

引用格式:惠子青,刘晓燕,严馨. 一种动态加权组合神经网络模型的软件测试方法 [J]. 微电子学与计算机, 2020, 37(4): 60-64. [HUI Z Q, LIU X Y, YAN X. Software testing method based on dynamic weighted combined neural network model [J]. Microeletronics & Computer, 2020, 37(4): 60-64.]

一种动态加权组合神经网络模型的软件测试方法

惠子青, 刘晓燕, 严馨

(昆明理工大学 信息工程与自动化学院, 云南 昆明 650500)

摘 要: 软件可靠性评估性能直接影响软件测试的工作量, 本文针对软件测试过程中故障检测和故障引进问题, 提出了一种基于动态加权组合的神经网络软件测试方法, 该方法考虑软件工程的多样性, 利用神经网络方法构建动态加权组合模型, 并结合故障引进过程完成故障检测和预测. 通过两组真实的失效数据集(DS1 和 DS2)的试验, 将所提方法与现有的软件可靠性增长模型(Software Reliability Growth Models, SRGMs)进行比较, 结果显示考虑故障引进的动态加权组合神经网络模型拟合效果最优, 表现出了更好地软件可靠性评估性能和模型通用性.

关键词: 软件可靠性; 动态加权组合; 故障引进; SRGMs

中图分类号: TP311

文献标识码: A

文章编号: 1000—7180(2020)04—0060—05

Software testing method based on dynamic weighted combined neural network model

HUI Zi-qing, LIU Xiao-yan, YAN Xin

(Kunming University of Science and Technology, Faculty of Information Engineering
and Automation, Kunming 650500, China)

Abstract: Software reliability assessment performance directly affects the workload of software testing. This paper proposes a neural network software testing method based on dynamic weighted combination for fault detection and fault introduction in software testing process. This method considers the diversity of software engineering. The neural network method is used to construct the dynamic weighted combination model, and the fault detection process is combined with the fault introduction process. Through the test of two sets of real failure data sets (DS1 and DS2), the proposed method is compared with the existing Software Reliability Growth Models (SRGMs), and the results show that the dynamic weighted combined neural network considering the fault introduction is considered. The network model has the best fitting effect, showing better software reliability evaluation performance and model versatility.

Key words: Software reliability; dynamic weighted combination; fault introduction; SRGMs

1 引言

软件可靠性是软件工程领域一个重要分支, 它在软件可靠性评估和保障中具有重要作用[1]. 软件可靠性是影响软件质量的最重要因素. 评价软件可

靠性的方法之一就是分析软件的测试过程[2]. 为了能够准确评估软件的可靠性, 现已开发出上百种应用于实际项目的软件可靠性增长模型(Software Reliability Growth Models, SRGMs). 这些开发出的模型都是基于各种各样的假设, 所以模型的通用

性和预测精度不高. 由于神经网络具备很强的非线性运算能力, 因此基于 ANN 的软件可靠性增长模型可以获得更高的预测精度和更强的通用性.

人工神经网络(Artificial Neural Networks, ANNs)是受生物神经系统行为启发而产生的一种非线性、自适应的信息处理系统^[3]. 其通过构造大量且高度互联的神经元, 使它们协同工作, 用于处理和解决复杂全局行为的特定问题. 此类全局行为由神经元和连接神经元之间的参数关系决定.

SRGM(software reliability and growth model)是目前建模可靠性及其过程提高的重要数学工具, 对可靠性的评测、保证以及测试资源管控和最优发布研究具有重要作用^[5].

许多通常表现出非线性模式的行为均是影响软件可靠性增长的因素, 如软件开发方法, 软件开发环境, 软件复杂性, 软件人员等. 现有的统计建模方法对实验结果的论证很大程度上取决于对测试过程的假设^[6].

由于神经网络模型仅需要将历史故障数据作为输入且不需要任何假设条件, 因此神经网络模型比现有的软件可靠性模型具有显著的优势. 因此, 近年来 ANN 软件可靠性模型引起了众多研究人员的注意.

