**实验报告**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **实验题目** | 1：线性分类器的设计与实现  2： | | | | |
| **姓名** | 杨晨烨 | **学号** | **2150300135** | **班级** | 电信钱51 |

# 一、问题描述

**题目1：线性分类器的设计与实现**

1. **实验目的**

掌握模式识别的基本概念，理解线性分类器的算法原理。

1. **实验要求**
2. 学习和掌握线性分类器的算法原理;
3. 在 MATLAB 环境下编程实现三种线性分类器并能对提供的数据进行分类;
4. 对实现的线性分类器性能进行简单的评估(例如算法适用条件，算法效率及复杂度等)。
5. **实验原理**

**判别函数**

判别函数（discriminant function）是指由的各个分量的线性组合而成的函数：

这里是权向量，是阈值权或偏置。

**两分类器**

对于有如上形式判别函数的两类线性分类器来说，要求：如果则判定，如果则判定。方程定义了一个判定面，它把归类于的点与归类于的点分开来。当是线性的，这个平面被称为超平面。

**广义判别函数**

给定训练样本集合

求线性判别函数

即求满足所有约束的向量：

做规范化处理，设：

则有：

也就是说，对于正确分类的样本，；对于错误分类的样本，。

**两类线性可分情况**

对于一个包含个样本的集合，一些标记为，另一些标记为，希望用这些样本确定一个判别函数的权向量。

等式确定了一个穿过权空间原点的超平面，为其法向量。解空间必须在每一个超平面的正侧。也就是说，解向量如果存在，一定在N个正半空间的交叠区，而且该区中的任意向量都是解向量。

对于这样的解空间中的解向量来说，离边界近的解很危险，因为容易把新来的测试样本分类错。这个问题有两种解决思路，一是引入约束，让最小距离最大化，找到中间位置，二是引入间隔。

寻找满足线性不等式的解时所采用的方法是：定义一个准则函数，当是解向量时，为最小。这样问题被转化为标量函数极小化问题，可以使用梯度下降法解决。

**感知器准则函数**

这里的是被错分的样本集，如果没有样本被错分，就是空的，这时我们定义。从几何上可知，与错分样本到判决边界距离之和成正比。

**算法原理**

1. 固定增量批处理感知器算法

1. **Begin** **initialize**
2. **do**
4. **until**
5. **return**
6. **end**

1. 固定增量单样本感知器算法

1. **Begin** **initialize**
2. **do** mod
3. **if** is misclassified by **then**
4. **until** all samples are classified correctly
5. **return**
6. **end**

1. Widrow Hoff最小均方差实现算法

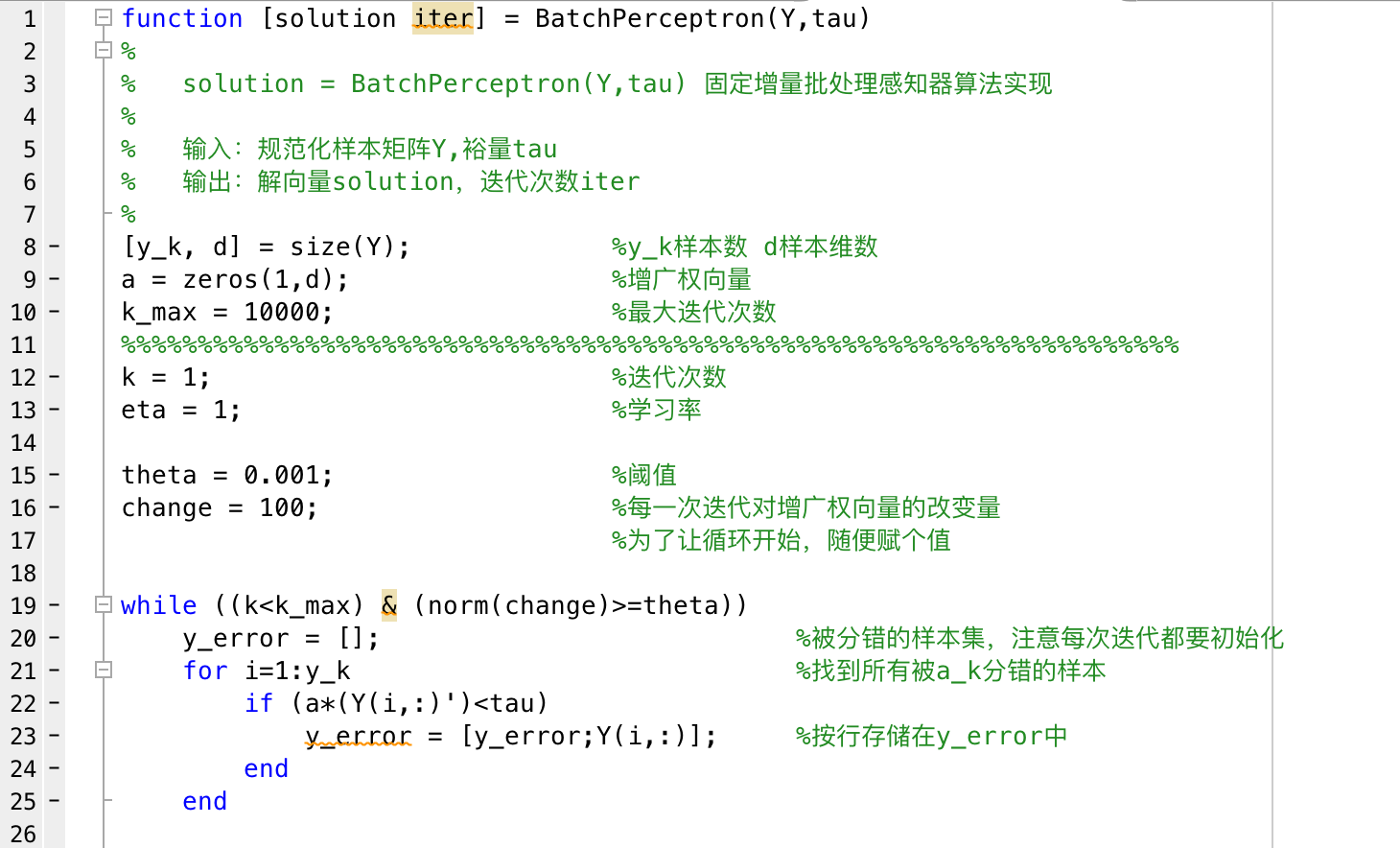
1. **Begin** **initialize**
2. **do** mod
4. **until**
5. **return**
6. **end**

