Course > Week 2 > Homework 2 > Homework 2

Homework 2

Homework 2-1

0.0/1.0 point (graded)

The binomial distribution $P(n) = \frac{M!}{n!(M-n)!} p^n (1-p)^{(M-n)}$ becomes equivalent to the Gaussian distribution $P(x) = \frac{1}{\boxed{(1)}} \exp\left[-\frac{\boxed{(2)}}{2Mp(1-p)}\right]$ in the limit when n and M are much greater than one. Choose the correct formula for $\boxed{(1)}$ and $\boxed{(2)}$ from the following combinations.

- $\boxed{(1)} = Mp(1-p)$, $\boxed{(2)} = (x - Mp(1-p))$
- $\boxed{(1)} = 2\pi Mp(1-p)$, $\boxed{(2)} = (x - Mp)^2$
- $\boxed{(1)} = \sqrt{2\pi Mp(1-p)}$, $\boxed{(2)} = (x - Mp)^2$
- $\boxed{(1)} = 2\pi Mp(1-p)$, $\boxed{(2)} = (x - Mp(1-p))^2$
- $\boxed{(1)} = \sqrt{2\pi Mp(1-p)}$, $\boxed{(2)} = (x - Mp(1-p))^2$

Submit

You have used 0 of 2 attempts

Homework 2-2

0.0/1.0 point (graded)

Consider the following pseudo random number generator, which generates a new "random" number X_{i+1}, from the previous number X_i, as

$$X_{i+1} = \mathrm{mod}\,(aX_i + b, M)\,,$$

with 'mod' the modulo operation, and a and M some constants (which should be very large numbers).

This generator will give numbers within the range (0,M-1). To obtain random numbers within the unit interval $\zeta_i \in [0, 1]$, we set

$$\zeta_i = X_i/M.$$

Which of the following codes correctly implement this random number generator to compute 'ntotal' numbers ζ_i with an initial seed X_0?

```
# C11
import numpy as np
def rng(seed, a, b, M, ntotal):
    data = np.zeros(ntotal)
    for i in range(1,ntotal):
        data[i] = np.mod((a*data[i-1]+b), M)
    return data
```

```
# C12
import numpy as np
def rng(seed, a, b, M, ntotal):
    data = np.zeros(ntotal)
    data[0] = seed
    for i in range(1,ntotal):
        data[i] = np.mod((a*data[i-1]+b), M)
    return data/np.float(M)
```

```
# C13
import numpy as np
def rng(seed, a, b, M, ntotal):
    data = np.zeros(ntotal)
    data[0] = seed
    for i in range(1,ntotal):
        data[i] = np.mod((a*data[i-1]+b), M)/np.float(M)
    return data
```

```
# C14
import numpy as np
def rng(seed, a, b, M, ntotal):
    data = np.zeros(ntotal)
    data[0] = seed
    for i in range(0,ntotal):
        data[i+1] = np.mod((a*data[i]+b), M)
    return data/np.float(M)
```

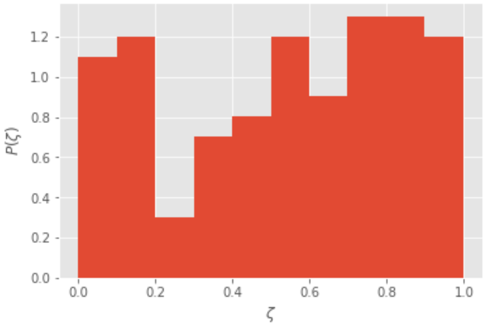
- C11 only
- C12 only
- C13 only
- C14 only

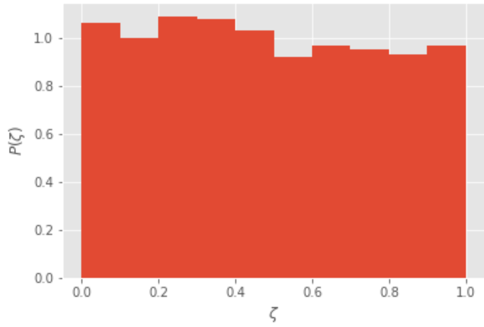
Submit You have used 0 of 2 attempts

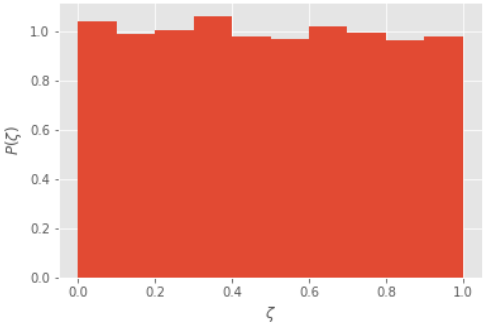
Homework 2-3

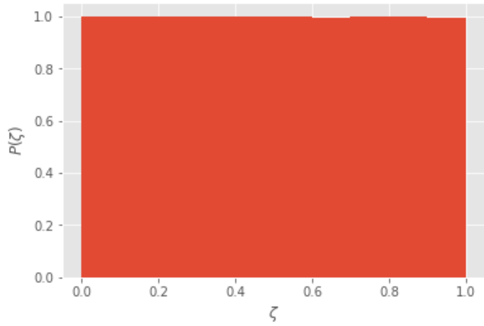
0.0/1.0 point (graded)