2 SRGM 的 ANN 建模

传统的 SRGM 本质上是通过一个复合函数来建立可靠性预测模型, 神经网络的实质也是通过构建一个多参数的复合函数来求解问题, 因此本文提出了一种基于神经网络的 SRGM, 不仅继承了传统 SRGM 的所有属性, 而且还具有超强的非线性运算能力和分布存储及学习能力.

首先用简单前馈神经网络架构的 SRGM(S_SRG)来说明神经网络方法与传统的 SRGM 的对应关系. S_SRG 由输入层、隐藏层、输出层和一个神经元组成, 如图 1 所示.



图 1 单神经元前馈网络图

在基于 NHPP 的故障检测模式中, 累计故障次数 $N(t)$ 通过强度函数 $\lambda(t)$ 和累计故障数的期望值 $m(t) = E(N(t))$ 来定义非齐次泊松过程(NHPP), $m(t)$ 通过以下式子得出:

$$m(t) = \int_0^t \lambda(x) dx \quad (1)$$

基于 NHPP 的模型都是假设在 $(t, t + \Delta t)$ 内检测出的故障总数与软件中剩余故障的数量呈正比, 由以下微分方程表示:

$$\frac{dm(t)}{dt} = b(t)[a(t) - m(t)] \quad (2)$$

式中, $b(t)$ 为故障检测率函数; $a(t)$ 为故障总数函数.

单神经元前馈网络模型和传统 NHPP 模型之间的一些关系如下:

- $\beta(t)$ 相当于 SRGM 的均值函数 $m(t)$.
- w_1 是软件失效率.
- w_2 是故障检测率.
- 隐藏层中 $\alpha(x)$ 的输出相当于分布函数

例如, 如果我们在隐藏层中构造一个具有激活函数 $\alpha(i) = 1 - e^{-i}$ 的神经网络, 在输出层中构造一个线性激活函数 $\beta(i) = i$, 并且在隐藏和输出中没有偏置. 那么权重为 w_1 的隐藏层输入为:

$$x(t) = w_1 t + b_1 \quad (3)$$

式中, b_1 是偏置.

隐藏层的输出可以表示为:

$$h(t) = \alpha(x(t)) = 1 - e^{-x(t)} \quad (4)$$

如果偏置 b_1 可以忽略, 我们假设其为零, 那么隐藏层的输出为: $h(t) = 1 - e^{-w_1 t}$ (5)

现在输出层的输入是:

$$y(t) = h(t)w_2 + b_0 \quad (6)$$

如果偏置 b_0 可以忽略, 可以假设为零, 则输出层的输出为:

$$g(t) = \beta(y(t)) = w_2(1 - e^{-w_1 t}) \quad (7)$$

如果我们假设 $w_1 = b$, $w_2 = a$, 则式(7)与传统的 Goel-Okumoto 模型相对应. 这里我们需要特别说明的是反向传播算法要求激活函数必须是处处连续且可微的, 由于上面的函数均属于初等函数范畴. 因此其具有处处连续和可微的性质.

2.1 动态加权组合模型

动态加权组合模型(Dynamic Weighted Combinational Model, DWCM)结合了各种现有模型, 是一种通用的神经网络软件可靠性评估模型(方法). 该模型使用前馈神经网络结构, 输入层和输出层中有单个神经元, 隐藏层中有三个神经元, 隐藏层中的每个神经元从单个输入神经元接收加权输入, 隐藏层中每个神经元的输出不同的比例加权, 隐藏层的组合加权的输出作为输入馈送到输出层, 输出层根

据输出层中激活函数确定 ANN 模型的输出. 该前馈神经网络架构如图 2 所示.

