

Stacking 集成学习模型在混合式成绩分类预测中的应用^①



章 刘¹, 陈逸菲², 袁加伟², 裴梓权¹, 梅鹏江¹

¹(南京信息工程大学 自动化学院, 南京 210044)

²(无锡学院 自动化学院, 无锡 214105)

通信作者: 章 刘, E-mail: 1797584210@qq.com

摘 要: 针对现有单一算法模型在成绩预测时存在泛化能力不强的问题, 提出一种基于多算法融合的 Stacking 集成学习模型, 用于混合式教学中学生成绩的预测. 模型以多项式朴素贝叶斯、AdaBoost 和 Gradient boosting 为初级学习器, 逻辑斯蒂回归为次级学习器组成两层融合框架. 通过混合式教学过程中所产生的学习行为数据对模型进行验证. 实验表明, Stacking 集成学习模型在测试集上分类预测准确率达到 76%, 分别高于多项式朴素贝叶斯、AdaBoost、Gradient boosting 和逻辑斯蒂回归 4 个单一算法模型 5%、6%、9% 和 6%. 与单一算法模型相比, Stacking 集成学习模型有着较强的泛化能力, 能更好地预测学生成绩, 为混合式教学的学习预警提供参考.

关键词: 混合式教学; 成绩预测; 机器学习; Stacking; 集成模型; 集成学习

引用格式: 章刘, 陈逸菲, 袁加伟, 裴梓权, 梅鹏江. Stacking 集成学习模型在混合式成绩分类预测中的应用. 计算机系统应用, 2022, 31(7): 325-332. <http://www.c-s-a.org.cn/1003-3254/8572.html>

Application of Stacking Ensemble Learning Model in Blended Performance Classification and Prediction

ZHANG Liu¹, CHEN Yi-Fei², YUAN Jia-Wei², PEI Zi-Quan¹, MEI Peng-Jiang¹

¹(School of Automation, Nanjing University of Information Science & Technology, Nanjing 210044, China)

²(School of Automation, Wuxi University, Wuxi 214105, China)

Abstract: To tackle the problem that existing single algorithm models have poor generalization ability in performance prediction, this study proposes a Stacking ensemble learning model based on multi-algorithm fusion for the prediction of students' performance in blended teaching. The model uses polynomial naive Bayes, AdaBoost and Gradient boosting as primary learners and logistic regression as secondary learners to form a two-level fusion framework. The model is verified by the learning behavior data generated in the process of blended teaching. Experimental results show that the classification and prediction accuracy of the Stacking ensemble learning model on the test set reaches 76%, which is 5%, 6%, 9% and 6% higher than that of the four single algorithm models of polynomial naive Bayes, AdaBoost, Gradient boosting and logistic regression, respectively. Compared with these single algorithm models, the Stacking ensemble learning model has strong generalization ability, which can better predict students' performance and provide a reference for the learning warning of blended teaching.

Key words: blended teaching; performance prediction; machine learning; Stacking; ensemble model; ensemble learning

① 基金项目: 江苏省高等学校自然科学研究面上项目 (19KJB520044); 江苏省应用型本科院校建设与发展研究课题 (2019y117); 江苏省产学研合作项目 (BY2019113); 江苏省高等学校大学生创新创业训练计划 (202113982023Y)

收稿时间: 2021-09-24; 修改时间: 2021-11-08; 采用时间: 2021-11-19; csa 在线出版时间: 2022-05-30

1 前言

混合式教学是一种线上与线下相结合的新兴教学模式,既通过在智慧教学平台上发布任务,上传资源等发挥教师的主导作用,又通过学生线上自主学习、完成任务等,引导学生积极主动学习。目前正被各大高校积极推广^[1]。学生可以不受时空限制,通过多种方式登录学习平台学习,所产生的学习行为数据被平台记录保存^[2]。学习行为数据反映学生的学习状态和习惯。而日常的学习状态和习惯对最终的学习成绩有着重要影响^[3]。混合式学习对学生的自觉性要求较高,在实际实施过程中,学生的参与度受各种因素干扰而参差不齐^[4,5]。因此,如何利用平台记录的学习行为数据分析出学生学习状态,掌握学生的学习情况,引导学生积极主动学习,成为教学管理中急需解决的问题。

通过对学生学习行为数据进行挖掘和分析,可以找出隐藏的规律,进而对学生成绩进行分类预测,及时发现学困生并干预指导,提高教学质量^[6,7]。近年来,众多学者基于在线学习行为数据对学习成绩预测的方法进行了多方面探索。喻铁朔等采用支持向量机回归对成绩进行预测,并在其基础上实现学业预警^[8]。吕品等对比感知机,支持向量机和神经网络等分类算法的预测效果,发现基于支持向量机的成绩预测模型具有更高的准确率^[9]。王凤芹等利用登录次数,视频观看时间等10个行为特征验证K近邻优化算法在预测MOOC学习成绩的有效性^[10]。

虽然通过优化算法和调节算法参数可以比原有的单一算法模型具有更高的精度,泛化能力也有所增强^[11,12]。但由于线上线下相结合的学习行为数据相对复杂多变,异常值较多,单一的预测算法有自身的局限性。为了提高单一算法模型的泛化能力,有学者尝试通过融合算法来建立成绩预测模型。贾靖怡等利用MOOC课程数据和RapidMiner大数据挖掘平台构建基于AdaBoost算法的学习成绩预测模型^[13]。宋洁通过实验对比了SVR、RF、GBDT和XGBoost四种不同算法模型在学习成绩预测中的效果,最终得出具有集成特点的XGBoost算法预测效果最佳^[14]。但XGBOOST和AdaBoost属于同质算法的集成模型,主要通过线性融合算法,无法将各个算法的差异与优势相融合,其泛化能力仍需加强^[15,16]。

为提高模型的泛化能力,充分发挥个体学习器的优势,本文采用Stacking集成方法,对多项式朴素贝叶

斯、AdaBoost、Gradient boosting与逻辑斯蒂回归4个异质学习器进行非线性组合,构建两层Stacking集成学习模型,基于超星平台的混合式教学中产生的数据集。通过对比Stacking集成学习模型与多项式朴素贝叶斯、AdaBoost、Gradient boosting和逻辑斯蒂回归在数据集上的预测准确率、召回率、精确率和F1得分等评价指标,验证Stacking集成学习模型的有效性。

2 Stacking集成学习预测模型

集成学习是通过一定的方式组合多个弱学习器,构造出比单一学习器性能更好的强学习器。常见的集成学习方法有Bagging, Boosting和Stacking等。Bagging和Boosting采用线性集成策略,根据确定性算法组合同质弱学习器形成强学习器。与Bagging和Boosting不同的是,Stacking采用非线性集成策略,通过组合异质学习器构成多层次强学习器,高层学习器能够通过低层学习器的输出结果对模型进一步泛化增强^[17]。

2.1 Stacking集成学习算法原理

Stacking集成学习算法是一种叠加式分层集成算法,以两层为例,在第1层中使用不同的初级学习器对数据进行训练,在训练数据集时,为了避免因测试集比例划分过小造成泛化能力不强问题,通常采用交叉验证方式进行训练。然后将第1层产生的多次训练的数据结果作为新的训练集和测试集,第2层次级学习器采用新的训练集与测试集进行训练和预测。具体算法如算法1所示。

算法1. Stacking集成学习算法

输入: 数据集 $D=\{(x_1,y_1),(x_2,y_2),\dots,(x_m,y_m)\}$, 初级学习器 s_1,\dots,s_T , 次级学习器 ζ

1. for $t=1,2,\dots,T$ do
2. $h_t=s_t(D)$
3. end for
4. $D'=\emptyset$
5. for $i=1,\dots,m$ do
6. for $t=1,\dots,T$ do
7. $z_{it}=h_t(x_i)$
8. end for
9. $D'=D'\cup\{(z_{i1},\dots,z_{iT}),y_i\}$
10. end for
11. $h'=\zeta(D')$

输出: 最终分类器 $H(x)=h'(h_1(x),\dots,h_T(x))$

2.2 两层Stacking集成学习成绩预测模型设计

Stacking集成学习模型采用的是多层次结构,每一

层的输出作为下一层的输入,但构造的层数越多,模型越复杂,训练的速度也会越慢.因此,本文选择采用两层结构.在第1层,为了保证模型的多元化,初级学习器选用多项式朴素贝叶斯,AdaBoost和Gradient boosting三种学习器,其中多项式朴素贝叶斯为一种适用于离散特征概率计算的学习器,AdaBoost和Gradient boosting为基于Boosting集成学习方式的学习器^[18,19];在第2层,为了防止发生过拟合现象,采用简单的逻辑斯蒂回归分类器作为次级学习器,次级学习器对初级学习器融合得到Stacking集成学习模型.Stacking集成学习模型中的个体学习器在训练过程中,通过sklearn库中的GridSearchCV(网格搜索法)选取个体学习器的最优参数.集成学习模型框架如图1所示.

两层Stacking集成学习模型的具体步骤如下:

步骤1.对学生的原始特征数据集进行预处理与特征变换,将目标特征即学生期末成绩转换为0和1,结合皮尔逊相关性分析选择有效特征构建新的数据集^[20].

步骤2.选择AdaBoost,多项式朴素贝叶斯和Gradient boosting作为初级学习器.

步骤3.将经过特征工程处理过的混合式教学下产生的学生特征数据集以一定比例切分为训练集 D 和测试集 B .并将处理好的训练集平均分为5份数据集 D_i , $i \in \{1,2,3,4,5\}$.

步骤4.在第一层预测模型中,将3个初级学习器分别采用5折交叉验证进行训练.以AdaBoost为例,每次交叉验证包含两个过程.首先用其中的4份数据集作为AdaBoost训练集进行训练,剩下的1份作为AdaBoost模型预测的测试集,得到预测数据集 a_i ($i \in \{1,2,3,4,5\}$),将其整合为1列记为 A_1 .然后再由上一步基于4份数据集训练的AdaBoost模型对测试集 B 进行预测,得到预测数据集 p_i ($i \in \{1,2,3,4,5\}$),再对5次预测的结果 p_i ($i \in \{1,2,3,4,5\}$)按行相加取均值,得到数据集 P_1 .

步骤5.在3个初级学习器完成训练及预测后会得到 (A_1, A_2, A_3) 和 (P_1, P_2, P_3) 两个新的矩阵数据集.在第二层,次级学习器逻辑斯蒂回归以 (A_1, A_2, A_3) 矩阵作为训练集, (P_1, P_2, P_3) 矩阵作为测试集,以其为基础进行训练,得到最终预测结果,即学生能否通过期末检测(0不通过,1通过).

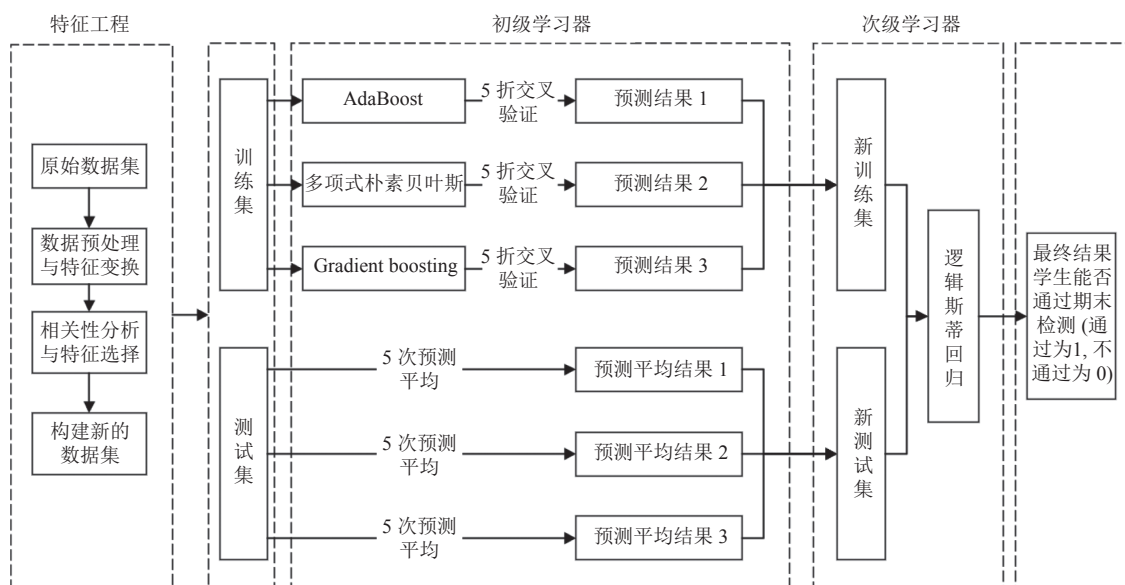


图1 Stacking集成学习模型框架

3 特征工程

特征工程指的是从原始数据集中提取实验特征的过程,提取的特征能够较好的描述数据集的内容,其一般包括数据预处理和特征分析与选择等.

3.1 数据集介绍

本文数据采集自学校超星平台与教务系统,共229名

学生,近6万多条线上学习记录和229条学生线下成绩记录.数据集中学生特征共有16个维度,分为学生的基本信息特征和学习行为特征两类,分别如表1和表2所示.

3.2 数据预处理

由于混合式学习特征数据多样化,需要根据数据

与算法模型的特点,对数据进行预处理.数据预处理主要包括不同考核标准特征数据的归一化和连续型数据的离散化等.本文在预处理过程中使用了 Python 中的 sklearn、pandas 和 numpy 等库.

表1 学生基本信息特征说明

序号	特征名称	数据类型	特征含义
1	student_id	varchar	学生的id编号索引
2	gender	varchar	学生的性别
3	faculty	varchar	所在的院系
4	class	varchar	所在的开课班级
5	semester	varchar	学期

表2 学习行为特征说明

序号	特征名称	数据类型	特征含义
1	study_count	int	登录超星平台学习课程内容的总次数
2	video	double	登录超星平台观看课程视频的总时长
3	discussion	int	讨论次数
4	chapter_quiz	double	章节测验
5	answer_mark	double	抢答得分
6	group_task	double	分组任务
7	course_point	double	课程积分
8	class_test	double	参加随堂测验的评分
9	homework	double	课后作业成绩评分
10	midterm	double	期中考试卷面成绩
11	final_mark	double	期末考试卷面成绩

3.2.1 归一化

每个开课班级在超星学习通上课程设置的任务点、学习资源以及课堂活动等有所差别,从而造成特征的数值区间范围也不同,需采用区间缩放法将数据映射到同一区间.区间缩放法是归一化的一种方式,本文利用特征数据的两个最值(最大值和最小值)进行缩放.缩放公式如下.

$$x_i' = \frac{x_i - \min(x)}{\max(x) - \min(x)}, i = 1, 2, \dots, m \quad (1)$$

其中, x 为同一特征集合,共有 m 个特征变量, x_i 为该特征集合中的第 i 个特征变量, x_i' 为 x_i 缩放后的值, $\max(x)$ 为该集合特征变量的最大值, $\min(x)$ 为该集合特征变量的最小值.

3.2.2 特征转换

本文使用的数据集中的数据主要为 varchar (可变字符型)、int (整数型) 和 double (浮点型) 3 种类型,而 sklearn 库要求模型输入的数据类型必须是数值型数据,且本文采用的预测模型为集成分类预测模型,其中个体算法模型 AdaBoost 和 Gradient boosting 为树模型

结构,为了保证模型输入数据类型的一致性,需使用分箱操作对连续型特征数据离散化.为此,本文通过 pandas 库中的 cut() 函数对表 2 的 11 个 int 型和 double 型数据特征等宽划分离散化处理,并对其进行数值型映射.由于表 1 的 5 个特征数据为 varchar 型定性离散变量,故直接将其映射为数值型变量.

3.3 特征相关性分析

在使用模型对目标预测前,通常需要先进行特征选择,通过特征选择可以移除不相关的特征,降低计算复杂度以及提高模型的可解释性.为此,本文通过皮尔逊系数计算特征之间的相关性.皮尔逊系数主要衡量变量之间的线性相关性,结果的取值区间为 $[-1, 1]$, -1 表示完全的负相关, $+1$ 表示完全的正相关^[21].实验获取的数据共有 16 个特征,包括学生的基本信息特征和学习行为特征,在经过数据预处理后,通过皮尔逊系数计算特征之间的相关性系数,并将相关性系数的结果以热力图的形式输出,其结果如图 2 所示.

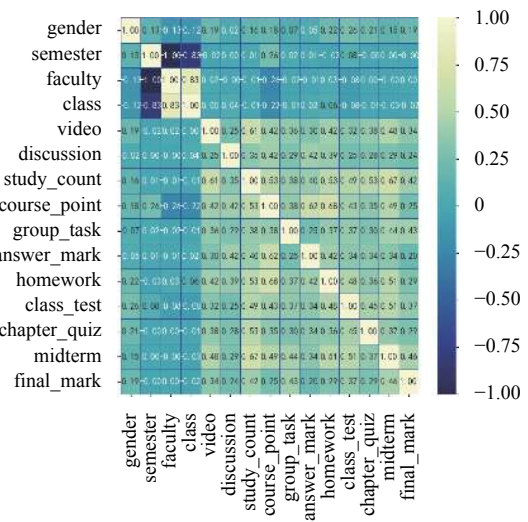


图2 特征相关性系数矩阵热力图

由图 2 可以看出 semester, gender, faculty, class 与目标特征 final_mark 之间的相关性系数小于 0.2,相关性较弱,其中 semester 和 faculty 与目标特征的相关性系数皆为 0,无相关性.与 final_mark 之间的相关性系数大于 0.4 的特征有 midterm, study_count, group_task, 其中特征 midterm 代表期中考试的成绩,与目标特征 final_mark 的相关性最高,达到了 0.46. discussion, course_point, answer_mark, chapter_quiz 等 7 个特征与 final_mark 的相关性相对较一般.故本文将相关性为

0 的无用特征 semester 和 faculty 删除, 保留 midterm, group_task, class_test 等 13 个相对重要的特征。

4 实验及结果分析

本实验在处理器为 Intel i7-8750U, 运行内存为 16 GB 的电脑上进行, 操作环境是 64 位的 Windows 10 系统. 算法实现工具为 Pycharm, 编程语言为 Python 3.7.0. 算法实现过程中使用到 Python 的工具库有: numpy、pandas、matplotlib、seaborn 和 sklearn 等. 将数据集中的 229 条学生行为数据随机切分出 70%, 共 160 条作为模型的训练集, 剩余的 30% 作为模型的测试集, 用于模型的验证。

4.1 模型的泛化能力评估指标

本文的目的在于通过相关的学生特征来预测学生能否通过期末检测, 将预测结果分为通过 (pass) 和挂科 (failed) 两类, 这是一个二分类预测问题. 二分类预测问题的混淆矩阵分析表如表 3 所示。

表 3 混淆矩阵分析表

混淆矩阵		预测值	
		挂科学生	通过学生
真实值	挂科学生	TP (真挂科学生)	FN (假通过学生)
	通过学生	FP (假挂科学生)	TN (真通过学生)

本文采用准确率 (A), 召回率 (R), 精确率 (P), $F1$ 得分 ($F1$) 4 个指标评价算法的预测结果. 准确率虽然能够判断整体模型预测的正确率, 但相比于通过学生, 本文更加关注真正挂科学生的预测情况, 因此应当引入召回率作为评估指标, 但若只注重召回率的提升, 会减少成功预测真正通过学生的情况, 即会导致精确率下降, 模型变得虚高^[22]. 召回率与精确率是一对相对矛盾的变量, 为此引入 $F1$ 得分来平衡召回率与精确率, 更为客观的对模型进行评价. 各项指标的计算方法的公式如式 (2)–式 (5)。

$$A = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

$$P = \frac{TP}{TP + FP} \quad (3)$$

$$R = \frac{TP}{TP + FN} \quad (4)$$

$$F1 = \frac{2P \times R}{P + R} \quad (5)$$

4.2 模型参数设置

超参数是模型在开始学习之前设置的函数参数, 选择合适的参数可以在一定程度上提高模型的分类能力. 本文 Stacking 集成学习模型中的个体学习器在训练集中训练时, 采用机器学习库 sklearn 中的 GridSearchCV (网格搜索法), 以准确率 (A) 作为各学习器预测效果的评价标准, 在各学习器的所有候选的参数选项中循环遍历, 并结合 5 折交叉验证方式, 通过对比在测试集中的预测效果, 寻找模型的最优参数. 各学习器的最优超参数集及其预测准确率如表 4 所示。

表 4 各学习器选择的最优超参数集

模型名称	超参数设置	准确率 (A)
多项式朴素贝叶斯	alpha=1.0, fit_prior=True, class_prior=None	0.71
AdaBoost	base_estimator=None, n_estimators=65, learning_rate=0.05	0.70
Gradient boosting	learning_rate=0.05, n_estimators=80, max_depth=5, min_samples_leaf=2, min_samples_split=2	0.67
逻辑斯蒂回归	tol=0.0001, max_iter=47	0.70

4.3 模型的泛化能力分析

集成学习的目的是通过一定的策略集合各个初级分类器的优势, 取长补短, 提高分类器的分类效果, 因此集成后的分类效果应好于单个基分类器的分类效果. 为验证所构建的 Stacking 集成学习预测模型的预测性能, 通过对比多项式朴素贝叶斯、AdaBoost、Gradient boosting、逻辑斯蒂回归和 Stacking 集成学习预测模型在数据集上的预测效果来验证本文所设计的模型的优越性. 最后根据第 4.1 节中的式 (2)–式 (5), 对各分类器的分类效果通过准确率、召回率、精确率和 $F1$ 得分等评价指标对各模型的预测性能进行对比分析验证, 各指标结果如图 3、图 4 和图 5 所示. 在对比分析验证过程中, 多项式朴素贝叶斯、AdaBoost、Gradient boosting 和逻辑斯蒂回归以及 Stacking 集成分类器中的初级学习器与次级学习器在训练与预测中所采用的参数如表 4, 各参数为通过机器学习库 sklearn 中的 GridSearchCV (网格搜索法) 中搜寻的最优参数。

从图 3、图 4 和图 5 中可以看出 Stacking 集成学习模型的各分类结果评价指标均明显高于多项式朴素贝叶斯、AdaBoost、Gradient boosting 和逻辑斯蒂回归 4 个单一分类模型. 其中, 在召回率上, Stacking 集成学习模型在 Pass 类别中的召回率为 87%, 高于单一分

类模型中表现最好的多项式朴素贝叶斯 7%; 在 Failed 类别中的召回率为 65%, 比单一分类模型中表现最好的模型 AdaBoost 高出 2%。在 F1 得分上, Stacking 集成学习模型在 Pass 类别中的 F1 得分为 80%, 分别高于多项式朴素贝叶斯, AdaBoost, Gradient boosting 和逻辑斯蒂回归 4 个单一分类模型 4%、8%、7% 和 7%; 在 Failed 类别中的 F1 得分为 71%, 分别高于多项式朴素贝叶斯、AdaBoost、Gradient boosting 和逻辑斯蒂回归 4 个单一分类模型 9%、4%、13% 和 6%。Stacking 集成学习模型的整体分类准确率为 76%, 分别高于多项式朴素贝叶斯、AdaBoost、Gradient boosting 和逻辑斯蒂回归 4 个单一分类模型 5%、6%、9% 和 6%。

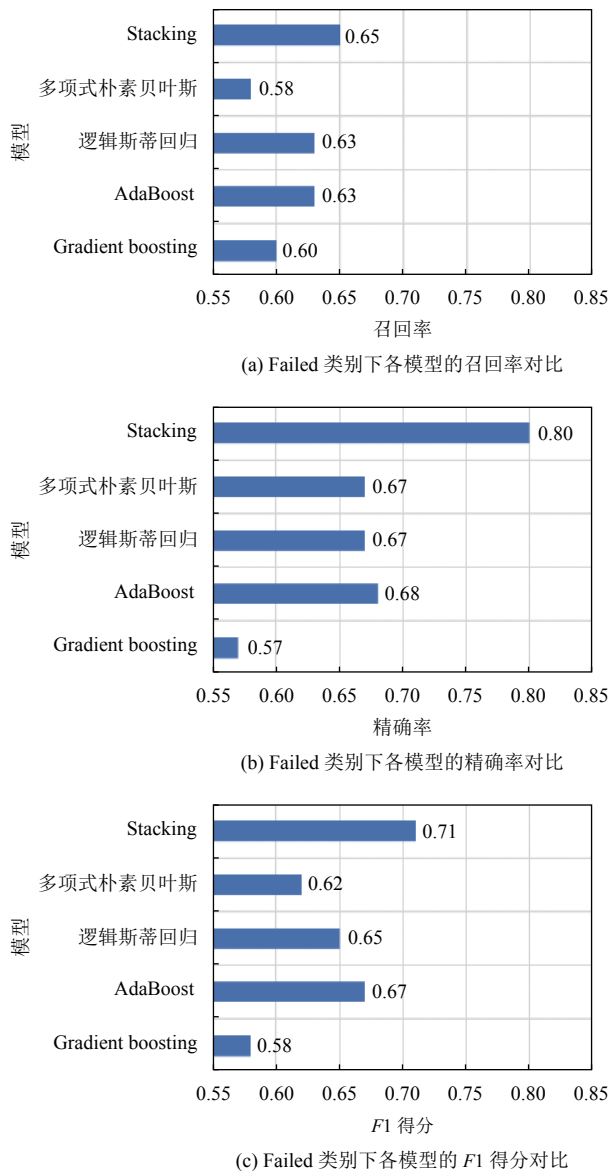


图 3 Failed 类别下各模型的评价指标对比

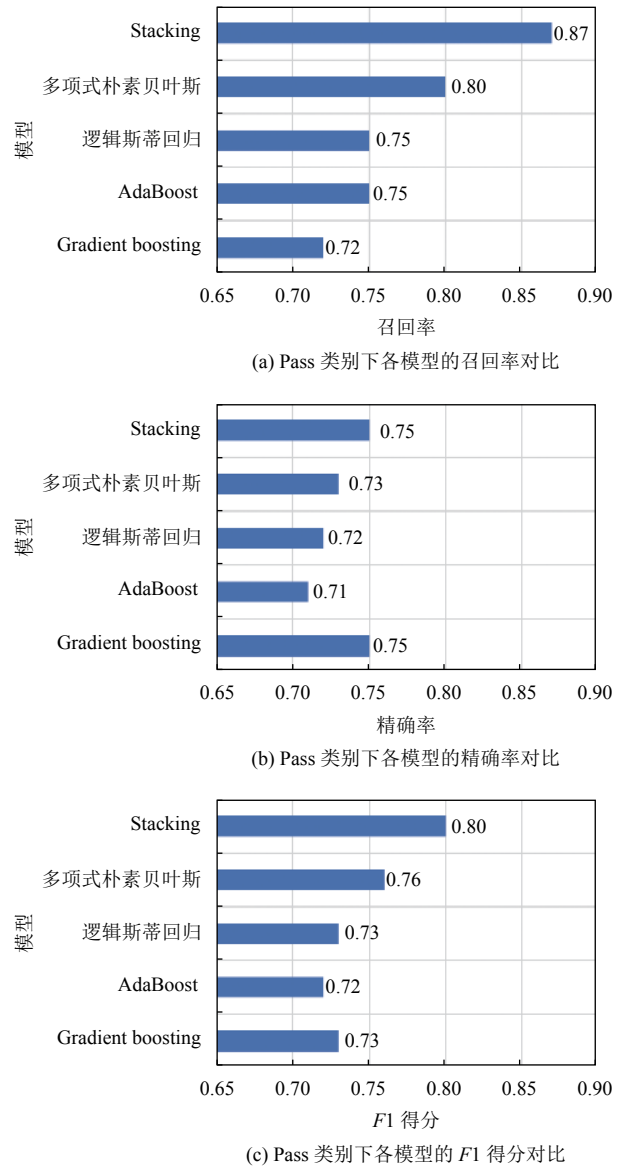


图 4 Pass 类别下各模型的评价指标对比

从以上实验结果来看, 通过 Stacking 融合多项式朴素贝叶斯、AdaBoost、Gradient boosting 和逻辑回归分类模型能够有效弥补单一算法模型的泛化能力不强问题, 这是由于 Stacking 集成学习模型通过学习几个不同的初级学习器, 然后通过学习一个次级学习器来组合初级学习器, 基于初级学习器的预测结果输出最终预测结果。从理论层面看, Stacking 集成学习模型优于个体模型的主要原因是 Stacking 集成学习模型可以充分发挥各个学习器的自身优势, 对于个体学习器在测试集中预测较差的部分采用舍弃策略, Stacking 集成学习模型采用多而不同的个体学习器, 取长补短, 可以在一定程度上有效减少单一模型泛化性能不佳的风

险. 另一方面, 从模型优化角度看, 单一模型训练的优化过程中, 模型往往会有陷入局部最小点的风险, 有的局部极小点所对应的模型泛化性能可能较差, 而通过多个基学习器运行之后进行结合, 可有效减少陷入局部极小点的风险. 此外, 从图3和图4的Failed类别与Pass类别预测的同一指标两种类别之间差异性可看出Failed类别与Pass类别数据分布较不平衡, 混合式教学过程中产生的数据一般分布较不平衡, Failed类别样本数据较少, 而Stacking集成学习模型相比于个体学习模型多了一层学习模型, 可以在同样训练数据下进一步进行学习, 因此, 采用Stacking集成学习方式后预测精度有所提升. 本质上Stacking集成学习模型是通过不同的方式分析数据, 然后根据学习器自生的算法规则建立模型, 次级学习器能够根据初级学习器的输出结果对模型进一步泛化增强, 当初级学习器得到了错误的预测结果时, 次级学习器可以将错误纠正回来. 因此, 融合差异性大的个体学习器可以充分发挥不同个体学习器的优势, 使得集成后的模型比单一算法模型具有更强的泛化能力.

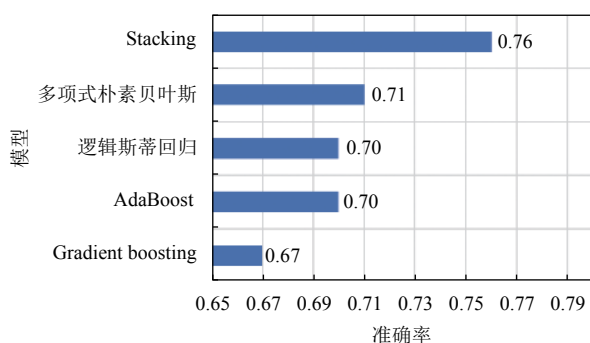


图5 各模型的准确率对比

5 总结

本文针对单一算法预测模型在学生成绩分类预测中存在泛化性能不强的问题, 提出一种Stacking集成学习模型, 其采用两层结构, 第一层通过融合多项式朴素贝叶斯, AdaBoost和Gradient boosting三个初级学习器, 第二层次级学习器采用逻辑斯蒂回归分类模型. 基于超星平台与教务系统获取的学生相关特征数据, 通过特征的皮尔逊相关性分析选择有效特征, 通过实验对Stacking集成学习模型与多项式朴素贝叶斯, AdaBoost和Gradient boosting和逻辑斯蒂回归进行比较. 结果表明, Stacking集成学习模型的召回率, 准确率等

各项预测性能指标皆高于单个初级学习器, 在混合式教学的成绩分类预测中具有较好的应用价值. 但本文训练的数据量不是足够大, 且所选取特征的相关性相对较一般, 后期可以尝试加入提交作业时间先后等更多相关性较强的特征, 以及积累更多的训练数据来提高模型的总体分类预测效果.

