基于 XGBoost 的事故预警方法

石雪怀[1]，**戚湧**[2]，**李千目**[3]

(南京理工大学计算机科学与工程学院 江苏南京 210094)

摘 要 （城市火灾作为社会公共安全的重要威胁源，极易造成巨大经济损失，甚至导致人员伤亡。城市火灾事故是随机事件，如何从火灾偶然性的表象出发分析其内部规律是该领域亟待突破的重要问题。本文提出了一种新的基于 XGBoost 的事故预警方法，该方法根据不同类型建筑的火灾特点，基于关联规则进行特征选择，融合定量定性的方法确定此火灾事故预警方法的预测指标，利用 Box-Cox 非线性变换方法对特征的连续响应变量进行处理，降低不可观测误差与预测变量间的相关性，通过组合/集成方式去除数据的不平衡性，最后通过 XGBoost 增强算法的训练，获取最优预测精度。实验表明本方法对火灾事故预警提供了一种较为可行的解决方案，有助于改善社会公共安全状况。）

关键词 XGBoost 算法，Box-Cox 非线性变换，事故预警

中图分类号：TP391

An Accident Warning Approach Based on XGBoost

XuehuaiShi[1] Yong Qi[2] Qianmu Li[3]

(School of Computer Science and Engineering, Nanjing University of Science and Technology, 210094, China)

Abstract (As an important threat to public security, urban fire accident causes huge economic loss and catastrophic collapse. Predicting and analyzing the interior rule of urban fire accident from its appearance needed to be solved in the field.In this paper, we propose a new urban fire accident warning approach based on XGBoost. The method determines the predictive indexes in a quantitative and qualitative way from different characteristics in various kinds of fire accidents. For screening the features we need, we adopt the feature selection algorithm based on association rules. For data cleaning, we use a method based on Box-Cox transformation that transforms the continual response variables from the feature space for removing the dependencies on unobservable errors and the predictor variable to some extent. Then we use the data to train the model based on XGBoost to obtain the best prediction accuracy. Experiments show that the method provides a feasible solution to urban fire accident warning. And the method contributes to improving the public security situation.)

Keywords XGBoost Algorithm, Box-Cox Transformation, Accident Warning

1 引言

2017 年 7 月 10 日英国伦敦西部建富大厦发生大火，摧毁了肯顿水门市场，超过 80 人死亡；同一天，一场大火席卷了美国西部和加拿大，摧毁了几千名居民的家园，造成过千居民流离失所，损失惨重。长期以来，住宅区、工厂、仓库、大型商场等各类城市建筑体时常发生大、特大火灾，这使得人们对于消防安全工作的关注度持续提升。

城市火灾事故预警是为了掌握未来火灾事故的状况，通过对区域火灾事故的历史数据和当前状态数据进行分析，利用现有数据特征，分析火灾相关隐私，对火灾事故未来信息进行预测，分析火灾事故预警结果，定制个性化决策从而极大限度避免火灾发生，减少损失所做出的阐述。

国内外学者基于不同的目标提出了不同的火灾预警系统。如在[1-3]中，作者从宏观角度，根据历史火灾数据，对本年度及下年度火灾进行预测；在[4]中，作者基于提供人工神经网络接口的硬件模型建立火灾检测和控制机制，此硬件模型包含温度感应器、烟雾感应器、火花感应器及微控制器单

到稿日期：2017-12-27 返修日期：2017-00-00 本文受国家重点研发计划政府间国际科技创新合作重点专项(2016YFE0108000)资助。

石雪怀（1995-），男，硕士研究生，主要研究方向：数据挖掘；戚湧（1970-），男，博士，教授，博士生导师，通讯作者，CCF 高级会员（E200022866S），主要研究方向：数据挖掘，E-mail:790815561@qq.com；李千目（1979-），男，博士，教授，博士生导师，主要研究方向：数据挖掘。

元，但没有考虑人为因素及建筑物老化等其他因素对火灾致灾的影响；在[6]中，作者依据天气参数及通过视频监控设备探测烟雾级别作为火灾事故预警信号来建立火灾事故预警模型，此模型评估隐患火灾，提供火灾危险指数，此模型适用于植物群落地区，通过探测烟雾级别作为火灾事故预警信号，适用场景较为局限，此外[7]和[8]致力于森林火灾预测领域，也做出了显著贡献；[9]基于城市GIS数据获取地形、人口密度、水道、道路、不合标准的住房和降雨信息使用多准则决策方法建立城市火灾预警系统。由于探测器的缺失，现阶段没有依据建筑物自身特点和建筑物内部动态数据进行相关性分析建立模型的思路，现有火灾事故预警几乎都是宏观的事故率预测，缺少详细可操作的微观事故风险预测。

由城市住宅火灾预警系统的研究现状可知，现阶段城市住宅火灾预警系统的建立存在以下几个难点：1.相关智能设备的缺失使得火灾预警信息数据获取难度较大；2.相对正常状况，火灾作为一种异常突发事件其相关数据相对正常状况具有不平衡性，如何对非平衡数据处理，是建立可靠的火灾预警系统必须解决的难题；3.以往建立消防预警系统，在指标体系的确定和风险大小计算方法消耗大量人物力，如何确定城市住宅火灾预警的指标体系和风险计算方法是城市火灾事故预警的主要难点。

本文提出一种基于区域的历史火灾大数据，根据建筑物使用性质进行分类，将建筑物分为厂房、仓库、公共、住宅，结合建筑物其他特征，结合指定日期气候状况，针对此建筑物历史消防安全评估指标添加火灾预警结果标签，针对火灾信息数据的不平衡性，采用基于关联规则的特征选择算法进行特征选择，使用Box-Cox非线性变换算法去除样本一些特征的噪声数据，通过组合/集成方式去除数据的不平衡性，最终使用XGBoost算法针对建筑物建立评估体系，增加预警结果的准确性与科学性。

2 基于XGBoost的事故预警方法

XGBoost继承于GBDT并对其进行改良优化的集成算法，属于Boosting算法。由于在火灾事故预警大数据中，数据样本较大，采用神经网络或其他决策树增强模型消耗时间较多。XGBoost针对处理海量数据，提供缓存感知预读取技术、分布式外存计算技术、AllReduce容错工具提高现有提升树增强算法运算速率，提升算法运行速率。本方法完整技术流程图如图1所示。

