2020 年 8 月 第 8 期
Vol. 41 No. 8 2020

一种基于双层融合结构的客户流失预测模型

李为康,杨小兵

(中国计量大学 信息工程学院,杭州 310018)
E-mail: ignatius. lee@ foxmail. com

摘 要: 针对客户流失预测精准性的提升,提出了一种基于双层融合结构的客户流失预测模型. 该模型不需要提前对数据集进行独热编码,避免了维度灾难和数据稀疏问题. 其主要思想是融合多个高准确率的基于树的机器学习算法组成一个包含 Stacking 层与 Voting 层的双层预测模型. 数据集经过处理后输入到 Stacking 层,然后 Stacking 层的预测结果与处理后的数据集合并传递给 Voting 层,同时将 Stacking 层加入到 Voting 层的预测中,最后输出最终的预测结果. 在 Kaggle 的电信客户公开数据集上的实验表明,与经典的客户流失预测模型和改进的客户流失预测模型相比,本模型明显提高了客户流失预测的准确率和精准率.

关 键 词: 客户流失预测; 准确率; 机器学习; 分类模型; 精准率
中图分类号: TP181 **文献标识码:** A **文 章 编 号:** 1000-1220(2020)08-1634-07

Customer Churn Prediction Model Based on Double Layer Fusion Structure

LI Wei-kang, YANG Xiao-bing
(College of Information Engineering, China Jiliang University, Hangzhou 310018, China)

Abstract: Aiming at the improvement of customer churn prediction accuracy, a customer churn prediction model based on double-layer fusion structure is proposed. The model does not need One-Hot encoding in advance for the data set, avoiding dimensional disasters and data sparseness. The main idea of the model is to combine multiple high-accuracy tree-based machine learning algorithms to form a double-layer prediction model including the Stacking layer and the Voting layer. The data set is processed and input to the Stacking layer, and then the prediction result of the Stacking layer and the processed data set are transmitted to the Voting layer, and the Stacking layer is added to the prediction of the Voting layer, finally the final prediction result is output. Experiments on Kaggle's telecom customer open dataset show that this model significantly improves the accuracy and precision of customer churn predictions compared to the classic churn prediction model and the improved customer churn prediction model.

Key words: customer churn prediction; accuracy; machine learning; classification model; precision

1 引 言

时至今日,各类市场日益饱和且竞争激烈,属于行业巨头的市场份额越来越大,各行业企业家们以往关注的重点在于推出新颖的定制服务来吸引新客户,并将已经拥有的客户转换成忠诚客户[1]. 而研究表明发展一个新客户的成本远高于维护一个老客户的成本[2],所以预防老客户的流失是各企业家们必须重视的问题.

因此,客户流失预测技术对于企业挽留老客户和推出各种定制服务来说是十分重要的. 比如电信企业,一个流失的客户如果不再使用运营商提供的服务,那么他就再也无法产生任何利润,这对于拥有千万级别数量客户的运营商而言,如果能降低百分之一的客户流失率,那将会带来可观的利润增长[3]. 及时并准确识别潜在的流失客户渐渐成为了各大行业巨头企业家们研究的重点.

客户流失预测技术是从管理学中的 CRM(Customer Relationship Management)发展而来,是 CRM 中十分重要的组成部分,其流程包含了业务分析,数据分析,数据预处理,模型的构建、评估和部署.

目前,在客户流失预测技术上的研究获得了很多成果. Kaizhu 等人[4]于 2014 年提出了可理解的支持向量机,该模型不仅在精度方面表现优异,而且通过构建朴素贝叶斯树,可以精准的分析客户流失的原因. 同年,Verbeke 等人[5]通过分析社交网络来划分客户群体,针对不同的客户群体使用不同的分类模型,既提高了预测表现,又分析了不同社交群体的流失原因. 文献[6]使用遗传算法来进行模型的构建并加入了 Benefit maximization 准则,在提高模型预测表现的基础上,还能为企业选择利益最大化的方案. 文献[7]使用了 Logistic 回归与决策树的混合算法来构建预测模型,该算法在预测表现与可解释性上都较好. 文献[8]提出了两种基于改进的多层感知机的客户流失预测模型,解决了独热编码后数据维度过高及数据稀疏带来的计算消耗等问题.

近些年,机器学习是人工智能及模式识别领域的共同研究热点,由于硬件条件的提升和大数据技术的发展,机器学习在图形识别,语音识别和分类预测等方面取得了巨大的进步,并且准确率远高于大部分传统模型. 在客户流失预测领域,机

收稿日期: 2019-10-09 收修改稿日期: 2019-11-19 基金项目: 国家自然科学基金项目(61303146)资助. 作者简介: 李为康,男,1992 年生,硕士研究生,研究方向为人工智能、数据挖掘; 杨小兵,男,1976 年生,博士,副教授,CCF 会员,研究方向为人工智能、数据挖掘、智能数据处理.

器学习的算法如强化学习算法的应用大幅提高了模型的准确率,但是单个算法在预测准确率上的提升还很有限.

为了提升电信客户流失预测的精确性,本文提出利用 GBDT(Gradient Boosting Decision Tree),LightGBM(Light Gradient Boosting Machine),XGBoost(eXtreme Gradient Boosting),CatBoost(Categorical Boosting),AdaBoost(Adaptive Boosting)五种基于树的算法构建双层融合模型运用在客户流失预测上.

2　预备工作

2.1　数据来源

本文实验采用的数据集来自于 Kaggle 数据科学竞赛中公开的数据集. 该数据集包含了 100000 个电信企业客户数据样本,其中流失客户标签数量为 49562 个,非流失客户标签数量为 50436 个; 特征数目为 100,包含有 79 个数值型特征和 21 个离散型特征. 由于该数据集样本数量较多,且流失客户数量与非流失客户数量基本保持一致,可以判定属于平衡型大样本数据集.

2.2　特征编码

在机器学习领域中的数据样本有两种特征类别: 连续型特征和离散型特征. 然而,在客户流失预测中,数据集中客户数据的离散特征基本没有连续性,这无法适应大部分机器学习算法. 为了解决上述问题,通常情况下都会使用独热编码来处理这些离散特征,比如特征{网络模式:{GSM,CDMA,WCDMA}}经过独热编码后变为{网络模式:{GSM{100},CDMA{010},WCDMA{001}}},显然,经过独热编码后会产生稀疏数据,如果样本量很大,那么独热编码产生的大量稀疏数据会影响模型预测的准确率. 而且,如果数据集中离散特征数量过多,独热编码很容易造成维度灾难,直接导致了模型的时间消耗成本. 独热编码的另外一个弊端就是转换离散特征中不同的值时是完全独立的,断裂了不同值之间的内在关系[9].

为了避免独热编码带来的弊端,本文决定选用标签编码来对数据集进行特征编码. 标签编码处理离散数据时会将特征如{网络模式:{GSM,CDMA,WCDMA}}转换为{网络模式:{GSM{1},CDMA{2},WCDMA{3}}}. 但是标签编码也会带来新的问题,如一些基于距离的模型,在计算时会出现“GSM”加上“WCDMA”的平均值是“CDMA”这样的情况,这显然是不正确的.

2.3　算法选择

由于特征编码方式选择了标签编码方式,且客户流失预测几乎都是二分类问题,所以本文选择基于树的机器学习算法来搭建模型. 基于树的算法在处理变量时,并不是基于向量空间度量,数值只是一种类别符号,即没有偏序关系,很好地解决了上文中提到的标签编码带来的问题,所以非常适合标签编码后的数据处理. 而如果用独热编码处理数据本质上只是增加树的深度.

本文实验选取了在 Kaggle 二分类预测竞赛中运用较多、效果较好的几种基于树的算法,在没有对算法进行调参的情况下,直接在完整的数据集上进行训练预测,最终将会选择准确率较高的算法进行模型的搭建. 选择的算法在实验采用的数据集上准确率对比如表 1 所示.

表 1　准确率对比表

Table 1　Accuracy comparison table

算法名称	准确率
决策树	54.66%
ExtraTrees	56.23%
随机森林	57.78%
Adaboost	61.14%
GBDT	60.67%
XGBoost	61.81%
LightGBM	61.79%
CatBoost	62.31%

2.4　算法介绍

经过对比实验,本文最终选择 GBDT,LightGBM,XGBoost,CatBoost 和 AdaBoost 这五种算法搭建模型.

AdaBoost 的核心思想是针对同一个训练集训练不同的分类器,即弱分类器,然后把这些弱分类器集合起来,构造一个更强的最终分类器. 一般来说,使用最广泛的 AdaBoost 弱学习器是决策树[10]. AdaBoost 的主要优点有: 作为简单的二元分类器时,构造简单,结果可理解,不容易发生过拟合[11].

GBDT 算法利用到了梯度下降法的思想,且无论用于分类还是回归,基函数一直都使用的是 CART 回归树[12]. GBDT 二分类算法的关键是利用当前模型的损失函数负梯度的值作为分类问题算法中的残差的近似值,拟合一个分类模型. GBDT 的优点是在相对较少的调参步骤下,预测的准确率也可以比较高. 而且 GBDT 使用了一些健壮的损失函数,对异常值的鲁棒性非常强.