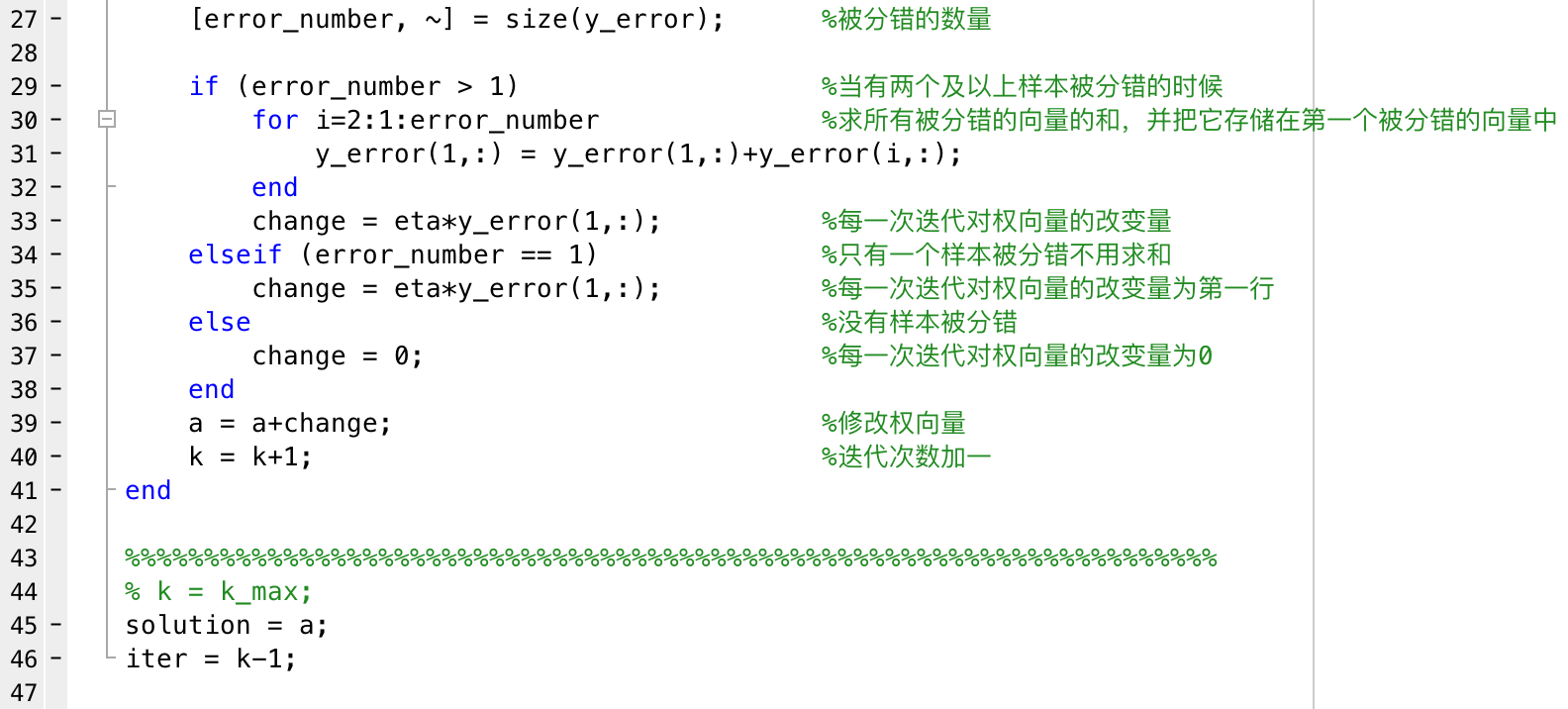
**题目2：**

# 二、复现代码（Matlab或Pyhton或其他）

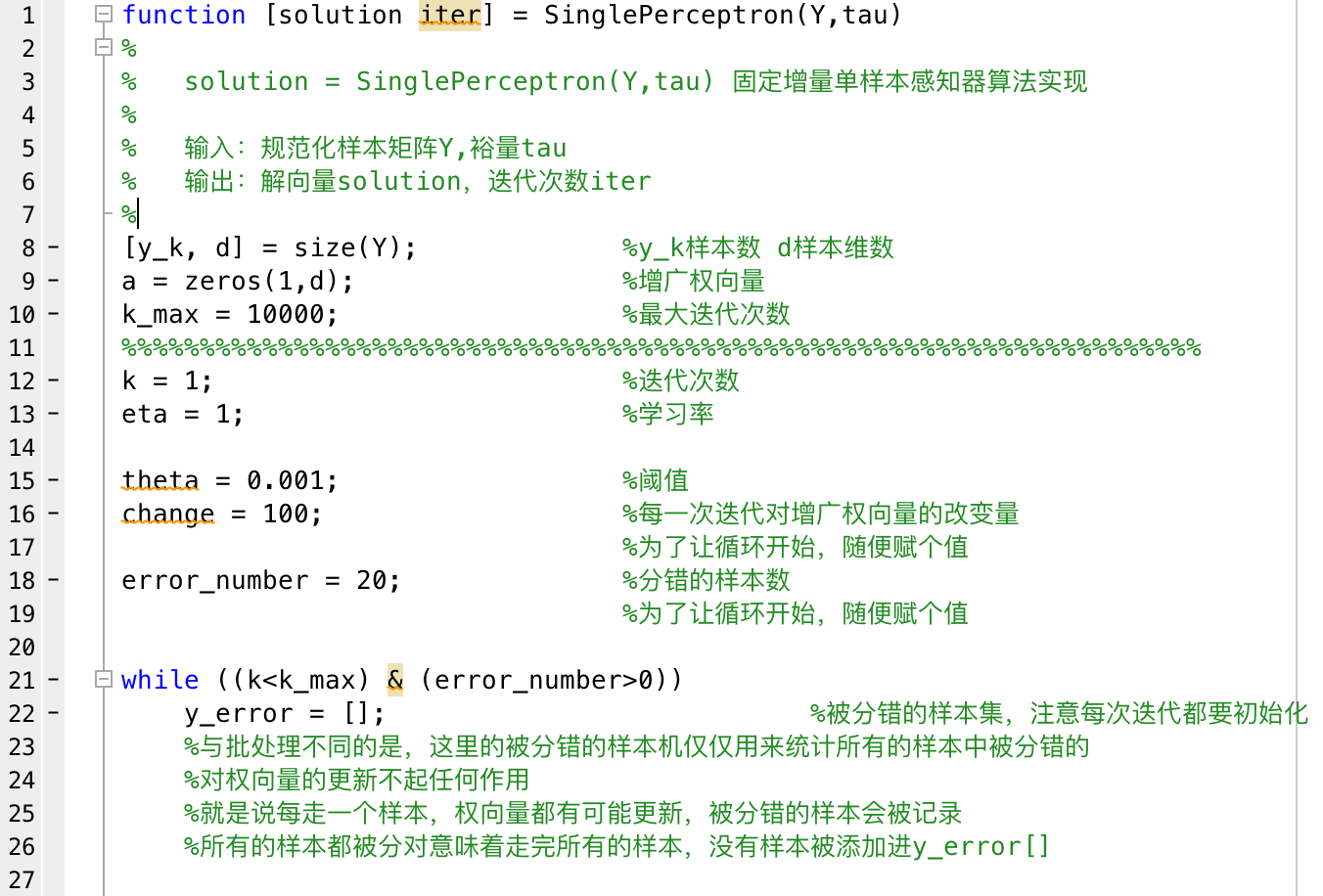
**题目1：线性分类器的设计与实现**

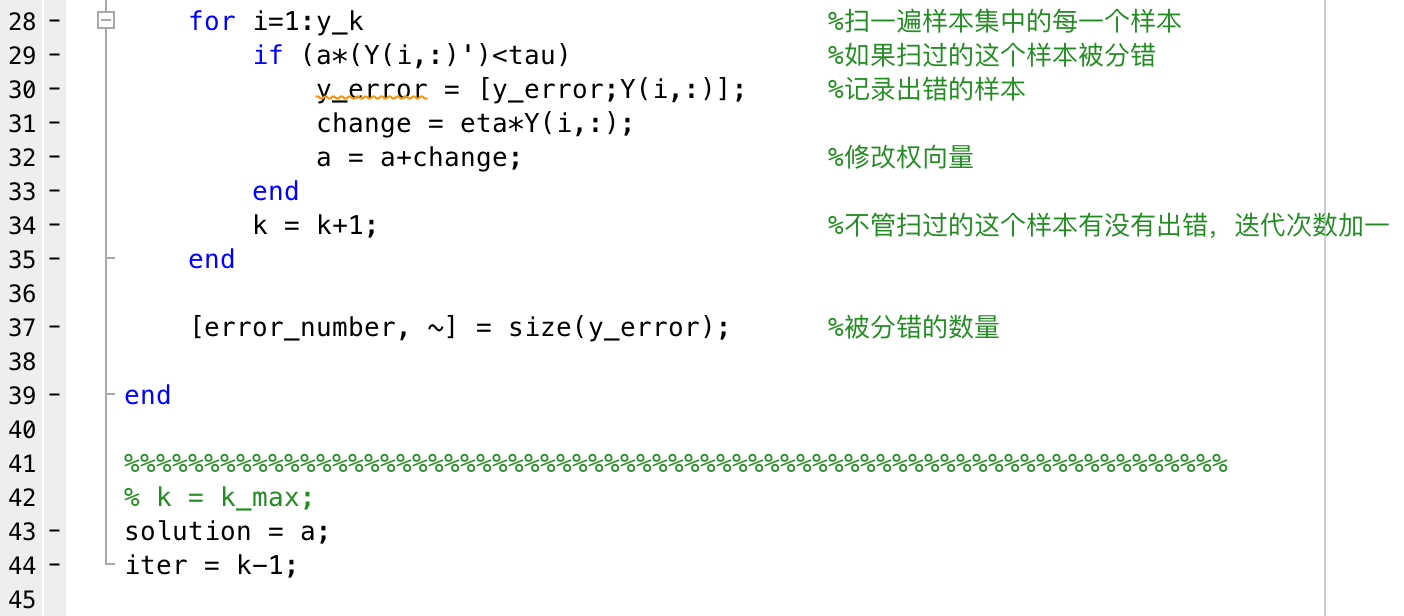
1. 固定增量批处理感知器算法



