P1 Martingale Report

Vishu Gupta

vgupta359@gatech.edu

guptavishu55@gmail.com

1 Questions

1.1: Question 1)

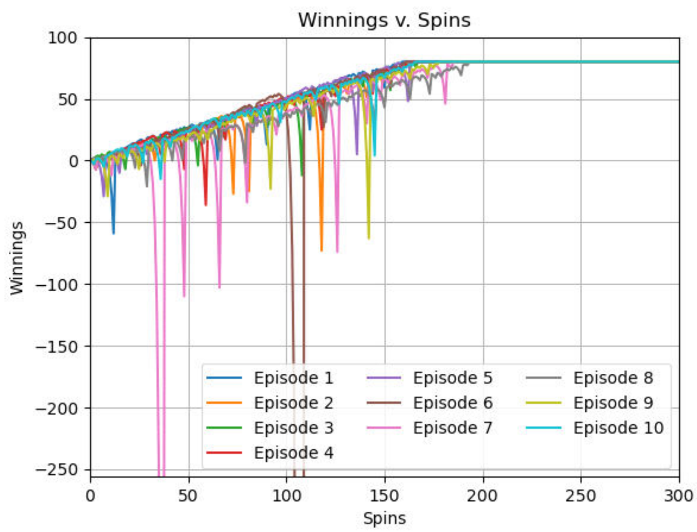

Figure 1

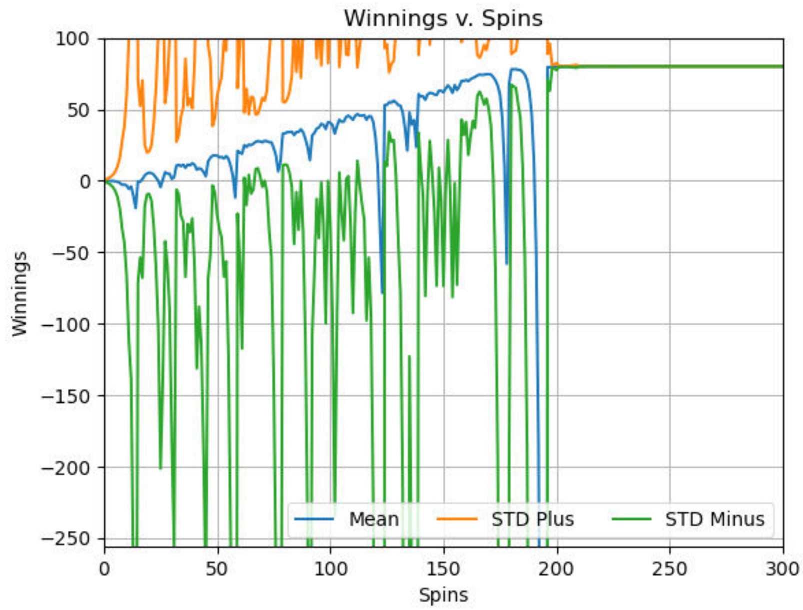

Figure 2

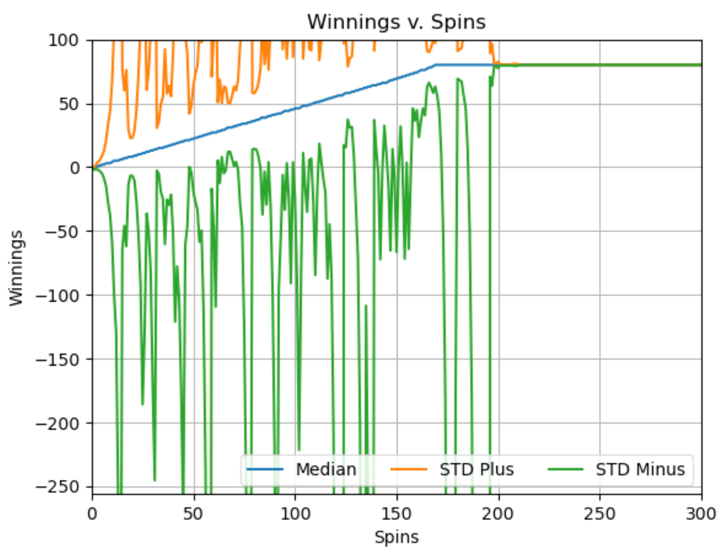

Figure 3

For this experiment, in the first situation we have 10 episodes. If we look at the 1000th spin for each episode we know that the value generated is always 80. Since the 1000th value is always 80 at the 1000th spin for this experiment based on mathematical computation we can say that the probability of winning $80 after one episode is 100%. The reason we can say this is because if we look at the 1000th spin for all the episodes there is no other number that the last spin of the episode equals other than 80. For example, if we had 5 episodes that ended with $0 winnings on the last spin, and 5 episodes representing $80 winnings on the last spin then we can say that the probability of having won $80 after one episode would be 50%. The probability of having won

