鲁东大学学报(自然科学版)
Journal of Ludong University(Natural Science Edition) 2021,37(2):131—138

基于模糊多目标线性规划的软件缺陷预测方法研究

吴瑞霞[1],张志旺[1],王 琰[1],周 莉[1],岳 峻[1],卢泰然[2]

(1.鲁东大学 信息与电气工程学院,山东 烟台 264039;2.山东交通学院 信息科学与电气工程学院(人工智能学院),济南 250357)

摘要:为了提高软件开发过程的可测性和可信性,本文在分析软件缺陷预测数据特点的基础上,提出了一种新的带特征选择的模糊多目标线性规划分类器 FMCLPC-FS。首先,定义了一个模糊隶属度函数来处理原始数据中的噪声和异常值;然后,利用核函数将非线性可分问题转化为线性可分或近似线性可分问题。此外,在多目标线性规划分类器 MCLPC 中引入了稀疏化函数,可以在分类过程中去除数据集中的冗余特征并选择出最少的重要特征。实验结果显示,与 MCLPC 和 SVC 相比,FMCLPC-FS 可以显著提高缺陷预测的准确性和分类的可解释性。

关键词:模糊隶属度;多目标线性规划;软件缺陷预测

中图分类号:TP391 **文献标志码**:A **文章编号**:1673-8020(2021)02-0131-08

近年来,软件缺陷预测引起了软件工程领域研究者的广泛关注[1—3],基于机器学习的软件缺陷预测方法也成为了研究热点[4—8]。软件缺陷预测方法主要根据经验建立预测模型,预测软件模块是否有缺陷,并且把软件模块划分为缺陷模块和无缺陷模块(即二元分类问题),以便为管理者提供决策依据,优化软件开发过程,并达到合理分配测试资源、提高测试效率的目的[9]。

在过去的研究中,已经提出了多种软件缺陷预测方法。典型的方法有决策树[10]、随机森林[11]、k 最近邻算法[12]、支持向量机(support vector machine,SVM)[13]和朴素贝叶斯分类方法[14]等。然而,决策树模型难以识别相关特征,且由于数据中存在噪声,容易发生过拟合。随机森林不容易过拟合,但计算速度较慢。k 最近邻算法不需要训练步骤,但是计算过程很昂贵[15]。基于 SVM 的软件缺陷预测模型表明,对于中小型数据集,SVM 分类器可以获得很好的结果,但是在大规模数据集上需要解决计算复杂度高的凸二次规划问题。

本文在前人的研究基础上,对多目标线性规划分类器(multi-criteria linear programming classifier,MCLPC)进行改进,引入模糊隶属函数来降低噪声和异常值的影响,用稀疏化方法选取重要特征,以期提高分类器的性能和效率。

本文先介绍 MCLPC,再介绍带特征选择的模糊多目标线性规划分类器(fuzzy multi-criteria linear programming classifier with feature selection,FMCLPC-FS),最后进行实验验证与结果分析。

1 多目标线性规划分类器

MCLPC 的主要思想是在属于不同类别的输入点的重叠程度与输入点到其类边界的总距离之间找到一种平衡[16]。对于二分类问题,给定训练样本集 $T=\{(\boldsymbol{x}_1,y_1),\cdots,(\boldsymbol{x}_n,y_n)\}$,每个样本点 $\boldsymbol{x}_i(\boldsymbol{x}_i\in\mathbf{R}^d)$ 属于具有标签 $y_i\in\{1,-1\}$ 的两个类别中的任何一个,其中 $i=1,2,\cdots,n$,n 是样本数,d 是输入空间的维度。如果 $\boldsymbol{x}_i$ 属于类 1,则 $y_i=1$;反之,$\boldsymbol{x}_i$ 属于类 2,则 $y_i=-1$。为了区分不同的类别,Freed 等[17]选择了两个度量:样本点偏离分割超平面的距离和样本点到分割超平面的距离,对于前者,样本点分类错误,则最小化它到超平面的距离;对于后者,样本点分类正确,应该最大化它到超平面的距离。随后,Glover 考虑了上述两种情况并构建了多目标规划分类模型[18]。

收稿日期:2020-12-16;修回日期:2021-03-04

基金项目:国家自然科学基金(61877061,61872170);山东省自然科学基金(ZR2016FM15)

第一作者简介:吴瑞霞(1994—),女,陕西汉中人,硕士研究生,研究方向为数据挖掘、机器学习。E-mail:ruiwrx@126.com

通信作者简介:张志旺(1973—),男,山西岚县人,教授,硕士研究生导师,博士,研究方向为数据挖掘与知识发现、机器学习、人工智能和自然语言处理。E-mail:zzwmis@163.com

(C)1994-2021 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

设$\alpha_i(\alpha_i \geqslant 0)$是错误分类的样本点$\boldsymbol{x}_i$偏离超平面的距离,则应最小化距离$\alpha_i$之和(称为重叠度),即最小化目标函数$\sum_{i=1}^{n}\alpha_i$,且优化问题可表示为:

$$\begin{aligned}&\min\sum_{i=1}^{n}\alpha_i\\&\text{s.t.}\,y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i-b)\geqslant-\alpha_i,\\&\alpha_i\geqslant 0,i=1,2,\cdots,n,\end{aligned}\tag{1}$$