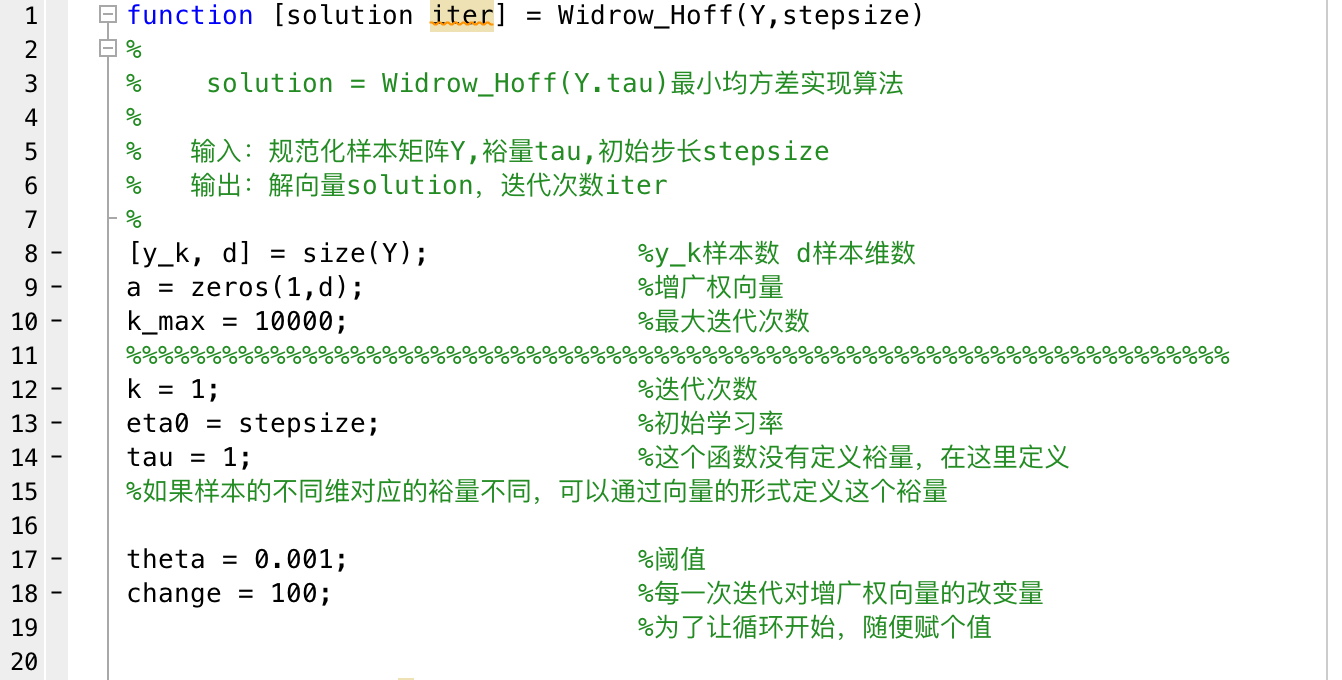


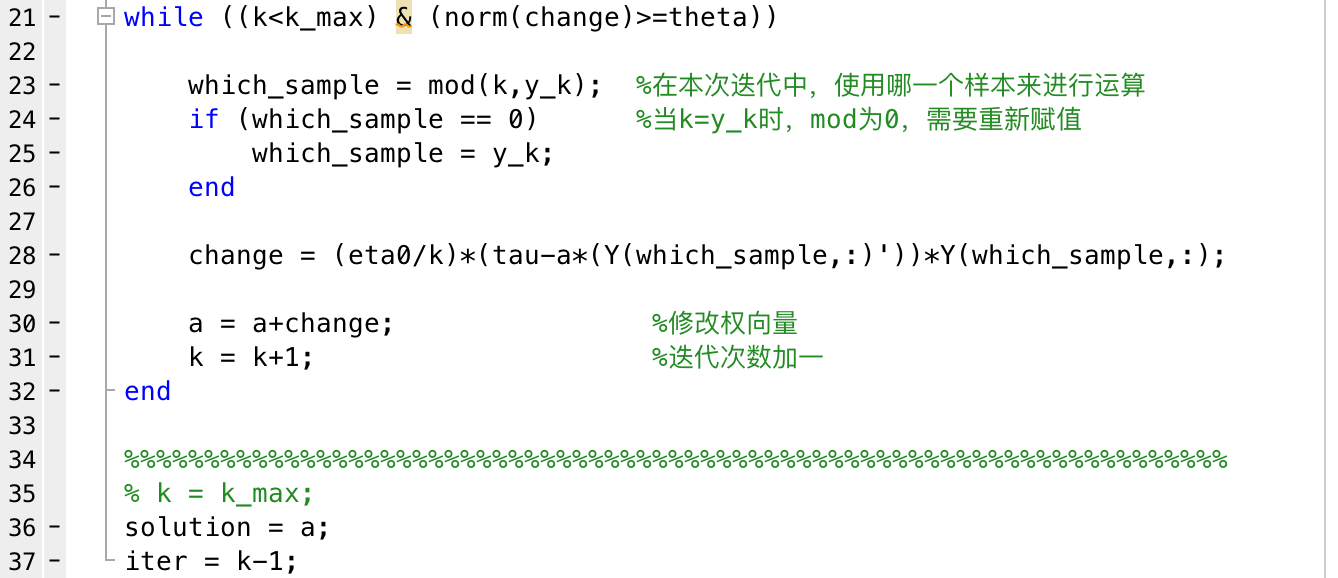
1. 固定增量单样本感知器算法





1. Widrow Hoff最小均方差实现算法



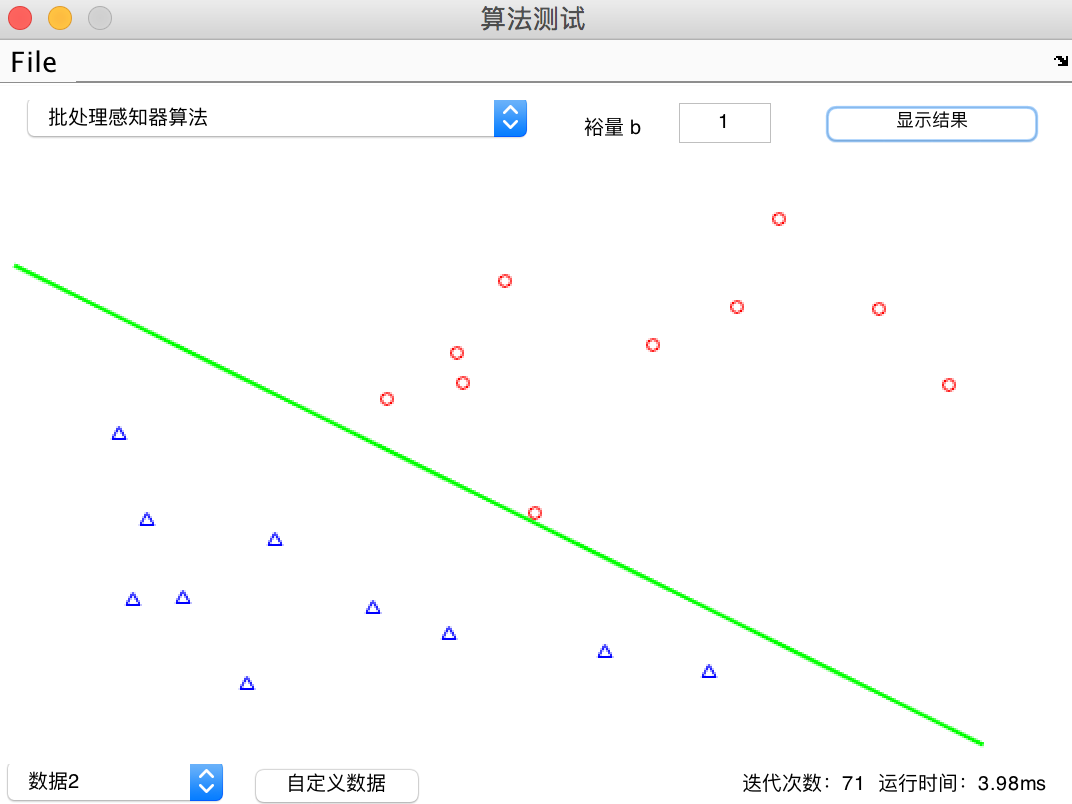
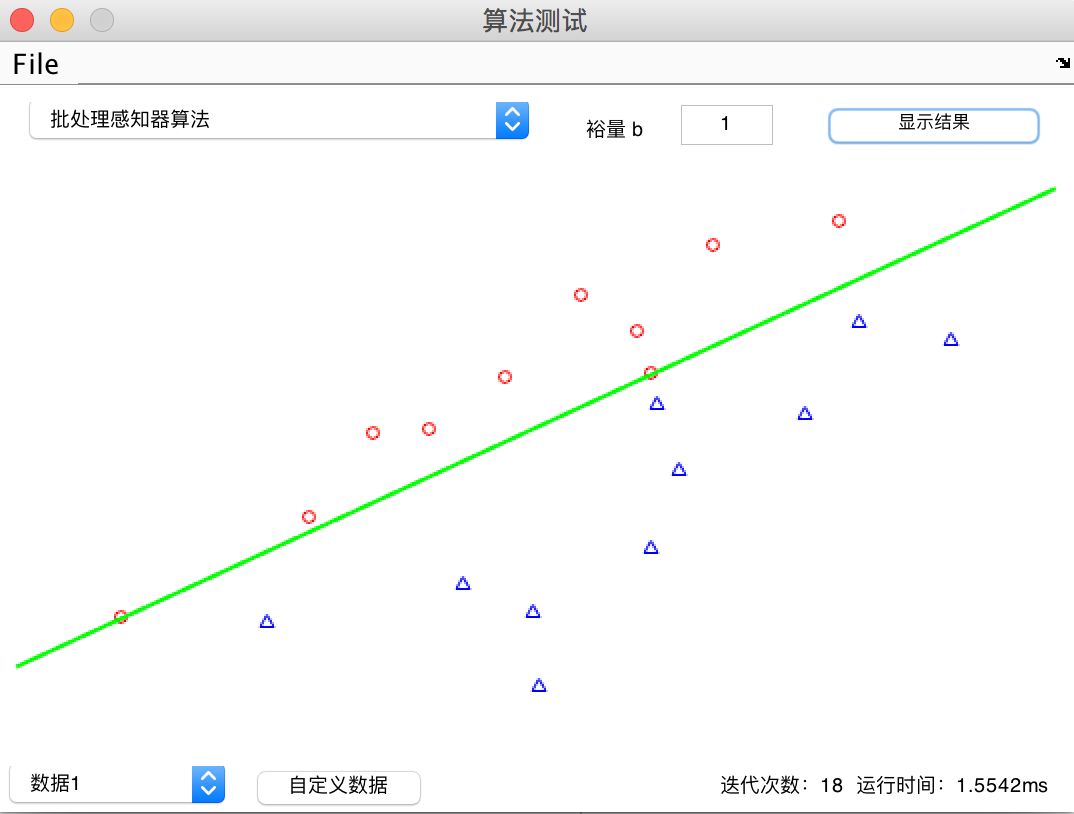


