

文章编号:1001-9081(2005)06-1330-04

遗传程序设计算法中选择调整参数的光滑拟合方法

王四春¹, 张泰山¹, 殷志云², 张楚文²

(1. 中南大学 信息科学与工程学院, 湖南 长沙 410083;

2. 湖南商学院 信息科学研究所, 湖南 长沙 410205)

(wscabcbank@sina.com)

摘 要:对遗传程序设计(GP)算法中的适应度评价函数光滑拟合问题进行了研究,结合 LAM (Linear Association Memory)和 HJ(Hook 和 Jeevs)两种方法,估计 GP 树数值权值,以减少 GP 树适应度值评价的计算代价。提出了一种选择调整参数的新方法,同时,给出了一个数学例子,并与广义交叉实验 B—样条函数仿真比较验证。

关键词:遗传程序设计(GP);光滑拟合;调整参数

中图分类号:TP181 **文献标识码:**A

Smooth fitting with selecting the regularization parameter under the genetic programming algorithm

WANG Si-chun¹, ZHANG Tai-shan¹, YIN Zhi-yun², ZHANG Chu-wen²

(1. College of Information Science and Engineering, Central South University, Changsha Hunan 410083, China;

2. Institute of Information Science, Hunan Business College, Changsha Hunan 410205, China)

Abstract: The smooth fitting problem of fitness evaluation function under the genetic programming(GP) algorithm was discussed. To reduce the computational cost required for evaluating the fitness value of GP trees, numerical weights of GP trees were estimated by adopting both linear associative memories (LAM) and the Hook and Jeeves (HJ) method. So, a novel method for choosing the regularization parameter was presented. One numerical example was given with the comparison of generalized cross-validation (GCV) B-splines.

Key words: Genetic Programming(GP); smooth fitting; regularization parameter

0 引言

考虑一个单变量逼近函数式(1),其中 $\mu(t)$ 为未知函数, $\varepsilon(t)$ 为噪音函数,数学期望 $E(\varepsilon(t))=0$,方差是 σ^2 。观测函数 $y(t)$, $t=t_0, t_1, \dots, t_n(t_0 < t_1 < \dots < t_n)$,构造学习集 $L\{(t_i, y_i) | y_i = y(t_i)\}_{i=0, \dots, n}$,搜索 $\mu(t)$ 的估计量 $\bar{\mu}(t)$ 。目前,有许多 $\bar{\mu}(t)$ 估计方法,如插值多项式法、局部加权回归法、样条插值和神经网络法等。

$$y(t) = \mu(t) + \varepsilon(t) \quad (1)$$

本文使用遗传程序设计^[11,2]的方法,搜索 $\mu(t)$ 的最优估计值首先必须解决两个问题:

1) GP 树上的结点数权值能够以有效的计算方式被估计,选择一个最优 GP 树,通过树结点数权值或权值进行评价。尽管一些 GP 树是 $\mu(t)$ 的好的候选解,但它们的数值权值愈低,获得的级别愈低,将在下一次进化过程中被淘汰。原始的 GP 算法主要集中在动态修改 GP 树的结构,缺乏数值权值估计技术。目前,国外主要采用 Sharman K C.、Watson A H.、Gray G J. 等人研究的非线性优化评价方法,由于种群通常是由数百或数千个 GP 树构成的,要计算每个 GP 树的数值权值非常耗时。本文采用 LAMs 和 HJ 搜索法的混合法,将极大地

减少计算代价。

2) 由于 L 受到噪音的影响,GP 算法适应度评价函数应该含有调整参数项,适应度函数应该包含两项,一项用来考虑 GP 树对 L 的适应性,另一项代表光滑适应度的调节项。这里,最重要的任务是选择一个合适的参数,使之在给定的 L 数据附近获得一个解,同时,尽可能地光滑。如参数太小,解结果不稳定;如参数太大,解结果过分光滑。目前,国内外在 GP 算法数值权值估计中,有许多流行的参数选择方法,但难度均较大^[8-12]。本文设计了一种简单启发式方法,该方法能够有效地计算并选择优良的 GP 树。