$0 after one episode would be 50%. And finally, the probability of winning any other amount other than $0 or $80 would be 0%. In this case, it is evident that the probability of winning $80 is 100% from both 10 episodes and 1000 episodes. Therefore, the probability of winning $80 within 1000 sequential bets is 100%.

Here is an example snippet of the code I used to calculate probability:

```
def caclulateProb1(arry):
    #prob = np.empty(80, dtype=float)
    thisdict = {}
    for i in range(1000):
        index = arry[i]
        thisdict[index] = thisdict.get(index, 0) + 1
    #print(thisdict)
```

Figure 4

```
{80.0: 1000}
```

Figure 5

P(80) = 1000/1000 = 1 = 100%

Note: arry is the 1000th spin for all 1000 episodes e.g. last column of matrix

As seen from the output all 1000 episodes have reached $80 of winnings on their 1000th spin.

1.2: Question 2)

Since the probability of winning $80 after the 1000th spin is 100% (P(X) = 80 -> 100%) the expected value here is 80. Since all other winning amounts are 0 when we calculate the probability of times the winning amount, we will always end up with 0 except for the case of 80. Therefore, when we sum this at the end, we will end up with a result of 80. For example,

E[X] = Sum(x,P(X=x)) where x = number of winnings, E[X] = expected value of winnings, and P(X=x) = probability of winning x number of dollars. Let's take an example of 0 to 100:

E[X] = Sum(x, P(X=x)) from (0 to 100) = 0(0) + 1(0) + 2(0) ... + 80(1) .. + 100(0) = 80

1.3: Question 3)

Here the standard deviation lines are going crazy. There is a lot of variability in each spin across the episodes. There is no apparent stabilization that is occurring for the standard deviations until they begin to converge towards the 80-dollar mark as the number of spins increases. It appears that around spin 200 almost all the episodes have reached 80 with little to no variance.

As the values in the spins across the episodes begin to match the standard deviations converges to 0.

1.4: Question 4)