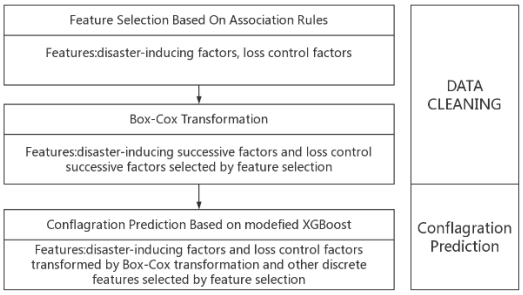

图 1. 总体技术流程图

2.1 基于关联规则特征选择

火灾事故关联挖掘的目的是要自动发现原始事故数据项之间存在的隐含关联关系。此方法基于关联规则特征选择通过优先选取后件为标签的短规则达到减小特征空间维度的目的，包括两个阶段：生成规则集及构造属性集。人工设置支持度、置信度和循环次数使其满足如下目标：1.选择特征子集纬度较低，2.使用此特征子集预测结果较理想。在第一阶段中，本文采用关联规则算法Apriori算法，使用此算法生成规则集合，选择出后件为标签且Lift>1的规则，生成新规则集；在第二阶段中，按照一定准则从第一阶段的新规则集中选择规则，将被选择规则前件无重复添加到属性集，最终生成的属性集为所选择的特征集合。

依据[10]给出的特征选择算法，假设$I=\{i_1,i_2,\ldots,i_m\}$是火灾事故指标项的集合，简称“项”，定义致灾相关的数据D是火灾事故环境的

大数据库事务的集合，其中每个事务 i 是项的集合，有$i \in I$。根据关联规则的现有理论，A⇒B 可用如下参数描述。

支持度：$Support(A \Rightarrow B)$是 D 中事务同时包含 A 和 B 二者的百分比，在火灾事故的数据背景下，可以作为对某个火灾事故规则重要性的描述。

$$Support(A \Rightarrow B) = P(A \cup B) \tag{1}$$

置信度：$Confidence(A \Rightarrow B)$是指 D 中事务同时包含 A,B 占只包含 A 的事务百分比。在火灾事故预警方法中，$Confidence(A \Rightarrow B)$越大说明 B 与 A 关系越密切。

$$Confidence(A \Rightarrow B) = \frac{sup(A \cup B)}{sup(A)} \tag{2}$$

提升度：$Lift(A \Rightarrow B)$表示的是 D 中事务同时包含 A,B 的与 A,B 相互独立前提下 D 中事务同时包含 A,B 的概率的比值[10]。在火灾事故预警方法中，Lift 用来衡量两特征之间关系，若接近于 1 说明两特征接近独立;如果 Lift 小于 1 或大于 1，则说明两特征互相抑制或激发，不独立。

$$Lift(A \Rightarrow B) = \frac{sup(A \cup B)}{sup(A) \times sup(B)} \tag{3}$$

主要算法步骤如下：

Step 1:使用 Apriori 生成强关联规则集；

Step 2:按照后件为标签且 Lift>1 的条件，对强关联规则集进行筛选，生成真实规则集；

Step 3:对真实规则集中规则按一定条件排序，本文中依据长度短、置信度大、支持度大顺序排序；

Step 4:若迭代次数已超过人工设定迭代次数，保存最终生成的特征集并结束算法；否则，从排序后的真实规则集中读取第一条规则，将此规则前件特征无重复加入特征集（即特征集中不应有重复特征），将此规则从真实规则集中去除；

Step 5:对真实规则集中规则按一定条件排序，本文中依据置信度大、支持度大顺序排序，转 Step 4。

2.2 Box-Cox 变换

在火灾事故预警中，样本某特征的连续响应向量不服从正态分布的情况非常普遍，如[11]中所述，为了一定程度上减小不可观测误差与预测变量之间的相关性，若在这时仍然直接对数据建立线性模型，模型拟合效果不好，因此本文对该连续响应变量进行一种非线性变换——Box-Cox 变换[11]。

主要思路：对于连续响应向量$(y^{(0)}, y^{(2)}, \ldots, y^{(n)})$中任意$y^{(i)}$，$y^{(i)} > 0$时$y^{(i)}$的 Box-Cox 变换公式如下：

$y^{(i)} > 0$:

$$y^{(i)(\lambda)} = \begin{cases} \frac{y^{(i)(\lambda)} - 1}{\lambda}, \lambda \neq 0; \\ logy^{(i)(\lambda)}, \lambda = 0 \end{cases} \tag{4}$$

$y^{(i)} \leq 0$:

$$y^{(i)(\lambda)} = \begin{cases} \frac{(y^{(i)(\lambda)} + a)^{\lambda} - 1}{\lambda}, \lambda \neq 0; \\ \log(y^{(i)(\lambda)} + a) y^{(i)(\lambda)}, \lambda = 0 \end{cases} \tag{5}$$

其中$y^{(i)}$为原始数据，$y^{(i)(\lambda)}$为转换后数据。

采用最大似然估计方法对 λ 进行估计：因为$y^{(\lambda)} \sim N(X\beta, \sigma^2 I)$，所以对固定的 λ有$\beta, \sigma^2$的似然函数：

$$L(\beta, \sigma^2) = \frac{1}{(\sqrt{2\pi\sigma^2})^n} \times e^{\{-\frac{1}{2\sigma^2}(Y^{(\lambda)} - X\beta)^T (Y^{(\lambda)} - X\beta)\}J} \tag{6}$$

$$J = \prod_{i=1}^{n} \left| \frac{d_{y_i}^{(\lambda)}}{d_{y_i}} \right| = \prod_{i=1}^{n} y_i^{(\lambda - 1)} \tag{7}$$

