Experiment 1 Perceptron Learning toward Linear Classification

一、Principle and Theory：

Assume that the samples of the two classes are linearly separable in the feature space. i.e., there exists a plane $G(X) = W^T X + w_{n+1} = 0$, where $W \in R^n$ and $X \in R^n$, such that all samples belonging to the first class are on one side of the plane, and all samples of the second class are on the opposite side. If such planes exist, the goal of the perceptron algorithm is to learn any one such plane, given the data points. Once the learning is completed and the plane is determined, it will be easy to classify new points in the future, as the points on one side of the plane will result in a positive value for $G(X) = W^T X + w_{n+1}$, while points on the other side will give a negative value.

According to the principle of perceptron learning, the weight vector $W \in R^n$ can be extended to $\widehat{W} = (w_1\, w_2 \cdots w_n\, w_{n+1})^T \in R^{n+1}$ and the feature vector may be extended to $\widehat{X} = (x_1\, x_2 \cdots x_n\; 1)^T \in R^{n+1}$ also, thus the plane of classification can be expressed as $G(X) = \widehat{W}^T \widehat{X} = 0$. The learning rule (algorithm) for the update of weights is designed as

$$\widehat{\mathbf{W}}(\mathbf{t}+\mathbf{1}) = \widehat{\mathbf{W}}(\mathbf{t}) + \frac{\mathbf{1}}{\mathbf{2}}\boldsymbol{\eta}\{\widehat{\mathbf{X}}(\mathbf{t}) - \widehat{\mathbf{X}}(\mathbf{t})\mathbf{sgn}[\widehat{\mathbf{W}}^{\mathbf{T}}(\mathbf{t})\widehat{\mathbf{X}}(\mathbf{t})]\}$$

$$= \begin{cases} \mathbf{W}(\mathbf{t}) & \widehat{W}^T(t)\widehat{X}(t) > 0 \\ \mathbf{W}(\mathbf{t}) + \boldsymbol{\eta}\widehat{\mathbf{X}}(\mathbf{t}) & \text{otherwise} \end{cases}$$

where $\boldsymbol{\eta}$ is the learning rate which may be adjusted properly to improve the convergence efficiency of the learning course.

二、**Objective**

The goals of the experiment are as follows:

(1) To understand the working of linear perceptron learning algorithm.

(2) To understand the effect of various parameters on the learning rate and convergence of the algorithm.

(3) To understand the effect of data distribution on learnability of the algorithm.

三、**Contents and Procedure**

(1) Design and compile the programme codes of perceptron learning toward the linear classification of two classes. Create a linearly separable pattern dataset with more than 50 samples for each one of two classes. Initialize the weight vector W (0) and select a proper value for the learning rate $\eta \in (0,1]$.Run your programme with your dataset and record the final result, pay attention to the number of iterations for convergence.

首先从 xls 中导入自己写的 100 个数据，分为两组，每组 50 个数据。

```
%读入 xls 中的数据
num=xlsread('data');
len=length(num);
a=num(:,1);
b=num(:,2);
ab=[a,b];
```

X=reshape(ab,len,2); % X 是数据集，Y 是分类结果

c=num(:,3);

Y=reshape(c,1,len);

感知器算法流程：

1） 给出 m 个带有标签的样本{(x1,y1),(x2,y2),...(xm,ym)},其中 yi= -1 or 1 (i=1,2,...m) 是样本 xi = (xi1,xi2,...,xin)的标签;

2) 将数据的标签并入训练样本，形成增广向量，每个数据的维数为 n+1;

3)在(0,1)均匀分布区间内生成 1x(n+1)权值矩阵 W;

4) 将数据依次送入感知器进行学习

如果 W*[data(k,:) yk] <=0 ,说明训练错误，则对权值进行惩罚，

W = W + c*[data(k,:) yk];

否则对权值进行奖励即不惩罚，W = W;

