Statistics and Application 统计学与应用, 2021, 10(2), 193-208
Published Online April 2021 in Hans. http://www.hanspub.org/journal/sa
https://doi.org/10.12677/sa.2021.102019

Hans汉斯

基于Stacking集成学习的电影票房预测研究

姚雅琪，徐秀丽

燕山大学理学院，河北 秦皇岛
Email: 475162929@qq.com

收稿日期：2021年3月18日；录用日期：2021年4月2日；发布日期：2021年4月14日

摘　要

电影票房作为电影行业最为主要的收入来源，研究票房影响因素并对其进行预测，有利于电影行业的发展和投资者做出正确投资决策。该文结合人工智能的前沿理论研究，提出了一种基于Stacking集成学习的电影票房融合模型的预测方法。选取2017年至2019年票房排名前100的电影票房及相关影响因素数据进行分析，并清洗量化规约各影响因素，通过XGBoost算法计算特征重要性筛选主要影响因素；利用Stacking模型融合多个机器学习算法，构建电影票房预测模型，通过网格交叉验证优化模型参数，对比评估得Stacking集成学习模型较单个机器学习预测模型具有更好的预测效果，在电影票房预测方面有较高的应用价值。

关键词

电影票房，集成学习，XGBoost，Stacking模型融合

Prediction Research on Movie Box Office Based on Stacking Ensemble Learning

Yaqi Yao, Xiuli Xu

School of Science, Yanshan University, Qinhuangdao Hebei
Email: 475162929@qq.com

Received: Mar. 18th, 2021; accepted: Apr. 2nd, 2021; published: Apr. 14th, 2021

Abstract

Film box office is the most important sources of income in the film industry. Researching the box

文章引用: 姚雅琪, 徐秀丽. 基于 Stacking 集成学习的电影票房预测研究[J]. 统计学与应用, 2021, 10(2): 193-208.
DOI: 10.12677/sa.2021.102019

office influencing factors and predicting them will help the development of the film industry and investors to make correct investment decisions. This article combines the cutting-edge theoretical research of artificial intelligence and proposes a prediction method of movie box office fusion model based on Stacking ensemble learning. The paper selects the box office data of the top 100 films and the relevant influencing factors from 2017 to 2019, cleans and quantifies influencing factors and uses the XGBoost algorithm to calculate feature importance to screen the main influencing factors. Then the paper uses the Stacking model to integrate several machine learning algorithms to build a box office prediction model, optimizes model parameters through grid cross-validation. Through comparative evaluation, it can be seen that the stacking integrated learning model obtained has a better prediction effect than a single machine learning prediction model, the model has high application value in movie box office prediction.

Keywords

Movie Box Office, Integrated Learning, XGBoost, Stacking Model Fusion

1. 引言

随着以电影为代表的文化产业在我国近年来的高速发展，中国电影行业的软硬实力都有了长足的进步和显著的提升，无论是电影票房收入还是观影人数规模每年都在飞速增长，对一部电影的投资回报等许多因素也都成为了被关注的焦点问题，电影票房作为衡量一部影片成功与否的重要指标，电影票房的多少成了许多投资机构不断追逐的目标，对电影票房的预测也成了各个投资机构对电影投资的重要指标，同时电影票房的准确预测是保障电影发行能够得到足够的投资回报、控制风险的重要手段，对于电影的投资决策具有十分重要的实践意义。但由于电影自身的特点，其生命周期较短且影响票房的因素多、量化难度大，想准确预测电影票房十分困难。

电影票房是衡量一部影片成功与否的重要指标，是电影利润的直接来源。近年来，大数据技术的发展使得票房预测模型得以进一步发展。研究者对于已有的电影票房预测模型从两个方向进行改进：一是引入新的影响因素和量化方法，其趋势是根据新的市场环境引入越来越多的变量和度量方法。二是改进所使用的回归或分类算法，其趋势是逐渐从线性算法变为非线性算法。Ramesh 和 Dursun 使用来自美国主要剧院公司的电影票房数据作为样本，并使用多层感知器神经网络预测票房，并获得了比回归方法更好的结果[1]。Lee 等利用贝叶斯网络计算电影票房成功概率，并且发现该模型的预测准确度高于神经网络模型[2]。Barman 等使用反馈神经网络算法预测电影票房的盈利情况，并有较好的预测效果[3]。2015 年 Kim 等人利用 SNS 数据基于机器学习算法建立了票房预测模型，实验结果证明预测模型的准确率高达 95% [4]。

与国外相比，国内电影票房预测仍处于探索阶段，没有较为成熟的有针对性的预测模型。郑坚等人利用神经网络构建 BRP 模型，利用导演、演员、类型、档期、国别等七个输入因子，构建有两个隐藏层的神经网络，进而对电影上映后每周的票房进行预测[5]。罗晓芃采用 GMM 估计建立了票房的预测模型，发现每天的票房都会对后一天的票房成绩产生显著的影响，并验证了电影的多种特征信息都会对票房产生显著的显著影响[6]。杨朝强提出了一种基于 LSTM 模型的电影票房算法，解决 BP 神经网络电影票房

预测精度不高的问题[7]。陈邦丽使用最小角回归(LARS)算法进行因素选取，再利用支持向量回归(SVR)算法对所选因素建立预测模型[8]。本文则根据对电影票房的影响因素进行研究分析，筛选量化主要影响因素，并通过集成学习的融合模型构建票房预测模型，为投资者减少了风险，对我国电影产业起到了积极作用，有助于院线对已上映的电影合理排片，并为未上映的电影宣传以及制作策略提供参考。

2. 预备知识

2.1. 回归模型

对于回归问题机器学习有很多模型，论文选取常用的三个模型 Ridge 回归、Lasso 回归、ElasticNet、与集成学习的 Boosting 中的改进模型进行对比。对于简单的线性回归，当样本特征很多且样本数量相对较少时，模型很容易陷入过度拟合。为了缓解过拟合问题，则采用减少特征数量和正则化的方法，正则化是指减少特征参数 W 的大小，是在线性回归损失函数的基础上增加一个正则化项，进而选择经验风险和模型复杂度同时较小的模型，Ridge 回归、Lasso 回归、ElasticNet 回归就是利用此原理避免模型过拟合，在保证最佳拟合误差的同时，使参数尽可能的简单，增强模型的泛化能力。

1) Ridge 回归

Ridge 回归在一般线性回归的损失函数中添加一个 $L2$ 正则化项，其损失函数为

$$l=\frac{1}{2m}\sum_{i=1}^{N}\left(y^{(i)}-\sum_{j}w_i x_j^{(i)}\right)^2+\frac{\lambda}{2}\sum_{j}w_j^2$$

