实验手册

任务：对散点图的数据确定

使用模型；非线性逻辑回归

实验步骤为：

数据读入

train = np.loadtxt('HTRU\_2\_train.csv',delimiter = ',')#获取训练级

test = np.loadtxt('HTRU\_2\_test.csv',delimiter = ',')#获取测试集

数据升维

def Linear\_mapping(train\_1,test\_1):

tr\_min = train\_1.min(axis = 0)

te\_min = test\_1.min(axis = 0)

tr\_max = train\_1.max(axis = 0)

te\_max = test\_1.max(axis = 0)

for i in range(1,len(train\_1[0])):

train\_1[...,i] = (train\_1[...,i]-tr\_min[i])/(tr\_max[i] - tr\_min[i])

test\_1[...,i] = (test\_1[...,i]-te\_min[i])/(te\_max[i] - te\_min[i])

return train\_1,test\_1

逻辑回归

梯度下降

得出精确度

使用函数：

Generating\_higher(train,test)

对数据集进行高次产生

Linear\_mapping(train\_1,test\_1)

线性映射到0，1

grad\_descent(dataMathIn,classLabels)

梯度下降函数

function\_1(x,weights)

预测函数

def function\_1(x,weights):

y = []

for i in x:

if (np.mat(i) \* np.mat(weights)) >= 0:

y.append(1)

else:

y.append(0)

return y

梯度下降的原理

对函数求导不断逼近最小值

非线性逻辑回归

在逻辑回归的基础上引入非线性使函数弯曲从而达到最大的预测

预测正确率为0.80952