Stacking 学习与一般集成方法的比较研究

鲁 莹，郑少智
（暨南大学经济学院，广州 510632）

摘要：集成学习（ensemble learning）因通过组合多个学习器实现更强的泛化能力而被广泛使用。目前一般的集成方法如 AdaBoost、Bagging 等均是基于一种算法，而 Stacking 集成是基于多种算法。本文针对分类集成问题，基于朴素贝叶斯（naive Bayes，NB）、logistic 回归、*k* 最近邻（*k* nearest neighbor，KNN）、决策树（C4.5）和规则学习（rule learner）5 种基分类器，构建 Stacking 学习框架，并与 AdaBoost、Bagging、随机森林（random forests，RF）、投票表决及交叉验证下的最佳分类器这 5 种方法进行比较。通过 2 组模拟数据和 36 组真实数据的实证分析发现，Stacking 在所有方法中表现最好，具有最强的泛化能力且更适合大样本的情况。

关键词：人工智能；集成学习；组合；Stacking；分类器；泛化能力
中图分类号：TP181 **文献标识码**：A **文章编号**：1674-2850(2018)04-0372-08

A comparative study of Stacking learning and general ensemble methods

LU Ying, ZHENG Shaozhi
(*College of Economics*, *Jinan University*, *Guangzhou* 510632, *China*)

Abstract: Ensemble learning has been widely used because of the great generalization ability by combining multiple learners. The general ensemble methods, such as AdaBoost and Bagging, are usually based on the same algorithm, but Stacking ensemble is based on different algorithms. Aiming at classification ensemble problems, this paper constructs a learning framework for Stacking consisted of 5 types of base classifiers, which are naive Bayes (NB), logistic regression, *k* nearest neighbor (KNN), decision tree and rule learner. And comparasion of Stacking with 5 other methods, AdaBoost, Bagging, random forests (RF), voting plan and selecting the optimal classifier using cross validation, is made. The experiments of 2 simulated and 36 real datasets are conducted and the results show that Stacking is the best among all other methods according to the highest generalization ability and more suitable for large sample cases.

Key words: artificial intelligence; ensemble learning; combination; Stacking; classifier; generalization ability

0 引言

分类问题是统计学习研究的一大基本问题。在有监督的学习条件下，有很多可供选择的分类算法，例如判别分析、logistic 回归、C4.5、NB、KNN、rule learner 等。而随着大数据时代的到来，数据挖掘难度逐渐提高，单一的模型已不能满足需求，集成学习应运而生。集成学习是组合一系列基学习器（学习器用于分类则称为分类器，本文对此不严格区分）的结果来实现对未知样本的预测的一种方法。DIETTERICH[1]曾指出学习器集成比单个学习器效果显著的三个主要原因：一是学习任务的假设空间一般很大，使用单个学习器通常不能学习到足够的信息；二是弱学习器的学习过程可能存在缺陷；三是单个

集成学习效果好的三个原因

作者简介：鲁莹（1993—），女，硕士研究生，主要研究方向：数据挖掘与统计分析
通信联系人：郑少智，教授，主要研究方向：统计学. E-mail: tzhengsz@jnu.edu.cn

学习器学习到的假设空间可能并不真实，而通过结合多个学习器可以尽可能学习到真实的假设空间。有理论研究表明，组合多个弱学习器通常可获得比单一学习器显著优越的泛化性能[2]，目前许多领域的研究也为此提供了一些实证支持。

分类集成中，由于解决同一分类问题时，基分类器之间不免存在较大相关性，表现为多样性不足，这也是目前集成学习在实现更强的泛化能力时所要解决的问题。根据 POLIKAR[3]的研究，实现多样性有 4 种方式：一是使用不同的训练集；二是使用不同的模型参数；三是使用不同的特征空间；四是使用组合异质分类器。一般的集成方法实现多样性通常基于前 3 种方式，例如 AdaBoost 和 Bagging 使用自助抽样（bootstrap）构造不同的训练集，RF 使用不同的随机特征空间等。这些方法的共性是基于同一算法集成，因而产生的多个基分类器是同质的，且分类器的组合方式一般为投票表决。Stacking 则与之不同，它是基于不同算法产生的多个异质分类器，在多个分类器的预测上进行再次学习来实现组合的集成方法。由于其理论研究较为棘手，至今也没有形成一种被广泛接受的架构。本文提出一种基于 5 种异质分类器的 Stacking 学习框架，选择 AdaBoost、Bagging、RF、5 个基分类器的投票表决（记为 Vote）及交叉验证下的最佳基分类器（记为 CV-Best）5 种方法作为对比，使用平均相对提升度和配对样本 t 检验来评价不同方法之间的差异并做出解释。

1 AdaBoost、Bagging 与 RF

AdaBoost[4]是 Boosting 算法家族中最著名的代表，指一种将弱学习器提升为强学习器的方法。AdaBoost 的工作机制为：首先从训练集中学习得到一个基学习器；再根据基学习器的表现对训练样本的分布进行调整，使前一个基学习器预测错误的样本在下一次学习的训练集中以更大的概率出现；然后基于调整后的训练集来学习下一个基学习器，如此重复进行直到基学习器的个数达到指定的数目为止；最后根据各基学习器的表现加权得到最终的结果。AdaBoost 算法的基学习器是串行产生的，各基学习器之间紧密相关，但每次学习不断提升的精度弥补了这一缺陷。

Bagging[5]与 AdaBoost 不同，从增强基学习器多样性的角度出发，每次均采用自助抽样法从原始数据集中抽取部分数据样本进行学习得到基学习器，然后通过进行简单投票或简单平均法得到最终的结果。由于基学习器是并行产生的，所以各基学习器之间不存在很强的依赖关系，因而其多样性一般比 AdaBoost 方法高。

RF[6]是 Bagging 的一种变体，在 Bagging 的基础上通过加入输入属性扰动，即每次在选择节点的划分属性时都是在样本所有属性中随机选择的部分属性中进行，这在理论上有助于提高基学习器的多样性。在一些实证中发现，RF 取得了比 Bagging 和 AdaBoost 更好的结果[7]。

2 Stacking

Stacking 是 stacked generalization[8]的缩写，一般分为两层学习过程：首先在原始数据集上分别学习多种算法，然后在所有预测结果和原始样本真实值组成的数据集上再次训练。假设给定包含 n 个样本的数据集 $\mathcal{L}=\{(\boldsymbol{x}_i,y_i),i=1,2,\cdots,n\}$，其中，$\boldsymbol{x}_i$ 为第 i 个样本的属性向量，y_i 为第 i 个样本的真实值，一共有 t 个不同算法的学习器。为防止过拟合问题，在构造第二层数据集时运用交叉验证的思想，具体做法是将原始数据集随机分成数目大致相等的 K 个部分 $\mathcal{L}_1,\mathcal{L}_2,\cdots,\mathcal{L}_K$，定义 $\mathcal{L}_k$ 和 $\mathcal{L}=\mathcal{L}^{-k}-\mathcal{L}_k(k=1,2,\cdots,K)$ 分别为第 k 折交叉验证的测试集和训练集，在训练集上分别训练 t 个算法，得到 t 个模型，然后在测试集上进行预测得到预测值，记为 Z_{ij}，表示第 i 个样本经由第 j 个算法的预测。这个过程重复 K 次，则原始数据集的每一个样本均有 t 个与之对应的预测值，这些预测值与对应样本的真实值组成第二层数据集

$\mathcal{L}_{cv}=\{(Z_{i1},Z_{i2},\cdots,Z_{it},y_i),i=1,2,\cdots,n\}$. 在 $\mathcal{L}_{cv}$ 上再次学习得到最终模型，至此一般的 Stacking 学习过程结束。为更好地理解 Stacking 学习过程，以三分类问题为例对 Stacking 学习框架下的数据集进行描述，如图 1 所示，图 1a 和图 1b 分别为原始数据集和第二层学习数据集。

