Journal of System Simulation

Volume 33 | Issue 9 Article 27

9-17-2021

Classification of Flight Delay Based on Nonlinear Weighted XGBoost

Tang Hong
Aeronautics Engineering College, Air Force Engineering University, X'an 710038, China;

Wang Dong
Aeronautics Engineering College, Air Force Engineering University, X'an 710038, China;

Song Bo
Aeronautics Engineering College, Air Force Engineering University, X'an 710038, China;

Wenkui Chu
Aeronautics Engineering College, Air Force Engineering University, X'an 710038, China;

See next page for additional authors

Classification of Flight Delay Based on Nonlinear Weighted XGBoost

Abstract

Abstract: Aiming at the classification of flight delay under imbalance data, *a novel method based on nonlinear weighted XGBoost (extreme gradient boosting) is proposed. The imbalance of flight delay data and the influence for classification performance caused by the data imbalance are analyzed. A heuristic nonlinear weighting method based on sample proportion is proposed, and the negative log likelihood loss function is optimized. The real flight delay dataset is used to validate the performance of the classification algorithm.* The experiment results show that the proposed nonlinear weighted XGBoost algorithm can improve the classification accuracy of flight delay, while ensuing a high overall classification accuracy. Compared to traditional methods, the proposed algorithm has good performance of statistical metrics and performance curves.

Keywords

extreme gradient boosting, gradient boosting, flight delay, data imbalance

Authors

Tang Hong, Wang Dong, Song Bo, Wenkui Chu, and Linyuan He

Recommended Citation

Tang Hong, Wang Dong, Song Bo, Chu Wenkui, He Linyuan. Classification of Flight Delay Based on Nonlinear Weighted XGBoost[J]. Journal of System Simulation, 2021, 33(9): 2261-2269.

第 33 卷第 9 期 系统仿真学报© Vol. 33 No. 9
2021 年 9 月 Journal of System Simulation Sep. 2021

基于非线性赋权 XGBoost 算法的航班延误分类预测

唐红，王栋，宋博，褚文奎，何林远
(空军工程大学 航空工程学院，陕西 西安 710038)

摘要：针对数据不平衡背景下的航班延误分类预测问题，*提出一种非线性赋权的极限梯度提升(eXtreme Gradient Boosting, XGBoost)算法。基于航班延误数据的不平衡特性及数据不平衡对分类预测性能的影响分析，提出基于样本比例的启发式非线性赋权方法，改进负对数似然损失函数，采用网格搜索和交叉检验法确定最优参数，并采用真实的航班延误数据集进行分类预测。*实验结果表明：非线性赋权 XGBoost 算法能够在保持整体分类准确率的同时，能够提高对延误情况的分类预测准确率，统计指标和性能曲线均优于传统算法。

关键词：极限梯度提升；梯度提升；航班延误；数据不平衡
中图分类号：U8；TP391.9 文献标志码：A 文章编号：1004-731X (2021) 09-2261-09
DOI: 10.16182/j.issn1004731x.joss.20-0372

Classification of Flight Delay Based on Nonlinear Weighted XGBoost

Tang Hong, Wang Dong, Song Bo, Chu Wenkui, He Linyuan
(Aeronautics Engineering College, Air Force Engineering University, Xi'an 710038, China)

Abstract: Aiming at the classification of flight delay under imbalance data, *a novel method based on nonlinear weighted XGBoost (extreme gradient boosting) is proposed. The imbalance of flight delay data and the influence for classification performance caused by the data imbalance are analyzed. A heuristic nonlinear weighting method based on sample proportion is proposed, and the negative log likelihood loss function is optimized. The real flight delay dataset is used to validate the performance of the classification algorithm.* The experiment results show that the proposed nonlinear weighted XGBoost algorithm can improve the classification accuracy of flight delay, while ensuing a high overall classification accuracy. Compared to traditional methods, the proposed algorithm has good performance of statistical metrics and performance curves.

Keywords: extreme gradient boosting; gradient boosting; flight delay; data imbalance

引言

随着航空运输需求的迅猛增长，民航航班延误情况频发。据中国民航局统计报告，2018 年中国民航航班正常率仅为 80.13%[1]。航班延误不仅造成航空公司和乘客的损失，也影响整个民用航空高效运营。因此，研究民航延误规律，建立航班延误预测模型，有助于为航管部门和航空公司提供决策支持，提高民航运行效率。

针对航班延误问题，国内外研究者开展了大量研究。航班延误预测问题的关键在于从海量航班飞行数据中挖掘并学习航班延误规律，建立预测模型。传统研究采用 *K*-近邻(K-Nearest Neighbor, KNN)[2]、支持向量机(Support Vector Machine, SVM)[3-4]、贝叶斯方法[5]和决策树(Decision Tree, DT)[6]方法进行航班延误预测。文献[2]分析了航班延误时间与机场能见度数据的关系，并采用 KNN

