

山东大学机器学习课程

实验报告

——实验八：以adaboost为例的集成学习的设计与实现

姓名：刘梦源

学院：计算机科学与技术学院

班级：计算机14.4

学号：201400301007

**一、实验目的：**

（1）学习集成学习的思想

（2）学习adaboost的算法原理

（3）根据已给数据集，编写代码完成adaboost分类器

（4）体会adaboost集成学习的优势

**二、实验环境：**

（1）硬件环境：

英特尔® 酷睿™ i7-7500U 处理器

512 GB PCIe® NVMe™ M.2 SSD

8 GB LPDDR3-1866 SDRAM

（2）软件环境：

Windows10家庭版64位操作系统

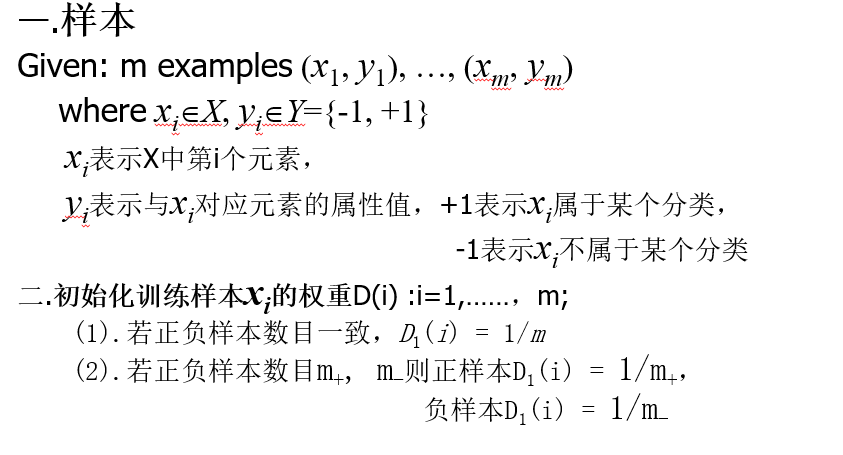
Matlab R2016a

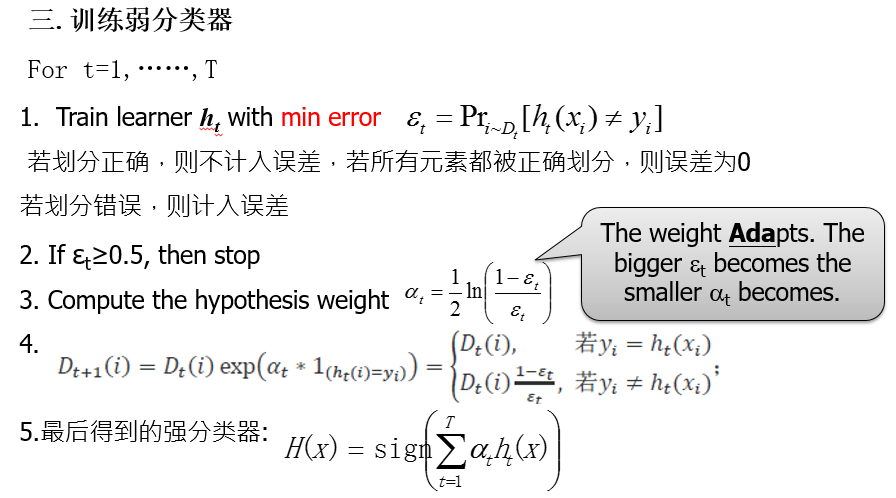
**三、实验内容**

**（1）Adaboost的原理**

Adaboost算法是经过调整的Boosting算法，其能够对弱学习得到的弱分类器的错误进行适应性(Adaptive)调整。上述算法中迭代了T次的主循环，每一次循环根据当前的权重分布对样本x定一个分布P，然后对这个分布下的样本使用弱学习算法得到一个弱分类器，对于这个算法定义的弱学习算法，对所有的样本都有错误率，而这个错误率的上限并不需要事先知道，实际上。每一次迭代，都要对权重进行更新。更新的规则是：减小弱分类器分类效果较好的数据的概率，增大弱分类器分类效果较差的数据的概率。最终的分类器是个弱分类器的加权平均。