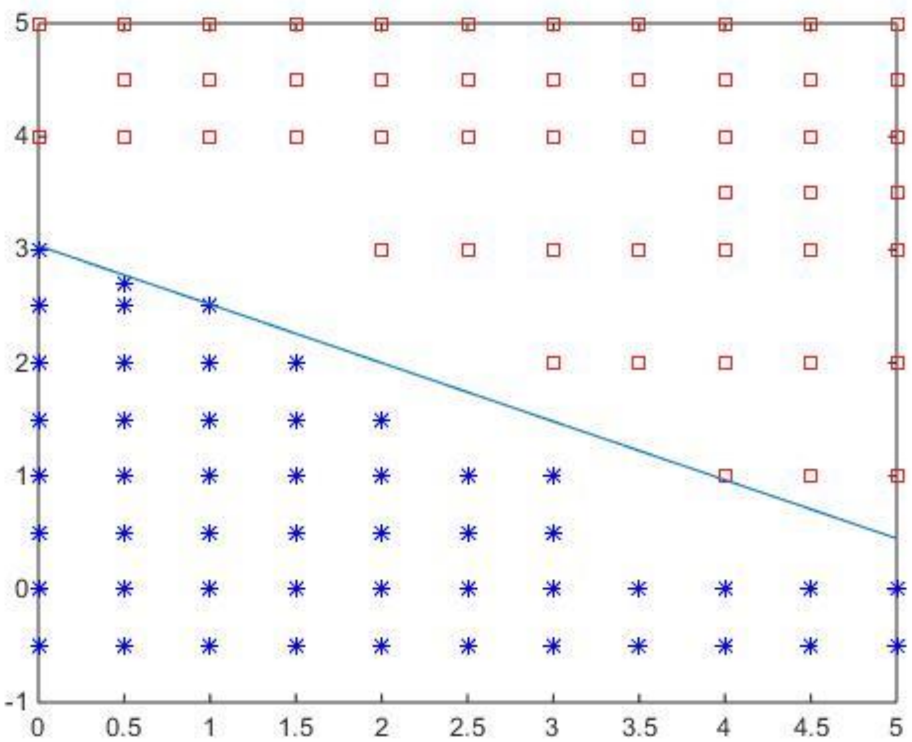
5) 对所有数据训练完成后，如果至少有一个数据训练错误，则要对权值进行重新训练，直到对所有数据训练正确，即可退出训练过程。

```
%感知器算法
maxStep = 500;
learnRate = 1;
[n,m] = size(X);
X = [X ones(n,1)];
W=zeros(m+1,1);
for step = 1:maxStep
    flag = true;
    for index = 1:n
        if sign(X(index,:) * W) ~= Y(index)
            flag = false;
            W = W + learnRate * Y(index) .* X(index,:)';
        end
    end
```

```
        if flag == true
            break;
        end
end
```

