**Create a vector (array) of 1XN dimension representing N- dimensional feature vector of a sample. Write a program to compute the mean and variance of the elements present in the array.**

import numpy as np  
n = int(input())  
sample\_x = np.random.random\_integers(1,10,size=n)  
print(sample\_x)

10  
[ 7 6 10 1 3 1 10 9 3 1]

<ipython-input-1-0e8ecd011eae>:3: DeprecationWarning: This function is deprecated. Please call randint(1, 10 + 1) instead  
 sample\_x = np.random.random\_integers(1,10,size=n)

**Sample Variance**

sum = 0  
for i in range(n):  
 sum = sum + (sample\_x[i]- np.mean(sample\_x))\*\*2  
print(sum/n-1)

11.690000000000001

**Population Variance**

sum = 0  
for i in range(n):  
 sum = sum + (sample\_x[i]- np.mean(sample\_x))\*\*2  
print(sum/n)

12.690000000000001

**Create two vectors each of dimension 1XM each representing N- dimensional feature vector of a sample. Write a program to compute the Covariance between them.**

n = int(input("Enter n"))  
X = np.random.random\_integers(1,20,n)  
Y = np.random.random\_integers(7,10,n)  
print(X)  
print(Y)  
cov = np.sum((X - np.mean(X))\*(Y-np.mean(Y)))  
cov = (cov)/n-1  
print(cov)

Enter n10  
[ 1 3 8 12 14 13 16 8 7 2]  
[10 9 9 7 10 10 9 10 7 7]  
0.07999999999999963

<ipython-input-12-428a950d9ca6>:2: DeprecationWarning: This function is deprecated. Please call randint(1, 20 + 1) instead  
 X = np.random.random\_integers(1,20,n)  
<ipython-input-12-428a950d9ca6>:3: DeprecationWarning: This function is deprecated. Please call randint(7, 10 + 1) instead  
 Y = np.random.random\_integers(7,10,n)

**Create two vectors each of dimension 1XN. Write a program to compute the Correlation between them.**

n = int(input("Enter n"))  
X = np.random.random\_integers(1,20,n)  
Y = np.random.random\_integers(7,10,n)  
print(X)  
print(Y)  
cov = np.sum((X - np.mean(X))\*(Y-np.mean(Y)))  
cov = (cov)/n-1  
print(cov)

Enter n10  
[20 9 9 1 1 17 2 14 2 16]  
[ 8 7 10 10 8 8 9 9 10 9]  
-3.5799999999999996

<ipython-input-13-428a950d9ca6>:2: DeprecationWarning: This function is deprecated. Please call randint(1, 20 + 1) instead  
 X = np.random.random\_integers(1,20,n)  
<ipython-input-13-428a950d9ca6>:3: DeprecationWarning: This function is deprecated. Please call randint(7, 10 + 1) instead  
 Y = np.random.random\_integers(7,10,n)

**Create a Matrix of MXN dimension representing the M-dimensional feature vector for N number of samples i. e (i,j)th entry of the matrix represents the ith feature of jth sample. Write a program to compute the covariance matrix and correlation matrix.**

m = int(input())  
n = int(input())  
matrix = np.random.rand(m,n)  
print("Matrix")  
print(matrix)  
print("Covariance Matrix")  
cov = np.cov(matrix,bias=False)  
print(cov)  
print("Correlation Matrix")  
corr = np.corrcoef(matrix)  
print(corr)

3  
5  
Matrix  
[[0.33767627 0.84860322 0.03096532 0.45651923 0.17078886]  
 [0.71655271 0.13461718 0.96926888 0.1032955 0.97011038]  
 [0.80602509 0.03058643 0.31391539 0.70218265 0.68997347]]  
Covariance Matrix  
[[ 0.09805377 -0.11712908 -0.04794281]  
 [-0.11712908 0.18699212 0.03905054]  
 [-0.04794281 0.03905054 0.10630783]]  
Correlation Matrix  
[[ 1. -0.86500963 -0.46957901]  
 [-0.86500963 1. 0.27696999]  
 [-0.46957901 0.27696999 1. ]]