**（2）Adaboost的算法**





笼统来看，有以下两个方面需要考虑：

1. 使用加权后选取的训练数据代替随机选取的训练样本，这样将训练的焦点集中在比较难分的训练数据样本上；

2. 将弱分类器联合起来，使用加权的投票机制代替平均投票机制。让分类效果好的弱分类器具有较大的权重，而分类效果差的分类器具有较小的权重。

**(3)本次实验的设计**

“半圆对拱形”的数据集过去做过，而本次实验意在探究集成学习的优势，所以我们摒弃了之前可以解决这类线性不可分的SVM和BP神经网络，因为这些分类方法单个模型就可以很好的解决这种问题，无需集成学习。

相应的，线性不可分的单层感知机是无法很好的分割两类样本，所以，这就是很好的集成学习对比工具，不妨用单层感知机的线性分类器充当我们的弱分类器。

所以我们训练了5个单层感知机，还是采取普适的梯度下降法训练每个弱分类器，不同的是，本次实验还需要考虑权重的概念，也就是说lost的准则函数还需要乘上数据的权重，体现到代码，也就是

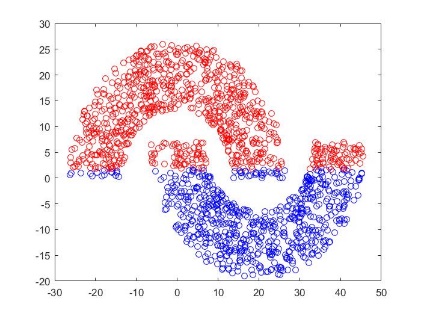
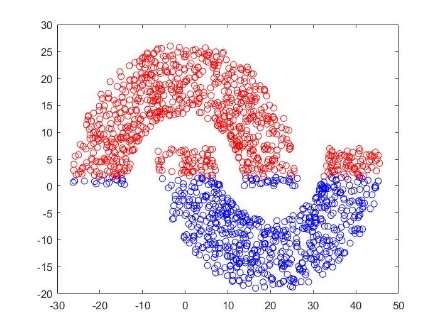
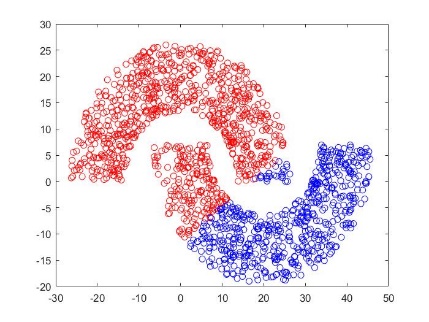
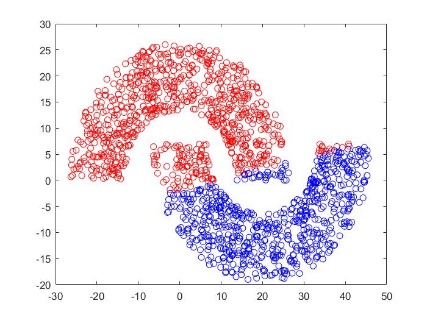
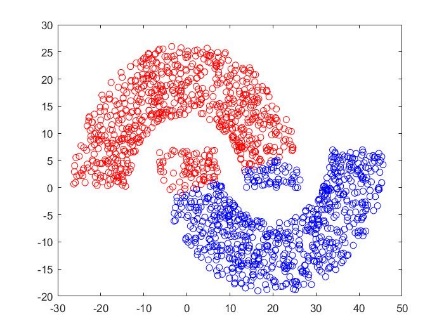


其中，dd是储存数据权重的向量，具体调整权重的计算公式在上边的算法中已经给出，不必赘述。

另外，需要强调的问题是，不同的弱分类器应该是串行训练的关系，而万万不可以设计成并行的，如果设计成并行训练的方式，就变成了我们的另一种集成学习方法，而失去了adaboost的核心思想。