收稿日期：2020-06-17 修回日期：2020-08-05
基金项目：国家自然科学基金(61701524)
第一作者：唐红(1967-)，女，硕士，副教授，研究方向为军事航空通信与导航。E-mail：th118th@163.com

算法预测航班的延误状况。文献[3-4]基于 SVM 回归建立航班延误预测模型，其中文献[4]分析了航班到港延误时间序列的混沌特性，并采用相空间重构处理构建支持向量回归模型的输入向量。文献[5]基于朴素贝叶斯(Naïve Bayes, NB)方法对气象数据和飞行数据进行特征选择，建立航班延误预测模型。文献[6]基于 C4.5 决策树构建航班延误预测模型。然而，航班延误数据信息包括机场数据、飞行数据、航空公司数据等多个数据属性，属于高维度大数据样本。已有研究方法大多针对小数据样本，处理高维度大数据样本时，需要对数据降维处理，容易造成信息损失，影响预测的正确性。

数据科学领域的大数据分析处理方法不断涌现，为海量数据挖掘和数据规律学习提供了有效的工具。极限梯度提升(eXtreme Gradient Boosting, XGBoost)算法[7-8]是一种适用于大规模数据集的并行集成算法[9]，具有多核并行运算、正则化提升、自定义目标函数、接口丰富、适合处理结构化数据等特点，且算法具有高准确率和可解释性。目前 XGBoost 算法已经在数据科学领域的 Kaggle、阿里天池等大数据竞赛中获得广泛应用，同时也逐渐应用于疾病诊断[10]、故障诊断[11-12]、用户分析[13-15]等领域。

本文提出一种非线性赋权的 XGBoost 算法(Nonlinear Weight-XGBoost, NW-XGBoost)用于航班延误分类预测。算法针对数据样本不平衡问题，提出基于样本比例和 sigmoid 函数确定样本权重，改进目标函数，实现对不平衡数据样本的有效学习和预测准确率的提高。

1 航班分类预测

1.1 航班数据集

航班数据采用美国交通运输部统计局统计的 2015 年航班运行数据[16]。数据集包括 580 万余条航班信息，包括航班时间、航线、航班号、航空公司、出发和到达机场、延误时间等 31 个特征。根据特征与航班延误相关程度以及特征信息的重复程度[17-18]，选择其中 11 个特征作为航班延误属性，选择“Arrival delay”作为分类预测对象，如表 1 所示。

表 1 11 种航班延误属性
Tab. 1 Attribute of 11 flight delay data

序号	属性	数据类型	描述
1	Month	数值	航班月份，取值范围：1-12
2	Day	数值	航班日期，取值范围：1-31
3	Day_of_week	数值	航班星期，取值范围：1~7
4	Airline	标称	航线，两位字母或数字
5	Flight_number	标称	航班号，三位或四位数字
6	Destination_airport	标称	目的机场，三位字母组成
7	Original_airport	标称	出发机场，三位字母组成
8	Air_time	数值	飞行时间，取值大于 0
9	Departure_time	数值	起飞时间，范围 00:00—24:00
10	Distance	数值	飞行距离
11	Arrival_delay	数值	延误时间

1.2 数据预处理

为了使数据能够用于分析模型，采用 one-hot 编码方式对表 1 中的标称属性和延误时间进行编码。其中对于分类预测对象“Arrival_delay”的 one-hot 编码为

$$Code=\begin{cases}1000 & \leqslant Arrival\ delay<10\,\text{min}\\ 0100 & 10\,\text{min}\leqslant Arrival\ delay<30\,\text{min}\\ 0010 & 30\,\text{min}\leqslant Arrival\ delay<60\,\text{min}\\ 0001 & Arrival\ delay>60\,\text{min}\end{cases} \tag{1}$$

式(1)根据延误程度将数据集划分为 4 类：正常、轻度延误、中度延误、严重延误，分别编码为 1000，0100，0010，0001。本文选取数据集中随机选取 2%的样本，剔除数据缺失样本后共获得 114 209 个有效样本。样本分布如图 1 所示。

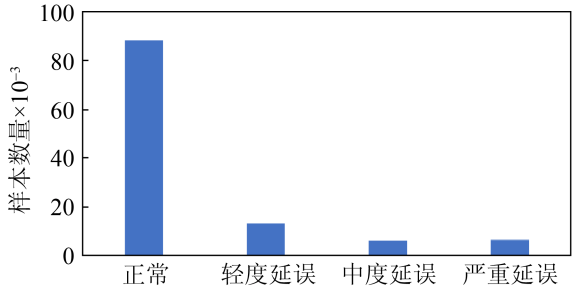

图 1 样本分布示意图
Fig. 1 Distribution of dataset

2 极限梯度提升算法

2.1 梯度提升决策树

梯度提升是集成学习中 Boosting 类算法的模型优化思想。这一思想的典型应用是梯度提升决策树(Gradient Boosting Decision Tree, GBDT)算法[19]。GBDT 由多个弱分类器线性加权构成，其数学模型可表示为

$$\hat{\boldsymbol{y}}_i = F(\boldsymbol{x}_i) = \sum_{m=1}^{M} f(\boldsymbol{x}_i;\boldsymbol{\theta}_m) \tag{2}$$