样本	属性向量	类别
1	$\boldsymbol{x}_1$	2
2	$\boldsymbol{x}_2$	3
3	$\boldsymbol{x}_3$	2
⋮	⋮	⋮
n	$\boldsymbol{x}_n$	1

a

样本	分类器 1	分类器 2	…	分类器 t	类别
1	Z_{11}	Z_{12}	…	Z_{1t}	2
2	Z_{21}	Z_{22}	…	Z_{2t}	3
3	Z_{31}	Z_{32}	…	Z_{3t}	2
⋮	⋮	⋮		⋮	⋮
n	Z_{n1}	Z_{n2}	…	Z_{nt}	1

b

样本	分类器 1	分类器 2	…	分类器 t	类别=2
1	P_{112}	P_{122}	…	P_{1t2}	1
2	P_{212}	P_{222}	…	P_{2t2}	0
3	P_{312}	P_{322}	…	P_{3t2}	1
⋮	⋮	⋮		⋮	⋮
n	P_{n12}	P_{n22}	…	P_{nt2}	0

c

图 1 Stacking 学习框架下的数据集示例图

Fig. 1 Figures of datasets in Stacking learning framework

a—原始数据集；b—Stacking 第二层学习数据集；c—StackingC 第二层学习数据集（类别 2）

a-Original dataset; b-Dataset for the second step in Stacking learning; c-Dataset for the second step in StackingC learning

WOLPERT[8]最早提出了 Stacking 思想，介绍了 Stacking 的一般架构，并应用于现实数据中得出 Stacking 是一种估计和修正偏差的工具，可以用来减小模型的泛化误差，为集成方法研究开辟了一条新途径。其后，BREIMAN[9]将 WOLPERT 使用的留一交叉验证改为十折交叉验证，大大提高了训练的效率。此后，Stacking 方法的研究主要集中于再次学习方面，也取得了一些成果[10~11]。而在解决多分类问题方面，StackingC[12]是一种比较好的再次学习方法，该方法使用了分类概率信息和多重响应回归模型。图 1c 给出了 StackingC 思想下类别为 2 的数据集，其中 P_{it2} 表示第 i 个样本被第 t 个基分类器预测为第 2 类的概率。根据图 1 的三分类，对应了 3 个类似的数据集，只需针对各个类别进行相应处理。由图 1 可以看出，相比使用预测值，StackingC 考虑了基分类器的分类可靠性信息，有助于提高模型分类的精度。

针对分类问题，本文构建了 Stacking 学习框架，如图 2 所示。第一层学习选择了机器学习中 5 种不同的算法：NB、logistic 回归、KNN、C4.5 和 rule learner，第二层选择了 StackingC 方法。

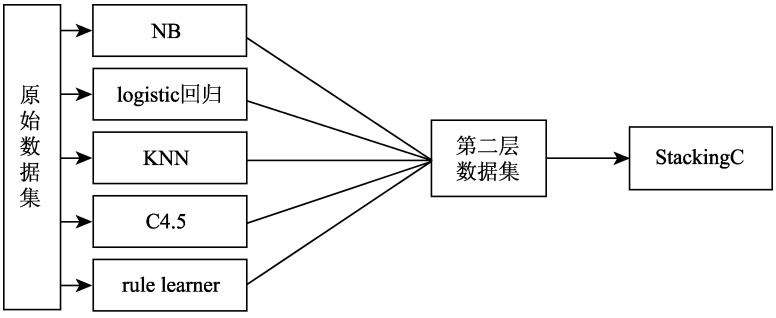

图 2 Stacking 学习框架图

Fig. 2 Diagram of Stacking learning framework

3 实验与结果

本节主要包括数据模拟和真实数据演练两部分。模拟数据有 2 个，由 R 软件产生；真实数据有 36 个，包括金融、生物、物理、医学、地质等多个领域，均来自 UCI 机器学习数据库。所有方法的性能分析部分均在 Weka[13]上完成，Weka 是新西兰怀卡托大学用 Java 开发的数据挖掘操作平台。实验选择 AdaBoost、Bagging、RF、Vote 和 CV-Best 5 种方法作为 Stacking 的对比，研究不同方法的性能差异。

