

# **ASSIGNMENT 1**

**Name:** Vivek Kamboj

**Roll No.:** 2018IMT-109

**Course:** Machine Learning Lab

**Course Code:** ITIT-4107

**Deadline:** 18 September (11:59 PM)

## **1.a:**

### **Aim:**

To Create a random dataset with two input features and one output feature (class labels). The two input features shall be random variables sampled from a gaussian distribution with a mean 8, the standard deviation of 1.5 as well as the output feature may be sampled from a binomial distribution with a probability of 1 as 0.6. Create 20 instances of the sampled data and plot the same.

### **Procedure:**

The first task is to create a Gaussian distribution. Gaussian distribution is a normal distribution. As per the question, this normal distribution has a mean of 8 and a standard deviation of 1.5. We need two input features that are random. For this, I will be using the random function of NumPy and then declare two input features, and later plot the histogram to show the values. Similarly, for the output feature, I will make binomial distribution assuming 100 trials with a probability of 0.6. The code for this procedure implementation is below:

### **Code:**

```
import numpy as np
import matplotlib.pyplot as plt
rand=np.random
fig,ax=plt.subplots(nrows=1,ncols=3,figsize=(12,7))
input1=rand.normal(8,1.5,20)
input2=rand.normal(8,1.5,20)
output= rand.binomial(100,0.6,20)

plt.subplot(1,3,1)
plt.title('input1')
```

```
plt.hist(input1,bins=50,color='green')
```

```
plt.subplot(1,3,2)
```

```
plt.title('input2')
```

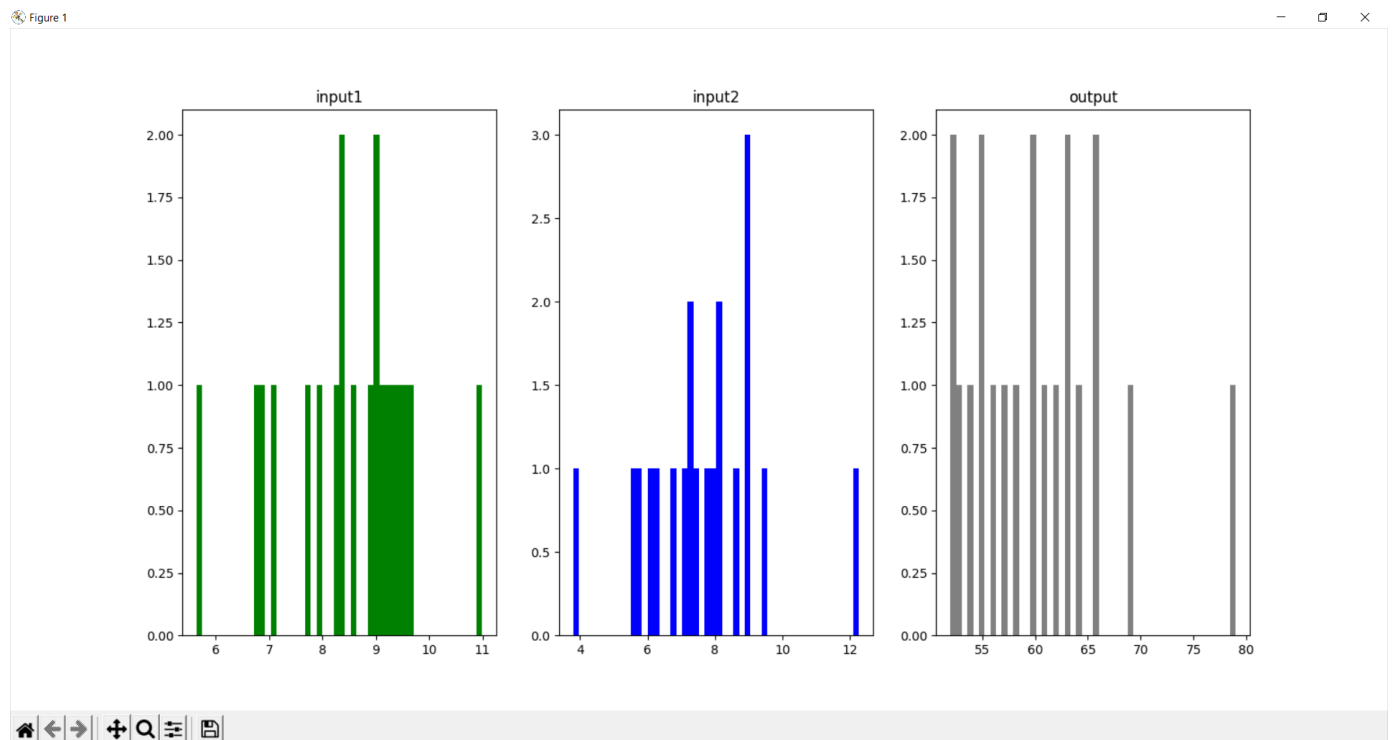
```
plt.hist(input2,bins=50,color='blue')
```

```
plt.subplot(1,3,3)
```

```
plt.title('output')
```

```
plt.hist(output,bins=50,color='gray')
```

```
plt.show()
```



### Inference:

From this, I have inferred and analyzed the way to generate a dataset in gaussian and binomial distribution using inbuilt functions of libraries in python.

