**电 子 科 技 大 学**

**实 验 报 告**

**学生姓名：刘文晨 学 号：2018080901006 指导教师：钟秀琴**

**实验地点：主楼A2-412 实验时间：2020-11-1 周日 9-节**

**一、实验室名称：计算机学院实验中心**

**二、实验项目名称：BP神经网络实验**

**三、实验学时：4学时**

**四、实验原理：**

误差逆传播(back propagation, BP)算法是一种计算单个权值变化引起网络性能变化的较为简单的方法。由于BP算法过程包含从输出节点开始，反向地向第一隐含层(即最接近输入层的隐含层)传播由总误差引起的权值修正，所以称为“反向传播”。BP神经网络是有教师指导训练方式的多层前馈网络，其基本思想是：从网络输入节点输入的样本信号向前传播，经隐含层节点和输出层节点处的非线性函数作用后，从输出节点获得输出。若在输出节点得不到样本的期望输出，则建立样本的网络输出与其期望输出的误差信号，并将此误差信号沿原连接路径逆向传播，去逐层修改网络的权值和节点处阈值，这种信号正向传播与误差信号逆向传播修改权值和阈值的过程反复进行，直训练样本集的网络输出误差满足一定精度要求为止。

**五、实验目的：**

编程实现BP神经网络算法；理解算法原理。

**六、实验内容：**

将Iris（鸢尾花）数据集分为训练集（Iris-train.txt）和测试集（Iris-test.txt），分别含75个样本，每个集合中每种花各有25个样本。为了方便训练，将3类花分别编号为1，2，3 。使用这些数据训练一个4输入（分别对应4个特征）、隐含层（10个神经元）、3输出（分别对应该样本属于某一品种的可能性大小）的神经网络（4\*10\*3）。