**题目2：**

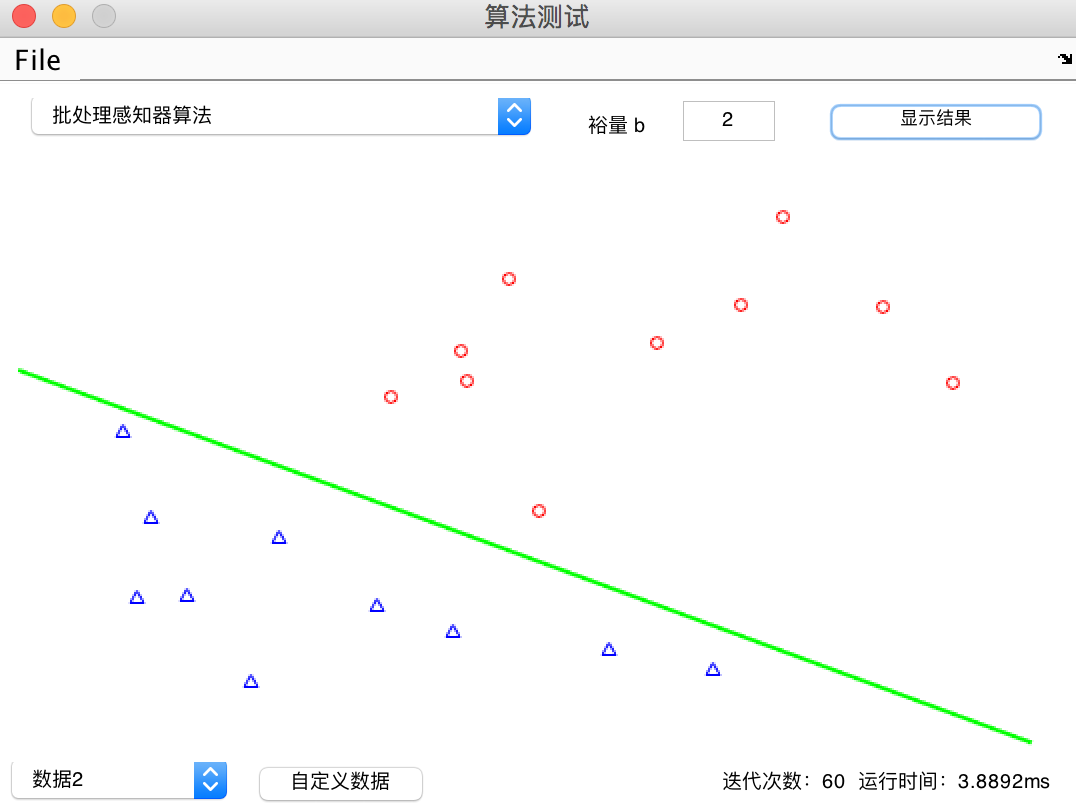
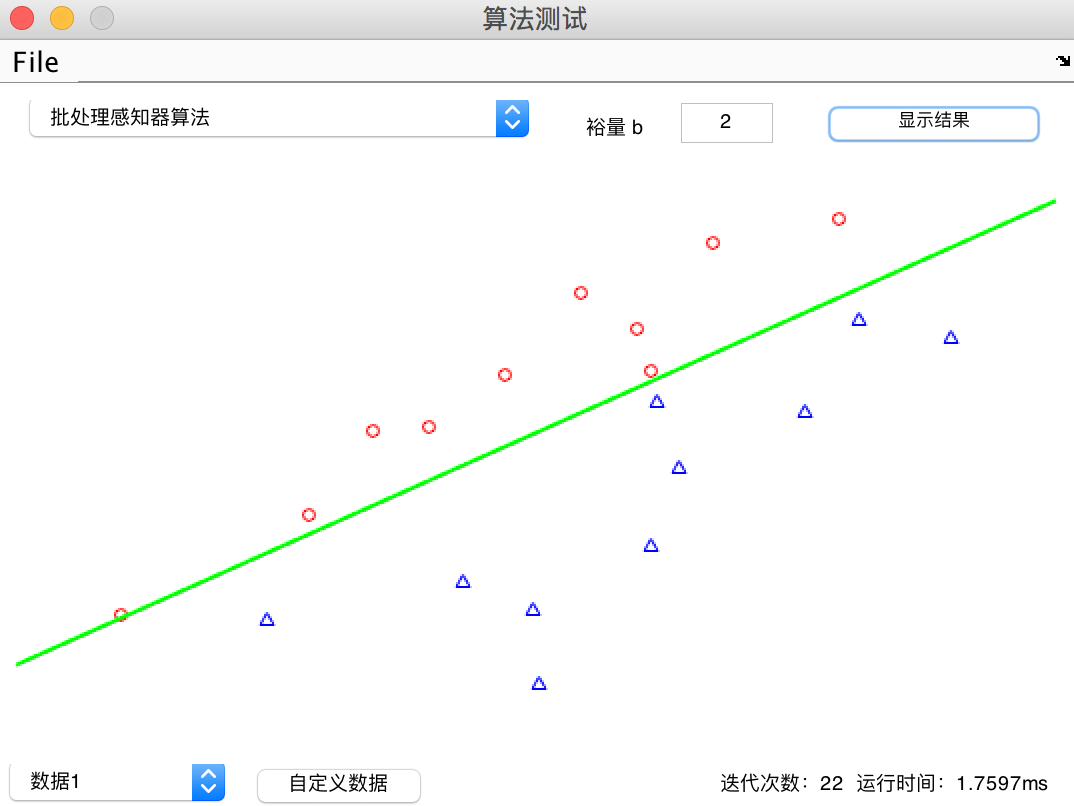
# 三、结果分析

**题目1：**

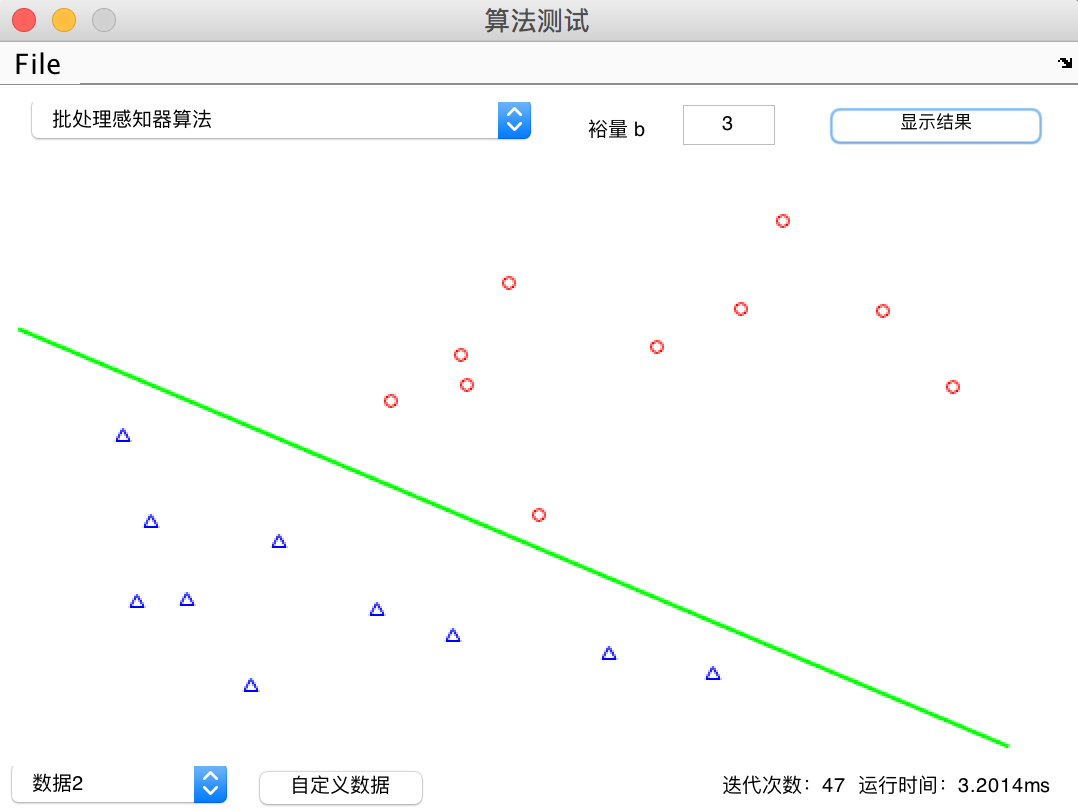
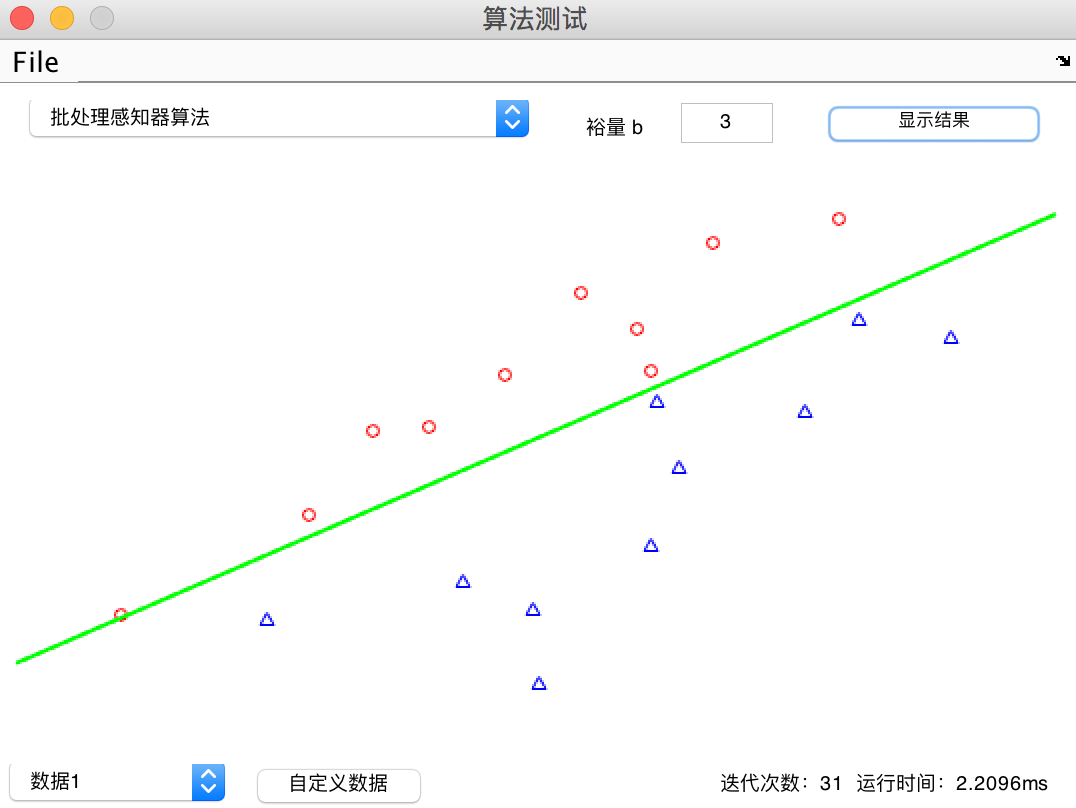
1. 固定增量批处理感知器算法



