Biologically Motivated Problems in Statistical Physics
Homework 2
Haydar Altuğ Yıldırım
090100252

```
#this is a call for to simulate coin tosses via a random number generator
from random import randint
#calling histogram functions
from pylab import figure, hist, show, subplot, xlabel, ylabel, title, grid, plot
#call for the binomial distribution.
from numpy.random import binomial

#define a function in order to simulate tossing action of coins for N
#times over M series.
def toss(N,M): #N and M values will be given to this function.
    heads=[] #creating a zero-array for the coins which will give the
    #result of heads.
    for i in range(M-1): #the procedure will be repeated M times inside
    #this loop
        count=0
        for j in range(N-1): #tossing the coins N times will occur inside
        #this loop
            coin=randint(0,1) #this is the point where coin tossing
            #occurs
            if coin==1: #Counting of the trials which gives the result of
            #heads
                count+=1
        heads.append(count) #this is speacial function(append) inside the
        #python which gives opportunity to add discrete values to
        #different elements of an array(in here named heads)
    return heads #returning the heads array.


#I think there is also another way of doing this which is not using the
#pre-determined function of histogram as we will use in a minute. This
#can be done with making a loop that is capable of doing a counting of
#the same elements inside heads array. This function is written to
#accomplish it but I think I can not make it work.

def manual(N,heads):
    x=0
    n=[]
    lenheads=len(heads)
    for i in range(N+1):
        for i in range(lenheads):
            if M[j] == i:
                x=x+1
        n.append(x)
```

```
    xlabel('# of Heads')
    ylabel('# of Trials')
    plot(heads,n)
    title('Heads vs Trials')
    show()

N=input('Enter the number of coins ') #user input for N.
M=input('Enter the number of trials ') #user input for M

heads=toss(N,M) #callin the function over the user given values.
binom=binomial(N, 0.5, M) #calling the specific binomial function for the
#second part of the HW.

#graphing the histogram:
figure(1)

subplot(211)
hist(heads, normed=1) #calling the heads array inside the histogram func.
#properties of the first histogram of coin tossing, simulated by random
#number generator.
title('Simulated Coin Toss', color='red')
xlabel('# of Heads', fontsize=10)
ylabel('# of Trials', fontsize=10)
grid(True)

subplot(212)
hist(binom, normed=1) #calling the heads array inside the binomial func.
#properties of the second histogram which is the binomial distribution.
title('Binomial Distribution', color='red')
xlabel('# of Heads', fontsize=10)
ylabel('# of Trials', fontsize=10)
grid(True)

show()
```

Graphs:

