bagging与随机森林

欲得到泛化性能强的集成，集成中的个体学习器应尽可能相互独立;虽然"独立"在现实任务中无法做到，但可以设法使基学习器尽可能具有较大的差异.给定一个训练数据集,一种可能的做法是对训练样本进行采样，由于训练数据不同，我们获得的基学习器可望具有比较大的差异.然而,为获得好的集成，我们同时还希望个体学习器不能太差，如果采样出的每个子集都完全不同，则每个基学习器只用到了一小部分训练数据,甚至不足以进行有效学习，这显然无法确保产生出比较好的基学习器.为解决这个问题，我们可考虑使用相互有交叠的采样子集.

Bagging

给定包含m个样本的数据集，我们先随机取出一个样本放入采样集中，再把该样本放回初始数据集，使得下次采样时该样本仍有可能被选中，这样，经过m
次随机采样操作，我们得到含m个样本的采样集，初始训练集中有的样本在采样集里多次出现，有的则从未出现.初始训练集中约有63.2%,
照这样，我们可采样出T个含m个训练样本的采样集，然后基于每个采样集训练出一个基学习器，再将这些基学习器进行结合.这就是Bagging的基本流程.在对预测输出进行结合时，Bagging通常对分类任务使用简单投票法，对回归任务使用简单平均法.若分类预测时出现两个类收到同样票数的情形，则最简单的做法是随机选择-一个，也可进一步考察学习器投票的置信度来确定最终胜者.
算法描述如下:
输入： 训练集$D = (x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)$
基学习算法ξ
训练轮次T
过程:

1. for t=1,2,3..T do
2. $h_t = \xi(D, D_{bs})$
3. end for

输出:

$$H(x) = arg\max_{y \in Y} \sum_{t=1}^{T} II(h_t(x) = y)$$

随机森林

随机森林(Random Forest,简称RF)是Bagging的一个扩展变体.RF在以决策树为基学习器构建Bagging集成的基础上，进一步在决策树的训练过程中引入了随机属性选择.具体来说，传统决策树在选择划分属性时是在当前结点的属性集合(假定有d个属性)中选择-一个最优属性;而在RF中,对基决策树的每个结点，先从该结点的属性集合中随机选择一个包含k个属性的子集，然后再从这个子集中选择一个最优属性用于划分.这里的参数k控制了随机性的引入程度:若令k=d,则基决策树的构建与传统决策树相同;若令k= 1,则是随机选择-一个属性用于划分;一般情况下，推荐值$k = log_2 d$