火龙果•整理
uml.org.cn

集成学习综述

梁英毅

摘要　机器学习方法在生产、科研和生活中有着广泛应用，而集成学习则是机器学习的首要热门方向 [1]。集成学习是使用一系列学习器进行学习，并使用某种规则把各个学习结果进行整合从而获得比单个学习器更好的学习效果的一种机器学习方法。本文对集成学习的概念以及一些主要的集成学习方法进行简介，以便于进行进一步的研究。

一、　引言

机器学习是计算机科学中研究怎么让机器具有学习能力的分支，[2] 把机器学习的目标归纳为“给出关于如何进行学习的严格的、计算上具体的、合理的说明”。[3] 指出四类问题的解决对于人类来说是困难的甚至不可能的，从而说明机器学习的必要性。

目前，机器学习方法已经在科学研究、语音识别、人脸识别、手写识别、数据挖掘、医疗诊断、游戏等等领域之中得到应用 [1, 4]。随着机器学习方法的普及，机器学习方面的研究也越来越热门，目前来说机器学习的研究主要分为四个大方向 [1]：

a) 通过集成学习方法提高学习精度；

b) 扩大学习规模；

c) 强化学习；

d) 学习复杂的随机模型；

有关 Machine Learning 的进一步介绍请参考 [5, 1,3, 4, 6]。

本文的目的是对集成学习的各种方法进行综述，以了解当前集成学习方面的进展和问题。本文以下内容组织如下：第二节首先介绍集成学习；第三节对一些常见的集成学习方法进行简单介绍；第四节给出一些关于集成学习的分析方法和分析结果。

二、　集成学习简介

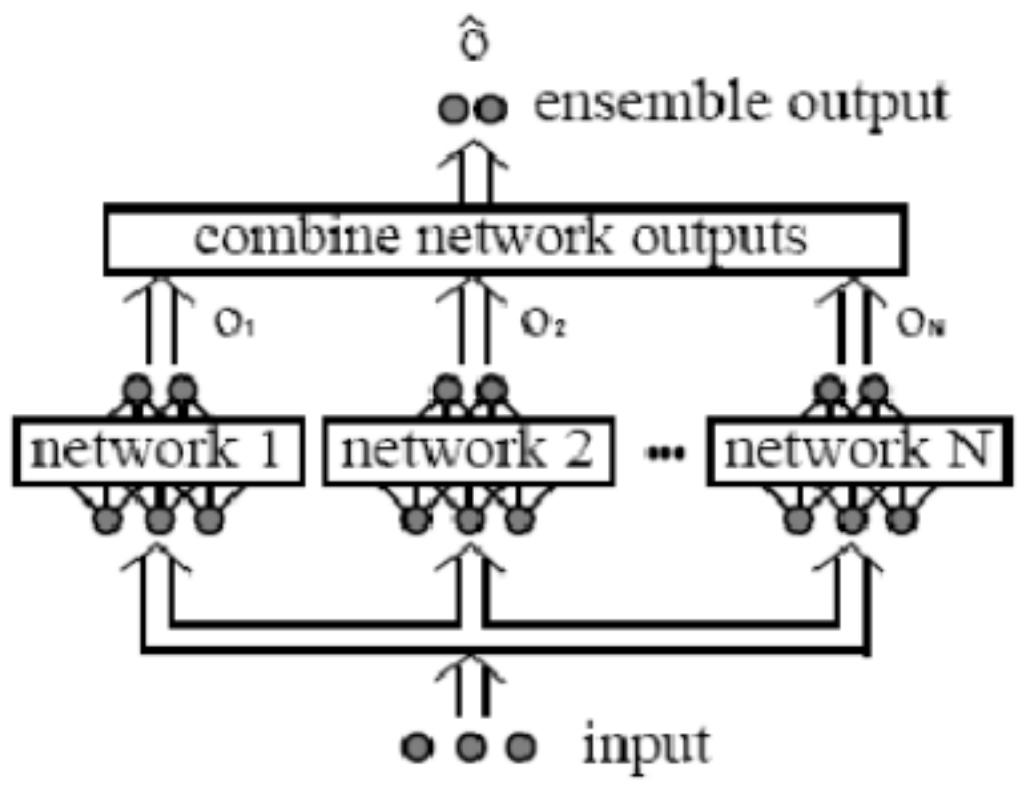
1、 分类问题

分类问题属于概念学习的范畴。分类问题是集成学习的基本研究问题，简单来说就是把一系列实例根据某种规则进行分类，这实际上是要寻找某个函数 $y = f(x)$，使得对于一个

给定的实例 x，找出正确的分类 y。机器学习中的解决思路是通过某种学习方法在假设空间中找出一个足够好的函数 h 来近似 f，这个近似函数 h 就叫做分类器 [7]。

2、 什么是集成学习

传统的机器学习方法是在一个由各种可能的函数构成的空间（称为“假设空间”）中寻找一个最接近实际分类函数 f 的分类器 h [6]。单个分类器模型主要有决策树、人工神经网络、朴素贝叶斯分类器 [6] 等等。集成学习的思路是在对新的实例进行分类的时候，把若干个单个分类器集成起来，通过对多个分类器的分类结果进行某种组合来决定最终的分类 [8, 9]，以取得比单个分类器更好的性能。如果把单个分类器比作一个决策者的话，集成学习的方法就相当于多个决策者共同进行一项决策。图表 1表示了集成学习的基本思想。图表 1中的集成分类器包括了 N个单一的人工神经网络分类器，对于同样的输入，N 个人工神经网络分别给出各自的输出（$O_1, O_2, \ldots, O_n$），然后这些输出通过整合以后得到集成分类器整体的输出结果作为最终分类。



