Dataset link

https://archive.ics.uci.edu/ml/datasets/Iris#targetText=UCI%20Machine%20Learning%20Repository%3A%
(https://archive.ics.uci.edu/ml/datasets/Iris#targetText=UCI%20Machine%20Learning%20Repository%3A%

In [20]:

```
%matplotlib inline
```

In [21]:

```
from sklearn import datasets
import numpy as np
import csv
import pandas as pd
from matplotlib.colors import ListedColormap
import matplotlib.pyplot as plt
from sklearn.metrics import accuracy_score
```

In [22]:

```
iris = datasets.load_iris()
x = iris.data[:, [0,2]]
y = iris.target
x_moded=np.delete(x,[100,101,102,103,104,105,106,107,108,109,110,111,112,113,114,115,116,11
y_moded=np.delete(y,[100,101,102,103,104,105,106,107,108,109,110,111,112,113,114,115,116,11

# splitting data into test and train data set
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x_moded,y_moded, test_size=0.3, random_
#print(x_train)
#print(y_train)


weights = np.zeros((1, 2))
gradient_matrix = np.zeros((2, 1))
hessian_matrix = np.zeros((2, 2))
result_matrix = np.zeros((2,1))
```

In [23]:

```
#creating activation function sigmoid function
def sigmoid(x):
    return 1/(1+np.exp(-x))
```

In [24]:

```
 training set
 = [0 for i in range(len(x_test))]
=0
or k in range(50):   #-------------------------------------------------------------epochs ar

   for i in range (0,len(x_train)-60): #--------------------------------------------training

       actual_result = weights[0][0]*x_train[i][0] + weights[0][1]*x_train[i][1]
       if actual_result > 0:
           fx = 1
              # print("greater than 0")
       if actual_result <= 0:
           fx = 0
               #print("less than 0")
        #-------------
       d=y_train[i]
       w1=weights[0][0]
       w2=weights[0][1]
       x1=x_train[i][0]
       x2=x_train[i][1]



       gradient_matrix[0][0] = -x1*(d-w1*x1-w2*x2) #-------------------------putting value
       gradient_matrix[1][0] = -x2*(d-w1*x1-w2*x2)

       hessian_matrix[0][0] = x1**2
       hessian_matrix[0][1] = x1*x2 #-----------------------------------------Putting values
       hessian_matrix[1][0] = x2*x1
       hessian_matrix[1][1] = x2**2


       A_inv = np.linalg.pinv(hessian_matrix)#pseudo matrix for matrix cannot be find
       #print(A_inv)

       result_matrix[0][0] = (A_inv[0][0]*gradient_matrix[0][0])+ (A_inv[0][1]*gradient_matr
       result_matrix[1][0] = (A_inv[1][0]*gradient_matrix[0][0])+ (A_inv[1][1]*gradient_matr


       #-------------

       if fx != y_train[i]:
           error = y_train[i] - fx

           weights[0][0] = weights[0][0] - result_matrix[0][0]# ---------------------------
           weights[0][1] = weights[0][1] - result_matrix[1][0]#-----------------------------
           #weights[0][0] = weights[0][0] + error*x_train[i][0]
           #weights[0][1] = weights[0][1] + error*x_train[i][1]
               #print(weights[:])

           p=p+1
           print(p)
           print("learning")
           #plt.plot(p)
       else:
           weights[0][0] = weights[0][0]
           weights[0][1] = weights[0][1]

           print("no error")
   #print(weights[0][i])

       for i in range (0,len(x_test)):                  #testing data included in the loop

           actual_result = weights[0][0]*x_test[i][0] + weights[0][1]*x_test[i][1]
           if actual_result > 0:
               fx = 1
              # print("greater than 0")
           if actual_result <= 0:
               fx = 0
               #print("less than 0")


           n[i]=fx
print("the Accuracy is\n")
rint ("Accuracy",+accuracy_score(y_test[:],n[:])) #finding accuracy #-----------------------
rint('\033[1m',  (accuracy_score(y_test[:],n[:]) *100),'%' , '\033[0m')
```

```
no error
1
learning
2
learning
no error
3
learning
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
```

```
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
  no error
```

```
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
no error
Accuracy 0.7
 70.0 %
```

In [ ]:

In [ ]:

In [ ]: