基于改进的非参数方法的软件失效预测模型

王宗会 周 勇 张德平
(南京航空航天大学计算机科学与技术学院 南京 210016)

摘 要 基于主成分分析(PCA)和改进的N-W非参数估计法(INW)提出了一种新的软件失效预测模型。首先,通过对非参数估计的训练样本集进行主成分分析来减少非参数回归估计和预测的输入因子数,再利用PCA计算的方差贡献率作为非参数方法中带宽矩阵的权重,消除各输入因子对结果的作用程度不同所造成的影响,进而建立软件失效预测模型。最后基于一组真实软件失效数据集Eclipse JDT进行实例分析。结果表明,基于改进的非参数方法的软件失效预测模型在预测的精度和稳定性上得到了进一步提高。在后10步的预测范围内,预测值的平均误差为16.2575,均方百分比误差为0.0726。

关键词 软件失效,主成分分析(PCA),N-W非参数估计,带宽
中图法分类号 TP311 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.6.032

Software Failure Prediction Model Based on Improved Nonparametric Method

WANG Zong-hui ZHOU Yong ZHANG De-ping
(College of Computer Science and Technology, Nanjing University of Aeronautics & Astronautics, Nanjing 210016, China)

Abstract Based on principal component analysis (PCA) and improved N-W nonparametric estimation method (INW), a new software failure prediction model was presented. First of all, through the principal component analysis of training sample set of nonparametric estimation, the input number of nonparametric method was reduced. Then the variance contribution ratio of PCA was used as the weight of the bandwidth matrix in nonparametric estimation method, the impact of each imput factor on the results was eliminated in a different extent and software failure prediction models were built. Finally, this paper gave example analysis based on one real software failure data set Eclipse JDT. The results show that the failure prediction model based on improved nonparametric method has made further improvement in prediction precision and stability. Within the forecast range of the last ten steps, the average error of predictive value is 16.2575, and the mean square error is 0.0726.

Keywords Software failure, Principal component analysis (PCA), N-W nonparametric estimation, Bandwidth

1 引言

随着软件技术在航空航天、国防、金融、能源和通信等诸多领域中的广泛应用,软件的可信性日益受到普遍关注。在可信软件研究领域中,软件缺陷和失效的预测技术是最为关键的研究问题之一。软件缺陷一般是指软件代码中的错误,而软件失效是指在软件运行期间,缺陷在特定条件下导致整个软件系统对其所要求行为的偏离现象。软件缺陷预测的主要目的是预测软件还存留的缺陷,根据软件的基本属性(规模、复杂度、开发方法、过程等)、软件已经发现的缺陷来预测软件可能还遗留的、尚未发现的缺陷。合理预测软件缺陷可以统计尚未发现但仍存在的软件缺陷数目及软件缺陷的分布,这样可以有效地帮助测试人员快速准确地定位并纠正软件缺陷,客观评价软件测试结果,可显著减少软件开发成本和提高软件可信性,为保证和提高软件质量起着非常重要的作用。

软件缺陷预测技术作为保证软件质量的一种重要手段,其研究在学术界与业界已得到广泛关注[1,2,4]。一般地,软件缺陷预测技术大体上可分为静态和动态两种,静态预测技术主要是指根据缺陷相关的度量数据,对缺陷的数量或分布进行预测[1];而动态预测技术则是基于缺陷或者失效产生的时间,对系统缺陷随时间的分布进行预测。这些缺陷预测技术依靠不同的软件度量指标(如软件规模、复杂度、内聚性、耦合度、继承深度等)[2]、软件技术(如OO,Web)、软件过程和执行过程等关联的度量指标,借助基于分类、回归分析与机器学习等技术的一些静态分析工具[3]对软件缺陷进行预测。比较典型的缺陷预测模型[1,4]如基于软件规模、软件复杂度、多维软件度量元等因素的缺陷预测模型,通过研究软件缺陷与代码行数、文档数量、复杂度等基本属性之间的关系预测软件可能存在的缺陷数,这些模型的实质都是通过基于度量指标与软件缺陷之间的关系进行分析(如相关性分析[10]、相依性[11]、一致性[12]、Bayesian因果关系[13]等),用数学描述(转换函数)的形式确立输入输出变量之间的一般关系,进而构建相应的预测模型进行软件缺陷预测。

