



相关向量机软件可靠性建模中失效数据选择

杨小明¹, 楼俊钢^{1,2}, 沈张果¹, 胡文军¹

(1. 湖州师范学院信息工程学院 湖州 313000;

2. 浙江大学控制科学与工程学系 杭州 310058)

摘要: 基于核函数的软件可靠性模型一般对软件失效时间数据与发生在其之前的 m 次失效时间数据的关系进行建模, 着重研究了 m 取值不同时, 其对核函数可靠性模型预测能力的影响。在 5 个不同类型失效数据集上, 采用 Mann-Kendall 检验观测到 m 值增大时模型预测能力逐渐下降, 说明现时失效时间数据能比较久之前观测的失效时间数据更好地用于预测未来, 通过把 m 的取值划分成几个区间, 运用配对 T 检验进行实验研究, 结果表明当 $m \in \{6, 7, 8, 9, 10\}$ 时, 模型能够得到最好的预测性能。

关键词: 软件可靠性预测模型; 相关向量机; 核函数; 软件失效数据; Mann-Kendall 检验

doi: 10.11959/j.issn.1000-0801.2015192

Selection of Failure Data in Software Reliability Modeling Based on RVM

Yang Xiaoming¹, Lou Jungang^{1,2}, Shen Zhangguo¹, Hu Wenjun¹

(1. School of Information Engineering, Huzhou University, Huzhou 313000, China;

2. Department of Control Science and Engineering, Zhejiang University, Hangzhou 310058, China)

Abstract: The high complexity of software is the major contributing factor of software reliability problems, and traditional parametric models may exhibit different predictive capabilities among different software projects, it is hard to select a suitable model for every software projects. Compared to traditional models, kernel based models could achieve better prediction accuracy, and had arouse the interesting of many researchers. The RVM learning scheme was applied to model the failure time data so as to capture the inner correlation between software failure time data and the m nearest failure time data. In addition, the trend of predictive accuracy with the varying of m was detected by way of Mann-Kendall test method. Thereupon, the reasonable value range of m was achieved, thus $m \in \{6, 7, 8, 9, 10\}$ through paired T -test in 5 common used software failure data.

Key words: software reliability predicting model, relevance vector machine, kernel function, software failure data, Mann-Kendall test

收稿日期: 2015-04-21; 修回日期: 2015-07-16

基金项目: 国家自然科学基金资助项目 (No. 61103051), 浙江省自然科学基金资助项目 (No. LY15F020018), 浙江省科技计划公益项目 (No. 2015C33247)

Foundation Items: The National Natural Science Foundation of China (No. 61103051), Zhejiang Provincial Natural Science Foundation of China (No. LY15F020018), Zhejiang Provincial Science and Technology Plan of China (No. 2015C33247)

论文引用格式: 杨小明, 楼俊钢, 沈张果等. 相关向量机软件可靠性建模中失效数据选择. 电信科学, 2015192

Yang X M, Lou J G, Shen Z G, *et al.* Selection of failure data in software reliability modeling based on RVM. Telecommunications Science, 2015192



1 引言

软件可靠性模型是指为预测软件可靠性,利用已有失效数据,根据对软件失效行为的假设,采用一定数学方法建立软件可靠模型的过程^[1]。近年来,研究者提出了许多软件可靠性模型,这些模型都是基于一些对软件的开发环境和使用环境的关键假设,利用测试获得的软件失效信息,对软件系统的失效过程进行建模,评估软件系统的可靠性,预测软件实际工作时的现场行为^[2]。软件可靠性是最重要的可信性属性之一,而软件可靠性建模是目前提高软件可靠性水平主要工具之一。软件可靠性模型多是基于一些对软件的开发环境和使用环境的关键假设,利用测试获得的软件失效信息,对软件系统的失效过程进行建模,评估软件系统的可靠性水平,预测软件实际工作时的现场行为。随机过程模型是软件可靠性建模研究中最广泛使用的,也是实际项目中应用最广泛的一类,目前这类模型的绝大部分研究工作集中于模型的统一、对非齐次泊松过程类(non-homogeneous Poisson process, NHPP)等模型的改进、可靠性成本模型等^[3-8],模型改进的出发点是提出更加合理的假设以提高模型预测精度,如考虑测试环境与实际运行环境的差别、考虑故障的相关性、不完美调试以及测试者学习能力、测试效用函数、测试工作量与覆盖率等。此外,也有学者提出采用马尔可夫更新过程、广义 Pareto 分析、排队论模型、顺序统计量模型、多维随机过程等来描述软件失效过程,取得了很多不错的研究成果^[9-11]。接着,人们尝试利用更多更复杂的数学方法对软件失效行为进行建模及分析,如人工神经网络、未确知理论、混沌理论、时间序列、模糊数学理论、遗传编程、强化学习^[12-15]等,相比随机过程模型,这些方法可以取得更好的预测性能或拟合效果,但在实际软件系统可靠性评估的应用还较为少见。最近,研究者开始考虑核函数方法在软件可靠性建模方面的研究,最主要的包括支持向量机(support vector machines, SVM)、相关向量机(relevance vector machine, RVM)等方法的应用,核函数方法一般对软件失效时刻与在它之前的发生 m 个失效时刻数据之间的对应关系进行建模。Tian 等把 SVM 引入软件可靠性建模中^[16],Xing 等^[17]、Yang 等^[18]、Yuan 等^[19]、Park 等^[20]对 SVM 用于可靠性建模进行了适当的改进。楼俊钢等人使用 RVM 等核函数技术在软件可靠性建模方面做了一些研究工作^[21,22]。与其他方法相比,SVM、RVM 等基于核函数模型具有自适应能力和学习功能,通过归纳学

