

DOI: 10.7652/xjtuxb201308003

贝叶斯证据框架下最小二乘支持向量机的 软件老化检测方法

苏莉^{1,2}, 陈鹏飞¹, 齐勇¹, 武义涵³

(1. 西安交通大学电子与信息工程学院, 710049, 西安; 2. 海南师范大学数学与统计学院, 571158, 海口;
3. 北京大学信息科学技术学院, 100871, 北京)

摘要: 针对当前软件老化的检测、分析和软件再生的不确定性问题,提出了一种基于贝叶斯证据框架的最小二乘支持向量机(LS-SVM)的软件老化检测方法,即:使用最小二乘支持向量机分类器进行数据分类,以此解决数据采集时出现的小样本、高纬度,非线性、局部最小值等问题;通过贝叶斯证据框架来优化 LS-SVM 的超参数,从而提高分类器的学习精度和泛化能力。实验结果表明,在状态清晰区间,软件老化的概率均在 0.7 至 0.9 之间,而高维模型检测出的软件老化的概率为 0 或 1。如果从概率粒度层来描述软件老化,则软件再生的时间点选取效率更高,根据概率值的变化可进一步解析软件老化的不确定性。实验结果及分析显示,概率粒度所描述的软件健康状态更符合软件老化的客观状况。

关键词: 软件老化;最小二乘支持向量机;贝叶斯证据框架

中图分类号: TP301 **文献标志码:** A **文章编号:** 0253-987X(2013)08-0012-07

A Software Aging Detection Method Based on Bayesian Framework Using Least Squares Support Vector Machine

SU Li^{1,2}, CHEN Pengfei¹, QI Yong¹, WU Yihan³

(1. School of Electronics and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China;
2. School of Mathematics and Statistics, Hainan Normal University, Haikou 571158, China;
3. School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China)

Abstract: A software aging detection method based on Bayesian evidence framework using least squares support vector machine (LS-SVM) is proposed to solve the uncertainties in software aging detection and analyzing, and software regeneration. The least squares support vector machine classifier is used to classify the data, so that the problems such as small sample of the data collected, the high-latitude, nonlinearity, and local minimum can be solved. Then, the Bayesian evidence framework is employed to get the optimal LS-SVM hyper-parameters, and hence, the learning accuracy and generalization ability of the classifier are improved. Experimental results in the state clear interval show that the probabilities of software aging given by the proposed method are all between 0.7 and 0.9, while the probabilities given by the high-dimensional model are 0 or 1. When the software aging is described by the probability granularity, the regeneration time can be more effectively selected; moreover, the uncertainty of

the software aging can be further analyzed from the changes of the probabilities. Experimental and analyzing results show that the software health status described by the probability granularity is better in line with the practical status of software aging.

Keywords: software aging; least squares support vector machine; Bayesian evidence framework

软件老化是软件系统在连续运行一段时间后,出现性能下降、失效率增高的现象。目前,已知的软件老化原因包括操作系统资源枯竭、数据损坏、软件错误累积等。软件老化对软件系统的可靠性是一个严峻的考验。随着云计算时代的到来和大数据分析的出现,提高系统可靠性的问题显得尤为突出。解决软件老化的方法是在系统故障发生之前,通过周期性地暂停软件的运行,清除错误累积的环境,在干净的初始状态或中间状态下重新启动,称之为软件再生。该方法需要一些性能度量标准,通过监测软件系统的运行状况、采集关键性能参数、利用回归技术或者机器学习算法,来分析软件的老化程度和判断再生动作的执行点。

软件老化的观察、检测分析及再生策略都具有不确定性。文献[1]采用周期模式的时间序列模型来检测和估计数据集的老化趋势,文献[2]采用非线性有源自回归(NARX)网络模型建立了辨识模型,文献[3]使用内核跟踪工具来收集 Linux 的一些与内核子系统相关的参数,并运用统计分析方法确定、分析与老化资源相关的子系统行为,文献[4]通过神经网络的方法预测了系统内存的使用量,以此判断系统老化的程度。

本文采用最小二乘支持向量机(LS-SVM)模型将采集到的数据分为正常和老化两类,然后通过贝叶斯证据框架对 LS-SVM 参数进行优化,得到分类的概率结果。

1 贝叶斯证据框架下 LS-SVM 模型

1.1 最小二乘支持向量机

LS-SVM 将解决凸二次规划(QP)问题转变为求解一个线性方程组,它是一种基于核函数的学习方法^[5]。传统观念认为,软件老化概念涉及到两种状态:正常状态,可以满足 QoS 的需求;老化状态,性能明显下降,错误率升高,不能满足 QoS 需求。老化检测能够识别出软件是否已处于老化状态,而软件状态是通过采集到的多维参数反映的,则状态的识别就演化为在多维空间的一个非线性二类识别的问题。因此,LS-SVM 能够很好地解决这个问题。