a) 数据1，b=1 b) 数据2，b=1



c) 数据1，b=2 d) 数据2，b=2



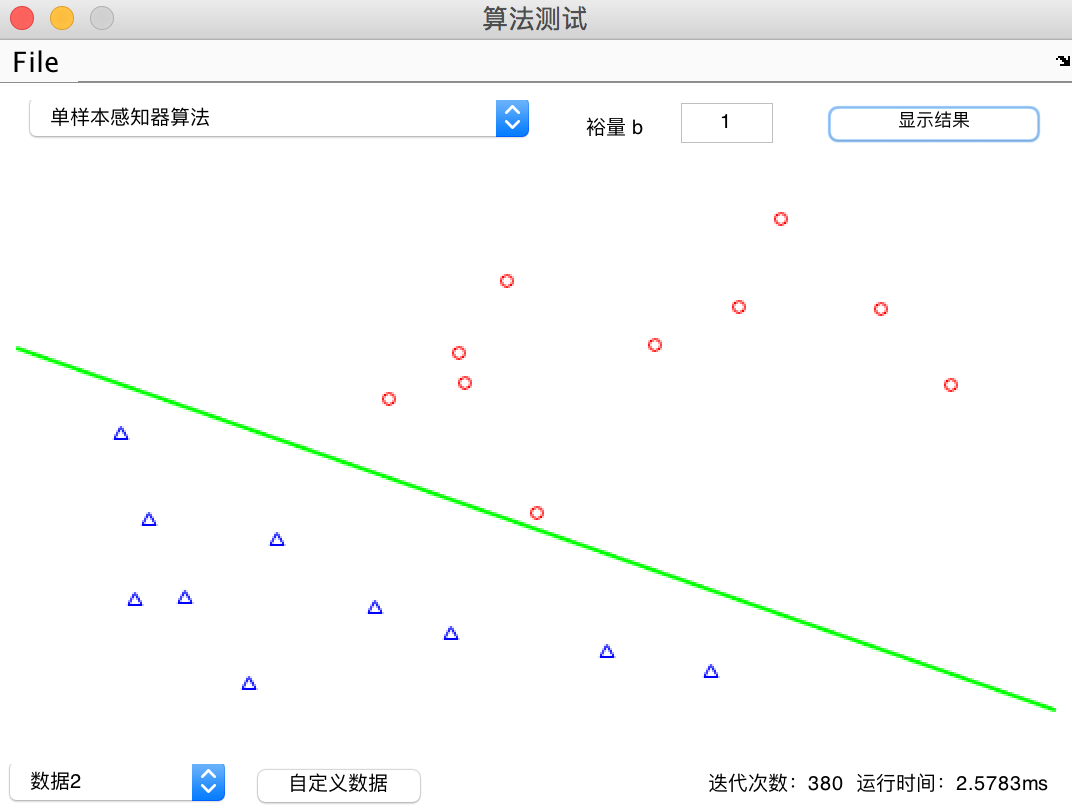
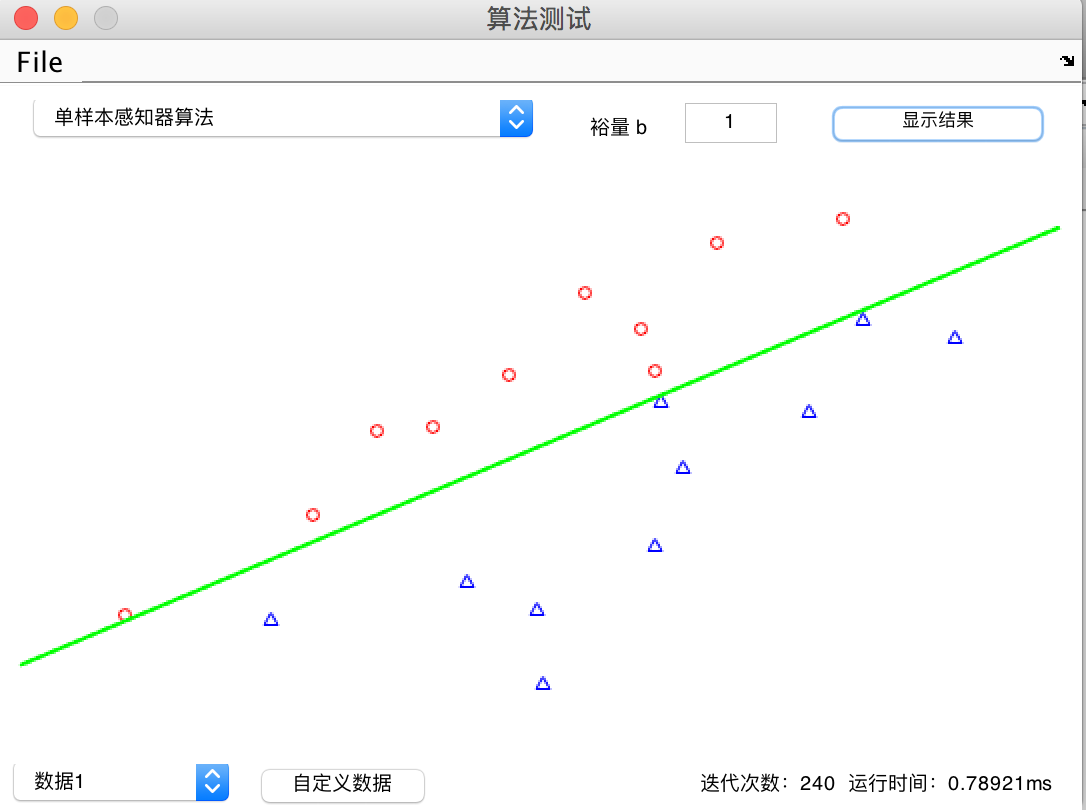
e) 数据1，b=3 f) 数据2，b=3

图1 固定增量批处理感知器

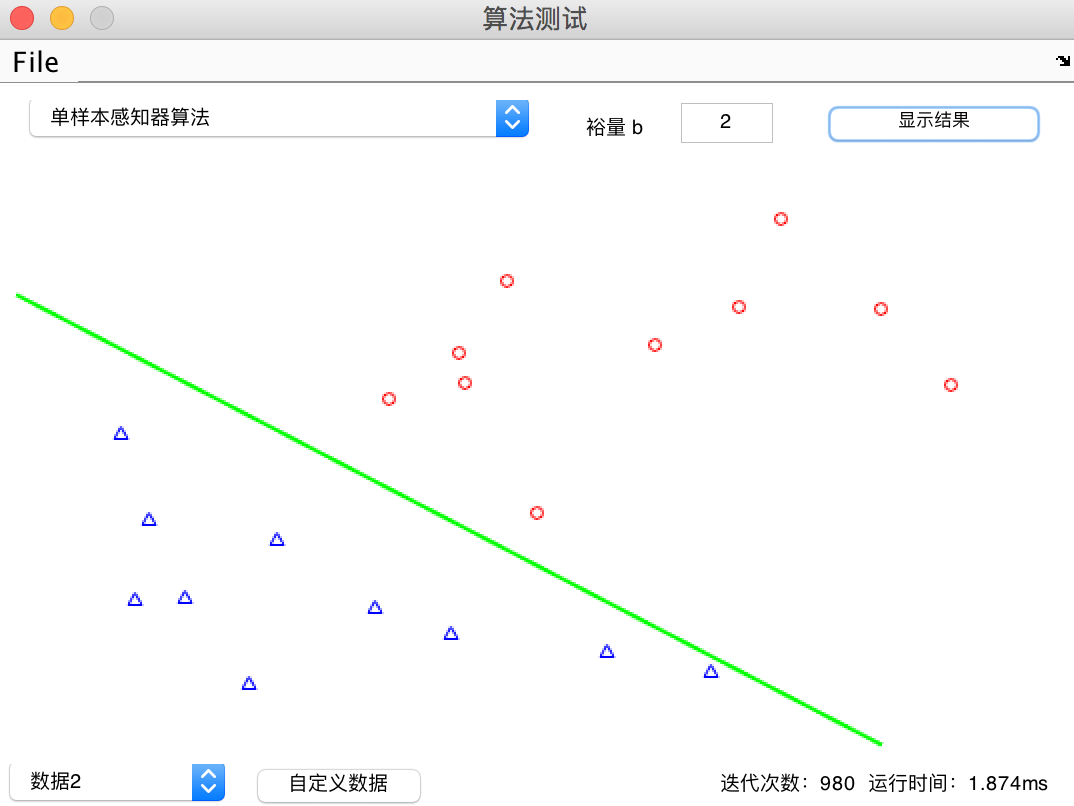
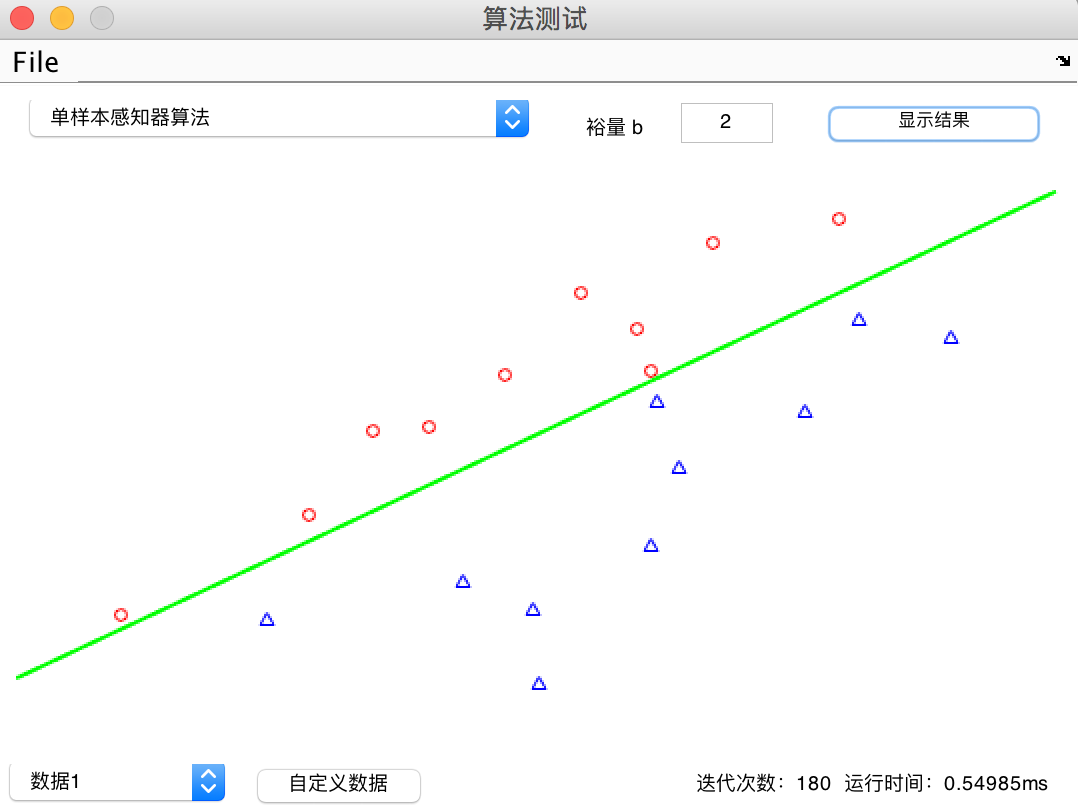
我们可以看到，固定增量批处理感知器可以较好地分类这两类样本，迭代次数均为几十次，运行时间在1-3ms之间，可以认为算法复杂度不算高。