其中:$\boldsymbol{w}=(w_1,\cdots,w_d)^{\mathrm{T}}$是权重变量;$b$表示截距,是一个标量。显然,如果样本点$\boldsymbol{x}_i$被正确分类,则$\alpha_i=0$;如果样本点$\boldsymbol{x}_i$被错误分类,则$\alpha_i>0$。

同理,设$\beta_i(\beta_i\geqslant 0)$是正确分类的样本点$\boldsymbol{x}_i$到超平面的距离,则应最大化距离$\beta_i$之和,即最大化目标函数$\sum_{i=1}^{n}\beta_i$,且优化问题可表示为:

$$\begin{aligned}&\max\sum_{i=1}^{n}\beta_i\\&\text{s.t.}\,y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i-b)\leqslant\beta_i,\\&\beta_i\geqslant 0,i=1,2,\cdots,n。\end{aligned}\tag{2}$$

如果样本点$\boldsymbol{x}_i$分类错误,则$\beta_i=0$;如果样本点$\boldsymbol{x}_i$分类正确,则$\beta_i>0$。

设α_i为样本点的最大下界,则式(1)中的不等式约束可转化为等式约束$y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i-b)=-\alpha_i$。设$\beta_i$为输入点$\boldsymbol{x}_i$的最小上界,则式(2)中的不等式约束可转化为等式约束$y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i-b)=\beta_i$。由于$\alpha_i>0$时,$\beta_i=0$,且$\beta_i>0$时,$\alpha_i=0$,则可将式(1)和式(2)分类器模型中的相应约束同时集成到一个分类器模型中,则MCLPC模型可定义为

$$\begin{aligned}&\min\sum_{i=1}^{n}\alpha_i\ \text{and}\ \max\sum_{i=1}^{n}\beta_i\\&\text{s.t.}\,y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i-b)=\beta_i-\alpha_i,\\&\alpha_i,\beta_i\geqslant 0,i=1,2,\cdots,n。\end{aligned}\tag{3}$$

另外,为了对错分样本进行惩罚,令$C(C>0)$为目标函数$\sum_{i=1}^{n}\alpha_i$的惩罚因子,它是输入参数。因此,本文可以将式(3)中的分类器模型的目标函数写为线性加权函数$C\sum_{i=1}^{n}\alpha_i-\sum_{i=1}^{n}\beta_i$,得到新的MCLPC模型:

$$\begin{aligned}&\min C\sum_{i=1}^{n}\alpha_i-\sum_{i=1}^{n}\beta_i\\&\text{s.t.}\,y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i-b)=\beta_i-\alpha_i,\\&\alpha_i,\beta_i\geqslant 0,i=1,2,\cdots,n。\end{aligned}\tag{4}$$

通过求解式(4)的方程组,可以得到权重向量$\boldsymbol{w}$和标量b的值。这样对于新的输入点$\boldsymbol{x}$,可用如式(5)所示的决策函数进行预测:

$$f(\boldsymbol{x})=\operatorname{sgn}(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}-b)。\tag{5}$$

2 带特征选择的模糊多目标线性规划分类器

2.1 模糊隶属度

由于MCLPC对数据中的噪声和异常值非常敏感,分类的准确度会降低。因此,在MCLPC模型中引入适当的模糊隶属度函数来解决这一问题。模糊隶属度的作用是对不同的输入点赋予不同的权值,将较大的模糊隶属度值分配给靠近类中心的点,将较小的模糊隶属度值分配给噪声和离群点。这样对不同的样本应用不同的惩罚因子,使得不同的样本在构造目标函数时具有不同的贡献,从而消除噪声和离群值的影响。在软件缺陷预测技术中,也需要考虑数据中噪声和异常值等特异点的影响。因此,本文引入模糊隶属度函数,可以计算出不同输入点对应的模糊隶属度值。

基于原样本集T构造新的样本集$T'=\{(\boldsymbol{x}_1,y_1,s_1),\cdots,(\boldsymbol{x}_n,y_n,s_n)\}$,其中$\boldsymbol{x}_i\in\mathbf{R}^n,y_i\in\{-1,1\}$。$s_i$为带有类标签的样本点$(\boldsymbol{x}_i,y_i,s_i)$的模糊隶属度值,且$0\leqslant s_i\leqslant 1,i=1,2,\cdots,n$。每个样本点$\boldsymbol{x}_i$的模糊隶属度$s_i$可定义为:

$$s_i=1-\frac{\|\boldsymbol{x}_i-\bar{\boldsymbol{x}}\|_2}{\max\|\boldsymbol{x}_i-\bar{\boldsymbol{x}}\|_2+\zeta},\tag{6}$$