Let us first examine the distribution of the random numbers which will be generated by the method explained in previous Homework. Using the correct 'rng' function defined in the previous problem, with $a = 8121$, $b = 28411$, $M = 134456$, and initial value (seed) $X_0 = 123456$, generate a sequence of $5 \cdot 10^5$ random numbers. Draw a histogram of these numbers. Which of the following graphs (G41 - G44) is the closest to what you obtained?

G41


G42


G43


G44


- ○ G41
- ○ G42
- ○ G43
- ○ G44

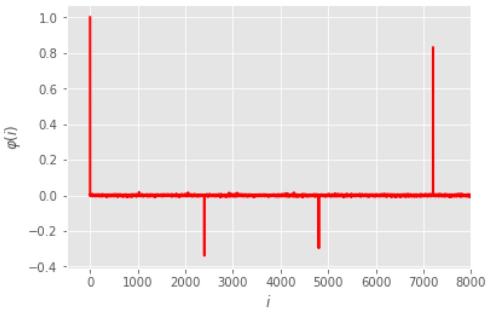
Submit You have used 0 of 2 attempts

Homework 2-4

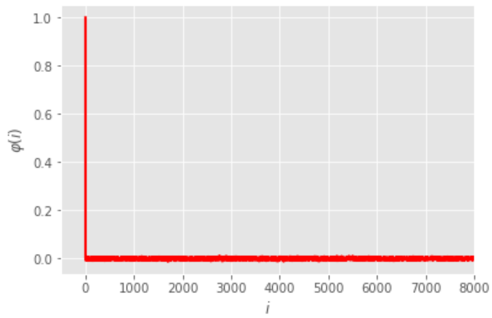
0.0/2.0 points (graded)

Let us also examine the correlation. Use the code example introduced in Part 2, to compute the correlation function for the sequence of random numbers generated using the 'rng' function of Homework 2 and 3. Using the same parameters (a,b,M) and seed (X_0) as in Homework 3, generate $N = 10^5$ random numbers and calculate the correlation function $\varphi(i)$ (Eq. D2). Plot the normalized correlation $\varphi(i)/\varphi(0)$ from $i = 0$ to $i = 8000$. Which of the following graphs (G51 - G54) is closest to what you obtained?

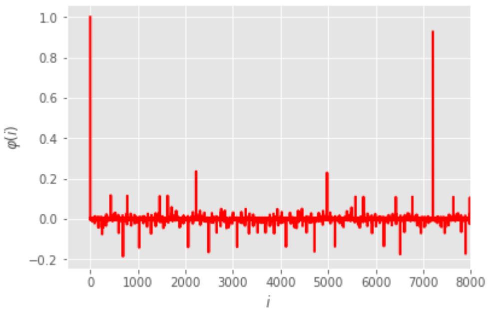
G51



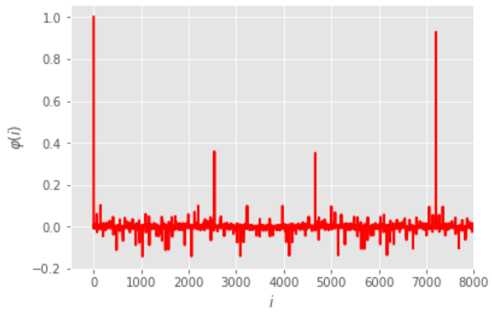
G52



G53



G54



- G51
- G52
- G53
- G54

Submit You have used 0 of 2 attempts

Homework 2-5

0.0/1.0 point (graded)

Generate $N = 10^5$ random numbers from a Gaussian distribution. Choose the closest values from below for the percentage of points that are within two standard deviations from the average.

- 50 %
- 90 %
- 95 %
- 98 %
- 99 %
- 99.9 %

Submit You have used 0 of 2 attempts

Homework 2-6

0.0/1.0 point (graded)
Imagine you have a distribution for a random variable that has a bell shape similar to that of a Gaussian. Looking at the distribution near the tails, you notice that 0.01 % of the points are more than 5σ away from the average. Can you say that the distribution is Gaussian?

- Yes
- No

Submit You have used 0 of 1 attempt