

```
import numpy as np
import matplotlib.pyplot as plt

N, p = 30, 20
np.random.seed(0)
X = np.random.randn(N,p)
Y = 2*np.random.randint(2, size = N) - 1

Theta = np.random.randn(20)

def calculate_loss(x,y,theta):
    return np.sum(np.log(1+np.exp(-y * (x@theta))))/N

max_iter = 19000
lr = 0.001

loss = []

for i in range(max_iter):
    index = np.random.randint(0,N)
    val = np.exp(-Y[index]*(X[index]@Theta))
    gradient = -(1/(1+val))*val*Y[index]*X[index]
    loss.append(calculate_loss(X,Y,Theta))
    Theta -= lr*gradient

print(Theta)

plt.plot(np.arange(0,max_iter) , loss)
plt.savefig('p1_loss.png')
plt.show()
```

Result

[0.25566333 0.19552217 -0.61144044 0.61714755 -0.6213656 0.09936487
-0.84369411 -0.50592759 1.30460593 -0.57560536 -0.52782133 0.73971214
1.04934594 -0.66045565 0.48271393 -0.68412194 -0.54934705 -0.66221099
-0.40259889 -0.44405776]