$L(\beta, \sigma^2)$关于β和σ求导，令其为 0，得到极大似然估计为：

$$\hat{\beta}(\lambda) = (X^T X)^{-1} X^T Y^{(\lambda)} \tag{8}$$

$$\sigma^2(\lambda) = \frac{1}{n} Y^{(\lambda)T} (I - X^T (X^T X)^{-1} X) Y^{(\lambda)}$$

$$= \frac{1}{n} SSR(\lambda, Y^{(\lambda)}) \tag{9}$$

其中残差平方和为：

$$SSR(\lambda, Y^{(\lambda)}) = Y^{(\lambda)T} (I - X^T (X^T X)^{-1} X) Y^{(\lambda)} \tag{10}$$

得到似然函数的最大值为：

$$L_{max(\lambda)} = L\left(\hat{\beta}(\lambda), \sigma^2(\lambda)\right) = (2\pi e)\text{^}(-n/$$

$$2)\, J(SSR(\lambda, Y\text{^}((\lambda)\,))/n)\text{^}(-n/2) \tag{11}$$

通过求$L_{max}(\lambda)$来确定 λ，因为$L_{max}(\lambda)$是单调函数，因此本问题可化为求$\ln(L_{max}(\lambda))$的最大值，略去与 λ 无关的常数项有：

$$In\left(L_{max}(\lambda)\right) = -\frac{n}{2} Ln\left(SSR\left(\lambda, Y^{(\lambda)}\right)\right) + lnJ$$

$$= -\frac{n}{2} SSR(\lambda, Z^{(\lambda)}) \tag{12}$$

$$其中 Z^{(\lambda)} = \left(Z_1^{(\lambda)}, Z_2^{(\lambda)}, \dots, Z_n^{(\lambda)}\right):$$

$$Z^{(\lambda)} = \begin{cases} \dfrac{y_i^{(\lambda)}}{\left(\prod_{i=1}^{n} y_i\right)^{\frac{\lambda-1}{n}}}, \lambda \neq 0; \\ (lny_i)\left(\prod_{i=1}^{n} y_i\right)^{\frac{1}{n}}, \lambda = 0; \end{cases} \tag{13}$$

为了获得$InL_{max}(\lambda)$的最大值，只需要求$SSR\left(\lambda, Z^{(\lambda)}\right)$的最小值。但是$SSR\left(\lambda, Z^{(\lambda)}\right)$最小值点 λ 通过解析式较难得出，在此使用枚举法，寻求接近最优解$\hat{\lambda}$。综上所述，该过程具体步骤如下：

Step 1:对每一个给定 λ 的值，计算$Z_i^{(\lambda)}$；

Step 2:计算残差平方和$SSR(\lambda, Z_i^{(\lambda)})$；

Step 3:以 λ 为横轴，$SSR(\lambda, Z_i^{(\lambda)})$为纵轴，描点并作出相应的曲线，找出使$SSR(\lambda, Z_i^{(\lambda)})$达到最小值的点；

Step 4:利用 Step 3，求出$Y^{(\lambda)}$。

2.3 基于改进 XGBoost 算法的事故预警

在一个拥有 n 样本个数 m 特征个数数据集$D = \{(x_i, y_i)\}(|D| = n, x_i \in R^m)$中，一个树的组合模型使用 K+函数预测输出值，如图 2 所示。

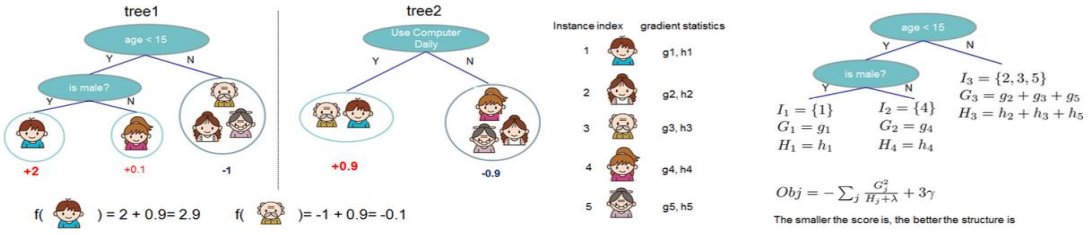

图 2.　左:树的组合模型. 右:树结构评分计算

$$\hat{y}_i = \emptyset(x_i) = \sum_{k=1}^{K} f_k(x_i), f_k \in F \tag{14}$$

$$F = \{f(x) = w_{q(x)}\}(q: R^m \to T, w \in R^m) \tag{15}$$

F 表示回归树空间;q 表示每个树结构,此树结构映射叶子索引实例;w 表示每个叶子特征向量[12]。对于给定样本，使用 q 给出的决策规则将样本分类到不同叶子中，通过累加计算样本对应的不同树的叶子集合对应分数，计算该样本最终预测值。最优化如下正则化公式来学习一个树集合：

$$L(\emptyset) = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k), \Omega(f)$$

$$= r^T + \alpha||w||^2 \tag{16}$$

此正则化公式用来获取训练集上算法预测结果$\hat{y}_i$的质量，正则项Ω用来计算模型复杂度，避免模型过拟合。模型以累加回归树的贪心方式训练，每个循环中，新增新树最优化对象，可以推导出计算新增树结构质量分数的公式：

$$\begin{cases} L^{(t)}(q) = -\frac{1}{2}\sum_{j=1}^{T} \frac{\left(\sum_{i \in I_j} g_i\right)^2}{\sum_{i \in I_j} h_i + \alpha} + r^T \\ g_i = \partial_{\hat{y}_i^{(t-1)}} l\left(\hat{y}_i, y_i^{(t-1)}\right) \\ h_i = \partial_{\hat{y}_i^{(t-1)}}{}^2 l\left(\hat{y}_i, y_i^{(t-1)}\right) \end{cases} \tag{17}$$

质量分数计算公式用于评价决策树，如图 2 右所示，计算叶子节点所有统计样本，然后应用公式(17)计算树质量分数，贪心算法将会迭代搜寻所有可能分裂候选，将最佳分裂添加到树中指导达到最大深度。