图表 1 人工神经网络集成示意

在各种文献中集成学习（Ensemble Learning）也被称为 committee、classifier fusion、combination、aggregation 等等 [9]，但是他们都是指把一系列的学习器集成起来共同解决一个问题。

3、 为什么集成学习有效

Thomas G. Dietterich [8, 7] 指出了集成学习有效可以归为三个方面的原因：统计上、计算

上和表示上：

a) 统计上的原因

对于一般的学习任务，往往要搜索的假设空间十分巨大，但是能够用于训练分类器的训练集中实例个数不足够用来精确地学习到目标假设，这个时候学习的结果便可能是一系列满足训练集的假设，而学习算法之能够选择这些假设的其中之一作为学习到的分类器进行输出。然而通过机器学习的过拟合问题 [6] 我们看到，能够满足训练集的假设不一定在实际应用中有同样好的表现，这样学习算法选择哪个假设进行输出的时候就面临着一定的风险，把多个假设集成起来能够降低这种风险（这可以理解为通过集成使得各个假设和目标假设之间的误差得到一定程度的抵消）。

b) 计算上的原因

已经证明了在人工神经网络学习和决策树学习中，学习到最好的人工神经网络或者是决策树是一个 NP-hard 问题 [1 , 8] ，其他的分类器模型也面临着类似的计算复杂度的问题。这使得我们只能用某些启发式的方法来降低寻找目标假设的复杂度，但这样的结果是找到的假设不一定是最优的。通过把多个假设集成起来能够使得最终的结果更加接近实际的目标函数值。

c) 表示上的原因

由于假设空间是人为规定的，在大多数机器学习的应用场合中实际目标假设并不在假设空间之中，如果假设空间在某种集成运算下不封闭，那么我们通过把假设空间中的一系列假设集成起来就有可能表示出不在假设空间中的目标假设。关于表示上的原因比较特别，进一步的说明可以参考 [8, 9] 。

4、集成学习有效的条件

虽然以上几方面的原因使得集成学习可能取得更好的学习效果，但是并不是所有的集成方式都有效的，集成学习有效的条件是每个单一的学习器错误率都应当低于 0.5，否则集成的结果反而会提高错误率 [11, 12, 9, 10]。

此外，进行集成学习的每个分类器还应当各不相同，这个要求可以这样理解：如果每个基本分类器分类结果差不多，则集成后的分类器整体和单个分类器做出的决策实际上没有什么差异，这样集成后的分类器就难以保证比单个分类器有更好的性能了。

三、 集成学习方法介绍

从前面对集成学习的介绍可以看出，考察一个集成学习方法的时候应该考虑以下几方面的问题：

a) 基本分类器之间是什么关系？

b) 怎么样生成多个不同的基本分类器？

c) 如何把多个基本分类器的分类结果整合起来？

我们下面将以这三个问题为线索对现在主要的集成学习方法进行简单的介绍：

1、 基本分类器之间的关系

按照基本分类器之间的种类关系可以把集成学习方法划分为异态集成学习和同态集成学习两种 [10]。

a) 异态集成学习

异态集成学习指的是使用各种不同的分类器进行集成，异态集成学习的两个主要代表是叠加法（ Stack Generalization） [13] 和元学习法（ Meta Learning ） [14] 。

叠加法的思想是把基本学习器分布在多个层次上，用多层的学习器来完成学习任务 [13] 。第一层学习器按照一般学习器的方式来进行分类预测，然后第一层的预测结果作为第二层的输入由第二层再进行预测……

William W. Cohen 等在 [15] 中利用叠加法的思想构造了一种新的串行学习算法，并且指出这种串行学习方法比不串行的单个学习器性能上有所改进。

[13] 中对叠加法进行了详细的介绍。

元学习法的思想是训练一个元分类器来对所有的基本学习器的输出进行处理，最终得到问题的输出。元学习法主要有仲裁法（ arbiter）和合并法（ combiner）两种，仲裁法是元分类器从所有基本分类器的输出中选择合理的结果作为输出，例如投票方式； 合并法是用某种组合方法把所有基本分类器的输出合并成最终输出，Bagging、Boosting 等集成方法都是属于合并法。

[14] 中对元学习法给出了一个综述。

关于异态分类器的进一步资料可以参考 [16] 。

b) 同态集成学习

同态集成学习是指集成的基本分类器都是同一种分类器，只是这些基本分类器之间的参数有所不同。同态集成的分类器包括有朴素贝叶斯集成、决策树集成 [17] 、人工神经网络集

成[18, 19, 20]、K-近邻集成等等 [10]，其中以决策树集成和人工神经网络集成居多。

2、 不同的基本分类器的获得方式

基本分类器的多样性是评价集成学习算法的重要标准 [10]，因此如何获取不同的基本分类器对集成学习的效果有着重要影响。 对于异态集成学习方法， 基本分类器的不同来源于他们本身种类的不同； 对于同态集成学习方法， 基本分类器的不同来自于多种可能， 以下把不同基本分类器的获取方式划分为五类，并分别予以说明：

a) 基本分类器种类不同

如果一种集成学习方式采用异态集成的方案， 则所得到的基本分类器很明显不同。 但是由于有效的分类器基本模型并不多， 因此进行异态集成学习的时候往往需要对于每一种分类器都给出若干不同的实例，这些同一种分类器的不同实例的产生需要用到下面几种获取方式。

b) 对训练数据进行处理

i. Bagging

Bagging （Bootstrap AGGregatING ）是 Breiman 在 [21] 中提出的，其思想是对训练集有放回地抽取训练样例， 从而为每一个基本分类器都构造出一个跟训练集同样大小但各不相同的训练集，从而训练出不同的基本分类器。 Bagging 算法如图表 2所示：