$\{y_k, x_k\}, k=1, \dots, N$ 是一个含有 N 个数据点

的训练集, $x_k \in R^n$ 是第 k 个输入模式, $y_k \in R$ 是第 k 个输出模式,则支持向量机的方法需要构建一个分类器

$$y(x) = \text{sign} \left[\sum_{k=1}^N \alpha_k y_k \Psi(x, x_k) + b \right] \quad (1)$$

式中: α_k 是正规化参数; b 是一个实数恒量; $\Psi(\cdot)$ 是核函数。当 $y(x)=1$ 时,说明软件当前为正常状态;当 $y(x)=-1$ 时,说明软件当前为老化状态。

用于软件状态分析而采集的数据为高维数据,因此对应的分类器转化为

$$y_k [\mathbf{w}^T \phi(x_k) + b] = 1 - e_k, \quad k = 1, \dots, N \quad (2)$$

式中: $\phi(\cdot)$ 是一个非线性函数,可将输入映射到一个高维空间。

根据结构风险最小化规则,即分类函数能够使经验风险最小,margin 最大化(风险上界最小),由此得到的优化问题

$$\min_{\mathbf{w}, b, e} \Gamma(\mathbf{w}, b, e) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + \gamma \frac{1}{2} \sum_{k=1}^N e_k^2 \quad (3)$$

满足式(2)的约束条件。采用拉格朗日乘子法,根据 Kuhn-Tucker 条件^[3],可以得到 α 和 b ,从而得到分类函数 $y(x)$ 。

1.2 贝叶斯证据框架

超参数是决定最小二乘支持向量机的学习精度和泛化能力的重要参数^[4],贝叶斯证据框架是一种超参数的优化估计方法,它包括三层贝叶斯推理,即模型参数、正规化超参数和网络结构的推理。

1.2.1 模型参数 $\mathbf{w}$ 和 b 的推理(第一层) 已知数据点 $D = \{(x_i, y_i)\}_{i=1}^N$,超参数 μ, ζ ,模型 H ,则模型参数 $\mathbf{w}$ 和 b 通过最大化后验概率 $p(\mathbf{w}, b | D, \lg \mu, \lg \zeta, H)$ 得到。经贝叶斯推理^[6],则

$$p(\mathbf{w}, b | D, \lg \mu, \lg \zeta, H) = \frac{p(D | \mathbf{w}, b, \lg \mu, \lg \zeta, H) p(\mathbf{w}, b | \lg \mu, \lg \zeta, H)}{p(D | \lg \mu, \lg \zeta, H)} \quad (4)$$

式中: $p(D | \lg \mu, \lg \zeta, H)$ 是一个正则化常数。假设 ζ 是高斯先验独立可分的, e_i 是符合高斯分布的, $p(\mathbf{w}, b | \lg \mu, \lg \zeta, H)$ 仅依赖于 $\mathbf{w}, b, \zeta$ 和 H ,则第一层推理为

$$p(\mathbf{w}, b | D, \lg \mu, \lg \zeta, H) \propto$$

$$\exp\left(-\frac{\mu}{2}\mathbf{w}^T\mathbf{w}-\frac{\zeta}{2}\sum_{i=1}^Ne_i^2\right) \quad (5)$$

式中: $e_i = y_i - (\mathbf{w}^T\varphi(x) + b)$ 为回归残差。 $\mathbf{w}_{\text{MP}}$ 和 b_{MP} 分别为 LS-SVM 分类器的 $\mathbf{w}$ 和 b 的优化值, $\mathbf{w}_{\text{MP}}$ 和 b_{MP} 的最大后验估计可以通过最小化式(5)的负对数得到。

1.2.2 μ 和 ζ 的推理(第二层) 对第二层应用贝叶斯规则

$$\frac{p(\lg\mu, \lg\zeta | D, H) = \frac{p(D | \lg\mu, \lg\zeta, H) p(\lg\mu, \lg\zeta | H)}{p(D | H)} \quad (6)$$

