

实验报告四

学号：20201060287

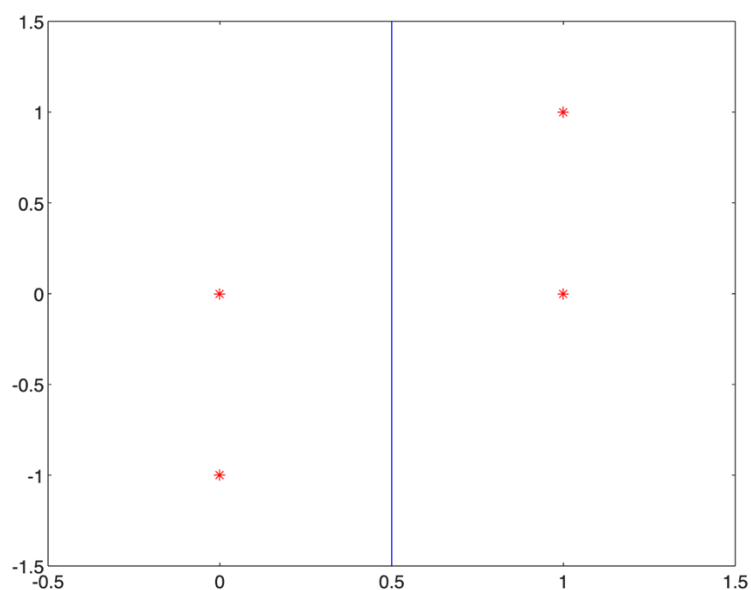
姓名：李昂

实验名称：感知器算法

实验内容：使用 MATLAB 编程环境，实现感知器算法并进行实验

实验要求及结果：

1.运行 demo_4.m 文件，画出该示例程序中四个二维训练样本的分布图以及决策面图（将分布图和决策面图画在一起即可），并给出增广权向量 w 的值和 w 的迭代更新次数。



$w(1)=-2.00$, $w(2)=0.00$, $w(3)=1.00$

```
function [W, k] = PA(X, W, c, classes)
    % X 为训练样本形成的矩阵，训练样本的个数为 N；W 为权向量；c 为校正增量
    % classes 为各训练样本的类别且为一个 N 维向量，w1 类用 1 表示，w2 类用 -1 表示
    [N, n] = size(X); % 训练样本的大小 N*n，N 即训练样本的个数，n 即每个训练样本的维数
```

```
A = ones(N, 1);
X1 = [X A]; % 将训练样本写成增广向量形式
% 对训练样本规范化
for i = 1:N
    X1(i, :) = classes(i) * X1(i, :);
end
k = 0; % 迭代次数
a = 0; % 每一轮迭代中判别函数小于或等于 0 的个数，即每轮中错判的次数
b = 0; % 迭代轮数的总数
b = b + 1;
for j = 1:N
    if dot(W, X1(j, :), 2) > 0
        k = k + 1;
        W = W;
    else
        a = a + 1;
        W = W + c * X1(j, :);
        k = k + 1;
    end
end
while (a >= 1)
    a = 0;
    b = b + 1;
    for j = 1:N
        if dot(W, X1(j, :), 2) > 0
            k = k + 1;
            W = W;
        else
            a = a + 1;
            W = W + c * X1(j, :);
            k = k + 1;
        end
    end
end
end
end
```

2. 改写上述示例程序，采用 iris 数据集中的 iris-setosa 和 iris-versicolor 这两类数据进行感知器实验，分别令修正系数 $c=1$ 和 $c=0.5$ ，给出相应的 w 的值和 w 的迭代更新次数。

$c=1.0$

```
>>> clear functions
run("/Users/wallanceleon/Desktop/模式识别/实验/实验四/demo_4_2.m")
```

```
>>> w的值为w(1)=1.30, w(2)=4.10, w(3)=-5.20, w(4)=-2.20, w(5)=1.00
w的迭代更新次数为400
```

$c=0.5$

```
>>> clear functions
run("/Users/wallanceleon/Desktop/模式识别/实验/实验四/demo_
>>> 4_2.m")
w的值为w(1)=0.65, w(2)=2.05, w(3)=-2.60, w(4)=-1.10, w(5)=0.50
w的迭代更新次数为400
```

问题分析：

请分析在第 2 部分实验中，分别取 $c=1$ 和 $c=0.5$ 对增广权向量 w 的值得影响。

当 $c=1$ 时，修正系数较大，算法的收敛速度比较快，因此 w 的值也相对较大。从结果可以看出，每个属性对应的权重值都比较接近，说明它们对分类的贡献相对均衡。

当 $c=0.5$ 时，修正系数较小，算法的收敛速度比较慢，因此 w 的值也相对较小。从结果可以看出，每个属性对应的权重值都较小，但是它们之间的差异比较大，说明某些属性对分类的贡献更加显著。

附采用 iris 数据集进行感知器实验的代码（无需给出子程序 PA 的代码，只给出主程序代码即可）：

```
close all;
clear all;
clc;

% 读取原始数据
[attrib1, attrib2, attrib3, attrib4, class] = textread('iris.data',
```

```
'%f%f%f%f%s', 'delimiter', ',');
X = [attrib1, attrib2, attrib3, attrib4];
label_set = char('Iris-setosa','Iris-versicolor','Iris-virginica');
label = zeros(150, 1);
label(strcmp(class, 'Iris-setosa')) = 1;
label(strcmp(class, 'Iris-versicolor')) = 2;
label(strcmp(class, 'Iris-virginica')) = 3;

% 只保留 setosa 和 versicolor 两类数据, 令 setosa 类为正类, versicolor 类为负类
X(label==3,:)=[];
label(label==3)=[];
label(label==2)=-1;
classes=label;
w0 = [0, 0, 0, 0, 0];
c = 0.5;
[w, k] = PA(X, w0, c, label);
% 输出 w 的值和 w 的迭代更新次数
fprintf('w 的值为 w(1)=%4.2f, w(2)=%4.2f, w(3)=%4.2f, w(4)=%4.2f, w(5)=%4.2f\n', w(1), w(2), w(3), w(4), w(5));
fprintf('w 的迭代更新次数为%d\n', k);
```