式中：$\boldsymbol{x}_i$ 为第 i 个训练样本；$F(\boldsymbol{x}_i)$为由 M 个分类器组合成的强分类器；$F(\boldsymbol{x}_i;\boldsymbol{\theta}_m)$为第 m 个弱分类器，参数为 $\boldsymbol{\theta}_m$。迭代训练过程中，通过增加弱分类器拟合上一次训练模型输出结果与样本数据的误差，实现模型性能的提升。新增弱分类器的参数优化问题可表示为

$$\hat{\boldsymbol{\theta}}_m = \arg\min_{\boldsymbol{\theta}_m} \sum_{i=1}^{N} l(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i) = \arg\min_{\boldsymbol{\theta}_m} \sum_{i=1}^{N} l(\boldsymbol{y}_i, F_{m-1}(\boldsymbol{x}_i) + f(\boldsymbol{x}_i;\boldsymbol{\theta}_m)) \tag{3}$$

式中：$l(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i)$ 为损失函数；$\boldsymbol{y}_i$ 为真实样本标签；$F_{m-1}(\boldsymbol{x}_i)$为当前模型；$\boldsymbol{\theta}_m$ 为待求解的新增分类器的参数。

在给定损失函数和样本数据的条件下，式(2)所示加法模型的学习是一个搜寻参数 $\boldsymbol{\theta}_m$ 使得损失函数极小化的过程。GBDT 算法的核心是使损失函数沿梯度方向下降方向减小。在每一轮迭代时，通过拟合当前模型结果与真实样本数据之间误差的负梯度学习新的分类器，实现模型的快速收敛和模型的性能提升。

2.2 极限梯度提升

梯度提升的关键在于目标函数的选取。GDBT 算法的目标函数仅包含损失函数 $l(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i)$。根据分类或回归问题的需要，GDBT 算法可以选择指数损失函数、似然对数损失函数、平方损失函数等。XGBoost 算法在 GDBT 算法损失函数的基础上，加入正则项，因此 XGBoost 算法的损失函数包括 2 部分：

$$L(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i) = \sum_{i=1}^{N} l(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i) + \sum_{m=1}^{M} \Omega(f_m) \tag{4}$$

式中：第 1 部分 $l(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i)$ 与 GBDT 算法损失函数相同，衡量模型对样本数据的拟合能力。第 2 部分为 $\Omega(f_m)$ 为正则项，用于控制模型复杂度，避免模型过拟合。XGBoost 算法将损失函数进行二阶泰勒展开，对于第 m 次迭代，式(4)可表示为

$$L_m(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i) = \sum_{i=1}^{N} l(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i^{m-1} + f_m(\boldsymbol{x}_i)) + \Omega(f_m) \simeq \sum_{i=1}^{N} [l(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i^{m-1}) + g_i f_m(\boldsymbol{x}_i) + \frac{1}{2} h_i f_m^2(\boldsymbol{x}_i))] + \Omega(f_m) \tag{5}$$

式中：g_i 和 h_i 分别 1 阶和 2 阶导数。定义正则项 $\Omega(f_m)$ 为

$$\Omega(f_m) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \tag{6}$$

式中：T 为叶子节点个数；w 为节点权重；γ 和 λ 分别为 l_1 正则和 l_2 正则的惩罚项参数。将式(6)代入式(5)，移除常数项 $l(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i^{m-1})$ 并化简可得最终的损失函数[7]：

$$L(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i) = \sum_{j=1}^{T} [G_j w_j + \frac{1}{2}(H_j + \lambda) w_j^2] + \gamma T \tag{7}$$

其中：$G_j = \sum_{i \in I_j} g_i$, $H_j = \sum_{i \in I_j} h_i$ 。

因此，计算式(7)的最小值，得到最优权重为

$$w_j^* = -G_j / (H_j + \lambda) \tag{8}$$

代入目标函数，可得：

$$L(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i) = -\frac{1}{2} \sum_{j=1}^{T} \frac{G_j^2}{H_j + \lambda} + \lambda T \tag{9}$$

3 航班延误分类预测模型

3.1 非线性赋权

根据图 1 的样本分布可知，不同延误程度的样本数量存在“长尾现象”，即样本数据分布不平衡。由于样本数量的差异，数量较多的“正常”类别将获得充分训练，而“中度延误”和“严重延误”的样本数量相对较少。传统的 XGBoost 算法以降低整体误差为目标，因此在模型学习过程中更多地关注多数类样本的分类预测性能，这将导致模型对少数类样本的分类性能得不到充分训练。在航班延误分类预测问题中，这也将影响模型对发生频次相对较少但造成实际影响更严重的延误情况的分类预测效果。