## 1.b:

### Aim:

To Create a random dataset with two input features and one output feature. Create 20 instances of data by sampling the two input features from a gaussian distribution with a mean of 8, a standard deviation of 1.5. Label these instances as 0. Generate another 20 instances of data by sampling the two input features from a gaussian distribution with a mean of 15 and standard deviation of 1.5 and label these instances as 1 and plotting these.

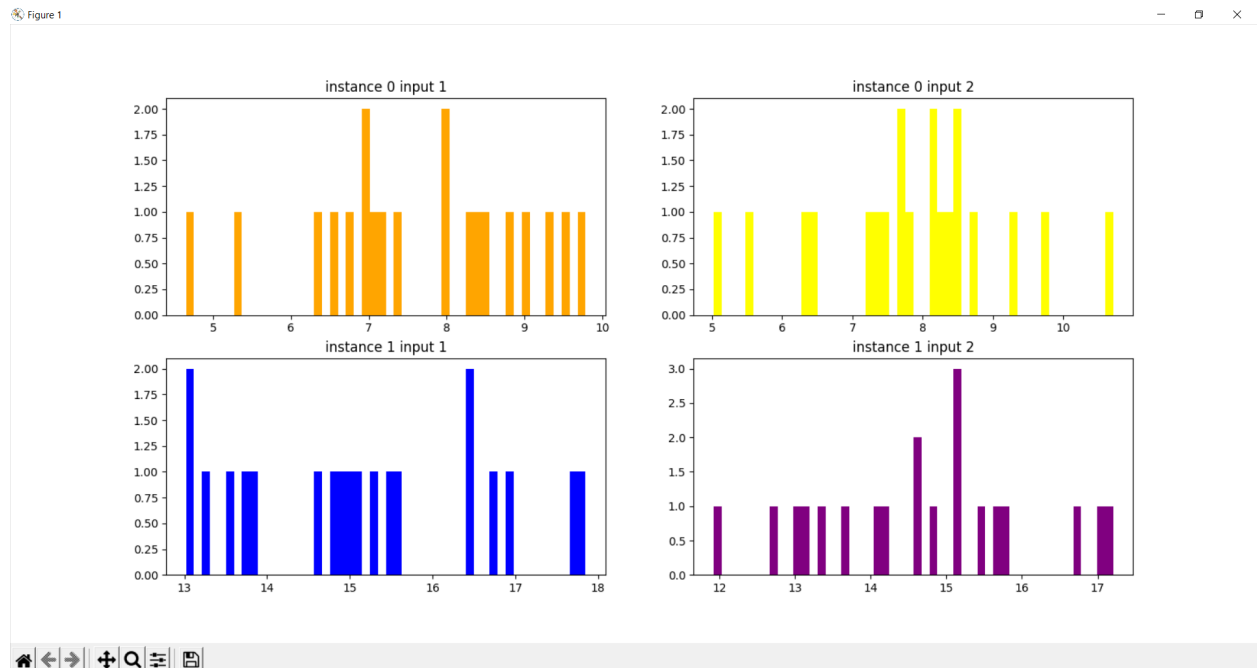
### Procedure:

As per the question, there is a need to create two normal distributions. First, one has a mean of 8 and a standard deviation of 1.5, the second one having a mean of 15 and a standard deviation of 1.5. We need two input features that are random for the first instance namely instance 0. For this, I will be using the random function of NumPy and then declare two input features, and later plot the histogram to show the values. Similarly, for the second instance namely instance 1 will repeat the above steps. The code for this procedure implementation is below:

### Code:

```
import numpy as np
import matplotlib.pyplot as plt
rand = np.random
fig, ax = plt.subplots(nrows=2, ncols=2, figsize=(12, 7))
i0input1=rand.normal(8,1.5,20)
i0input2=rand.normal(8,1.5,20)
i1input1=rand.normal(15,1.5,20)
i1input2=rand.normal(15,1.5,20)
plt.subplot(2, 2,1)
plt.title('instance 0 input 1')
plt.hist(i0input1, bins=50,color='orange')
plt.subplot(2, 2,2)
plt.title('instance 0 input 2')
plt.hist(i0input2, bins=50,color='yellow')
plt.subplot(2, 2,3)
plt.title('instance 1 input 1')
plt.hist(i1input1, bins=50,color='blue')
plt.subplot(2, 2,4)
plt.title('instance 1 input 2')
plt.hist(i1input2, bins=50,color='purple')
```

plt.show()



### Inference:

From this, I have inferred and analyzed the way to generate a dataset in two different Gaussian distributions using inbuilt functions of libraries in python and random function.