算法中设置的裕量越大，分类效果越好，分类面离最靠近它的点越远，分类面越处在两类的中间位置。

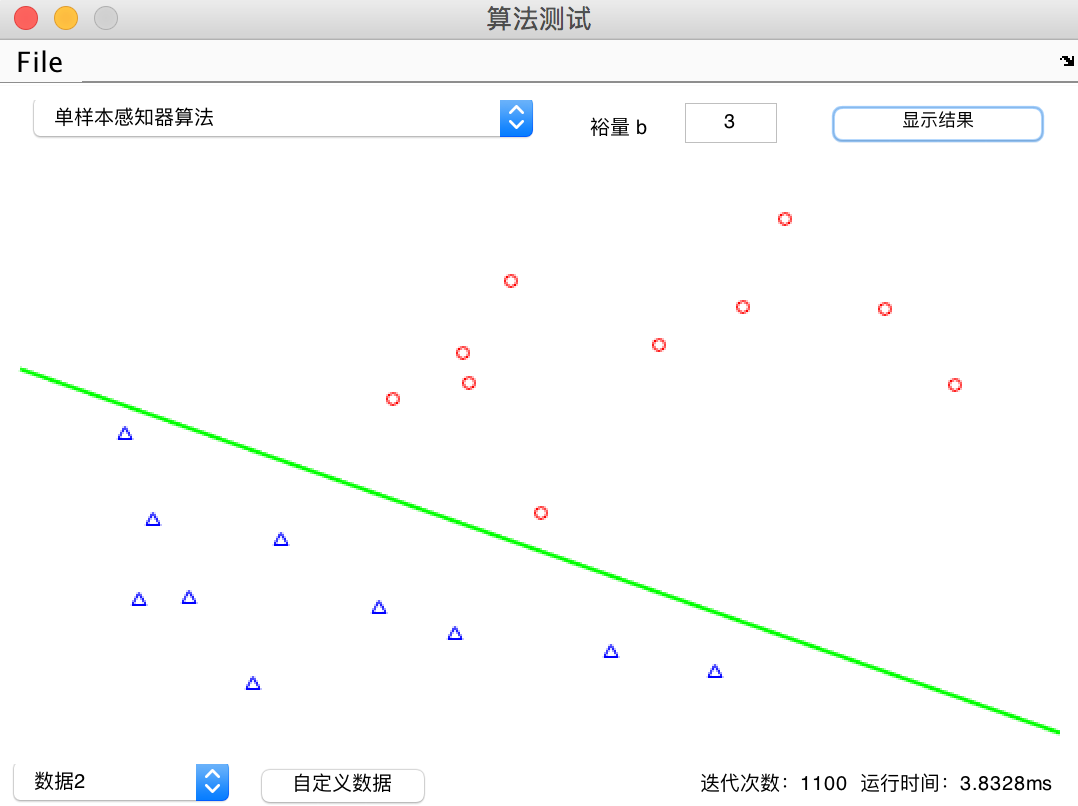
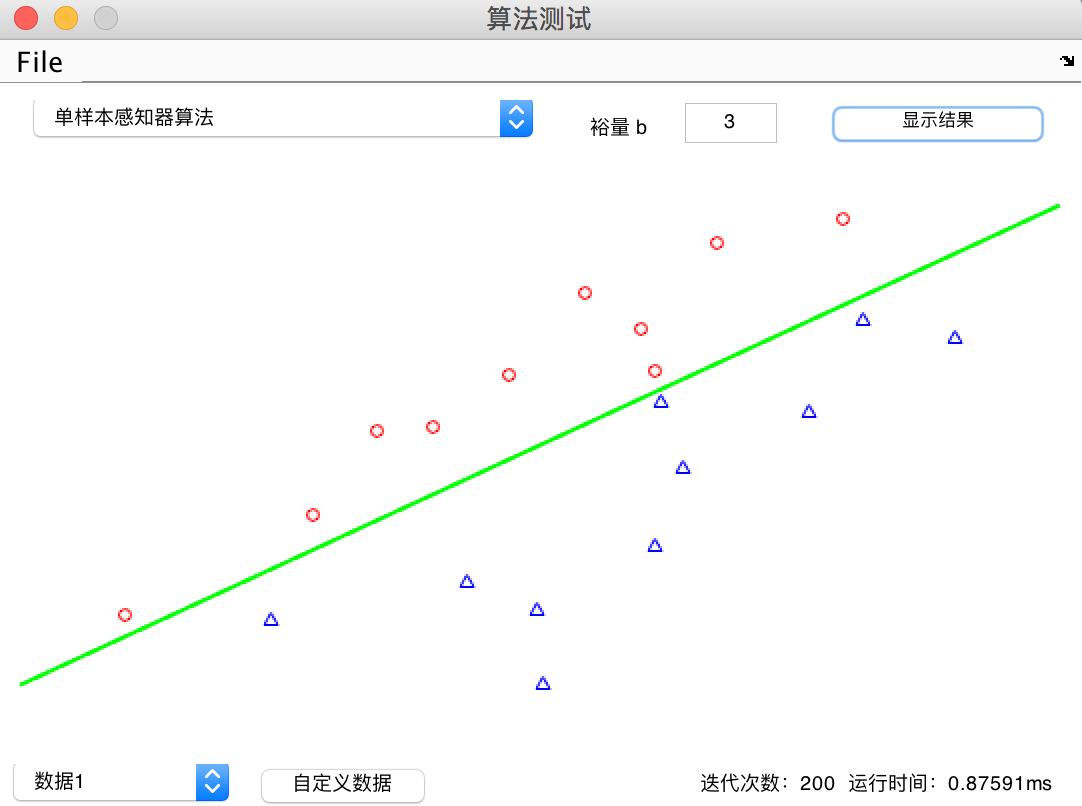
1. 固定增量单样本感知器算法



a) 数据1，b=1 b) 数据2，b=1



c) 数据1，b=2 d) 数据2，b=2



e) 数据1，b=3 f) 数据2，b=3

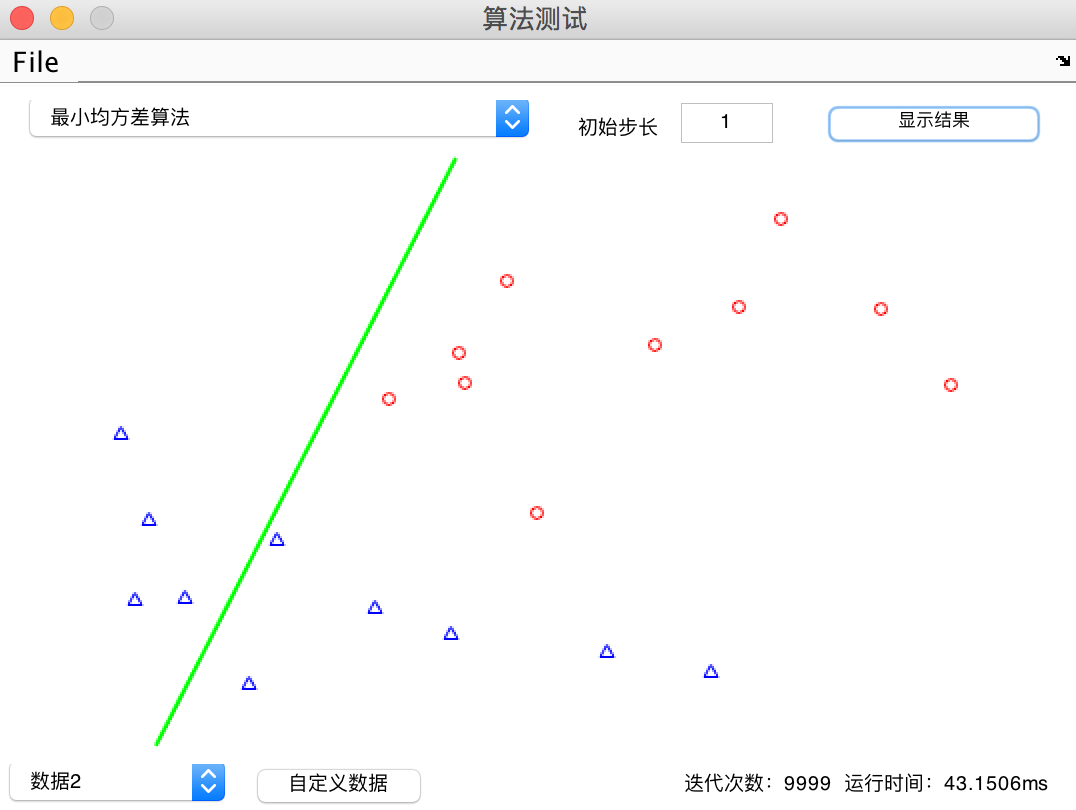
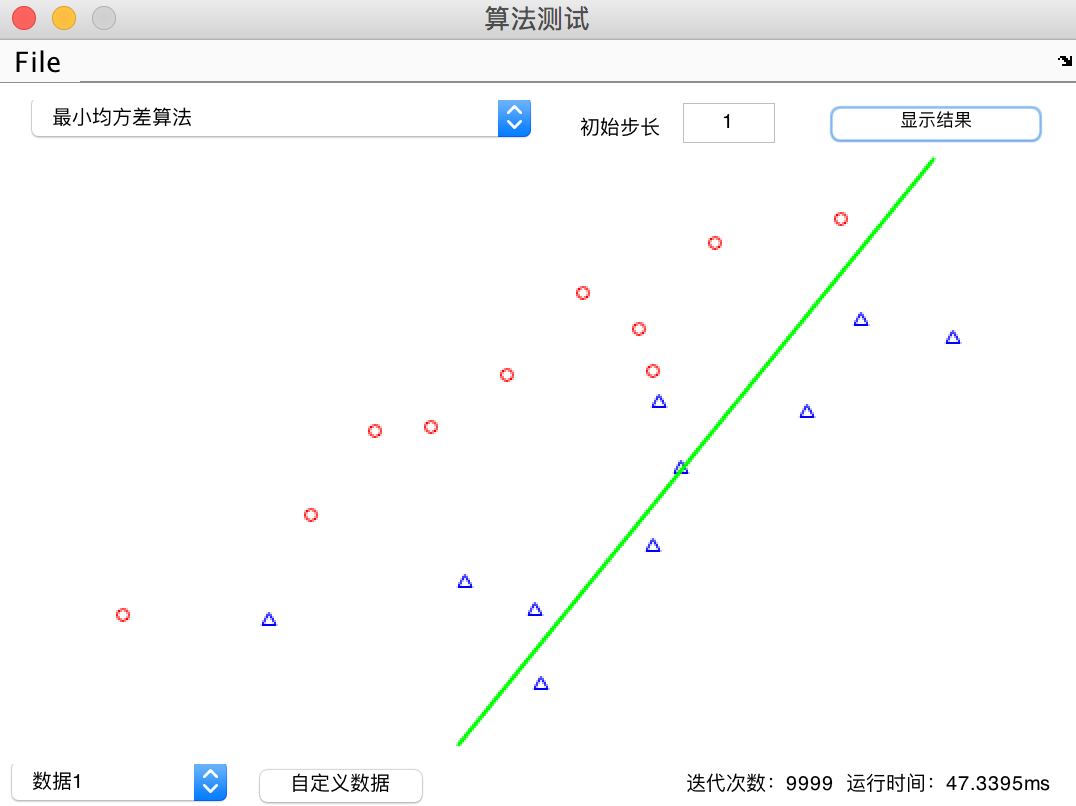
图2 固定增量单样本感知器