因此，本文提出一种非线性赋权方法提高数据不平衡下的 XGBoost 模型性能，基本思想是采用启发式函数对不同类别样本进行非线性赋权，样本数量与样本权重呈负相关。权重计算方法如下：

(1) 计算样本比例。

$$v_k = d_k / D \tag{10}$$

式中：D 为样本总数；d_k 为第 k 类样本数量；v_k 为第 k 类样本占总样本的比例。

(2) 计算非线性赋权函数。简单的基于样本比例的赋权思想可以采用式(10)中的 v_k 倒数作为权重，虽然能够提高少数类样本的权重，但是占样本多数的类别的权重减少幅度较大，可能导致权重差异过大。如果多数类样本占的比例远超少数类样本，则有可能出现多数类样本权重极小，而少数类样本权重过高的情况，可能导致模型训练效能低下，且影响模型整体正确率。因此本文提出基于样本比例的非线性赋权函数。

$$w_k = \frac{1}{2} + \alpha \cdot \frac{1}{1+\mathrm{e}^{v_k}} \tag{11}$$

式(11)所示的非线性赋权函数具有两大优点：①基于 sigmoid 函数的权重函数平滑可导；②由于权重过小将影响模型训练效率并导致过拟合，因此式(11)所示赋权函数中加入常数 0.5 以保证权重不会过小。根据式(11)可知函数的取值范围为 $[0.5+\alpha/(1+\mathrm{e}), 0.5+\alpha/2]$，$\alpha$ 为权重范围控制参数。在本文研究中，根据参数优化结果，将 α 值设为 1.5，得到的正常、轻度延误、中度延误、严重延误的权重分别为：0.974，1.208，1.228，1.230。

3.2 目标函数改进

本文研究的航班延误分类预测为多分类问题，因此选择负对数似然损失函数作为损失函数。对于第 k 类样本 x，其负对数似然损失函数可表示为

$$l(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i) = -\sum_k \boldsymbol{y}(k) \log \hat{\boldsymbol{y}}(k) \tag{12}$$

由于类别 k 采用 one-hot 编码，因此式(12)可进一步表示为

$$l(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i) = -\log \hat{\boldsymbol{y}}(k) \tag{13}$$

根据式(11)所示非线性赋权函数计算不同类别样本权重 w_k，可得改进后的损失函数为

$$l(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i) = -w_k \log \hat{\boldsymbol{y}}(k) \tag{14}$$

3.3 算法流程

本文提出一种基于非线性赋权 XGBoost 算法的航班延误预测算法。算法分为模型训练和测试验证阶段。模型训练阶段通过迭代训练，逐步增加新的分类器拟合当前迭代中的训练误差，优化模型对训练样本的拟合效果。测试验证阶段采用测试集验证模型的分类预测性能。算法流程如图 2 所示。

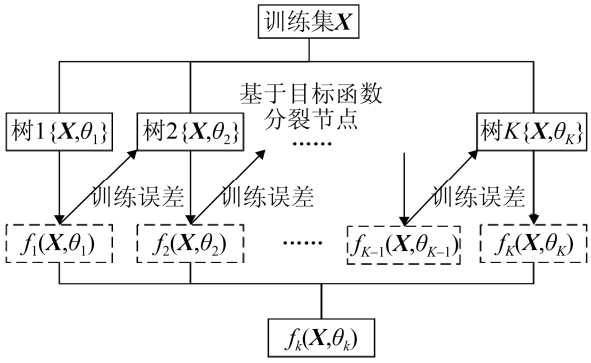

图 2 XGBoost 模型训练流程图

Fig. 2 Training flow chart of XGBoost model

4 实验分析

4.1 实验设置

实验数据包括 114 209 条真实的航班数据。其

DOI: 10.16182/j.issn1004731x.joss.20-0372

中 75%为训练集，25%为测试集。为了比较模型性能，将非线性赋权 XGBoost 算法与朴素贝叶斯(Naïve Bayes, NB)、Logistic 回归(Logistic Regression, LR)、随机森林(Radom Forest, RF)、决策树(Decision Tree, DT)、SVM、KNN、GBDT 以及 XGBoost 算法进行比较。各分类器的参数设置如表 2 所示。