3　事故预警模型建立

3.1 数据准备

本文收集了 2011 年至 2017 年全国发生重大火灾的建筑物、当时天气相关数据以及该火灾致灾原

因相关数据，数据表头包括{天气,温度,风级,风向，湿度，建筑物数据特征,标签}。其中，天气数据特征分为：阵雨，小雨，中雨，大雨，多云，晴，阴；温度数据特征为连续数据，为指定日期指定地点的日平均温度；风级数据特征分为 1-5 级风，其中 5 级风包括(>5 级)；风向数据特征表示当前日期当地风向；湿度数据特征表示当前日期当地空气湿度；住宅建筑物数据特征如表 1 所示，公共建筑、厂房建筑和仓库建筑特征与住宅建筑有细微差别，主要表现在损失控制因子。根据建筑物不同使用性质，具有不同特征，这些特征用数值描述。标签数据特征为火灾预警等级，包括：0. 正常无隐患 1. 低度隐患 2. 中度隐患 3. 高度隐患。

表 1. 住宅建筑数据特征

建筑类型	住宅建筑		
致灾因子	燃气方式	人口密度	电气设备安全级别
	内部装修安全级别	建筑物结构	建筑物高度
		屋龄	
损失控制因子	耐火等级	防火隔离级别	安全疏散级别
	安全监控装备级别	室内消火栓装备级别	手提灭火器装备级别

影响火灾发生的因素非常复杂，除了建筑物自身特性，还有商业地区易燃物聚集以及住宅区电路老化、居民防火意识薄弱等等。现阶段无法针对所有影响火灾发生因素进行分析，且选取数据信息因调研目的不同而有所区别。基于上述原因，本文以选取住宅建筑作为实验数据，对住宅建筑建立火灾事故预警系统，其他三类建筑模型建立方法与此类似。采用基于关联规则特征选择算法，研究模型筛选数据特征为：天气(阴晴多云等)，日平均温度，日平均风级，住宅建筑物数据特征作为研究属性。

3.2 数据非线性变换分析

根据基于关联规则的特征选取，此住宅建筑火灾预警模型基于 2011 至 2017 年全国住宅火灾案例详细数据，选取特征包括天气(阴晴多云等)，日平均温度，日平均风级，住宅建筑物数据特征，对原始数据符号标记为：天气—weat,日平均温度—temp,日平均风级—wind, 建筑物使用性质—use,以及住宅建筑物数据特征。

对于燃气方式、建筑物结构等类别特征，本系统对其进行独热处理，如将四分类特征燃气方式变换为燃气方式_0、燃气方式_1、燃气方式_2、燃气方式_3 四个二分类特征，建筑物结构特征也依此变换。

根据 2.2 数据变换过程，对数据集中连续变量进行变换寻求最优 λ，降低数据中异常数据噪声等难以去除的误差与标签之间的关联程度。对住宅建筑日平均温度 res_temp，人口密度 res_popu，建筑物高度 res_heig,屋龄 res_age 进行 Box-Cox 非线性变换。变换后符号标记为：日平均温度 bc_res_temp，人口密度 bc_res_popu，建筑物高度 bc_res_heig，屋龄 bc_res_age。

对各气象因子和建筑物信息的原始数据做非线性变换。原始数据中一部分数据数值为 0，需要选取 box-cox 变换公式的拓展公式，日平均温度选项 $a_1 = 30$，建筑物高度选项 $a_2 = 1$,对建筑物屋龄选项 $a_3 = 1$,对建筑物易燃物聚集指数选项 $a_4 = 1$。最终计算得 res_temp,res_heig,res_age 最优 λ 分别为 4.51039571649、-0.775774467944、0.988957448167。对于 res_popu，进行 Box-Cox 变换后，建筑物人口密度包含负数且数据值分布不理想，保持原数据；对于 res_temp 使用最大似然估计原理方法确定，产生的值远远大于其他数据，因

此使用枚举法多次尝试，最终选择最优 λ 值为0.5，最终结果如表 2 所示。其中天气特征中 0 表示大雨，1 表示中雨，2 表示小雨，3 表示阵雨，4 表示阴，5 表示多云，6 表示晴，7 表示其他天气情况；燃气方式_0 表示是否为管道输送（1 表示 0，0 表示否，以下同理），燃气方式_1 表示是否为高压瓶装天气；建筑物结构_0 表示是否为框架结构，建筑物结构_1 表示是否为砖瓦结构，建筑物结构_2 表示是否为混合结构，建筑物结构_3 表示是否为其他结构。

表 2. Box-Cox 变换后的火灾致灾因素与各特征因子数据

编号	天气	风级	燃气方式_0	燃气方式_1	人口密度	电气设备安全级别	内部装修安全级别
0	5	4	1	0	0.4	2	1
1	0	2	0	1	0.05	2	2
2	3	3	1	0	0.4	2	2
3	3	2	1	0	0.03	3	2
4	5	2	1	0	0.5	3	2
…	…	…	…		…	…	…
3321231	3	3	1	0	0.80061	3	2

编号	防火隔离级别	安全疏散级别	安全监控装备级别	室内消火栓装备级别	手提灭火器装备级别	温度
0	0	3	2	0	2	11.856
1	2	1	2	0	0	12.282
2	0	0	1	0	1	10.806
3	1	3	2	1	3	12
4	1	0	2	1	2	12.560
…	…	…	…	…	…	…
3321231	2	2	3	3	2	11.360

编号	建筑物结构_0	建筑物结构_1	建筑物结构_2	建筑物结构_3	高度	耐火等级	屋龄
0	1	0	0	0	1.112798	2	9.821004
1	0	0	0	1	1.112798	1	11.7669
2	1	0	0	0	1.112798	2	7.871179
3	0	1	0	0	1.227055	2	6.894527
4	0	0	0	1	1.227055	1	5.916523
…	…	…	…	…	…	…	…
3321231	1	0	0	0	12	3	7.871179

