**Boosting算法**

Boosting算法是一种**集成学习法（Ensemble Learning）**：将多个分类方法聚集在一起，以提高分类的准确率。（这些算法可以是不同的算法，也可以是相同的算法。）在boosting算法产生之前，还出现过两种比较重要的将多个分类器整合为一个分类器的方法，即bootstrapping方法和bagging方法。首先简要介绍一下这两种方法。

1）bootstrapping方法

思想：利用有限的样本资料经由多次重复抽样，重新建立起足以代表样本分布的新样本。

主要步骤：

　　i)重复地从一个样本集合D中采样n个样本

　　ii)针对每次采样的子样本集，进行统计学习，获得假设Hi

iii)将若干个假设进行组合，形成最终的假设Hfinal

2）bagging方法

bagging方法也叫bootstrap aggregation方法。算法流程如下：

1：对于给定的训练样本集合,通过n次的随机有放回采样,从原始的样本集合中构建一个Bootstrap样本集合。

2：对于每一个Bootstrap样本集合构建一个分类器。

3：重复1-2步，获取更多的分类器。

4：是每一个分类器对输入样本X的分类进行投票

5：计算所有的投票数目，并以投票最多的一个分类标签作为X的分类结果。、

（相比之下，随机森林是对训练集的行（样本）和列（特征）同时进行采样。）

3）boosting算法

boosting（提升）算法是一个迭代的过程，用于自适应地改变训练样本的分布，使得基分类器聚焦在那些很难分的样本上。

Schapire提出的一种早期的boosting算法，其主要过程如下：

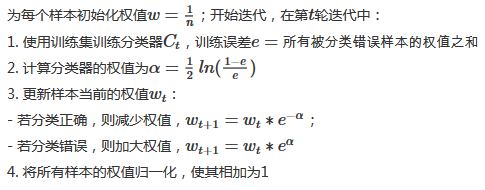
　　i)从样本整体集合D中，不放回的随机抽样n1 < n个样本，得到集合 D1，训练弱分类器C1

　　ii)从样本整体集合D中，抽取 n2 < n个样本，其中合并进一半被C1 分类错误的样本。得到样本集合 D2，训练弱分类器C2

　　iii)抽取D样本集合中，C1 和 C2 分类不一致样本，组成D3，训练弱分类器C3

　　iv)用三个分类器做投票，得到最后分类结果

AdaBoost（Adaptive Boosting）是一种常见的boosting算法。每轮迭代中会在训练集上产生一个新的分类器，然后使用该分类器对所有样本进行分类，以评估每个样本的重要性（informative）。 具体来说，若某个样本点已被分类正确，则将其权值降低；若样本点未被正确分类，则提高其权值。整个迭代过程直到错误率足够小或达到一定次数为止。训练过程:



bagging和boosting都可以有效地提高分类的准确性。在大多数数据集中，boosting的准确性比bagging高。在有些数据集中，boosting会引起Overfit。