Input:

L: a learning algorithm

N: an integer

For $i = 1$ to N

$\bar{T}$ = bootstrap sample from training set T

$h_i = L(\bar{T})$

End For

Output:

$$h_f(\mathbf{x}) = \underset{y \in \mathbf{Y}}{\operatorname{argmax}} \sum_i^N h_i(\mathbf{x}) = y$$

图表 2 Bagging 算法示意

Bagging 是基于对训练集进行处理的集成方法中最简单、最直观的一种。使用 Bagging 算法的时候，理论上每个基本分类器的训练集中有 63.2%的重复样例 [21, 10]。

Breiman 在 [21] 中同时指出， 要使得 Bagging 有效， 基本分类器的学习算法必须是不稳定

的，也就是说对训练数据敏感。基本分类器的学习算法对训练数据越敏感，Bagging 的效果越好，因此对于决策树和人工神经网络这样的学习算法 Bagging 是相当有效的。另外由于 Bagging 算法本身的特点，使得 Bagging 算法非常适合用来并行训练多个基本分类器，这也是 Bagging 算法的一大优势。

那么使用 Bagging 算法的时候应该用多少个基本分类器合适呢？Breiman 指出基本分类器数目应当随着分类种数增多而增加。

[22, 23] 给出了 Bagging 算法和其他集成算法的实验结果比较。关于 Bagging 的更多资料请参考 [24, 25 , 23, 8, 16] 等等。

ii. Boosting

Boosting 最先由 Robert T. Schapire 在 [26] 中提出，其思想是对每那些容易分类错的训练实例加强学习，就好像一个人背英语单词一样，首先第一遍背完以后有一些容易记住的单词就记住了，还有一些不容易记住的，则第二遍的时候对那些不容易记住的单词多看几眼，第三遍又对第二遍还记不住的单词再多看几眼……具体实施的时候 Boosting 方法是这样进行的：首先给每一个训练样例赋予相同的权重，然后训练第一个基本分类器并用它来对训练集进行测试，对于那些分类错误的测试样例提高其权重（实际算法中是降低分类正确的样例的权重），然后用调整后的带权训练集训练第二个基本分类器，然后重复这个过程直到最后得到一个足够好的学习器。图表 3给出了最常用的 Boosting 算法——由 Yaov Freund和 Schapire[27] 共同提出的 AdaBoost （Adaptive Boosting ）算法的描述。

Input: a set S, of m labeled examples: $S = \{(\mathbf{x}_i, y_i), i = 1, 2, \ldots, m\}$,
labels $y_i \in Y = \{1, \ldots, K\}$
LEARN (a learning algorithm)
a constant L.

[1] **initialize for all** i: $w_1(i) := 1/m$ — *initialize the weights*
[2] **for** $\ell = 1$ **to** L **do**
[3] **for all** i: $p_\ell(i) := w_\ell(i)/(\sum_i w_\ell(i))$ — *compute normalized weights*
[4] $h_\ell := \text{LEARN}(p_\ell)$ — *call* LEARN *with normalized weights.*
[5] $\epsilon_\ell := \sum_i p_\ell(i)[\![h_\ell(\mathbf{x}_i) \neq y_i]\!]$ — *calculate the error of* h_ℓ
[7] **if** $\epsilon_\ell > 1/2$ **then**
[8] $L := \ell - 1$
[9] **goto 13**
[10] $\beta_\ell := \epsilon_\ell/(1 - \epsilon_\ell)$
[11] **for all** i: $w_{\ell+1}(i) := w_\ell(i)\beta_\ell^{1-[\![h_\ell(\mathbf{x}_i)\neq y_i]\!]}$ — *compute new weights*
[12] **end for**

[13] **Output:** $h_f(\mathbf{x}) = \underset{y \in Y}{\operatorname{argmax}} \sum_{\ell=1}^{L} \left(\log \frac{1}{\beta_\ell}\right) [\![h_\ell(\mathbf{x}) = y]\!]$

图表 3 AdaBoost 算法

Boosting 的基本目标是把一个弱学习器转化为一个任意高精度的学习器 [26]，什么是弱学习器呢？ Schapire 指出弱学习器是指一个学习效果比随机分类略好的学习器 [26]。Boosting 方法为什么有效呢？近年的研究指出 Boosting 算法实际上是一个逐阶段最小化一个特定的误差函数的梯度下降过程 [8]。

Boosting 的原本目的是用于学习二值分类器，但是应用到训练多值分类器的方法也已经被研究过了 [10]。与 Bagging 对于噪声容忍程度比较好不同，Boosting 似乎对噪声容忍程度不高 [8]。关于 Boosting 算法的实验结果比较可以参考 [22, 23]，进一步的资料可以参考 [28, 9, 29, 17, 30, 32, 31] 等。

iii. 交叉验证

交叉验证法来自于 K 重交叉验证（K-fold Cross Validation）的测试方法，属于无放回抽抽样 [1, 9, 10]。K 重交叉验证的思想是把整个训练集合划分为互不相交的 K 等份，然后每次选定一份作为测试集，其他的 K-1 份作为训练集来进行学习，用这种思路可以训练出 K 个不同的基本分类器。跟 Bagging 一样，这种集成方法也适用于多个基本分类器并行进行训练。

c) 对输入特征进行处理

这种方法的思想是对于具有多种输入特征的实例，通过抽取不同的输入特征子集分别进行训练，从而获得不同的分类器 [10, 8, 9]，把这些分类器的分类结果适当整合能够获得比任何一个基本分类器的分类精度都要高的分类器 [7]。特征子集的选取又叫做集成特征选取（Ensemble Feature Selection），根据特征子集的选取方式不同，又有随机子空间法（随机抽取特征子集）和少量余留法（Input Decimation，只抽取和问题最相关的那些特征）[9]；Alexey Tsymba 等在 [33] 中介绍了一种新的遗传搜索方法来选择输入特征子集，并指出这种方法虽然比一般方法耗时长，但是具有更佳的学习效果。