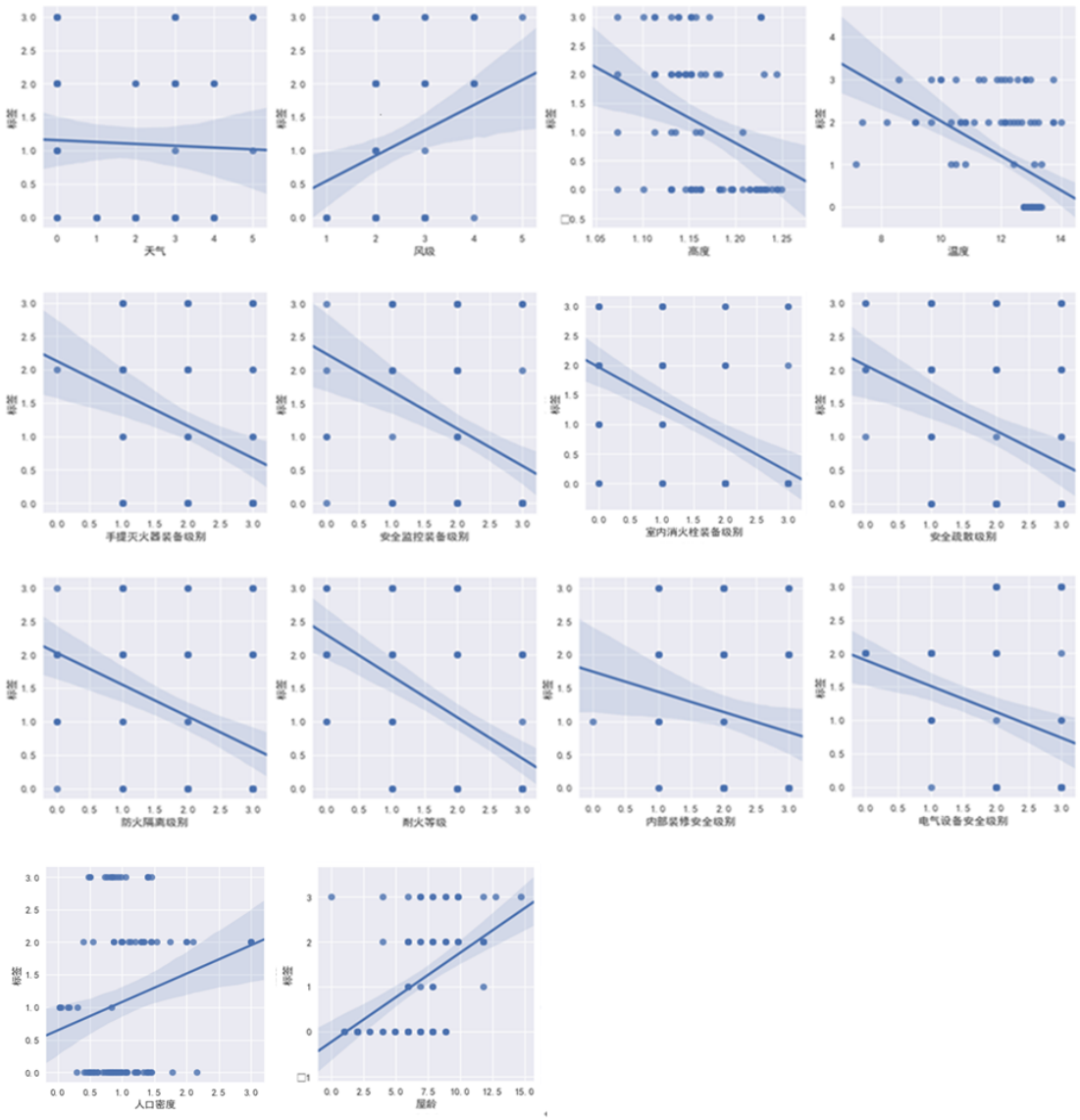

图 3.　不同特征与结果标签对应线性回归模型图

本文尝试用线性回归模型图的方式指出火灾致灾隐患与各影响要素之间的相关关系，确定火灾致灾隐患与各影响要素之间是否存在一定关系，如图 3 所示。由图 3 所示，风级较大情况相对风级较小时更易产生火灾；从高度与标签线性回归模型图中可看出，一定高度区域中，高度越低，产生火灾可能性相对强，这可能由于较低高度的住宅楼更偏向于屋龄较大的住宅楼，其一些其他致灾因子相对更易产生火灾；从温度与标签的线性回归模型图中可看出，相对高温，较小的处理后温度更易产生火灾，这令人困惑，但伴随着温度的降低，电气使用率增加等原因会导致火灾的发生；从电气设备安全级别、防火隔离级别、内部装修安全级别、手提灭火器装备级别、建筑物耐火等级、安全监控装备级别、安全疏散级别、室内消火栓装备级别与标签的线性回归模型图中得出其等级越高，越不易发生火灾；从人口密度与标签的线性回归模型图中得出人口密度的增大更易导致火灾，不过其趋势较为平缓；从屋龄与标签的线性回归模型图中得出随着屋龄增加，一定程度上房屋偏于老化时，发生火灾概率增加。

4　结果分析

由于数据集的不平衡性，本系统采用组合/集成方式对数据集进行批次训练，即将非正常类别样本

（即结果标签为 1、2、3 的样本）随机分为 100 份（当然也可以分更多），每份 100 条数据，每批次中包含正常类别样本（100 条）和随机抽取的非正常类别样本（共 300 条），如此反复进行集成训练。

将清洗完成的数据使用 XGBoost 算法进行训练，输入空间选取的特征包括天气(阴晴多云等)，日平均温度，日平均风级，住宅建筑物数据特征。XGBoost python module 工具包包含三种参数，分别为通用参数、Booster 参数和学习目标参数，根据每个参数之间关系，依次调试(max_depth, min_child_weight)，(gamma)，(subsample, colsample_bytree)，(reg_alpha)，通过迭代并逐步添加参数调试，最终选取最优参数，具体模型参数如表 3 所示。

表 3.　XGBoost 模型参数表

参数类型	参数	解释
通用参数	*booster*	选择每次迭代的模型：gbtree 或 gbliner
	silent	设置成 0 可以将运行信息输出，1 则无信息输出
	nthread	算法使用线程数，不设置则默认启动最多线程
Booster 参数	*eta*	算法学习速率
	min_child_weight	定义一个孩子结点中所有样本最小权值和，用以控制过拟合
	max_depth	构建树的深度，值过大会导致过拟合，默认值为 6
	gamma	节点分裂所需最小损失优化值，默认值为 0，
	subsample	随机采样训练样本，值过大会导致过拟合，默认值为 1
	colsample_bytree	生成树时进行的列采样，值过大会导致过拟合，默认值为 1
	lambda	控制模型复杂度的 L2 正则化项参数
	alpha	控制模型复杂度的 L1 正则化项参数
	scale_pos_weight	用于加速倾斜类数据的收敛，默认值为 1
学习目标参数	*objective*	定义最优化损失函数
	eval_metric	验证数据的度量标准，常用参数如：rmse,mae 等
	seed	用于产生可重复结果，默认值为 0