1 GP 树适应度函数的调整

遗传程序设计算法是在遗传算法的基础上发展起来的,组成群体的个体采用一种动态树状结构表示计算机程序。本文的候选模型为 $\mu(t)$,GP 树的结点由终结点(Terminals)、原始函数(primitive functions)与运算符组成。其中终结点也称叶结点,它是将问题分级为子问题后最基本的解的成分。这种树状的层与结点均是可变化的。终结点是问题的原始变量,根结点和中间结点统称为内部结点,它们则是组合这些原始变量的函数。在进化过程中,GP 树结构通过遗传算子动态

收稿日期:2005-01-02 基金项目:国家自然科学基金资助项目(60373083)

作者简介:王四春(1965-),男,湖南长沙人,副教授,博士研究生,主要研究方向:计算机人工智能; 张泰山(1939-),男,湖南长沙人,教授,博士生导师,主要研究方向:智能控制、人工生命科学与先进控制理论; 殷志云(1960-),男,湖南长沙人,教授,博士生导师,主要研究方向:计算机代数及先进控制理论; 张楚文(1965-),女,湖南长沙人,副教授,硕士研究生,主要研究方向:系统工程。

变化。遗传操作包括复制、交叉和变异,每个算子应用于被选择的个体,复制是根据它们的适应度值复制个体,交叉是从当前群体中,根据适应度值挑选两个个体的不同部件(如子树、子程序、子路径、子公式等)重新组合产生新个体。变异是由程序随机产生一棵新的子树,以代替被变异概率选中结点以下的原有子树部分。

GP在求解函数逼近时,是一种非常有力的工具。GP树结构在进化过程中,动态变化以获得最优的函数形式。为了改善GP树结构的质量,连接在GP树结点上的数值权值估计总是向着进一步极小化适应度函数方向发展。如果学习样本受到噪音破坏,问题将变得难于解决。GP树应该约束学习样本同它的光滑函数逐渐逼近。构造光滑函数的最有效方法是采用调整适应度评价函数中的调整项(光滑约束条件)。适应度函数采用 $f = E + \lambda\theta$ 形式, E 代表GP树适应逼近度, θ 是光滑估计量, λ 是调整参数。这里只是尽可能地通过使用适当的调整参数的近似数值权值估计来促使GP树作为一个光滑函数与它的期望相符。事实上,仅仅使用遗传算子很难产生这样一个GP树。

1.1 适应度函数的调整方法

给出式(2)中的适应度函数和它的调整项,其中:

$$f_A = \frac{1}{n+1} \sum_{i=0}^n (y(t_i) - \bar{\mu}(t_i))^2, \\ f = f_A + \lambda\theta \quad (2)$$

θ 是GP树函数的光滑估计量, λ 是调整参数。如果选择任何向量作为 θ ,称为零—状态Tikhonov调整^[6],则GP树在进化过程中变成一个高度非线性函数,有时GP树的值呈现非预期的行为,在 t_i 到 t_{i+1} 之间的某一点上突然变得很大或很小。这种行为不能在 $t_i (i=0, \dots, n)$ 某一点上测定到,因此,零—状态Tikhonov调整或基于算子的微分调整不能充分有效地消除多余的震动或在 t_i 到 t_{i+1} 之间的大峰现象^[6,8]。采用式(3)的光滑函数测定 $\theta, \bar{\mu}'_L$ 是 $\bar{\mu}'(t) = \frac{d\bar{\mu}}{dt}$ 在 t_0 到 t_n 之间的曲线长度,能够非常有效地在 $t_i \sim t_{i+1}$ 之间获得无任何大峰的光滑GP树。

$$\theta = \frac{1}{n+1} \left\{ \sum_{i=0}^n \bar{\mu}(t_i)^2 \right\} (|\bar{\mu}'(t_0)| + |\bar{\mu}'(t_n)| + \bar{\mu}'_L) \quad (3)$$

为使进化过程不断进行,应用式(2)中给出的适应度函数,通过优化计算,测定 λ 和估计权值,计算种群中的所有GP树。这种处理方法计算相当费时。所以,我们使用一种简单的方式,用 f_A 的值代替GP树的真实适应度值,当使用遗传算子时,不同时考虑光滑处理。求 f_A 的值,LAMs方法^[1-3]是一种非常有效的计算方法。从群体中随机选择的GP树,只有个别好的GP树具有 f_A 的极小值,然后采用HJ搜索方法^[4-5]选择 λ 参数,求适应度函数 f 的极小值。

