University of Southern California EE511

Coin Flips Simulation

Project #1

Wu Jiawei
2017-1-17

Abstract

In the project of tossing coin simulation, five experiments are conducted using Matlab. The core method for the project is to simulate the result of tossing coin by generating random numbers and comparing them with the threshold. The experiments are repeated multiple times under variable control conditions to avoid accidental errors. The results prove that the probability for outcome “Head” or “Tail” is nearly 50% for a fair coin.

Introduction

Five experiments were conducted in the lab. The goals of first three trials are simulating tossing a fair or biased coin multiple times to count the number of heads, record the longest run of heads and generate a histogram for Bernoulli outcomes. The aim of forth experiment is to generate a histogram showing the heads run lengths. The objective of the final trial is to count the amount of tosses until reaching a user-specified number of heads.

Methodology & Results

Experiment No.1

Question:

Simulate tossing a fair coin 50 times. Count the number of heads. Record the longest run of heads. Generate a histogram for the Bernoulli outcomes

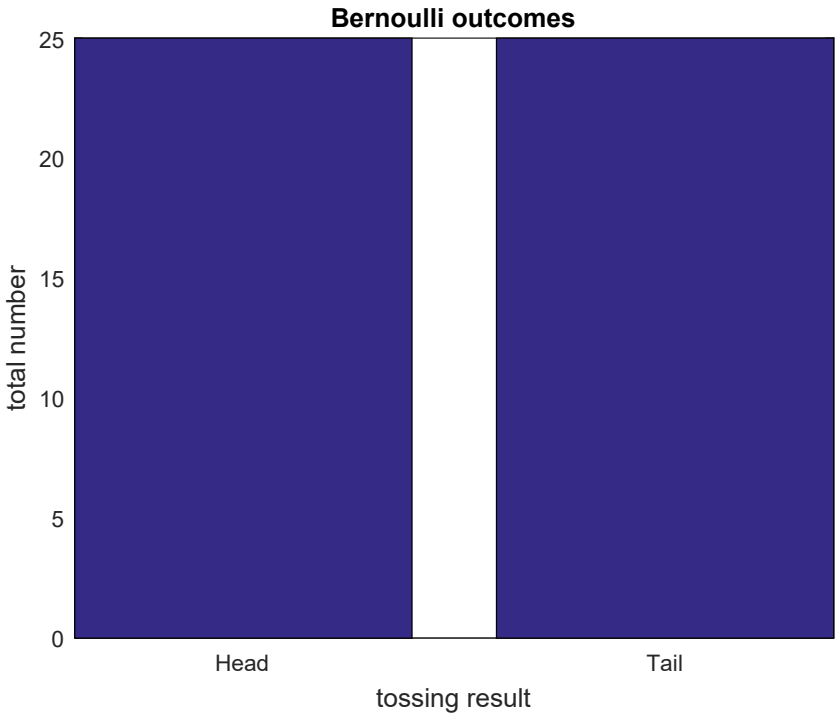
Code:

```
function f=coin(N)%Define the function to count the number of "Heads"
NumH=0;%"NumH" represents the total number of "Head"
NumT=0;%"NumT" represents the total number of "Tail"
NumCH=0;%"NumCH" represents the number of continuous outcomes of "Head"
arr=[];%create an empty array to store variable "NumCH"
j=1;%"j" indicates the index for the empty array
r=rand(1,N);%Generate an 1*N array with random value.
for i=1:N
    if r(i)>0.5%assume the threshold as "0.5" and above 0.5 as "Head"
        NumH=NumH+1;
        NumCH=NumCH+1;
    else
        NumT=NumT+1;
        arr(j)=NumCH; %store amount of “NumCH” into empty array
        NumCH=0;%reset the counter "NumCH" when outcome is "Tail”
        j=j+1;
    end
end
NumH
r=arr;%Count the number of "Head"
[MaxRun,Location]=max(r)%Record the longest run of heads
z=[NumH,NumT];%Create a histogram for the Bernoulli outcomes
bar(z)
title ('Bernoulli outcomes')
xlabel ('tossing result')
```

```
ylabel ('total number')
set(gca,'xticklabel',{'Head','Tail'})
```

