文章编号：2095-2163(2023)07-0064-08 中图分类号：TP399 文献标志码：A

# 基于最优基模型集成算法的信贷违约预测研究

高艺婕
（上海外国语大学 数据科学与大数据技术系，上海 201620）

**摘 要**：为了保障金融机构的金融安全，应用机器学习进行信贷违约预测已成为研究重点。为此，构建了 6 个机器学习基模型，调至最优参数后再分别用 Voting、Stacking、Adaboost 方法集成。实验表明，在多种基模型中，随机森林（RF）取得了较好的效果；而在集成方法中，Adaboost 对基模型的提升最显著。文中构建的 Adaboost-RF 模型在信贷预测上的交叉验证得分达到了0.904，明显优于其它方法，该方法对金融机构制定信贷决策具有一定的借鉴意义。

**关键词**：信贷预测；机器学习；集成学习；随机森林

# Study on credit default prediction based on optimal base model ensemble algorithm

GAO Yijie
（Department of Data Science and Big Data Technology, Shanghai International Studies Univesity, Shanghai 201620, China）

**【Abstract】** In order to ensure the financial safety of financial institutions, the application of machine learning in credit default prediction has become a research focus. To this end, six machine learning base models are constructed, and after tuning to optimal parameters, they are integrated separately using Voting, Stacking and Adaboost methods. The experiment shows that among multiple base models, the Random Forest (RF) achieves better results; while in the ensemble methods, Adaboost had the most significant improvement on the base models. The Adaboost-RF model achieves a cross-validation score of 0.904 in credit prediction, which is significantly better than other methods, and this method has certain reference value for financial institutions in making credit decisions.

**【Key words】** credit forecasting; machine learning; integrated learning; Random Forest

## 0 引 言

历史证明，信贷风险是商业银行最主要的风险之一，因此有效地预测和管理信贷风险对于银行和其他金融机构都具有重要意义[1]。随着大数据和人工智能技术的发展，银行收集到的大量用户行为数据能够得到更好的利用，传统信贷模式下信息不对称、违约风险高等问题也随即获得了解决和改善的有效途径和方法[2]。

预测借款人是否违约是一个二元分类问题。由于机器学习可以解决大样本量和多特征之间的复杂关系，故可广泛用于信贷违约预测[3-4]。为此，本文研究考虑了随机森林、决策树、逻辑回归、K 近邻、朴素贝叶斯、BP 神经网络六种机器学习基模型，用贝叶斯优化分别寻找其最优参数，随后用 Voting、Starking、Adaboost 集成方法建立提高预测精度的集成学习模型。

## 1 数据来源与预处理

### 1.1 数据来源

本文的数据来源为 UCI 数据库中的台湾信贷数据集，包含 5 个数据维度、23 个数据指标。数据集中字段说明见表 1。由表 1 可知，申请人信息包含了性别、婚姻状况、受教育程度等，与客户的消费观念和消费行为具有一定的关联；历史偿还情况、历史账单金额、历史还款金额维度的指标则能体现借款人过去的消费习惯和信用状况，并以此为根据预测其未来的还款行为。

### 1.2 不平衡样本处理

总样本中正负样本的占比不均衡的时候，模型的输出就会偏向于多数类的结果。因此，对不平衡数据进行建模前应首先将其转化为平衡数据。本文所采用的数据共 30 000 条有效样本，其中正例仅有 6 636 条，占总样本的 22.2%；而负例有 23 364 条，

作者简介：高艺婕(2002-)，女，本科生，主要研究方向：机器学习、深度学习。
通讯作者：高艺婕 Email：gaoyijiegyj@ 163.com

收稿日期：2022-09-01

占总样本的 77.8%，显然是非平衡的样本。对于不均衡的数据，最直接、最有效的方法就是生成少数类的样本。这种方法称为过采样（Over Sampling），通过对少数类样本的随机采样增加样本个数，使得每类样本的比例为 1∶1。