1.2 终点集和函数集

在本文中,终点集只含有一个变量 t ,随机大小小于1。 t 有一个数值权值, t 的加权值代表结点的实际输出。经过仔细选择的函数使得GP树产生稳定的解。选择函数集 $F, F = \{g_1,$

$$g_2, g_3, g_4, +, -, *, \div\}, g_1 = \frac{a_1}{1 + e^{-b_1(x-c_1)^2}}, g_2 = \frac{a_2}{e^{[(x-b_2)/c_2]^2}},$$

$$g_3 = \frac{a_3}{b_3} \sqrt{b_3^2 + (x - c_3)^2}, g_4 = a_4(b_4x - c_4)^2, g_1 \text{ 通常是一个}$$

前向神经网络中使用的S形函数, g_2 是高斯核心函数, g_3 是径向基函数(RBF), g_i 含有三个受‘+’,‘-’和‘*’算术算子决定的数值权值(a_i, b_i 和 c_i),同样,它们中的每一个都含有一个输出结点的加权数值权值。此外,本文使用的函数集中,可以用到其他反映特定问题的函数,如:周期函数 $\mu(t)$,三角函数等。

1.3 GP树的适应权值

计算GP树权值过程由两个步骤组成:

1) 采用LAM方法,使用仅含 f_A 的适应度函数粗略计算种群中的GP树的估计权值^[2,3]。

2) 从种群中选择几个具有 f_A 最小值的GP树,用包含 f_A 和 θ 的适应度函数 f 评价它们的权值,采用HJ搜索方法,通过极小化 f 搜索最优权值。HJ搜索方法与其他方法相比,快速而且有效,HJ搜索方法主要特点是采用局部模式而不是采用空间搜索^[4,5]。值得注意的是,当HJ搜索方法完成时,GP树的实际适应度值又变成 f_A ,而遗传算子又被下一代使用。

2 调整参数的选择方法

目前,有许多选择 λ 的方法,包括误差法(DP)、交叉验证法(CV)、复合余数和光滑算子法(CRESO)、L—曲线法^[10,11]、零—交叉法(ZC)^[12]。

DP方法:需要了解方差 σ^2 知识,这对实际应用不利。

CV方法:虽不需要掌握方差 σ^2 知识,但如学习样本不大,CV误差估计较大。

CRESO方法:采用 λ 的最小绝对值作调整参数,产生函数 $\theta(\lambda) + 2\lambda \frac{d}{d\lambda} \theta(\lambda)$ 的相对极大值。由于HJ方法不能产生

$\theta(\lambda)$ 的连续值,所以几乎不可能计算 $\frac{d}{d\lambda} \theta(\lambda)$,同时也很难找到这样一个 λ 值,因为 λ 值的微小变化,会引起HJ方法产生不同的 $\theta(\lambda)$ 值。

L—曲线方法:调整参数的标准是 λ 的选择,相当于在L—曲线的“拐点”有极大曲率,即条件上需要在 f_A 和 θ 的拐点同时获得一个折中合理的数据拟合和光滑,L—曲线方法存在许多弊端,一是曲线的不连续性,二是 λ 值的轻微变化产生的结果差异较大。

ZC方法:一种基于实验证据的启发式方法,即找到一个使 η 最小或为零的最佳参数 λ ,即 $(\lambda) = |f_A(\lambda) - \lambda\theta(\lambda)|$ 。ZC方法的优点是容易计算 η ,但通过实验观测,ZC方法的最大缺点是在许多情况下,容易使GP树过分光滑。

为了弥补目前调整参数选择方法的不足,本文采用了一种基于实验结果的简单启发式方法, λ 的选择通过一个简单的经验公式(4)计算,即:

$$\lambda = \frac{1}{s_\theta} \tau^\alpha \quad (4)$$

$$\tau = \frac{1}{2n} \sum_{i=0}^{n-1} |y(t_{i+1}) - y(t_i) - (\bar{\mu}(t_{i+1}) - \bar{\mu}(t_i))|^2,$$

$s_\theta = \theta, \alpha$ 通过式(5)计算。

$$\alpha = \begin{cases} \max\{100(\max(f_A, \tau) - \min(f_A, \tau)), \tau\} + \xi & \text{if } f_A > \tau \\ \xi & \text{otherwise} \end{cases} \quad (5)$$