$L2$ 正则假设参数的先验分布是 Gaussian 分布，可以保证模型的稳定性，使参数的值不会过大或者过小。$L2$ 范数是每个参数的平方和再求平方根，令 $L2$ 范数的正则项最小，则特征参数 W 接近于 0。但是与 $L1$ 范数不同，它不会使元素为 0，而仅接近 0。参数越小，模型越简单，过度拟合的可能性就越小。岭回归是带二范数惩罚的最小二乘回归。当特征是低维稠密的，使用 $L2$ 正则。

2) Lasso 回归

Lasso 回归在一般线性回归的损失函数中添加了一个正则化项，其损失函数为

$$l=\frac{1}{2m}\sum_{i=1}^{N}\left(y^{(i)}-\sum_{j}w_i x_j^{(i)}\right)^2+\lambda\sum_{j}\left|w_j\right|$$

$L1$ 正则假设参数的先验分布是 Laplace 分布，可以保证模型的稀疏性，$L1$ 正则化是指特征参数 W 中各元素的绝对值之和。Lasso 回归的特征在于，在构建广义线性模型时，Lasso 对数据要求低，同时还能过滤变量并降低模型的复杂性，并有选择地将变量放入模型中以获得更好的性能参数对于高维稀疏特征的数据，就使用 Lasso 回归。

3) ElasticNet 回归

ElasticNet (弹性网络回归)，当使用 Lasso 模型出现太多特征被稀疏为 0 时，同时岭回归的正则化也不够，即回归系数衰减太慢时，则考虑使用 ElasticNet 回归来进行，ElasticNet 综合 $L1$ 和 $L2$ 正则化，其损失函数为

$$l=\frac{1}{2m}\sum_{i=1}^{N}\left(y^{(i)}-\sum_{j}w_i x_j^{(i)}\right)^2+\lambda\alpha\sum_{j}\left|w_j\right|+\frac{1-r}{2}\alpha\sum_{j}w_j^2$$

2.2. 集成学习算法

集成学习是机器学习中的有监督学习算法，将多个学习器结合起来完成预测任务，比单一学习器获

得显著优越的泛化性能。机器学习的目的是学习到一个稳定且表现良好的模型，但是很多时候我们得到的模型是有偏好的，为了解决这一问题，考虑将多个学习器(有偏好)结合起来提高学习能力。集成学习算法分为两类 Bagging 和 Boosting，前者在并行生成时保持基本学习器的独立性，通过有放回的随机抽样方法来训练多样化的基本学习器，降低方差，其代表算法是随机森林(Random)。后者保持基础学习器串行生成时的依赖关系，通过对错判训练样本重新赋权来重复训练，以提高基础学习器准确性，降低偏差，其代表算法是 AdaBoost、GBDT 和 XGBoost。对于学习器的结合策略有三大类：投票法(分类)、平均法(连续数值)、学习法(Stacking)。本文选用 Boosting 中的 GBDT 算法、XGBoost 算法、LigthGBM 算法进行电影票房的预测。

GBDT 算法(Gradient boosting Decision Tree)是梯度提升决策树，将梯度提升算法与决策树相结合。主要思想是通过弱分类器即决策树迭代训练以得到最优模型，该模型具有训练效果好、不易过拟合等优点。其主要包括三部分：回归树，总体流程类似于分类树，但是回归树的每个节点都会得到一个预测值；提升树，是迭代多颗回归树来共同决策，即整个迭代过程生成的回归树的累加；梯度梯度提升决策树,提升树利用加性模型(弱学习器的线性组合)和前向分步算法实现学习的通过梯度下降法求解最优化的过程。

XGBoost 极限梯度提升算法，是由陈天奇博士于 2015 年提出的一种在 Gradient Boosting 框架下实现提升决策树(GBDT)和广义的线性机器学习算法[9]。该算法是在梯度提升(Gradient Boosting)的框架下进行的改进和优化，也是串行生成的集成学习中的一类。该算法的思想是加法模型和前向分布算法，通过特征分裂生长树添加在模型中，每次生成新树的本质是学习一个新函数，用来拟合上棵树预测的残差。这种加法模型以上一次预测(前面 $t-1$ 棵树的组合模型)的误差为参考，建立下一棵树(第 t 棵树)，直至训练完成得到 k 棵树。最终要实现对新样本的预测，新样本在每棵树中有一个对应的叶子节点，该节点又会产生相应的得分值，最终将每棵树相应的叶子结点得分相加，其结果作为新样本的模型预测值。相比于 GBDT，XGBoost 优点在于：目标标损失函数增加正则项，可以约束损失函数的下降，同时可以避免模型过于复杂，防止模型过拟合；将损失函数通过泰勒公式展开到二阶导，展开到二阶导能够提高模型的预测精度，XGBoost 算法是 GBDT 算法的高效实现，整体的优点就是提高了运行速度；XGBoost 算法中的基学习器支持线性分类器，GBDT 只支持 CART 树[10]。

LigthGBM 是轻量级梯度提升机(Light Gradient Boosting Machine, Light GBM)，它和 XGBoost 都是对 GBDT 的高效实现，原理上它和 GBDT 及 XGBoost 类似都采用损失函数的负梯度作为当前决策树的残差近似值，去拟合新的决策树。该算法使用深度限制的叶子生长(leaf-wise)策略，从当前叶子节点中找到增益值最大的节点进行分裂，并对树的深度进行限制，防止过拟合，缩短寻找最优深度树的时间。同时保证分裂次数相同的情况下，能够降低误差，得到更高精度。在构建树的过程中，最浪费时间和计算机资源的是寻找最优分裂节点的过程，对此，LightGBM 使用直方图算法、单边梯度抽样算法(Gradient Based One-side Sampling, GOSS)和互斥特征捆绑算法(Exclusive Feature Building, EFB)来提升运行效率[11]。其具有训练速度快、消耗内存少、预测精度高的特点，在训练速度和准确率之间达到了平衡。

3. 票房影响因素分析

论文主要通过 Python 软件和数据库对豆瓣电影、时光网、灯塔专业版和百度指数等网站的相关数据进行抓取与整理，抓取了 2017 年至 2019 年票房排名前 100 名的电影，共获得 300 部万元电影票房和相关影响因素数据。每年票房排名的前 100 名电影的累计票房收入总和均超过当年年度总票房收入的 90%，具有足够的代表性。电影特征方面主要有：类型、时长、上映日期、导演和主演的喜爱度、票房和电影数量、网络搜索量等；电影评论方面主要有：短评文本、评分、评分人数、想看人数、五星比率等。其中，总票房、首周票房来自时光网；每部电影的评分、类型、时长、导演、主演、想看人数等数据来自