到稿日期:2015-05-20 返修日期:2015-08-28 本文受国防科工局十二五重大基础科研项目(c0420110005)资助。
王宗会(1990-),男,硕士生,主要研究方向为软件测试与软件可靠性建模,E-mail:852118094@qq.com;周 勇(1975-),男,副教授,主要研究方向为人工智能、专家系统等;张德平(1973-),男,博士,主要研究方向为软件测试与软件可靠性建模。

在软件缺陷预测技术中，参数方法是经常使用的一类预测技术，但传统的参数方法过于依赖总体分布的假定，在许多实际问题中，当解释变量局限于一定的范围时，通常我们对其可能的模型认识并不十分清楚，或者因实验的误差导致所得数据并不完全可靠甚至有错误，这就使得在此种条件下用数据拟合模型会导致模型与现实相背离，产生模型偏差。而非参数方法完全从数据本身获得所需的信息，无需对总体分布强加假定条件，可以选择与数据最为匹配的模型，从而纠正了传统参数方法可能导致的模型偏差。近年来，Dharmasena等[5]利用非参数回归方法估计和预测软件的可靠性增长曲线，主要针对一维的软件失效数据进行了回归估计。Couto等[6,7]研究了软件缺陷与软件度量指标之间的因果关系，研究结果表明，在由 D'Ambros[8] 提供的 bug 数据集中，大约有64%～93%的软件缺陷与软件度量指标之间存在因果关系，由于仅适用于研究两个序列之间的因果关系，其默认假设是不存在其它因素(时间序列)的影响，但实际影响软件缺陷的因素远超过两个。Hwang[9]研究了非参数多变量回归估计技术，并针对不同的核函数以及不同的变量维度进行了多组基于实际数据的对比试验，但试验数据的维度仅限于 1 维到 5 维。由于影响软件缺陷的因素很多，若将所列举的影响因素全部作为输入量纳入样本集之内，则势必导致样本集维数的膨胀，进而影响预测模型的精度和稳定性；并且以上所采用的非参数方法均未考虑到各变量对结果作用程度不同的问题，这也会直接影响到最终估计和预测数据的精度。

针对以上问题，利用数据预处理中降低维度的主成分分析(PCA)技术对非参数估计的训练样本集进行主成分分析，以减少回归估计和预测的输入因子数；再利用 PCA 计算的方差贡献率作为由 AMISE 经验算法求得的带宽矩阵的权重，在一定程度上消除了各输入因子对结果的作用程度不同造成的影响，进一步提高了回归估计和预测的精度以及稳定性。

2 基于非参数方法的软件失效预测模型

在软件失效预测技术中，基于非参数方法的软件失效预测也是广泛使用的一类预测技术。尤其针对多种失效因素影响的软件失效预测，由于无法确定数据的拟合模型，因此传统的参数方法已经不足以建立模型。而非参数方法的回归函数形式可以任意，没有任何约束，解释变量和被解释变量的分布也少有限制，因而适应性较强。因此对诸多因素引起的软件失效的预测，非参数回归模型有更好的拟合效果。

影响软件缺陷的诸多因素所构成的数据之间一般具有较强的相关性：可设 X 表示 k 个影响软件缺陷因素所构成的向量变量。令 $X^k=(x_1,x_2,\cdots,x_k)^{\mathrm{T}}$，若已知向量 X^k 的 n 个样本 $X_i^k=(y_{1i},y_{2i},\cdots,y_{ki})^{\mathrm{T}}$，$i=1,2,\cdots,n$，则向量 X^k 联合概率密度函数 $f(X^k)$ 的核密度估计定义为：

$$\hat{f}(X^k)=\frac{1}{n}\sum_{i=1}^{n}\frac{1}{h_1h_2\cdots h_k}\prod_{j=1}^{k}K\left(\frac{x_j-y_{ji}}{h_j}\right) \tag{1}$$

h_j 是第 j 个变量的平滑系数，也称为带宽或窗宽，表征了核函数在样本点附近的作用范围；$K(\cdot)$ 称为窗函数或核函数。$\hat{f}_h$ 可以继承核函数 $K(\cdot)$ 的可微性和连续性，因此当选用高斯函数作为核函数时，$\hat{f}_h$ 可以进行任意阶微分。