Simulation Result:

```
Command Window
>> coin(50)

NumH =

    25


MaxRun =

     5


Location =

     8
```



Comment:

In the first trial, I set the threshold as 0.5 since it is a fair coin. I define several variables to count the total amount of “Head” and “Tail”. I increase the “NumCH” when “Head” appear continuously and reset it when “Tail” occurs. I store the “NumCH” into an empty array and use function “max” to find the max value of the array which means the longest head run length and use the function “bar” to draw the histogram.
In the first experiment, when tossing a fair coin 50 times, the number of head is 25, the longest run of heads is 5, and the histogram shows that the ratio between “Head” and “Tail” is 25:25.

Experiment No.1.a

Question:

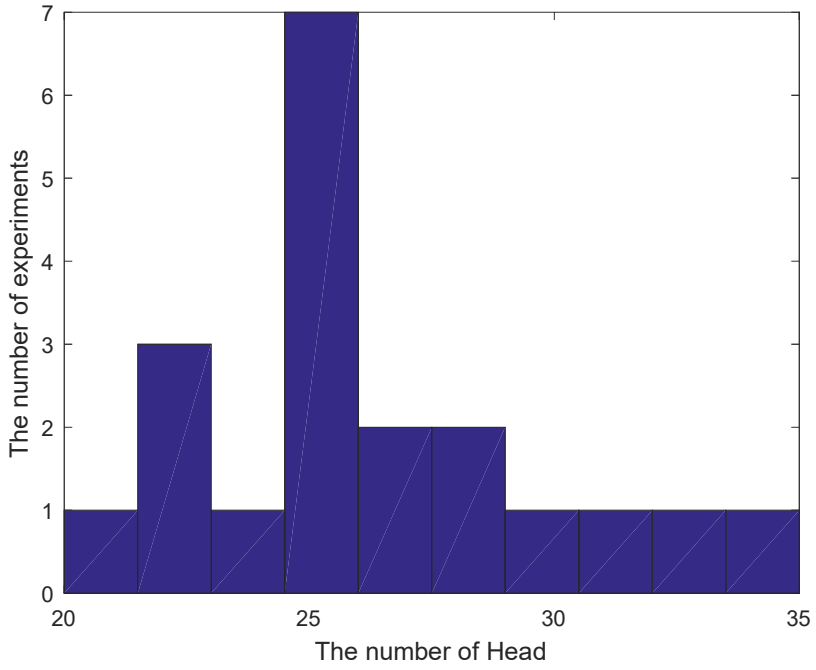
Repeat the above experiment 20, 100, 200 and 1000 times. Generate a histogram for each showing the number of heads in 50 flips. Comment on the limit of the histogram

Code:

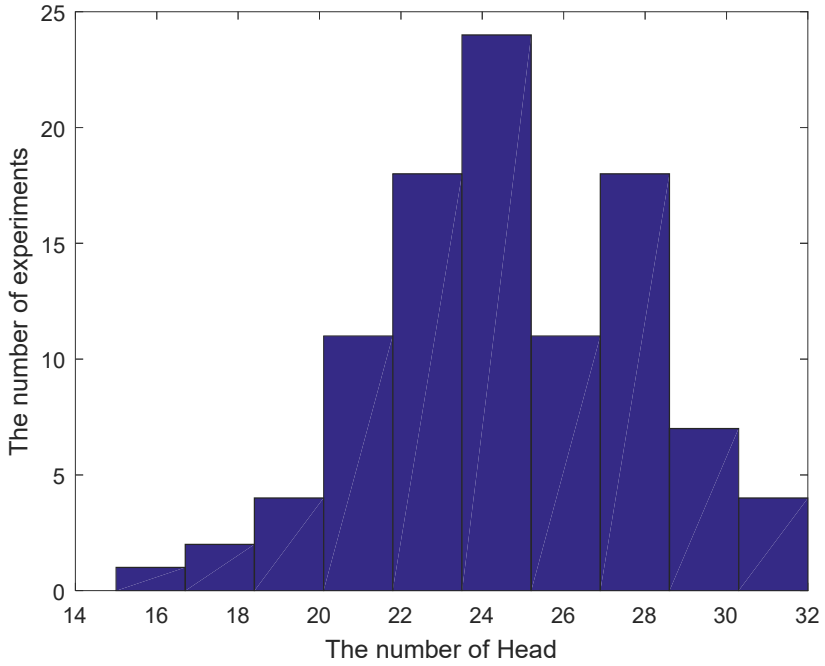
```
function f=count(N) %Define the function to count the number of heads
EachNumH=0;% "EachNumH" represents the number of "Head" in each 50 flips
j=1;% "j" indicates the index of the empty array
arr=[];%create a new empty array to store variable "EachNumH"
for i=1:N
for k=1:50%Loop the program N*50 times which simulate tossing a coin N*50
times
    r=rand(1,50)
    if r(k)>0.5%Define the threshold as "0.5" since it is a fair coin
        EachNumH=EachNumH+1;
    end
end
arr(j)=EachNumH; %store the amount of "Head" in 50 flips into an empty array
EachNumH=0;%reset the variable "EachNumH" when loop 50 times
j=j+1;
end
q=arr;
hist(q)%Generate a histogram to show the number of heads in 50 flips
xlabel ('The number of Head')
ylabel ('The number of experiments')
```

Simulation Results:

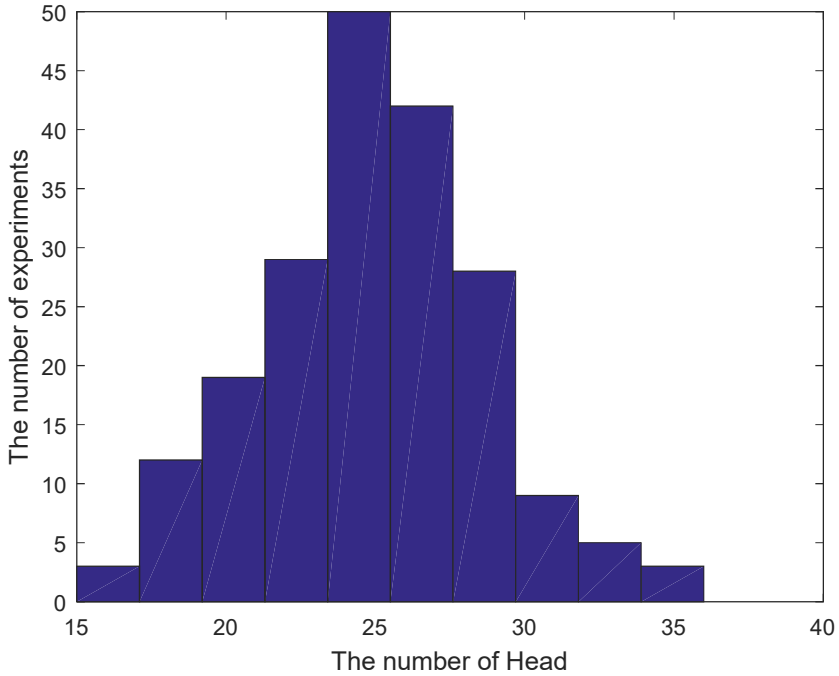
1) N=20



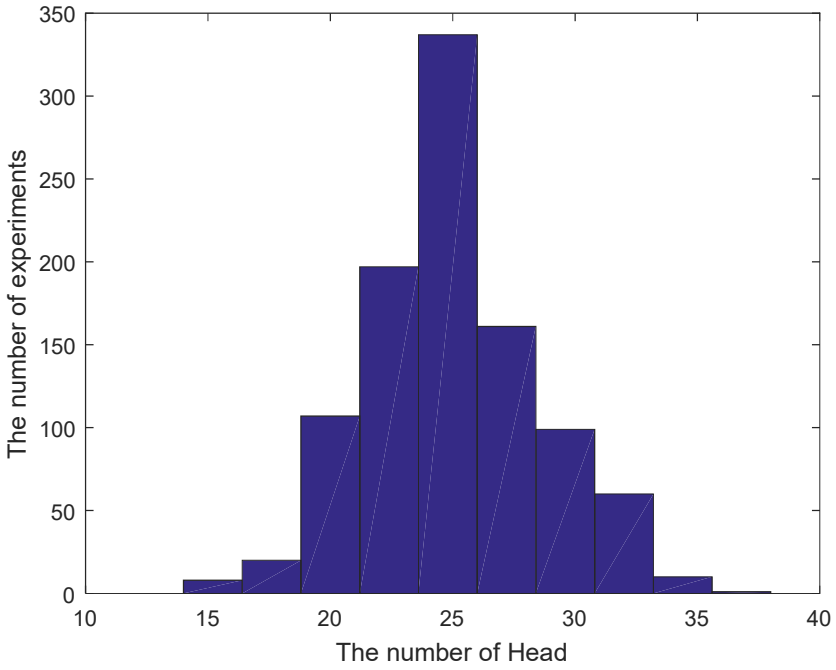
2) N=100


The number of experiments
The number of Head
0
5
10
15
20
25
14
16
18
20
22
24
26
28
30
32

3) N=200


The number of experiments
The number of Head
0
5
10
15
20
25
30
35
40
45
50
15
20
25
30
35
40

4) N=1000



Comment:

In the experiment, I define the variable “EachNumH” to record the amount of heads in 50 flips and use “for loop” multiple times to count the number of “Head” in each 50 flips. I use the function “hist” to draw a histogram to show the simulation result.

When increasing the tossing times to 1000, 5000 and 50000, the simulation histogram become increasingly obeying to the normal distribution(Gaussian distribution). The general law can been seen from the histogram when tossing 50000 times, the majority amount of the “Head” is 25, few results are between 20-25 or 25-30, and even small results are between 15-20 or 30-35, no result are below 15 or beyond 38. Thus, the probability for outcome “Head” is obviously close to 50% after repeating the experiment a large amount of times. This is because multiple repeating could avoid accidental factors and reflect general law.

Experiment No.2

Question:

Simulate tossing a biased coin 200 times where P[HEAD] = 0.80. Count the number of heads. Record the longest run of heads. Generate a histogram for the Bernoulli outcomes.

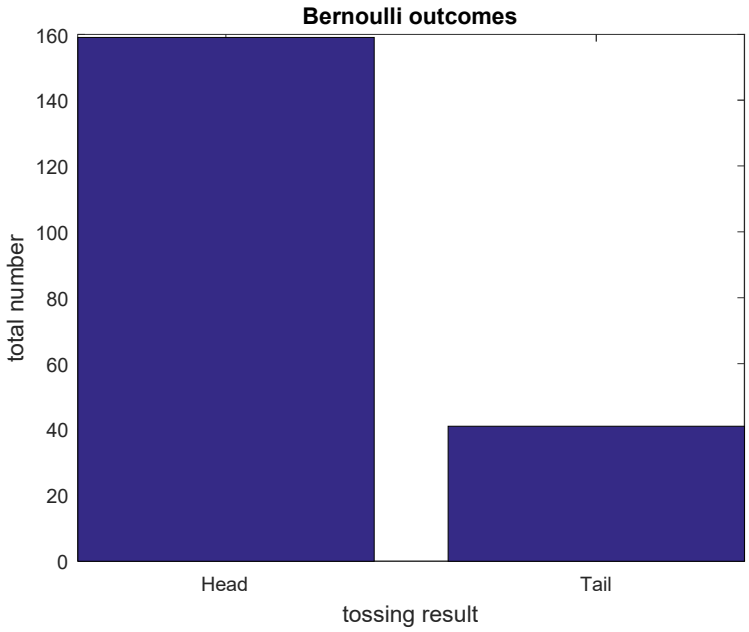
Code:

```
function f=coin(N)%Define the function to count the number of "Heads"
NumH=0;%"NumH" represents the total number of "Head"
NumT=0;%"NumT" represents the total number of "Tail"
NumCH=0;%"NumCH" represents the number of continuous outcome of "Head"
arr=[];%create an empty array to store variable "NumCH"
j=1;%"j" indicates the index for the empty array
```

```
r=rand(1,N);%Generate an 1*N array with random value.
for i=1:N
    if r(i)<0.8%assume the threshold as "0.8" and below 0.8 as "Head"
        NumH=NumH+1;
        NumCH=NumCH+1;
    else
        NumT=NumT+1;
        arr(j)=NumCH; %store the amount of "NumCH" into an empty array
        NumCH=0;%reset "NumCH" when "Tail" appears
        j=j+1;
    end
end
NumH
r=arr;%Count the number of "Head"
[MaxRun,Location]=max(r)%Record the longest run of heads
z=[NumH,NumT];%Create a histogram for the Bernoulli outcomes
bar(z)
title ('Bernoulli outcomes')
xlabel ('tossing result')
ylabel ('total number')
set(gca,'xticklabel',{'Head','Tail'})
```

