

CALIFORNIA STATE UNIVERSITY, LONG BEACH
EE 381 – Probability and Statistics with Applications to
Computing
Laboratory Projects

Project on
Random Numbers and Stochastic Experiments

0. Introduction and Background Material

0.1. Simulating Coin Toss Experiments

As mentioned in class, there are many ways to model stochastic experiments. The following two programs simulate the toss of a fair coin N times, and calculate the experimental probability of getting heads (p_{heads}) or tails (p_{tails}). Both programs provide the same results, but they differ in the way the models are coded.

- The first model is programmed in Python using "for loops".
- The second model makes use of the arrays, and it is computationally very efficient.

MODEL 1

```
import numpy as np
def coin(N):
    heads, tails = 0, 0
    for k in range(0,N):
        coin=randint(0,2)
        if coin==1:
            heads=heads+1
        else:
            tails=tails+1
    #
    p_heads=heads/N
    p_tails=tails/N
    print('probability of heads = ', p_heads)
    print('probability of tails = ', p_tails)
```

MODEL 2 – MORE EFFICIENT CODE

```
import numpy as np
def coin2(N):
    coin=randint(0,2,N)
    heads=sum(coin)
    tails=N-heads
    #
    p_heads=heads/N
    p_tails=tails/N
    print('probability of heads = ', p_heads)
    print('probability of tails = ', p_tails)
```

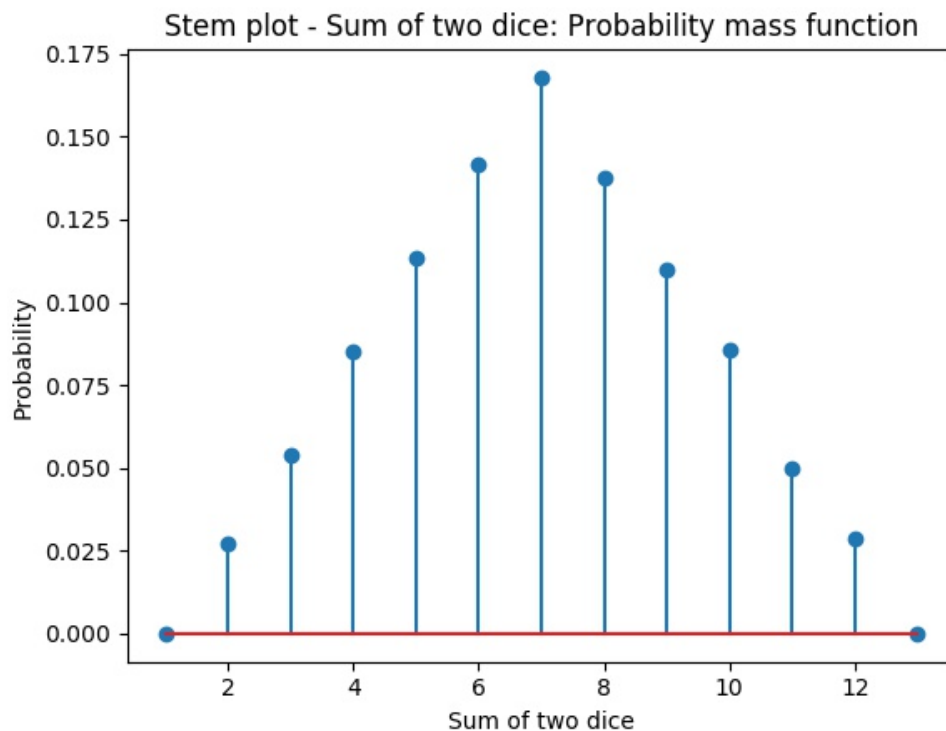
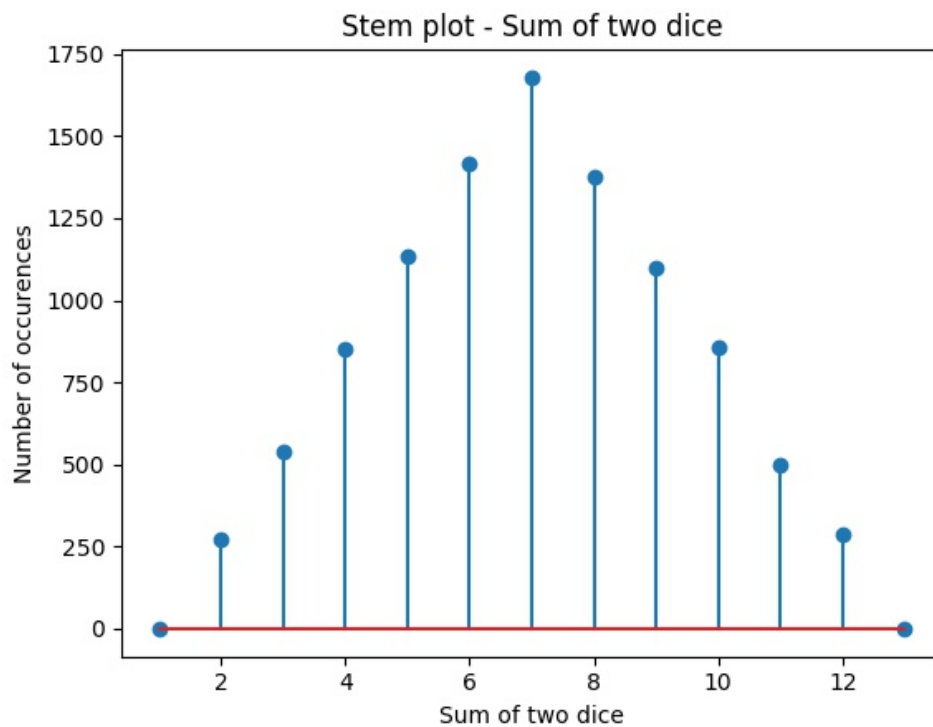
0.2. Roll of Two Fair Dice; Probability Mass Function (PMF)

