import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
from sklearn.model\_selection import train\_test\_split  
  
  
data = pd.read\_csv('E:\\机器学习\\机器学习作业\_数据/regress\_data1.csv')  
X = data['人口'].values.reshape(-1, 1)  
y = data['收益'].values.reshape(-1, 1)  
  
X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)  
  
  
m\_train, n = X\_train.shape  
X\_train = np.hstack([np.ones((m\_train, 1)), X\_train])  
m\_test, \_ = X\_test.shape  
X\_test = np.hstack([np.ones((m\_test, 1)), X\_test])  
  
  
theta = np.zeros((n + 1, 1))  
  
  
learning\_rate = 0.01  
iterations = 1000  
  
  
train\_losses = []  
test\_losses = []  
  
  
for i in range(iterations):  
  
 y\_train\_pred = np.dot(X\_train, theta)  
  
 train\_loss = np.mean((y\_train\_pred - y\_train) \*\* 2) / 2  
 train\_losses.append(train\_loss)  
  
  
 y\_test\_pred = np.dot(X\_test, theta)  
  
 test\_loss = np.mean((y\_test\_pred - y\_test) \*\* 2) / 2  
 test\_losses.append(test\_loss)  
  
  
 error\_train = y\_train\_pred - y\_train  
 gradient = (1 / m\_train) \* np.dot(X\_train.T, error\_train)  
  
  
 theta -= learning\_rate \* gradient  
  
  
plt.plot(train\_losses, label='Training Loss')  
plt.plot(test\_losses, label='Test Loss')  
plt.xlabel('Iterations')  
plt.ylabel('Loss')  
plt.title('Training and Test Loss over Iterations')  
plt.legend()  
plt.show()