通过迭代并逐步添加参数调试，最终选取最优参数，过程如下：

由图 4 可知，随着迭代次数的增加，损失函数呈指数下降，即迭代次数越多，损失函数越小，XGBoost 模型预测结果越精确。因此，在调试参数时，需不断增加迭代次数，保证获取最优参数。

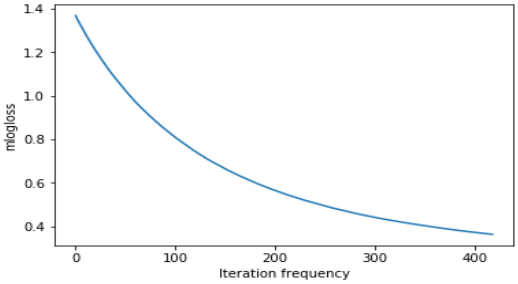

图 4.　损失函数与迭代次数关系图

Step 1：调试 max_depth 和 min_child_weight 两参数，首先调试此参数是因为这两个参数对于模型输出结果具有最显著的影响。由图 5 可得 max_depth=4,min_child_weight=0 时，此时模型对于测试集的预测精度为 86.34%，模型最优。

```
([mean: 0.84190, std: 0.08176, params: {'max_depth': 3, 'min_child_weight': 1},
  mean: 0.82372, std: 0.07071, params: {'max_depth': 3, 'min_child_weight': 3},
  mean: 0.78262, std: 0.06429, params: {'max_depth': 3, 'min_child_weight': 5},
  mean: 0.86195, std: 0.07728, params: {'max_depth': 5, 'min_child_weight': 1},
  mean: 0.82372, std: 0.07071, params: {'max_depth': 5, 'min_child_weight': 3},
  mean: 0.78262, std: 0.06429, params: {'max_depth': 5, 'min_child_weight': 5},
  mean: 0.85142, std: 0.06794, params: {'max_depth': 7, 'min_child_weight': 1},
  mean: 0.82372, std: 0.07071, params: {'max_depth': 7, 'min_child_weight': 3},
  mean: 0.78262, std: 0.06429, params: {'max_depth': 7, 'min_child_weight': 5},
  mean: 0.85142, std: 0.06794, params: {'max_depth': 9, 'min_child_weight': 1},
  mean: 0.82372, std: 0.07071, params: {'max_depth': 9, 'min_child_weight': 3},
  mean: 0.78262, std: 0.06429, params: {'max_depth': 9, 'min_child_weight': 5}],
 {'max_depth': 5, 'min_child_weight': 1},
 0.8619503303713831)

([mean: 0.86339, std: 0.06536, params: {'max_depth': 4, 'min_child_weight': 0},
  mean: 0.83324, std: 0.05726, params: {'max_depth': 4, 'min_child_weight': 0.1},
  mean: 0.85186, std: 0.04947, params: {'max_depth': 4, 'min_child_weight': 0.5},
  mean: 0.84190, std: 0.08176, params: {'max_depth': 4, 'min_child_weight': 1},
  mean: 0.83181, std: 0.05438, params: {'max_depth': 5, 'min_child_weight': 0},
  mean: 0.83037, std: 0.04847, params: {'max_depth': 5, 'min_child_weight': 0.1},
  mean: 0.86095, std: 0.04654, params: {'max_depth': 5, 'min_child_weight': 0.5},
  mean: 0.86195, std: 0.07728, params: {'max_depth': 5, 'min_child_weight': 1},
  mean: 0.83181, std: 0.05438, params: {'max_depth': 6, 'min_child_weight': 0},
  mean: 0.84090, std: 0.05513, params: {'max_depth': 6, 'min_child_weight': 0.1},
  mean: 0.83080, std: 0.01411, params: {'max_depth': 6, 'min_child_weight': 0.5},
  mean: 0.85142, std: 0.06794, params: {'max_depth': 6, 'min_child_weight': 1}],
 {'max_depth': 4, 'min_child_weight': 0},
 0.8633857370699477)
```

图 5.　XGBoost 关于 max_depth 和 min_child_weight 的模型训练过程

Step 2：基于 Step 1 的结果调试 gamma，由图 6 可得 gamma=0 时，此时此时模型对于测试集的预测精度为 86.34%，模型最优。

```
([mean: 0.86339, std: 0.06536, params: {'gamma': 0.0},
  mean: 0.83181, std: 0.04300, params: {'gamma': 0.1},
  mean: 0.83181, std: 0.04300, params: {'gamma': 0.2},
  mean: 0.82272, std: 0.04927, params: {'gamma': 0.3},
  mean: 0.82272, std: 0.04927, params: {'gamma': 0.4},
  mean: 0.83181, std: 0.04300, params: {'gamma': 0.5},
  mean: 0.83181, std: 0.04300, params: {'gamma': 0.6},
  mean: 0.83181, std: 0.04300, params: {'gamma': 0.7},
  mean: 0.83181, std: 0.04300, params: {'gamma': 0.8}],
 {'gamma': 0.0},
 0.8633857370699477)
```