上式提供了 k 个影响软件失效的因素所构成的联合概率密度分布，在短期预测中，通常前 $k-1$ 个影响软件缺陷数的因素所构成的数据集已知，视为自变量，希望由此得到第 k 个预测值 x_k，即软件的故障数。这实际上是一个求解条件概率密度的问题。即：

$$f(x_k\mid(x_1,x_2,\cdots,x_{k-1}))=\frac{f(X^k)}{f(X^{k-1})}=\frac{f(X^k)}{\int_{-\infty}^{\infty}f(X^k)\mathrm{d}x_k} \tag{2}$$

条件密度的一个优点是它具有更多的信息，不仅是对预测值，而且还指出了预测误差的可能大小。根据上式，可以求得 x_k 的期望值，并可以此作为第 k 个时刻的预测值：

$$E(x_k)=\int x_k f(x_k\mid(x_1,x_2,\cdots,x_{k-1}))\mathrm{d}x_k=\frac{\int x_k f(X^k)\mathrm{d}x_k}{\int f(X^k)\mathrm{d}x_k} \tag{3}$$

采用 N-W 非参数估计法，经简化计算后可得下面的回归函数的近似表达式：

$$\hat{m}_{NW}(x)=E(x)=\frac{\sum_{i=1}^{n}\{y_i\prod_{j=1}^{k-1}K_{h_j}(x_j-y_{ji})\}}{\sum_{i=1}^{n}\{\prod_{j=1}^{k-1}K_{h_j}(x_j-y_{ji})\}} \tag{4}$$

其中，$K_{h_j}(x_j-y_{ji})=h_j^{-1}K\{(x_j-y_{ji})/h_j\}$，当选取高斯核 $K(x)=(2\pi)^{-1/2}\exp(-x^2/2)$ 时，式(5)即为非参数核密度估计的软件失效预测模型表达式：

$$\hat{m}_{NW}(x)=\frac{\sum_{i=1}^{n}\{y_i\exp(-\sum_{j=1}^{k-1}\frac{(x_j-y_{ji})^2}{2h_j^2})\}}{\sum_{i=1}^{n}\{\exp(-\sum_{j=1}^{k-1}\frac{(x_j-y_{ji})^2}{2h_j^2})\}} \tag{5}$$

非参数方法中大多涉及核函数带宽的估计。研究表明，在非参数估计过程中，不同核函数的选取对估计的结果影响并不大，而核函数带宽的选择决定最终的结果偏差。因此带宽的选择方法成为非参数估计过程中的关键。

3 基于改进的非参数方法的软件失效预测模型

3.1 基于 PCA 的 N-W 非参数估计方法

在基于多变量的非参数软件失效预测模型中，影响软件缺陷的诸多因素所构成的数据之间一般具有较强的相关性。而主成分分析(PCA)是一种常用的特征提取方法，这种技术可以在保证信息损失最少的前提下，一方面降低高维数据的维数，另一方面消除各维数据特征向量的相关性，从一定意义上体现了特征向量中不同维度对识别结果贡献大小的不同。

另外，由非参数回归估计中变带宽核密度估计[15]的理论可知，若掌握解释变量分布的一些信息或者各变量对结果的作用程度等信息，对各变量的带宽选择进行加权处理，会在一定程度上提高估计和预测的效率。本文则利用 PCA 计算的方差贡献率作为 N-W 非参数估计方法中带宽矩阵的权重，消除解释变量对结果的作用程度不同所造成的影响。改进后的模型在预测的精度和稳定性上将能够得到进一步提高。

首先，利用主成分分析法计算各维度的方差贡献率以及每个数据的主成分得分。主成分分析法是对多个样本的输入变量形成的数矩阵求取相关矩阵，根据相关系数矩阵的特征值，获得累计方差贡献率，在此基础计算特征值对应的特征向量，最后确定主成分。具体的计算分为 5 个步骤。