其中$\bar{x}$表示样本的均值,且$\zeta(\zeta>0)$是一个足够小的常数,用于避免式(6)的分母出现零除。相应的模糊隶属度向量$\boldsymbol{s}=(s_1,\cdots,s_n)^{\mathrm{T}},\boldsymbol{s}\in\mathbf{R}^n$。

2.2 FMCLPC-FS模型

L1范数具有产生稀疏模型的能力,使用L1范数作为正则项的LASSO具有特征选择和特征空间降维的功能[19]。基于此,本文对权重向量$\boldsymbol{w}$加以正则项进行诱导,使其更加稀疏,让大部分的权值都为0。另外,引入稀疏因子$F(F>0)$,构造稀疏函数$F\|\boldsymbol{w}\|_1$并应用到式(4)目标函数中,得到一种新的MCLPC模型:

$$\min F\|\boldsymbol{w}\|_1+C\sum_{i=1}^{n}\alpha_i-\sum_{i=1}^{n}\beta_i$$

(C)1994-2021 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

$$\text{s.t.}\, y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i - b) = \beta_i - \alpha_i,$$
$$\beta_i \geqslant 0, 0 \leqslant \alpha_i \leqslant C, i = 1,2,\cdots,n。 \quad (7)$$

由于式(7)中有一个绝对值 $\|\boldsymbol{w}\|_1 = \sum_{m=1}^{d}|w_m|$,不便于计算,因此,引入一个新的向量 $\boldsymbol{t} = (t_1,\cdots,t_m)$ $(\boldsymbol{t} \in \mathbf{R}^d, m = 1,2,\cdots,d)$,令 $|w_m| \leqslant t_m$,得到一个新的 MCLPC:

$$\min F\sum_{m=1}^{d} t_m + C\sum_{i=1}^{n}\alpha_i - \sum_{i=1}^{n}\beta_i$$
$$\text{s.t.}\, y_i\boldsymbol{w}^{\mathrm{T}}(x_i - b) = \beta_i - \alpha_i,$$
$$0 \leqslant \beta_i, 0 \leqslant \alpha_i \leqslant C, -t_m \leqslant w_m \leqslant t_m,$$
$$0 \leqslant t_m \leqslant F, i = 1,2,\cdots,n, m = 1,2,\cdots,d。 \quad (8)$$

但是,式(8)所示的模型只适用于数据集线性可分的情况,如果数据集是非线性可分的,需要引入合适的基函数通过非线性变换将特征空间映射到新的空间。

对于训练集中的任意两个输入点 $\boldsymbol{x}_i$ 和 $\boldsymbol{x}_j$,给定维数基函数 $\varphi(\boldsymbol{x}_{im})$,它将样本点 $\boldsymbol{x}_i$ 的特征值 $\boldsymbol{x}_{im}$ 从原始输入空间映射到新的特征空间。

对于特征集 X 中的任意一个特征 $\boldsymbol{f}_m = (x_{1m},\cdots,x_{nm})^{\mathrm{T}}$ $(\boldsymbol{f}_m \in \mathbf{R}^n, m = 1,2,\cdots,d)$,给定样本点 $\boldsymbol{x}_i$ 的特征值 x_{im} 与模糊隶属度向量 $\boldsymbol{s}$ 的映射函数 $\varphi(x_{im})$,定义与第 m 个特征对应的列核向量 $\boldsymbol{u}_m(\boldsymbol{u}_m \in \mathbf{R}^n)$ 为

$$\boldsymbol{u}_m = (\boldsymbol{f}_m\boldsymbol{f}_m^{\mathrm{T}})\boldsymbol{s} = [(x_{1m},\cdots,x_{nm})^{\mathrm{T}}(x_{1m},\cdots,x_{nm})]\boldsymbol{s} = [\varphi(x_{1m}),\cdots,\varphi(x_{nm})]^{\mathrm{T}}[\varphi(x_{1m}),\cdots,\varphi(x_{nm})]\boldsymbol{s} =$$
$$\begin{bmatrix} \varphi(x_{1m})^{\mathrm{T}}\varphi(x_{1m}) & \cdots & \varphi(x_{1m})^{\mathrm{T}}\varphi(x_{nm}) \\ \vdots & \ddots & \vdots \\ \varphi(x_{nm})^{\mathrm{T}}\varphi(x_{1m}) & \cdots & \varphi(x_{nm})^{\mathrm{T}}\varphi(x_{nm}) \end{bmatrix}\begin{pmatrix} s_1 \\ \vdots \\ s_n \end{pmatrix} =$$
$$\begin{bmatrix} K(x_{1m},x_{1m}) & \cdots & K(x_{1m},x_{nm}) \\ \vdots & \ddots & \vdots \\ K(x_{nm},x_{1m}) & \cdots & K(x_{nm},x_{nm}) \end{bmatrix}\begin{pmatrix} s_1 \\ \vdots \\ s_n \end{pmatrix} =$$
$$\left[\sum_{j=1}^{n} s_j K(x_{1m},x_{jm}),\cdots,\sum_{j=1}^{n} s_j K(x_{nm},x_{jm})\right], \quad (9)$$

