1. 集成学习

   集成学习(ensemble learning)通过构建并结合多个学习器来完成学习任务,有时也被称为多分类器系统(multi-classifier system)、基于委员会的学习(committee-based learning)等.集成学习通过将多个学习器进行结合，常可获得比单一学习器显著优越的泛化性能.这对“弱学习器”(weak learner)尤为明显，因此集成学习的很多理论研究都是针对弱学习器进行的，而基学习器有时也被直接称为弱学习器，但需注意的是，虽然从理论上来说使用弱学习器集成足以获得好的性能，但在实践中出于种种考虑，例如希望使用较少的个体学习器，或是重用关于常见学习器的一些经验，人们往往会使用比较强的学习器.

   在一般的经验中，如果把好坏不等的东西掺到一起，那么通常结果会是比最坏的要好一些，比最好的要坏一些.那集成学习如何能获得比最好的单一学习器更好的性能呢？

   考虑一个简单的例子：在而分类任务中，假定三个分类器在三个测试样本上的表现如图8.2所示，其中√表示分类正确，×表示分类错误，集成学习的结果通过投票法(voting)产生，即“少数服从多数”.

2. 个体学习器

   A 个体学习器(individual learner)通常由一个现有的学习算法从训练数据产生,例如C4.5决策树算法、BP神经网络算法等，此时集成中只包含同种类型的个体学习器，例如“决策树集成”中全是决策树，“神经网络集成中”全是神经网络，这样的集成是“同质”的(homogeneous).同质集成中的个体学习器亦称为“基学习器”(base learner),相应的学习算法称为“基学习算法”(base learning algorithm).集成也可包含不同类型的个体学习器，例如同时包含决策树和神经网络，这样的集成是“异质”的(heterogenous.)异质集成中的个体学习器由不同的学习算法生成，这时就不再有基学习算法；相应的个体学习器一般不称为基学习器，常称为“组件学习器”(component learner)或直接称为个体学习器.

3. 为什么集成学习有效? 考虑二分类问题y取值是-1或者1和真实函数f,假定基分类器的错误率为ϵ即对每个基分类器h_i有

$$P(h_i(x) = f(x)) = \epsilon$$

假设集成通过简单投票法结合T个基分类器,若有超过半数的基分类器正确，则集成分类就正确：

$$H(x) = sign(\sum_{i}^{T} h_i(x))$$

假设基分类器的错误率相互独立，则由Hoeffding不等式可知,集成的错误率为

$$\begin{aligned} P(H(x) = f(x)) &= \sum_{k=0}^{[T/2]} \binom{T}{k} (1-\epsilon)^k \epsilon^{T-k} \\ &\leq exp(-\frac{1}{2}T(1-2\epsilon)^2). \end{aligned}$$

上式显示出，随着集成中个体分类器数目T的增大，集成的错误率将指数级下降，最终趋向于0.

然而我们必须注意到，上面的分析有一个关键假设：基学习器的误差相互独立.在现实任务中，个体学习器是为解决同一个问题训练出来的，它们显然不可能相互独立！事实上，个体学习器的“准确性”和“多样性”本身就存在冲突.一般的，准确性很高之后，要增加多样性就需牺牲准确性.事实上，如何产生并结合“好而不同”的个体学习器，恰是集成学习研究的核心. 根据个体学习器的生成方式，目前集成学习方法大致可分为两大类，即个体学习器间存在强依赖关系、必须串行生成的序列化方法，及个体学习器不存在强依赖关系、可同时生成的并行化方法；前者是Boosting,后者的代表是Bagging和“随机森林”(Random Forest). 4. Boosting Boosting是一族可将弱学习器提升为强学习器的算法.这族算法的工作机制类似：先从初始训练数据集训练出一个基学习器，再根据基学习器的表现对训练样本分布进行调整，使得先前基学习器做错的训练样本在后续受到更多关注，然后基于调整后的样本分布来训练下一个基学习器；如此重复进行，直至基学习器数目达到事先指定的值T，最终将这T这个基学习器进行加权结合. Boosting族算法最著名的代表是AdaBoost。 AdaBoost算法有多种推导方式，比较容易理解的是基于“加性模型”，即基学习器的线性组合

$$H(x) = \sum_{t=1}^{T} a_t h_t(x)$$

来最小化指数损失函数

$$l_{exp}(H|D) = E_{x-D}[e^{-f(x)H(x)}]$$