**表 1　数据集字段说明**

**Tab. 1　Explanation of datasets**

| 维度 | 子项指标 | 指标说明 |
|---|---|---|
| 授信金额/元 | 银行对公司授予的信用额度 $x_{11}$ | 授信金额（元） |
| 申请人信息 | 贷款申请人性别 $x_{21}$ | 1 - 男；2 - 女 |
| | 贷款申请人受教育程度 $x_{22}$ | 1 - 研究生及以上；2 - 本科；3 - 高中；4 - 初中及以下 |
| | 贷款申请人婚姻状况 $x_{23}$ | 1 - 已婚；2 - 单身；3 - 其他 |
| | 贷款申请人年龄 $x_{24}$ | 年龄（岁） |
| 历史偿还情况 | 前 1 个月的还款情况 $x_{31}$ | - 1 - 按时还款；1 - 延迟还款 1 个月；2 - 付款还款 2 个月；……8 - 延迟还款 8 个月；9 - 延迟还款 9 个月或以上 |
| | 前 2 个月的还款情况 $x_{32}$ | |
| | 前 3 个月的还款情况 $x_{33}$ | |
| | 前 4 个月的还款情况 $x_{34}$ | |
| | 前 5 个月的还款情况 $x_{35}$ | |
| | 前 6 个月的还款情况 $x_{36}$ | |
| 历史账单金额 | 前 1 个月的账单金额 $x_{41}$ | 账单金额（元） |
| | 前 2 个月的账单金额 $x_{42}$ | |
| | 前 3 个月的账单金额 $x_{43}$ | |
| | 前 4 个月的账单金额 $x_{44}$ | |
| | 前 5 个月的账单金额 $x_{45}$ | |
| | 前 6 个月的账单金额 $x_{46}$ | |
| 历史还款金额 | 前 1 个月的还款金额 $x_{51}$ | 还款金额（元） |
| | 前 2 个月的还款金额 $x_{52}$ | |
| | 前 3 个月的还款金额 $x_{53}$ | |
| | 前 4 个月的还款金额 $x_{54}$ | |
| | 前 5 个月的还款金额 $x_{55}$ | |
| | 前 6 个月的还款金额 $x_{56}$ | |
| 标签 | 是否违约 $y$ | 0 - 否，1 - 是 |

## 2　信贷预测方法

### 2.1　机器学习方法

（1）逻辑回归（Logistic Regression）。是一种广义的线性回归分析模型，属于机器学习中的监督学习，常用来解决分类问题[5]。因为二分类问题的标签是 0 或 1，在特征和权值线性相乘累加之后，逻辑回归用逻辑函数将最后的预测值映射到[0，1]之间，能够避免线性回归的输出值远大于 1 或远小于 0 的问题[6]。本实验中的逻辑回归模型采用 *Sigmod* 逻辑函数，可用如下公式进行描述：

$$Sigmod(x) = \frac{1}{1 + e^{-x}} \tag{1}$$

$$y = \begin{cases} 1 & Sigmod(x) \geqslant 0.5 \\ 0 & Sigmod(x) < 0.5 \end{cases} \tag{2}$$

（2）决策树。在机器学习中，决策树是一种分类预测模型，表示对象属性与对象值之间的映射关系。该方法的训练步骤是：从一个根节点出发，对一个样本进行测试，不断将样本分配到分裂的子树上，直至该样本被指派到其对应的一个最小子节点上。通过这种方式，样本被递归地测试和分配，直至到达叶节点，最后分配给叶节点的类[7]。

在本实验中，采用基尼系数作为分裂标准的决策树、即 CART 决策树。这种决策树在信用评分领域被广泛使用。基尼系数的计算公式见如下：

$$Gini = \sum_{k=1}^{K} p_k(1 - p_k) = 1 - \sum_{k=1}^{K} p_k^2 \quad (3)$$

其中，$p_k$ 是在树的第 $k$ 个分区或节点中被错误分类的样本的比例。基尼系数越低、表示模型的区分度越高，越适合用于分类预测。

(3)随机森林(Random Forest)。是一个包含多个决策树的分类器，其输出的类别由其中所有单个分类器输出的众数而定。随机森林在使用 Bagging 技术集成决策树的基础上，进一步在决策树的训练过程中引入了随机选择机制，使每棵子树分别在随机采样的样本子集上训练[8]。此外，随机森林中，基决策树每个结点的划分属性也是从随机的属性子集中选择的。这种随机性使得随机森林中的每颗决策树能够学习到不同属性对分类的贡献，一定程度上避免了过拟合的问题[9]。

(4)K 近邻(K-Nearest Neighbor)。是基于先例的学习，是惰性学习一种。K 近邻示意如图 1 所示。研究中，已知某待分类的测试样本，可以选择适当的距离度量，确定训练集合中最相邻的 $k$ 个样本，并用其来实现预测分类[10]。对 K 最近邻法的分类效果，主要受距离计算方法、查询邻居的数目和判别方法等因素的影响。常见距离的计算公式是曼哈顿距离、欧氏距离、切尔谢夫距离。对于邻近点的数量 $k$，如果 $k$ 的数值比较小，那么邻近点的计算复杂度就会很高，并且很可能会发生过拟合；如果 $k$ 值较大，模型就会变得太过简单，产生欠拟合。所以，正确选取 $k$ 值对于分类效果有很大的影响。在选择分类决策算法的时候，通常采用的是多数票投票，也就是由输入实例的 $k$ 个近邻中所属类别最多的类别来确定输入实例的分类。

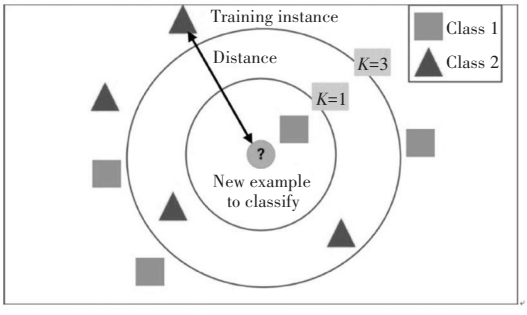



**图 1 K 近邻示意图**

**Fig. 1 Illustration of K-Nearest Neighbors**

(5)朴素贝叶斯。是一种经典的机器学习算法，其原理是以贝叶斯概率论为基础的，通过对先验概率和条件概率建模来进行分类判别。相较于其他绝大多数的分类算法、如决策树、KNN、逻辑回归和支持向量机等采用的判别方法，即对输出 $Y$ 和特征 $X$ 之间关系的直接学习，朴素贝叶斯更易于理解和实现。

朴素贝叶斯算法定义中有 2 个关键内容：特征之间强假设独立和贝叶斯定理。朴素贝叶斯的训练过程就是以概率模型为基础，在条件独立的前提下，根据已有训练集的分布来估计后验概率 $P(c \mid x)$：

$$P(c \mid x) = \frac{P(x \mid c)P(c)}{P(x)} = \frac{P(c)}{P(x)} \prod_{i=1}^{d} P(x_i \mid c) \quad (4)$$

(6)BP 神经网络。BP 神经网络模型的学习过程由正向传播和误差反向传播两个部分组成。在正向传递过程中，输入的数据由输入层传递，再由各个隐含层对其进行逐层处理，最后传递到输出层。当输出结果与预期结果不一致时，则通过误差反向传播的方式，对各神经元间的链接权重矩阵进行调整，从而达到减小误差的目的。在不断地学习中，将误差降低到了可以接受的程度。具体步骤是：

(1)从训练集中选择一批样本输入到神经网络中进行训练。

(2)通过各个节点之间的连通性，进行正向、逐层的传播，从而获得神经网络的实际输出。

(3)根据损失函数计算实际输出与期望输出的误差。

(4)把误差逆向传递到网络每一层，通过对损失函数的微分来改变网络每一层中的参数权重，将整体神经网络的预测朝着误差降低的方向进行调节。

(5)对于训练集中的每一个样本，重复上述步骤，直到整个训练样本集合的误差减小至满足要求[11]。

### 2.2 集成学习方法

(1)Stacking。在复杂的分类任务、如信贷预测任务中，可以使用多种机器学习方法，而这些方法有时差异并不明显。另外，使用单个模型的泛化能力往往比较弱。因此，使用模型集成的方法可以将这些模型都保留下来，结合多个模型的优点，提升总模型的预测精度。Stacking 就是一种典型的多模型集成学习方法，Stacking 示意如图 2 所示。研究可知，Stacking 就是一个多层模型，将第一层各模型的预测结果作为第二层的训练集，来学习一个新模型。这种方法通常被认为能够提高模型的准确性和稳健性。为了防止过拟合，第二层学习器宜选用简单模型。如在回归问题中，可以使用线性回归；在分类问题中，可以使用逻辑回归。

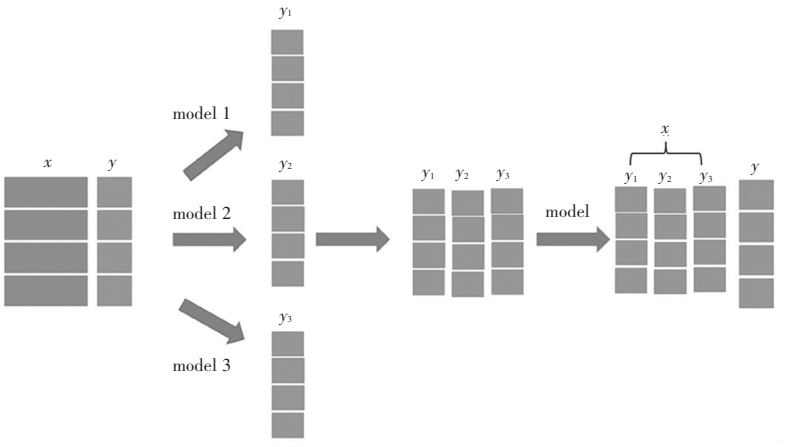



**图 2　Stacking 示意图**

**Fig. 2　Illustration of Stacking**

(2)投票集成算法(Voting)。是一种将多个基本模型进行组合的集成方法,通过对基模型的预测结果进行投票或加权投票,从而获得最终的预测结果,可以应用于分类和回归问题。常见的投票集成算法包括加权投票、软投票和硬投票等。加权投票是指为每个基本模型分配一个权重,并将其加权后进行投票。软投票是指对所有基本模型的预测结果进行加权平均,并将平均值作为最终预测结果。硬投票是指对所有基本模型的预测结果进行投票,并将得票最多的类别或数值作为最终预测结果。

(3)Adaboost。Boosting 集成算法是一种以弱分类器为基础,对弱分类器进行迭代增强,从而提高整合模型性能的方法。Boosting 算法的核心思想就是加强对训练集中被错误分类样本的训练,使这些样本在下一轮训练中得到更多的关注,从而提高分类器的性能。Boosting 算法的优点是能够在不增加模型复杂度的情况下提高分类器的准确性,同时也能够处理高维数据和噪声数据。其中, Schapire 和 Freund[12] 提出的 AdaBoost 算法是最具代表性的成果之一。

自适应增强(AdaBoost)、即 Adaptive Boosting,其自适应性表现在每轮迭代中,对于被前一个基本分类器误分类的样本,其权值会增加,而对于正确分类的样本,其权值会减小。依次调整样本权重后,再用这些样本来训练下一个基本分类器。在每一次迭代中,都将添加一个新的弱分类器,直到整体模型达到预定的错误率要求或预先指定的最大迭代次数,才会确定最终的强分类器。Adaboost 示意如图 3 所示。

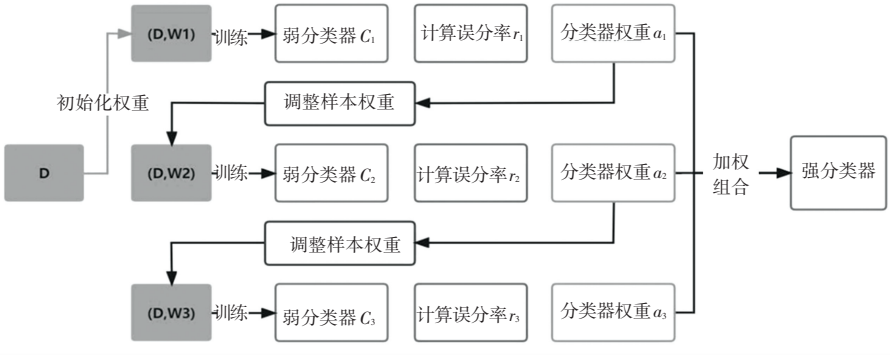



**图 3　Adaboost 示意图**

**Fig. 3　Illustration of Adaboost**

## 3　实验设置

### 3.1　贝叶斯优化调整参数

贝叶斯优化算法基于贝叶斯定理和高斯过程模型,是一种用于优化黑箱函数的方法。在贝叶斯优化中,通过不断地选择函数的输入来尝试优化函数的输出。每次选择都基于之前的尝试结果和先验知识,以便更好地探索和利用函数的特征。贝叶斯优

化常用于调参问题,其中目标是最小化目标函数的输出,而输入是超参数的取值。通过贝叶斯优化,相比网格优化和随机优化,可以在更少的尝试次数内找到较优的超参数组合。贝叶斯优化为待优化函数 $f: x \to D$ 构造了概率模型,利用该模型选择下一个评估点,依次迭代循环得到超参数最优解[13]。

假设有一组超参数组合 $X = x_1, x_2, \cdots, x_n$ 以及待优化函数 $f$, 贝叶斯优化假设超参数与待优化函数存在一个函数关系,需要在 $x \in X$ 内找到:

$$x^* = \underset{x \in X}{\operatorname{argmin}} f(x) \tag{5}$$

首先,根据不确定性和最小成本的原则,从采集函数中选择一些需要评估的候选点,即高斯过程回归中的决策规则:

$$y_{optimal} \mid x^* = \operatorname{argmin} \int p(y^* \mid x^*, X, y) L(y^*, y_{guess}) \, \mathrm{d}\, y^* \tag{6}$$

贝叶斯优化算法主要核心步骤是先验函数和采集函数两部分。其中,先验函数采用高斯回归过程,基于贝叶斯定理,将先验概率模型转换为后验概率分布。采集函数采用改进概率(probability of improvement, PI)选择下一个评估点[14]:

$$\operatorname{argmin} PI_n(x) = \operatorname{argmin} P(f(x) \geqslant \hat{y}_n^* + \in) = \operatorname{argmin} \varnothing\left(\frac{\mu_n(x) - \hat{y}_n^* - \in}{\delta_n(x)}\right) \tag{7}$$

### 3.2 评价指标选取

模型后训练完成后,需要选择合适的指标来衡量不同模型的效果,以确认最终选择哪个模型。对于二分类问题,样本分为正例和负例。在本实验中,正例表示信贷违约的客户,负例表示按时还款的客户[15]。对比样本的预测类别和真实类别,可以得到混淆矩阵,见表 2。在此基础上,又衍生出了一系列二分类模型评价指标。对此拟做阐释分述如下。

**表 2 混淆矩阵**

**Tab. 2 Confusion matrix**

| 真实类别 | 预测类别 | |
|---|---|---|
| | 1 | 0 |
| 1 | True Positive (*TP*) | False Negative (*FN*) |
| 0 | False Positive (*FP*) | True Negative (*TN*) |

(1)准确率 (*Accuracy*)。是衡量分类效果的一个常用指标,是对某一数据进行正确分类的样本数量在总样本数量中所占的比例。在样本均衡的条件下,准确率可以对模型进行客观的评估,但是在样本类型不均衡的情况下,尤其是在有极偏的数据的情况下,准确率很难对算法的优劣进行客观的评估。这里推得的 *Accuracy* 的计算公式为:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{8}$$

(2)召回率 (*Recall*)。是指正确预测的样本占总样本的比重。这里推得的 *Recall* 的计算公式为:

$$Recall = \frac{TP}{TP + FN} \tag{9}$$

(3)精确率 (*Precision*)。是预测为正的样本占所有样本的比重。这里推得的 *Precision* 的计算公式为:

$$Precision = \frac{TP}{TP + FP} \tag{10}$$

(4) $F_1$ 分数($F_1$ *score*)。精准率和召回率存在一定此消彼长的关系,这是因为模型要么更倾向于将错误样本预测为正,要么更倾向于将正确样本预测为负。为了综合考量精准率和召回率,提出了 $F_1$ *score*。这里推得的 $F_1$ *score* 的计算公式为:

$$F_1 = 2 * \frac{Precision * Recall}{Precision + Recall}$$

(5) *ROC* 曲线及 *AUC* 值。经过训练集的训练后,二分类模型可用于预测测试集数据的类别概率。为绘制 *ROC* 曲线,需将分类阈值在[0,1]区间不断调整,计算出对应的真正例率(*TPR*)和假正例率(*FPR*),并将其绘制在二维坐标系上,其中横轴为假正例率(*FPR*),纵轴为真正例率(*TPR*)。*ROC* 曲线越接近坐标系的左上角[0,1]点,说明分类器效果越好。*ROC* 曲线下的面积、即为 *AUC*(Area Under Curve)值,其意义在于反映正样本预测结果优于负样本预测结果的概率。因此,*AUC* 值可用于评估分类器对样本进行排序的能力,值越大代表分类器效果越好。

(6)交叉验证。交叉验证是对模型进行检验的一种方法,能够科学评价模型的优劣,筛选出性能最佳的模型,并能有效地防止过拟合和欠拟合。在本实验中,将采取五折交叉验证进行模型评估。交叉验证示意如图 4 所示。交叉验证的主要步骤如下:

① 将全部样本划分成 $k$ 个大小相等的样本子集。

② 依次遍历这 $k$ 个子集,每次把当前子集作为验证集,其余所有子集作为训练集,进行模型的训练和和测试。

③ 把 $k$ 次测试得分的平均值作为最终的交叉验证得分。

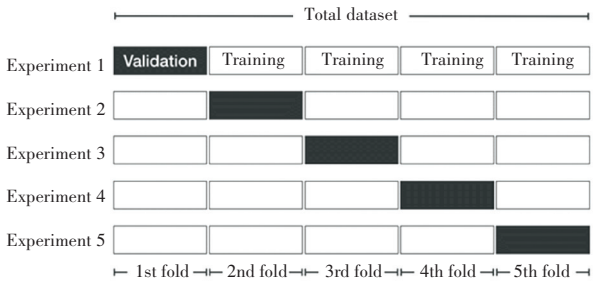



**图 4　交叉验证示意图**

**Fig. 4　Illustration of cross-validation**

## 4　实验结果

### 4.1　机器学习基模型的预测效果

对各模型使用贝叶斯算法调参，其最优参数见表 3。准确率、召回率、精确率、$F_1$ 分数和五折交叉验证得分见表 4。各个最优基模型在信贷数据集上预测的 *ROC* 曲线和 *AUC* 值如图 5 所示。

**表 3　各基模型最优参数**

**Tab. 3　Optimal parameters of each base model**

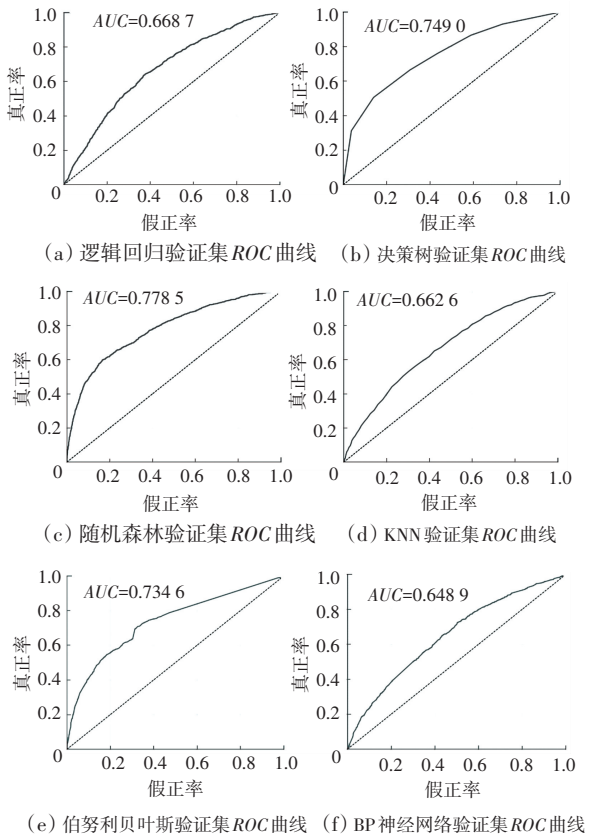
| 基模型 | 参数 | 最优参数 |
|---|---|---|
| 逻辑回归(lbfgs) | *C* | 1.075 168 076 152 866 6 |
| | *tol* | 9.548 893 804 190 575e－06 |
| 决策树 | *max_depth* | 9.256 784 156 276 586 |
| | *max_features* | 0.999 |
| | *min_samples_leaf* | 0.1 |
| | *min_samples_split* | 0.2 |
| 随机森林 | *max_depth* | 12.330 478 902 532 004 |
| | *max_features* | 0.165 964 189 255 098 76 |
| | *min_samples_split* | 18.454 128 266 811 42 |
| | *n_estimators* | 171.243 455 130 161 5 |
| K 近邻(brute) | *n_neighbors* | 88.379 395 152 199 17 |
| | *p* | 1(曼哈顿距离) |
| 朴素贝叶斯 | *alpha* | 0.923 633 695 662 881 |
| BP 神经网络(lbfgs) | *alpha* | 1e－05 |
| | *beta_1* | 0.9 |
| | *beta_2* | 0.999 |
| | *epsilon* | 1e－08 |
| | *hidden_layer_sizes* | (96,32) |
| | *learning_rate_init* | 0.077 |
| | *max_iter* | 2 000 |