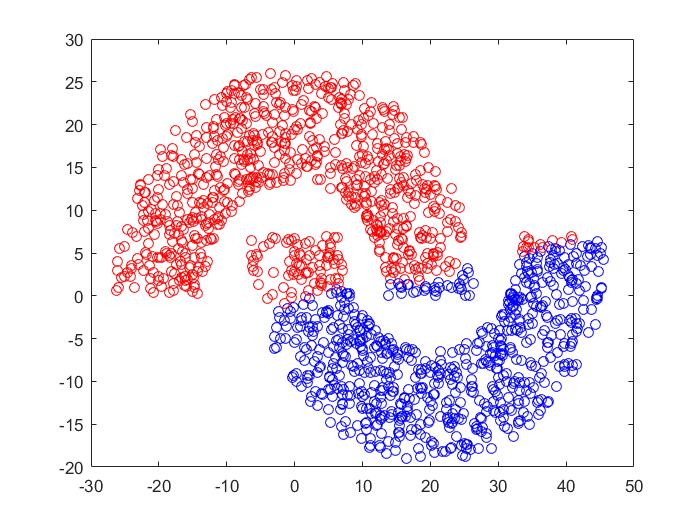
**四、实验结果**

图1是我用梯度下降训练的5个单层感知机线性分类器，它们存在着不同程度的线性不可分程度。

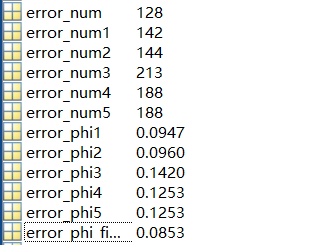
****

**图1** 五个弱分类器分类情况

图2是最终的强分类器分类情况：



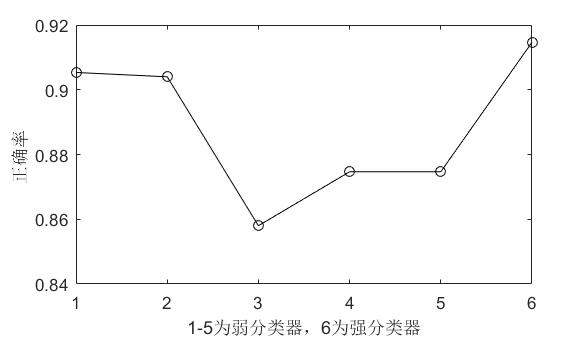
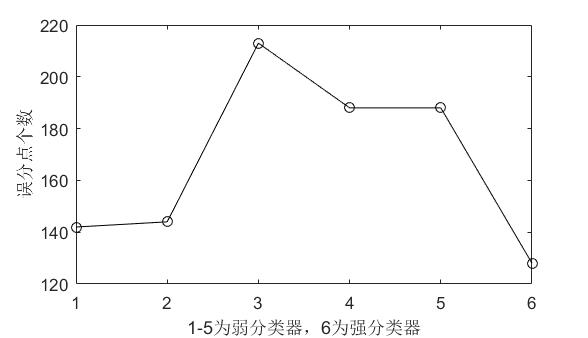
**图2** 强分类器分类情况



**图3** matlab数据截图

左图是matlab的数据截图，128个（一共测试1500个）的error\_num来自强分类器，其余来自弱分类器；0.0853的错误率来自强分类器，其余来自弱分类器；

比较统计图作图如下：



**图4** 统计数据折线图

可以看出，强分类器的128个五分点是最少的，0.915的正确率是最高的，可以看出，adaboost集成学习的分类效果相比单个线性分类器有明显提升。

实验效果已经比较理性，但还可以有所突破，只不过，我们只使用了5个弱分类器，试想使用15个、25个…最终可以实现彻底的线性不可分，并且不用担心overfitting问题。这在周志华教授的文章里有过详细证明。

**五、总结**

最后，我们可以总结下adaboost算法的一些实际可以使用的场景：

1）用于二分类或多分类的应用场景

2）用于做分类任务的baseline，无脑化，简单，不会overfitting，不用调分类器

3）用于特征选择（feature selection)

4）Boosting 框架用于对badcase的修正

只需要增加新的分类器，不需要变动原有分类器由于adaboost 算法是一种实现简单，应用也很简单的算法。Adaboost算法通过组合弱分类器而得到强分类器，同时具有分类错误率上界随着训练增加而稳定下降，不会过拟合等的性质，应该说是一种很适合于在各种分类场景下应用的算法。

至此，实验比较的达到了预期目标。