使用训练集对网络进行训练，再预测测试集中每个样本的标签，并输出预测准确率（独立运行10次，列出10次的准确率，并输出平均准确率和标准差）。

要求：

（1）提交源代码及可执行文件。

（2）提交实验报告，内容包括：对代码的简单说明、运行结果的截图及说明等。

注意：实验报告中，需指明梯度下降过程中的学习率，并与相应的源码对应。

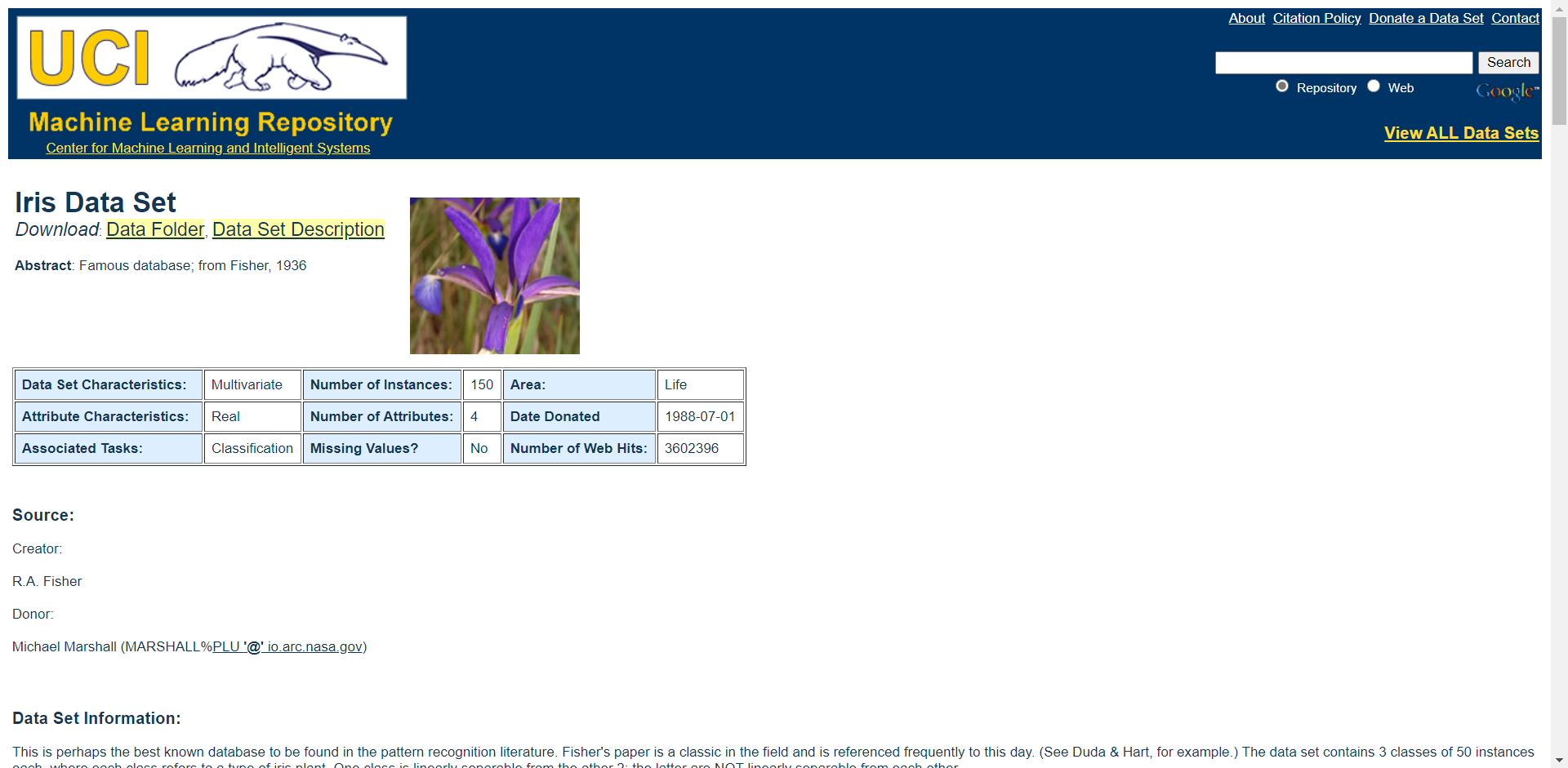
**七、实验器材（设备、元器件）：**

PC微机一台

**八、实验步骤：**

1. **下载Iris数据集并导入**

进入[Iris Data Set](http://archive.ics.uci.edu/ml/datasets/Iris)下载鸢尾花数据集iris.data：



导入语句为：

f**=**fopen**(**'C:\Users\81228\Desktop\资料\人工智能\实验\实验3\iris.data'**);**

括号里是iris.data的地址。

1. **数据归一化**

%数据归一化处理

x\_max**=**max**(**data**);**

x\_min**=**min**(**data**);**

data**=(**data**-**ones**(**m**,**1**)\***x\_min**)./(**ones**(**m**,**1**)\*(**x\_max**-**x\_min**));**

1. **划分训练样本和测试样本**

将Iris（鸢尾花）数据集分为训练集（Iris-train.txt）和测试集（Iris-test.txt），分别含75个样本。

%划分训练样本和测试样本 大约1/2用作训练，1/2用作测试

num**=**round**(**m**/**2**/**3**);**

**for** i**=**1**:**3

temp**=**data**(**1**+**50**\*(**i**-**1**):**50**\***i**,:);**

sel**=**randperm**(**50**,**num**);**

test**(**1**+**num**\*(**i**-**1**):**num**\***i**,:)=**temp**(**sel**,**1**:**4**);**

test\_label**(**1**+**num**\*(**i**-**1**):**num**\***i**,:)=**temp**(**sel**,**5**:**7**);**

temp**(**sel**,:)=[];**

train**(**1**+(**50**-**num**)\*(**i**-**1**):(**50**-**num**)\***i**,:)=**temp**(:,**1**:**4**);**

train\_label**(**1**+(**50**-**num**)\*(**i**-**1**):(**50**-**num**)\***i**,:)=**temp**(:,**5**:**7**);**