图 6.　XGBoost 关于 gamma 的模型训练过程

Step 3：基于 Step 2 的结果调试 subsample 和 colsample_bytree，经过多次尝试分析，由图 7 可得当 subsample=0.6 ，colsample_bytree=0.7 此时模型对于测试集的预测精度为 88.34%，模型最优。

```
[mean: 0.85186, std: 0.03660, params: {'colsample_bytree': 0.6, 'subsample': 0.6},
 mean: 0.86339, std: 0.06536, params: {'colsample_bytree': 0.6, 'subsample': 0.7},
 mean: 0.85286, std: 0.06368, params: {'colsample_bytree': 0.6, 'subsample': 0.8},
 mean: 0.83181, std: 0.04300, params: {'colsample_bytree': 0.6, 'subsample': 0.9},
 mean: 0.88344, std: 0.06004, params: {'colsample_bytree': 0.7, 'subsample': 0.6},
 mean: 0.86339, std: 0.07335, params: {'colsample_bytree': 0.7, 'subsample': 0.7},
 mean: 0.85142, std: 0.04899, params: {'colsample_bytree': 0.7, 'subsample': 0.8},
 mean: 0.86195, std: 0.06128, params: {'colsample_bytree': 0.7, 'subsample': 0.9},
 mean: 0.88243, std: 0.04310, params: {'colsample_bytree': 0.8, 'subsample': 0.6},
 mean: 0.85286, std: 0.06368, params: {'colsample_bytree': 0.8, 'subsample': 0.7},
 mean: 0.86339, std: 0.06536, params: {'colsample_bytree': 0.8, 'subsample': 0.8},
 mean: 0.85142, std: 0.04899, params: {'colsample_bytree': 0.8, 'subsample': 0.9},
 mean: 0.85186, std: 0.03660, params: {'colsample_bytree': 0.9, 'subsample': 0.6},
 mean: 0.82272, std: 0.05946, params: {'colsample_bytree': 0.9, 'subsample': 0.7},
 mean: 0.83324, std: 0.06623, params: {'colsample_bytree': 0.9, 'subsample': 0.8},
 mean: 0.82272, std: 0.04927, params: {'colsample_bytree': 0.9, 'subsample': 0.9}],
{'colsample_bytree': 0.7, 'subsample': 0.6},
0.8834358623832307)

([mean: 0.83224, std: 0.05234, params: {'colsample_bytree': 0.6, 'subsample': 0.5},
  mean: 0.86238, std: 0.05182, params: {'colsample_bytree': 0.6, 'subsample': 0.55},
  mean: 0.85186, std: 0.03660, params: {'colsample_bytree': 0.6, 'subsample': 0.6},
  mean: 0.86339, std: 0.06536, params: {'colsample_bytree': 0.6, 'subsample': 0.65},
  mean: 0.84277, std: 0.04778, params: {'colsample_bytree': 0.65, 'subsample': 0.5},
  mean: 0.85186, std: 0.03660, params: {'colsample_bytree': 0.65, 'subsample': 0.55},
  mean: 0.86238, std: 0.03972, params: {'colsample_bytree': 0.65, 'subsample': 0.6},
  mean: 0.86339, std: 0.06536, params: {'colsample_bytree': 0.65, 'subsample': 0.65},
  mean: 0.84277, std: 0.04778, params: {'colsample_bytree': 0.7, 'subsample': 0.5},
  mean: 0.86238, std: 0.03972, params: {'colsample_bytree': 0.7, 'subsample': 0.55},
  mean: 0.87291, std: 0.05198, params: {'colsample_bytree': 0.7, 'subsample': 0.6},
  mean: 0.85286, std: 0.06368, params: {'colsample_bytree': 0.7, 'subsample': 0.65},
  mean: 0.86282, std: 0.05787, params: {'colsample_bytree': 0.75, 'subsample': 0.5},
  mean: 0.87191, std: 0.04818, params: {'colsample_bytree': 0.75, 'subsample': 0.55},
  mean: 0.87291, std: 0.05198, params: {'colsample_bytree': 0.75, 'subsample': 0.6},
  mean: 0.88344, std: 0.06004, params: {'colsample_bytree': 0.75, 'subsample': 0.65}],
 {'colsample_bytree': 0.7, 'subsample': 0.6},
 0.8834358623832307)
```

图 7.　XGBoost 关于 subsample 和 colsample_bytree 的模型训练过程

Step 4：基于 Step 3 的结果调试正则项 reg_alpha，经过多次尝试分析，由图 8 可得当 reg_alpha=0.000001，此时模型对于测试集的预测精度为 89.991%，模型最优。

```
([mean: 0.87291, std: 0.05198, params: {'reg_alpha': 0},
  mean: 0.89991, std: 0.05198, params: {'reg_alpha': 1e-05},
  mean: 0.87291, std: 0.05198, params: {'reg_alpha': 0.01},
  mean: 0.85286, std: 0.05429, params: {'reg_alpha': 0.1},
  mean: 0.84377, std: 0.07083, params: {'reg_alpha': 1},
  mean: 0.50194, std: 0.03063, params: {'reg_alpha': 100}],
 {'reg_alpha': 0.01},
 0.8999095465937571)
```

图 8.　XGBoost 关于 reg_alpha 的模型训练过程

根据所调整参数，生成火灾预警模型对建筑物进行实时火灾预警分析，其模型生成报告如图 9 所示，Feature Importance 描述了不同特征对于预警结果的重要性。采用火灾预警模型生成的火灾预警系统界面截图如图 10 所示。由图 9 可知，本系统在测试集上平均误差为 0.1，测试集上误差的标准差为 0.044721，由 Feature Importance 图可知，人口密度、天气、风级、电气设备安全级别对火灾预警系统影响较大（其中['r_weat','r_EDM','r_wind','r_PD','r_EESL','r_ISL','r_BS','r_FRL','r_FIL','r_SEL','r_ML','r_HEL','r_FEEL','r_temp','r_BH','r_AOB']分别表示['天气','燃气方式','风级','人口密度','电气设备安全级别','内部装修安全级别','建筑物结构','耐火等级','防火隔离级别','安全疏散级别','安全监控装备级别','室内消火栓装备级别','手提灭火器装备级别','温度','高度','屋龄']）。