其中,本文用特征核函数 $K(x_{im},x_{jm})$ 取代基函数的内积 $\varphi(x_{im})^{\mathrm{T}}\varphi(x_{jm})$。

在式(9)中,对于任意样本点 $\boldsymbol{x}_i$,对应于该特征的列核矩阵 $\boldsymbol{U}_{im}(\boldsymbol{U}_{im} \in \mathbf{R}^{n\times d}, m = 1,2,\cdots,d)$ 的形式为

$$\boldsymbol{U}_{im} = \sum_{j=1}^{n} s_j K(x_{im},x_{jm}), \quad (10)$$

其中特征核函数 $K(x_{im},x_{jm})$ 本文选择了线性核函数和径向基核函数(RBF),分别定义为:

$$K(x_{im},x_{jm}) = x_{im}x_{jm}, \quad (11)$$
$$K(x_{im},x_{jm}) = \exp\left[-\frac{(x_{im} - x_{jm})^2}{2\sigma^2}\right] (\sigma > 0)。 \quad (12)$$

因此,样本点 x_i 的列核向量 $\boldsymbol{v}_i(\boldsymbol{v}_i \in \mathbf{R}^{d+1})$ 可以记为 $\boldsymbol{v}_i = (\boldsymbol{U}_{i1},\cdots,\boldsymbol{U}_{id}, -1)^{\mathrm{T}}$,则该样本点列核矩阵 $\boldsymbol{V} \in \mathbf{R}^{n\times(d+1)}$) 可记为

$$\boldsymbol{V} = [\boldsymbol{v}_1,\cdots,\boldsymbol{v}_n]。 \quad (13)$$

根据式(13)中的矩阵构造,将式(8)中的 $(\boldsymbol{x}_i - b)$ 替换为式(13)中的矩阵 $\boldsymbol{V}$,并将 b 合并到权重向量 $\boldsymbol{w}$ 中,可得权重向量为 $\boldsymbol{w}' = (w_1,\cdots,w_d, b)^{\mathrm{T}}$,从而得到新的模型 FMCLPC-FS:

$$\min F\sum_{m=1}^{d} t_m + C\sum_{i=1}^{n}\alpha_i - \sum_{i=1}^{n}\beta_i$$
$$\text{s.t.}\, y_i\boldsymbol{w}'^{\mathrm{T}}\boldsymbol{V}_i = \beta_i - \alpha_i$$
$$0 \leqslant t_m \leqslant F, 0 \leqslant \alpha_i \leqslant C, 0 \leqslant \beta_i,$$
$$i = 1,2,\cdots,n, m = 1,2,\cdots,d。 \quad (14)$$

对于任何来自测试集的样本点 $\boldsymbol{x}_i$,它的类标签 y_i 都可以通过决策函数

$$f(x) = \text{sgn}(\boldsymbol{w}'^{\mathrm{T}}\boldsymbol{V}_{x_i}) \quad (15)$$

进行预测。

2.3　FMCLPC-FS 算法

基于上述对 FMCLPC-FS 模型的描述,FMCLPC-FS 模型的算法包括输入、输出和由不同步骤组成的处理流程。

算法 1: FMCLPC-FS 算法

输入: 数据集 $T = \{(\boldsymbol{x}_1,y_1),\cdots,(\boldsymbol{x}_n,y_n)\}$,稀疏因子集 Δ_1 和惩罚因子集 Δ_2,参数 ζ,RBF 核的带宽 σ。

输出: 预测结果以及性能度量等数值。

1) 将原始数据集分割为训练集和测试集,并初始化参数 ζ、σ;

2) 通过式(6)计算模糊隶属度值 s_i;

3) 计算式(13)的列核矩阵 $\boldsymbol{V}$;

4) 利用线性规划方法求解式(14)中的 FMCLPC-FS 模型,得到 $\boldsymbol{w}'$ 的解;

5) 利用公式(15)中的决策函数计算测试集中样本点预测的类标签;

6) 对分类结果进行评估。

如算法 1 所示,首先指定 RBF 核的初始核带宽,将式(6)中的模糊隶属度值代入到式(9)中,进一步计算出式(10)特征的列核矩阵。其次,设

(C)1994-2021 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

置稀疏因子和惩罚因子,预定义参数集,通过式(13)得到样本的列核矩阵,然后计算分类器的权值向量和截距。最后,构造式(15)中的决策函数进行分类,计算预测输出和一些统计值。

从算法复杂度来看,MCLPC采用内点法求解线性规划问题,确定特征变量集后基于等式约束条件寻找最优解,因此它的计算复杂性为$O(n^{3.5}d^2)$。FMCLPC-FS也采用内点法求解线性规划问题,但是该算法确定相关的数据后基于特征变量集来寻找满足约束条件的最优解,因此它的计算复杂性为$O(n^2d^{3.5})$。显然,当处理较大规模的数据时,FMCLPC-FS要比MCLPC的计算效率要高。