习和训练,能发现数据输入与输出之间的关系,经过自适应调整求取问题的解,适用于系统开发环境较复杂,对问题的机理不能用数学模型表示的系统,对大量原始数据的处理往往表现出极大的灵活性和自适应性,容错和抗干扰能力较强,在软件可靠性建模问题上也得到了较好地应用,在模型适用性以及评估预测能力上均有较好的表现,是目前软件可靠性模型研究中较为重要的一个突破口。笔者在使用核函数理论进行软件可靠性预测的前期研究中,发现用于建模的软件失效数据数量的改变对模型的预测性能和适用性有较大影响^[22]。然而目前的研究工作中,还很少有专门针对软件失效数据数量选取的工作,本文应用 Mann-Kendall 及配对 T -检验等统计方法在 5 个常用软件失效数据集上对基于相关向量机的软件可靠性预测模型中 m 值选取问题进行研究。

2 基于相关向量机的软件可靠性建模

假设已发生的软件失效时间为 $t_1, t_2, \dots, t_n$, 基于相关向量机的软件可靠性预测问题可以描述为:从已知失效时间间隔数据序列 $\{t_i\}_{i=1}^n$ 预测未知的软件失效时间间隔数据 t_{i+d} , 使用核函数对软件失效时间与在它之前的 m 个失效时间数据建模,以此捕捉失效时间内在的依赖关系,一般地,令 $t_i = f(t_{i-m}, t_{i-m+1}, \dots, t_{i-1})$, 则 t_i 服从固定但未知的条件分布函数 $F(t_i | t_{i-m}, t_{i-m+1}, \dots, t_{i-1})$ 。假设用于学习的失效时间序列为 $t_1, t_2, \dots, t_k (k > m)$, 则在 $t_1, t_2, \dots, t_k$ 已知条件下对 t_{k+1} 进行预测变为:已知 $k-m$ 个观测 $(T_1, t_{m+1}), (T_2, t_{m+2}), \dots, (T_{k-m}, t_k)$ 和第 $k-m+1$ 个输入 T_{k-m+1} 的情况下,估计第 $k-m+1$ 个输出值 $\hat{t}_{k+1}$, 其中 T_i 表示 m 维向量 $[t_i, t_{i+1}, \dots, t_{i+m}]$ 。把 $[t_2, t_{i+1}, \dots, \hat{t}_{k+i}]$ 作为输入,则可以预测 $\hat{t}_{k+2}$, 同理可以预测得到 $\hat{t}_{k+3}, \dots, \hat{t}_{k+d}, \dots$ 。对可以获得的系统输出数据即失效数据进行分析,将其蕴涵的系统失效特征用核函数进行学习并表达出来,构造软件故障过程的等价系统,从而完成对软件系统失效行为的刻画,并依据所建模型完成对系统未来失效行为的预测。对基于相关向量机的软件可靠性模型的详细描述可参见参考文献^[13,14]。

预测的具体实现步骤如图 1 所示。具体介绍如下。

2.1 使用的数据集和性能比较标准

采用 10 个来自不同类型软件的失效数据集^[23]对 m 不同时模型预测性能进行实验分析,见表 1,在实验过程中,取所有数据集的前 2/3 作为学习数据,对后面 1/3 数据进

