

文章编号: 1001-0920(2007)02-0132-06

在线稀疏最小二乘支持向量机回归的研究

王定成^{1,2}, 姜斌²

(1. 南京信息工程大学 计算机与软件学院, 南京 210044; 2. 南京航空航天大学 自动化学院, 南京 210016)

摘要: 现有最小二乘支持向量机回归的训练和模型输出的计算需要较长的时间, 不适合在线实时训练. 对此, 提出一种在线稀疏最小二乘支持向量机回归, 其训练算法采用样本字典, 减少了训练样本的计算量. 训练样本采用序贯加入的方式, 适合在线获取, 并且该算法在理论上是收敛的. 仿真结果表明, 该算法具有较好的稀疏性和实时性, 可进一步用于建模与实时控制等方面的研究.

关键词: 在线; 稀疏; 最小二乘支持向量机

中图分类号: TP18 **文献标识码:** A

Online sparse least square support vector machines regression

WANG Ding-cheng^{1,2}, JIANG Bin²

(1. Institute of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China; 2. College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China. Correspondent: WANG Ding-cheng, E-mail: dcwang2005@126.com)

Abstract: Least square support vector machines regression without sparsity needs longer training time currently, and is not adapted to online real-time training. A better method of online sparse least square support vector machines regression (SVMR) is proposed. Less training time is needed by using sample dictionary. The online SVMR whose samples is available one by one by adding the samples sequentially is convergence theoretically. The simulation result shows that the algorithm has a better sparsity and real-time performance and could be applied in modeling and real-time control etc.

Key words: Online; Sparsity; Least square support vector machines

1 引言

支持向量机(SVM)以其模型简单、具有良好的推广能力和全局最优解等特点,已应用于时间序列建模与预测、非线性过程建模、智能控制与软测量等方面的研究.然而,其基于二次规划的优化计算方法不适合海量数据训练,更不适合在线训练所要求的实时性.因此,为了从真正意义上将支持向量机用于建模与控制领域,研究在线支持向量机是非常必要的.在线训练一个突出的特点是支持向量机回归的学习不是一次离线进行的,而是数据逐一加入,不断进行的优化过程.而常规的 SVM 训练是所有样本都一次性地加入并进行训练,或是批量加入,但本质是所有样本都加入训练.在线学习方法要求在获得下一个数据之前的间隔必须完成训练,否则会影响下一步决策或任务的完成,因此这种算法更要考虑

训练的速度.增量学习算法是被研究较多的一种在线学习算法^[1-4].序贯训练算法也是经常被采用的一种在线训练算法^[5-7].

本文在介绍最小二乘支持向量机(LS-SVM)的基础上,针对最小二乘支持向量机不具有稀疏性的特点,提出一种在线稀疏最小二乘支持向量机回归(OS-LSSVMR),并对该算法的收敛性进行了分析.

2 最小二乘支持向量机回归

对于非线性样本数据 $(x_1, y_1), (x_2, y_2), \dots, (x_i, y_i), \dots, (x_N, y_N)$, $x_i, y_i \in R$, 采用最小二乘支持向量机回归进行函数估计,优化问题就变成

$$\min \frac{1}{2} \|w\|^2 + \sum_{i=1}^l e_i^2, \quad (1)$$

约束条件为

$$y_i = w^T(x_i) + b + e_i, i = 1, 2, \dots, N,$$

收稿日期: 2005-12-10; 修回日期: 2006-05-08.

基金项目: 国家自然科学基金项目(60574083); 江苏省高校自然科学基金项目(06KJB210049).

作者简介: 王定成(1967—),男,安徽霍山人,副研究员,博士,从事智能计算与智能控制理论及应用的研究;姜斌(1966—),男,江西鄱阳人,教授,博士生导师,从事智能控制与故障诊断等研究.