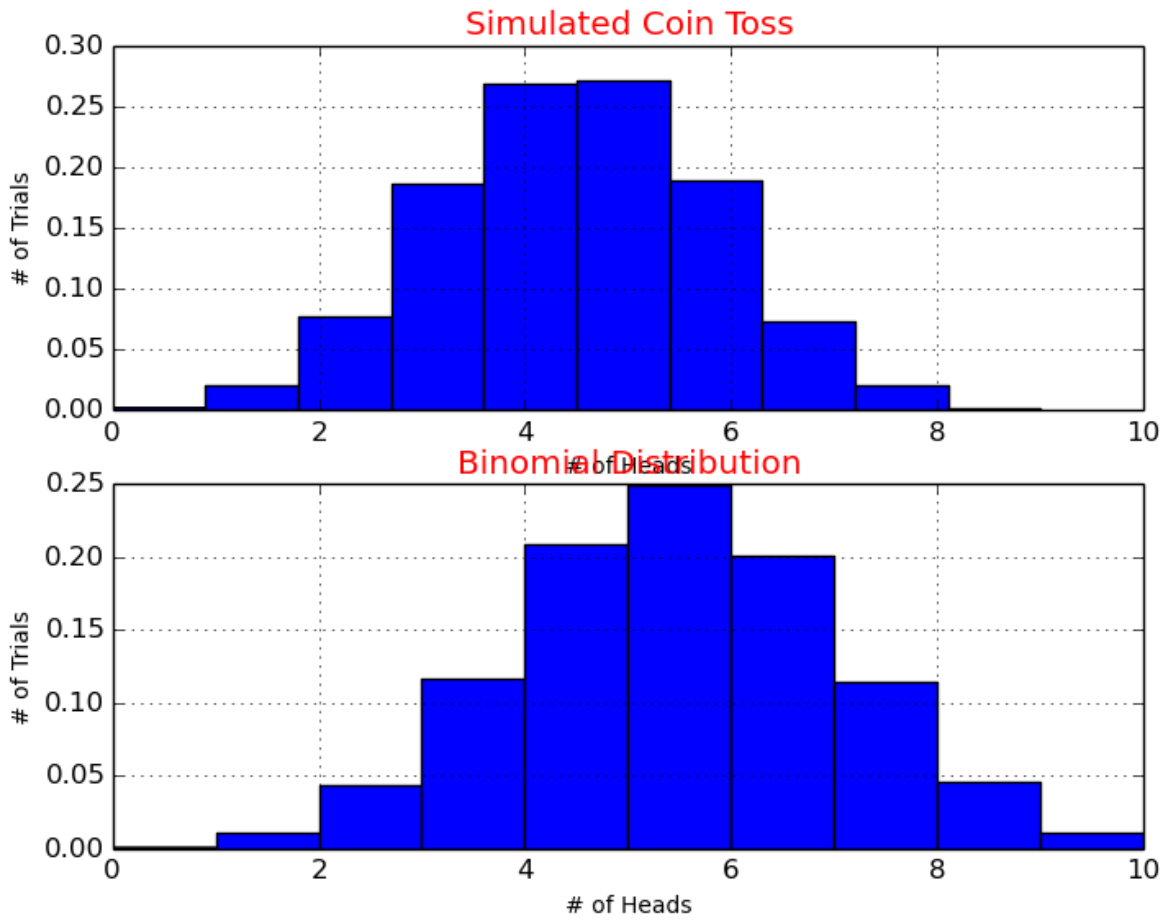

figure 1. N=10, M=10000

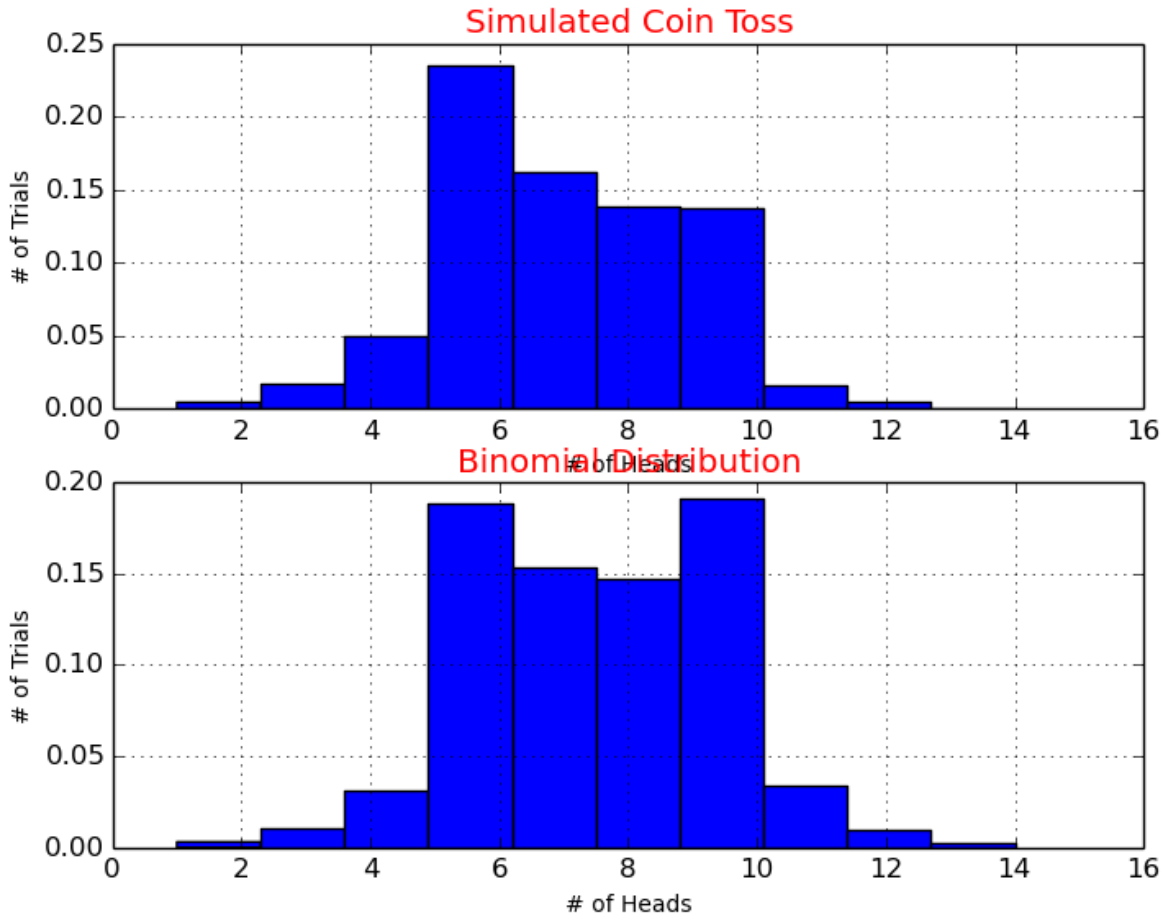