(1)原始数据标准化

由于协方差矩阵易受指标的量纲及数量级的影响，因此需要对原始数据进行标准化，即

$$x_j^* = \frac{x_j - E(x_j)}{\sqrt{Var(x_j)}} (j=1,2,\cdots,p) \tag{6}$$

其中，x_j 为原始软件失效数据集矩阵 $X_{n\times p}$ 每列的数值；n 为样本数据的组数；p 为待评价指标的个数；$E(x_j)$ 和 $Var(x_j)$ 分别表示 x_j 的均值和方差。

(2)计算样本相关系数矩阵

$R=(r_{ij})_{p\times p} r_{ij}(i,j=1,2,\cdots,p)$ 为变量 x_i 和 x_j 的相关系数。$r_{ij}=r_{ji}$，其计算公式为：

$$r_{ij} = \frac{\sum_{k=1}^{n}(x_{ki}-\bar{x_i})(x_{kj}-\bar{x_j})}{\sqrt{\sum_{k=1}^{n}(x_{ki}-\bar{x_i})^2 \sum_{k=1}^{n}(x_{kj}-\bar{x_j})^2}} \tag{7}$$

(3)计算特征值与特征向量

首先通过求解特征方程 $|\lambda I-R|=0$ 得出特征值 $\lambda_i(i=1,2,\cdots,p)$，并将其按大小顺序排列，即 $\lambda_1\geqslant\lambda_2\geqslant\cdots\geqslant\lambda_p\geqslant 0$；然后分别求出对应的特征值 λ_i 的特征向量 $e_i(i=1,2,\cdots,p)$。

(4)计算主成分贡献率和累计贡献率

主成分贡献率(%)为：

$$\beta_i = \left(\frac{\lambda_i}{\sum_{i=1}^{p}\lambda_i}\right)\times 100\% \tag{8}$$

累计贡献率(%)为：

$$\beta_\Sigma(k) = \sum_{i=1}^{k}\beta_i \tag{9}$$

选取主成分的个数取决于累计贡献率。通常累计贡献率大于 85%～90%，对应的前 k 个主成分便包含 p 个原始变量所能提供的绝大部分信息，则主成分个数就是 k。

(5)计算主成分载荷

$$l_{ij} = \sqrt{\lambda_j} e_{ij} (i=1,2,\cdots,p, j=1,2,\cdots,k) \tag{10}$$

由此进一步计算主成分得分为：

$$Z=(z_{ij})_{nk} \tag{11}$$

然后，对软件失效数据集进行主成分分析，可计算得到主成分方差贡献率 $\beta=[\beta_1,\beta_2,\cdots,\beta_n]^T$ 以及累计贡献率 $\beta_\Sigma(k)$，在决定主成分的个数时，应该在 $\beta_\Sigma(k)\geqslant 0.95$ 的条件下尽可能地减少主成分的个数，在此假设当 k 取某一特定值时，$\beta_\Sigma(k)\geqslant 0.95$，则可用前 k 个主成分来代替数据集中 p 个原始变量的信息；且记 $W=[\beta_1,\beta_2,\cdots,\beta_k]^T, \beta_i\in\beta(i=1,2,\cdots,k)$，$\beta_i$ 为降维后各变量的方差贡献率。对 W 进行归一化处理，得到 $W'=[\beta_1',\beta_2',\cdots,\beta_k']^T$，其中 $\beta_1'+\beta_2'+\cdots+\beta_k'=1$。以 β_i' 作为多变量非参数核密度估计中的多带宽矩阵的权重系数，即 $H=diag[h_1'^2,h_2'^2,\cdots,h_k'^2], h_i'=h_i*\beta_i', i=1,2,\cdots,k$，故而最终的加权多带宽核密度估计的预测表达式为：

$$\hat{m}_{NW}(x) = \frac{\sum_{i=1}^{n}\{y_i \exp(-\sum_{j=1}^{k}\frac{(x_j-y_{ji})^2}{2h_j'^2})\}}{\sum_{i=1}^{n}\{\exp(-\sum_{j=1}^{k}\frac{(x_j-y_{ji})^2}{2h_j'^2})\}} \tag{12}$$