XGBoost 是陈天奇博士在 2014 年提出的一个优化的分布式梯度增强库[13],它在 Gradient Boosting 框架下实现机器学习算法. XGBoost 本质上是多个 CART 回归树的集成,和 GBDT 有些类似,但是与之相比有很多创新和提高. 比如提出了一种新颖的用于处理稀疏数据的基于树的学习算法,是第一种处理各种稀疏模式的统一方法; 在代价函数中加入了正则化项,用于控制模型的复杂度,防止过拟合,等. XGBoost 已经在大量的机器学习和数据挖掘竞赛中被广泛地认可[14].

2017 年 1 月微软在业内知名的开源软件项目托管平台 GitHub 上开源了 LightGBM,它是一种高性能的基于决策树算法的梯度提升框架. 相比 XGBoost,同样的实验条件,LightGBM 在不降低准确率的前提下,预测所消耗的时间减少了十倍左右,占用系统内存却下降了三倍左右[15]. 它采用最优 Leaf-wise 算法分裂叶子节点,而不是 Level-wise 算法. 当拥有相同数量的叶子节点时,Leaf-wise 算法比 Level-wise 算法损失得更少,因此 LightGBM 拥有更高的精准率. 而其他的任何已存在的梯度提升算法都不能够达到这样的精准率. 而且 LightGBM 用到的直方图做差带来了一定的正则化的效果[16],能够使拟合出来的模型避免过拟合且具有更好的推广性.

在微软开源 LightGBM 三个月后,俄罗斯顶尖技术公司 Yandex 也在 GitHub 上开源了 CatBoost 算法. CatBoost 的核心是对称完全二叉树思想,即每一次只划分出两条路径,划分路径的顺序是随机的. 特征维数在划分后不会减小,不过用来划分的特征会与一个其他类别特征通过贪婪算法[17]的方式

相结合形成新特征. 然后,在样本的逐个添加的过程中,算法可以自动检测并剔除干扰样本,随着样本数量的累积,预测结果会变得更为准确. 根据官方网站[1] 数据显示,同样的实验条件下,CatBoost 性能要优于 XGBoost 和 LightGBM,文献[18,19]中也证明了相比较于经典算法 CatBoost 的表现十分优异. 而文献[20]将 CatBoost、XGBoost 和 LightGBM 融合构建出来的模型运用在地质探测研究上性能也很优异.

3 模型搭建

本文提出的分类预测模型由双层结构组成,分为 Stacking 层和 Voting 层,每层单独搭建好后再进行融合,构成最终的客户流失预测模型.

3.1 Stacking 层

Stacking 是一种运用堆叠思想的集成学习算法,目前在分类问题上运用的也比较广泛[21]. 其核心思想是通过组合多个基础分类器构建初级分类模型,再基于训练集训练初级分类模型; 然后,初级分类模型输出的训练集预测结果和测试集预测结果用来训练次级分类模型. 图1所示是Stacking初级分类模型中单个基础分类器 3 折验证模型工作的过程示意图.

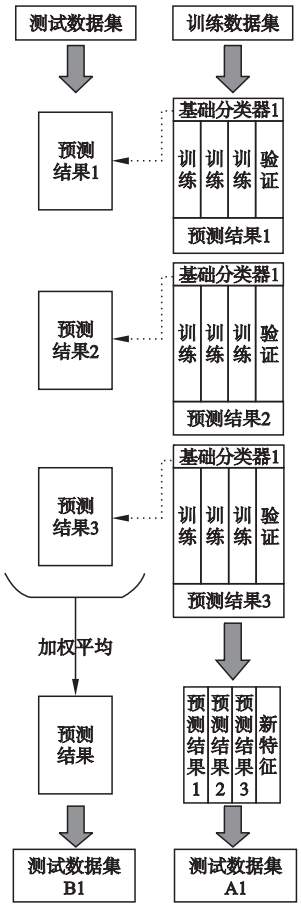


图 1 单个基础分类器 3 折验证模型工作过程示意图

Fig. 1 Working process diagram of single basic classifier 3 fold validation model

首先将数据集划分为训练集和测试集(假设训练集为 999 条数据,测试集为 210 条数据),然后一级分类模型中的单个基础分类器 1 进行 3 折交叉验证,使用训练集中的 666 条作为喂养集,剩余 333 条作为验证集. 每次验证使用 666 条数据训练出一个模型,再用训练出的模型对验证集进行验证得到 333 条数据,同时对测试集进行预测,得到 210 条数据. 这样经过 3 次交叉检验,可以得到新特征也就是 3 × 333 条预测结果和 3 × 210 条测试数据集的预测结果.

接下来会将 3 × 333 条预测结果拼接成 999 行 1 列的矩阵,标记为训练数据集 A1. 而对于 3 × 210 行的测试数据集的预测结果进行加权平均,得到一个 210 行 1 列的矩阵,测试数据集 B1. 这是单个基础分类器在数据集上的预测结果,如果再集成两个基础分类器,比如基础分类器 2、基础分类器 3,那么最后会得到 A1、A2、A3、B1、B2、B3 一共六个矩阵.

最后将 A1、A2、A3 并列在一起成 999 行 3 列的矩阵作为训练数据集,B1、B2、B3 合并在一起成 210 行 3 列的矩阵作为测试数据集,让次级分类模型基于这样的数据集再训练.

为了避免基础分类器之间相关性过高,实验初期搭建了基础分类器分别是 XGBoost,LightGBM,CatBoost 的初级分类模型和基础分类器为 AdaBoost,GBDT 的次级分类模型来组成 Stacking 层. Stacking 层的实验结果在训练集和测试集上的准确率分别为 99.35% 和 62.96%,ROC(Receiver Operating Characteristic) 曲线[22] 如图 2 所示.

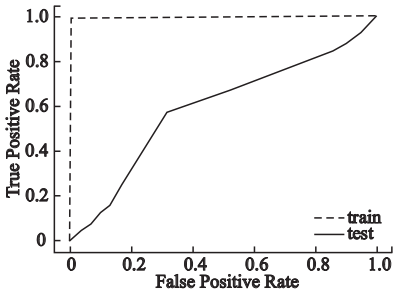


图 2 Stacking 层训练集和测试集 ROC 曲线

Fig. 2 ROC Curve of Stacking layer training set and test set

根据图 2 可以看出,Stacking 层搭建好后在训练集上准确率比较高,在测试集上的准确率相比文章选取的 5 种算法也有提升,但是准确率提升不到 1%,不够理想.

3.2 Voting 层

由于 Stacking 层的效果不够理想,所以本文又在实验中加入加权投票算法[23] 的思想,融合了双层结构组成了最终的客户流失预测模型,加权投票算法示意如图 3. 加权投票算法的思想是每个基础分类器都对样本做出自己的判断,并对它判断的类进行投票. 如图 3 所示,假设基础分类器有 5 个,那么他们对数据进行训练并预测的结果最后会进行加权平均. 而且基础分类器在投票时的权重可以随分类器的准确率而设定,准确率较高的基础分类器可以具有较大的权重值. 最终根据计算后概率最高的类确定样本的判定结果. 因此,与单独的

[1] https://catboost.ai/

基础分类器相比,加权投票算法的使用可以提高最终结果的准确率.

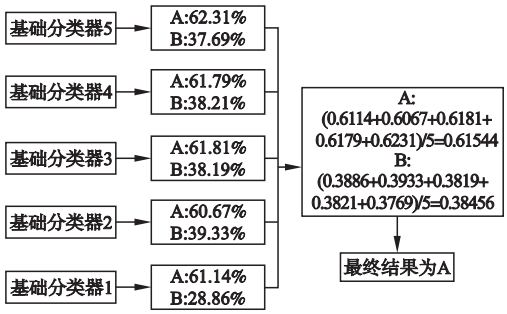


图 3　加权投票算法流程示意图

Fig. 3　Flow diagram of weighted voting algorithm

Voting 层搭建好后,对比实验了文章选取的 5 个基础分类器不同权重下的准确率,最终确定基础分类器权重设置为{{AdaBoost:1},{GBDT:1},{XGBoost:1},{LightGBM:2},{CatBoost:2}},在训练集和验证集准确率分别为 91.12% 和 63.67%,ROC 曲线如图 4 所示.

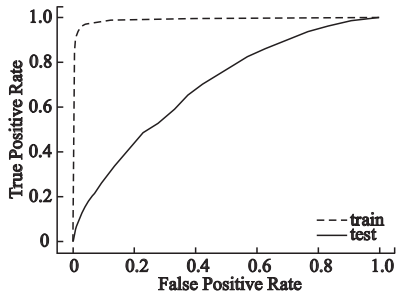


图 4　Voting 层训练集和测试集 ROC 曲线

Fig. 4　ROC curve of Voting layer training set and test set

根据图 4 可以看出,相较于 Stacking 层,Voting 层虽然在训练集上的准确率下降了,但在测试集上的准确率却有提升.

3.3　双层模型融合

在进行模型融合实验时,发现如果按照第一层使用 Voting 层,第二层使用 Stacking 层的结构搭建模型,那么预测结果和单独使用 Stacking 层的预测结果几乎一致. 而如果是第一层使用 Stacking 层,第二层使用 Voting 层的结构搭建模型,那么准确率有明显提升. 且实验中将 Stacking 层训练好后作为一个基础分类器赋予高权重加入到 Voting 层会使模型的准确率再次到提升. 双层模型融合好之后搭建的完整客户流失预测模型的流程图如图 5 所示.

原始数据集被读取后进行标签编码,随后按照 7:3 的比例划分训练集和测试集. 将划分好的数据集输送到 Stacking 层进行训练预测,经过 5 折交叉验证将结果与划分后的数据集合并再送入 Voting 层进行训练预测,同时将 Stacking 层赋予高权重作为一个基础分类器加入到 Voting 层参与预测.

3.4　算法时间复杂度分析

文章采用了多个分类器的融合模型,融合之后算法时间复杂度应有较大的提升. 假设样本数量是 N,特征数量是 D,树的深度是 M,弱分类器数量是 T,直方图宽度是 K,随机排序次数是 S,那么各算法时间复杂度如表 2 所示.

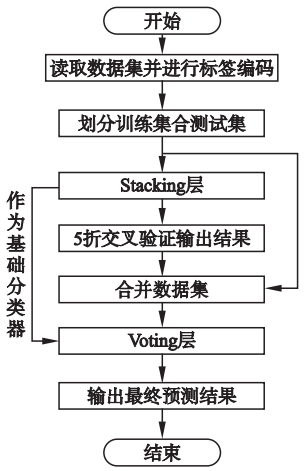


图 5　客户流失预测模型流程图

Fig. 5　Flow chart of customer churn prediction model

表 2　算法时间复杂度

Table 2　Time complexity of algorithm

算法名称	时间复杂度
GBDT	O(D* M* N* logN)
Adaboost	O(T* M* N)
XGBoost	O(D* (N + log(D)))
LightGBM	O(K* (D + log(K)))
CatBoost	O(M* S* N)

根据表 2 可以估算出在相同条件下,时间复杂度优越性理论上从高到低分别为 CatBoost、LightGBM、XGBoost、GBDT、Adaboost. 根据双层模型的结构特性,融合之后整个模型的间复杂度为应为 2* (O(M* S* N) + 2* (O(T* M* N))). 但实际各部分的时间消耗如表 3 所示.

根据表 3 可以看出 LightGBM 比 XGBoost 花费的时间还要多,且融合之后的双层模型比单层相加的时间要少,不符合上文分析的结果.

表 3　实验中的消耗时间

Table 3　Time consumed in the experiment

算法名称	时间/秒
GBDT	97.0326
Adaboost	159.7427
XGBoost	88.6598
LightGBM	89.44
CatBoost	84.91
Stacking 层	165.35
Voting 层	132.64
Stacking 层 + Voting 层	224.24

观察分析后发现,为了获得高准确率,调参之后模型消耗时间相比不调参有较大改变,且每次运行程序时也会有略微不同. 再由于进入 Stacking 层和 Voting 层之前各分类器就已经训练好,相比单层分别训练再进入要节省不少时间,故会出现表 3 中的结果. 而本文是针对客户流失预测精准性的提升,融合之后的模型相比单个分类模型的时间消耗提升在本文实

验条件下处于接受范围之内.

4 实验结果及分析

4.1 实验环境

本文实验在 Windows10 操作系统环境下,工程软件为 Pycharm. 使用深度学习框架 TensorFlow 构建双层模型,并利用机器学习库 Sklearn 构建对比实验. 硬件条件为 4 核 4 线程 2.5GHzCPU(酷睿 i5 7300),6Gb 显存的显卡(GeForce 1050TI),电脑内存是双通道 16G 内存.

4.2 评判指标及模型参数

目前,分类预测模型的评价指标一般使用准确率(*accuracy*),精准率(*precision*),召回率(*recall*)和 F1 值(*F*1)这三个指标. 不过,如果流失预测模型预测判定某一客户会流失,但实际上没有,这种错误在客户流失预测中是完全可以接受的. 但如果流失预测模型预测判定某一客户不会流失,而实际上该客户却属于流失客户,那这种错误是不可接受的. 因此,对于均衡型数据集,以上三个指标中最重要的是精准率和召回率. 公式如下:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

$$F1 = 2 * \frac{precision * recall}{precision + recall}$$

其中,*TP* 为正确划分为流失客户的样本数; *TN* 为正确划分为非流失客户的样本数; *FP* 为错误划分为流失客户的样本数; *FN* 为错误划分为非流失客户的样本数.

实验中各模型参数的调参过程使用了 Sklearn 库中的 GridSearch 函数,GridSearch 函数本质上是穷举搜索,循环遍历候选的参数,尝试每一种可能,输出表现最好的参数组合. 寻找最适合的参数能提高算法的准确率,实验所采用的各算法参数如表 4,未设置的参数都采用默认参数.

由表 4 可以看出,learning_rate 与 n_estimators 参数是大部分模型都调整的参数.

learning_rate 就是学习率,用来控制模型学习的进度,在监督学习中最常见. 学习率在机器学习中的作用可以表示为 $w_i = w_i - \lambda \partial F(w_i) / \partial w_i$,其中 w_i 是模型参数,F 是成本函数,λ 是学习率,$\partial F(w_i) / \partial w_i$ 是一阶导数. 学习率越大模型学习速度越快,但会因容振荡而错失最优值; 学习率越小模型学习速度越慢,会产生过拟合,收敛速度会很慢. 所以学习率对于算法性能的表现至关重要.

n_estimators 参数在不同的模型中有不同的含义,在 LightGBM、XGBoost、GBDT 中它代表树的棵数,而在 AdaBoost 和 Bagging 中它代表最大弱分类器个数. 理论上 n_estimators 数值越大,模型性能越好,预测也越稳定,但这也会减慢计算速度.

GBDT 算法中调整了树的最大深度(max_depth)、子节点最少样本数(min_samples_leaf)、子树划分条件(min_samples_split)、子采样比例(subsample)四个参数. 其中子采样比例可以用来防止过拟合,由于 GBDT 模型可以表示为决策树的加法模型,即 $f_m(x) = f_{m-1}(x) + T(x; \Theta_m)$,$T(x; \Theta_m)$ 为决策树,Θ_m 为决策树参数,m 为当前步数. 所以树的深度、子节点最少样本数、子树划分条件作为 Θ_m 的重要成员成为 GBDT 较为重要的参数.

表 4 实验算法参数

Table 4 Experimental algorithm parameters

算法名称	参数
AdaBoost	learning_rate = 0.1; n_estimators = 1800
GBDT	max_depth = 11; min_samples_leaf = 40 min_samples_split = 800; subsample = 0.8
XGBoost	learning_rate = 0.01; n_estimators = 1000
LightGBM	learning_rate = 0.02; n_estimators = 1000 lambda_l1 = 0.001; lambda_l2 = 1.0
CatBoost	learning_rate = 0.01
Bagging	base_estimator = ExtraTrees n_estimators = 180
Logistic Regression	C = 1; Penalty = L2 multi_class = ovr
MLP	learning_rate = 0.001
KNN	n_neighbors = 30 p = 2

LightGBM 和 Logistic Regression 的参数设定涉及到了 lambda_l1 和 lambda_l2 参数,即 L1 正则化和 L2 正则化. 正则化是结构风险最小化的一种策略. 在优化模型时对经验风险与模型复杂度做一个权衡,同而时符合偏差和方差分析,通过降低模型复杂度,得到更好的泛化能力,降低模型对训练数据的拟合程度. L1 正则化是在损失函数加上 L1 范数,容易得到稀疏解而 L2 正则化是在损失函数后加上 L2 范数,使得得出的解比较平滑. 两种参数的设定也要根据实际情况而定,比如在本次实验中 XGBoost 算法经过穷举调参得出最优解中并没有设定正则化参数.

KNN 是基于距离度量找出训练集中与其最靠近的 K 个训练样本,然后基于这 K 个"邻居"的信息来进行预测. 所以在进行调参时,对 KNN 影响较大的 K 值以及距离算法由 n_neighbors 和 p 两个参数表示,P = 2 代表模型采用欧式距离.

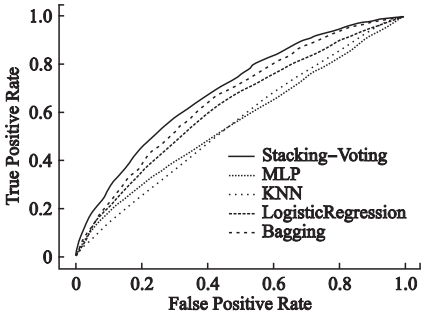
4.3 结果分析

本文选择 Bagging,KNN,Logistic Regression 三种经典的客户流失预测模型和 MLP 神经网络模块作为实验的对比对象,由于数据集相同,与文献[8]中提出的融合自编码器的 MLP、融合实体嵌入的 MLP 两种模型进行数据对比. 对比实验的 ROC 曲线与 P-R(Precision Recall)曲线[24]如图 6 所示,在测试集上的实验结果数据如表 5 所示.

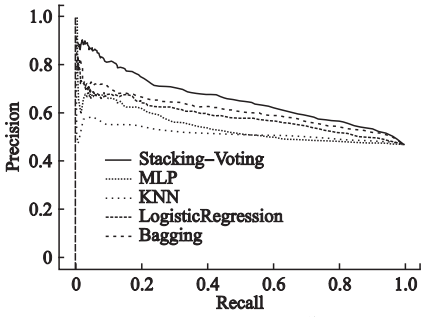
相对于未改进的 MLP 解决了独热编码问题,文献[8]中改进后的两种算法将高维的数据映射到低维的空间,降低了网络收敛于局部最优解的可能性,增加了数据间的关联性,改善了离散属性的度量方式,所以获得了较高的准确度.

KNN 划分客户群体实质上是通过计算欧氏距离来预测中心点周围的部分样本,选择多数类别作为预测值输出. 但这样就忽略了每个属性的数据分布范围,默认各属性属于同一数据范围,所以在属性值较多的情况下模型精度不高.

Logistic Regression 由于通过最大判别函数学习,对特征输出线性表达,且在训练时,不管特征之间有没有相关性,它都能找到最优的参数,所以在本次实验数据集上表现良好. 虽然 Logistic Regression 输出结果可解释性较好,但对模型中自变量多重共线性较为敏感. 且容易欠拟合,不能很好地处理大量多类特征或变量.



(a) 对比试验ROC曲线



(b) 对比试验P-R曲线

图 6　对比实验 ROC 曲线和 P-R 曲线

Fig. 6　ROC curve and P-R curve of comparative experiment

Bagging 采用了均匀取样,且训练集的选择是随机的,各轮训练集之间相互独立,而又因为基模型选择的是 Extra-Trees,又为整体多加了一层随机性,在对连续变量特征选取最优分裂值时,不会计算所有分裂值的效果来选择分裂特征,而是对每一个特征在它的特征取值范围内随机生成一个分割值,再通过选取一个特征来进行分裂. 所以对于平衡型数据集该模型表现良好.

表 5　实验结果数据

Table 5　Experimental data

模型名称	准确率	精准率	召回率	F1 值
MLP	56.54%	65.56%	15.24%	24.73%
融合自编码器的 MLP	62.44%	62.03%	63.75%	62.88%
融合实体嵌入的 MLP	62.64%	62.35%	62.72%	62.54%
KNN	54.33%	62.12%	35.31%	45.03%
Logistic Regression	59.32%	58.86%	47.59%	52.63%
Bagging	61.00%	47.20%	31.82%	52.83%
本文模型	64.38%	85.26%	57.51%	60.25%

本文模型在实验采用的公开数据集上有很好的表现. 融合了基于树型模型的 5 种强模型,同时避免了维度灾难和数据稀疏问题,保证了特征之间的关联性,在可接受范围内的时间复杂度的提升下带来了准确率和精准率的巨大提升,与选取的其他客户流失预测模型相比准确率平均高出 8.81%,并且与基于 MLP 改进后的两种模型相比都高出 1.7% 以上. 而在精准率和召回率方面,本文模型虽然召回率表现一般,但是精准率提高了 23% 左右. 综合对比下,本文提出的模型性能要优于对比实验中的各类模型. 能够在二分类预测比如信用评估、灾难预测等运用中有不错发挥.

5　结束语

由于采用了准确率较高的基于树的机器学习算法,同时融合了 Stacking 和 Voting 的方法搭建了双层预测模型来提高精度,加上针对性的数据处理方式,本文提出的模型在选取的电信客户数据集上进行客户流失预测的表现十分良好,在召回率差距不大的情况下,准确率和精准率比文中提到的经典的客户流失预测模型和改进的客户流失预测模型都要高. 不过,本文的研究仍旧存在些许不足,将来的研究工作将努力解决以下问题:

第一,本文提出的模型在大样本的平衡型数据集上有着良好的表现,但是对于非平衡型数据集和小样本的数据集是否还能延续良好的性能还需要继续研究; 第二,在机器学习算法的调参过程中,由于使用了 GridSearch 函数,调参的时间成本非常巨大,很多参数设定的跨度区间较大,最终选用的是默认参数,如何能在调参过程更加精细的前提下减短调参的耗时是十分重要的; 第三,对于本文模型的召回率的提高是否能通过模型内的不同层或不同级之间的算法调整来解决也是一个值得研究的问题.

References:

[1] Xia X, Zeng L, Yu R. HMM of telecommunication big data for consumer churn prediction [C] //Proceedings of 2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation, Guangzhou: Institute of Electrical and Electronics Engineers (IEEE), 2018: 1903-1910.

[2] Spanoudes P, Nguyen T. Deep learning in customer churn prediction: unsupervised feature learning on abstract company independent feature vectors [J]. ArXiv CoRR, 2017, abs/1703.03869.

[3] Arno De Caigny, Kristof Coussement, Koen W De Bock. A new hybrid classification algorithm for customer churn prediction based on logistic regression and decision treesp [J]. European Journal of Operational Research, 2018, 269(2): 760-772.

[4] Kaizhu H, Zheng D, Sun J, et al. Sparse learning for support vector classification [J]. Pattern Recognition Letters, 2014, 31(13): 1944-1951.

[5] Verbekea W, Martensb D, Baesens B. Social network analysis for customer churn prediction [J]. Applied Soft Computing, 2014, 14(3): 431-446.

[6] Stripling E, Broucke S, Antoniob K, et al. Profit maximizing logistic model for customer churn prediction using genetic algorithms [J]. Swarm and Evolutionary Computation, 2018, 40(14): 116-130.

[7] Caigny A D, Coussement K, Koen W. A new hybrid classification algorithm for customer churn prediction based on logistic regression and decision trees [J]. European Journal of Operational Research,

2018,269(2):760-772.

[8] Xia Guo-en,Tang Qi,Zhang Xian-quan. Application of improved multi-layer perceptron in customer churn prediction [J]. Computer Engineering and Application,2019,55(12):1-9.

[9] Guo G,Berkhahn F. Entity embeddings of categorical variables [J]. ArXiv CoRR,2016,abs/1604.06737.

[10] Michał Bereta. Regularization of boosted decision stumps using tabu search [J]. Applied Soft Computing Journal,2019,79(1):424-438.

[11] Chen Xu,Liu Lei,Deng Yu-bin. Vehicle detection based on visual attention mechanism and adaboost cascade classifier in intelligent transportation systems [J]. Optical and Quantum Electronics,2019,51(263):1-18.

[12] Shangkun Deng,Chenguang Wang,Mingyue Wang,et al. A gradient boosting decision tree approach for insider trading identification: an empirical model evaluation of China stock market [J]. Applied Soft Computing,2019,83(3):5652-5661.

[13] Chen Tianqi,Carlos Guestrin. XGBoost: a scalable tree boosting system [C]//Proceedings of the 22nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining,San Francisco: Knowledge Discovery and Data Mining(KDD),2016:785-794.

[14] Shi Xiu-peng,Yiik Diew,Michael Lee. A feature learning approach based on XGBoost for driving assessment and risk prediction [J]. Accident Analysis & Prevention,2019,129(9):170-179.

[15] Ma Xiao-jun,Sha Jing-lan,Wang De-hua. Study on a prediction of P2P network loan default based on the machine learning LightGBM and XGboost algorithms according to different high dimensional data cleaning [J]. Electronic Commerce Research and Applications,2018,31(2):24-39.

[16] Chen Cheng,Zhang Qing-mei,Ma Qin. LightGBM-PPI: predicting protein-protein interactions through LightGBM with multi-information fusion [J]. Chemometrics and Intelligent Laboratory Systems,2019,191(15):54-64.

[17] Mironov S V,Sidorov S P. Duality gap estimates for weak chebyshev greedy algorithms in banach spaces [J]. Computational Mathematics and Mathematical Physics,2019,59(6):904-914.

[18] Huang Guo-min,Wu Li-feng,Ma Xin. Evaluation of CatBoost method for prediction of reference evapotranspiration in humid regions [J]. Journal of Hydrology,2019,574(4):1029-1041.

[19] Arkaprabha Sau,Ishita Bhakta. Screening of anxiety and depression among seafarers using machine learning technology [J]. Informatics in Medicine Unlocked,2019,22(7):228-249.

[20] Vikrant A Dev,Mario R Eden. Formation lithology classification using scalable gradient boosted decision trees [J]. Computers & Chemical Engineering,2019,128(2):392-404.

[21] Chang Xi-ming,Wu Jian-jun,Liu Hao. Travel mode choice: a data fusion model using machine learning methods and evidence from travel diary survey data [J]. Transportmetrica a-Transport Science,2019,15(2):1587-1612.

[22] Jonathan Aaron Cook. ROC curves and nonrandom data [J]. Pattern Recognition Letters,2017,85(1):35-41.

[23] Li Yan,Liu Aie,Ding Li. Machine learning assessment of visually induced motion sickness levels based on multiple biosignals [J]. Biomedical Signal Processing and Control,2019,49(9):202-211.

[24] Guang-Hui Fu,Feng Xu,Bing-Yang Zhang,et al. Stable variable selection of class-imbalanced data with precision-recall criterion [J]. Chemometrics and Intelligent Laboratory Systems,2017,171(15):241-250.

附中文参考文献:

[8] **夏国恩,唐 琪,张显全. 改进的多层感知机在客户流失预测中的应用 [J]. 计算机工程与应用,2019,55(12):1-9.**