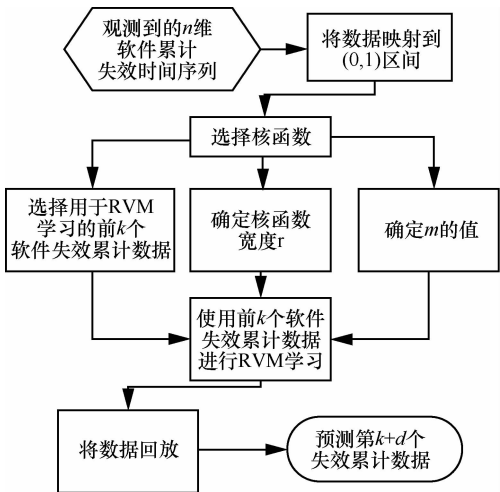


图 1 基于相关向量软件可靠性预测的基本步骤

表 1 评估模型性能所使用的失效数据集

数据集	代码行数	失效数	软件系统类型
数据集 1	21 700	136	实时命令控制系统
数据集 2	10 000	118	飞行器控制软件
数据集 3	35 000	279	硬件控制系统
数据集 4	870 000	535	实时控制系统
数据集 5	200 000	481	监控及实时控制系统

行预测后与真实数据进行比较。

使用软件可靠性预测模型性能分析中最常用的指标平均相对预测误差 (average relative prediction error, AE) [23] 对模型性能进行评价, AE 的值越小说明模型的预测能力越强, 计算式为: $AE = \frac{1}{n-k} \sum_{i=k}^n \left| \frac{\hat{t}_i - t_i}{t_i} \right|$, 其中, n 表示失效数据集总共记录的失效次数, k 表示用于核函数学习的失效次数, $\hat{t}_i$ 表示第 i 次失效时间估计值而 t_i 失效时间实测值。

2.2 核函数参数选取

前期工作[22]表明模型中选用高斯核函数 $K(x_i, x_j) = \exp \left\{ -\frac{\|x_i - x_j\|^2}{r^2} \right\}$ 时可以取得最好预测效果, $r > 0$ 是高斯核函数的带宽参数。核函数参数选取是一个最优化问题, 采用网格搜索法进行核函数参数选择, 令 $r \in [r_1, r_2]$, 设置变化步长为 r_s , 选取其中预测效果最好的值作为模型参数。在数据集 1~数据集 5 上采用不同的 r 值进行的实验研究结果如图 2 所示, 其中, r_1, r_2, r_s 的取值分别为 0、12 和 0.1, 其余取值如下: $m=8$, $\alpha_i (i=1, 2, 3, \dots, m)$ 的初始值均为 0.5, σ^2 的初始值为 1。从图 2 中可以看出, r 的取值对模型预测性能有较大的影响, 本文实验中对 5 个数据集上 r 的取值见表 2。

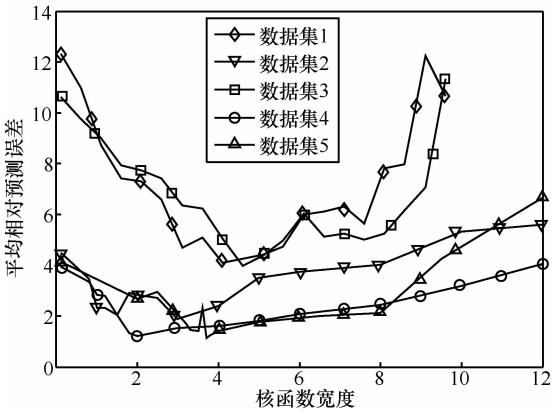


图 2 10 个数据集上 r 值不同时模型预测性能变化曲线

表 2 模型性能最佳时的 r 值

数据集	数据集 1	数据集 2	数据集 3	数据集 4	数据集 5
r	3.8	3.1	5.8	1.9	4.4
最优 AE 值	1.11	2.04	3.96	1.21	4.11