表 2 分类器参数取值范围
Tab. 2 Parameters space of classifiers

分类器	参数设置
NW-XGBoost	树数量：[10, 30, 50, 100] 树最大深度：[3, 5, 8, 10] 最小叶子节点权重和：[1, 3, 6, 9] 学习速率参数：[0.05, 0.1, 0.15, 0.2] 权重控制参数：[0.5, 1, 1.5, 2, 2.5]
XGBoost	与 NW-XGBoost 设置相同
NB	多项式朴素贝叶斯分类器 平滑参数：[0.01, 0.1, 1]
LR	L2 惩罚项 停止求解门限：[0.000 1, 0.001, 0.01] 优化求解器：liblinear
SVM	核函数类型：RBF 核函数 惩罚系数：[0.01, 0.1, 1, 10] 核参数 δ:[0.01, 0.001, 0.0001]
KNN	近邻数量：[3, 5, 8, 10] 最大叶子数：[5, 8, 10, 30]
DT	树最大深度：[3, 5, 8, 10] 分裂准则：[gini, entropy]
RF	树数量：[10, 30, 50, 100] 最大深度范围：[3, 5, 8, 10] 分裂准则：[gini, entropy]
GBDT	树数量：[10, 30, 50, 100] 最大深度范围：[3, 5, 8, 10] 学习速率范围：[0.05, 0.1, 0.15, 0.2]

模型训练阶段采用网格搜索法搜索各个算法的所有可能的参数组合，对于每一种参数组合采用 3–折交叉验证实验，根据交叉验证结果确定各算法的最优参数。实验算法主要基于 Python 3.7.7 和 Scikit-learn 工具包实现，计算机配置为 Intel Core i5-8300H CPU@2.3 GHz 处理器，16 G 内存。

4.2 评价指标

为了评价不同模型的表现，本文基于混淆矩阵的相关统计指标和性能曲线比较不同模型性能，其中指标包括：准确率(Accuracy, ACC)、平均准确率(Balanced Accuracy, BACC)、F1-score 和 Kappa 系数；性能曲线包括：受试者工作特性(Receiver Operating Characteristic, ROC)曲线、精准率–召回率 (Precision-Recall, PR) 曲线以及学习曲线(Learning Curve)。

4.2.1 混淆矩阵相关指标

ACC 为正确分类的样本数与样本总数之比，反映模型对数据集整体的分类正确率，其定义如式(15)所示。然而，*ACC* 指标受样本中多数类分类准确率的影响更大，对于类别数量不平衡的数据集，*ACC* 指标的评价不够全面。因此要全面地衡量模型对不同类别样本的分类准确率，需要采用 *BACC* 指标。*BACC* 是模型对不同类别分类准确率的平均，其定义如式(16)所示。由于数量较少的类别与数量较多的类别在 *BACC* 指标计算中的贡献一致，因此 *BACC* 更能反映模型对不同类别的分类准确率。*ACC* 和 *BACC* 指标的定义：

$$ACC=\frac{TP+TN}{TP+TN+FP+FN}\times 100\% \tag{15}$$

$$BACC=\frac{1}{K}\left(\sum_{k=1}^{K}\frac{TP_k+TN_k}{TN_k+FP_k+TP_k+FN_k}\right)\times 100\% \tag{16}$$

式中：*TP*，*TN*，*FP*，*FN* 分别为真正、真负、假正、假负的样本数。

*F*1-*score* 是精确率和召回率的算术平均值：

$$F1\text{-}score=\frac{2\times Precision\times Recall}{Precision+Recall} \tag{17}$$

*F*1-*score* 综合了精确率和召回率的结果，能够客观全面地反映模型性能。*F*1-*score* 的取值范围为[0, 1]，当分数越接近 1，表示模型的分类性能越好。

Kappa 系数是统计学中用于一致性检验的指标，也可以用于衡量分类效果：

$$Kappa=(p_0-p_e)/(1-p_e) \tag{18}$$

式中： $p_0=ACC=(TP+TN)/(TP+TN+FP+FN)$；

$$p_e=\frac{(TP+FP)\cdot(TP+FN)+(TN+FN)\cdot(TN+FP)}{(TP+TN+FP+FN)^2}$$。

可以看出 *Kappa* 系数的取值范围为[–1, 1]，其中 p_0 即为 *ACC*，而 p_e 则为所有类别实际数量与预测数量的乘积的总和除以样本总数的平方。因此，在相同的 *ACC* 值下，如果模型对不同类别的分类准确率越高，则混淆矩阵越平衡，p_e 值越低，Kappa 系数越高。因此，可以用 *Kappa* 系数衡量模型在数据不平衡情况下的分类效果。

4.2.2 性能曲线

ROC 曲线和 PR 曲线可以直观地评估分类器性能。其中，ROC 曲线反映分类真正率(True Positive Rate, TPR)和假正率(False Positive Rate, FPR)之间的关系，PR 曲线反映精确率与召回率之间的关系。同时，可以通过比较不同模型的曲线下面积(Area Under Curve, AUC)大小来评估模型的分类性能。式(19)和(20)分别为真正率和假正率的求解公式。

$$FPR=\frac{FP}{FP+TN}\times 100\% \tag{19}$$

$$TPR=\frac{TP}{TP+FN}\times 100\% \tag{20}$$

学习曲线是在不同比例的训练集下，模型在训练集和交叉验证集上的得分变化曲线。学习曲线通常用于判断模型状态，比如判断模型为过拟合或者欠拟合，在分类问题中也可以通过分析不同模型的学习曲线的变化趋势，分析不同模型的收敛速度和分类性能。