分类结果：

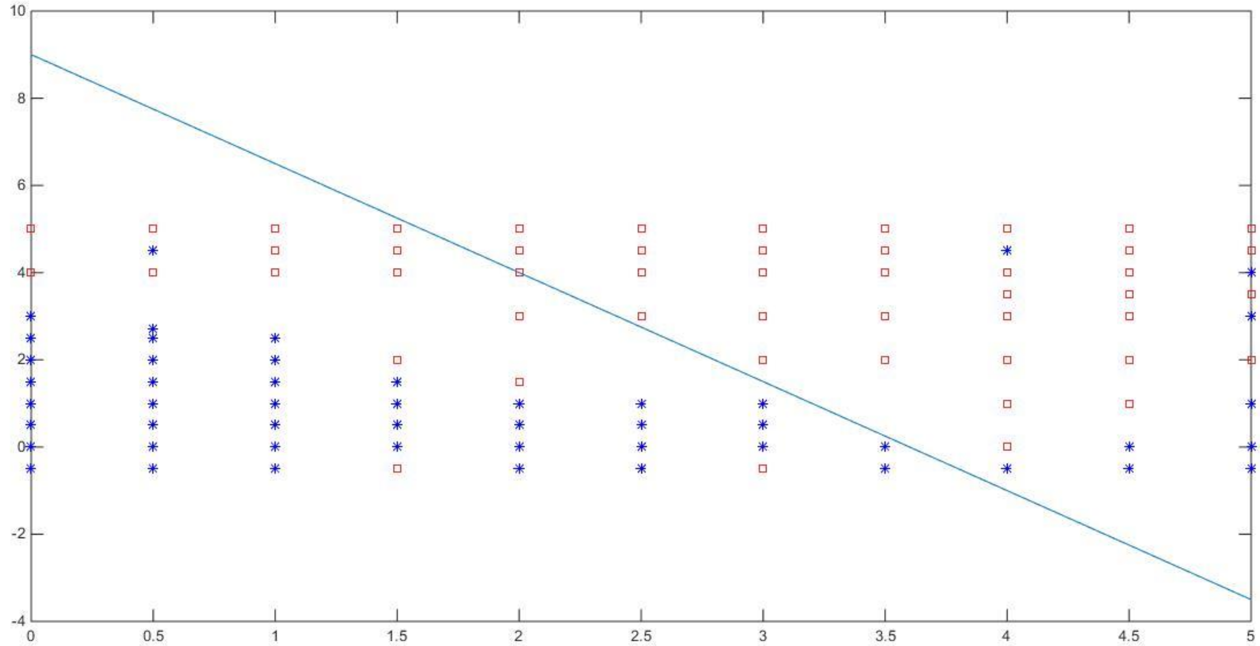
对 100 个数据的分类结果如下：



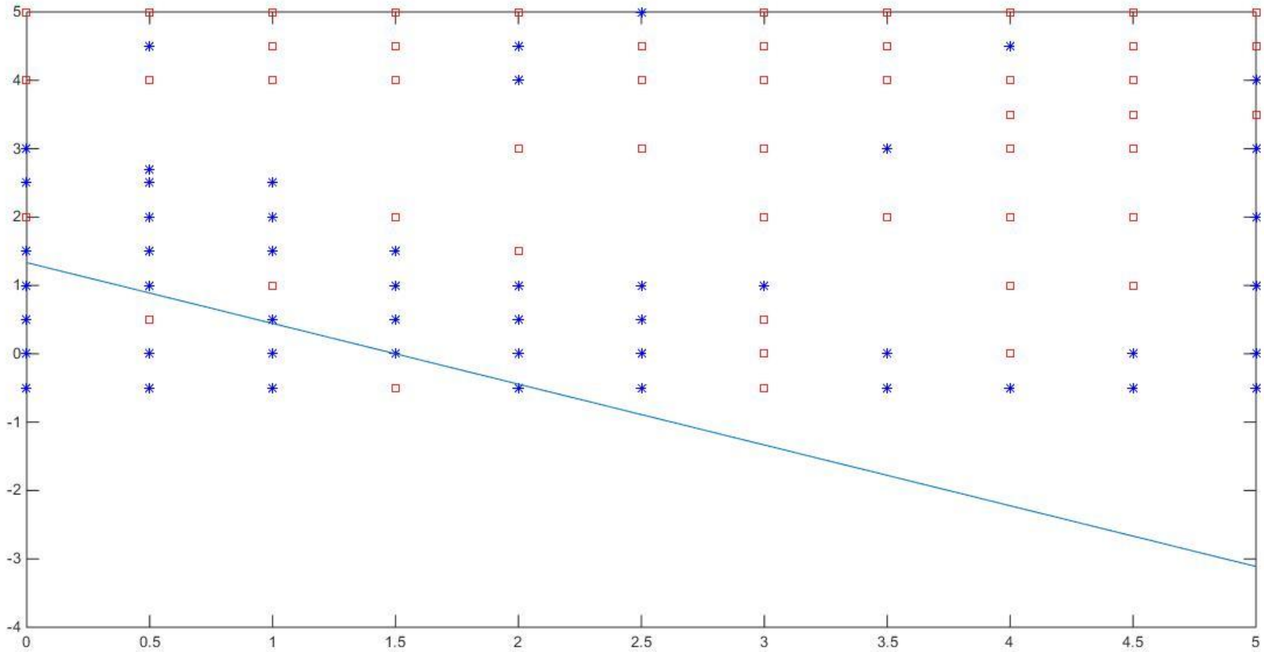
我们可以看到，对于此数据集，分类效果很好，不同的两类数据完美地分布在直线两边。

（2）Repeat the above (1) for the different datasets with varying amount of separation between two classes of patterns. Note down your observations.