3 软件缺陷预测实验

3.1 数据集

本文采用了美国国家航空航天局提供的NASA数据库来进行实验测试。该数据库可以在相应的网站下载[20]。在实验中,以5组软件缺陷预测数据集作为基准来评估MCLPC、SVC和FMCLPC-FS的预测性能,数据集的具体信息如表1所示。

表1 软件缺陷数据集

Tab.1 The software defect datasets

数据集	样本数	正类数	负类数	特征数
KC3	458	43	415	40
CM1	505	48	457	40
PC2	5589	23	5566	40
PC1	1107	76	1031	40
PC4	1458	178	1280	40

在软件缺陷数据集中,每个数据集都包含大量的度量特征。从方法级度量标准的角度来看,有四种类型的度量标准:代码行度量、Halstead度量[21]、McCabe度量[22]和其他度量标准。McCabe度量是对程序内部结构的复杂性分析。Halstead度量是根据程序中的操作符和操作数来度量程序的复杂性。根据表1的统计结果表明,每个数据集都有40个特征,如何从现有的众多特征中筛选出对分类影响较大的特征,建立统一的软件质量度量标准,对当前软件的度量和缺陷数据的收集具有重要意义。

3.2 实验设计

在实验中,从数据集KC3、CM1、PC2、PC1和PC4中分别选取400、450、2500、1000和1200个负类及40、40、20、70和150个正类组成相应的训练集,其余作为测试集。使用5折交叉验证方法在对比分类器MCLPC、SVC和本文分类器FMCLPC-FS上训练数据集,计算测试集的平均预测性能。

此外,使用min-max标准化方法将数据集中的每个特征值X归一化为一个新的变量X',范围从0到1,公式如下

$$X' = \frac{X - X_{\min}}{X_{\max} - X_{\min}}, \tag{16}$$

其中$X_{\min}$和$X_{\max}$分别为特征值的最小值和最大值。

在训练不同分类器的整个过程中,通过网格搜索方法从预定义的集合中选择FMCLPC-FS、SVC和MCLPC的最优参数。分类器的参数集设置为:稀疏因子F和惩罚因子C从集合$\Delta_1=\Delta_2=\{1,2,5,10,20,50,100,200,500,1000\}$中选取,核函数RBF的带宽$\sigma$从集合$\{0.01,0.1,0.2,0.5,1,2,5,10\}$选取。

软件缺陷预测是一个二分类问题,其预测过程将产生4种不同的结果,如表2所示,其中正类为有缺陷模块,负类为无缺陷模块。

表2 分类结果混淆矩阵

Tab.2 Classification result confusion matrix

真实值	预测值	
	正类	负类
正类	*TP*(true positive)	*FN*(false negative)
负类	*FP*(false positive)	*TN*(true negative)

其中TP是正类中正确预测为正类的数量,FN是正类中错误预测为负类的数量。FP是负类中错误预测为正类的数量,TN是负类中正确预测为负类的数量。

在本文的实验中,使用四种度量来评估分类器的预测性能,以下将介绍这四种性能评价指标。

1) 准确率(accuracy,记为A)。准确率是正确分类的样本数与总类数之比。准确率反映了分类器对整个样本的判断能力。定义公式如下:

$$A = \frac{TP + TN}{TP + FN + FP + TN}。 \tag{17}$$

2) 召回率(recall,记为R),正确预测的正类数占总的正类数的比率,定义公式如下:

(C)1994-2021 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

$$R = \frac{TP}{TP + FN}。 \tag{18}$$

3) F-measure(记为 F_1) ,基于第一类分类准确率和第二类分类准确率的调和平均。定义公式如下:

$$F_1 = \frac{2TP}{2TP + FN + FP}。 \tag{19}$$

4) 约简率(reduction rate,记为 R_r) 。对于给定数据集的特征集 X_o和约简的特征集X_r,本文将约简率定义为弱相关特征数与原始特征数的比率,得到

$$R_r = \frac{|X_o| - |X_r|}{|X_o|} \times 100\%, \tag{20}$$

其中|·|为特征集的基数。

4　实验结果与分析

4.1　分类性能比较

通过 SVC、MCLPC 和 FMCLPC-FS 模型在训练子集上进行训练,并在相应的验证子集上进行验证,然后在独立的测试集上进行测试,从而计算出这些分类器的最佳预测性能的平均值,如图 1~3 所示。从图 1~3 可以看出,相较于其它两个模型,本文提出的 FMCLPC-FS 模型总体上具有更好的预测性能,可以有效地处理具有异常和非线性可分性的不确定数据集,减少异常值和数据异常以及非线性可分离情况的影响,可以在一定程度上有助于提高软件缺陷预测的性能。此外,很明显,在分类器的核函数选择上,RBF 核的总体预测性能略优于线性核。