其中，h_i 由 AMISE 的经验算法求得，即采用正态参考准则，以 d 维正态分布 $N_d(u,\Sigma)$ 作为参考分布(u 表示样本均值向量，Σ 表示样本协方差矩阵)，并使用 d 维正态分布 $N_d(0,I_d)$ 作为核函数，由此推导出使 AMISE 取极小值的最优对角带宽矩阵：

$$h_i = \sigma_i\{4/[(d+2)n]\}^{1/(d+4)} (i=1,2,\cdots,d) \tag{13}$$

其中，σ_i 为第 i 个历史数据的样本标准差。

3.2 基于 PCA-INW 的软件失效预测模型

基于 PCA-INW 的软件失效预测分析分为 3 个阶段：初步定性分析、PCA-INW 非参数方法的回归估计与软件失效预测，其流程如图 1 所示。

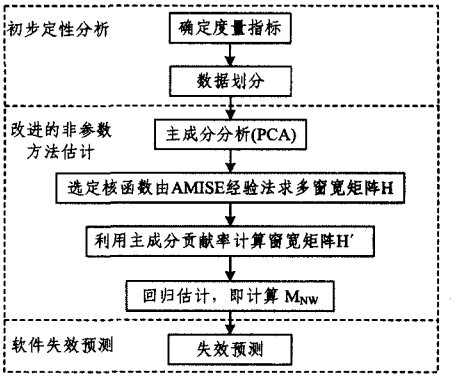

图 1 基于 PCA-INW 的软件失效预测模型

初步定性分析主要包括选择确定进行软件失效预测的各个软件度量指标、归一化处理数据集样本并进行数据划分。首先，影响软件失效的因素很多，可以出现在整个软件生命周期的各个阶段，主要包括外部因素与软件本身内部因素。根据专家知识、经验以及相关研究确定可能影响软件失效(结果变量)Y 的软件度量指标(因素)集合 $\{X_1,X_2,\cdots,X_k\}$。然后将数据样本集(N 个数据样本)分为训练集 A(training set)和检测集 B(testing set)($N_W=N_A+N_B, W=A\cup B$)。若建立预测模型，则将数据样本集分为学习集 A、检测集 B 和预测集 C(checking set)，$N_W=N_A+N_B+N_C, W=A\cup B\cup C$。

划分数据后，首先对软件失效数据中的训练集 A 进行主成分分析，得出各维数据的方差贡献率 β，求出累计贡献率 $\beta_\Sigma(k)$，并求得每个数据的主成分得分。由 $\beta_\Sigma(k)\geqslant 0.95$ 的条件确定 k 个主成分来代替数据集中 p 个原始变量的信息，然后选定非参数方法中的核函数(这里选择高斯核函数)，并根据已有的数据集，利用 AMISE 经验法求得多变量非参数方法中 k 维变量的带宽矩阵 H。将主成分分析中获得的 k 维数据的方差贡献率 $W=[\beta_1,\beta_2,\cdots,\beta_k]^T$ 进行归一化处理后，得到 $W'=[\beta_1',\beta_2',\cdots,\beta_k']^T$，以 β_i' 作为带宽矩阵 H 的权重系数而形成新的带宽矩阵 H'。

最后利用 PCA-INW 非参数方法与软件失效有因果关系的软件度量指标建立软件失效预测模型，将训练集 A 利用主成分分析降维后的数据输入改进后的模型进行回归训练，利用检测集 B 和 C 分别进行失效估计与预测。

4 实验分析

本文实验部分选取一组真实的软件失效数据集 Eclipse JDT[14] 对改进后的非参数方法进行短期预测验证，其中，数据集 Eclipse JDT 包含 182 个失效观察值。观察值一共包含 8 个测量指标，分别是 fanIn、fanOut、lackOfCohesionInMethod、numberOfAttributes、numberOfLinesOfCode、numberOfMethods、weightedMethodCount 以及 defects，本文利用 defects 作为因变量来评估软件失效的预测效果。观察值为一