4.3 性能分析

采用 4.1 节所述 9 个分类器对航班数据进行分类预测，得到各个分类器算法的 4 项统计指标、ROC 曲线和 PR 曲线。

4.3.1 混淆矩阵相关指标

首先对比不同模型在测试样本集上的分类统计指标，如表 3 所示。可以看到经过网格搜索和 3 折交叉验证后，除朴素贝叶斯分类器外，各个分类器均达到了较好的分类预测效果。综合来看，在各项指标中 NW-XGBoost 算法表现最好，其次为 XGBoost 算法，朴素贝叶斯算法的性能表现最差。通过在 XGBoost 算法基于样本比例进行非线性赋权改进负对数似然损失函数，模型不仅保持了较高的整体分类准确率，而且提高了对少数类样本的分类准确率，这意味着模型能够提高对延误情况的分类预测能力，具有积极意义。

表 3 分类器准确率对比
Tab.3 the comparison of classifiers

分类器	*ACC*/%	*BACC*/%	*F*1-*score*	*Kappa*
NB	42.84	29.33	0.23	0.05
KNN	77.03	26.11	0.24	0.03
LR	77.34	25.02	0.22	0.01
SVM	77.34	25.00	0.22	0.00
DT	77.46	26.53	0.25	0.03
RF	77.52	25.91	0.24	0.02
GBDT	77.78	27.52	0.26	0.06
XGBoost	77.78	27.79	0.27	0.07
NW-XGBoost	**77.85**	**28.36**	**0.28**	**0.08**

4.3.2 ROC 曲线

ROC 曲线是衡量分类器算法泛化性能的有效工具，其曲线下面积值反映算法的分类准确性，AUC 值越大表明分类器性能越好。同时，ROC 曲线具有样本分布变化时曲线形状保持不变的特点，因此适用于样本数据不平衡情况下的分类器性能评估。

图 3 为各分类器算法的 ROC 曲线以及 ROC 曲线的 AUC 值比较。由图 3 可知，9 种分类器中 NW-XGBoost 算法的 AUC 值最高，达到 0.911。

4.3.3 PR 曲线

PR 曲线能够反映分类器精确率与召回率之间的关系。与 ROC 曲线相比，PR 曲线对样本数据不平衡非常敏感，当样本分布极不均匀时，分类器的 ROC 曲线形状保持不变，而 PR 曲线表现变差。因此，PR 曲线更能反映数据不平衡条件下，分类器的性能优劣。同样地，采用 PR 曲线 AUC 值比较分类器性能，AUC 值越高，分类器性能越好。

DOI: 10.16182/j.issn1004731x.joss.20-0372

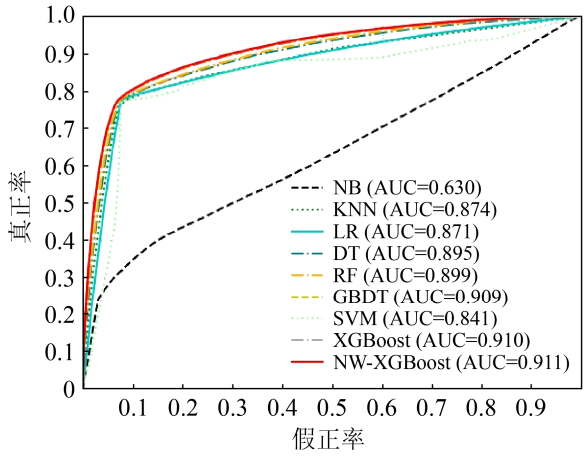

图 3 算法 ROC 曲线对比
Fig. 3 Comparison of ROC

图 4 为分类器 PR 曲线及 PR 曲线的 AUC 值比较，可以看出，NW-XGBoost 算法具有更高的 AUC 值，优于其他 8 种分类器。

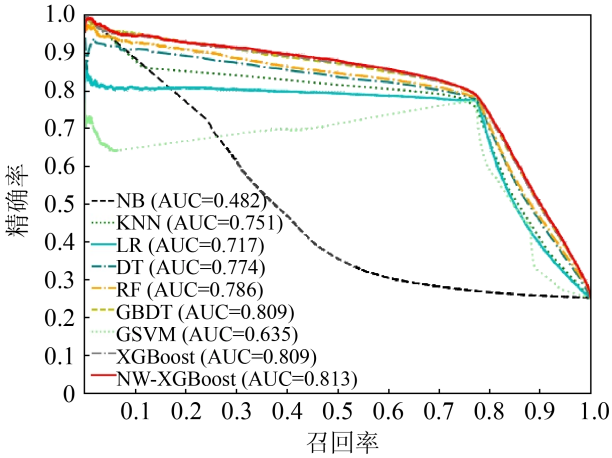

图 4 算法 PR 曲线对比
Fig. 4 Comparison of PR

4.3.4 学习曲线

模型对数据集的学习能力和分类性能可以通过学习曲线进行评估：一方面，通过对模型的学习曲线变化趋势衡量模型学习的收敛速度；另一方面，可以从最终的得分来衡量模型的学习效果。在实验中，按照不同比例将训练样本划分为训练集和交叉验证集，得到了模型在训练集和交叉验证集上的学习曲线，如图 5 和图 6 所示。

