the convergence variance when the limit is reached.

1.3.1 *FIGURE 6*

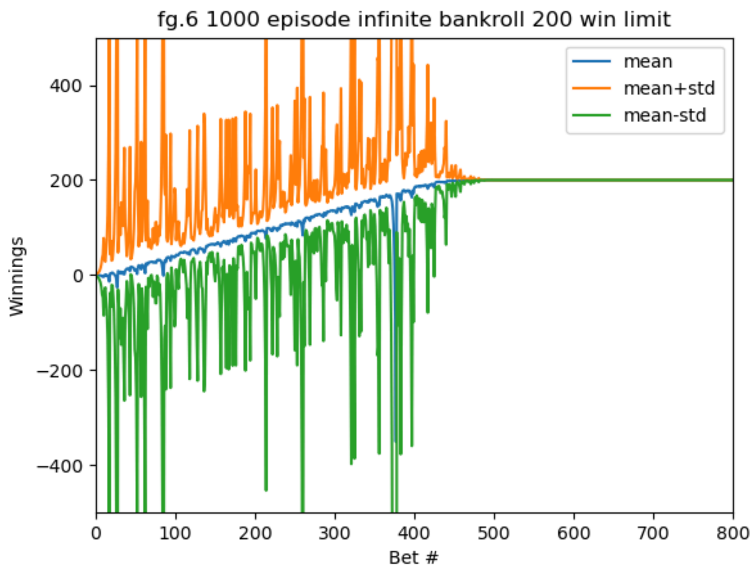


Figure shows the mean winnings of a simulation of 1000 episodes with a $200 win limit. The standard deviation is very volatile right until limit. The Standard deviation at 300-400 spins is higher than then 100-200 spins. There is no clear pattern on standard deviation until the limit is reached.

2.4 Estimated probability of winning $80 within 1000 sequential bets with 256 bankroll [question 4]

To calculate the true probability we need to sum up the probability over all possible sequences of wins and losses that reaches $80 without losing the whole bankroll. This is pretty complex to calculate due to the high factorial amount of combinations. Instead we will simulate the 1000 sequential bet 1000 times and check the value of the last integer of every episode. Then we can calculate the probability of winning $80 by

Count of array ending in 80 / 1000

In our simulation 65.94% of 1000 episodes successfully reached $80 without losing the bankroll.

2.4.1 *FIGURE 7*

```
def happy_ratio(array):  1 usage  new *
    happy = 0
    sad = 0
    for i in range(len(array)):
        if array[i,-1] == 80:
            happy = happy + 1
        else:
            sad = sad + 1
    return ('probability of winning: ' + str(happy/(len(array[0])+1)*100) +
            '% | probability of losing: ' + str(sad/(len(array[0])+1)*100) +'%')

def test_code():
    """
martingale ×
/opt/anaconda3/envs/ml4t/bin/python /Users/davidshi/Documents/ML4T_2024Fall/martingale/martingale.py
probability of winning: 63.536463536463536% | probability of losing: 36.36363636363637%
```

Figure shows a function that gets the last integer from every episode and calculates the probability of reaching $80 by using the ratio of (count of array ending in 80 / 1000).

2.5 Estimated expected value of winnings after 1000 sequential bets with 256 bankroll [question 5]

Using the expected value function

E(X) = Σ (x * P(x)) = μ

(0.6594 * 80) + (0.3636 * -256) = -$40.33

Expected value = $-40.33

2.6 Upper lower maximum minimum standard deviation lines with 256 bankroll [question 6]

The upper and lower standard deviations stabilize at around the 200th bet mark. The standard deviation lines don't converge and become more parallel as the sequential bet increases. As the sequential bet, it dwindles down to two outcomes. Win $80 or lose your whole bankroll with no inbetween.