时间序列，记录了当前观察与上次观察的各测量指标缺陷数的增减情况。数据集保留最后 10 个数据观察值作为验证集，其余数据作为训练集。

在实验中采用以下两种性能评价指标进行比较。

(1)均值误差(average error，AE)

$$AE=\frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i-y_i'}{y_i}\right|\times 100 \tag{14}$$

(2)均方百分比误差(MSPE)

$$MSPE=\frac{1}{n}\sqrt{\sum_{i=1}^{n}\left(\frac{y_i-y_i'}{y_i}\right)^2} \tag{15}$$

以上各式中，y_i 表示数据的实际值，y_i'表示数据的预测值。显然，AE 值和 $MSPE$ 值越小，则表明预测值与实际值越接近，模型拟合或预测性能越好。

本文选取的实验数据是 Eclipse JDT 数据集，将各测量指标数据进行归一化处理，其中 4 个测量指标如图 2 所示。

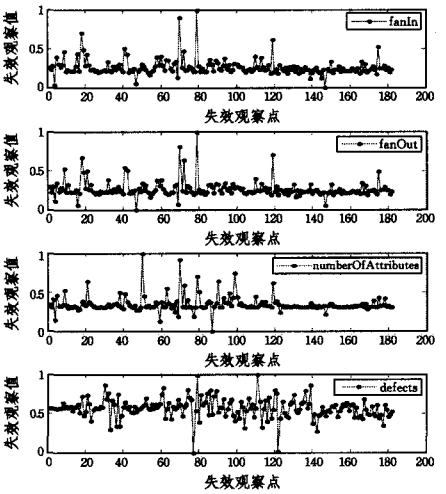

图 2　归一化处理后的 Eclipse JDT 数据集

将测量指标 defects 作为因变量，其余 7 个测量指标作为自变量，对原始输入矩阵 $X_{182\times 7}$ 进行主成分分析。1)对输入变量进行标准化处理后计算相关系数矩阵；2)由相关系数矩阵计算特征值以及各个主成分的贡献率，具体如表 1 所列。

表 1　特征值及主成分贡献率

主成分	特征值	贡献率/%	累计贡献率/%
1	5.5766	79.6658	79.6658
2	0.5573	7.9621	87.6280
3	0.4713	6.7333	94.3613
4	0.2018	2.8830	97.2443
5	0.1161	1.6582	98.9025
6	0.0520	0.7426	99.6451
7	0.0248	0.3549	100

根据表 1 可知，前 4 个主成分的累计贡献率已高达 97.99%(大于 95%)，说明前 4 个主成分提供了原始数据比较充足的信息，因此可用 4 个互不相关的新变量来代替原有的 7 个变量。对于 4 个特征值分别求出其特征向量，再计算各变量在主成分上的载荷，进而求出各主成分的得分，构成新的样本空间。

将新样本空间中的前 172 个数据作为训练数据直接输入非参数估计模型中。图 3、图 4 为改进前与改进后针对训练数据集进行非参数回归的数据图。

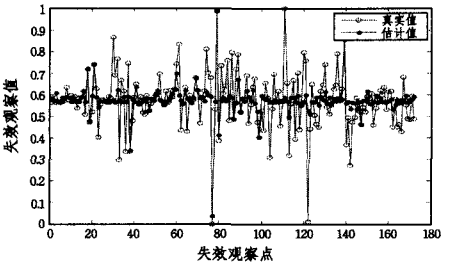

图 3　JDT 训练数据集在 N-W 非参数估计模型中估计值与真实值的对比

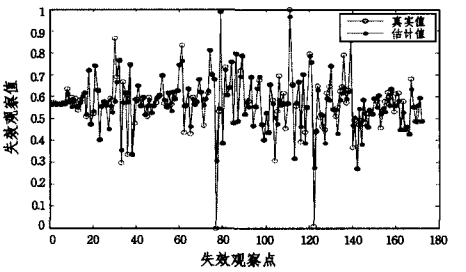

图 4　JDT 训练数据集在 PCA-INW 非参数估计模型中估计值与真实值的对比

由 JDT 训练数据集在 N-W 非参数估计模型中以及在 PCA-INW 非参数估计模型中估计值与真实值的数据对比可以看到，改进后的非参数方法在回归估计中的回归值更加趋近真实值，并在提高回归精确度的同时确保了不造成训练数据的过度拟合。