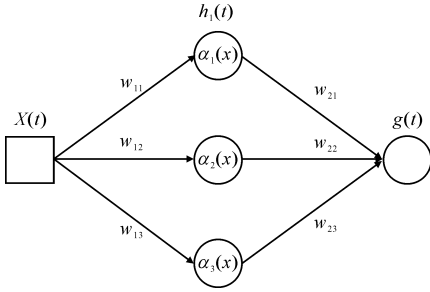


图 2 动态加权组合模型的网络体系结构

隐藏层中的第 j 个神经元的激活函数为 $\alpha_j(x)$. 该图 4 中隐藏层单元的激活函数定义为:

$$\alpha_1(x) = 1 - e^{-x} \quad (8)$$

$$\alpha_2(x) = 1 - (1 + x)e^{-x} \quad (9)$$

$$\alpha_3(x) = \frac{1}{1 + e^{-x}} \quad (10)$$

输出层神经元的激活函数定义为:

$$\beta(x) = x \quad (11)$$

w_{1j}, w_{2j} ($j = 1, 2$ 和 3) 分别是网络结构的输入层分配给隐藏层和隐藏层到输出层的权重. 我们假设该网络架构隐藏层和输出层的神经元没有偏置. 则隐藏层的第一, 第二和第三神经元的输入分别是:

$$x_1(t) = w_{11}t \quad (12)$$

$$x_2(t) = w_{12}t \quad (13)$$

$$x_3(t) = w_{13}t \quad (14)$$

对应于上述输入, 隐层中每个神经元的输出分别为:

$$h_1(t) = \alpha_1(w_{11}t) = 1 - e^{-w_{11}t} \quad (15)$$

$$h_2(t) = \alpha_2(w_{12}t) = 1 - (1 + w_{12}t)e^{-w_{12}t} \quad (16)$$

$$h_3(t) = \alpha_3(w_{13}t) = \frac{1}{1 + e^{-w_{13}t}} \quad (17)$$

输出层单个单元的输入为:

$$y(t) = \left[\begin{aligned} &w_{21}(1 - e^{-w_{11}t}) + w_{22}(1 - (1 + w_{12}t)e^{-w_{12}t}) + \\ &w_{23}\left(\frac{1}{1 + e^{-w_{13}t}}\right) \end{aligned} \right] \beta \quad (18)$$

因此输出层每个单元的输出为:

$$g(t) = w_{21}(1 - e^{-w_{11}t}) + w_{22}(1 - (1 + w_{12}t)e^{-w_{12}t}) + w_{23}\left(\frac{1}{1 + e^{-w_{13}t}}\right) \quad (19)$$

注意, 如果我们假设 w_{1j} ($j = 1, 2$ 和 3) 等价于传统 SRGM 的故障检测率, w_{2j} ($j = 1, 2$ 和 3) 等价

于软件中总故障内容的比例, 即, $w_{11} = b_1, w_{21} = a_1, w_{12} = b_2, w_{22} = a_2, w_{13} = b_3, w_{23} = a_3$. 由上可知公式(19)可以写成下式:

$$m(t) = a_1(1 - e^{-b_1t}) + a_2(1 - (1 + b_2t)e^{-b_2t}) + \frac{a_3}{1 + e^{-b_3t}} \quad (20)$$

上述神经网络模型是指数型、s 延迟型和灵活学习模型的动态加权组合神经网络模型, 动态组合模型可以根据数据集的特点自动确定各模型的权重, 由于该模型将三种类型的模型组合成一个单一的模型, 因此适合于许多实际应用.

用于软件可靠性评估的神经网络方法故障预测步骤如图 3 所示.