相应的 Lagrangian 形式为

$$L = \frac{1}{2} w^2 + \sum_{i=1}^l e_i^2 + \sum_{i=1}^l \lambda_i \{w^T(x_i) + b + e_i - y_i\}. \quad (2)$$

由 KKT 条件,有如下结果:由 $\partial L / \partial w = 0$,得

$$w = \sum_{i=1}^N \lambda_i (x_i); \quad (3)$$

由 $\partial L / \partial b = 0$,得

$$\sum_{i=1}^N \lambda_i = 0; \quad (4)$$

由 $\partial L / \partial e_i = 0$,得

$$\lambda_i = e_i; \quad (5)$$

由 $\partial L / \partial \lambda_i = 0$,得

$$w^T(x_i) + b + e_i - y_i = 0. \quad (6)$$

在式(2)中采用式(3)~(6)消去 e_i 和 w 后,得到求解最小二乘支持向量机的实现形式为

$$\begin{bmatrix} 0 & \mathbf{1}^T \\ \mathbf{1} & +^{-1} \mathbb{I} \end{bmatrix} \begin{bmatrix} b \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ y \end{bmatrix},$$

其中: $y = [y_1, y_2, \dots, y_N]$, $\mathbf{1} = [1, 1, \dots, 1]$, $+^{-1} = [1, 2, \dots, N]$, \mathbb{I} 为单位矩阵, $\lambda_{ij} = K(x_i, x_j) = (x_i)(x_j)$, $i, j = 1, 2, \dots, N$, $K(x_i, x_j)$ 即为支持向量机的核函数. 求解可得

$$f(x) = \sum_{i=1}^l \lambda_i K(x_i, x) + b.$$

由于没有不敏感区域,最小二乘支持向量机的解没有稀疏性. 一种解决最小二乘支持向量机解的稀疏性的算法^[8]如下:

Step1: 采用 N 个样本点训练最小二乘支持向量机;

Step2: 从样本点中移去少量 λ_i 频谱中最小值的样本点(例如选取样本点中的 5%);

Step3: 在缩减的样本集中重新训练最小二乘支持向量机;

Step4: 返回 Step2, 直到最小二乘支持向量机的性能变差为止.

该方法可使最小二乘支持向量机的解具有稀疏性,但需要对所有样本进行反复的优化,不适应快速训练的需要. 因此,本文提出一种在线稀疏最小二乘支持向量机回归.

3 在线稀疏最小二乘支持向量机回归

在线稀疏最小二乘支持向量机回归算法采用文献[9]中所定义的样本字典方法来解决最小二乘支持向量机的稀疏性问题.

3.1 增强支持向量机解稀疏性的线性方法

定理 1 对于最小二乘支持向量机的解

$$f(x) = \sum_{i=1}^l \lambda_i K(x_i, x) + b, \quad (7)$$

若支持向量样本存在以下关系:

$$K(x, x_k) = \sum_{i=1, i \neq k}^m c_i K(x, x_i), \quad (8)$$

其中 c_i 为常数,则式(7)可表示为

$$f(x) = \sum_{i=1, i \neq k}^m \lambda_i K(x_i, x) + b,$$

即第 k 个支持向量样本 x_k 被映射到高维空间后,线性依赖于 m 个 (x_1, x_2, \dots, x_m) 样本的支持向量,则样本 x_k 可从样本数据中移走.

证明 将式(8)代入式(7),有

$$f(x) = \sum_{i=1, i \neq k}^m \lambda_i K(x_i, x) + \sum_{i=1, i \neq k}^m c_i \lambda_i K(x_i, x) + b,$$

即

$$f(x) = \sum_{i=1, i \neq k}^m (\lambda_i + c_i \lambda_i) K(x_i, x) + b.$$

令 $\lambda_i = \lambda_i + c_i \lambda_i$,则有

$$f(x) = \sum_{i=1, i \neq k}^m \lambda_i K(x_i, x) + b.$$

这样,支持向量 x_k 在新的支持向量机中就消失了,从而减少了支持向量. 文献[11]采用该方法将支持向量机训练结果的支持向量进行了缩减.

3.2 改变偏移项 b 的形式

在线学习需要迭代计算,在式(7)中存在常数项 b ,不适合迭代优化计算,因此必须去掉. 令 $(x) = ((x_1), (x_2), \dots, (x_l))$, $w = (w_1, w_2, \dots, w_l, b')$, 是一个常系数,即在支持向量中增加一维,相应的权矢量也增加一维^[6]. 则式(1)变为

$$\min \frac{1}{2} w^2 + \sum_{i=1}^l \frac{1}{2} \lambda_i^2.$$

约束条件变为

$$y_i = w^T(x_i) + e_i, i = 1, 2, \dots, N,$$

这样,最小化 w^2 的问题就变成了最小化 $w^2 = w^2 + b'^2$. 相应地,式(4)变为

$$f(x) = \sum_{i=1, i \neq k}^m \lambda_i K(x_i, x),$$

其中 $K(x_i, x) = K(x_i, x) + b'^2$. 为叙述方便,将 λ_i 和 $K(x_i, x)$ 分别写成 λ_i 和 $K(x_i, x)$.