figure 2. N=15, M=10000

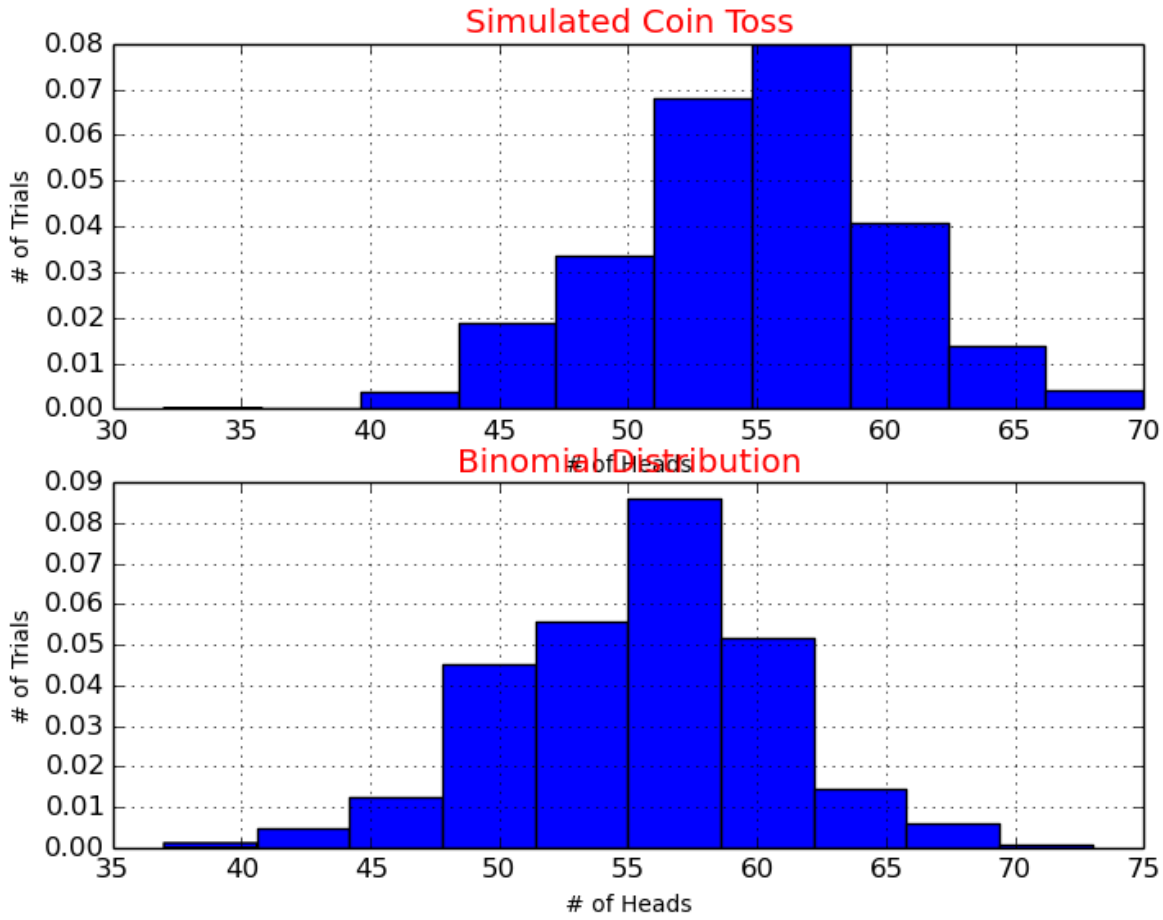

figure 3. N=10, M=1000