3.1 评价标准

分类方法效果评价有多重标准，这里使用最常见的精确度（Accuracy）评价，即计算所有被正确分类的样本占总体的比例。分类误差（error=1−Accuracy），采用 10 次十折交叉验证进行估计，每次均采用不同的随机种子以保证每次交叉验证时不同方法使用的训练集（九折）及估计误差所用的测试集（一折）均保持一致。对于给定的数据集，每种方法的误差均通过计算 10 次十折交叉验证的平均误差作为误差估计。方法 M 的误差记作 error(M). 由于本文研究了不同方法的效果差异，因此当评价方法 M_1 在某一给定数据集上相比于方法 M_2 效果的优劣时，使用了相对提升度（relative improvement，RI）这一概念：

$$\mathrm{RI}=1-\frac{\mathrm{error}(M_1)}{\mathrm{error}(M_2)}. \tag{1}$$

显然 RI 值大于 0 时，说明方法 M_1 好于 M_2，反则否。不同的数据集可能会偏好不同的方法，为减少偶然性的发生，计算 M_1 在 s 个数据集中相比于 M_2 的平均优劣，称作平均相对提升度（average relative improvement，ARI）：

$$\mathrm{ARI}=1-\sqrt[s]{\Pi_{i=1}^{s}\frac{\mathrm{error}(M_1^{(i)})}{\mathrm{error}(M_2^{(i)})}}, \tag{2}$$

其中，$M_1^{(i)}$为方法 M_1 在第 i 个数据集上的误差。需要注意的是，ARI 并不满足对称性。为检验不同方法的性能是否存在统计意义上的显著差异，在置信度为 95%的水平上对每 2 种方法在给定数据集上的效果进行配对样本 t 检验，得到显著好、显著差或不存在显著差异 3 种结果。

3.2 模拟实验

模拟部分选择来自 BREIMAN 等[14]设计的三值分类问题波形数据集，此数据基于 3 种原始波形 $h_1(m)$、$h_2(m)$和 $h_3(m)$，如图 3a 所示。每一类别（Class1～Class3）的样本均从 3 种原始波形中选择 2 种进行随机加权组合，并加上随机干扰。具体说来，类别 j 的第 i 个样本为 $a_{ij}=\{(x_{(i,1,j)},x_{(i,2,j)},\cdots,x_{(i,21,j)}),y_i\}$，其中 $i=1,2,\cdots,n$，一共 n 个样本，$y_j=j$，$j=1,2,3$ 为类别值，$x_{(i,m,j)}$为类别 j 的第 i 个样本的第 m 个取值，$i=1,2,\cdots,n$，$m=1,2,\cdots,21$，$j=1,2,3$. 为产生类别 1 的数据，独立生成一个服从 0～1 均匀分布的随机数 μ 和 21 个服从标准正态分布 $N(0,1)$的随机干扰项 $\varepsilon_1,\varepsilon_2,\cdots,\varepsilon_{21}$，设定

$$x_{(i,m,1)}=\mu h_1(m)+(1-\mu)h_2(m)+\varepsilon_m,\ m=1,2,\cdots,21. \tag{3}$$

重复上述过程，分别进行如下设定生成类别 2 和类别 3 的属性值：

$$x_{(i,m,2)}=\mu h_1(m)+(1-\mu)h_3(m)+\varepsilon_m,\ m=1,2,\cdots,21, \tag{4}$$

$$x_{(i,m,3)}=\mu h_2(m)+(1-\mu)h_3(m)+\varepsilon_m,\ m=1,2,\cdots,21. \tag{5}$$

三类别使用相等的先验概率产生相同数目的样本，共同组成目标数据集 wave1，图 3b 给出了三类别样本的波形图。同时为使模拟的数据更加接近现实生活中产生的数据，在 wave1 数据的基础上增加了 19 个

服从标准正态分布的随机属性变量，得到 40 维属性的数据集 wave2.