Dietterich 在 [1, 8] 中通过两个项目中使用特征子集法的不同效果的比较指出这种方法在输入特征有大量冗余的情况下才有效。Hiroshi Mamitsuka 等在 [34] 中也对特征子集的选取问题进行了实验研究。

d) 对输出结果进行处理

Dietterich、Bakiri 等在 [35] 中提出了一种纠错码方法来产生多个不同的基本分类器。这

种方法的基本原理是 [35] 在目标分类数目很多的情况下，采用某种编码方式对目标分类进行二进制编码，使得每个目标分类都分配到一个长度为 L 位的编码，并且保证不同的目标分类分到的编码之间的海明码距足够大，这样学习任务就转化为学习 L 个不同的二值分类，这样可以通过在相同的训练集上训练 L 个不同的二值分类器来完成，最后把每一个分类器的结果合起来找出编码值最接近这个结果的目标分类作为整个分类器的输出即可。

他们的具体做法是这样的：对于有 K 个目标分类的学习任务，按照编码要求计算出编码长度 L，然后对于 L 个二值分类器的每一个做这样的处理：把所有的目标分类随机划分为 A、B 两部分，把划分到 A 的那些实例的标号改成 0，其它实例的标号改成 1，然后用更改标号以后的训练集来训练该二值分类器。重复这个过程 L 次则产生了 L 个不同的二值分类器。进行分类的时候对于每一个分类器的分类结果，如果是 0，则给划分到对应这个分类器的 A 部分那些目标分类投一票，否则给其它目标分类投一票，最后看哪个目标分类得票最多来作为最终的分类结果。

在 [35] 里面他们通过用决策树和人工神经网络进行实验的结果说明纠错码这种集成方法不仅对于一系列难度比较大的多目标分类的问题能够有效改善基本分类器的分类精度，还有以下的优点：

i. 对于训练集大小很小的情况这种方法仍然有效；

ii. 不依赖于每一个目标分类的具体编码值。

同时他们也指出编码长度并不是越长越好，一种中等长度的编码会比很长的编码学习效果更好。这种方法的详细介绍请参见 [1, 8, 9] 等。

e) 引入随机扰动

随机扰动法（ Injecting Randomness ）的思想是在每个基本分类器的学习过程之中引入随机扰动，使得学习出来的每个基本分类器都不同，这是用于进行人工神经网络集成学习的最常见方法 [1] 。只要基本学习器对随机扰动敏感，则随机扰动法能够有效地产生多个不同的基本分类器。对于人工神经网络，使用后向传递算法来进行学习的时候对于每个神经网络的初始权值都随机分配，则产生的基本分类器会有很明显的不同 [1] ，而对于决策树，则可以在每个决策结点的特征选取的时候从某个特征子集（例如前 20 个导致熵增益最大的特征构成的集合）里面随机选取一个作为该结点的分类特征。

Dietterich 等在 [1, 22, 8] 中给出了使用随机扰动法的各种实验结果，他们在其论文中并且指出在噪声相对比较低的情况下随机扰动法效果甚至会比 AdaBoost 算法要好，但是在噪声

比较高的情况下随机扰动法的效果并不令人满意。

随机扰动法除了对学习精度的改进以外还有一个很特别的优点，就是它可以很容易地跟前面介绍的其他集成方法结合起来，从而取得更好的学习效果。

3、 基本分类器分类结果的整合方式

现在我们有多个基本分类器的分类结果了，但是怎么根据这么多的分类结果来给出最终决策呢？ Lei Xu 等在 [36] 中对分类器的整合问题进行了深入的研究，他们把基本分类器的整合方式归纳为三个层次：

a) 抽象层次：每个基本分类器只提供一个目标分类或者目标分类子集；
b) 排位层次：每个基本分类器提供一个可能的目标分类列表，其中的目标分类按照可能性大小排列；
c) 度量层次：每个基本分类不仅提供分类结果，还提供每种分类结果的可能性。

本文中我们主要介绍抽象层次的整合方式，我们把这种整合方式归为四类，并分别进行说明：

a) 简单投票

投票法的基本思想是多个基本分类器都进行分类预测，然后根据分类结果用某种投票的原则进行投票表决，按照投票原则的不同投票法可以有一票否决、一致表决、少数服从多数、阈值表决等等 [36, 16]。

一票否决的思想是当且仅当所有的分类器都把实例 x 划分到类 C_i 的时候才把 x 划分到 C_i，否则拒绝这个实例；一致表决的思想是没有分类器反对把 x 划分到 C_i 的时候就把 x 划分到 C_i；少数服从多数顾名思义就是当让各个基本分类器进行投票（加权或者不加权），得票数多的那个分类 C_i 作为对应实例 x 的最终分类；阈值表决是首先统计出把实例 x 划分为 C_i 和不划分为 C_i 的分类器数目分别是多少，然后当这两者比例超过某个阈值的时候把 x 划分到 C_i。

此外，对于连续目标值的分类问题，还可以采用平均取值、加权求和、中位数、最大值等等方式进行整合 [37]。

b) 贝叶斯投票

简单投票法假设每个基本分类器都是平等的，没有分类能力之间的差别，但是这种假设并不总是合适的，在实际生活中，我们听取一个人的意见的时候会考虑到这个人过去的意见是否有用，贝叶斯投票法就是基于这种思想来提出的。贝叶斯投票法是基于每一个基本分类器在过去的分类表现来设定一个权值，然后按照这个权值进行投票，其中每个基本分类器的权值基于贝叶斯定理来进行计算 [36, 8]。