每个 τ, s_θ 和 α 值在HJ方法开始估计权值前被确定。

HJ 方法形式描述: $HJ(w^{-}, w^0, r, f)$, 开始点为 w^0 , 最佳点 w^{-} 通过求最小参数值函数 f 寻找, 直到 r 达到最大迭代数。

1) 所有 GP 树的权值通过 LAMs 方法粗略计算。从群体中选择最好的 GP 树, 对它们中的每一个, 计算 τ^1, s_θ^1 , 符号上标代表迭代数。设 $\xi = 0.95$, 计算 α^1 , 使用 $\tau^1, s_\theta^1, \alpha^1$ 通过公式(4)计算 λ^1 。这样, GP 树的权值向量 w^0 就通过 LAMs 方法找到。接着开始 $HJ(w^{-1}, w^0, r, f^1)$, ($f^1 = f_A + \lambda^1 \theta$), 获得权值向量 w^{-1} 。

2) 按照 1), 计算 $\tau^2, s_\theta^2, \alpha^2$ 和 λ^2 , 除 $\xi = 0.8$ 外, 计算 α^2 , 开始 $HJ(w^{-2}, w^0, r, f^2)$ 。

3) 直到当前迭代数 k 达到最大允许的数 h , 重复 2), 这里, 假如 k 是奇数, 设 $\xi = 0.95$; 假如 k 是偶数, 设 $\xi = 0.8$ 。如 h 是偶数, 且 $|f_A^h - f_A^{h-1}| / f_A^h > 0.1, 2)$ 用 $\xi = 0.95$ 重复多次。这里, 当 $HJ(w^{-h}, w^{-h-1}, r, f^h)$ 时 f_A^h 相当 f_A 的值。

本文设计 $r = 1000, h = 4$, 因为 r 太小, GP 树的权值不能充分估计。

3 GP 算法的光滑拟合

算法一

GP 算法中的光滑拟合研究步骤如下:

1) 随机产生一个种群或通过遗传算子从前代产生一个新的种群, 所有树的初始权值按 Koza 方法设为标准的 GP 树, 值为 1。群体中每个树, 增加随机数, 大小小于 0.5, 整个树的权值, 使用 LAMs 方法进行估计。LAMs 方法设法求 f_A 的最小值, 即 f_A 的值代表整个树的适应度值。

2) 选择几个好的 GP 树, $\{\mu_i\}_{i=1, \dots, \theta}$, 获得较小的 f_A 值的树。 θ 代表选择树的数目, 本文选择 5 个好树, 每个被选择的 GP 树, 其权值 $\vec{w}_i^{LAM} = (w_{i,j}^{LAM})_{j=0, \dots, p_i}$ 采用 HJ 方法, 通过调整参数 λ 的选择得到。这里, \vec{w}_i^{LAM} 是 p_i -空间向量。由于 HJ 方法对开始点非常敏感, 轻微的不同开始点可能引起不同的结果, 为避免这个问题, 需要选择几个不同的开始点, 以得到好的结果。本文通过增加随机数到初始权值 \vec{w}_i^{LAM} , LAMs 方法已经发现 $\vec{w}_i^{LAM} = (w_{i,j}^{LAM} + \delta_{i,j})_{j=0, \dots, p_i}$, $\delta_{i,j}$ 是随机数, 大小小于 0.5 $|w_{i,j}^{LAM}|$ 。从 \vec{w}_i^{LAM} 中, HJ 搜索开始, 重复 ρ 次, 本文为了减少计算代价, 设为 5 次。

算法二

由于通过 f 最小估计权值, 不容易产生最好的 GP 树 μ 估计, 所以我们设计一种选择适当权值的启发式方法, 算法描述如下:

对每个 μ' , 权值向量估计从第 k 个点开始, $\vec{w}_{\mu'}^{HJ}(k=1, \dots, \rho)$, 为简便起见, 定义两个符号, 即 $f_{A(k)}$ 和 $\theta_{\mu'}(k)$, $f_{A(k)}$ 代表 $\vec{w}_{\mu'}^{HJ}$ 同 μ 的 f_A , $\theta_{\mu'}(k)$ 代表 μ 的 θ 同 $\vec{w}_{\mu'}^{HJ}$ 。

a) 初始 $\vec{w}_{\mu'}^{HJ}$ 为 $\vec{w}_{\mu'}^{HJ}, \vec{w}_{\mu'}^{HJ}$ 代表最佳权值。

b) 假如 $f_{A(k)} + \xi \theta_{\mu'} > f_{A(k)} + \xi \theta_{\mu'}(k \geq 2)$, $\xi = \frac{\{\frac{1}{2}(f_{A(k)} + f_{A(k)})\}^{1.3}}{\frac{1}{2}(\theta_{\mu'} + \theta_{\mu'})}$, 那么 $\vec{w}_{\mu'}^{HJ}$ 变成 $\vec{w}_{\mu'}^{HJ}$ 。

c) 重复 2) $\rho - 1$ 次, $k = k + 1$, 得到最后 $\vec{w}_{\mu'}^{HJ}$ 。

μ_i 的适应度值变成 $f_{A(k)}$, 遗传操作算子基于这个值在

下一代被应用。从 $\{\mu_i\}_{i=1, \dots, \theta}$ 中选择最好的树 $\mu_{\mu'}$, 每个树的权值向量为 $\vec{w}_{\mu'}^{HJ}$, 定义两个符号 $f_{A(k)}$ 和 $\theta_{\mu'}$, $f_{A(k)}$ 代表 $\vec{w}_{\mu'}^{HJ}$ 同 μ_i 的 f_A , $\theta_{\mu'}$ 代表 μ_i 的 θ 同 $\vec{w}_{\mu'}^{HJ}$ 。

d) 选择 $\mu_{\mu'}$ 作为 $\mu_{\mu'}$ 。

e) 如果 $f_{A(k)} + \xi \theta_{\mu'} > f_{A(k)} + \xi \theta_{\mu'}(i \geq 2)$,

$\xi = \frac{\{\frac{1}{2}(f_{A(k)} + f_{A(k)})\}^{1.3}}{\frac{1}{2}(\theta_{\mu'} + \theta_{\mu'})}$, 那么 $\mu_{\mu'}$ 变成 μ_i 。

f) 重复 e) $\theta - 1$ 次, $i = i + 1$, 最后找到好树 $\mu_{\mu'}$ 。

3) 通过遗传算子产生一个新的种群, 重复 1) 和 2), 通过 d) ~ f) 选择当前代最好的树 μ_C 和上一代最好的树 μ_P , 计算 μ_C 是否比 μ_P 更好。

如果 $f_{A(P)} + \xi \theta_P > f_{A(C)} + \xi \theta_C(k \geq 2)$, 那么 μ_C 被选择, 否则 μ_P 被保留, μ_C 被淘汰。这里, $\xi = \frac{\{\frac{1}{2}(f_{A(P)} + f_{A(C)})\}^{1.2}}{\frac{1}{2}(\theta_P + \theta_C)}$ 。

4) 重复 3), 直到达到允许的代数。

影响好的 GP 树选择的重要因素是 ρ 和 θ 。当它们的值增加时, 找到好的 GP 树的可能性愈大, 但却增加计算代价。

4 数值仿真及结果分析

学习集设为 $\{(t_i, \mu(t_i) + \varepsilon_i) \mid t_i = a + i(b - a)/n, i=0, \dots, n, a \text{ 和 } b \text{ 分别为开始点和结束点}, n=50, \varepsilon_i \text{ 为接近于 } 0 \text{ 高斯噪音, 方差 } \sigma^2. \text{ 准备学习集, 考虑 } \sigma=0.1 \text{ 和 } \sigma=0.05 \text{ 两种情况。尽管高斯噪音的方差几乎相同, 但由于噪音有时严重干扰原函数的特征, 造成 GP 结果或 GCV B-样条函数结果却很不相同, 所以, 每种情况准备 5 个不同的学习集数据。}$

一个学习集的拟合度好坏由 $MSE_L^{\mu} = \frac{1}{n+1} \sum_{i=0}^n \{y_i - \mu(t_i)\}^2$ 决定, $\mu(t)$ 的好坏被定义为 $MSE_T^{\mu} = \frac{1}{n+1} \sum_{i=0}^n \{\mu(t_i) - \bar{\mu}(t_i)\}^2$, 来自噪音学习样本 L 的导数函数 $\mu'(t)$ 的好坏被定义为 $MSE_T^{\mu'} = \frac{1}{n+1} \sum_{i=0}^n \{\mu'(t_i) - \bar{\mu}'(t_i)\}^2$ 。