于豆瓣；百度指数提供了电影上映期间的检索量数据等，其他数据均来自于灯塔专业版。

由于所爬取的数据来源于多个网站，在各网站数据合并后，数据存在缺失、异常、变量重复等噪声。论文的数据清洗主要是剔除与主题无关和重复的变量、删除重复评论数据、平滑噪声数据，并对缺失值、异常值等进行处理。具体清洗过程为：通过 pandas 的 merge 整合数据，检查变量是否有缺失值，确定缺失值的比例和范围，删除多次重复爬取的列变量和缺失较多的变量，补全部分必需的缺失值；对格式内容进行清洗：将单位亿和万统一化，将时间文本变量格式化，替换删除特殊符号等不需要的字符和文本；去除修改逻辑错误的数据：去重，针对出现异常值分析其原因，其中使用正则表达式、replace、if 和 for 进行匹配，进而提高清洗的效果。

论文所选影响因素中电影类型、上映时间、导演及演员等变量为非数值型数据，需要量化、规约、加权处理，对相似的影响因素进行整合，提高电影票房预测的准确率。

3.1. 影响因素量化

1) 电影档期

论文在对前人对电影档期的研究做了充分总结的基础之上，最终将电影档期划分为种贺岁档(11 月 20 日至次年 2 月底)、五一档(4 月 28 日至 5 月 3 日)、暑期档(6 月 1 日至 8 月 31 日)、国庆档(9 月 28 日到 10 月 7 日)和其他档期共五个档期，转化为分类型变量。

2) 导演

为量化导演的影响力，论文通过该导演主导的所有电影的平均票房量和观众的喜爱度来衡量。具体公式为

$$Dir_i = \alpha * \overline{Box_i} + \beta * \overline{NSV_i}$$

其中，Dir_i 表示第 i 个导演的影响力，α 和 β 表示平均票房以及观众喜爱度的重要性系数，$\overline{Box_i}$ 表示其作为导演主导电影的平均票房，$\overline{NSV_i}$ 表示观众对该导演的喜爱度。

3) 主演

为了量化演员的影响力和号召力，论文侧重于演员的流量程度，故选用百度指数中的电影上映一个月内演员的平均流量指数。该指数是以百度智能分发和推荐内容数据为基础，将网民的阅读、评论、转发、点赞、不喜欢等行为的数量加权求和得出资讯指数。

4) 网络搜索量

电影在上映期间搜索量从侧面反映了潜在观众对电影的关注。尽管不同的潜在观众在搜索后会做出不同的观影决策，但是从另一个层面来说，更多的人关注，就表明可能有更多的潜在观众会选择观看电影。论文根据图 1 电影上映前后的搜索热度，选取电影上映一周内百度指数的搜索指数的平均值，该指数是以网民在百度的搜索量为数据基础，以关键词为统计对象，分析并计算出各个关键词在百度网页搜索中搜索频次的加权和。

5) 影评情感特征

论文将电影短语评论文本为两类：积极特征、消极特征，并通过情感分析模型计算得每条评论的正面评论得分和负面评论得分。再将各条评论的情感得分乘以该评论所对应的其他用户的点赞数，则可表示该条评论的综合情感得分。再将该电影的每条评论的综合情感得分求和，得出该电影的综合情感得分。

此处创新性地将点赞数和影评的情感倾向结合在一起。这是由于用户点赞行为说明了用户之间的情感和观点认同，一定程度上可以认为其他点赞用户与被点赞的评论所持的观点一致。将点赞数和情感倾向得分融合，更能体现一部电影所有评论用户的整体感情倾向，这提高数据的代表性。

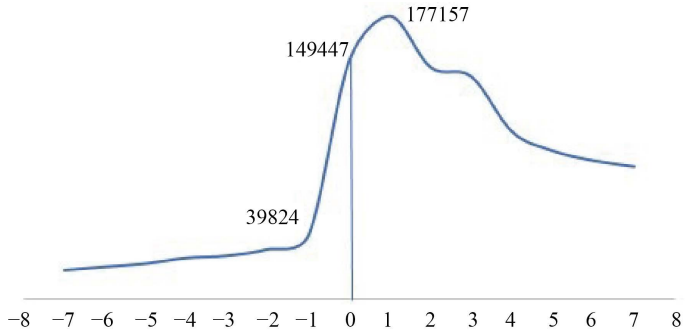

Figure 1. Search volume of Baidu Index before and after movie release
图 1. 电影上映前后百度指数的搜索量

综上，参考各文献已有票房预测模型中的影响因素和国内外电影市场的情况，综合分析论文所选取的影响电影票房的影响因素，将电影类型、电影档期、导演、主演、时长、情感分析值、网络搜索量、评分、评分人数、想看人数作为论文电影票房的主要影响因素。

3.2. 影响因素归一化

论文选取的变量具有不同特征，其量纲不同造成数据间差别较大可能会对票房预测造成影响，则需对数据进行归一化处理。电影票房影响因素数据具有类别和数值型的特性。当特征的类别向量化后的值和数值值组合在一起时，会使特征值的范围不同。为了解决这个问题，论文对于类别变量如电影类型为非有序类别的变量，对其清洗归类后进行 One-hot 编码，将分类变量转换成新的变量。对于数值型变量如电影首周票房数据值较大，采用归一化的方法处理定量数据，对其进行按照一定比例缩放，使之落在一个特定区域，便于综合分析和模型构建。

论文选用常用的归一化和标准化方法为：线性归一化是指对原数据进行线性变化，将数据映射到[0,1]区间内，其具体原理为：$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$。标准化是指将原始数据映射到均值为 0、标准差为 1 的分布上，是针对于每个特征维度来标准化的，其原理为：$x' = \frac{x - \mu}{\sigma}$。

图 2 分别对数值型影响因素电影时长和首周票房进行标准化和归一化的处理后的标准差，对比可知归一化比标准化的标准差小，则论文选用归一化方法来缩放数据，使数据更集中在均值附近。经过以上归一化，可以使得不同维度之间的数值具有一定的可比性，同时加快了梯度下降求最优解的速度加速收敛，在一定程度上提高模型的准确率。

3.3. 电影票房影响因素分析

对论文抓取的 300 部电影的基本信息进行数据分析，从整体上研究 2017 年至 2019 年排名靠前的电影中各影响因素与电影票房的关系和对电影票房的贡献力度，从而确定主要影响因素。

3.3.1. 数值型因素对票房的影响

论文的数值型影响因素主要包括首映周票房、导演、主演、时长、影评情感特征、网络搜索量、评分、评分人数八个影响因素，部分属于分类型数据如导演、演员、情感特征已将其量化为新的特征，为分析其与票房的关系，做散点图如图 3。

由图 3 可知，电影时长、电影首周票房、电影评分、电影评分人数、影评情感特征与票房存在明显的线性关系，网络搜索量与票房存在指数关系，可知所选取的变量都对电影票房的有一定的影响。