将原来两类数据中的五个点改变属性，变成另一类的点，可以看到，分类结果错误率大幅上升，这也说明了感知器线性分类算法的局限性。



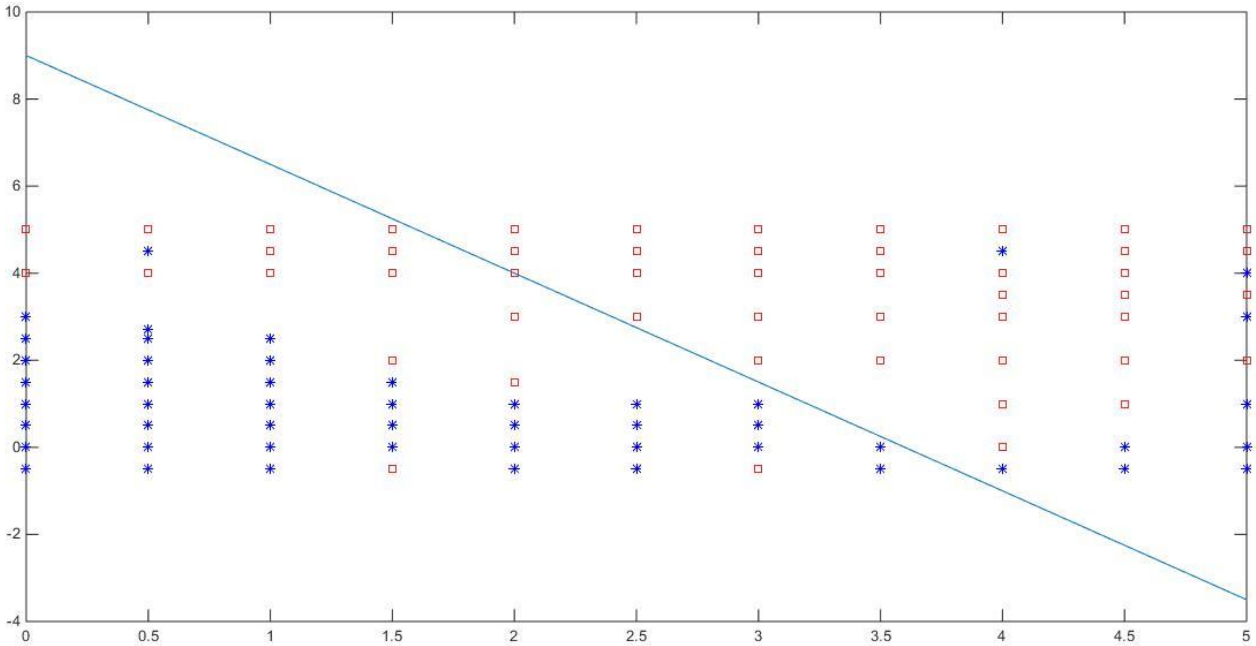
将原来两类数据中的十个点改变属性，变成另一类的点，可以看到，分类结果惨不忍睹，感知器线性分类算法面对这样的数据集已经接近失效。