数据训练完成后，利用训练后的模型对后 10 个数据进行预测。图 5 分别给出了 N-W 非参数模型与 PCA-INW 非参数模型在后 10 个数据预测值上的误差百分比。

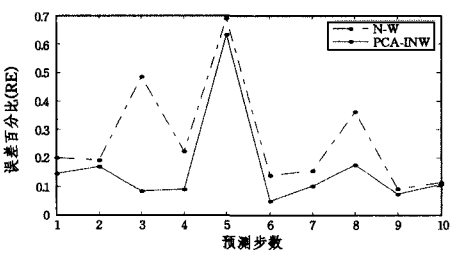

图 5　JDT 数据集在 N-W/PCA-INW 模型中的预测误差百分比

表 2 分别给出了利用 N-W 非参数模型与 PCA-INW 非参数模型对 JDT 数据进行预测的相应评价指标值。

表 2　JDT 数据集在各模型的预测性能指标

预测性能指标	模型	
	N-W	PCA-INW
AE	26.5296	16.2575
MSPE	0.1017	0.0726

由图 5 的预测误差百分比对比以及表 2 的预测性能指标值可以看出，改进非参数方法后的软件失效预测模型在后 10 步的预测范围内，预测值的平均误差比未改进前的预测值误差要小得多，说明预测的精度较改进前模型的预测精度得到了进一步的提高；另外，均方百分比误差 MSPE 为0.0726，说明改进后的模型在预测的稳定性上也要优于改进前的模型。

结束语　本文基于主成分分析(PCA)与改进的 N-W 非参数方法提出了一种软件失效预测模型，借鉴数据预处理中

(下转第 178 页)

增强了实用效果。

结束语 本文提出了业务流程模型转换规则算法、执行轨迹到 Prolog 的自动转换算法以及 Prolog 对于时间约束的验证规则来验证分析带有时间约束的 BPMN。将模型转换为执行轨迹的集合后利用 Prolog 语言直接对其时间约束进行检验，实现了业务流程模型的自动化验证。

参 考 文 献

[1] Fan Yu-shun, Wu Cheng. Research on workflow modeling to improve system flexibility[J]. Journal of Software, 2002, 13(4): 833-839(in Cinese)
范玉顺，吴澄. 一种提高系统柔性的工作流建模方法研究[J]. 软件学报，2002，13(4)：833-839

[2] Van Der Aalst W, Van Hee K M. Workflow management: models, methods, and systems[M]. MIT press, 2004: 30-150

[3] Aalst V D, Wil M P. Business Process Management: A Comprehensive Survey[J]. Isrn Software Engineering, 2012, 2013(2): 125-143

[4] Börger E. Approaches to modeling business processes: a critical analysis of BPMN, workflow patterns and YAWL [J]. Software & Systems Modeling, 2012, 11(3): 305-318

[5] Wu N Q, Zhou M C. Modeling, analysis and control of dual-arm cluster tools with residency time constraint and activity time variation based on Petri nets[J]. IEEE Transactions on Automation Science and Engineering, 2012, 9(2): 446-454

[6] L Ye-bai, M Fu-qi. Research of the verification in workflow process modeling on the application of Petri nets[C]// International Conference on e-Education, e-Business, e-Management, and e-Learning, 2010(IC4E'10). IEEE, 2010: 21-24

[7] Szpyrka M, Nalepa G J, Ligęza A, et al. Proposal of formal verification of selected BPMN models with Alvis modeling language [M] // Intelligent Distributed Computing V. Springer Berlin Heidelberg, 2012: 249-255

[8] Lanz A, Weber B, Reichert M. Time patterns for process-aware information systems [J]. Requirements Engineering, 2014, 19 (2): 113-141

[9] Zhou N F. The language features and architecture of B-Prolog [J]. Theory and Practice of Logic Programming, 2012, 12(1/2): 189-218