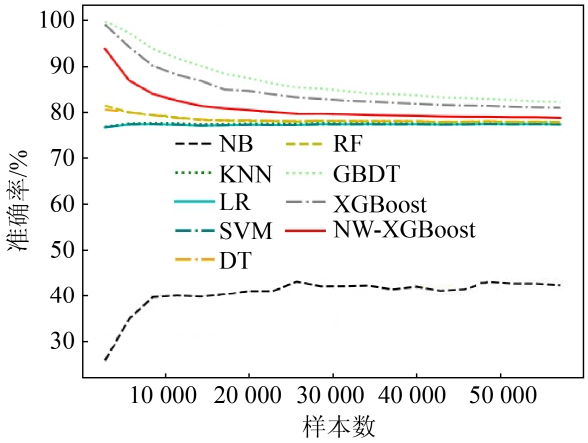

图 5 训练集上的学习曲线对比
Fig. 5 Learning curves on training set

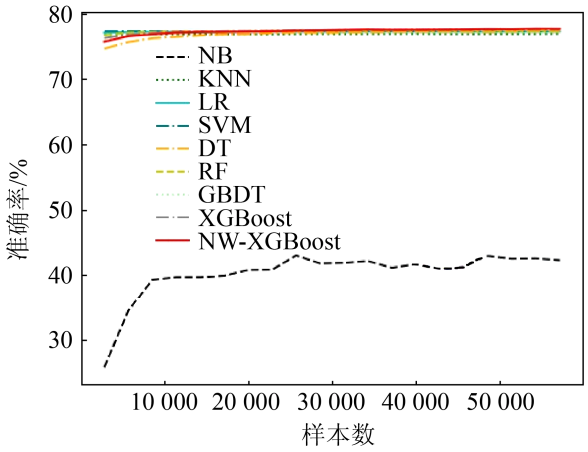

图 6 交叉验证集上的学习曲线对比
Fig. 6 Learning curves on cross-validation set

从图 5 可以看到，NW-XGBoost 算法具有最快的收敛速度，其次分别是 XGBoost 算法和 GBDT 算法，NB 算法在训练集和交叉验证集上的得分最低，其余算法则进入过拟合状态，学习提升幅度很小。同时，从图 6 可以看到 NW-XGBoost 算法对交叉验证集的得分最高，这与表 3 中算法对测试集的实验结果一致。

综合各项指标和性能曲线结果，可以看到 NW-XGBoost 算法能够实现良好的航班分类预测性能。其中，非线性赋权方法可以在保证整体准确率的同时，提高少数类的分类准确率。同时，XGBoost 算法也体现出了相比于传统分类器在收敛速度和学习能力上的优势。需要指出的是，由于 NW-XGBoost 算法相比于 XGBoost 算法增加了权

重控制参数 α，而且其参数范围中有 5 种可能的取值，因此在参数寻优过程中 NW-XGBoost 算法的参数组合是 XGBoost 算法的 5 倍，需要更多的参数寻优时间。

5 结论

准确的航班延误分类预测有助于实现民航系统的高效运营。本文在深入分析真实航班延误数据的不平衡特点的基础上，提出基于非线性赋权 XGBoost 的航班延误分类预测算法。算法基于样本比例和非线性赋权思想改进目标函数，提高了算法在数据样本不平衡下的分类预测能力。理论分析和试验结果表明，相比于传统算法，基于非线性赋权的 XGBoost 算法能够实现较高的分类准确率，具有更好的分类预测性能。

参考文献：

[1] 中国民用航空局. 2019 年全国民航工作会议专题速递[EB/OL] [2020-06-16]. http://www.caac.gov.cn/ZTZL/RDZT/2019Q GMHGZHY/.
Civil Aviation Administration of China. 2019 National Civil Aviation Working Conference (Special Express) [EB/OL] [2020-06-16]. http://www.caac.gov.cn/ ZTZL/RDZT/2019QG MHGZHY/.

[2] 续长青, 王永忠. 基于 KNN 算法的双流机场航班延误时间预测研究[J]. 信息技术与信息化, 2019(2): 81-84.
Xu Changqing, Wang Yongzhong. Research on Flight Delay Prediction of Shuangliu Airport Based on KNN Algorithm[J]. Information Technology & Informatization, 2019(2): 81-84.