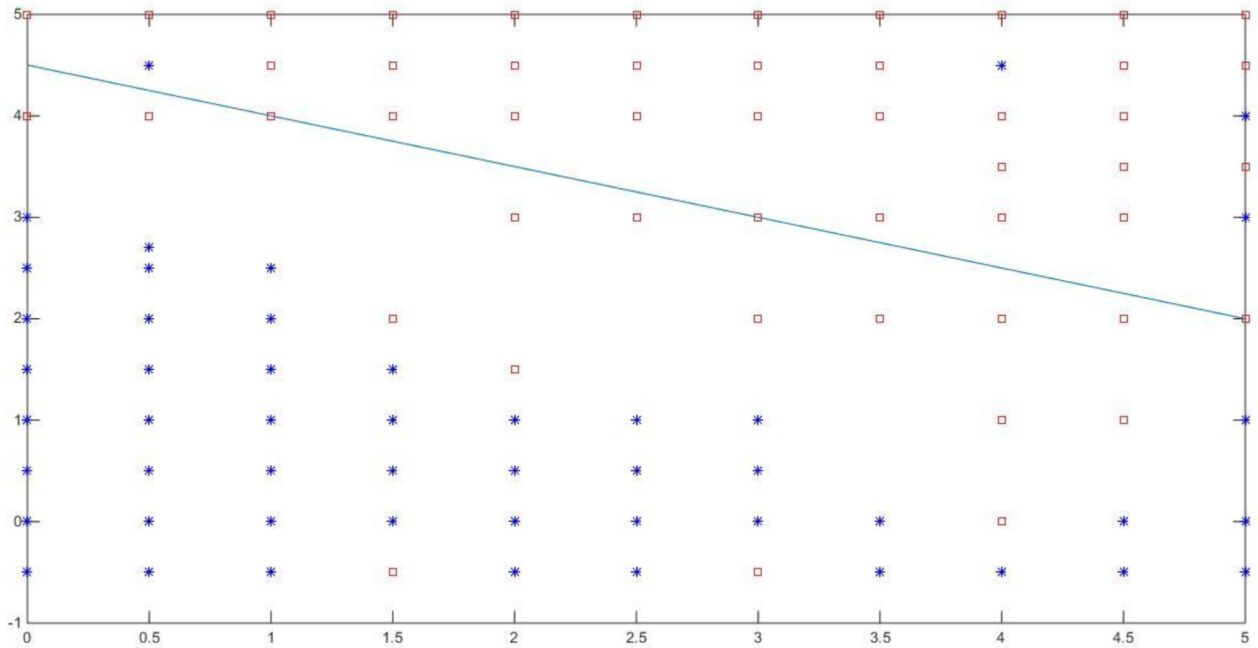
由上面两个例子，我们可以看到，感知器线性分类算法对待干扰较少，边界清晰的数据集能够非常好地工作，但是面对干扰较多，无法用一条直线简单分类时，感知器线性分类算法就工作得不如人意了，甚至无法分类，几乎失效。

（3）Study the effect of varying the learning rate η with different amounts of separability.