可以推出 μ 和 ζ 的近似值。 $p(D | \lg\mu, \lg\zeta, H)$ 同式(4), 是正则化常数, 由此得到

$$p(\lg\mu, \lg\zeta | D, H) \propto p(\lg\mu, \lg\zeta | H) \quad (7)$$

为了优化问题, 取 $\gamma = \zeta/\mu$, 再经优化可以得到 γ , 即

$$\min_{\gamma} J(\gamma) = \sum_{i=1}^{N-1} \lg\left[\lambda_{G,i} + \frac{1}{\lambda}\right] + (N-1)\lg[E_{\text{W}}(\mathbf{w}_{\text{MP}}) + \gamma E_{\text{D}}(\mathbf{w}_{\text{MP}}, b_{\text{MP}})] \quad (8)$$

式中: λ 是代价函数 $\Gamma(\mathbf{w}, b) = \mu E_{\text{W}} + \zeta E_{\text{D}}$ 的特征值, 其中

$$E_{\text{W}} = \frac{1}{2}\mathbf{w}^T\mathbf{w} \quad (9)$$

$$E_{\text{D}} = \frac{1}{2}\sum_{i=1}^N(y_i - [\mathbf{w}^T\varphi(x_i) + b])^2 \quad (10)$$

1.2.3 贝叶斯模型对比(第三层) 第三层需要选取一个最优模型, 即通过用贝叶斯规则得到的 H 的后验概率

$$p(H_j | D) \propto p(D | H_j)p(H_j) \quad (11)$$

来判断当前数据点状态的概率。

在所有可能的模型中 $p(H_j)$ 是一样的, 则式(11)可以化为 $p(H_j | D) \propto p(D | H_j)$ 。

第三层的假设与第一层、第二层类似, 由此可得

$$p(D | H_j) \propto \left(\frac{\mu_{\text{MP}}^{N_{\text{eff}}} \zeta_{\text{MP}}^{N-1}}{(\gamma_{\text{eff}} - 1)(N - \gamma_{\text{eff}}) \prod_{i=1}^{N_{\text{eff}}} (\mu_{\text{MP}} + \zeta_{\text{MP}} \lambda_{G,i})}\right)^{1/2} \quad (12)$$

1.3 老化检测环境下模型特殊参数设置

老化数据一般包括 CPU、内存、I/O 操作以及服务器网络带宽等基本资源的使用状况信息。不同维数的数据含义不同, 变化趋势也不同。在本文 LS-SVM 模型中, 构造支持向量机的核函数为径向基函数, 其中涉及到欧氏距离, 欧氏距离的计算会使不同维数的输入 x 相互抵消。由实验可知, 数据变

化趋势不同, 尤其是趋势相反的不同维数据, 使得采用贝叶斯证据框架下的 LS-SVM 模型进行老化趋势的判断结果的概率在 0.5 附近, 比如将 Memory\Cache Bytes 和 Processor(_Total)\Privileged Time 作为输入, 二者变化趋势相反。因此, 在老化检测中, 对于模型中的 x 首先要进行变化趋势的一致性判断。

正则化参数 γ 是最小拟合误差和平滑程度的一个权衡值。针对收集到的数据, 通过实验分析得出, γ 越小, 模型的复杂度越低, 模型的泛化和逼近能力随着 γ 的增大而提高, 但当 γ 增大到一定值之后, 模型精度提高的幅度很小, 实验显示 γ 在 12.64 左右时模型精度变化幅度开始锐减, 因此建模时 γ 的初始值应小于等于 12.64。

2 软件老化检测

2.1 实验环境描述与数据采集

HelixServer 是一款主流的流媒体应用服务器软件, 支持 RTSP、RTP、HTTP、MMS 等多种流媒体协议。HelixServer 具有多格式、跨平台和开源性等优点, 所以应用十分广泛, 且源代码数量庞大, 属于有代表性的复杂系统软件。因此, 本文选取 HelixServer 作为老化实验的研究对象, 利用 RTSP、RDT 协议开发了 HelixClientEmulator 软件, 以模拟大量客户端访问服务器上的流媒体文件。HelixClientEmulator 采用多线程机制, 可以真实模拟多个客户端同时访问流媒体的行为, 每个客户端包括 3 个线程: 音频流处理、视频流处理、会话管理(建立连接、中断连接以及 QoS 参数统计等)。实验平台包括 3 台客户端, 1 台 Helix Server 服务器, 1 台千兆带宽的交换机, 构成小型局域网。实验平台拓扑关系如图 1 所示。