Simulation Result:

```
Command Window
>> coin(200)

NumH =

   159


MaxRun =

    14


Location =

    19
```



Comment:

In the second experiment, I hypothesis that the ratio between “Head” and “Tail” is 0.8 : 0.2 since it is a biased coin where P[HEAD] = 0.80. From the histogram for the Bernoulli outcomes, The number of “Head” is 159 and the amount of “Tail” is 41 and the ratio between them is approximately 0.8:0.2 which proves the hypothesis.

Experiment No.3

Question:

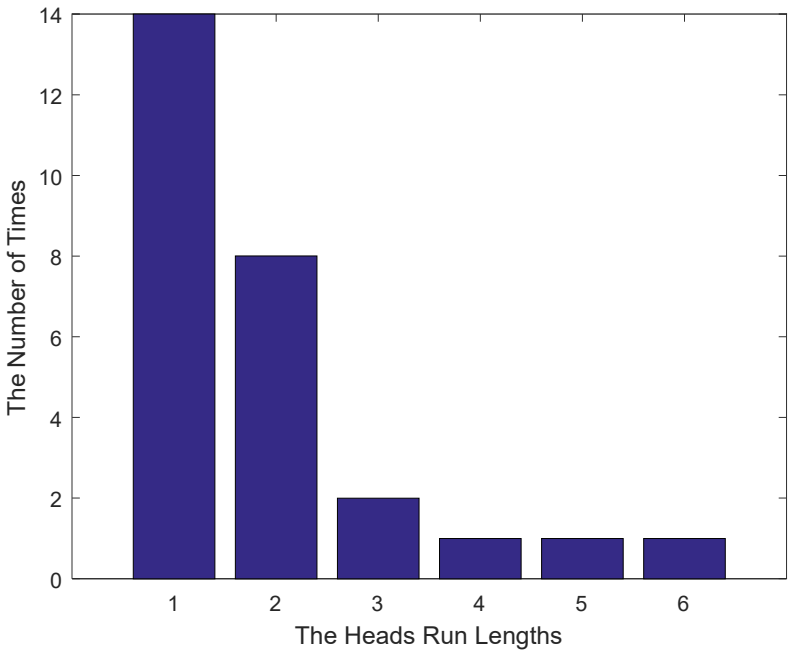
Simulate tossing a fair coin 100 times. Generate a histogram showing the heads run lengths.