如图 1 所示,FMCLPC-FS 模型的准确率都在 90%以上,在线性核中 CM1、PC2 数据集的准确率甚至接近 100%,在 RBF 核中数据集 CM1、PC2、PC1 和 PC4 的准确率都接近 100%,说明 FMCLPC-FS 模型在软件缺陷预测的准确率上能取得较好的结果。如图 2 所示,FMCLPC-FS 模型在召回率上数值较高,表明分类器对正类的识别能力较强。如图 3 所示,FMCLPC-FS 模型在 F-measure 度量上也优于 SVC 和 MCLPC,说明本文的模型在软件缺陷预测应用上相较于其他两个分类器更有效。

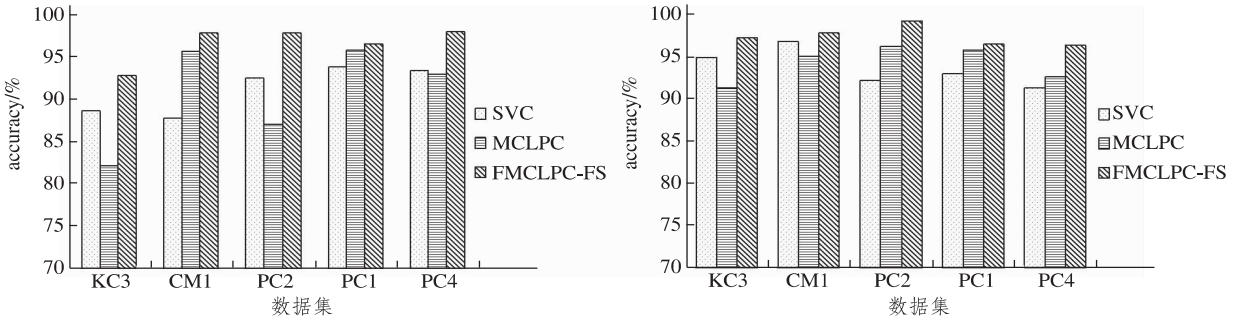

图 1　基于不同核的分类器在 5 个数据集的准确率比较

Fig.1 The accuracy comparison of classifiers based on different kernel functions in 5 datasets

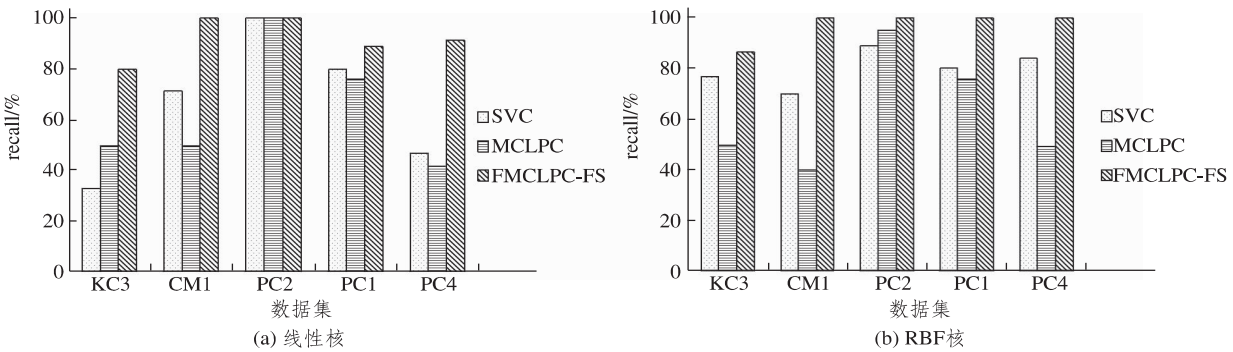

图 2　基于不同核的分类器在 5 个数据集的召回率比较

Fig.2 The recall comparison of classifiers with different kernel functions in 5 datasets

(C)1994-2021 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

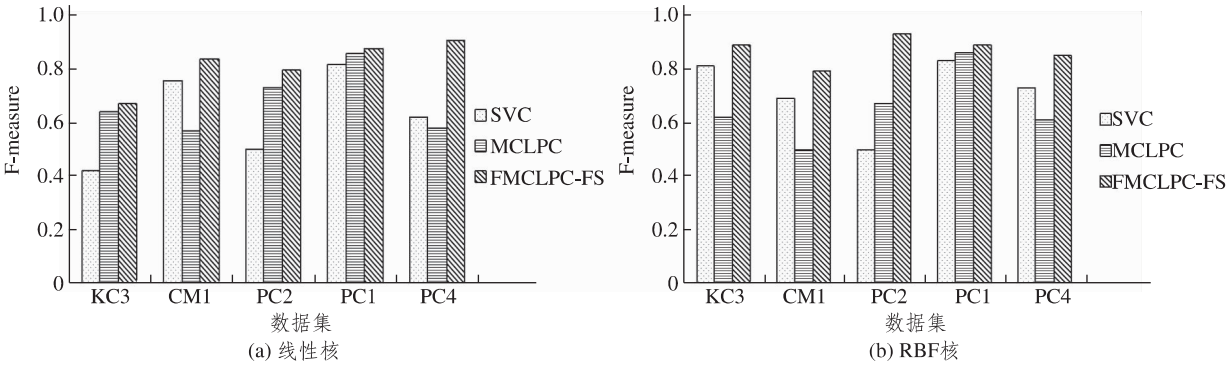

图3 基于不同核的分类器在5个数据集的F-measure比较

Fig.3 The F-measure comparison of classifiers with different kernel functions in 5 datasets