**表 4　各基模型评价结果**

**Tab. 4　Performance of each base model**

| 基模型 | 交叉验证 | 准确率 | 召回率 | 精确率 | *F*1 分数 |
|---|---|---|---|---|---|
| 逻辑回归(LR) | 0.615 | 0.625 | 0.513 | 0.660 | 0.577 |
| 决策树(DT) | 0.687 | 0.683 | 0.860 | 0.635 | 0.731 |
| 随机森林(RF) | 0.729 | 0.714 | 0.836 | 0.672 | 0.745 |
| K 近邻(KNN) | 0.763 | 0.615 | 0.547 | 0.632 | 0.587 |
| 朴素贝叶斯(NB) | 0.681 | 0.679 | 0.799 | 0.644 | 0.713 |
| BP 神经网络(BP) | 0.591 | 0.600 | 0.623 | 0.595 | 0.609 |

实验结果表明，K 近邻和随机森林是表现较好的机器学习基模型，并在测试数据集上的五折交叉验证得分分别达到了 0.763 和 0.729，证明这 2 个模型有不错的分类能力。然而，K 近邻算法的准确率、召回率、精确率和 $F_1$ 分数都低于随机森林算法，表明 K 近邻算法容易产生过拟合，并不是足够可靠的模型，对数据的排序能力也弱于随机森林。对比之下，随机森林在拥有良好分类能力的同时，还表现出更高的可靠性和泛化能力。所以，最终选择随机森林作为最优基模型，并将其运用于后续的集成方法中，以进一步提升信贷违约预测的准确率。




(a) 逻辑回归验证集 *ROC* 曲线　(b) 决策树验证集 *ROC* 曲线

(c) 随机森林验证集 *ROC* 曲线　(d) KNN 验证集 *ROC* 曲线

(e) 伯努利贝叶斯验证集 *ROC* 曲线　(f) BP 神经网络验证集 *ROC* 曲线

**图 5　各基模型 *ROC* 曲线和 *AUC* 值**

**Fig. 5　*ROC* curves and *AUC* values of each base model**

### 4.2　Stacking 集成效果

用 Stacking 多模型集成方法将不同的模型堆叠起来，可以结合多个模型的学习结果，提升总模型的预测精度。选取基模型中表现较好、可靠性也较好的朴素贝叶斯算法、随机森林算法、决策树算法和逻辑回归算法进行实验。经过 Stacking 集成后，集成模型的准确率比单个模型都有了显著的提升，结果见表 5。由表 5 分析发现，逻辑回归、随机森林、决策树的组合以及朴素贝叶斯、随机森林、决策树的组合取得了更好的效果，在测试集上的交叉验证得分为 0.815。

表 5 Stacking 集成效果

Tab. 5 Performance of Stacking ensemble model

| 模型组合 | 五折交叉验证得分 |
|---|---|
| NB+RF+LR | 0.808 |
| LR+RF+DT | 0.815 |
| NB+RF+DT | 0.815 |
| NB+RF+DT+LR | 0.812 |

### 4.3 Voting 集成效果

用 Voting 集成方法组合不同的模型同样选取朴素贝叶斯算法、随机森林算法、决策树算法和逻辑回归算法进行实验,实验结果见表 6。使用相同模型时,软投票的效果优于硬投票,然而其效果仍然不如 Stacking 方法。Voting 方法仅将多个模型的学习效果结合在一起,整体模型只进行了一层训练。而 Stacking 模型整体训练了 2 层,比起 Voting 方法集成效果更好。

表 6 Voting 集成效果

Tab. 6 Performance of Voting ensemble model

| 模型组合 | 五折交叉验证得分 |
|---|---|
| NB+RF+LR(软投票) | 0.796 |
| LR+RF+DT(软投票) | 0.812 |
| NB+RF+DT(软投票) | 0.792 |
| NB+RF+DT+LR(软投票) | 0.802 |
| NB+RF+LR(硬投票) | 0.789 |
| LR+RF+DT(硬投票) | 0.802 |
| NB+RF+DT(硬投票) | 0.795 |
| NB+RF+DT+LR(硬投票) | 0.795 |