## **1.c:**

### **Aim:**

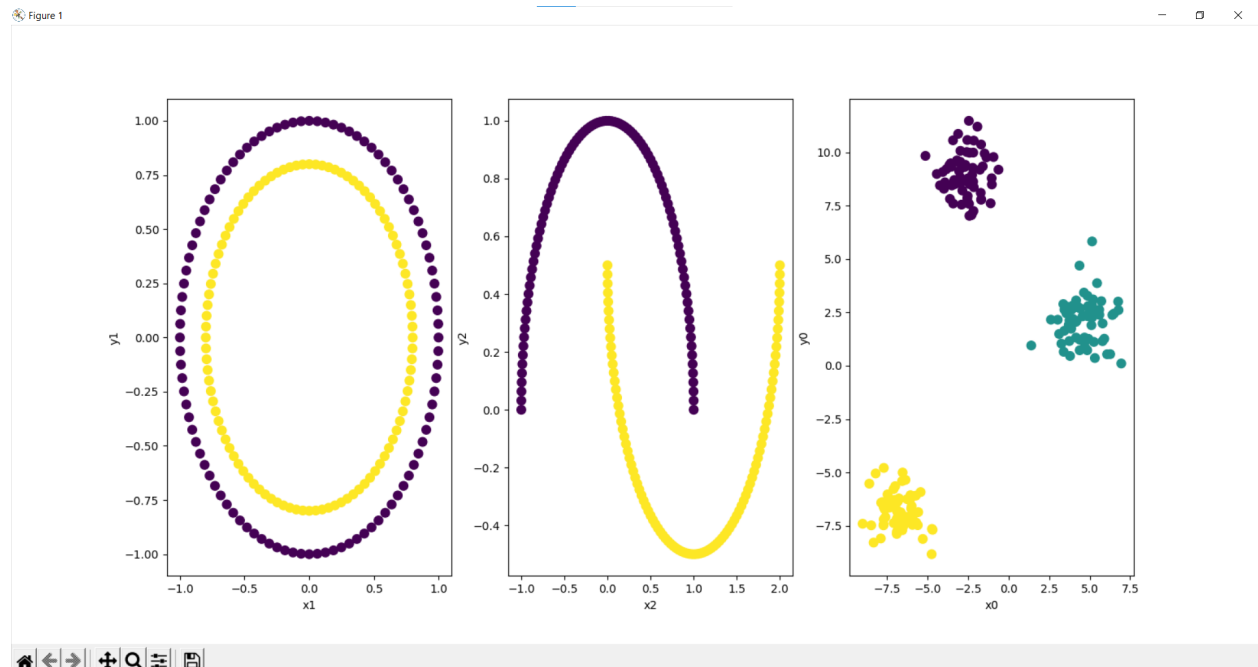
Generate dataset using make\_moon, make\_circles, make\_blobs functions from sklearn library and plot them.

### **Procedure:**

There are inbuilt functions in the sklearn library to make the moon, make circles and make blob. I will be importing that dataset and then using matplotlib library I will plot the generated datasets. The code for this procedure implementation is below:

### **Code:**

```
import numpy as np
from sklearn import datasets
import matplotlib.pyplot as plt
X, y = datasets.make_circles(200, noise=0, random_state=42)
X1, y1 = datasets.make_moons(200, noise=0, random_state=42)
X2, y2= datasets.make_blobs(200, random_state=42)
fig, ax = plt.subplots(nrows=1, ncols=3, figsize=(12, 7))
plt.xlabel("x0", fontsize=10)
plt.ylabel("y0", fontsize=10)
plt.subplot(1,3,1)
plt.scatter(X[:,0], X[:,1], s=60, c=y)
plt.xlabel("x1", fontsize=10)
plt.ylabel("y1", fontsize=10)
plt.subplot(1,3,2)
plt.scatter(X1[:,0], X1[:,1], s=60, c=y1)
plt.xlabel("x2", fontsize=10)
plt.ylabel("y2", fontsize=10)
plt.subplot(1,3,3)
plt.scatter(X2[:,0], X2[:,1], s=60, c=y2)
plt.show()
```



### Inference:

From this, I have learned the use of `make_moon`, `make_circles`, `make_blobs` functions to generate datasets and visualize them with the help of a plot.