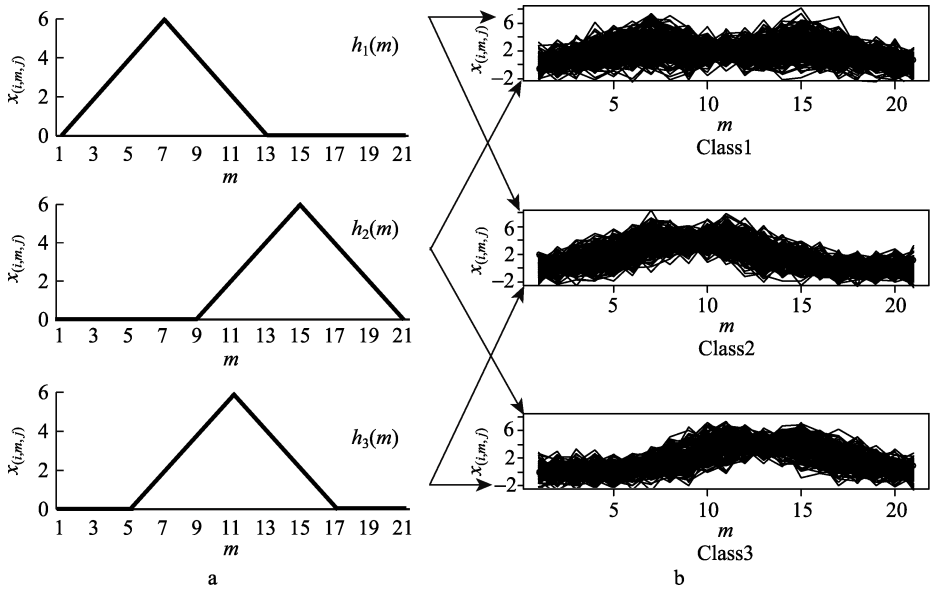

图 3 三类别模拟数据 wave1 示意图

Fig. 3 Figures of the three categories simulated dataset wave1

a—3 种原始波形；b—三类别样本的波形图

a-3 original waves; b-Three generated waves

以其他 5 种方法分别作为 M_2，计算 Stacking 的 RI 和 ARI，如表 1 所示。数据表明，Stacking 方法在 2 个模拟数据集上的效果比较明显：最差的结果（相比于 RF 的效果）均有超过 10%的 RI. 为进一步探究方法之间的差异，对比 Stacking 分别对其他 5 种方法的性能进行配对 t 检验，结果如表 2 所示。表 2 中的数据表明，Stacking 在 2 个模拟数据集上的效果均显著好于其他方法。综合表 1 和表 2 的结果可知，增加数据的无关属性干扰确实会降低方法的效果，但程度不明显；集成方法比单一模型方法效果好；Stacking 在模拟数据中的效果确实不错，比一般的集成方法效果好。为探究这种方法在现实生活中是否也能达到显著的效果，本文接下来对此进行了检验。

表 1 模拟数据方法 RI 结果表（%）

Tab. 1 Results of the simulated data on the RI (%)

模拟数据集	CV-Best	AdaBoost	Bagging	RF
wave1	44.49	29.29	27.59	13.28
wave2	46.46	28.76	26.76	10.29
ARI	45.48	29.03	27.17	11.80

表 2 模拟数据分类精确度（%）和显著性 *t* 检验表

Tab. 2 Results of the simulated data classification on the Accuracy (%) and paired *t*-test

模拟数据集	Stacking	CV-Best	AdaBoost	Bagging	RF
wave1	87.01	76.60*	81.63*	82.06*	85.02*
wave2	86.75	75.25*	81.40*	81.91*	85.23*
汇总	v/ /*	0/0/2	0/0/2	0/0/2	0/0/2

注：“*”表示 Stacking 方法显著好；“v”表示 Stacking 方法显著差；“v/ /*”分别表示 Stacking 相较于其他方法显著差、无显著差异、显著好的数据集个数

3.3 真实数据实验

UCI 机器学习库在世界范围内作为机器学习数据集的主要来源而被广泛使用，其中包含了大量不同领域的调查或实验数据集。这里选取了学习库中 36 个分类数据集，基本资料如表 3 所示。数据集的样本数跨度从少于 100 到超过 3 000，类别数目由 2 至 19 不尽相同；数据集的属性既包括连续型，也包括名义型，比模拟数据更加符合现实要求。分别计算每 2 种方法在 36 个数据集上的 ARI，结果如表 4 所示，表中数据表示对应纵列方法在相应横行方法上的结果。结果显示，Bagging 在其他方法上的 ARI 均为负，说明其他方法的效果均比 Bagging 好；Stacking 均为正，说明其表现均优于其他方法；中间 CV-Best、AdaBoost、RF、Vote 的 ARI 效果逐渐变好。这表明以 ARI 作为分类效果的衡量时，Stacking 体现了优秀的性能表现，而其他方法尤其是 Bagging 则逊色很多。