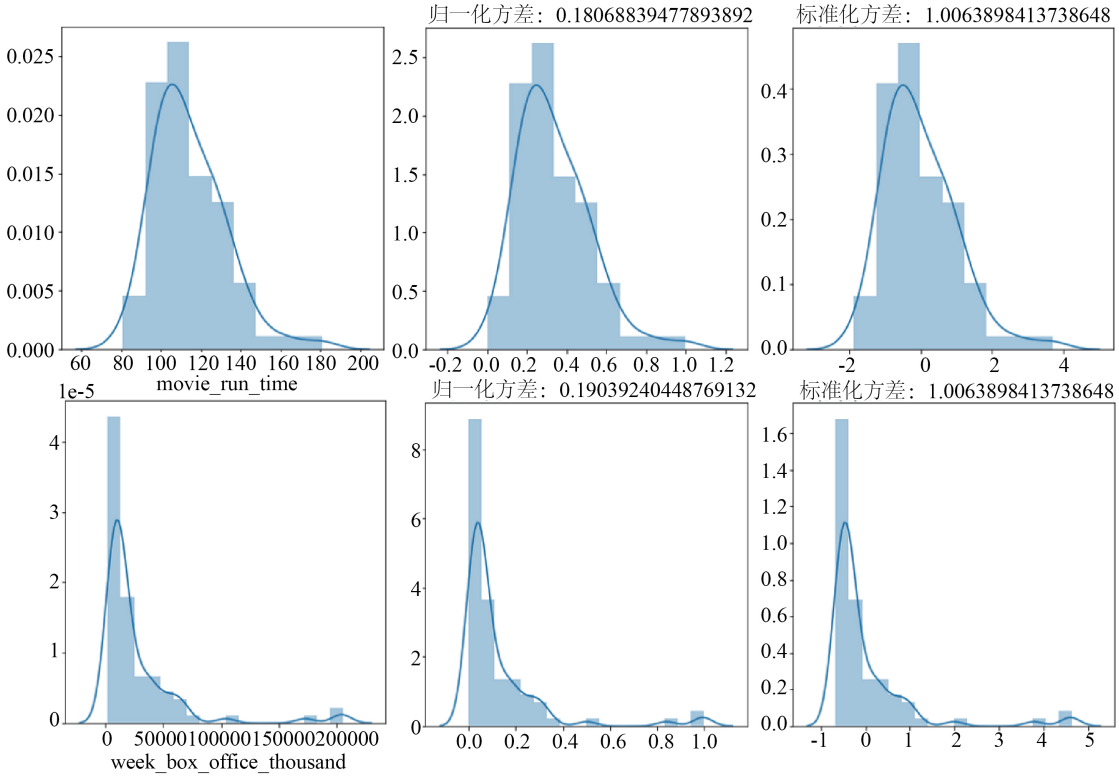

Figure 2. Standard deviation after normalization and standardization
图 2. 归一化和标准化后的标准差

对数值型影响因素进行相关性分析，具体如表 1 可知，电影首周票房、情感特征、网络搜索量、导演、电影评分数量五个特征相关系数较大，与票房存在显著的相关性，对票房的影响较大，在预测电影票房时将其作为主要影响因素，电影时长、评分与票房相关性较弱，演员因素与票房相关性最低，存在负相关，对票房的影响较小。

Table 1. Correlation coefficients among influencing factors
表 1. 各影响因素间的相关系数

	电影票房	首周票房	情感特征	网络搜索量	导演	评分数量	评分	时长	演员
电影票房	1	0.96	0.75	0.7	0.68	0.68	0.13	0.06	−0.08
首周票房	0.96	1	0.63	0.6	0.58	0.6	0.08	−0.02	−0.07
情感特征	0.75	0.63	1	0.83	0.63	0.85	0.45	0.15	0
网络搜索量	0.7	0.6	0.83	1	0.48	0.96	0.56	0.29	0.05
导演	0.68	0.58	0.63	0.48	1	0.43	0.19	0.08	−0.3
评分数量	0.68	0.6	0.85	0.96	0.43	1	0.65	0.28	0.09
评分	0.13	0.08	0.45	0.56	0.19	0.65	1	0.4	0.05
时长	0.06	−0.02	0.15	0.29	0.08	0.28	0.4	1	0.43
演员	−0.08	−0.07	0	0.05	−0.3	0.09	0.05	0.43	1