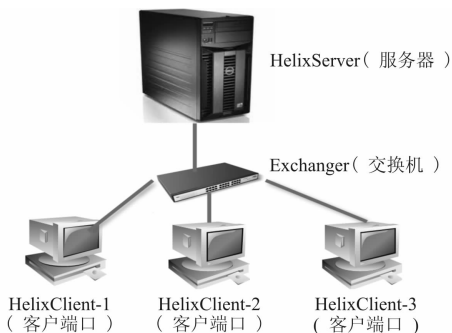


图 1 实验平台拓扑关系

HelixServer 上分布了 100 个不同等级的 rmvb 文件。采用 DOE(Design of Experiment)方法产生

负载,负载由元组(client_count, file_ploy, file_object, file_max_object, sleep_time, file_difference)表示。

实验中客户数 client_count 恒为 600(服务器容量的最大连接数为 900,因此不会因过载而导致伪老化)。HelixclienEmulator 参数集中的数据采集周期是不固定的,其他参数的采集周期为 15 s,实验持续时间为 291 480 s,约 81 h,采集到的样本数共计 19 432。实验中采集的参数如表 1 所示,可全面地描述 CPU、内存、I/O 操作以及 VOD 服务器网络带宽等基本资源的使用状况。

表 1 实验采集的参数

参数名	参数内容
Client	Client Num
	Jitter in Last Ten Seconds
	Total Jitter
	Average Bandwidth
	Average Response Time
OS	\Memory\Available Mbytes
	\Memory\Cache Bytes
	\Memory\Cache Faults Per Second
	\Memory\Page Faults Per Second
	\Memory\Page Reads Per Second
	\Memory\Pool Paged Bytes
	\Memory\SystemCacheResidentBytes
	\PhysicalDisk(_Total)\% Disk Time
	\PhysicalDisk(_Total)\Avg. DiskQueueLength
	\Processor(_Total)\% Interrupt Time
Helix-Server	Players Connected
	Average Bandwidth Output Per Player
	Total Actual Bandwidth Output
	Memory Stats(Bytes In Use)
	LoadState
Rmserver Process	\Process(rmserver)\% Privileged Time
	\Process(rmserver)\% User Time
	\Process(rmserver)\IO Data Bytes Per Second
	\Process(rmserver)\Virtual Bytes
	\Process(rmserver)\Working Set
	\Process(rmserver)\Private Bytes

2.2 数据的预处理及主成分分析

实验中采集到的数据需进行平滑处理,包括填写缺失的值,光滑噪声数据,识别或删除离群点,解决一致性的问题。由于参数的维数较多,各参数的

单位不一致,所以样本数据集在训练之前要进行归一化处理,以保证程序运行时加快收敛。

实验中收集了 30 个内存参数,15 个 CPU 参数,27 个应用服务器参数,在排除了零方差参数及具有相同意义而单位不同的变量之后,还剩余 50 个参数。如果 50 个参数全部用于创建 LS-SVM 模型,则模型规模较大,训练时间较长。因此,采用基于主成分分析预处理的属性约简算法先分析主成分,得出前 9 个主成分的累计贡献率为 88.686%,再计算主成分的载荷,得出第 1 主成分与 Memory\ Page Reads/sec、Memory\Pages Input/sec、Memory\ Pages/sec、Memory\ Transition Pages RePurposed/sec、PhysicalDisk(_Total)\Disk Bytes/sec、PhysicalDisk(_Total)\Disk Read Bytes/sec、PhysicalDisk(_Total)\Disk Reads/sec、PhysicalDisk(_Total)\Disk Transfers/sec 具有正相关关系,第 2 主成分与 PhysicalDisk(_Total)\Avg. Disk sec/Transfer、PhysicalDisk(_Total)\Avg. Disk sec/Read 具有正相关关系,以此类推共获取了 16 个特征量,它们可以代替原先的 50 个特征量。依据粗糙集理论简化决策表中条件属性,采用 navie 算法对 16 个特征量进行离散,从而化简决策表,剔除变化趋势不一致的维数,最终得到 6 个变量,即 Memory \ Pages/sec、Memory \ Cache Bytes、PhysicalDisk (_Total)\ Disk Transfers/sec、PhysicalDisk (_Total)\ Disk Read Bytes/sec、Processor (_Total)\ Privileged Time、PhysicalDisk(_Total)\Avg. Disk Write Queue Length,将作为模型的输入、输出。