### 4.4 Adaboost 集成效果

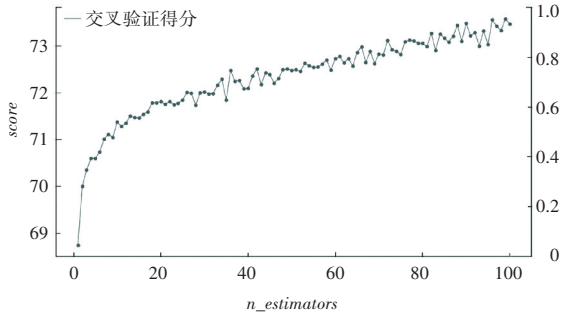
用 Adaboost 方法对随机森林基模型进行集成,并分别实验了基模型数量为 1~100 不同情况时的模型效果。Adaboos 集成效果值见表 7。

表 7 Adaboost 集成效果

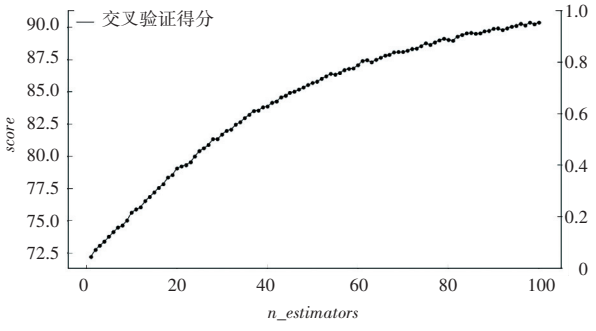
Tab. 7 Performance of Adaboost ensemble model

| 集成模型 (*n_estimators* = 100) | 五折交叉验证最高分 |
|---|---|
| DT | 0.736 |
| RF | 0.904 |

Adaboost 能够通过迭代来强化弱分类器的性能,以提升整个集成模型的性能。Adaboost 集成效果对比曲线如图 6 所示。实验发现,迭代的模型数量越多,Adaboost 集成的分类效果越好,且这种优化的趋势在模型数量刚开始增加时最明显,后续逐渐趋于平缓。基于决策树的 Adaboost 模型五折交叉验证得分最高为 0.736;而模型迭代到足够次数时,基于随机森林的 Adaboost 模型五折交叉验证得分能够稳定地保持在 0.9 以上,最高达到了 0.904。Adaboost_RF 模型在多种基模型的多种集成方法尝试中对基模型的提升效果最为显著,取得了最好的分类效果。




(a) 不同 *estimators* 数量下模型交叉验证得分(Adaboost_DT)




(b) 不同 *estimators* 数量下模型交叉验证得分(Adaboost_RF)

图 6 Adaboost 集成效果对比

Fig. 6 Performance comparison of Adaboost ensemble model

## 5 结束语

本文通过贝叶斯优化调参方法,训练出了 6 种最优参数下的机器学习分类模型,并选取其中效果较好的基模型进行集成实验。其中,用 Adaboost 方法集成随机森林的信贷违约评估模型(RF-Adaboost)取得了最好的分类效果。该模型明显优于其他机器学习方法,对银行等金融机构的信贷决策提供了具有一定科学依据的参考。同时,也提出了一种具有泛用性的分类模型训练方法,能够被应用于更多不同的研究领域。

## 参考文献

[1] 陈霞. 信用逾期预测中不同机器学习模型对比分析[J]. 计算机系统应用,2022,31(10):382-388.

[2] 薛可桢,汤琪,朱鑫雨. 大数据视角下商业银行信贷业务的风险管控研究[J]. 中国商论,2023(04):106-108.

[3] LIU Yi, YANG Menglong, WANG Yudong, et al. Applying machine learning algorithms to predict default probability in the online credit market: Evidence from China [J]. International Review of Financial Analysis,2022,79:101971.

[4] RAFN G B, SEPPE V B, BART B, et al. Deep learning for credit scoring: Do or don't? [J]. European Journal of Operational Research,2021,295(1).

[5] 梁颢严. 利用 Logistic Regression 建立贷款申请最大化利润模型[J]. 中国集体经济,2020 (07):71-73.

(下转第 75 页)