This experiment models the roll of a pair of dice for N times. The sum each roll is recorded, and stored in vector "s". The probability of each possible outcome is calculated and plotted in a "Probability Mass Function" (PMF) plot. To create the plots, the simulation has been run for N=100000 times.

SUM OF THE ROLLS OF TWO FAIR DICE

```
def sum2dice(N):
    d1=randint(1,7,N)
    d2=randint(1,7,N)
    s=d1+d2
    b=range(1,15)
    h1, bin_edges = histogram(s,bins=b)
    b1=bin_edges[0:13]
    close('all')
    #
    fig1=plt.figure(1)
    plt.stem(b1,h1)
    plt.title('Stem plot - Sum of two dice')
    plt.xlabel('Sum of two dice')
    plt.ylabel('Number of occurrences')

    #
    fig2=plt.figure(2)
    p1=h1/N
    plt.stem(b1,p1)
    plt.title('Stem plot - Sum of two dice: Probability mass function')
    plt.xlabel('Sum of two dice')
    plt.ylabel('Probability')
```



1. Number of rolls needed to get a "7" with two dice

Consider the following experiment:

- You roll a pair of fair dice and calculate the sum of the faces. You are interested in the number of rolls it takes until you get a sum of "7". The first time you get a "7" the experiment is considered a "success". You record the number of rolls and you stop the experiment.
- You repeat the experiment $N=100,000$ times. Each time you keep track of the number of rolls it takes to have "success".
- After the N experiments are completed you should generate a **probability mass function** for the number of rolls it takes for "success".
- **SUBMIT** the PMF plot and the computer code in a Word file.

2. Getting 50 heads when tossing 100 coins

Consider the following experiment:

- You toss 100 fair coins and record the number of "heads". This is considered a single experiment. If you get exactly 50 heads, the experiment is considered a "success".
- You repeat the experiment $N=100,000$ times. After the N experiments are completed count the total successes, and calculate the probability of success, i.e. the probability of getting exactly 50 heads.
- **SUBMIT** the calculated answer and the code in a Word file. *Use the table below for your answer.* Note: You will need to replicate the table in your Word file, in order to provide the answer in your report. Points will be taken off if you do not use the table.

Probability of 50 heads in tossing 100 fair coins	
Ans.	$p =$

3. Getting 4-of-a-kind

Consider the following experiment:

- You draw 5 cards from a deck of 52 cards. This is considered a single experiment. If you get 4-of-a-kind the experiment is considered a "success".
- You repeat the experiment $N=100,000$ times, keeping track of the "successes".
- After the N experiments are completed count the total successes, and calculate the probability of getting 4-of-a-kind.
- **SUBMIT** the calculated answer and the code in a Word file. *Use the table below for your answer.* Note: You will need to replicate the table in your Word file, in order to provide the answer in your report. Points will be taken off if you do not use the table.

Probability of 4-of-a-kind	
Ans.	$p =$

4. The Password Hacking Problem

Your computer system uses a 4-letter password for login. For our purposes the password is restricted to lower case letters of the alphabet only. It is easy to calculate that the total number of passwords which can be produced is $n = 26^4$.

- A hacker creates a list of $m = 10^5$ random 4-letter words, as candidates for matching the password. Note that it is possible that some of the $m = 10^5$ words may be duplicates.
- You are given your own 4-letter password and you are going to check if the hacker's list contains at least one word that matches your password. This process of checking is considered one experiment. If a word in the list matches your password, the experiment is considered a success. Repeat the experiment for $N = 1000$ times and find the probability that **at least one of the words** in the hacker's list will match your password.
- If the hacker creates a list of $m = 10^6$ random 4-letter words, repeat the previous experiment for $N = 1000$ times and find the probability that **at least one of the words** in the hacker's list will match your password.
- Repeat the previous experiment for $N = 1000$ times to find the **approximate number** (m) of words that must be contained in the hacker's list so that the probability of at least one word matching the password is $p = 0.5$. You should do this by trial and error: assume a value for (m) and calculate the corresponding probability as you did in the previous part. The answer will be value of (m) that makes this probability **approximately equal** to $p = 0.5$.
- **SUBMIT** your answers and the code in a Word file. *Use the table below for your answers.* Note: You will need to replicate the table in your Word file, in order to provide the answers in your report. Points will be taken off if you do not use the table.

$m = 10^5$ Prob. that at least one of the words matches the password	$p =$
$m = 10^6$ Prob. that at least one of the words matches the password	$p =$
$p = 0.5$ Approximate number of words in the list	$m =$

5. References

- [1] "Introduction to Probability," by H. Pishro-Nik. Available online at: <https://www.probabilitycourse.com>