我们可以看到，固定增量单样本感知器可以较好地分类这两类样本，迭代次数在300-1100次之间，运行时间在0.5-3.8ms之间，可以认为算法复杂度不算高。

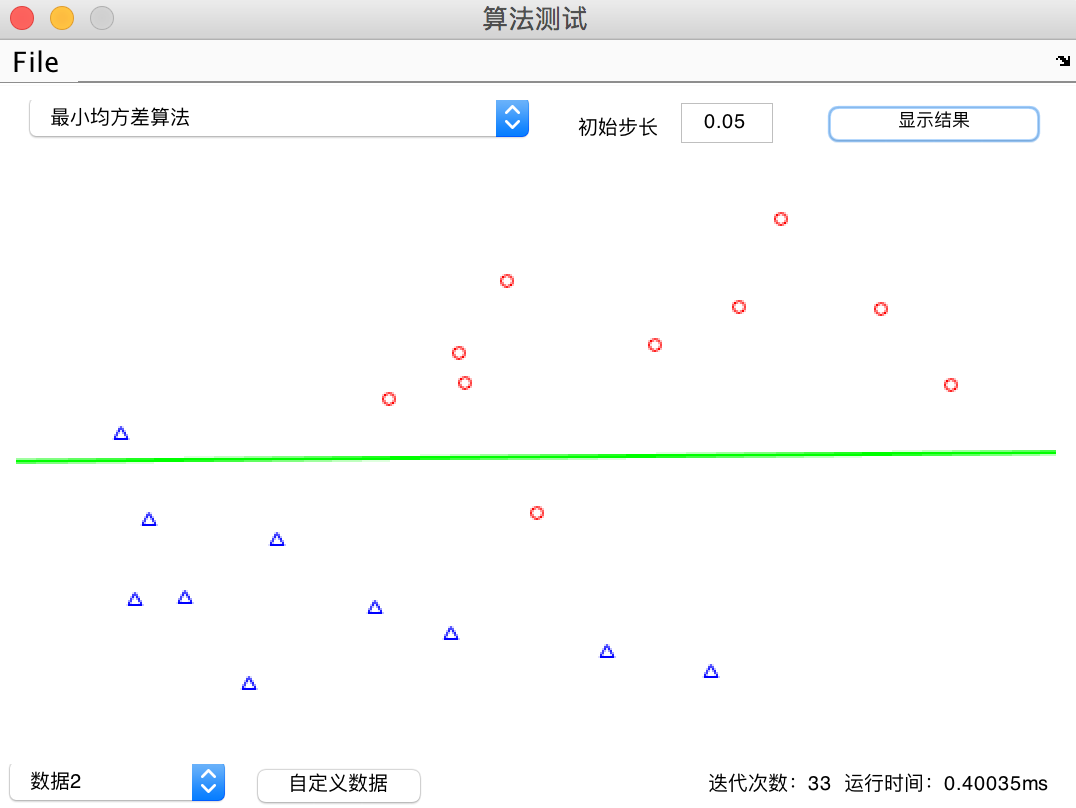
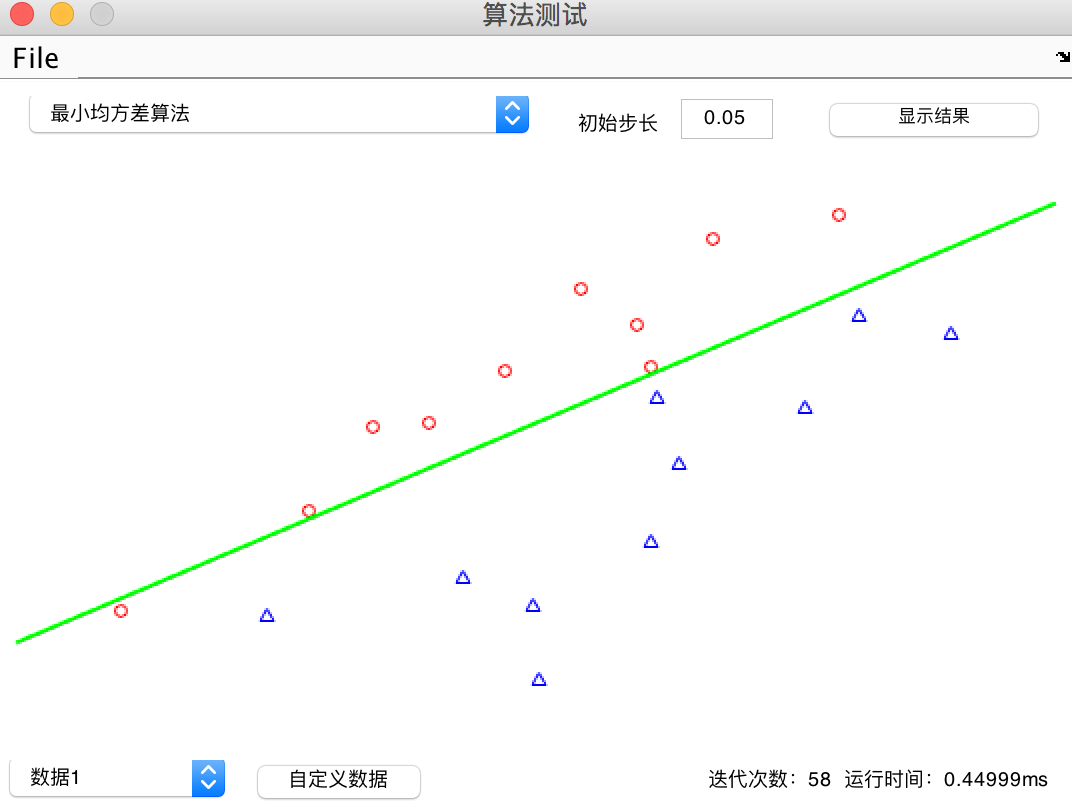
与固定增量批处理感知器算法相比，虽然单样本算法普遍迭代次数多，但是运行时间两者相差并不大。这是因为批处理算法中一次迭代里存在更多的向量加法与数值存储，而这需要更多的时间开销。

算法中设置的裕量越大，分类效果越好，分类面离最靠近它的点越远，分类面越处在两类的中间位置。

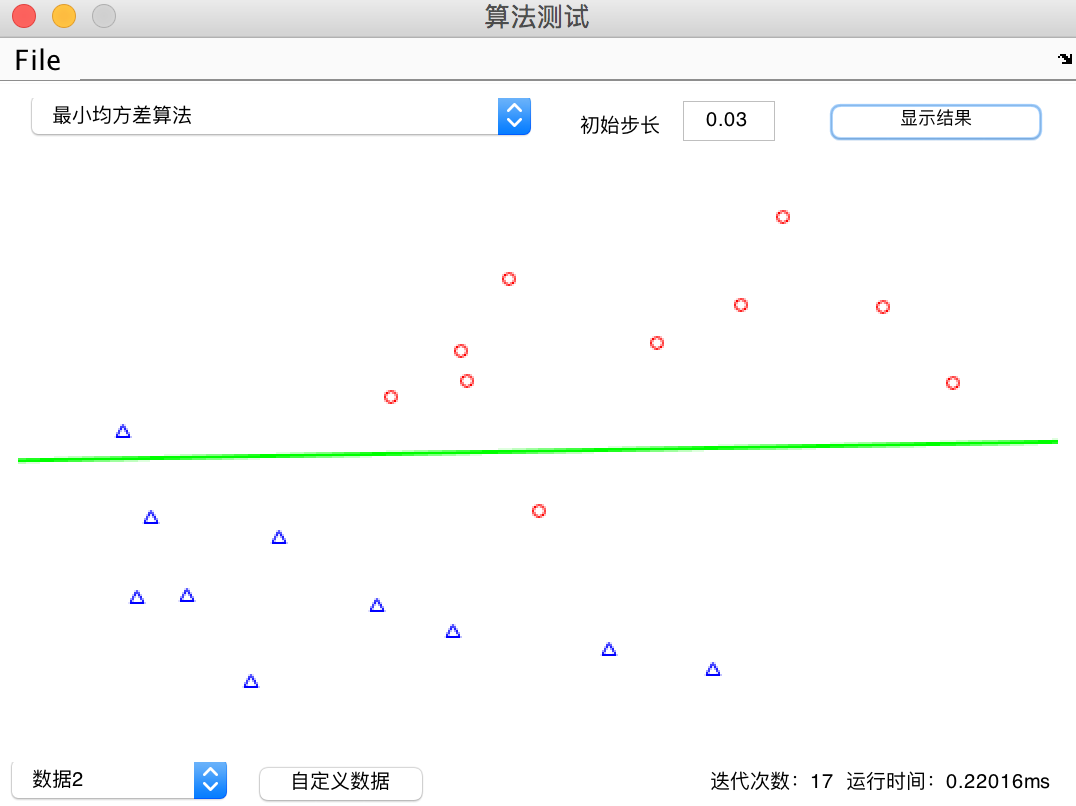
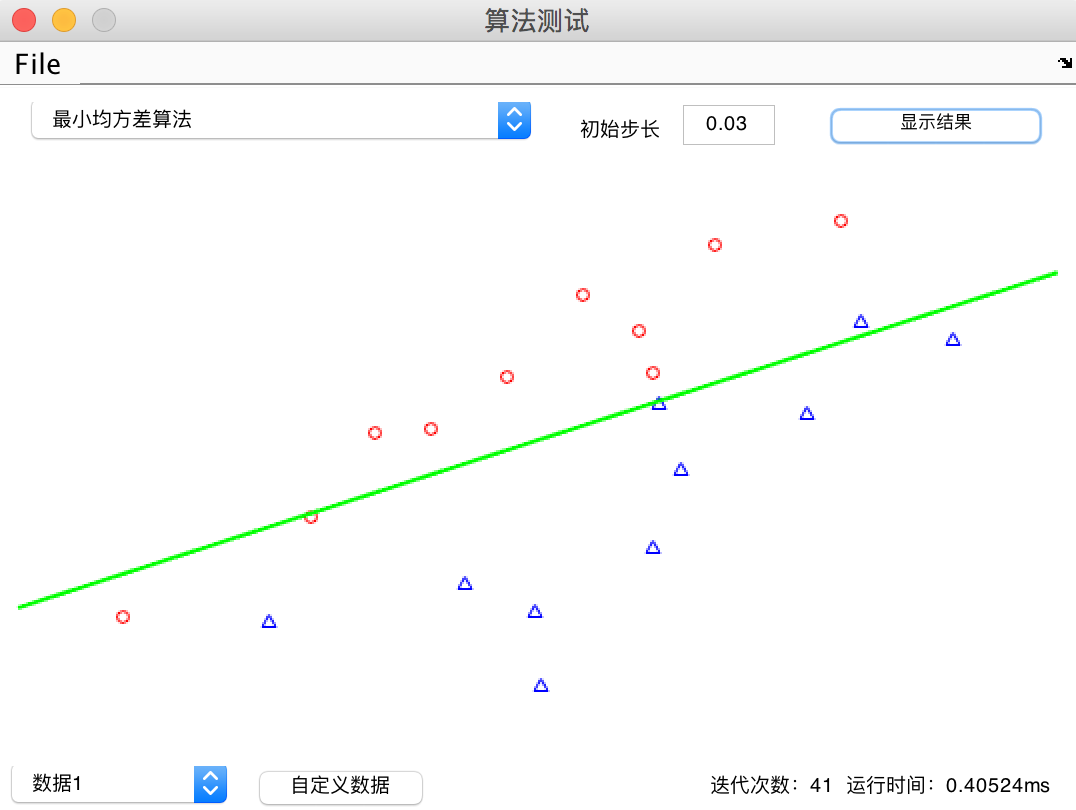
1. Widrow Hoff最小均方差实现算法