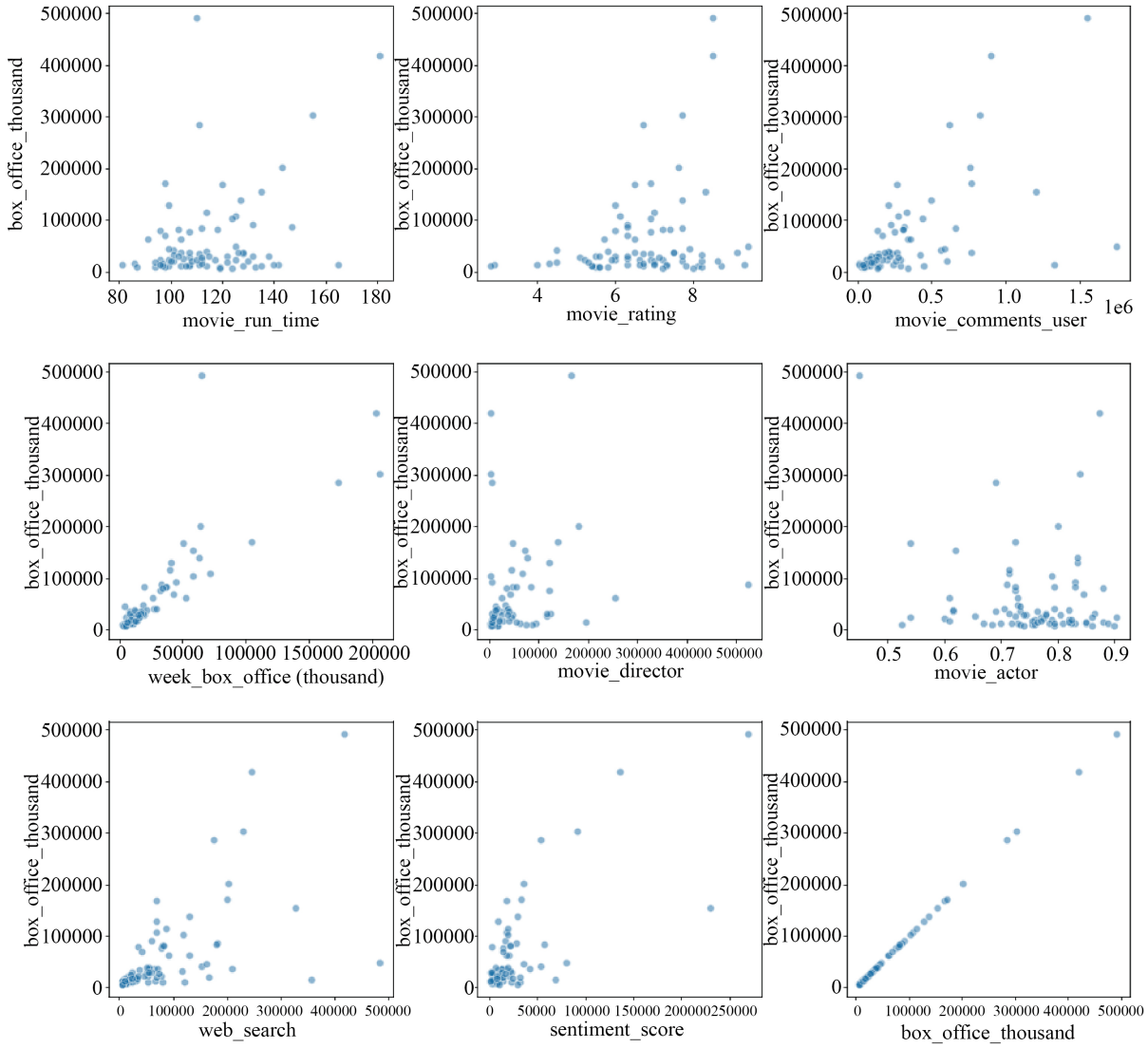

Figure 3. Scatter plot of influencing factors
图 3. 影响因素散点图

具体来说，首周票房对最终的票房有很大影响，两者存在明显的线性关系。这表明首周票房一定程度上可以反映电影票房的总体趋势。在电影的早期阶段，人们会由于电影宣传，评论和社交网络传播等因素而选择观看电影。随着同档期电影数量的持续增加，若没有在短期内吸引观众，将影响电影票房的后续表现。

影评情感特征对票房的影响也有较大影响，论文对情感得分与各条评论点赞数汇总得到各部电影的情感特征值，在控制其他影响因素的情况下，重点研究情感特征对票房的影响得其偏相关系数为 0.637，且 P 值为 0 通过检验，说明影评的情感得分与票房有较大的关系。

3.3.2. 分类型因素对票房的影响

论文的分类型因素主要为：第一类型、第二类型、国家、上映档期，结合数据和相关文献了解到电影的类型和电影的上映档期对电影票房影响较大。则重点分析电影类型和档期对票房的影响。

1) 电影类型对票房的影响

结合原数据的电影类型，按照剧情、动作、喜剧、悬疑、爱情、科幻等关键词对电影类别进行分类，分析电影的第一类型和第二类型中不同电影类型票房的占比。根据所电影分类绘制饼图如图 4 所示，可知剧情、动作、喜剧、科幻四个类型的电影票房的贡献最大，其占比分别为 29.6%、16.6%、14.6%和 10.9%。根据该占比进行数据清洗，将电影类型整合为剧情、动作、喜剧、悬疑、爱情、科幻六种类型，便于引入模型成为新的变量，降低维度。同时该占比可以为电影拍摄选择类型时提供借鉴。

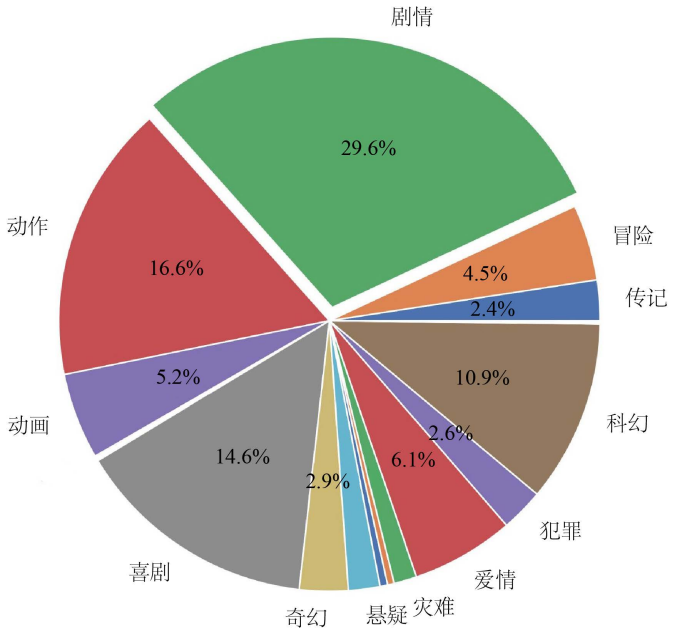

Figure 4. Percentage of box office of different movie genres
图 4. 不同电影类型票房所占百分比

对各类电影进行方差分析，判断各类不同电影类型的电影票房是否有显著差异，方差分析表结果如表 2 所示，可知 $F = 2.3171 > F_{\alpha} = 2.2447$，在显著性水平 0.05 条件下，认为各类电影票房具有显著性差异，电影类型对电影票房有一定影响。

Table 2. Movie box office variance analysis table by type
表 2. 电影分类型票房方差分析表

差异	平方和 SS	自由度 df	均方 MS	F 值	P 值	F 临界值
组间	4.9E+10	5	9.8E+09	2.3171	0.0002	2.2447
组内	1.243E+12	294	4.229E+09	_	_	_
总和	1.292E+12	299	_	_	_	_

2) 档期对票房的影响

论文将电影档期分为贺岁档、五一档、暑期档、国庆档和其他档期，分析不同档期的票房，做箱线图如图 5。可知不同档期的电影票房明显不同，说明档期对电影票房有很大影响。其中国庆档均值、最大值相对较高，说明国庆档电影票房最好，春节档和暑期档也存在较高票房的情况，而五一档和其他档期电影票房较低，其可以为电影上映提供参考。

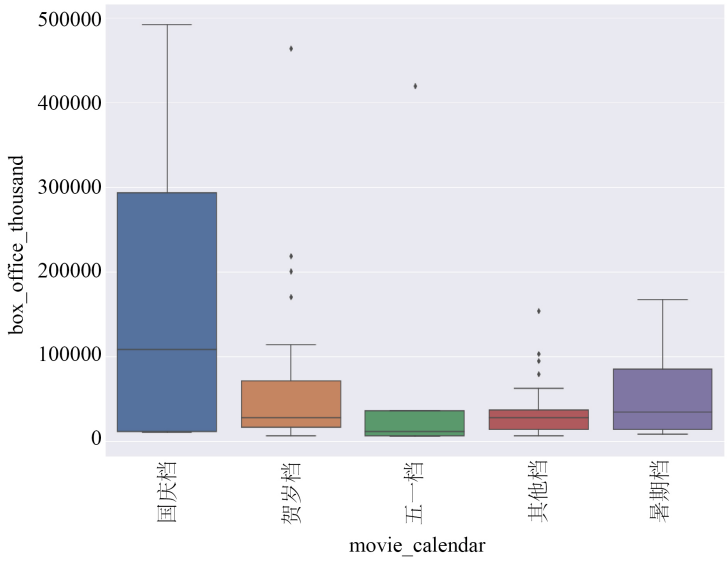

Figure 5. Box plot: Movie box office of different schedules
图 5. 不同档期电影票房箱线图

4. 电影票房预测评估

在量化电影票房影响因素的基础上将 2017~2019 年 300 部电影数据输入模型构建电影票房预测模型进行拟合预测，分别采用 Lasso 、ElasticNet、Ridge、GBDT 算法、LightGBM 算法、XGBoost 算法六个常用的机器学习模型进行票房预测。研究中，选用的集成学习算法是基于树模型在训练样本有限、训练时间短、调参具有独特优势。在代价函数中加入了正则项，可以控制模型的复杂度；支持并行处理和支持用户自定义目标函数和评估函数，灵活性高；内置交叉验证，可以很方便的获取最优迭代次数。相比于近年流行的神经网络能够更灵活处理表格数据，并具有更强的可解释性，适合具有多类型特征的电影影响因素数据。