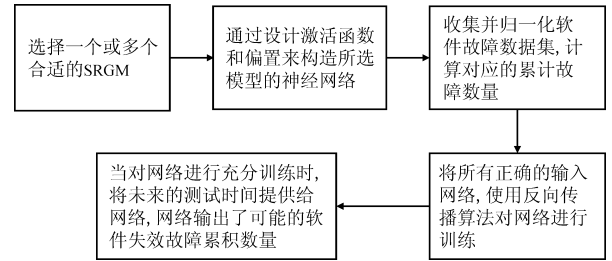


图 3 SRGM 构建神经网络的步骤

2.2 故障引进过程

文献[7]中提出一个基于 NHPP 的不完美调试的软件可靠性增长模型, 其为了建模方便假设故障引进率为常数, 在实际的软件测试过程中, 故障引进率随测试时间发生变化呈现出先增后减的变化^[8], log-logistic 分布函数可以捕获故障引进率随测试时间先增后减的变化, 因此我们用 log-logistic 分布函数来模拟故障引进的过程.

$$a(t) = \frac{a(\alpha t)^d}{1 + (\alpha t)^d} + C \quad (21)$$

式中, a 表示引进的故障总量的期望值; $F(t)$ 是故障引进分布函数; C 是软件内最初存在的故障. 将(21)式两边求导瞬时故障引进强度函数用 $\eta(t)$ 表示为:

$$\eta(t) = aF'(t) = \frac{a\alpha d(\alpha t)^{d-1}}{[1 + (\alpha t)^d]^2} \quad (22)$$

为了表示故障引进率随测试时间变化的趋势, 我们可以将故障引进强度函数 $\eta(t)$ 表示为:

$$\begin{aligned} \eta(t) &= [a + C - a(t)] \frac{F'(t)}{1 - F(t)} \\ &= [a + C - a(t)] h(t) \end{aligned} \quad (23)$$

由于故障引进过程遵循 log-logistic 分布. 因此我们假设被引进的每一个故障遵循故障引进率函数

$h(t)$,其中 α 为率参数, d 为形状参数.因此得出故障引进率函数和考虑故障引进的均值函数为:

$$h(t)=\frac{F(t)}{1-F(t)}=\frac{\alpha d(\alpha t)^{d-1}}{1+(\alpha t)^d}$$

(24)

$$m_c(t)=\int_0^t\eta(t)+\lambda(t)dx$$

(25)

这里需要说明的是^[8],我们为了实现用以实际的软件测试,提出模型的假设条件如下:1)故障总数服从 log-logistic 分布,故障引进率随测试时间呈现先增后减的趋势;2)软件失效发生时,故障立即被排除.新的故障可能会被引进.

3 参数估计和实验

3.1 可靠性评价标准

为了验证所提出模型的性能,能否更好地拟合故障数据集和预测未来软件失效的时间,我们分别采用了三种不同的的评价准则^[9-10],对模型拟合结果进行评价,分别为均方平方误差 (Mean Square Error, MSE), Akaike 信息比较标准 (Akaike information criterion, AIC) 相对误差 (Relative Error, RE).

为了验证提出的故障引进的神经网络模型,首先需要应用最小二乘法进行模型的参数估计,在本文实验中应用了动态加权组合的神经网络模型,对动态加权组合模型的均值函数 $m(t)$ 的参数应用最小二乘法进行估计.同样,对故障引进模型的参数进行估算,表 1 可以看出动态加权组合模型和故障引进模型共同提供了最适合本文数据集的参数估计值.

表 1 模型参数估计值

模型参数	最小二乘法	
DWCM	$a_1=1268.52$	$a_2=1145.36$
	$a_3=118.35$	$b_1=0.125$
	$b_2=0.168$	$b_3=0.182$
log-logistic	$a=123.824$	
	$b=0.1253$	
	$\alpha=1.382$	
	$d=1.565$	
	$C=498.36$	

3.2 实验结果分析

本文采用了两组失效数据集电信系统通讯故障数据集(DS1)^[11]和 IBM 入门开发软件包故障数据集(DS2)^[12],对提出的方法进行了统计学分析和模型拟合度分析,两个失效数据集的测试周期分别为 31 周和 12 周,与现有的原模型进行了性能比较.

表 2 给出了各模型在两组失效数据集上的拟合

结果.