双峰函数如式(6)定义, 由于双峰函数学习样本的噪音在 $t=0$ 附近可能破坏原函数的特征, 所以容易导致 $\mu'(t)$ 的估计产生很差的结果。这将严重影响 GCV B-样条函数的行为特征。实验结果如表 1 所示。

一方面, GCV B-样条 MSE_T^{μ} 的值在许多情况下很大且差异悬殊, 另一方面, 有些情况下, $\bar{\mu}_{SP1}(t)$ 的结果比 $\bar{\mu}_{CP}(t)$ 略好, 如第 1 和第 3 种情况 ($\sigma=0.05$)、第 2 和第 5 种情况 ($\sigma=0.1$)。对 $\sigma=0.1$, 图 1 表示最好的 $\bar{\mu}_{CP}(t)$ 和它相应的 $\bar{\mu}'_{CP}(t)$ (情况 4), 图 2 表示最差的情况 (情况 2)。观察两个图形, 在靠近 $t=0$ (在最差条件下 $t=0.25$) 区域结果显示同原函数比较图形有些区别, 这是因为该区域学习样本受到噪音干扰所致。

$$\mu(t) = 1.5 \{ \exp(-30(t-0.25)^2) + \sin^2(\pi(t-0.2)) \} \quad 0 \leq t \leq 1 \quad (6)$$

表1 双峰函数 GP 和 GCV B-样条函数结果

		平均值	方差	GCV B-样条			GP				
				GCV 错误	$M_L^{\bar{\mu}}$	$M_T^{\bar{\mu}}$	$M_T^{\bar{\mu}'}$	$M_L^{\bar{\mu}'}$	$M_T^{\bar{\mu}'}$	$M_T^{\bar{\mu}'}$	#结点
噪音： $\varepsilon \sim N(0, 0.05^2)$											
1	-0.0206	0.0490	0.0030	0.0013	0.0011	0.7230	0.0025	0.0012	0.5560	14	
2	0.0060	0.0495	0.0027	0.0012	0.0009	1.0500	0.0002	0.0009	0.3200	11	
3	0.0098	0.0509	0.0031	0.0018	0.0006	0.3540	0.0035	0.0008	0.4910	10	
4	0.0101	0.0520	0.0029	0.0001	0.0013	2.7800	0.0022	0.0005	0.3410	11	
5	0.0065	0.0501	0.0029	0.0012	0.0012	1.6300	0.0025	0.0008	0.4230	11	
噪音： $\varepsilon \sim N(0, 0.1^2)$											
1	0.0316	0.1000	0.0146	0.0005	0.0040	5.2300	0.0011	0.0021	0.3120	19	
2	0.0046	0.1110	0.0140	0.0094	0.0027	1.1000	0.0107	0.0028	0.6340	18	
3	-0.0162	0.1000	0.0104	0.0044	0.0043	4.6900	0.0086	0.0016	0.4590	23	
4	0.0027	0.1070	0.0141	0.0089	0.0015	0.5460	0.0118	0.0009	0.0510	15	
5	-0.0164	0.1080	0.0140	0.0009	0.0020	0.9050	0.0101	0.0023	0.5810	25	

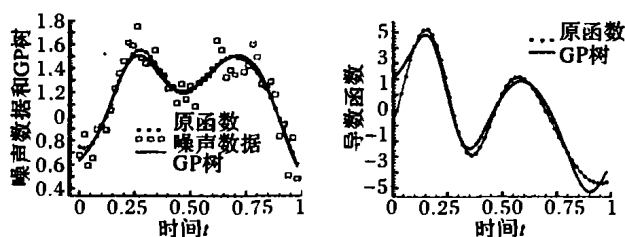


图1 最好的情况

图1中两个峰函数最好的 GP 树 $\bar{\mu}_{GP}(t)$ 和它相应的 $\bar{\mu}'_{GP}(t)$ 结果, 当 $\sigma = 0.1$, $MSE_{\bar{\mu}}^2$ 和 $MSE_{\bar{\mu}'}^2$ 的值分别是 $9.43E-4$ 和 $5.15E-1$ 。