**end**

1. **BP神经网络设置**

alpha**=**4**;**%输入神经元数目

beta**=**10**;**%隐层神经元数目

lamda**=**3**;**%输出神经元数目

rng**(**'shuffle'**)**

W1**=**rand**(**alpha**,**beta**);**%输入层和隐层之间的权值矩阵

W2**=**rand**(**beta**,**lamda**);**%隐层和输出层间的权值矩阵

B1**=**rand**(**1**,**beta**);**%隐层阈值矩阵

B2**=**rand**(**1**,**lamda**);**%输出层阈值矩阵

B22**=**B2**;**

W11**=**W1**;**

Eta**=**1**;**%学习率

iter\_max**=**10000**;**%最大迭代次数

iter**=**1**;**

1. **测试**

%测试

**[**result**,**accuracy**]=**BP\_test**(**test**,**test\_label**,**W1**,**W2**,**B1**,**B2**);**

disp**(**'结果为：'**)**

result**'**

disp**(**strcat**(**'准确率为'**,**num2str**(**accuracy**\***100**),**'%'**))**

**function** **[**result**,**accuracy**]=**BP\_test**(**test**,**test\_label**,**W1**,**W2**,**B1**,**B2**)**

%返回分类的结果

%accuracy准确率

Hidden\_in**=**test**\***W1**;**%隐层输入

Hidden\_out**=**sigmod**(**Hidden\_in**-**B1**);**%隐层输出

Output\_in**=**Hidden\_out**\***W2**;**%输入层输入

Output\_out**=**sigmod**(**Output\_in**-**B2**);**

**[~,**result**]=**max**(**Output\_out**,[],**2**);**

**[~,**index2**]=**max**(**test\_label**,[],**2**);**

right**=**sum**(**result**==**index2**);**%统计分类正确的个数

total**=**size**(**test**,**1**);**%总个数

accuracy**=**right**/**total**;**

**end**

%sigmod函数在matlab里是logsig函数

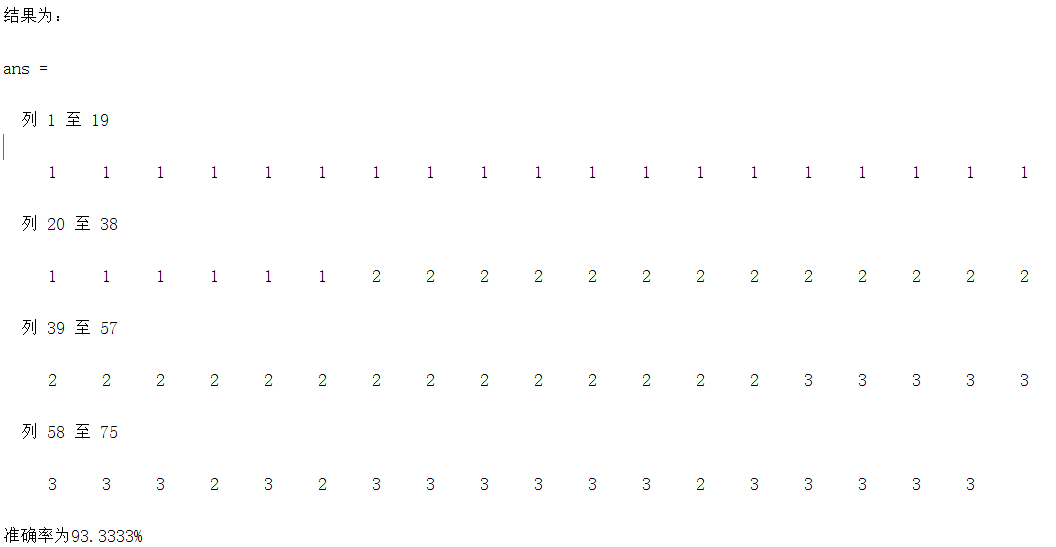
**function** **[**Y**]=**sigmod**(**X**)**

Y**=**1.**/(**1**+**exp**(-**1**).^**X**);**

**end**

**九、实验数据及结果分析：**

单次运行结果为：



独立运行10次，预测准确率分别为：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 96% | 93.3333% | 97.3333% | 93.3333% | 98.6667% |
| 89.3333% | 94.6667% | 96% | 96% | 97.3333% |

平均准确率为95%，标准差为0.025438487。

**十、实验结论：**

BP神经网络算法能够很好地解决鸢尾花种类识别问题。

**十一、总结及心得体会：**

掌握了BP神经网络算法解决鸢尾花种类识别问题，加深了对机器学习的理解。

**十二、对本实验过程及方法、手段的改进建议：**

利用PSO算法对参数进行优化。

**报告评分：**

**指导教师签字：**