3 实验分析与比较

3.1 Mann-Kendall 检验

假定 $x_1, x_2, \dots, x_n$ 为 m 值不同时的 AE 值序列变量, n 为序列的长度。检验统计值可由 $S = \sum_{j=1}^{n-1} \sum_{k=j+1}^n \text{sgn}(x_k - x_j) = \sum_{j=1}^{n-1} \begin{cases} 1, & x_k - x_j > 0 \\ 0, & x_k - x_j = 0 \\ -1, & x_k - x_j < 0 \end{cases}$ 计算[25], 如果检验统计值 S 的值接近于 0, 则序列数据中不存在趋势; 如果检验统计值 S 的绝对值较大, 则可以断定数据中存在趋势。定义检验统计值: $Z_{\text{statistic}} = \frac{S - \text{sgn}(S)}{\sqrt{\text{Var}(S)}}$, 在给定的 α 置信水平上, $|Z| \geq Z_{1-\alpha/2}$ 时拒绝原假设。即在 α 置信水平上, m 取值不同时的 AE 值序列数据存在明显的上升或下降趋势, 其中 $\text{Var}(S) = \frac{1}{18} [n(n-1)(2n+5)]$ 。

3.2 检验结果及趋势分析

表 3 列出了在 m 取值分别为 6, 7, $\dots$, 30 情况下各个数据集上模型的预测 AE 值, 图 3 显示了其变化趋势。模型中 σ^2 的初始值为 1, $\alpha_i (i=1, 2, 3, \dots, m)$ 的初始值均为 0.5, r 的取值为 1、10 以及表 2 中的值。从表 3 中可以看出, 随着 m 值的不同, 模型预测性能也有极大差异。例如, 在使用数据集 1、 $r=3.8$ 时, AE 的值在 0.87 ($m=9$) 与 2.42 ($m=28$) 之间浮动; 使用数据集 4、 $r=1$ 时, AE 的值在 1.68 ($m=13$) 与 6.15 ($m=$

表 3 数据集 1~数据集 5 上 m 值不同时模型单步预测值

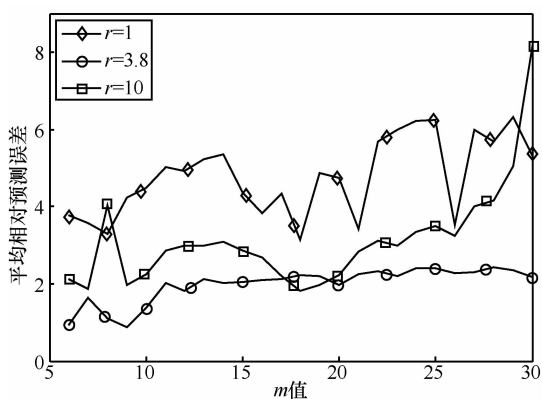
$m \backslash r$	数据集 1			数据集 2			数据集 3			数据集 4			数据集 5		
	1	3.8	10	1	3.1	10	1	5.6	10	1	1.9	10	1	4.4	10
6	3.78	0.92	2.12	2.58	2.02	4.45	8.34	4.34	10.31	2.45	0.82	1.32	8.94	4.60	10.26
7	3.58	1.64	1.86	2.24	2.12	4.22	7.78	4.53	10.89	1.78	0.29	1.43	8.89	4.36	10.32
8	3.30	1.11	4.03	2.67	2.04	4.11	8.93	3.96	9.24	2.37	1.21	2.51	8.67	4.11	11.18
9	4.23	0.87	1.98	2.94	2.09	5.08	8.88	4.73	10.72	2.34	0.65	1.64	9.76	5.03	11.40
10	4.49	1.39	2.25	3.12	2.12	4.24	9.11	5.01	10.64	1.93	0.82	1.89	10.25	5.24	12.14
11	5.03	2.01	2.87	3.06	2.03	5.00	10.19	4.99	11.32	2.94	0.76	1.73	10.08	5.09	11.81
12	4.93	1.83	2.99	2.98	2.21	5.17	12.21	5.03	12.43	3.03	0.79	1.83	10.22	5.19	12.05
13	5.22	2.12	3.00	3.17	2.04	4.42	11.79	5.25	12.65	1.68	0.45	1.99	10.46	5.21	12.45
14	5.34	2.02	3.08	2.79	1.75	4.11	12.02	5.32	12.18	2.87	0.84	2.04	9.86	4.54	11.90
15	4.35	2.05	2.85	2.65	2.53	5.53	11.98	4.96	12.01	3.24	1.03	2.16	10.14	5.18	12.30
16	3.83	2.09	2.68	3.14	3.23	5.35	12.15	5.11	12.67	3.47	0.96	2.54	11.48	6.37	14.02
17	4.34	2.13	2.21	3.45	2.36	4.76	11.99	5.26	12.84	3.59	1.10	2.64	11.07	5.81	13.71
18	3.13	2.22	1.83	4.02	2.07	5.09	11.26	4.72	13.11	4.23	1.12	2.74	10.81	6.09	13.55
19	4.87	2.21	1.98	2.42	1.65	4.87	12.09	5.04	12.98	3.62	1.19	1.85	9.11	4.07	10.96
20	4.73	1.96	2.23	3.65	2.23	5.23	11.37	4.87	12.56	3.86	1.01	3.18	10.75	5.88	13.93
21	3.43	2.25	2.84	3.85	2.25	5.53	12.17	5.25	9.74	4.14	1.24	3.28	11.35	6.10	14.63
22	5.67	2.32	3.12	3.79	2.35	6.25	12.43	6.02	12.86	4.35	1.03	3.46	12.16	6.14	15.62
23	5.99	2.20	2.99	3.68	2.53	5.23	12.53	6.15	13.10	4.86	1.18	3.22	12.36	6.21	15.58
24	6.21	2.40	3.34	2.34	2.06	5.46	12.36	6.35	12.02	4.53	1.14	3.31	10.75	4.40	14.06
25	6.24	2.41	3.49	3.43	2.38	5.43	13.02	6.12	13.14	6.15	0.98	2.96	11.93	5.81	14.89
26	3.53	2.27	3.24	3.66	2.57	4.89	12.26	5.53	13.21	4.84	1.25	3.29	11.76	6.23	15.05
27	5.98	2.29	4.00	3.98	2.89	5.22	12.46	6.00	20.56	5.94	1.43	3.52	12.87	6.87	16.39
28	5.69	2.42	4.15	2.24	2.03	5.57	12.64	6.43	12.99	5.62	0.87	3.19	10.74	4.27	13.89
29	6.32	2.36	5.04	4.15	3.31	6.19	13.09	7.24	12.53	5.43	0.99	2.90	14.70	7.46	17.60
30	5.31	2.17	8.21	4.37	2.26	7.23	12.59	7.34	14.17	5.48	1.11	2.53	13.97	6.63	16.50

