**Homework 1: The role of probability in machine learning due on 03/02/2023**

Note: in python, you will need to include a line to include a library called “random”.

import random

This library is automatically included when you import python. You can download python (version like 3.X ) from https://www.python.org/

Please study the note on the sample python code and modify it to answer the question. Please print out your result and submit it in class.

TA email for emergency: 蕭方凱 zeus950068@gmail.com

**HW1 problem**

1.(35%) Use Monte Carlo method to calculate

$$\int_0^1 x^2\,dx$$

You need to do this by counting the random event hitting the area under the curve $y= x^2$ for x and y lying between 0 and 1. Using sample number $N= 10^6$

2. (35%) Use Monte Carlo method to calculate volume of an octant of a sphere of radius one (八分之一球).

In other world, this volume is bounded by

$0<x<1$ $0<y<1$ $0<z<1$

$x^2 +y^2 +z^2 <1$

You can do this by adding one extra variable z to the python code.

3. (30%) Adding noise to a quadratic curve (second order polynomial)

For a quadratic regression model described by

$y= w_2 x^2+ w_1 x+w_0 + \varepsilon$

$\varepsilon$ is the Gaussian noise with mean $\mu$=0 and standard deviation $\sigma$= 0.1. w2=2.0 w1=0.5 and w0= 0.3

Print out 30 data point and plot it in excel. Fit it both with a line and a second order polynomial in excel. (each 15%)

Note: A Gaussian distribution with mean $\mu$ and standard deviation $\sigma$

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

In python, the code for a Gaussian noise with mean and standard deviation is given by

random.gauss(mean, standard deviation)

Note on library usage and python download: in python, you will need to include a line to include a library called “random”. This library is automatically included when you import python. You can download python (version like 3.X ) from https://www.python.org/

```
import random
```

**Monte Carlo method to calculate pi**

First of all, we need to realize probability can be used a rule for computation. A notable example is calculation of pi. We can simply generate a random position (x, y) from a uniform distribution from 0 to 1, i.e., x~ U(0,1) and y~U(0,1)

In python code,

```
x= random.uniform(0,1)
y=random.uniform(0,1)
```

In a unit square defined by

$0<x< 1$ $0<y< 1$

x and y are the coordinates in two dimension space.

We can carry out N experiments and count the number of experiments when the position (x, y) fall into the circle, in other words

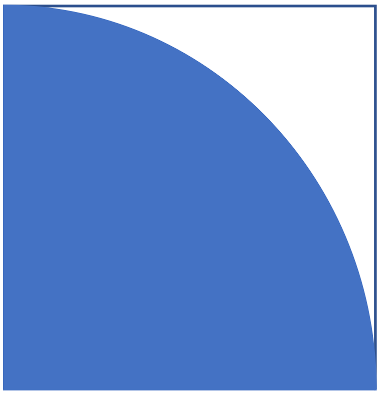
$x^2 +y^2 <= 1$

In python code,

```
(x*x+y*y)**0.5 <=1
```

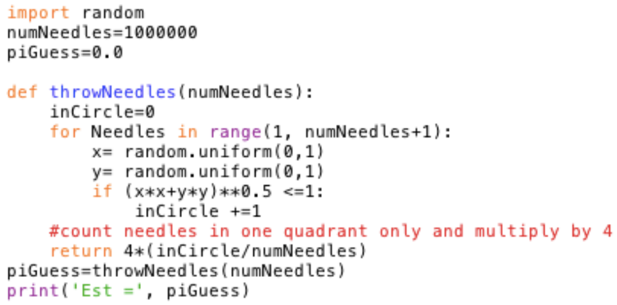
Then the area of a quarter of circle is given by

$$\frac{\pi}{4} = \frac{\#\ of\ sample\ inside\ the\ circle}{\#\ of\ experiments}$$



You may need N~ $10^6$ to get some descent estimation of pi. This may sound stupid at first glance when you realize 1 million experiments is needed. Nowsday, computer is fast so you do not need to way for a long time for the result to come out.

The code is shown below



```
import random
numNeedles=1000000
piGuess=0.0

def throwNeedles(numNeedles):
    inCircle=0
    for Needles in range(1, numNeedles+1):
        x= random.uniform(0,1)
        y= random.uniform(0,1)
        if (x*x+y*y)**0.5 <=1:
            inCircle +=1
    #count needles in one quadrant only and multiply by 4
    return 4*(inCircle/numNeedles)
piGuess=throwNeedles(numNeedles)
print('Est =', piGuess)
```

We first define a function called throwNeedles and set numNeedles to 1000000, which is the sample number N. The function returns the count for needles inside the circle divided by the total number of needles. In the end, we print the result.

**Generating a data set with added Gaussian noise**

We would like to show that python can be used to generate synthetic data point. For a linear regression model described by

$y= w_1 x+w_0 + \varepsilon$

$\varepsilon$ is the Gaussian noise added to the target function, which is a line in this case. We pick x from uniform distribution from interval [0,1] and set Gaussian noise with mean $\mu$=0 and standard deviation sigma $\sigma$=0.5. A Gaussian distribution with mean $\mu$ and standard deviation $\sigma$ is given by this formula

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

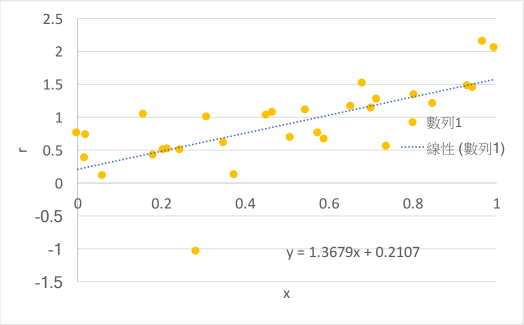
We use w1= 1.3 and w0=0.2.

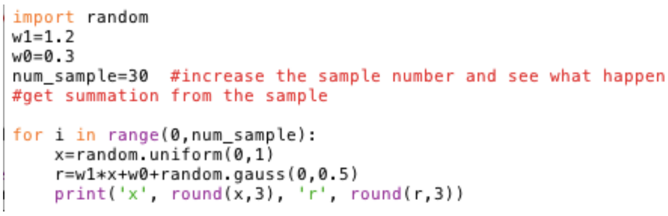
In python code

x=random.uniform(0,1)

r=w1*x+w0+random.gauss(0,0.5)

in the end, we can print out 30 data point and plot it in excel. We fit it with a trend line and obtain y= 1.3679x+0.2107. This is very close to w1=1.3 and w0=0.2 used in the code.

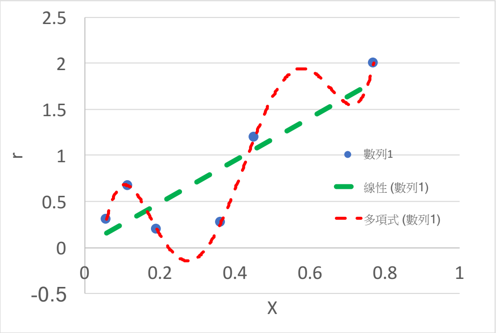




```
import random
w1=1.2
w0=0.3
num_sample=30  #increase the sample number and see what happen
#get summation from the sample

for i in range(0,num_sample):
    x=random.uniform(0,1)
    r=w1*x+w0+random.gauss(0,0.5)
    print('x', round(x,3), 'r', round(r,3))
```

Now we set num_sample to be 6 and plot the data point. This time we fit it with a line and with a 6th order polynomial using built in fitting function in excel.


r
X
數列1
線性 (數列1)
多項式 (數列1)