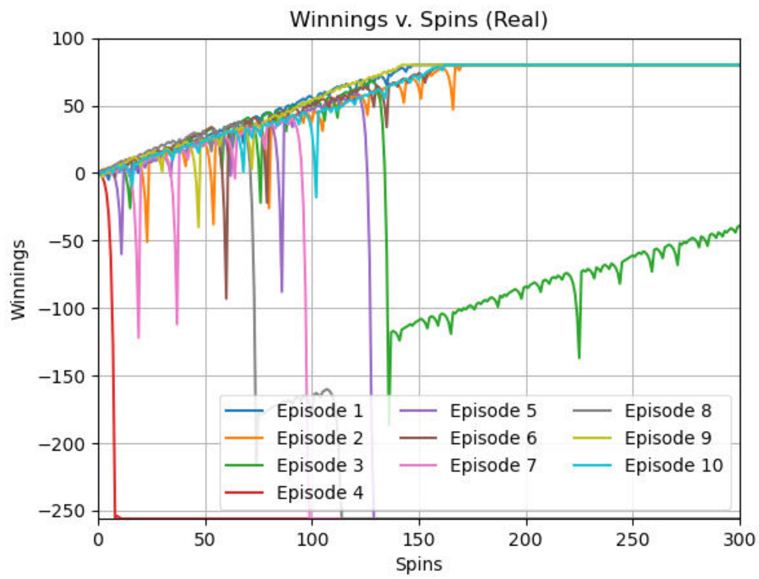

Figure 5

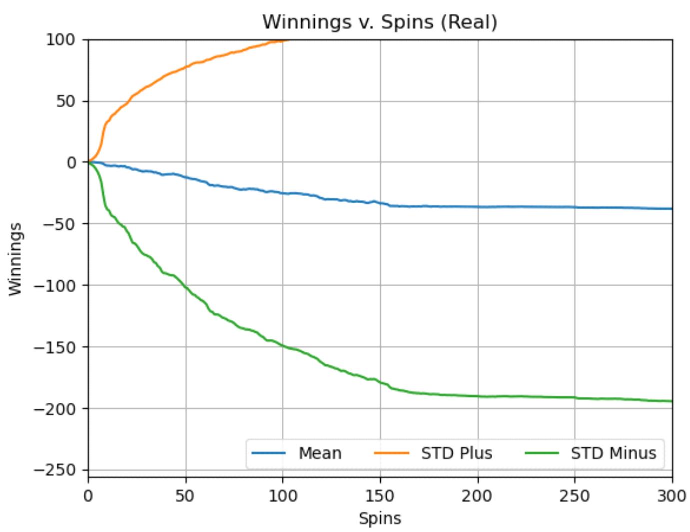

Figure 6

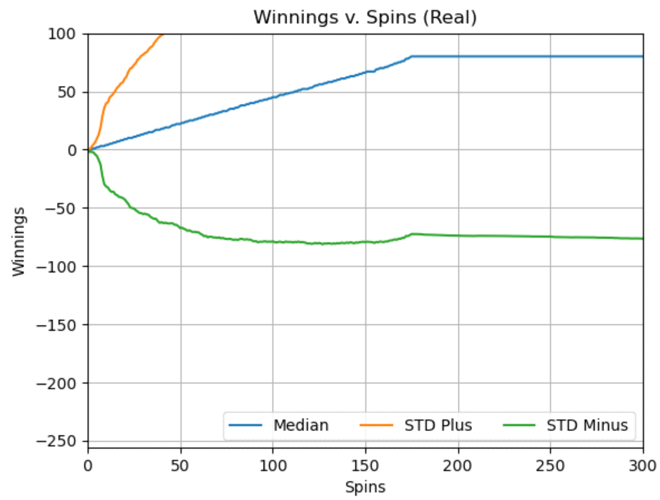

Figure 7

To calculate the probability, I have used the same helper function as above to calculate probability:

```
def caclulateProb1(arry):
    #prob = np.empty(80, dtype=float)
    thisdict = {}
    for i in range(1000):
        index = arry[i]
        thisdict[index] = thisdict.get(index, 0) + 1
    #print(thisdict)
```

Figure 8

Note: arry is the 1000th spin for all 1000 episodes e.g. last column of matrix

This time the result was

```
{80.0: 642, -256.0: 354, 37.0: 1, 52.0: 1, -107.0: 1, -103.0: 1}
```

Figure 9

Therefore, we have:

P(80) = 642/1000 = .642

P(-256) = 354/1000 = .354

P(37) = 1/1000 = .001

P(52) = 1/1000 = .001

P(-107) = 1/1000 = .001

P(-103) = 1/1000 = .001

The probability of achieving another number of winnings besides the ones seen above results in a probability of 0.   P(99) = 0

P(80) = 642/1000 = .642 = 64.2 %

1.5: Question 5)

The expected value can be calculated by multiplying the number of winnings with the probability of that winning number. Hence:

E[X] = Expected value of winnings = sum(x,P(X=x)) = (80)(.642) + (-256)(.354) + (37)(.001) + (52)(.001)+(-107)(.001)+(-103)(.001) = -39.422

Based off this calculation, in a more realistic scenario, the expected value will be a loss of $39.42.

1.6: Question 6)

In this case the standard deviations do reach a max/min and then stabilize. As the number of bets increases the standard deviation lines do not converge with each other but do converge to their own stabilization line. There is not a lot of variability in this realistic situation as opposed to the previous experiment. At around spin 200 the standard deviation lines begin to stabilize. This may be because most of the episodes have either reached $80 or -256 dollars.

1.7: Question 7)

The benefit of using expected values when conducting experiments is a more realistic interpretation of the probability, data, and what is actually happening. With more data collected our expected value becomes more accurate. By only doing one episode our expected value would be the end value of that episode. But after conducting 1000 episodes we can now have a more realistic estimate of the expected value. The expected value is a more realistic average as it gives us a long-term mean over a probability distribution. It gives a way to include the missing piece and provide more accurate predictions.

2 References

https://stackoverflow.com/questions/9622163/save-plot-to-image-file-instead-of-displaying-it-using-matplotlib

https://www.w3schools.com/python/matplotlib_grid.asp

https://www.geeksforgeeks.org/python-range-function/

https://www.tutorialspoint.com/how-to-show-legend-elements-horizontally-in-matplotlib https://numpy.org/doc/stable/reference/generated/numpy.median.html

https://www.tutorialspoint.com/python/dictionary_get.htm

https://www.tutorialspoint.com/python/python_tuples.htm