表 3 36 个数据集基础信息表

Tab. 3 Basic information of the 36 datasets

编号	数据集名称	样本	类别	属性	编号	数据集名称	样本	类别	属性
1	Balance	625	3	4/0	19	Heart	270	2	7/17
2	Car	1 728	4	0/6	20	Patient	90	3	0/8
3	Banknote	1 372	2	4/0	21	Australian	690	2	6/8
4	Mammographic	961	2	5/0	22	BLOGGER	100	2	0/5
5	Tic-Tac-Toe	958	2	0/9	23	Breast	699	2	9/0
6	Blood	748	2	4/0	24	Climate	540	2	18/0
7	Bench	208	2	60/0	25	Voting	435	2	0/16
8	German	1 000	2	7/13	26	Hepatitis	155	2	6/13
9	Image	2 310	7	19/0	27	Ionosphere	351	2	34/0
10	Musk	476	2	166/0	28	Iris	150	3	4/0
11	Diabetes	768	2	8/0	29	Planning	182	2	12/0
12	QSAR	1 055	2	41/0	30	Bankruptcy	250	2	0/6
13	Soybean	305	19	0/35	31	seeds	210	3	7/0
14	Yeast	1 484	10	8/0	32	SPECT Heart	267	2	0/22
15	Cardioto	2 126	3	21/0	33	User	403	4	5/0
16	Chess	3 196	2	0/36	34	Vertebral	310	4	6/0
17	Dermatology	366	6	1/33	35	Wine	178	3	13/0
18	Ecoli	336	8	7/0	36	Zoo	101	7	0/16

表 4 不同方法之间的 ARI 对比表（%）

Tab. 4 Comparision among different methods on ARI (%)

方法	CV-Best	AdaBoost	Bagging	RF	Vote	Stacking
CV-Best	—	10.22	−11.25	15.76	17.37	20.70
AdaBoost	−11.38	—	−23.91	6.18	7.97	11.68
Bagging	10.11	19.30	—	24.28	25.73	28.72
RF	−18.71	−6.58	−32.07	—	1.91	5.86
Vote	−21.03	−8.66	−34.65	−1.95	—	4.03
Stacking	−26.10	−13.22	−40.29	−6.23	−4.19	—

为进一步了解分类精度绝对值的不同是否存在统计上的显著性，同样分别以 6 种方法逐一作为参照，计算其余 5 种方法在每组 36 次对比实验中的结果，如表 5 所示。表中横行表示参照方法，纵列表示实验

方法，每个数据单元格包含 2 个数字含义：实验方法分类精度超过参照方法的次数和配对 t 检验表明显著高于参照方法的次数（括号中的数字）。例如横行为 CV-Best，纵列为 Stacking，两者交叉的单元格为 34（14），表示 Stacking 在 36 个数据集中的分类精度有 34 次高于 CV-Best，其中 14 次显著。综合来看，每种方法一共实验对比了 180 次，表 5 中最后一行给出了实验方法赢的次数、显著的次数分别占 180 次实验的比率。单纯从精度绝对比较出发，CV-Best、AdaBoost、Bagging、RF、Vote 和 Stacking 优于其他方法的次数分别为 45，76，46，112，117，144 次，分别占 180 次对比的 0.25、0.42、0.26、0.62、0.65 和 0.80，表明 6 种方法中后 3 种效果优于前 3 种，其中 Stacking 性能最佳，在显著性比率上也更胜一筹。

表 5　不同方法之间的效果及检验汇总表

Tab. 5　Summary of performances and test results on different methods

方法	CV-Best	AdaBoost	Bagging	RF	Vote	Stacking
CV-Best	—	23（10）	20（5）	27（12）	31（15）	34（14）
AdaBoost	13（2）	—	15（6）	26（5）	23（8）	27（8）
Bagging	16（6）	21（9）	—	30（9）	32（10）	35（11）
RF	9（2）	10（3）	6（4）	—	20（5）	23（6）
Vote	5（0）	13（1）	4（0）	16（1）	—	25（3）
Stacking	2（0）	9（0）	1（0）	13（0）	11（0）	—
合计	45（10）	76（23）	46（15）	112（27）	117（38）	144（42）
比率	0.25（0.06）	0.42（0.13）	0.26（0.18）	0.62（0.15）	0.65（0.21）	0.80（0.23）