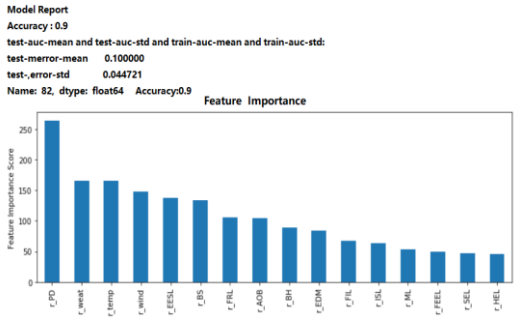

图 9.　模型信息报告

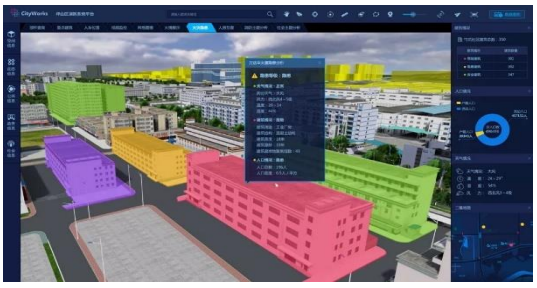

图 10.　火灾预警系统界面截图

5　结论

本文提出一种新的基于 XGBoost 的事故预警方法，该方法使用基于关联规则特征选择方法，减少

数据特征集合大小，清除不必要数据特征，减少算法运行时间，对连续响应变量进行 Box-Cox 非线性变换减小不可观测误差与预测变量之间的相关性；在火灾事故预警系统中，数据量较大，采用神经网络或其他决策树增强模型消耗时间较多。XGBoost 针对处理海量数据，提供缓存感知预读取技术、分布式外存计算技术、AllReduce 容错工具提高现有提升树增强算法运算速率，提升算法运行速率。通过实验对 XGBoost 参数调整，最终使得模型对测试集的预测精度达到 90%，效果较为理想。

未来可以采用多个模型对数据进行预处理，去除更多脏数据。获取更多火灾相关数据与火灾产生原因进行相关性分析，结合时间序列这一关键因素提取各类信息在时间维度上的特征，将分析结果作为新的特征加入火灾预警模型，进一步提升模型预测精度。因此，将此方法应用于消防预警系统，更好的为城市消防提供帮助，使城市消防趋向智能化，这也是智慧城市建设中重要的一部分。

参考文献

[1] 郑双忠，邓云峰，蒋清华. 基于火灾统计灾情数据的城市火灾风险分析[J]. 中国安全生产科学技术，2005，1(3):15-18.

[2] 张学林，孙志友，汪金辉,等. 基于马尔可夫链的城市火灾预测[J]. 火灾科学，2006，15(3):168-171.

[3] 刘德志. 城市火灾报警智能监控终端的研究与应用[D]. 广东工业大学，2013.

[4] Sharma D, Singh K, Aggarwal S. Implementation of Artificial Neural Fuzzy Inference System in a Real Time Fire Detection Mechanism[J]. International Journal of Computer Applications, 2016, 146(10).

[5] Mavroeidis D, Marchiori E. Feature selection for k-means clustering stability: theoretical analysis and an algorithm[J]. Data Mining and Knowledge Discovery, 2014, 28(4): 918-960.

[6] Alamgir N, Boles W, Chandran V. A Model Integrating Conflagration prediction and Detection for Rural-Urban Interface[C]// International Conference on Digital Image Computing: Techniques and Applications. IEEE, 2016:1-8.

[7] Gao D, Lin H, Jiang A, et al. A forest conflagration prediction system based on rechargeable wireless sensor networks[C]// IEEE International Conference on Network Infrastructure and Digital Content. IEEE, 2015:405-408.

[8] Wang X, Wotton B M, Cantin A S, et al. cffdrs: an R package for the Canadian Forest Fire Danger Rating System[J]. Ecological Processes, 2017, 6(1):5.

[9] Oliveira A, Nero M. Application of Fuzzy Logic in Prediction of Fire in João Pessoa City - Brazil[M]// Geo-Informatics in Resource Management and Sustainable Ecosystem. Springer Berlin Heidelberg, 2013:323-334.

[10] Chandrashekar G, Sahin F. A survey on feature selection methods[J]. Computers & Electrical Engineering, 2014, 40(1): 16-28.

[11] Bicego M, Baldo S. Properties of the Box-Cox Transformation for Pattern Classification[J]. Neurocomputing, 2016, 218:390-400.

[12] Chen T, Guestrin C. XGBoost: A Scalable Tree Boosting System[J]. 2016:785-794.

[13] Taylor S W, Alexander M E. Field guide to the Canadian Forest Fire Behavior Prediction (FBP) System. (BINDER)[M]. 2016.

[14] 伍爱友，施式亮，王从陆. 基于神经网络和遗传算法的城市火灾风险评价模型[J]. 中国安全科学学报，2006，16(11):108-113.

[15] Stojanova D, Kobler A, Ogrinc P, et al. Estimating the risk of fire outbreaks in the natural environment[J]. Data Mining and Knowledge Discovery, 2012, 24(2): 411-442.

[16] 方正，陈娟娟，谢涛,等. 基于聚类分析和 AHP 的商场类建筑火灾风险评估[J]. 东北大学学报(自然科学版)，2015，36(3):442-447.

[17] Ooi C H, Chetty M, Teng S W. Differential prioritization in feature selection and classifier aggregation for multiclass microarray datasets[J]. Data Mining and Knowledge Discovery, 2007, 14(3): 329-366.

[18] Bandyopadhyay S, Wolfson J, Vock D M, et al. Data mining for censored time-to-event data: a Bayesian network model for predicting cardiovascular risk from electronic health record data[J]. arXiv preprint arXiv:1404.2189, 2014.

[19] Konda P, Kumar A, Ré C, et al. Feature selection in enterprise analytics: a demonstration using an R-based data analytics system[J]. Vldb Demo, 2013, 6(12):1306-1309.

[20] Pibiri G E, Venturini R. Clustered Elias-FanoIndexes[J]. ACM Transactions on Information Systems (TOIS), 2017, 36(1): 2.

[21] 武建华，宋擒豹，沈均毅,等. 基于关联规则的特征选择算法[J]. 模式识别与人工智能，2009, 22(2):256-262.

[22] Box G E P, Cox D R. An analysis of transformations[J]. Journal of the Royal Statistical Society. Series B (Methodological), 1964: 211-252.

[23] Yang L, Dawson C W, Brown M R, et al. Neural network and GA approaches for dwelling fire occurrence prediction[J]. Knowledge-Based Systems, 2006, 19(4):213-219.