[3] 何洋, 朱金福, 周秦炎. 基于支持向量机回归的机场航班延误预测[J]. 中国民航大学学报, 2018, 36(1): 30-36.
He Yang, Zhu Jinfu, Zhou Qinyan. Airport Flight Delay Prediction Based on SVM Regression[J]. Journal of Civil Aviation University of China, 2018, 36(1): 30-36.

[4] 罗赟骞, 陈志杰, 汤锦辉, 等. 采用支持向量机回归的航班延误预测研究[J]. 交通运输系统工程与信息, 2015, 15(1): 143-149.
Luo Yunqian, Chen Zhijie, Tang Jinhui, et al. Flight Delay Prediction Using Support Vector Machine Regression[J]. Journal of Transportation Systems Engineering and Information Technology, 2015, 15(1): 143-149.

[5] M D Hellwig. Predicting Irregular Flight Operations Using a Binary Machine Learning Approach Based on National Meteorological Data[D]. Grand Forks: University of North Dakota-Grand Forks, 2014.

[6] 程华, 李艳梅, 罗谦, 等. 基于 C4.5 决策树方法的到港航班延误预测问题研究[J]. 系统工程理论与实践, 2014, 34(增 1): 239-247.
Cheng Hua, Li Yanmei, Luo Qian, et al. Study on Flight Delay with C4.5 Decision Tree Based Prediction Method[J]. System Engineering Theory and Practice, 2014, 34(S1): 239-247.

[7] Chen T Q, Guestrin C. XGBoost: a scalable tree boosting system[C]// ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Francisco: USA: ACM, 2016: 785-794.

[8] Chen T Q, He T, Benesty M. Xgboost: Extreme Gradient Boosting[R], R Package Version 0.4-2, 2015: 1-4.

[9] 王桂兰, 赵洪山, 米增强. XGBoost 算法在风机主轴承故障预测中的应用[J]. 电力自动化设备, 2019, 39(1): 73-77.
Wang Guilan, Zhao Hongshan, Mi Zengqiang. Application of XGBoost Algorithm in Prediction of Wind Motor Main Bearing Fault[J]. Electric Power Automation Equipment, 2019, 39(1): 73-77.

[10] Pang L, Wang J, Zhao L, et al. A Novel Protein Subcellular Localization Method with CNN-XGBoost Model for Alzheimer's Disease[J]. Frontiers in Genetics (S1664-8021), 2019, 9: 751.

[11] Zhang D, Qian L, Mao B, et al. A Data-Driven Design for Fault Detection of Wind Turbines Using Random Forests and XGBoost[J]. IEEE Access (S2169-3536), 2018, 6: 21020-21031.

[12] 赵洪山, 闫西慧, 王桂兰, 等. 应用深度自编码网络和 XGBoost 的风电机组发电机故障诊断[J]. 电力系统自动化, 2019, 43(1): 81-90.
Zhao Hongshan, Yan Xihui, Wang Guilan, et al. Fault Diagnosis of Wind Turbine Generator Based on Deep Autoencoder Network and XGBoost[J]. Automation of Electric Power Systems, 2019, 43(1): 81-90.

[13] Tadesse M M, Lin H, Xu B, et al. Personality Predictions Based on User Behavior on the Facebook Social Media Platform[J]. IEEE Access (S2169-3536), 2018, 6: 61959-61969.

[14] Xia Y, Liu C, Li Y Y, et al. A Boosted Decision Tree Approach Using Bayesian Hyper-Parameter Optimization for Credit Scoring[J]. Expert Systems with Applications (S0957-4174), 2017, 78: 225-241.

[15] Li Y, Yang L, Yang B, et al. Application of Interpretable Machine Learning Models for the Intelligent Decision[J]. Neurocomputing (S0925-2312), 2019, 333: 273-283.

[16] Kaggle. 2015 Flight Delays and Cancellations: Which Airline Should You Fly on to Avoid Significant Delays[EB/OL] [2020-06-16]. https://www.kaggle.com/usdot/flight-delays.

[17] 刘金元, 丁勇, 李涛. 基于梯度提升决策树的航班延误分类预测[J]. 数学的实践与认识, 2018, 48(4): 1-7.
Liu Jinyuan, Ding Yong, Li Tao. Classification of Flight Delay Based-on GDBT[J]. Mathematics in Practice and Theory, 2018, 48(4): 1-7.

[18] Cai K, Zhang J, Zhou C, et al. Using Computational Intelligence for Large Scale Air Route Networks Design[J]. Applied Soft Computing (S1568-4946), 2012, 12(9): 2790-2800.

[19] Friedman J H. Greedy Function Approximation: a Gradient Boosting Machine[J]. The Annals of Statistics (S0090-5364), 2001, 29(5): 1189-1232.