2.6.1 ***FIGURE 4***

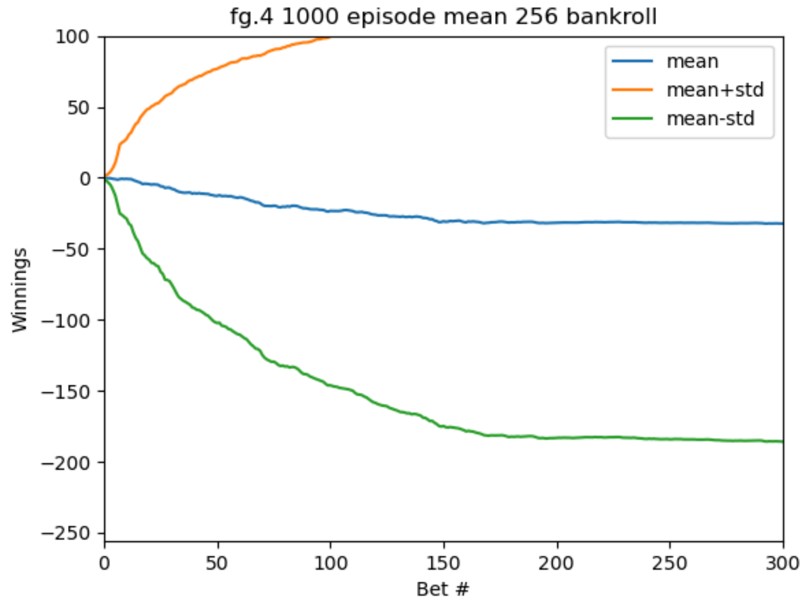


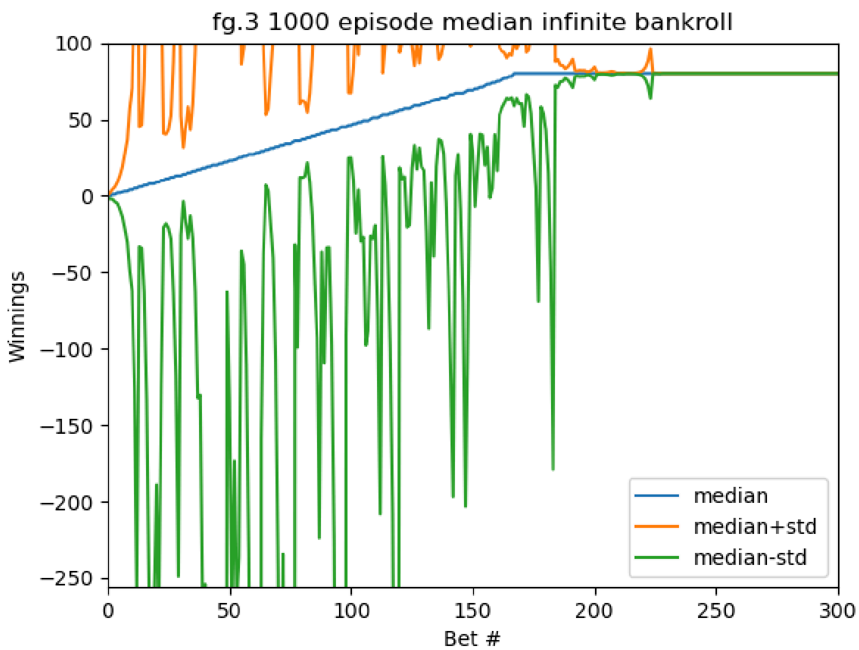
Figure shows mean, upper and lower standard deviation stabilizing as the sequential bet increases. All three lines become parallel showing no convergence.

2.7 Benefits of expected values when conducting experiments instead of simply using the result of one specific random episode? [question 7]

Simply relying on one specific random episode can lead to risky exposure to chance. In experiment 2, if I relied on only one random episode which happened to be successful then I would be sitting at a roulette table right now. However after running the simulation 1000 times we can clearly see the plan only works 65% of the time. By running the simulation many times and averaging the results we reduce noise and variability to ultimately predict more consistent outcomes. The average results or expected value will smooth out randomness and outliers.

2.8 Rest of Figures

2.8.1 *FIGURE 3*



2.8.2 *FIGURE 5*