表 2 不同模型的模型拟合结果对比

模型	DS1		DS2	
	MSE	AIC	MSE	AIC
DWCM	150.35	85.2	23.43	84.2
log-logistic	120.48	77.6	18.24	76.8
本文方法	90.03	75	15.62	74.5

由 DS1 故障数据集的拟合效果可以看出.相比于实验中其它两种 SRGM,本文提出的方法的拟合效果最好,MSE=90.03 和 AIC=75 都为最小值.

从 DS2 故障数据集的拟合效果可以看出,相比于实验中其它两种 SRGM,本文提出的方法的拟合效果最好,MSE=15.62 和 AIC=74.5 都为最小值.

4 结束语

本文提出的基于神经网络的软件可靠性测试方法,在所选两组数据集上都显示出最优的拟合效果,显著高于其它模型的的可靠性评估,从而证明了提出方法在软件可靠性预测方面具有更高的预测精度和更好的适用性,所提出的模型相较于传统模型具有以下优势:1)对故障数据集的拟合效果更优.2)结合了 log-logistic,解决了在故障检测过程中可能引入新故障的问题,更好的贴近了实际.在均方平方误差、Akaike 信息比较标准、相对误差三个评价指标上均表现出更高的性能,即可靠性更高.

参考文献:

[1] ZHANG C, MENG F C, KAO Y G, et al. Survey of Software Reliability Growth Model [J]. Journal of Software ,2017,28(9):2402-2430.

[2] LITTLEWOOD B, VERRALL J L. A Bayesian Reliability Growth Model for Computer Software[J]. Applied Statistics, 22(3):332-346.

[3] CUI X,GAO J H . Combined Prediction Model Based on GMDH Neuro Network[J]. Journal of Chinese Computer Systems, 2016, 37(6):1164-1167.

[4] NAGESWARA RAO A,VIJAYAPRIYA P. A robust neural network model for monitoring online voltage stability[J]. International Journal of Computers and Applications, 2019;1-10.

[5] WANG J Y,WU Z B,SHU Y J,et. al. Software Reliability Model with Irregular Changes of Fault Detection Rate [J]. Journal of Software, 2015, 26 (10): 2465-2484.

[6]

ZHENG J,FENG E Q,LIU C. An Non-parametric Input-domain Based Model of Software Reliability[J]. Journal of Chinese Computer Systems, 2012, 33(4): 751_753.

[7]

GOEL A L. Software reliability models : assumptions, limitations and applicability[J]. IEEE Trans. Software Eng. 1985, 11(12):1411-1423.

[8]

WANG J Y,Research on Software Reliability Growth Models Considering Fault Introduction [D]. HEILONGJIANG,Harbin Institute of Technology ,2015.

[9]

HUANG C Y, KUO S Y, LYU M R. An Assessment of Testing-Effort Dependent Software Reliability Growth Models[J]. IEEE Transactions on Reliability, 2007, 56(2):198-211.

[10]

AHMAD N , KHAN M G M , RAFI L S . A study of testing - effort dependent inflection S - shaped software reliability growth models with imperfect debugging[J]. International Journal of Quality & Reliability Management, 2010, 27(1):89-110.

[11]

MUSA J D, IANNINO A, OKUMOTO K. Software Reliability Measurement Prediction Application[M]. New York: McGram-Hill, 1987: 1-469.

[12]

SHIGERU YAMADA, MITSURU OHBA, SHUNJI OSAKI. s-Shaped Software Reliability Growth Models and Their Applications[J]. IEEE Transactions on Reliability, 1984, R-33(4):289-292.

作者简介:

惠子青 男,(1994-),硕士研究生. 研究方向为软件工程与可靠性建模.

刘晓燕(通讯作者) 女,(1964-),博士,副教授,硕士生导师. 研究方向为模型驱动方法和软件可靠性技术.

E-mail:2730410382@qq. com.

严 馨 女,(1967-),博士,副教授 ,硕士生导师. 研究方向为自然语言处理与软件工程.