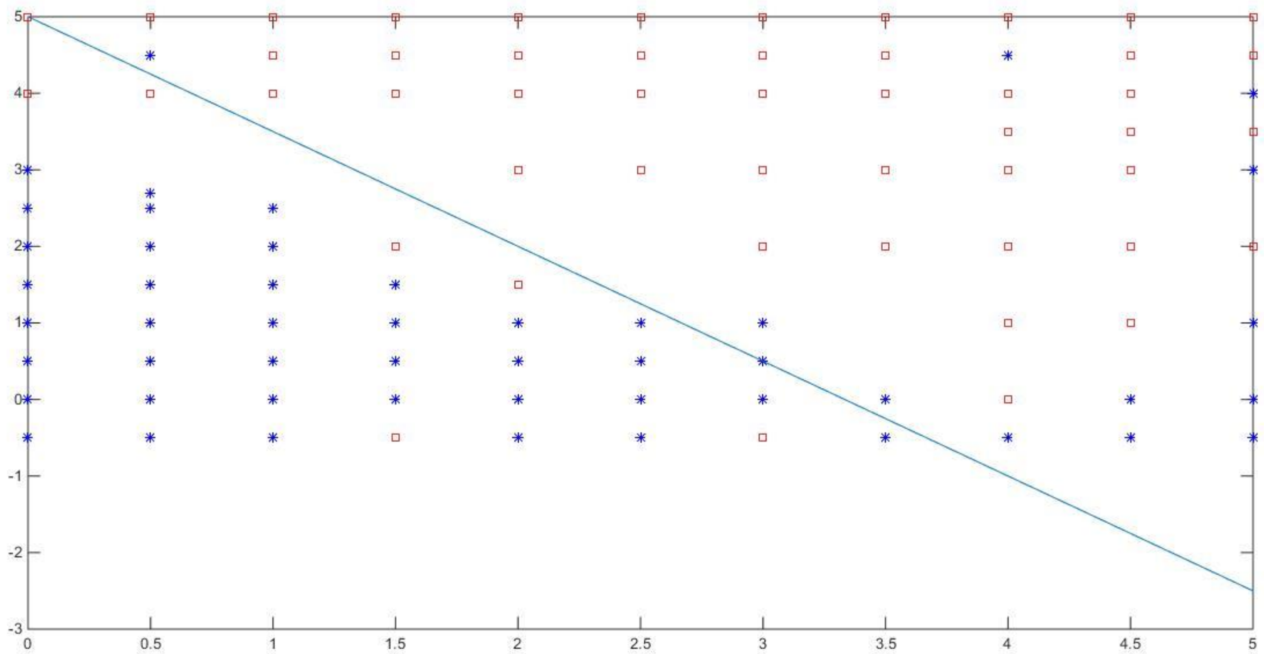
当 η =1 时，感知器线性分类的结果如下图所示。



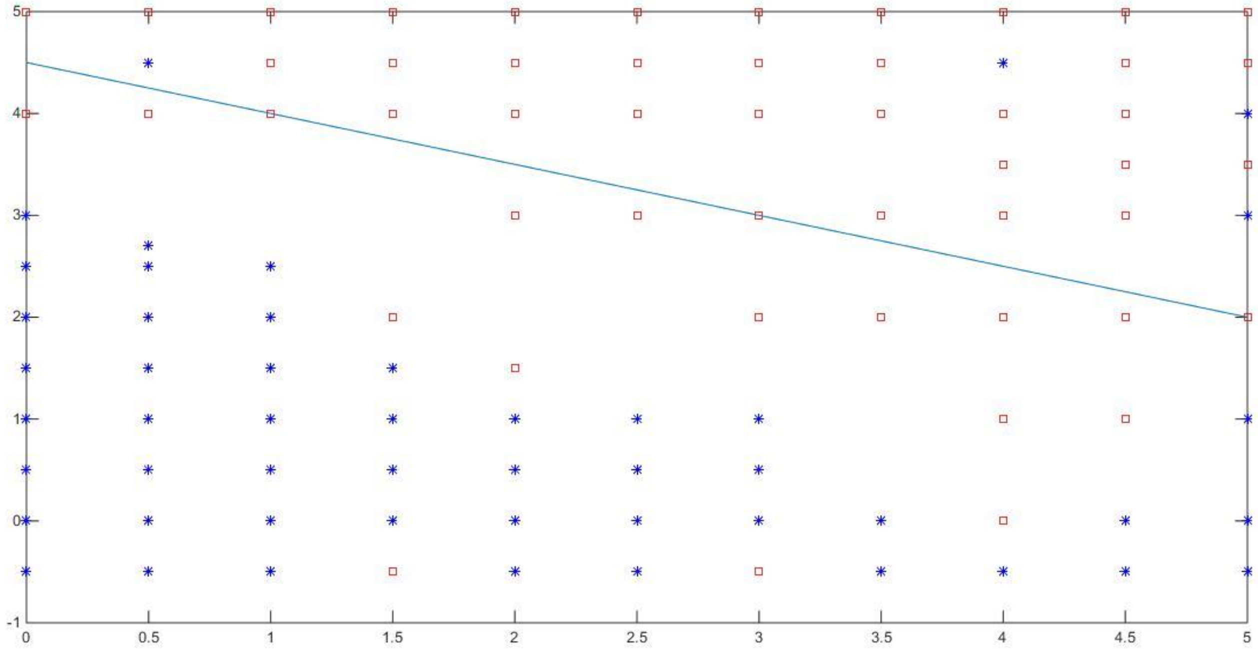
当η=0.8 时，感知器线性分类的结果如下图所示，可以看到，相对η=1 时，分类的效果有不少提升。



η=0.6 时，感知器线性分类的结果如下图所示，可以看到，相对η=0.8 时，分类的效果有不少提升。



η=0.2 时，感知器线性分类的结果如下图所示，可以看到，相对η=0.6 时，分类的效果有下降了。



由上面三个例子可以看到，对于 learning rate η∈(0,1]，对于同一个数据集，分类的效果并不是η越大越好或越小越好，应存在一个大于 0 小于 1 的η，使得数据集能得到最佳的分类效果。