2.3 基于 LS-SVM 的老化预测

采用 LS-SVM 模型时,系统处于正常和老化这 2 种状态的数据由分类器来训练。正常状态下的数据用 1 标记,老化状态下的数据用-1 标记。采集的样本为系统负载在各个元组值为(600,1,100,100,0,1 000)时的数据,实验运行 11.5 h。实验中发现:每个播放器的平均带宽在经历快速下降后逐渐趋近稳定,但是仍存在下降的趋势;HelixServer 耗用的内存总量及页面出错率有缓慢上升的趋势;系统缓冲的文件大小一直在增加;处理器利用率和输出网络带宽也有缓慢下降的趋势。这反映出软件系统的性能下降,可以判定系统开始老化,由此收集到 2 种状态的数据(见图 2、图 3)。实验中流媒体类型为 rmvb,带宽在 200 kb。从图 2 可以看出:在 1 500~4 500 s之间带宽大约在 400 kb 以上,可以满足 QoS 要求,这段数据可作为正常状态数据,分

类标记为 1; 在 10 500 ~ 42 000 s 之间, 数据在 200 kb 上下浮动, 说明该区间流媒体不能满足 QoS

要求, 可见 22 500 ~ 27 000 s 的数据具有代表性, 可作为老化数据, 分类标记为 -1。

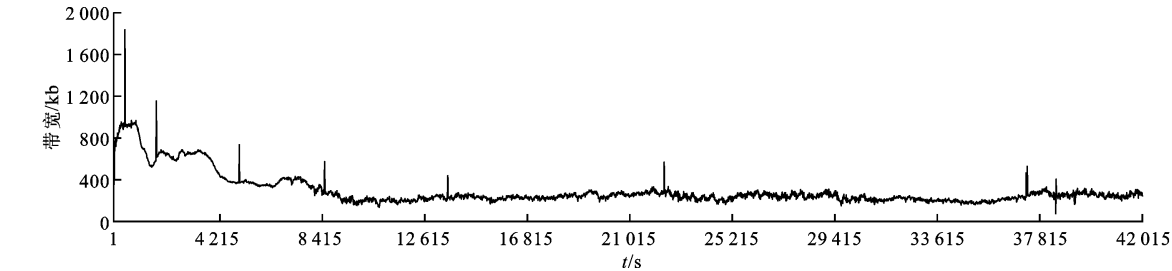


图 2 平均输出带宽

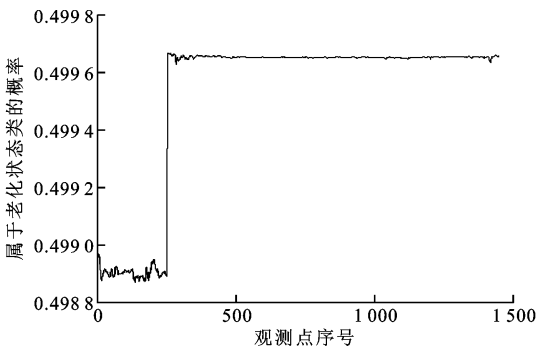


图 3 测试点属于老化状态类的概率

确定 LS-SVM 模型需要先确定 γ 和 σ (带宽) 2 个参数, 然后选择径向基核函数。本文中 $\gamma=10, \sigma^2=0.2$, 则 LS-SVM 模型的分类结果如图 4 所示。

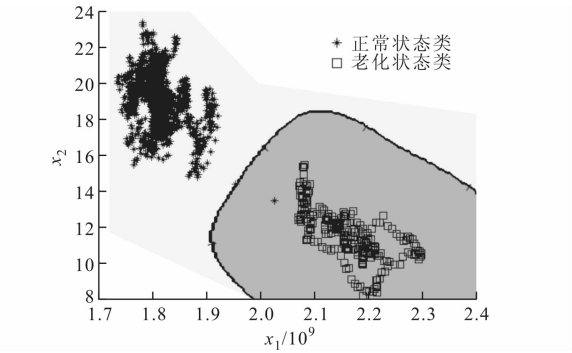


图 4 LS-SVM 模型的分类结果

通过贝叶斯证据框架推理参数时, 首先利用 tunelssvm 函数调整参数, 参数的选用范围分别为 $\gamma \in [10^{-3}, 105]$ 和 $\sigma^2 \in [0.01, 3]$, 初值分别为 $\gamma=4.767\ 1, \sigma^2=0.009\ 427\ 4$ 。根据贝叶斯推理可以得到超参数下的初始值分别为 $\gamma=54.598\ 2, \sigma^2=4.517\ 3$; 通过第 2 层推理得到的最优值 $\gamma=562\ 756$; 通过第 3 层推理得到最优值 $\sigma^2=4.517\ 3$ 。使用贝叶斯推理获得的最优值进行分类后的结果如图 5 所示, 由贝叶斯推理得出的最优值在 4 500 ~