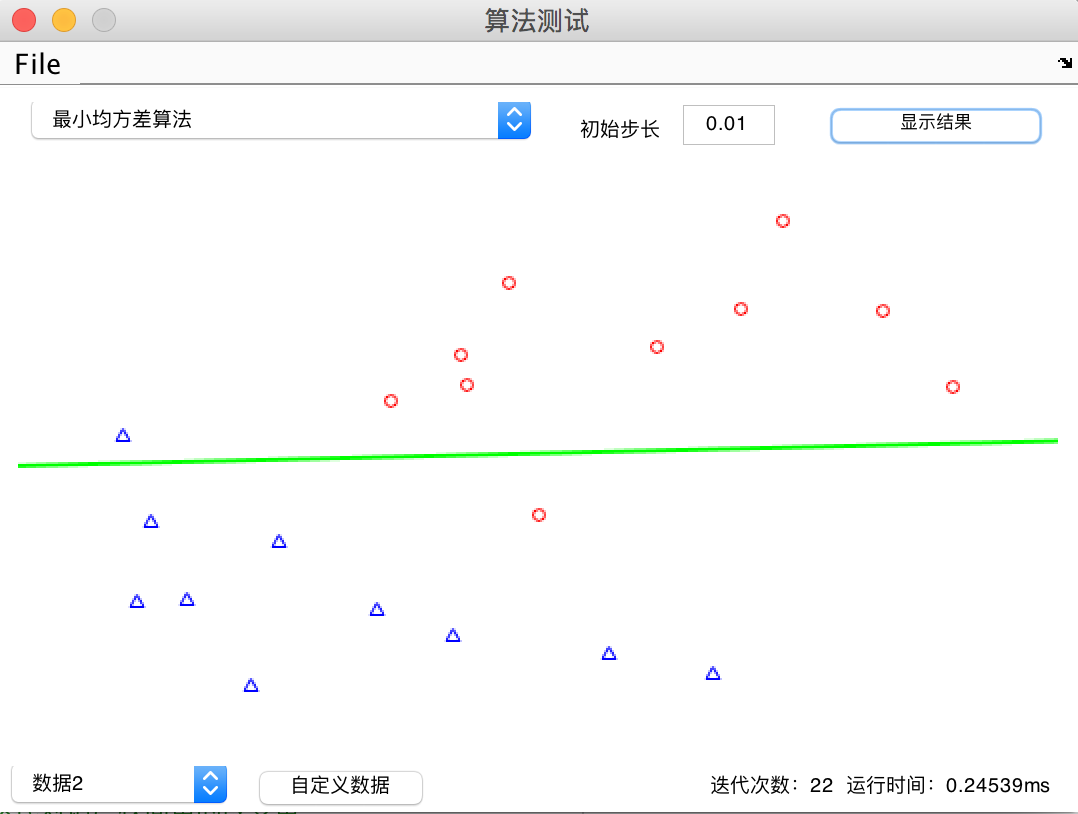
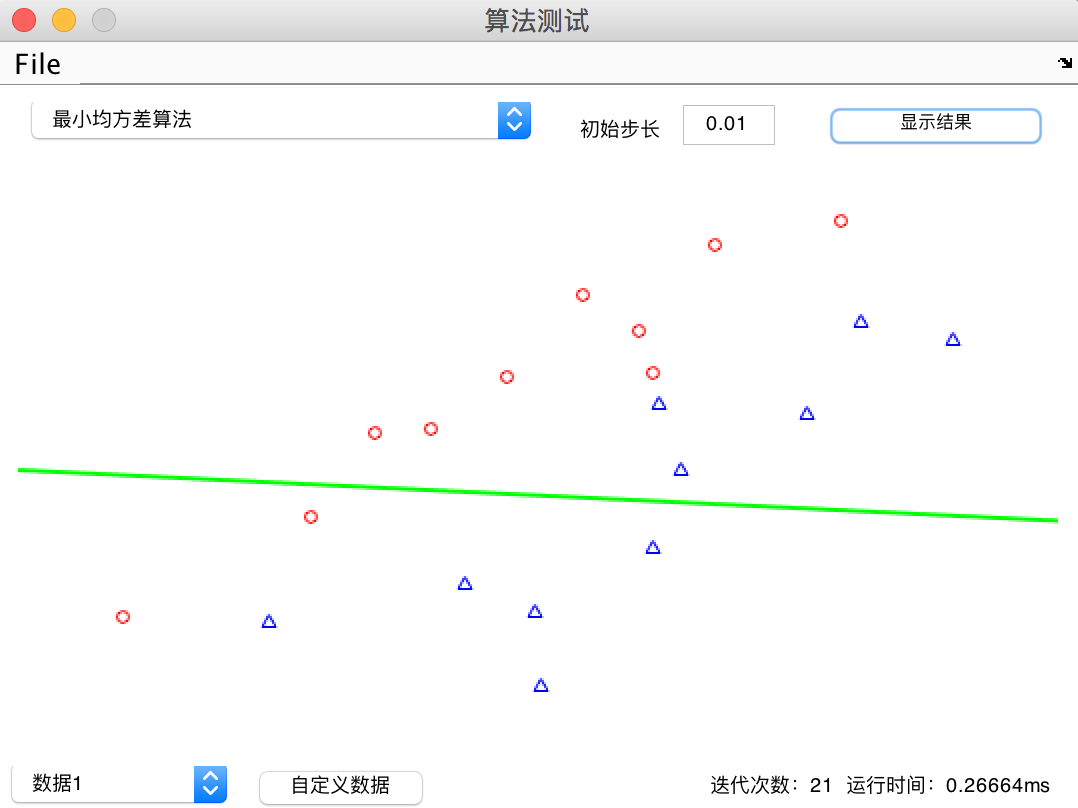
a) 数据1，eta0=1 b) 数据2，eta0=1



c) 数据1，eta0=0.05 d) 数据2，eta0=0.05



e) 数据1，eta0=0.03 f) 数据2，eta0=0.03



g) 数据1，eta0=0.01 h) 数据2，eta0=0.01

图3 Widrow Hoff最小均方差

观察实验结果，可以发现，Widrow Hoff算法对初始步长（初始学习率）的要求很高，当初始步长的选取不合适时，算法无法收敛，将一直运行直到达到最大迭代次数。而步长选取合适时，算法将很快收敛，需要的迭代次数和运行时间明显小于固定增量批处理感知器和固定增量单样本感知器，时间复杂度最低，算法性能好。

Widrow Hoff的初始步长需要在一个合适的值（如上图3 c d e f）附近，过大或过小都不合适，过大（如上图3 a b）会导致无法收敛，过小（如上图3 g h）则算法虽然收敛但是训练得到的分类面并不能将样本合理分类。

Widrow Hoff会有错分的样本点。

**题目2：**

# 四、实验总结

与固定增量批处理感知器算法相比，虽然单样本算法普遍迭代次数多，但是运行时间两者相差并不大。这是因为批处理算法中一次迭代里存在更多的向量加法与数值存储，而这需要更多的时间开销。

算法中设置的裕量越大，分类效果越好，分类面离最靠近它的点越远，分类面越处在两类的中间位置。

Widrow Hoff算法对初始步长（初始学习率）的要求很高，当初始步长的选取不合适时，算法无法收敛，将一直运行直到达到最大迭代次数。而步长选取合适时，算法将很快收敛，需要的迭代次数和运行时间明显小于固定增量批处理感知器和固定增量单样本感知器，时间复杂度最低，算法性能好。