25)之间浮动。

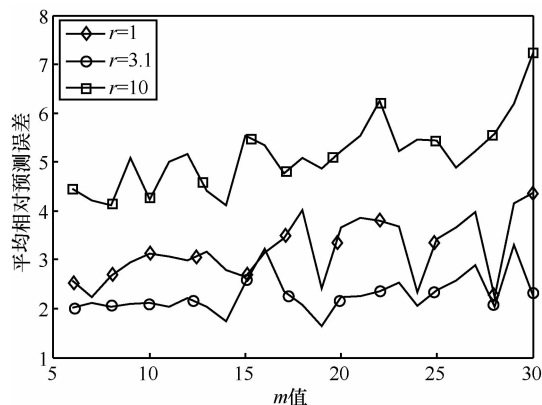
表 4 中给出了各数据集上 m 值变化时的 Z 统计量及置信区间为 95% ($\alpha=0.5, Z_{1-\alpha/2}=1.96$) 时的变化趋势分析。从表 4 中可以看出, m 值增加时, 在各数据集的 AE 值均存在上升趋势, 说明随着 m 的增加, 模型的预测性能存在下降的趋势。合理的解释是, 最近的失效数据更能反映失效过程中的最新特性, 早期失效数据对预测未来短期失效行为作用较小, 现时失效时间数据能比很久之前观测的失效时间数据更好地用于预测未来。

3.3 配对 T 检验

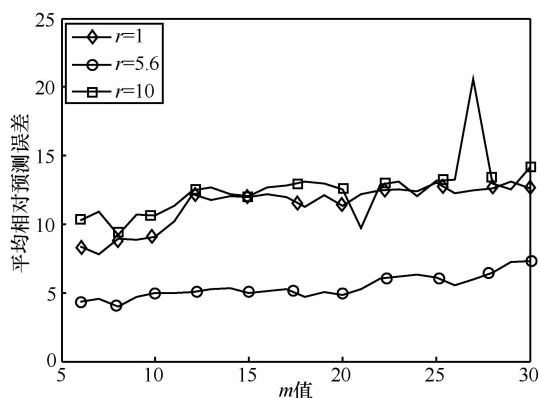
本节将设计随机化实验, 采用配对 T 检验的方法找出 10 个数据集上模型预测能力相对较好的 m 值。配对 T 检验的过程, 是对两个同质的样本分别接受两种不同的处理, 判断不同的处理是否有差别, 实验中, 将 m 值作为处理, 判断其不同时模型预测能力是否有差别。令 $\{X\}$ 、 $\{Y\}$ 为 m 值取值集合, 定义集合运算 $>$, $\{X\}>\{Y\}$ 表示采用集合 X 中的数值比采用集合 Y 中数值具有更好的预测性能。实验中, 把 m 值分成 5 组:



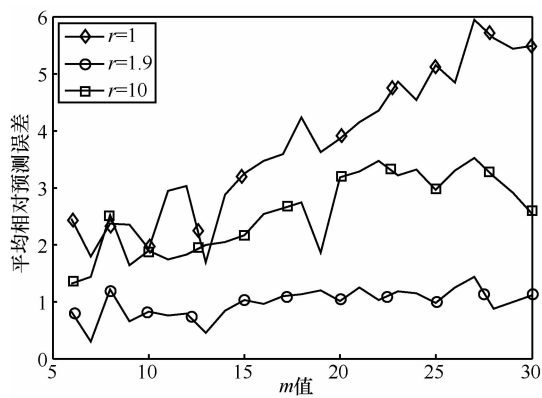
(a) 数据集 1



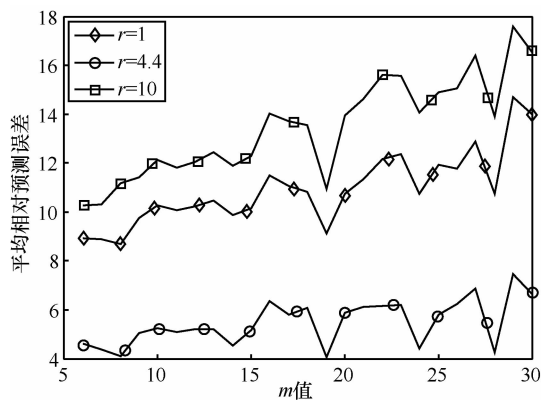
(b) 数据集 2



(c) 数据集 3



(d) 数据集 4



(e) 数据集 5

 图 3 各数据集上 m 值不同时模型预测值变化趋势

 $\{A\}=\{6,7,8,9,10\}$
 $\{B\}=\{11,12,13,14,15\}$
 $\{C\}=\{16,17,18,19,20\}$
 $\{D\}=\{21,22,23,24,25\}$
 $\{E\}=\{26,27,28,29,30\}$,

配对 10 次:

 $\{A\} \rightarrow \{B\}, \{A\} \rightarrow \{C\}, \{A\} \rightarrow \{D\}, \{A\} \rightarrow \{E\}$
 $\{B\} \rightarrow \{C\}, \{B\} \rightarrow \{D\}, \{B\} \rightarrow \{E\}$
 $\{C\} \rightarrow \{D\}, \{C\} \rightarrow \{E\}$
 $\{D\} \rightarrow \{E\}$ 。

在 $H_0: \mu_1 - \mu_2 \geq D_0$ 假设下,小样本情况下配对检验可用统计量 $t = \frac{\bar{d} - D_0}{\sigma_d} \sqrt{n} \approx \frac{\bar{d} - D_0}{S_d} \sqrt{n}$ 表示^[18],其中 n 表示样本数量, d_i 第 i 个配对样本数据的差值 $d_i = x_i - y_i, i=1, 2, \dots, n$,

表示配对样本数据差值的平均值, $\mu_1 - \mu_2 = \sum_{i=1}^l d_i, \sigma_d$ 表示配



对样本数据差值的总体标准差, S_d 表示配对样本数据差值的标准差。表 5 中给出了 $D_0=0$ 的检验结果, $Sig.$ 表示 significance, 在置信度 α 取值为 95% 的双侧检验情况下, 如果 $0.01 < Sig. < 0.05$ 则表示差异显著, 如果 $Sig. < 0.01$ 表示差异极显著, 这两种情况下, 拒绝 H_0 假设。从表中可以看出, $\{A\} > \{C\}$ 、 $\{A\} > \{D\}$ 、 $\{A\} > \{E\}$ 、 $\{B\} > \{C\}$ 、 $\{C\} > \{D\}$ 、 $\{C\} > \{E\}$ 的结论均极显著, 因此可以得出结论: 使用基于 RVM 的软件可靠性模型时, 在区间 $[6, 30]$ 中, $m \in \{6, 7, 8, 9, 10\}$ 为最佳取值。