3.3 线性相关性的判断

在序贯加入的支持向量机训练样本中,假定在第 $t + 1$ 步训练样本加入之前,共有 t 个样本加入到训练样本中.

定义 1 样本字典: $\exists x_i$, 将样本映射到高维空间,即 $x_i = \Phi(x_i)$, 存在 m 个线性无关的基向量

$\{\phi(\tilde{x}_j)\}_{j=1}^m$ 满足以下条件:

$$\phi(\tilde{x}_j) - \sum_{k=1, k \neq j}^m a_k \phi(\tilde{x}_k) = 0, \quad (9)$$

当且仅当 $a_1, a_2, \dots, a_k, \dots, a_m$ 等于 0 时成立, 则称这 m 个样本基矢量组成的集合为样本字典.

根据定理 1, 任何与字典里基矢量线性相关样本的支持向量都可消去, 这样便可用样本字典里的样本代替所有训练样本, 从而减少了训练样本和支持向量.

在第 t 步, 新加入的样本 x_t , 测试其与样本字典里的基矢量是否线性相关. 如果线性无关, 则将其加入到样本字典; 如果线性相关, 则不加入到字典里. 在线训练时, 新字典样本的加入不能改变已有样本的基矢量, 否则重新计算基矢量会浪费大量的时间. 因此新样本基矢量采用贪婪方式加入^[9], 即直接加入新样本的基矢量而不改变已有样本的基矢量.

在理论上应采用式 (9) 来判断新加入样本与字典里样本的线性相关性, 这样才能精确地判断相关性. 但如果采用式 (9) 进行线性相关性的判断, 有可能导致数值计算的不稳定^[9], 而且还会增加字典的样本数. 因此, 采用以下近似的相关条件进行判断:

$$\epsilon_t = \phi(x_t) - \sum_{j=1}^m a_{t,j} \phi(\tilde{x}_j)^2,$$

其中 ϵ_t 为小的正常数. 由上式得

$$\begin{aligned} \epsilon_t &= \phi(x_t) \phi(x_t)^T - 2 \sum_{j=1}^m a_{t,j} \phi(\tilde{x}_j)^T + \\ &\sum_{i=1}^m \sum_{j=1}^m a_{t,i} a_{t,j} \phi(\tilde{x}_i) \phi(\tilde{x}_j)^T = \\ &K(x_t, x_t) - 2 \sum_{j=1}^m a_{t,j} K(x_t, \tilde{x}_j) + \\ &\sum_{i=1}^m \sum_{j=1}^m a_{t,i} a_{t,j} K(\tilde{x}_i, \tilde{x}_j). \end{aligned} \quad (10)$$

最小化式 (10) 的左式, 就可求出输入样本 x_t 的系数 $a_t = (a_{t,1}, a_{t,2}, \dots, a_{t,m})$. 为了减少字典样本的数目, 适当降低 $a_t = (a_{t,1}, a_{t,2}, \dots, a_{t,m})$ 的精度, 在式 (10) 中加入 l_2 范数正则项 $\|a_t\|_2^2$. 这样最小化式 (10) 的左边就变为

$$\begin{aligned} \min & K(x_t, x_t) - 2 \sum_{j=1}^m a_{t,j} K(x_t, \tilde{x}_j) + \\ &\sum_{i=1}^m \sum_{j=1}^m a_{t,i} a_{t,j} K(\tilde{x}_i, \tilde{x}_j) + \|a_t\|_2^2, \end{aligned} \quad (11)$$

其中 λ 为正常数. 令 $[K]_{i,j} = K(\tilde{x}_i, \tilde{x}_j)$, $K_t = K(x_t, x_t)$, $(\tilde{K})_i = K(x_t, \tilde{x}_i)$, $i, j = 1, 2, \dots, m$. 则式 (11) 可写为

$$\min K_t - 2a_t \tilde{K}_t + a_t^T a_t \tilde{K} + \|a_t\|_2^2 =$$

$$a_t^T (\tilde{K} + I) a_t - 2a_t \tilde{K}_t + K_t.$$

解上式得

$$a_t = (\tilde{K} + I)^{-1} \tilde{K}_t, \quad (12)$$

相应地,

$$\epsilon_t = K_t - \tilde{K}_t a_t. \quad (13)$$

将式 (12) 代入 (13) 即可求出 ϵ_t . 判断其是否小于 ϵ , 如果不小于 ϵ 则将其加入样本库, 否则不加入. 对于支持向量机而言, 虽然这一步增加了计算量, 但 \tilde{K} 只与字典样本有关, 并且已存储在内存里, 因此每步只需计算 $K_t = K(x_t, x_t)$, $(\tilde{K})_i = K(x_t, \tilde{x}_i)$, 而且字典样本数目较小, 因而计算量也较小.