虽然理论上贝叶斯投票法在假设空间所有假设的先验概率都正确的情况下能够获得最优的集成效果，但是实际应用中往往不可能穷举整个假设空间，也不可能准确地给每个假设分配先验概率，从而使得在实际使用中其他集成方法也会优于贝叶斯投票法。

c) 基于 D-S 证据理论的整合方式

Lei Xu[36]、Ke Chen[38] 和 Ahmed Al-Ani[37] 等先后提出了若干种基于 D-S 证据理论 [77] 的整合方式，这些整合方式的基本思想是通过识别率、拒绝率等一系列参数计算出每个目标分类的信任范围，从而最终推断出分类结果。关于基于 D-S 证据理论的整合方式的详细说明可以参考 [36] 和[37]。

d) 对基于不同特征子集得到的基本分类器的整合

Ke Chen 等在 [38] 中研究了基于不同特征子集所产生的基本分类器如何整合的问题。在他们的文章当中他们研究了线性整合、Winner-Take-All 和证据推理几种整合方式。其中线性整合指的是使用各个基本分类器输出的线性组合来作为分类结果；Winner-Take-All 指的是对于每一种输入模式都分别选定某一个基本分类器作为胜利者，把这个基本分类器的结果作为分类结果进行输出；证据推理用的就是前面说到的基于 D-S 证据理论的计算方法。

除此以外 Mohamed S. Kamel 和 Nayer M. Wanas 还讨论了数据相关性对于整合基本分类器的影响 [39]；在基本分类器个数问题上 [18] 中指出只拿多数基本分类器进行整合可能比拿全部基本分类器的结果进行整合效果更佳；关于基本分类器分类结果整合的更多资料请参考 [40, 41, 42, 2, 16, 12, 36]。

4、 其它的集成学习方法

上面介绍的集成学习方法都是离线集成学习方法，也就是说这些集成学习方法都是通过

给定一个训练集，学习以后输出一个集成的分类器。Alan Fern[43] 和 Alexey Tsymbal[44] 等在文章中讨论了在线集成学习方法，其基本的原理是一边进行分类一边对某些基本分类器的参数进行更新，或者是添加新的基本分类器，从而实现整个分类器的动态集成。他们实现了两个在线集成学习算法并且应用到处理器的分支预测问题上面，通过比较实验结果他们指出基于Boosting 的在线学习方法比基于 Bagging的在线学习方法有更好的表现。

此外除了上面介绍的通用集成学习方法以外，还有一些跟特定基本分类器有关的集成学习方法 [1, 9]。例如 [1] 中扼要介绍了一些跟人工神经网络有关的特殊集成学习算法，而 Giorgio Valentini 等则在 [9] 中对其它的特殊集成方式进行了介绍和讨论。

Kristin P. Bennet 等 [45] 介绍了一种半指导的集成学习方法 ASSEMBLE （Adaptive Semi-Supervised EnsMBLE ），他们对未标号数据的处理方式是依照某种规律赋予一个目标分类值，并且证明了这样的做法跟 Boosting 方法有类似的功能。

Yuhong Guo 等 [GUO05] 提出另外一种集成学习的思路，他们的做法不是通过学习多个基本分类器来实现集成学习，而是只采用一个基本分类器，但是训练和测试这个基本分类器的时候使用成对的样例。他们在论文中指出对于大多数的机器学习应用来说，数据独立同分布的假设并不成立，从而从理论上指出利用数据的相关性可能可以通过增强假设的表示能力来改善学习效果。同时他们在文章中的实验数据表明这种集成学习的方式跟 Boosting 有相类似的效果。

另外，在集成学习中往往需要训练多个分类器，如果数据的输入特征数目太大，则会严重影响学习性能——而且这种额外的时间开销是不必要的，因此我们可以考虑只选取一部分输入特征作为正式的输入特征，在 [10, 1] 详细地讨论了各种特征选取的方法，Dietterich 等在 [1] 中指出把特征选取和一般的集成学习算法结合起来以提高集成学习算法的学习规模，文中他们还给出了一些把特征选取跟集成学习结合起来的方法。

四、 集成学习的分析

1、 Bias-Variance 分解分析

Bias plus Variance 分解分析法由 [47] 提出，在 [46] 和 [10] 中对这种分析方法进行了介绍。Bias plus Variance 分解分析法把一个基于特定目标函数和训练集规模的分类错误分解成以下三项 [46]：

a) 一个偏置项（ bias）：衡量学习到的分类器的平均效果与实际目标函数的差距；

b) 一个容差项（ variance）：衡量各个基本学习器之间的差别；

c) 一个衡量使用最优贝叶斯分类器来近似目标函数的时候的最小分类错误程度。

Leo Breiman 等认为 Bagging 和 Boosting 都能够通过减少容差项来降低错误率 [32]，而 Freund 等认为 Boosting 关注于从分类错误的实例中学习，从而还能够通过减少偏置项来降低错误率 [27]。Eric Bauer 等还进一步指出不仅 Boosting 在某些训练集上能够减少偏置项，而且在这些训练集上 Bagging 也能够减少偏置项 [23]。