表 4 各数据集上的 AE 值趋势检验

数据集	r	Z 统计量	解释
数据集 1	1	2.872 7	上升
	3.8	5.021 3	上升
	10	3.479 9	上升
数据集 2	1	2.989 4	上升
	3.1	2.499 0	上升
	10	2.805 4	上升
数据集 3	1	4.719 0	上升
	5.6	4.857 8	上升
	10	4.133 8	上升
数据集 4	1	5.722 0	上升
	1.9	3.012 8	上升
	10	4.414 1	上升
数据集 5	1	4.530 9	上升
	4.4	3.269 7	上升
	10	5.161 4	上升

表 5 配对 T -检验分析结果

配对	平均值	T 统计量	$Sig.$ (双边)	性能比较
pair 1 $\{A\} \rightarrow \{B\}$	-0.542 07	-7.460	0.048	$\{A\} > \{B\}$
pair 2 $\{A\} \rightarrow \{C\}$	-0.906 20	-10.409	0.009	$\{A\} > \{C\}$
pair 3 $\{A\} \rightarrow \{D\}$	-1.506 00	-14.866	0.002	$\{A\} > \{D\}$
pair 4 $\{A\} \rightarrow \{E\}$	-2.083 00	-16.327	0.007	$\{A\} > \{E\}$
pair 5 $\{B\} \rightarrow \{C\}$	-0.364 13	-5.609	0.004	$\{B\} > \{C\}$
pair 6 $\{B\} \rightarrow \{D\}$	-0.963 93	-11.960	0.010	$\{B\} > \{D\}$
pair 7 $\{B\} \rightarrow \{E\}$	-1.540 93	-12.951	0.010	$\{B\} > \{E\}$
pair 8 $\{C\} \rightarrow \{D\}$	-0.599 80	-8.812	0.009	$\{C\} > \{D\}$
pair 9 $\{C\} \rightarrow \{E\}$	-1.176 80	-9.999	0.005	$\{C\} > \{E\}$
pair 10 $\{D\} \rightarrow \{E\}$	-0.577 00	-5.811	0.026	$\{D\} > \{E\}$

4 结束语

基于相关向量机理论, 对软件失效时间数据与在其之前发生的 m 个失效时间数据进行建模, 对 10 个数据集上用于建模的失效数据数量进行实验研究。首先, 通过 Mann-Kendall 检验发现, m 值变大时, 模型预测性能存在下降趋势, 然后采用配对 T 检验发现, 在 10 个数据集上, $m \in \{6, 7, 8, 9, 10\}$ 时, 模型具有最好的预测性能。

进一步工作包括以下内容: 采用模糊遗传算法、粒子群算法或者模拟退火算法等优化技术对核函数参数的自动赋值算法进行研究; 适合于软件可靠性建模的核函数选择及构建, 进一步提高模型预测性能。

参考文献

- 1 IEEE Std 1633-2008. IEEE Recommended Practice on Software Reliability, 2008
- 2 Inoue S, Yamada S. Generalized discrete software reliability modeling with effect of program size. IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, 2007, 37(2):170~179
- 3 Liu M X, Miao L, Zhang D Q. Two stage cost sensitive learning for software defect prediction. IEEE Transactions on Reliability, 2014, 63(2):676~686
- 4 李海峰, 王栓奇, 刘畅等. 考虑测试工作量与覆盖率的软件可靠性模型. 软件学报, 2013, 24(4): 749~760
Li H F, Wang S Q, Liu C, et al. Software reliability model considering both testing effort and testing coverage. Journal of Software, 2013, 24(4):749~760
- 5 Peng W, Huang H Z, Xie M, et al. A Bayesian approach for system reliability analysis with multilevel pass fail, lifetime and degradation data sets. IEEE Transactions on Reliability, 2013, 62(3):689~699
- 6 陆文, 徐锋, 吕建. 一种开放环境下的软件可靠性评估方法. 计算机学报, 2010, 33(3): 452~462
Lu W, Xu F, Lv J. An approach of software reliability evaluation in the open environment. Chinese Journal of Computers, 2010, 33(3):452~462
- 7 Sun Z B, Song Q B, Zhu X Y. Using coding based ensemble learning to improve software defect prediction. IEEE Transactions on Systems, Man, Cybernetics, 2012, 42(6): 1806~1817
- 8 Chatzis S, Andreou A. Maximum entropy discrimination poisson regression for software reliability modeling. IEEE Transactions on Neural Networks and Learning, 2015, has been accepted by inclusion in a future issue