30 000 s 的观测点经过二元分类器分类后属于正常状态类、老化状态类的概率如图 6 及图 7 所示。

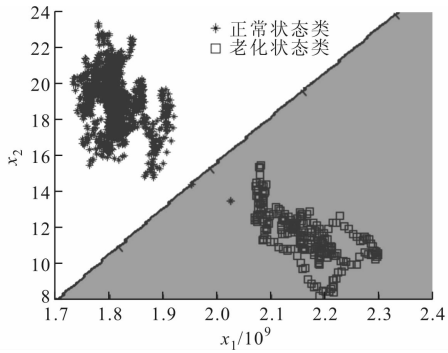


图 5 由贝叶斯推理获得的最优值经分类后的结果

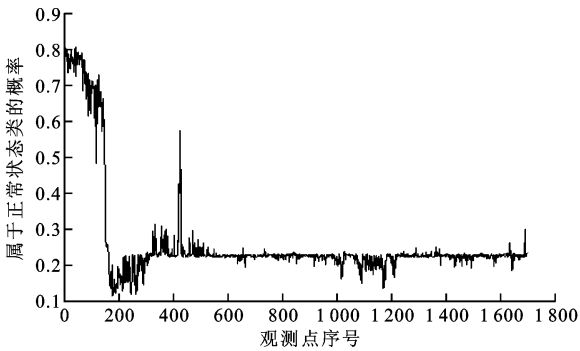


图 6 测试点属于正常状态类的概率

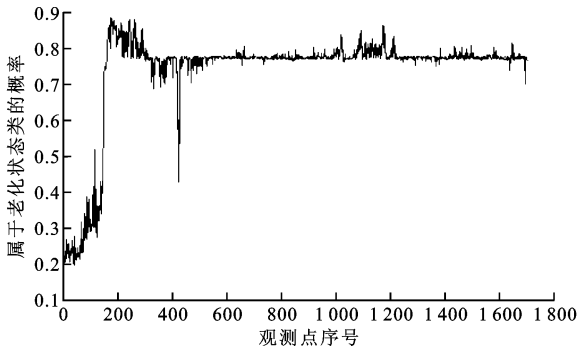


图 7 测试点属于老化状态类的概率

实验采样周期为 15 s,即每个观测点序号之间相隔 15 s。预测老化时发现:0~2 400 s 的观测点属于正常状态类的概率在 0.8~0.6 之间,说明此区间属于正常状态的概率很高;2 550~3 750 s 的观测点属于正常状态类的概率在 0.5 附近,且波动非常大,说明此区间数据发生异常,根据建模数据很难界定该区间是绝对的正常或绝对的老化;3 900~27 000 s 的观测点属于老化状态类的概率在 0.75~0.9 之间,说明此区间属于老化的概率很高。

2.4 高维模型下的老化检测结果

为了观察经贝叶斯证据框架优化后 LS-SVM 模型的维数与属于各类概率精度的关系,对较高维数据进行了实验。通过主成分分析获得了能够反映原有的绝大部分信息的 14 个变量,并将其作为 LS-SVM 模型的输入,同时使用预处理后的数据进行模拟训练,结果属于正常状态类的概率如图 8 所示。从图 8 可以看出:模型输入的维数越高,模拟效果就越好,正常状态类的概率接近于 0 和 1,区间状态清晰;5 250~7 500 s 的观测点的概率波动很大,通过 30 次重复实验得出,高维模型的训练时间均值比低维模型减少了 72 s(实验主机为 Intel(R)Core i5-2400 CPU 处理器和 4.00 GB 内存)。

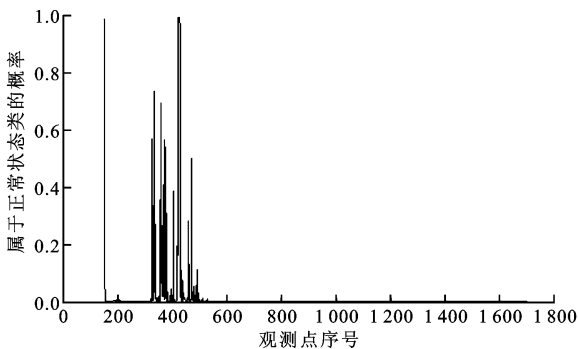


图 8 高维测试属于正常状态类的概率

2.5 基于 NARX 模型的序贯概率比检验方法的老化检测

通常情况下,NARX 网络模型的性能优于全回归神经网络,可与全回归神经网络模型互相转换^[7],因此 NARX 网络模型对于数据的拟合度较高。当系统处于老化状态时,NARX 模型的输出会偏离正常水平,可以使用序贯概率比检验 (SPRT)^[8] 方法对模型的残差进行假设检验,检验阈值 $A=\ln[(1-\alpha)/\beta]=4.595$,其中 α 为误报警率, β 为漏报警率。 $\lambda(n)$ 为第 n 次观测点实际数据与模型预测数据的残差, $\lambda(n)\geq A$ 时可以判定系统发生老化。