4.2 特征选择结果

在FMCLPC-FS方法的实验中,本文通过设置停止准则来获得最佳核权值,并选择预先确定的阈值对最佳特征子集进行特征选择或降维。对于线性核和RBF核的FMCLPC-FS模型,特征选择结果分别如表3和表4所示。最后一列中括号里的数字表示每个特征在分类过程中的重要程度,正数为正相关,负数为负相关。对于实际应用中的每一个数据集,可以根据特征的重要性提供预测值的原因分析,从而得到可追溯、可解释的结果。

如表3所示,基于线性核的FMCLPC-FS模型的约简率超过49%,其中,数据集PC1的约简率高达82.5%,从40个特征选出了7个最重要的特征。由表4可知,基于RBF核的FMCLPC-FS模型的约简率大于61.5%,其中,数据集CM1的约简率高达72.5%,从40个特征选出了11个最重要的特征。综合来看,基于RBF核的FMCLPC-FS模型的约简率普遍高于基于线性核的FMCLPC-FS模型的约简率。

表3 基于线性核的FMCLPC-FS模型在5个数据集上的特征选择结果

Tab.3 Feature selection results of FMCLPC-FS model based on linear kernel on 5 datasets

数据集	原始特征数	所选特征数	约简率/%	最重要的5个特征(权重)
KC3	40	12	70	LOC comments (0.83) NUM unique operators (-0.38) Design density (0.02) Cyclomatic density (-1.74) Decision count (0.40)
CM1	40	20.4	49	Number of lines (1.00) LOC comments (-0.87) LOC code and comment (-0.72) Halstead length (1.00) NUM operands (1.00)
PC2	40	15.8	60.5	LOC blank (-0.68) LOC comments (-0.69) NUM unique operands (0.72) Modified Condition count (-1.00) Halstead effort (0.98)
PC1	40	7	82.5	LOC total (-1.00) LOC code and comment (0.62) Halstead level (1.00) Error est (-0.71) Halstead difficulty (-0.93)
PC4	40	19.6	51	LOC executable (-1.00) Halstead difficulty (1.00) Edge count (-0.71) Halstead length (0.83) Modified Condition count (-1.00)

(C)1994-2021 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

表 4　基于 RBF 核的 FMCLPC-FS 模型在 5 个数据集上的特征选择结果

Tab.4 Feature selection results of FMCLPC-FS model based on RBF kernel on 5 datasets

数据集	原始特征数	所选特征数	约简率/%	最重要的 5 个特征(权重)
KC3	40	12	70	LOC total (-0.96) Design density (0.98) LOC comments (-1.00) NUM operands (1.00) NUM operators (0.94)
CM1	40	11	72.5	LOC executable (-0.75) Error est (0.94) Decision count (-1.00) Halstead length (-0.78) Edge count (0.99)
PC2	40	15.2	62	Design complexity (-1.00) Error est (-1) Global data density (-0.81) Parameter count (1.00) NUM operands (-0.67)
PC1	40	11.2	72	Number of lines (0.74) NUM operands (0.97) Design complexity (-0.71) Decision count (-0.82) Design density (-0.61)
PC4	40	15.4	61.5	Halstead difficulty (-1.00) Halstead level (-0.89) Global data density (0.65) Decision density (0.78) Parameter count (1.00)

综上所述,本文从每个数据集中自动去除了不重要或不相关的特征,并且识别提取出较少的重要特征,数据集的特征约简率较高,说明 FMCLPC-FS 分类器在特征约简方面具有很大的优势,可以有效地找到最优特征子集,给出重要的可解释结果。

5　结语

本文基于 MCLPC 模型提出了一种新的 FMCLPC-FS 模型。FMCLPC-FS 扩展了 MCLPC 的功能,除了解决分类问题外,还实现了特征选择和降维的双重功能。FMCLPC-FS 为每个特征引入相对应的模糊隶属度值,使用核函数将原始空间映射到新的特征空间,获得每个特性的贡献值和重要性分类,不仅提高了软件缺陷预测的整体性能,而且增强了分类的可解释性。改进的单线性规划分类器可以有效地减少异常值和数据异常以及非线性可分离情况的影响。通过 5 个不同的数据集对 FMCLPC-FS 进行了测试,实验结果和统计分析表明,FMCLPC-FS 方法的预测性能优于 SVC 和 MCLPC,是一种更有效的软件缺陷预测分类器,在其他实际应用中具有很大的潜力。在后续的研究中将进一步修改和完善 FMCLPC-FS 模型和算法,以期在未来将其推广应用于解决多类分类问题。

参考文献:

[1] 宫丽娜,姜淑娟,姜丽.软件缺陷预测技术研究进展[J].软件学报,2019,30(10) : 3090-3114.