- 9 谢景燕,安金霞,朱纪洪.考虑不完美排错情况的 NHPP 类软件可靠性增长模型.软件学报, 2010, 21(5): 942~949
Xie J Y, An J X, Zhu J H. NHPP software reliability growth model considering imperfect debugging. Journal of Software, 2010, 21(5): 942~949
- 10 陆文,徐锋,吕建.一种开放环境下的软件可靠性评估方法.计算机学报, 2010, 33(3): 452~462
Lu W, Xu F, Lv J. An Approach of Software Reliability Evaluation in the Open Environment. Chinese Journal of Computers, 2010, 33(3): 452~462
- 11 Schneidewind N, Hinchey M. A complexity reliability model. Proceedings of 20th International Symposium on Software Reliability Engineering, Mysuru, India, 2009: 1~10
- 12 Cotroneo D, Pietrantuono R, Russo S. Combining operational and debug testing for improving reliability. IEEE Transactions on Reliability, 2013, 62(2): 408~423
- 13 Liu M X, Miao L, Zhang D Q. Two stage cost sensitive learning for software defect prediction. IEEE Transactions on Reliability, 2014, 63(2): 676~686
- 14 Wayne M, Modarres M. A bayesian model for complex system reliability growth under arbitrary corrective actions. IEEE Transactions on Reliability, 2015, 64(1): 206~220
- 15 Li X N, Mabu S, Hirasawa K. A novel graph based estimation of the distribution algorithm and its extension using reinforcement learning. IEEE Transactions on Evolutionary Computation, 2014, 18(1): 98~113
- 16 Tian L, Noore A. Dynamic software reliability prediction: an approach based on support vector machines. International Journal of Reliability, Quality and Safety Engineering, 2005, 12 (4): 309~321
- 17 Xing F, Guo P, Lyu M R. A novel method for early software quality prediction based on support vector machines. Proceedings of International Symposium on Software Reliability Engineering (ISSRE'05), Chicago, Illinois, USA, 2005: 213~222
- 18 Yang B, Li X. A study on software reliability prediction based on support vector machines. Proceedings of IEEE International Conference on Industrial Engineering and Engineering Management, Singapore, 2007: 1176~1180
- 19 Yuan F Q, UDAY K, MISRA K B. Complex system reliability evaluation using support vector machine for incomplete data-set. International Journal of Performability Engineering, 2011, 7 (3): 32~42
- 20 Park J, Lee N, Baik J. On the long-term predictive capability of data-driven software reliability model: an empirical evaluation. Proceedings of 25th International Symposium on Software Reliability Engineering, Naples, 2014: 45~54
- 21 楼俊钢,蒋云良,申情等.软件可靠性预测中不同核函数的预测能力评估.计算机学报, 2013, 33(6): 1303~1311
Lou J G, Jiang Y L, Shen Q, et al. Evaluating the prediction performance of different kernel functions in kernel based software reliability models. Chinese Journal of Computers, 2013, 36(6): 1303~1311
- 22 Lou J G, Jiang J H, Shuai C Y. A study on software reliability prediction based on transduction inference. Proceedings of IEEE 19th Asian Test Symposium, Los Alamitos: IEEE Computer Society, Shanghai, China, 2010: 77~80
- 23 Ohba M. Software reliability analysis models. IBM Journal of Research and Development, 1984, 28(4): 428~443
- 24 Kendall M G. A new measure of rank correlation. Biometrical, 1938, 30(2): 81~93
- 25 Douglas C D. Design and Analysis of Experiments, 6th Edition. New York: John Wiley & Sons. Inc, 2007

[作者简介]



杨小明,男,湖州师范学院实验师、实验中心主任,主要研究方向为软件可靠性建模、大数据分析、可信计算等。



楼俊钢,男,博士,湖州师范学院副教授、系主任,主要研究方向为软件可靠性建模、软件测试、大数据分析等。



沈张果,男,湖州师范学院讲师,主要研究方向为大数据分析、软件可靠性建模、智能交通等。



胡文军,男,博士,湖州师范学院副教授、副院长,主要研究方向为模式识别、人工智能等。