参考文献

- 冯晓英, 王瑞雪, 吴怡君. 国内外混合式教学研究现状述评——基于混合式教学的分析框架. 远程教育杂志, 2018, 36(3): 13–24.
- 邱燕楠, 李政涛. 从“在线教学胜任力”到“双线混融教学胜任力”. 中国远程教育, 2020, 41(7): 7–15, 76.
- 江凤娟. 混合式教学环境中大学生学习的行为意愿影响因素研究. 电化教育研究, 2021, 42(6): 105–112, 128.
- Christie M, De Graaff E. The philosophical and pedagogical underpinnings of active learning in engineering education. European Journal of Engineering Education, 2016, 42(1): 5–16.
- 王永固, 许家奇, 丁继红. 教育4.0全球框架: 未来学校教育教育与模式转变——世界经济论坛《未来学校: 为第四次工业革命定义新的教育模式》之报告解读. 远程教育杂志, 2020, 38(3): 3–14.
- 柴艳妹, 雷陈芳. 基于数据挖掘技术的在线学习行为研究综述. 计算机应用研究, 2018, 35(5): 1287–1293.
- 郎波, 樊一娜. 利用学习向量化样本分类的在线学习成绩预测. 计算机系统应用, 2019, 28(3): 215–222. [doi: 10.15888/j.cnki.csa.006799]
- 喻铁朔, 甘琤, 李霞, 等. SVR回归在成绩预测预警中的应用研究. 中国教育信息化, 2020, (11): 76–80.
- 吕品, 于文兵, 汪鑫, 等. 基于机器学习的学生成绩预测及教学启示. 计算机技术与发展, 2019, 29(4): 200–203. [doi: 10.3969/j.issn.1673-629X.2019.04.040]
- 王凤芹, 李瑛, 韩庆龙. 基于k-近邻优化算法慕课学习成绩预测研究. 计算机与数字工程, 2019, 47(4): 785–788.
- 姜绍萍. 学生行为相关性分析及改进GA-BP学业预警算法. 计算机系统应用, 2021, 30(4): 199–203. [doi: 10.15888/j.cnki.csa.007868]
- Galar M, Fernandez A, Barrenechea E, et al. A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 2012, 42(4): 463–484. [doi: 10.1109/TSMCC.2011.2161285]
- 贾靖怡, 李玉斌, 姚巧红, 等. 基于AdaBoost的MOOC学

- 习成绩预测模型研究. 软件导刊, 2021, 20(3): 242–246. [doi: [10.11907/rjdk.201618](https://doi.org/10.11907/rjdk.201618)]
- 14 宋洁. 基于校园大数据的学习行为分析及学习成绩预测 [硕士学位论文]. 武汉: 华中师范大学, 2020.
- 15 De Carvalho Santos SGT, De Barros RSM. Online AdaBoost-based methods for multiclass problems. *Artificial Intelligence Review*, 2020, 53(2): 1293–1322. [doi: [10.1007/s10462-019-09696-6](https://doi.org/10.1007/s10462-019-09696-6)]
- 16 Su YY, Wang SX, Li Y. Research on the improvement effect of machine learning and neural network algorithms on the prediction of learning achievement. *Neural Computing and Applications*, 2022, 34(12): 9369–9383. [doi: [10.1007/s00521-021-06333-8](https://doi.org/10.1007/s00521-021-06333-8)]
- 17 Nti IK, Adekoya AF, Weyori BA. A comprehensive evaluation of ensemble learning for stock-market prediction. *Journal of Big Data*, 2020, 7(1): 20. [doi: [10.1186/s40537-020-00299-5](https://doi.org/10.1186/s40537-020-00299-5)]
- 18 Li W, Jiao GE. Prediction of poor students' classification based on Adaboost algorithm integrated learning model. *Journal of Physics: Conference Series*, 2020, 1574(1): 012172. [doi: [10.1088/1742-6596/1574/1/012172](https://doi.org/10.1088/1742-6596/1574/1/012172)]
- 19 Lee Y. Effect of uninterrupted time-on-task on students' success in massive open online courses (MOOCs). *Computers in Human Behavior*, 2018, 86: 174–180. [doi: [10.1016/j.chb.2018.04.043](https://doi.org/10.1016/j.chb.2018.04.043)]
- 20 盛钟松, 朱海景, 余谅. 基于时序特征和集成算法的用户购买预测. *计算机系统应用*, 2021, 30(10): 264–270. [doi: [10.15888/j.cnki.csa.008074](https://doi.org/10.15888/j.cnki.csa.008074)]
- 21 Shi HW, He ST, Huang JS. The method of rapidly improving pass rate of CET-4 based on Gaussian/steed variance. *Procedia Computer Science*, 2020, 176: 1652–1658. [doi: [10.1016/j.procs.2020.09.189](https://doi.org/10.1016/j.procs.2020.09.189)]
- 22 Mourdi Y, Sadgal M, El Kabtane H, *et al.* A machine learning-based methodology to predict learners' dropout, success or failure in MOOCs. *International Journal of Web Information Systems*, 2019, 15(5): 489–509. [doi: [10.1108/IJWIS-11-2018-0080](https://doi.org/10.1108/IJWIS-11-2018-0080)]

(校对责编: 孙君艳)