Python: problem solving and simulations

Second tutorial session

Python lab sessions: some tips

Lab sessions

1. Login to student account
2. Login to internet.iitb.ac.in using LDAP id, password and authenticator password (or, captcha)
3. Login to moodle.iitb.ac.in using LDAP id and password
4. The question paper and answer submission will be via moodle
5. You are allowed to use any material on moodle; you are also allowed to use help in python terminal
6. Please do not use your phones, internet or any other tool for solving problems; take help only from TAs; even though you are allowed to discuss with your neighbours, that should not be the primary mode.

Writing scripts and running them

1. Make a directory on Desktop with your roll number
2. Save everything to the directory you have created and delete the directory at the end of the lab
3. Use a text editor (such as notepad or gedit) or command (such as vim, emacs, nano) for writing the scripts; save them with .py extension
4. Invoke python from terminal in the same directory in which you have saved the script
5. python3 *filename*.py is the command for running the script

Recall

1. You are expected to know how to import modules and libraries:

   ```
   import numpy as np
   import math
   import matplotlib.pyplot as plt
   from numpy import linalg
   ```

2. You are expected to call library functions with appropriate input parameters, or, use help to figure out the syntax and parameters

Python: problem solving

Problem 1

1. There are eight pairs of shoes; we randomly pick four shoes. Let us find out in how many ways can we pick four shoes.

Picking shoes

```
import math as m
print(m.comb(16,4))

a =
['1a','1b','2a','2b','3a','3b','4a','4b','5a','5b','6a','6b','7a','7b','8a','8b']

from itertools import combinations
x  = combinations(a,4)
for i in list(x):
    print(i)
```

Problem 2

1. There are eight pairs of shoes; we randomly pick four shoes. Let us find out in how many of the picks result in exactly one correct pair chosen.

Counting cases with exactly one correct pair of shoes

```
count = 0
for i in list(x):
    if( (i[0][0] == i[1][0]) and (i[2][0] != i[3][0])):
        count = count + 1
    elif(i[1][0] == i[2][0]):
        count = count + 1
    elif( (i[0][0] != i[1][0]) and (i[2][0] == i[3][0])):
        count = count + 1
```

Counting cases with exactly one pair of shoes

```
import math as m
N = m.comb(16,4)

probability = count/N

print(probability)
```

Python: simulation

Problem 3

There are eight pairs of shoes; we randomly pick four shoes. Let us simulate the process of such random picks and see if we can gain an idea of the probability from the simulation.

Note that there are two ways to simulate; randomly choose from the 16 shoes 4 each time and keep counting the number of times we get only one correct pair. The other way is to use the list we have generated of all possible combinations and choose randomly from the list. We follow the second strategy here!

Simulating the shoe choosing process

```
import random
N = 400
count = 0
for i in range(N):
#   r = x[random.randint(0,1820)] // You can uncomment this and
comment next # line; it works both ways!
    r = random.choice(x)
        if( (r[0][0] == r[1][0]) and (r[2][0] != r[3][0])):
            count = count + 1
```

Simulating the shoe choosing process

```
        elif(r[1][0] == r[2][0]):
            count = count + 1
        elif( (r[0][0] != r[1][0]) and (r[2][0] == r[3][0])):
            count = count + 1

probability = count/N;

print(probability)
```

Simulations

Notice: every time you simulate, you get a different number.

Why?

Like tossing coins. If you toss 10 times, in different experiments, you will get different number of heads each time! That is the nature of the experiment!

What happens if we change N? Does the value converge to a specific value?

Simulation: effect of N

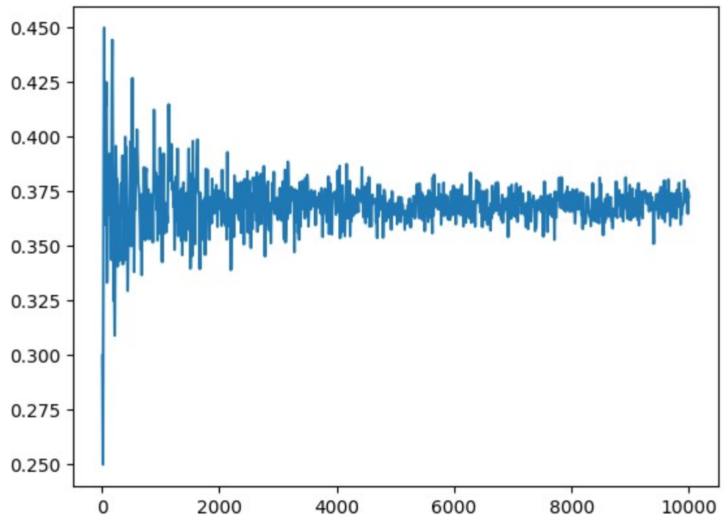
```
M = 1000
X = []
Y = []
N = 10
for j in range(M):
    X.append(N)
    count = 0
        for i in range(N):
        r = random.choice(x)
```

Simulation: effect of N

```
    if( (r[0][0] == r[1][0]) and (r[2][0] != r[3][0])):
        count = count + 1
  elif(r[1][0] == r[2][0]):
        count = count + 1
  elif( (r[0][0] != r[1][0]) and (r[2][0] == r[3][0])):
        count = count + 1
probability = count/N
Y.append(probability)

N = N+10
```

Simulation: effect of N

```
import matplotlib.pyplot as plt

plt.plot(X,Y)

plt.show()
```



Summary

Simulation result: converges to that obtained by explicit counting

Look at all the combinations: can we now solve the problem 2(b) using pen and paper?

random: a library that is useful for simulations

Notice: Sampling can be with and without replacement!

Thank you!!

ALL THE BEST!