论文构建票房预测模型的过程大致分为：票房数据分布分析并正态化；通过 XGBoost 模型的重要性特征图筛选主要影响因素；通过 MinMaxScaler 函数将影响因素归一化；利用训练数据通过网格搜索和交叉验证对模型进行调参优化，利用调参后的模型进行训练和对比评估；通过 Stacking 方法融合效果较好的模型并评估；对比影评情感特征引入模型前后的拟合效果；对比不同基学习器 Stacking 模型的拟合效果；最后通过拟合较好的模型对电影票房进行预测得到最终的预测结果。

4.1. 票房数据正态化

对票房数据做拟合得图 6 左图，可知数据呈右偏分布，为保证均值和方差相互独立，且残差服从正态分布，便于更好地拟合，论文通过对票房数据取对数，将其正态化如图 6 右所示。

4.2. 筛选主要影响因素

通过 XGBoost 模型筛选主要影响因素，作出重要性特征图如图 7，其是用来度量自变量在 XGBoost 模型中对因变量预测的贡献度，是以某特征作为分裂节点带来的平均增益，从而有效地进行特征的筛选。其作为衡量变量重要性的指标，值越高说明该变量对因变量越重要，影响越大。

根据图 7 可知，在所有的影响因素中，电影首周票房和网络搜索量的平均增益最大，说明在影响票房所有因素中发挥的作用最大，其次是评论人数、情感特征、电影类型等影响因素，另外通过对图 7 中

变量平均增益的观察可以看出，有部分影响因素的重要性分数很小几乎接近于零，表明这些因素在对票房的影响方面发挥的作用很小，相对于其他的重要性分数较大的因素其作用几乎可以忽略，这些因素包括电影演员及第二类型，因此为了简化后续的票房预测模型输入，论文在进行票房影响因素的选择时只选取影响力较大的因素，去掉一些作用很小的影响因素，从而在输入层对预测模型进行简化。因此，筛选后的票房影响因素共包含电影首周票房、网络搜索量、评论人数、情感特征、电影类型、评分、导演和档期等因素。

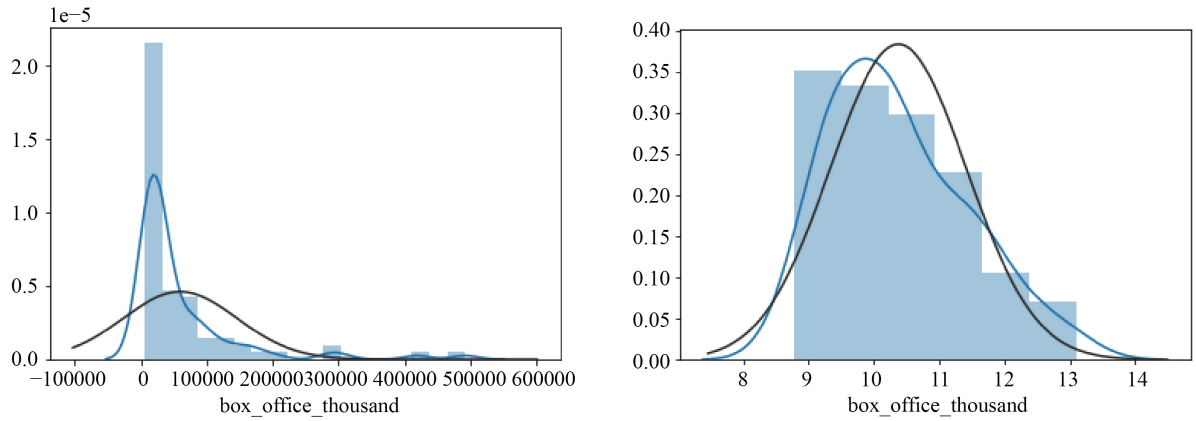

Figure 6. Before and after movie box office take logarithm
图 6. 电影票房取对数前后对比

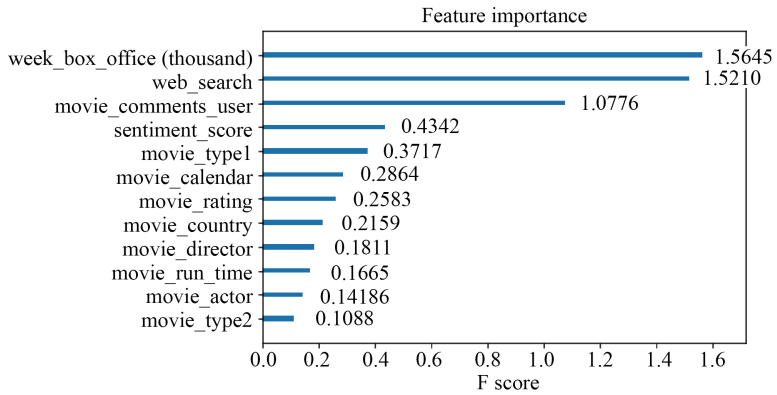

Figure 7. XGBoost importance feature map
图 7. XGBoost 重要性特征图

4.3. 构建票房预测模型

XGBoost 算法汇总多个弱评估器，该算法模型中有多种超参数，为找到最优的参数组合,论文选用 sklearn 中的 GridSearchCV 通过网格搜索交叉验证调整模型参数。网格搜索是指定参数的一种穷举搜索方法，根据经验将可能对模型造成影响的参数值设定区间范围，并对其进行排列组合，将所有可能的组合结果列出用于 XGBoost 训练，然后通过交叉验证对模型进行评估。通过评分函数对每个超参数值进行打分并选择得分最高的参数值，最后得到模型的最优参数组合。

对 XGBoost 模型调试参数的过程为：分别对 XGBoost 模型、树、学习目标三个层面的参数进行交叉验证，选用模型均方差最小状态下的参数。对于 XGBoost 模型调参的具体为：调整树的最大深度，最小