Code:

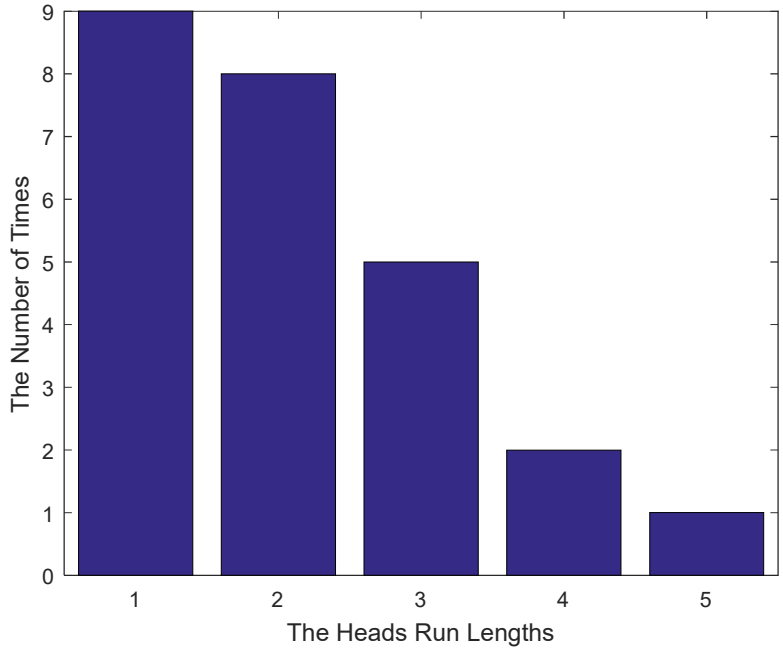
```
function f=coin(N)%Define the function the count the Head run lengths
data = zeros(1,N);%Create a zero array to store different head run length
random = rand(1,N);
NumCH = 0;%"NumCH" represents the number of continuous outcome of "Head"
for i=1:N
    if random(i)>=0.5
        NumCH = NumCH+1;
    elseif random(i)<0.5
        if NumCH~=0
            data(NumCH) = data(NumCH) + 1
        end;%Not showing the zero at the beginning of array
        NumCH = 0;
    end
end
for i = 1:100
    if sum(data(i:100))==0
        data = data(1:i-1);%Not showing the zero at the end of array
        break
    end
end
bar(data)
xlabel ('The Heads Run Lengths')
ylabel ('The Number of Times')
```

Simulation Result:

>> draw(100)



>> draw(100)



Comment:

In the third experiment, I add one to the variable “NumCH” once “Head” appears and reset it when the outcome of tossing is “Tail”. I store the number of head run lengths into an zero array. I use two “for loop” to avoid drawing zeros at the start and end of the array. Two histograms are created to show the heads run lengths.

It can be seen from two simulation pictures that the majority number of heads run lengths is 1, few number of heads run lengths is 2 and 3, and even small number of heads run lengths “Head” is 4 and 5, little or no amount of heads run lengths is more than 6.

Experiment No.4

Question:

Simulate tossing a fair coin and count the number of tosses until reaching a user-specified positive number of heads

Code:

```
function f=count_1(N)%Define a function to count the total number of tossing
x=0;
i=0;%"i" indicates the amount of tossing
while 1
    i=i+1;
    r=rand();
    if (r>0.5)&(x~=N)%Define outcome as "Head" if the random number is above
0.5 and the total tossing times is less than N
        x=x+1;
    end
    if x==N%Exit loop when the required number of "heads" are completed
        break
    end
end

D=i
```

Simulation Results:

Comment：

In the fourth experiment, I use a “while loop” to simulate tossing a coin multiple times and exit loop using “break” when the user-specified amount of “Head” are completed. I set the threshold as “0.5” since it is a fair coin and define the tossing outcome as “Head” when the random value is more than 0.5 and the amount of “Head” is below the user-specified number.

In the experiment, I assume that the user-specified positive number of heads are 500 and 1000. The simulation outcomes show that the total number of tossing is 951 and 1988 respectively which is approximately two times of the demand “Head” amount. The outcome also proves the law that when tossing a fair coin, the ratio of number between “Head” and “Tail” is approximately 0.5 : 0.5 and the probability for each is closely to 50%.

Conclusion

Overall, the simulation results of four experiments prove the theory that with a large number of experiments, the probability for outcome “Head” or “Tail” is nearly equal to 50% when tossing a fair coin, and the ratio between the number of “Head” and “Tail” are 0.8 : 0.2 when tossing a biased coin(P[HEAD] = 0.80).