关于 Bias plus Variance 分解分析法的更详细的说明可以参考 [47, 46, 10]。

2、 实验比较

[46, 23, 25] 等对各种流行的集成学习方法进行了实验比较，根据实验结果他们指出以下的一些结论：

a) Bagging 方法得到的集成分类器几乎总是比单个分类器效果要好；

b) Boosting 方法得到的集成分类器的效果可以明显优于 Bagging 方法得到的分类器和单个分类器；

c) 然而对于某些数据集 Boosting 的结果有可能比单个分类器的性能还低；

d) 使用随机化人工神经网络的初始权值来进行集成的方法往往能够取得和 Bagging 同样好的效果；

e) Boosting 方法的效果至少有一定程度依赖于数据集，而 Bagging 方法对数据集的依赖却没有那么明显。

具体的实验方法和详细实验数据可以参考 [46, 23, 25]。

五、 结论

本文介绍了集成学习的基本概念、基本原理和主要的集成学习方法。虽然本文中介绍了不少关于集成学习的情况，但是为免导致本文内容繁杂结构零乱，还有相当多的内容没有覆盖到或者只是简单进行了介绍，关于这些内容更详细的说明可以参考 [37, 33, 32, 31, CAR04, ESP04, KSM02] 等以及本文后附录的参考文献。

评价集成学习方法的时候一般有三条基本标准 [10]：学习精度、学习效率和基本分类器的多样性。前面介绍的 Bagging、AdaBoost、随机扰动等方法都是目前集成学习中非常常用并且被实践证明有效的方法，基于决策树或者是人工神经网络的集成学习更是被应用到各种

场合。

集成学习现在还是机器学习中一个热门的研究方向，诸如 AdaBoost 等集成学习方法背后的具体机理，如何提高集成学习的学习效果，如何提高集成学习解决问题的规模，各种集成学习方法之间的具体关系，集成学习和数据复杂性之间的关系等等都是尚未完全解决的问题[1, 9, 2, 30, 16, 5] 。

六、 参考文献

[1] T.G. Dietterich. Machine Learning Research: Four Current Directions. AI Magazine, 18(4):97--136, 1997

[2] Angluin, D. (1992). Computational learning theory: survey and selected bibliography. Proceedings of the twenty-fourth annual ACM symposium on Theory of computing (pp. 351--369). New York: ACM Press.

[3] Thomas G. Dietterich. Machine Learning. ACM Computing Surveys, 28(4es), December 1996. Available on the World-Wide Web at http://www.acm.org/pubs/citations/journals/surveys/1996-28-4es/a3-dietterich/

[4] Eric M., Dennis D.: Machine Learning for Science: State of the Art and Future Prospects. Science, Vol.293, 2001. pp2051-2055.

[5] Thomas G.D.: Fundamental Experimental Research in Machine Learning. Available at http://web.engr.oregonstate.edu/~tgd/publications/

[6] Tom M. Mitchell: Machine Learning. McGraw Hill, 1997.

[7] Thomas G. Dietterich. Ensemble learning. In The Handbook of Brain Theory and Neural Networks, Second Edition, 2002.

[8] T.G. Dietterich. Ensemble Methods in Machine Learning. In Multiple Classier Systems, Cagliari, Italy, 2000.

[9] G. Valentini and F. Masulli, "Ensembles of learning machines," in Neural Nets WIRN Vietri-02, Series Lecture Notes in Computer Sciences, M. Marinaro and R. Tagliaferri, Eds.: Springer-Verlag, Heidelberg (Germany), 2002,. Invited Review.

[10] ShiXin Yu: Feature Selection and Classifier Ensembles: A Study on Hyperspectral Remote Sensing Data (2003) Available at http://143.129.203.3/visielab/theses/shixin/thesis_yu.pdf

[11] H.S. Seung, M. Opper, and H. Sompolinsky. Query by committee. In Proceedings of the Fifth Workshop on Computaional Learning Theory, pages 287--294, San Mateo, California, 1992. Morgan Kaufmann.

[12] Kittler, J., Hatef, M., Duin, R. P., and Matas, J. 1998. On Combining Classifiers. IEEE Trans. Pattern Anal. Mach. Intell. 20, 3 (Mar. 1998), 226-239.

火龙果•整理
uml.org.cn

[13] Wolpert, D.H. (1992), Stacked Generalization, Neural Networks, Vol. 5, pp. 241-259, Pergamon Press.

[14] Ricardo V., Youssef D.: A perspective view and survey of meta-learning. Artificial Intelligence Review, 18(2):77 – 95, 2002.

[15] William W. Cohen, Vitor R. Carvalho: Stacked Sequential Learning. IJCAI 2005: 671-676

[16] Dennis Bahler and Laura Navarro. Methods for Combining Heterogeneous Sets of Classifiers. In 17th Natl. Conf. on Artificial Intelligence (AAAI), Workshop on New Research Problems for Machine Learning, 2000.

[17] Kearns, M., & Mansour, Y. (1996). On the boosting ability of top-down decision tree learning algorithms. In Proceedings of the Twenty-Eighth Annual ACMSymposium on the Theory of Computing.

[18] Z.-H. Zhou, J. Wu, and W. Tang. Ensembling Neural Networks: Many Could Be Better than All. Artificial Intelligence, 2002, 137(1-2): 239-263.

[19] Z.H. Zhou, J.Wu, W.Tang, and Z.Q. Chen. Selectively Ensembling Neural Classifiers. In International Joint Conference on Neural Networks, volume 2, 2002.

[20] Hansen, L. K. and Salamon, P. 1990. Neural Network Ensembles. IEEE Trans. Pattern Anal. Mach. Intell. 12, 10 (Oct. 1990), 993-1001.

[21] Breiman, L. (1996a), Bagging Predictors, Machine Learning, Vol. 24, No. 2, pp. 123-140.

[22] Dietterich, T. G. (2000). An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization. Machine Learning, 40, 139--157.