叶子的权重通过迭代的方式不断修正，缩小范围；调整样本随机采样率和特征采样率，确定最佳采样比例；根据已有经验在设定小范围内调大节点分裂阈值、调小学习率来测试，防止过拟合。对票房影响因素的研究和多次调参优化后，最终对于 XGBoost 模型调试的参数如表 3。

Table 3. XGBoost model parameter configuration
表 3. XGBoost 模型参数配置

参数	含义	取值
booster	学习器	Getree
objective	目标函数	Reg:linear
eta	学习速率	0.3
subsample	随机采样率	0.7
Max_depth	树的最大深度	3
Colsample bytree	特征采样率	0.7
Min_child_weight	最小叶子的权重	3
gamma	节点分裂阈值	0.1
n_estimators	生成树的个数	500

由于数据量较少，则选用十折交叉验证来优化各个模型的参数，通过训练数据构建单个票房预测模型，利用均方根误差和标准差来进行模型评估。均方根误差评判机器学习模型的准确度，RMSE 越小，模型越准确，它度量了算法的预测值同真实值之间的偏离程度，刻画了学习算法本身的拟合能力。均方根误差的公式见式(1)。标准差评判的是机器学习模型的稳定性，方差越小，模型越稳定。它度量了训练集变动所导致的学习性能变化，刻画了数据扰动所造成的影响。

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{N}} \tag{1}$$

调试后的模型均方根误差和标准差结果见表 4，可知调整模型参数后，GBDT 模型均方根误差和标准差最小，说明该模型拟合效果较好，在采用单个模型时优先考虑用此模型进行电影票房的预测；同时 XGBoost、LigthGBM 模型均方根误差和标准差相对其他模型也较小，说明两个模型准确率和预测精度较高，在模型融合时可以优先考虑采用该两模型。

Table 4. Model evaluation before and after the introduction of emotional features
表 4. 引入情感特征前后的模型评估

模型	RMSE	标准差
Lasso	0.146346	0.028864
ElasticNet	0.143952	0.027534
Ridge	0.143934	0.025430
GBDT	0.087363	0.011990
LightGBM	0.100069	0.027869
XGBoost	0.100995	0.018498

综上，使用单个模型进行电影票房预测，容易陷入过拟合，降低了模型预测的准确度。而集成学习包含了多种不同的学习器，由于每个学习器的决策边界与误差不相同，在一定策略下将其进行组合后可以有效降低总误差，类似对单个学习器的纠错行为，能够较好的缓解过拟合的问题，增强模型泛化能力。则论文将使用强学习器进行组合的 Stacking 集成学习方法，对模型进行融合从而获得准确的预测。

4.4. Stacking 模型融合

模型融合是通过组合一些比较简单的算法，以保留这些算法方差低的优势；在此基础之上，它又能引入复杂的模型，来扩展简单算法的预测空间，集成学习模型融合能够减少标准差和偏差。论文对电影票房数据进行分割，将 70%的数据作为训练集，30%作为测试集。在训练集上训练各个模型，得 GBDT、XGBoost、LigthGBM 拟合效果较好，通过 Stacking 方法对这三个模型进行融合。

Stacking 模型的主要思想是训练模型来学习使用底层学习的预测结果。论文使用两层融合模型。第一层由多个基学习器组成并进行特征抽取，其输入为原始训练集；模型的第二层使用第一层基学习器的输出作为训练集进行再训练，从而获得完整的 Stacking 模型。论文通过 mlxtend 库提供的 StackingCVRegressorstacking 模型融合了拟合较好的三个模型，每个模型内部进行五折交叉验证。模型融合的具体过程如下：

1) 拆解训练集。将训练数据随机且大致均匀的拆为 5 份，每次选择其中 4 份作为训练集，1 份作为验证集。

2) 针对三个基模型，分别进行 5 次训练，同时在验证集和测试集上进行预测。每次训练保留一份样本用作训练时的验证，如在模型 1 中利用 4 份训练数据进行训练，对 1 份验证集进行预测得 42 条数据，同时在真正的测试集上预测得 90 条数据；如此重复 5 次，将训练集上 5 次预测结果叠加为 1 列 trian_meta1 (210 条)，将测试集上 5 次结果取均值融合为 1 列 test_meta1 (90 条)；此步骤结束后，将三个基模型的验证集预测结果并列得到 train_meta (210 * 3)作为下一层的训练数据，将三个基模型的测试集预测结果并列得到 test_meta (90 * 3)作为下一层的测试数据。

3) 第二层选用 lasso 回归模型，去学习新的训练集 train_meta，然后对新测试集 test_meta 进行预测，得到模型融合之后的预测结果。

Table 5. Root mean square error of the fusion model
表 5. 融合模型的均方根误差

模型	RMSE
GBDT	0.028831
LightGBM	0.052431
XGBoost	0.000861
Stacking_model	0.037613

通过构建融合模型，对比各模型的拟合效果，训练模型后计算得到的 RMSE 结果如表 5。可知，XGBoost 的 RMSE 值最小，其预测效果最好；GBDT、LigthGBM 和融合模型的 RMSE 值也较小，预测准确率也较高。各算法单独进行预测时，XGBoost 的预测误差较小，这是因为 XGBoost 对损失函数进行了二阶泰勒展开，优化过程使用了一阶和二阶导数信息进行更新迭代，使得模型训练更充分。但若单独使用 XGBoost 模型容易过拟合，使其在训练集上表现好，但对预测值得预测能力差，而此处的融合模型精度相对 XGBoost 和 GBDT 较低，但泛化能力强。

Table 6. Movie box office prediction results of Stacking model and XGBoost model
表 6. Stacking 模型和 XGBoost 模型的电影票房预测结果

电影	票房(万)	Stacking_model	XGBoost	Stacking_model 预测误差	XGBoost 预测误差
大侦探皮卡丘	285200	303465.2	418421.6	6.40%	46.71%
烈火英雄	170900	173821.5	191094.4	1.71%	11.82%
哥斯拉 2 怪兽之王	138000	138953.5	123081.1	0.69%	10.81%
海王	114500	107460.2	90528.57	6.15%	20.94%
我和我的祖国	108100	117871.7	119299.6	9.04%	10.36%

为验证 Stacking 集成学习模型的预测性能，通过对比 Stacking 模型和 XGBoost 模型的电影票房预测结果见表 6。其中预测误差是通过平均绝对百分比误差衡量，其指标越小，表明模型的预测精度越高。其公式为

$$MAPE = \frac{100\%}{N}\sum_{i=1}^{N}\left|\frac{\hat{y}_i - y_i}{y_i}\right|$$

由表 6 可知，Stacking 集成模型的预测误差较小，采用 Stacking 集成模型的预测效果优于 XGBoost 模型预测效果。同时计算得 Stacking 模型和 XGBoost 模型的总预测误差分别为 7.24%和 13.53%，说明 Stacking 模型的预测效果较好，在预测时泛化能力更强，更适合用来进行票房的预测。