由 NARX 模型和 SPRT 残差分析所得的老化判断结果^[2]如图 9 所示。从图 9 可以看出,残差在 0~3 700 左右,老化现象呈逐渐增大的态势,软件在经历了正常状态后出现了老化,老化现象越来越明显,且与本文模型所得概率结果的趋势一致(见图 7)。本文方法判断出软件老化过程有两个明显的阶段,即 3 000 s 前属于正常的可以满足 QoS 的状态,3 000 s 后处于老化状态。

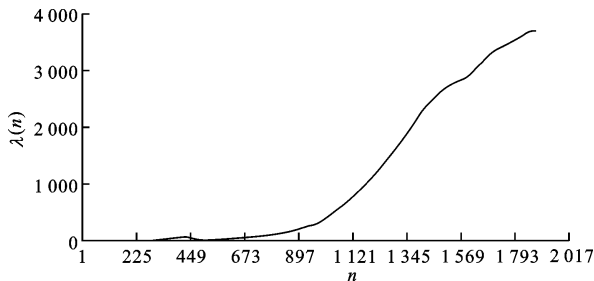


图 9 NARX 模型老化判断曲线

从图 9 还可以看出:在 11 400 s($n=760$)左右,曲线开始离开纵轴 0 值而迅速增大;在 12 000 s($n=800$)后可以确定软件处于老化状态。实验数据表明,8 400 s($n=560,\lambda(n)>4.595$)时系统发生老化^[2]。本文检测结果是:在 2 550~3 750 s 区间内,系统属于正常状态类和老化状态类的概率均为 0.5,说明此区间老化现象不能确定;在 3 750 s 后,概率值明显表示出系统进入老化状态。可见,基于 NARX 模型的序贯概率比检验方法和本文方法对老化转折点的判断基本一致。

NARX 模型中 $\lambda(n)$ 在 $A=4.595$ 前后,系统状态变化不明显,而本文的方法可更准确地描述软件老化的过程,即本文方法判断:在 2 550 s 前后,概率波动较大;在 2 550~4 500 s 之间,老化概率较高;在 4 500 s 后老化概率有所降低,之后逐渐稳定。经数据分析发现,在 2 775 s 后系统因内存超过阈值而进行自动清理工作,此时老化概率有所降低,但单独依赖软件的某些再生方法不能高效地进行清理,因此老化概率仍然保持在 0.78 左右。

从实验结果分析可以得出,本文方法能反映软件老化的客观现状,符合老化因素不确定性的特征,对于再生点的判断更加科学,对于概率的波动可以有针对性地进行分析,从而掌握软件的具体状况,揭示软件状态变化的本质。

3 结 论

本文提出了一种基于 LS-SVM 的软件老化检测方法,该方法的学习精度和泛化能力参数由贝叶

斯证据框架来优化,并以概率来保证检测结果的正确性。通过与基于 NARX 模型的 SPRT 检测对比发现,本文方法的分类结果与实际数据契合度更高,更具实际意义。由于软件老化具有不确定性,所以分类结果在状态清晰区间的概率为 $0.7 \sim 0.9$,在状态不确定区域的概率在 0.5 附近波动。高维模型检测结果表明,在状态清晰区间,检测结果的概率为 0 和 1 ,概率波动区间与低维模型基本一致,但时间消耗少量增加。如果考虑负载的影响,通过分析概率值、解析软件老化的不确定性,则有利于进一步揭示软件老化的根本原因。

本文采用 LS-SVM 遵从了传统观点,即将软件状态仅分为正常和老化状态 2 类,而实际的老化是一种连续的过程,因此下一步的研究是将软件状态分为多个类,这样更符合老化的客观现状。

参考文献:

- [1] LI Lei, VAIDYANATHAN K, TRIVEDI K S. Analysis of software aging in a Web server [J]. IEEE Transactions on Reliability, 2006,55(3):411-420.
- [2] SU Li, QI Yong. Software aging detection based on NARX model [C]//Web Information Systems and Applications Conference. Los Alamitos, CA, USA: IEEE Computer Society, 2012:105-110.
- [3] COTRONEO D, NATELLA R, PIETRANTUONO R, et al. Software aging analysis of the Linux operating system [C]//IEEE 21st International Symposium on Communication, Networking & Broadcasting; Computing & Processing(Hardware/Software); Power, Energy, & Industry Applications. Piscataway, NJ, USA: IEEE, 2010:71-80.
- [4] TRYBULA W J. Annual review of information science and technology (ARIST) [J]. Data Mining and Knowledge Discovery, 1997,32:197-229.
- [5] SUYKENS J A K, VANDEWALLE J. Least squares support vector machine classifiers [J]. Neural Processing Letters, 1999,9(3):293-300.
- [6] MACKAY D J C. Bayesian interpolation [J]. Neural Computation, 1992,4(3):415-447.
- [7] SIEGELMANN H T, HORNE B G, GILES C L. Computational capabilities of recurrent NARX neural networks [J]. IEEE Transactions on Cybernetics, 1997, 27(2):208-215.
- [8] 蒋浩天,拉塞尔 E L,布拉茨 R D. 工业系统的故障检测与诊断 [M]. 段建明,译. 北京:机械工业出版社, 2003.

[本刊相关文献链接]

- 杜小智,齐勇,侯迪,等.重配置与周期再生相结合的软件再生模型. 2010,44(1):91-95. [doi:10.7652/xjtuxb201001020]
- 蒋曹清,应时,文静,等.面向服务软件中异常处理的形式化建模方法. 2013,47(4):118-124. [doi:10.7652/xjtuxb201304020]
- 毛文涛,冯云芝,闫桂荣.一种可自适应分组的多输入多输出支持向量机算法. 2013,47(6):50-54. [doi:10.7652/xjtuxb201306009]
- 翟治年,奚建清,卢亚辉,等.任务状态敏感的访问控制模型及其有色网仿真. 2012,46(12):85-91. [doi:10.7652/xjtuxb201212015]
- 司刚全,娄勇,张寅松.鲁棒最小二乘支持向量机及其在软测量中的应用. 2012,46(8):15-21. [doi:10.7652/xjtuxb201208003]
- 赵金伟,冯博琴,闫桂荣.泛化的统一比雪夫多项式核函数. 2012,46(8):43-48. [doi:10.7652/xjtuxb201208008]
- 梁炎明,刘丁.规则递归 T-S 模糊模型及其辨识方法. 2012, 46(8):54-58. [doi:10.7652/xjtuxb201208010]
- 吴德,刘三阳.支持向量域多分类器. 2012,46(6):87-91. [doi:10.7652/xjtuxb201206015]
- 丁要军,蔡皖东.采用两阶段策略模型(KTSVM)的 P2P 流量识别方法. 2012,46(2):45-50. [doi:10.7652/xjtuxb201202008]
- 侯重远,江汉红,芮万智,等.工业网络流量异常检测的概率主成分分析法. 2012,46(2):70-75. [doi:10.7652/xjtuxb201202012]
- 王翔,郑建国,张超群,等.采用约束蚁群优化的贝叶斯网络结构学习算法. 2011,45(8):54-61. [doi:10.7652/xjtuxb201108010]
- 张琛,段振华.应用 UML2.0 模型的测试用例生成方法. 2011,45(8):18-23. [doi:10.7652/xjtuxb201108004]
- 陈刚,蔡远利,穆静,等.海量信息异常检测问题的异常概率排序算法. 2011,45(4):36-40. [doi:10.7652/xjtuxb201104007]
- 杨宏晖,戴健,孙进才,等.用于水声目标识别的自适应免疫特征选择算法. 2011,45(12):28-32. [doi:10.7652/xjtuxb201112006]
- 贾立新,郑伊飞,曹晖.利用极值蚁群优化的制粉出力建模变量选择算法. 2011,45(10):7-12. [doi:10.7652/xjtuxb201110002]
- 丁军平,蔡皖东.面向元信息分类的支持向量机改进技术. 2011,45(8):37-42. [doi:10.7652/xjtuxb201108007]
- 郑爱军,仲兆平,戴佳佳,等.生物油催化酯化过程中乙酸转化率智能预测. 2011,45(7):118-122. [doi:10.7652/xjtuxb201107022]

(编辑 苗凌 赵大良)