[10] Combi C, Gozzi M, Posenato R, et al. Conceptual modeling of flexible temporal workflows[J]. ACM Transactions on Autonomous and Adaptive Systems (TAAS), 2012, 7(2): 451-457

(上接第 159 页)

降低数据维度的思想，通过对非参数估计的训练样本集进行主成分分析来改进样本的输入因子数；再利用主成分分析中的方差贡献率作为由经验算法求得的带宽矩阵的权重，改进了传统的非参数估计法，建立了一种新的软件失效预测模型。模型从复杂系统建模的角度研究了变量间的关系，其优点在于对于单输出(结果)多维系统，不需诸多限制条件，可以检验变量之间在非线性意义上的因果关系。实验结果表明：该方法在预测时，一定程度上消除了各输入因子对结果的作用程度的不同所造成的影响，在预测的精度和稳定性上得到了进一步提高。

本模型由于使用改进后的非参数方法进行训练，因此数据的训练集应尽可能大，以使模型得到充分训练，这样才能达到更好的预测效果。另外，本文算法是进行单步预测，每预测一步后，再把其加入训练集来预测下一个点，随着预测点数的增加，导致在后面点数的预测中误差会越来越大，所以此模型更适合于进行短期预测。

参 考 文 献

[1] Wang Q, Wu S J, Li M S. Software defect prediction[J]. Journal of Software, 2008, 19(7): 1565-1580(in Chinese)
王青，伍书剑，李明树. 软件缺陷预测技术[J]. 软件学报，2008，19(7)：1565-1580

[2] Nagappan N, Ball T, Zeller A. Mining metrics to predict component failures[C]// 28th International Conference on Software Engineering (ICSE). 2006: 452-461

[3] Liu Ya-nan, Wei Zhi-nong, Zhong Lin-juan, et al. Study on the forecasting model of power supply reliability based on PCA and RVM[J]. Power System Protection and Control, 2012, 40(20): 101-105(in Chinese)
刘亚南，卫志农，钟淋涓，等. 基于 PCA 和 RVM 的电网供电可靠性预测模型研究[J]. 电力系统保护与控制，2012，40(20)：101-105

[4] Catal C. Software fault prediction: A literature review and current trends[J]. Expert Systems with Applications, 2011, 38(4): 4626-4636

[5] Sandamali Dharmasena L, Zeephongsekul P. Fitting software reliability growth curves using nonparametric regression methods [J]. Statistical Methodology, 2010, 7(2): 109-120

[6] Couto C, Montandon J E, Silva C, et al. Static correspondence and correlation between field defects and warnings reported by a bug finding tool[J]. Software Quality Journal, 2013, 21(2): 241-257

[7] Catal C. Software fault prediction: A literature review and current trends[J]. Expert Systems with Applications, 2011, 38(4): 4626-4636

[8] D'Ambros M, Lanza M, Robbes R. An extensive comparison of bug prediction approaches[C]// 7th IEEE Working Conference on Mining Software Repositories (MSR). 2010: 31-41

[9] Hwang J N, Lay S R, Lippman A. Nonparametric multivariate density estimation: a comparative study[J]. IEEE Transactions on Signal Processing, 1994, 42(10): 2795-2810

[10] Couto C, Montandon J E, Silva C. Static correspondence and correlation between field defects and warnings reported by a bug finding tool[J]. Software Quality Journal, 2013, 21(2): 241-257

[11] Nagappn N, Ball T. Using software dependencies and churn metrics to predict field failures: an empirical case study[C]// First International Symposisum on Empirical Software Engineering and Measurement. 2007: 364-373

[12] Lee H J, Naish L, Ramamohanarao K. Study of the relationship of bug consistency with respect to performance of spectra metrics [C]// 2nd IEEE International Conference on Computer Science and Information Technology. 2009: 501-508

[13] Okutan A, Yildiz O T. Software defect prediction using Bayesian networks[J]. Empir Software Eng, 2014, 19(1): 154-181

[14] Couto C, Piresa P. Predicting software defects with causality tests[J]. Journal of Systems and Software, 2014, 93: 154-181

[15] 叶阿忠. 非参数和半参数计量经济模型理论[M]. 北京：科学出版社，2008：30-150