这是由于 Stacking 集成学习的方式使用第二层的学习器对所有基学习器产生的预测结果进行泛化，有效减少单一模型泛化性能不佳的风险，以此来提高模型的整体预测精度。同时对于模型优化方面，单一模型优化时，模型经常会陷入局部最小值，一些局部极小值所对应的模型泛化性能较差，而将多个基学习器的融合可有效降低陷入局部极小值的风险。因此，采用 Stacking 集成学习方式后电影票房的预测精度有所提升。

为了进一步验证 Stacking 集成模型中基学习器选择对预测结果的影响，表 7 分别给出了不同基学习器组合方式的预测结果。其中，Stacking 模型 1 融合了六种不同的回归模型，Stacking 模型 2 选用精度最小的基学习器组合方式(XGBoost、GBDT、LightGBM)。预测结果显示，使用不同的基学习器对预测结果有较大影响，使用拟合较好的基学习器的 Stacking 模型比结合全部回归模型学习器的 Stacking 模型表现更优异。一方面是由于选择学习能力强的基学习器能够整体提升 Stacking 模型的预测能力。另一方面是

Table 7. Stacking algorithm box office prediction error of different base model combinations
表 7. 不同基模型组合方式的 Stacking 算法票房预测误差

模型	Stacking_model1 RMSE	Stacking_model2 RMSE
Lasso	0.119191	_
ElasticNet	0.117959	_
Ridge	0.102024	_
GBDT	0.028831	0.028831
LightGBM	0.052431	0.052431
XGBoost	0.000861	0.000861
Stacking_model	0.042411	0.037613

因为针对不同的机器学习算法，其本质是在不同的数据空间观测数据，再根据自身算法规则构建相应模型。而选择差异度较大的算法能够最大程度体现不同算法的优势，基模型的多样性和差异性使得集成结果会更加稳健、精确，从而使得预测效果获得更大的提升。因此论文选择预测效果较好的三个模型(XGBoost、GBDT、LightGBM)通过 Stacking 方法进行融合，对测试集进行预测，测试结果如图 8。

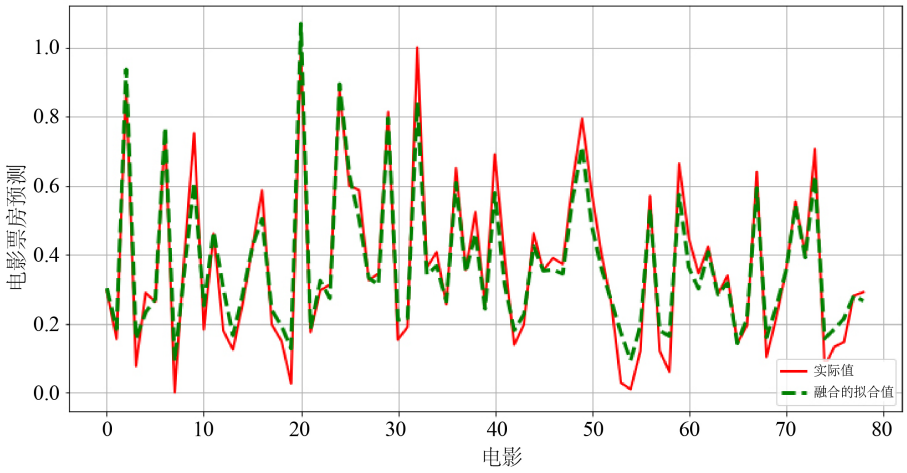

Figure 8. Box office prediction based on Stacking integrated learning
图 8. 基于 Stacking 集成学习的票房预测

5. 结论

论文在已有研究的基础上，提出了一种基于 Stacking 集成学习的电影票房融合模型的预测方法，通过网格交叉验证优化调整模型参数，并利用 XGBoost 算法获取样本特征作为分裂节点带来的平均增益，筛选主要影响因素为电影首周票房、网络搜索量、评论人数、情感特征、电影类型、评分、导演和档期等因素。对比评估并融合拟合效果较好的模型，所构造的包含 GBDT、XGBoost、LigthGBM 的 Stacking 集成学习模型较单个机器学习预测模型泛化能力更强，具有更好的预测效果，在电影票房预测方面有较高的应用价值。在今后的研究中，可以引入更多的电影相关特征和机器学习模型对票房预测模型不断改进与完善。

参考文献

[1] Sharda, R. and Delen, D. (2006) Predicting Box-Office Success of Motion Pictures with Neural Networks. *Expert Systems with Applications*, **30**, 243-254. https://doi.org/10.1016/j.eswa.2005.07.018

[2] Lee, K.J. and Chang, W. (2009) Bayesian Belief Network for Box-office Performance: A Case Study on Korean Movies. *Expert Systems with Applications*, **36**, 280-291. https://doi.org/10.1016/j.eswa.2007.09.042

[3] Barman, D., Chowdhury, N. and Singha, R.K. (2012) To Predict Possible Profit/Loss of a Movie to Be Launched Using MLP with Back-Propagation Learning. 2012 *International Conference on Communications, Devices and Intelligent Systems* (*CODIS*), 322-325. https://doi.org/10.1109/CODIS.2012.6422203

[4] Kim, T., Hong, J. and Kang, P. (2015) Box-Office Forecasting Using Machine Learning Algorithms Based on SNS Data. *International Journal of Forecasting*, **31**, 364-390. https://doi.org/10.1016/j.ijforecast.2014.05.006

[5] 郑坚, 周尚波. 基于神经网络的电影票房预测建模[J]. 计算机应用, 2014, 34(3): 742-748.

[6] 罗晓芃, 齐佳音, 田春华. 电影首映日后票房预测模型研究[J]. 统计与信息论坛, 2016, 31(11): 94-102.

[7] 米传民, 鲁月, 林清同. 基于加权 K-Means 和局部 BPNN 的票房预测模型[J]. 计算机系统应用, 2019, 28(2): 15-23.

[8] 杨朝强, 蒋卫丽, 邵党国. 基于 LSTM 模型的电影票房预测算法[J]. 数据通信, 2019, 32(5): 34-37.

[9] Chen, T. and He, T. (2015) Higgs Boson Discovery with Boosted Trees. *Proceedings of the* 2014 *International Confe-*

rence on High-Energy Physics and Machine Learning, Montreal, 69-80.

[10] 李劲彬, 夏天, 黄烁, 等. 基于 XGBoost 的集成式隔离断路器状态评估[J]. 高电压技术, 2020, 46(5): 1800-1806.

[11] Bian, L.Y., Zhang, L.L., Zhao, K., *et al*. (2020) Ethereum Malicious Account Detection Method Based on LightGBM. *Netinfo Security*, **20**, 73.