[2] 曾路,汪浩.基于机器学习的虚拟仪器软件缺陷预测模型研究[J].自动化与仪器仪表,2020(5) : 59-62.

[3] 李莉,纪欣沅,宋嵩.回环软件缺陷数量预测模型[J/OL].计算机工程与应用: 1-8 [2020-11-15]. http: // kns. cnki. net/kcms/detail/11. 2127. tp. 20201009.1454.006.html.

[4] NAM J,PAN S J,KIM S.Transfer defect learning[C] // International Conference on Software Engineering, 2013: 382-391.

[5] LARADJI I H, ALSHAYED M, GHOUTI L. Software defect prediction using ensemble learning on selected features [J]. Information and Software Technology, 2015,58: 388-402.

[6] ZHANG F, ZHENG Q, ZOU Y, et al. Cross-project defect prediction using a connectivity-based unsupervised classifier [C] // 2016 IEEE/ACM 38th International Conference on Software Engineering(ICSE) , 2016: 309-320.

[7] 李梦奇.基于机器学习的软件缺陷预测方案研究[D].北京: 北京邮电大学,2019.

(C)1994-2021 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

[8] 张志武.基于机器学习的软件缺陷预测方法研究[D].南京:南京邮电大学,2018.
[9] 于巧.基于机器学习的软件缺陷预测方法研究[D].徐州:中国矿业大学,2017.
[10] HE P, LI B, LIU X, et al. An empirical study on software defect prediction with a simplified metric set[J]. Information and Software Technology, 2015, 59: 170-190.
[11] ZHOU Y M, XU B W, LEUNG H, et al. An in-depth study of the potentially confounding effect of class size in fault prediction[J]. ACM Transactions on Software Engineering and Methodology, 2014, 23(1): 1-51.
[12] KHOSHGOFTAAR T M, GAO K, NAPOLITANO A, et al. A comparative study of iterative and non-iterative feature selection techniques for software defect prediction[J]. Information Systems Frontiers, 2014, 16(5): 801-822.
[13] GHOTRA B, MCINTOSH S, HASSAN A E. Revisiting the impact of classification techniques on the performance of defect prediction models[C]// IEEE International Conference on Software Engineering, 2015, 1: 789-800.
[14] GAO K H, KHOSHGOFTAAR T M, WANG H J, et al. Choosing software metrics for defect prediction: an investigation on feature selection techniques[J]. Software: Practice and Experience, 2011, 41(5): 579-606.
[15] TANTITHAMTHAVORN C, MCINTOSH S, HASSAN A E, et al. Automated parameter optimization of classification techniques for defect prediction models[C]// 2016 IEEE/ACM 38th International Conference on Software Engineering(ICSE), 2016: 321-332.
[16] ZHANG Z W, GAO G X, TIAN Y J. Multi-kernel multi-criteria optimization classifier with fuzzification and penalty factors for predicting biological activity[J]. Knowledge-Based Systems, 2015, 89: 301-313.
[17] FREED N, GLOVER F. Simple but powerful goal programming models for discriminant problems[J]. European Journal of Operational Research, 1981, 7(1): 44-60.
[18] GLOVER F. Improved linear programming models for discriminant analysis[J]. Decision Sciences, 1990, 21(4): 771-785.
[19] TIBSHIRANI R. Regression shrinkage and selection via the lasso[J]. Journal of the Royal Statistical Society: Series B (Methodological), 1996, 58(1): 267-288.
[20] SHEPPERD M. NASA IV&V facility metric data program[EB/OL]. (2004-12-02)[2020-10-20]. http://openscience.us/repo/defect/mccabehalsted/.
[21] HALSTEAD M H. Elements of software science[M]. New York: Elsevier, 1977.
[22] MCCABE T J. A complexity measure[J]. IEEE Transactions on Software Engineering, 1976(4): 308-320.

Software Defect Prediction Method Based on Fuzzy Multi-criteria Linear Programming

WU Ruixia[1], ZHANG Zhiwang[1], WANG Yan[1], ZHOU Li[1], YUE Jun[1], LU Tairan[2]

(1.School of Information and Electrical Engineering, Ludong University, Yantai 264039, China;
2.School of Information Science and Electrical Engineering(School of Artificial Intelligence), Jinan 250357, China)

Abstract: In order to improve the testability and credibility of software development process, a new fuzzy multi-criteria linear programming classifier with feature selection FMCLPC-FS was proposed based on the analysis of the characteristics of software defect prediction data. First, a fuzzy membership degree function was defined to deal with the noise and outliers in the original data. Then, the nonlinear separable problem was transformed into the linearly separable or approximate linearly separable problem by using kernel function. In addition, a sparse function was introduced into the MCLPC model to remove redundant features from the dataset and select the least number of important features in the classification process. The experimental results show that compared with MCLPC and SVC, FMCLPC-FS can significantly improve the accuracy of defect prediction and the interpretability of classification.

Keywords: fuzzy membership degree; multi-criteria linear programming; software defect prediction

(责任编辑 李秀芳)

(C)1994-2021 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net