4　结论

本文构建了一个 Stacking 学习框架，并与其他 5 种方法的效果进行比较。在最初 2 个模拟数据集上，Stacking 表现出了非常明显的卓越性能：ARI 均有较大提高，同时分类效果的差异在 2 个数据集上均显著高于其他方法。在 36 个真实数据的分析中，Stacking 也表现出了比较好的性质：180 次全数据全方法的对比实验中精度比较达到 80%的赢面，显著性也最高。综合分析，6 种方法的效果从低到高排序依次为 CV-Best、Bagging、AdaBoost、RF、Vote 和 Stacking，这从各自的算法思想中不难解释。CV-Best 方法借助交叉验证选择了 5 种基分类器中表现最佳的一种，其本质是单一模型，因此性能劣于其他多模型组合方式；Bagging、AdaBoost 和 RF 都是组合多个树模型，但 AdaBoost 的串行算法每次运算均能提高树的精度，最终组合的方式也是根据树的表现进行加权，对比 Bagging 的简单投票组合性能更优，而 RF 由于加入了随机属性选择而极大地增加了树的多样性，从而达到了更好的效果；Vote 是在 5 种基分类器中投票，平均所有基分类器的结果，效果优于 RF，说明算法的改变比随机属性选择能够实现基分类器更高程度的多样性；Stacking 效果最佳，相比于 Vote，Stacking 在结合基分类器上舍弃了简单投票的方式，采取再次学习方式，构造了一个复杂的学习过程，进而学习到更多的信息。因此，Stacking 是基于不同算法和二次学习两方面原因而实现最优泛化效果的。此外，分析 Stacking 显著好的数据集特征发现，Stacking 方法更适合样本数较大的情况。

值得注意的是 Stacking 中基分类器只用了 5 种，虽然也囊括了算法领域常见的五大类，但在基分类器多样性程度上仍然还存在较大的提升空间，例如通过增加基分类器的种类或改变分类器的参数构造同种分类器的多个模型，然后再次学习。同时，StackingC 方法只是 Stacking 家族中的一种目前比较流行的再学习方法，从提高模型分类精确度的角度出发，再次学习方式也是值得研究的问题。

[参考文献] (References)

[1] DIETTERICH T G. Ensemble methods in machine learning[C]//Proceedings of the First International Workshop on Multiple Classifier Systems. Cagliari: Springer, 2000: 1-15.

[2] SCHAPIRE R E. The strength of weak learnability[J]. Machine Learning, 1990, 5(2): 197-227.

[3] POLIKAR R. Ensemble based systems in decision making[J]. IEEE Circuits & Systems Magazine, 2006, 6(3): 21-45.

[4] FREUND Y, SCHAPIRE R E. A decision-theoretic generalization of on-line learning and an application to boosting[J]. Journal of Computer and System Sciences, 1999, 55(1): 119-139.

[5] BREIMAN L. Bagging predictors[J]. Machine Learning, 1996, 24(2): 123-140.

[6] BREIMAN L. Random forests[J]. Machine Learning, 2001, 45(1): 5-32.

[7] 盛夏，李斌，张迪. 基于数据挖掘的上市公司信用评级变动预测[J]. 统计与决策，2016（15）：159-162.
SHENG X, LI B, ZHANG D. Credit rating changes prediction of public company credit rating changes based on data mining[J]. Statistics and Decision, 2016(15): 159-162. (in Chinese)

[8] WOLPERT D H. Stacked generalization[J]. Neural Networks, 1992, 5(2): 241-259.

[9] BREIMAN L. Stacked regressions[J]. Machine Learning, 1996, 24(1): 49-64.

[10] TING K M, WITTEN I H. Issues in stacked generalization[J]. Journal of Artificial Intelligence Research, 1999, 10(1): 271-289.

[11] DŽEROSKI S, ŽENKO B. Is combining classifiers with stacking better than selecting the best one?[J]. Machine Learning, 2004, 54(3): 255-273.

[12] SEEWALD A K. How to make stacking better and faster while also taking care of an unknown weakness[C]//Proceedings of the Nineteenth International Conference on Machine Learning. San Francisco: Morgan Kaufmann Publishers, 2002: 554-561.

[13] WITTEN I H, FRANK E, HALL M A, et al. Data mining: practical machine learning tools and techniques[M]. 3rd ed. San Francisco: Morgan Kaufmann Publisher, 2011.

[14] BREIMAN L, FRIEDMAN J H, OLSHEN R A, et al. Classification and regression trees[M]. New York: Chapman & Hall, 1984.