[23] E. Bauer and R. Kohavi. An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants. To appear in Machine Learning (available at: http://reality.sgi.com/ronnyk/vote.ps.gz), 1998.

[24] Oza, N. C. and Russell, S. 2001. Experimental comparisons of online and batch versions of bagging and boosting. In Proceedings of the Seventh ACM SIGKDD international Conference on Knowledge Discovery and Data Mining (San Francisco, California, August 26 - 29, 2001). KDD '01. ACM Press, New York, NY, 359-364.

[25] Richard Maclin and David Opitz. An empirical evaluation of bagging and boosting. In Proceedings of the Fourteenth National Conference on Artificial Intelligence, pages 546-551, 1997.

[26] Robert E. Schapire. The Strength of Weak Learnability. Machine Learning, 5(2):197--227, 1990.

[27] Freund, Y. & R.E. Schapire (1996), Experiments with a New Boosting Algorithm, in Proceedings of the Thirteenth International Conference on Machine Learning, pp.

148156, Morgan Kaufmann.

[28] R. E. Schapire. The Boosting Approach to Machine Learning: An Overview. In MSRI Workshop on Nonlinear Estimation and Classification, Berkeley, CA, Mar. 2001. See http://stat.bell-labs.com/who/cocteau/nec/ and http://www.research.att.com/~schapire/boost.html .

[29] Y. Freund and R. Schapire, Discussion of the Paper ‘Arcing Classifiers’ by Leo Breiman, The Annals of Statistics, vol. 26, no. 3, pp. 824-832, 1998.

[30] Y. Freund and R. Schapire. A Short Introduction to Boosting, J. Japan. Soc. for Artif. Intel. 14(5) (1999), 771-780. 11

[31] Breiman, L. (1998). Arcing Classifiers. The Annals of Statistics, 26(3), 801--849.

[32] Breiman,L. (1996) "Bias, Variance, and Arcing Classifiers", Technical Report 460, Statistics Department, University of California

[33] Alexey T., Mykola P., Padraig C.: Sequential Genetic Search for Ensemble Feature Selection. In Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI-05)

[34] Hiroshi Mamitsuka. "Empirical Evaluation of Ensemble Feature Subset Selection Methods for Learning from a High-Dimensional Database in Drug Design," bibe, p. 253, Third IEEE Symposium on BioInformatics and BioEngineering (BIBE'03), 2003.

[35] Thomas G. Dietterich and Ghulum Bakiri. Solving Multiclass Learning Problems via Error-Correcting Output Codes. Journal of Artificial Intelligence Research, 2:263--286, January 1995.

[36] Xu, L., et al, Methods of Combining Multiple Classifiers and Their Applications to Handwriting Recognition, IEEE Transactions on Systems, Manand Cybernetics (1992), May/June.

[37] Al-Ani, A. and Deriche, M. (2002) "A New Technique for Combining Multiple Classifiers using The Dempster-Shafer Theory of Evidence", JAIR, Volume 17, pages 333-361.

[38] K. Chen, L. Wang, and H. Chi. Methods of combining multiple classifiers with different features and their applications to text-independent speaker identification. Int. J. of Pattern Recognition and Artificial Intelligence, 1997.

[39] M. Kamel and N. Wanas, "Data Dependence in Combining Classifiers", Multiple Classifier Systems, 4th international workshop, MCS2003.

[40] Saso D?eroski, Bernard ?enko: Is Combining Classifiers with Stacking Better than Selecting the Best One? Machine Learning, Volume 54 , Issue 3, Pages: 255 - 273 , 2004

[41] Zhipeng Xie, Hsu, W., Li Lee, M.: Mode committee: a Novel Ensemble Method by

Clustering and Local Learning. Tools with Artificial Intelligence, 2004. ICTAI 2004. 16th IEEE International Conference on 15-17 Nov. 2004 Page(s):628 - 633

[42] Ji C. & Ma S. (1997), Combinations of Weak Classifiers. IEEE Transaction on Neural Networks, 8(1):32--42.

[43] Alan Fern and Robert Givan. Online Ensemble Learning: An Empirical Study. In Proceedings of the Seventeenth International Conference on Machine Learning, 2000.

[44] Tsymbal, A., Puuronen, S.: Bagging and Boosting with Dynamic Integration of Classifiers, Principles of Data Mining and Knowledge Discovery, Proc. PKDD 2000.

[45] Bennett, K. P., Demiriz, A., and Maclin, R. 2002. Exploiting unlabeled data in ensemble methods. In Proceedings of the Eighth ACM SIGKDD international Conference on Knowledge Discovery and Data Mining (Edmonton, Alberta, Canada, July 23 - 26, 2002). KDD '02. ACM Press, New York, NY, 289-296.

[46] Opitz, D. and Maclin, R. "Popular Ensemble Methods : an Empirical Study". J. Art. Intell. Research, 11, 169-198,1999.

[47] S. Geman, E. Bienenstock, and R. Doursat. 1992. Neural networks and the bias/variance dilemma. Neural Computation, 4:1--58.

[48] Esposito, R., Saitta, L.: A Monte Carlo Analysis of Ensemble Classification. In Proc. 23th International Conference on Machine Learning, Morgan Kaufmann (2004)

[49] YongSeog Kim, Street, W.N., Menczer, F.: Meta-evolutionary ensembles. Neural Networks, 2002. IJCNN '02. Proceedings of the 2002 International Joint Conference on Volume 3, 12-17 May 2002 Page(s):2791 - 2796

[50] Nigel Duffy, David P. Helmbold: Boosting Methods for Regression,

[51] Dong, Y. and Han, K. 2005. Boosting SVM classifiers by ensemble. In Special interest Tracks and Posters of the 14th international Conference on World Wide Web (Chiba, Japan, May 10 - 14, 2005). WWW '05. ACM Press, New York, NY, 1072-1073.

