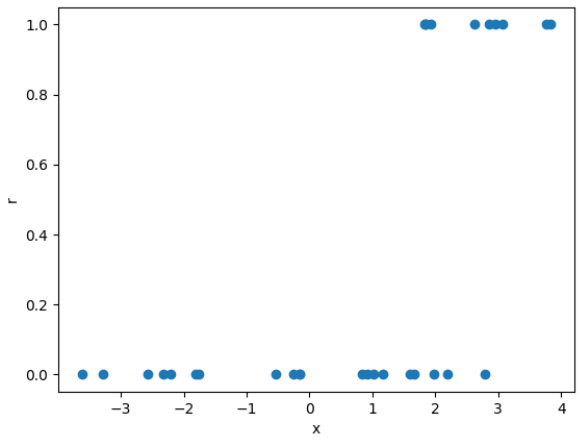
1.

```python
import random
import math
import matplotlib.pyplot as plt
import numpy as np

x=0.0
outcome=0
w1, w0, eta = 0.01, 0.01, 0.01
x_record, r_record, entropy_record = [], [], []

def sigmoid(x, w1, w0):
    return 1 / (1 + math.exp(-w1*x-w0))

def func(x):    # y(x)
    if x<1:
        return 0
    elif x>3:
        return 1
    else:
        return 0.5*(x-1)
```

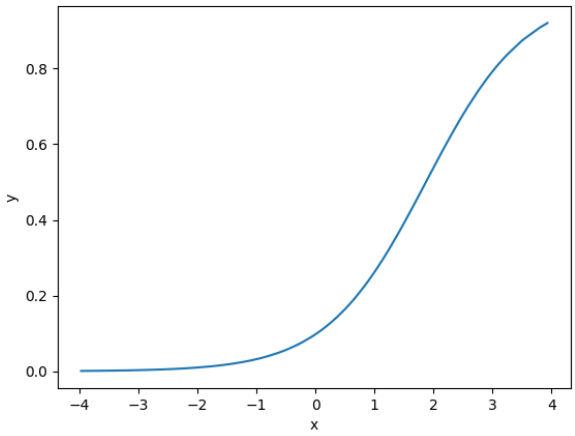
```python
for iteration in range(30): # generate 30 sample points
    x = random.uniform(-4, 4)
    zeta = random.uniform(0, 1)
    if zeta < func(x):
        outcome = 1
    else:
        outcome = 0
    x_record.append(float(x))
    r_record.append(int(outcome))

epoch=100
for iteration in range(epoch):
    sum1, sum2, sum_en = 0.0, 0.0, 0.0
    for i in range(30):
        y = sigmoid(x_record[i], w1, w0)
        sum1 += (r_record[i]-y)*x_record[i] # dE/dw1
        sum2 += (r_record[i]-y) # dE/dw0
        sum_en -= (r_record[i]*math.log(y) + (1-r_record[i])*math.log(1-y)) # cross entropy
    w1 += eta*sum1
    w0 += eta*sum2
    if iteration%10==0:
        entropy_record.append(sum_en)
        #print(f"iteration = {iteration}, error = {sum_en}")

print(f"w1 = {w1}\nw0 = {w0}")
```

1(a).

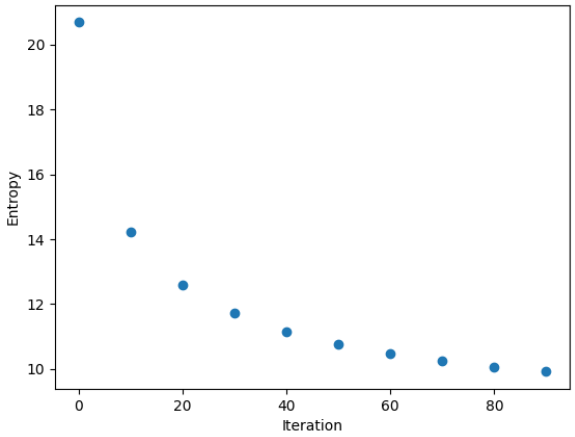


1(b).

```
w1 = 1.187439367433231
w0 = -2.2229277634653726
```



1(c).



2.

```python
import random
import numpy as np
import math

x1, x2 = 0.0, 0.0
y1, y2 = 0.0, 0.0
count1_error, count2_error = 0, 0

mu1, sigma1 = np.array([0, 0]), 1
mu2, sigma2 = np.array([3, 4]), 3

xc = ((sigma2**2)*mu1 - (sigma1**2)*mu2) / (sigma2 - sigma1)
r_square = ((sigma1**2 * sigma2**2)/(sigma2**2 - sigma1**2))*(np.sum(np.square(mu1+mu2))/(sigma2**2 - sigma1**2) + 4*math.log(sigma2/sigma1))
print("center =", xc)
print("radius =", math.sqrt(r_square))
print("=====")

# class 1 with mean = (0,0), sigma = 1
# (horizontal, vertical) = (x1, x2)
for i in range(1000):
    x1=random.gauss(0,1)
    x2=random.gauss(0,1)

    if (x1 - xc[0])**2 + (x2 - xc[1])**2 > r_square:
        count1_error += 1
print(f"error rate 1 = {count1_error/1000}")
print("=====")
```

```python
# class 2 with mean = (3,4), sigma = 3
# (horizontal, vertical) = (y1, y2)
count2_error_list = []
for _ in range(10):
    count2_error = 0
    for i in range(1000):
        y1=3.0+random.gauss(0,3)
        y2=4.0+random.gauss(0,3)

        if (y1 - xc[0])**2 + (y2 - xc[1])**2 < r_square:
            count2_error += 1
    #print(f"error rate 2 = {count2_error/1000}")
    count2_error_list.append(count2_error/1000)

tmp=np.array(count2_error_list)
print(count2_error_list)
print(f"mean = {np.mean(tmp)}")
print(f"std = {np.std(tmp)}")
print(f"count 2 error = {np.mean(tmp)} +- {np.std(tmp)}")
```

2(a).

```
center = [-1.5 -2. ]
radius = 2.908501383703727
```

2(b).

```
error rate 1 = 0.41
```

2(c).

```
[0.042, 0.03, 0.024, 0.021, 0.037, 0.027, 0.04, 0.037, 0.026, 0.02]
mean = 0.030400000000000003
std = 0.007631513611335565
count 2 error = 0.030400000000000003 +- 0.007631513611335565
```