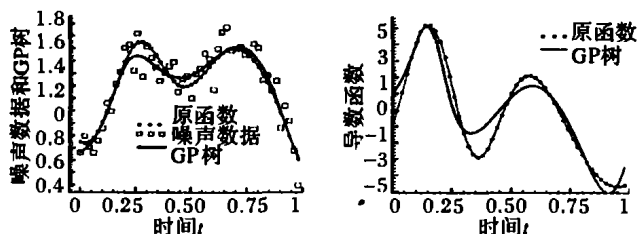


图2 最差的情况

图2两个峰函数最差的 GP 树 $\bar{\mu}_{GP}(t)$ 和它相应的 $\bar{\mu}'_{GP}(t)$ 结果, 当 $\sigma = 0.1$, $MSE_{\bar{\mu}}^2$ 和 $MSE_{\bar{\mu}'}^2$ 的值分别是 $2.75E-3$ 和 $6.34E-1$ 。

5 结语

本文提出了基于 GP 的光滑拟合方法。该方法成功的关键是通过调整参数的选择快速估计 GP 树的权值。为了估计权值, 我们引入了 LAMs 方法首先粗略估计种群整个树的权值, 以减少计算代价, 然后采用 HJ 方法通过调整参数选择搜索潜在的 $\bar{\mu}'(t)$ 好的候选解, 精确计算树的权值。调整参数选择依赖启发式方法, 这样克服了传统的方法如 CV 和 L-曲线方法, 不能快速被应用的不足。同样, 通过调整参数的选择可以获得较好的 GP 树。数值仿真对比了 GP 同 B-样条的性能, 结果显示, 在许多情况下, 估计 $\mu(t)$ 和 $\bar{\mu}'(t)$, GP 胜过 B-样条。

参考文献:

[1] LIN JC, DURAND DM. Weighted linear associative memory ap-

proach to nonlinear parameter estimation. *Journal of Optimization Theory and Applications* [J]. 1996, 90(1): 139-159.

- [2] YEUN YS, LEE KH, YANG YS. Function approximations by coupling neural networks and genetic programming trees with oblique decision trees [J]. *Artificial Intelligence in Engineering*, 1999, 13(2): 223-239.
- [3] YEUN YS, SU JC, YANG YS. Function approximation by superimposing genetic programming trees: with application to engineering problems [J]. *Information Sciences*, 2000, 122, (2-4).
- [4] HOOKE R, JEEVES TA. Direct search solution of numerical and statistical problems [J]. *Journal of ACM*, 2001, 8(4): 212-229.
- [5] MICHAEL GP. Optimization methods for use in computer-aided ship design [J]. in: *Proceedings of the first ship technology and research (STAR) symposium*. The Society of Naval Architects and Marine Engineers, 2001, 13: 1-27.
- [6] MOROZOV VA. *Regularization Methods for Ill-Posed Problems* [M]. Boca Raton: CRC Press, 2001.
- [7] MOROZOV VA. *Methods for Solving Incorrectly Posed Problems* [M]. New York: Springer Press, 1984.
- [8] CRAVEN P, WAHBA G. Smoothing noisy data with spline functions [J]. *Numerische Mathematik*, 2001, 31: 377-403.
- [9] COLLI-FRANZONE P, GUERRI L, TACCARDI B, et al. Finite element approximation of regularized solutions of the inverse potential problem of electrocardiography and applications to experimental data [Z]. *alcolo*, 2001, XXII (1).
- [10] HANSEN PC. Analysis of discrete ill-posed problems by means of the L-curve [J]. *SIAM Review*, 2002, 34: 561-580.
- [11] HANSEN PC, O'LEARY DP. The use of the L-curve in the regularization of discrete ill-posed problems [J]. *SIAM J Sci Stat Comput*, 2002, 14: 1487-1503.
- [12] PETER RJ, RAMESH MG. A new method for regularization parameter determination in the inverse problem of electrocardiography [J]. *IEEE Transactions on Biomedical Engineering*, 2002, 44: 19-39.
- [13] WOLTRING H. A Fortran package for generalized cross-validation spline smoothing and differentiation [J]. *Adv. Eng. Sof, t* 2003, 8(2): 104-113.