[52] F ü rnkranz, J. 2002. Pairwise Classification as an Ensemble Technique. In Proceedings of the 13th European Conference on Machine Learning (August 19 - 23, 2002). T. Elomaa, H. Mannila, and H. Toivonen, Eds. Lecture Notes In Computer Science, vol. 2430. Springer-Verlag, London, 97-110.

[53] Miller, D.J., Lian Yan: Some Analytical Results on Critic-Driven Ensemble Classification. Neural Networks for Signal Processing IX, 1999. Proceedings of the 1999 IEEE Signal Processing Society Workshop 23-25 Aug. 1999 Page(s):253 - 262.

[54] R?tsch, G., Onoda, T., and M ü ller, K. 2001. Soft Margins for AdaBoost. Mach. Learn. 42, 3 (Mar. 2001), 287-320.

[55] N. Ueda, Optimal Linear Combination of Neural Networks for Improving Classification Performance, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (2000): 207-215.

火龙果•整理
uml.org.cn

[56] A. Krogh and J. Vedelsby. "Neural Network Ensembles, Cross Validation and Active Learning". In D.S. Touretzky G. Tesauro and T. K. Leen, editors, Advances in Neural Information Processing Systems, volume 7, pages 231--238. The MIT Press, 1995.

[57] Oliveira, L.S., Sabourin, R., Bortolozzi, F., Suen, C.Y.: Feature selection for ensembles:a hierarchical multi-objective genetic algorithm approach. In Proceedings of Seventh International Conference on Document Analysis and Recognition, 2003. pp. 676- 680.

[58] Y. Crammer and Y. Singer. On the learnability and design of output codes for multiclass problems. In Proceedings of thr Thirteenth Annual Conference on Computational Learning Theory, 2000.

[59] Huang, K., Yang, H., King, I., and Lyu, M. R. 2004. Learning large margin classifiers locally and globally. In Proceedings of the Twenty-First international Conference on Machine Learning (Banff, Alberta, Canada, July 04 - 08, 2004). ICML '04, vol. 69. ACM Press, New York, NY, 51.

[60] J. Chan, C. Huang and R. Defries "Enhanced algorithm performance for land cover classification from remotely sensed data using bagging and boosting," IEEE Transactions of Geoscience and Remote Sensing, vol. 39, pp. 693-695, 2001.

[61] Jin, R., Huan Liu: A Novel Approach to Model Generation for Heterogeneous Data Classification. The Nineteenth International Joint Conference on Artificial Intelligence (IJCAI 2005)Edinburgh, Scotland, 30 July - 5 Aug., 2005.

[62] Caruana, R., Niculescu-Mizil, A., Crew, G., and Ksikes, A. 2004. Ensemble selection from libraries of models. In Proceedings of the Twenty-First international Conference on Machine Learning (Banff, Alberta, Canada, July 04 - 08, 2004). ICML '04, vol. 69. ACM Press, New York, NY, 18.

[63] M.T. Rosenstein, Z. Marx, L.P. Kaelbling, and T.G. Dietterich. To appear, NIPS 2005 Workshop on Inductive Transfer: 10 Years Later, 2005.

[64] Blum, A., & Mitchell, T. (1998). Combining labeled and unlabeled data with co-training.

[65] Opitz, D. submitted. Feature selection for ensembles. In Proceedings of the Sixteenth National Conference on Artificial Intelligence. Randic, M. 1975. On characterization of molecular branching. Journal of American Chemical Society 97:6609--6615.

[66] Prem Melville and Raymond J. Mooney: Diverse Ensembles for Active Learning. Proceedings of the 21st International Conference on Machine Learning (ICML-2004), pp. 584-591, Banff, Canada, July 2004.

[67] Prem Melville: Creating Diverse Ensemble Classifiers. Ph.D. proposal, Department of Computer Sciences, University of Texas at Austin, October 2003. 54 pages. Also appears as Technical Report UT-AI-TR-03-306, Artificial Intelligence Lab, University of Texas at Austin, December 2003.

火龙果•整理
uml.org.cn

[68] Carlos H.E.,Mercedes F.R., Mamen O.G.: A comparison of ensemble methods for Multilayer Feedforward networks. In Proc. of International Joint Conference On Neural Networks (IJCNN'2003). USA: 20-07-2004

[69] Alexander B., Victor E., Eugene T.: Boosting Flexible Learning Ensembles with Dynamic Feature Selection. NIPS2003. Available at http://clopinet.com/isabelle/Projects/NIPS2003/.

[70] Isabelle A., Stephan B.: Ranking and Selection of Cases with Decision Trees: a Geometric Method that Preserves Intelligibility. In Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI-05)

[71] Hema R., Omid M., Rosie J.: InterActive Feature Selection. In Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI-05)

[72] Peter F., Shaomin W.: Repairing Concavities in ROC Curves. In Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI-05)

[73] Jonathan L., Tanzeem C., Nicky K., Gaetano B., Blake H.: A Hybrid Discriminative: Generative Approach for Modeling Human Activities. In Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI-05)

[74] Shay C., Eytan R., Gideon D.: Feature Selection Based on the Shapley Value.In Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI-05)

[75] Yuhong G., Russ G., Dale S., Learning Coordination Classifiers. In Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI-05)

[76] Meir R., Ratsch G.: An Introduction to Boosting and Leveraging. LECTURE NOTES IN ARTIFICIAL INTELLIGENCE 2600: 118-183, 2002.

[77] 马少平，朱小燕：人工智能．清华大学出版社，2004.