3.4 OS-LSSVM 算法描述

定义 2 字典向量系数 $\tilde{A} = A^T$, $\tilde{A} \in R^{m \times R'}$, 在第 t 步有 t 个样本, 设 $[A]_{i,j} = a_{i,j}$, $i = 1, 2, \dots, t, j = 1, 2, \dots, m$; $[K]_{i,j} = K(x_i, x_j)$, $i, j = 1, 2, \dots, t$, 则当 $\epsilon_t = 0$ 时, 有 $K = A \tilde{K} A^T$. 值通常很小, 这里取

$$K = A \tilde{K} A^T.$$

由定义 2 和有关广义逆的知识^[11], 有

$$f = K^{-1} y = A \tilde{K} A^T^{-1} y = A \tilde{K}^{-1} y.$$

在线稀疏最小二乘支持向量机的训练算法采用序贯训练方式来实现, \tilde{A} 的更新公式为

$$\tilde{A} = A^T (I - \alpha y y^T) / (\alpha (y - f)), \quad (15)$$

$$\tilde{A} = \tilde{A} + \tilde{A}.$$

其中 $y = (y_1, y_2, \dots, y_t)^T$, α 是小的正学习速率常数, “ \cdot ” 表示矩阵相对应的元素相乘.

OS-LSSVM 算法如下:

Step1: 初始化. 选择支持向量机核的种类及参数以及设置初始参数 α 和 ϵ , 设置初始样本库以及初始 \tilde{A} 值等.

Step2: 加入新样本. 对于新加入的样本根据式 (12) 和 (13) 在特征空间判断其是否与样本字典向量线性相关, 如果不相关则将其加入样本字典.

Step3: 采用式 (15) 和 (16) 来修改样本字典样本的系数 $\tilde{A} = \tilde{A} + \tilde{A}$.

Step4: 转 Step2, 继续执行样本训练程序, 直到在线训练完成.

4 OS-LSSVM 收敛性分析

在线稀疏最小二乘支持向量机回归收敛性分析需用到广义逆的有关知识. 有关矩阵的广义逆方面的知识详见文献 [11].

4.1 OS-LSSVM 收敛性分析

由式 (3) ~ (6), 式 (2) 可表示为

$$\max L(\alpha) = -\frac{1}{2} \alpha^T K \alpha - \frac{1}{2} \alpha^T y. \quad (17)$$

设 $L(\alpha) = L(\alpha) + \bar{L}(\bar{\alpha})$, 将 $L(\alpha)$ 分解为两个正交的子

空间 \wedge 和 $\bar{\wedge}$, 其中

$$\wedge = AA^+, \bar{\wedge} = (I - AA^+), \quad (18)$$

$A^+ = (A^T A)^{-1} A^T$ 是 A 的广义逆. 由式(18), 有 $\wedge + \bar{\wedge} = I$, 代入式(17) 有

$$L(\wedge) = -\frac{1}{2}(\wedge + \bar{\wedge})^T K(\wedge + \bar{\wedge}) - \frac{1}{2}(\wedge + \bar{\wedge})^T (\wedge + \bar{\wedge}) + (\wedge + \bar{\wedge})^T y = -\frac{1}{2}\wedge^T (K + \mathbf{1})\wedge - \frac{1}{2}\wedge^T (K + \mathbf{1})\bar{\wedge} - \frac{1}{2}\bar{\wedge}^T (K + \mathbf{1})\wedge - \frac{1}{2}\bar{\wedge}^T (K + \mathbf{1})\bar{\wedge} + (\wedge + \bar{\wedge})^T y.$$

因为 $\wedge, \bar{\wedge}$ 是正交的, 因此第 2 项和第 3 项为 0, 所以有

$$\begin{aligned} \underline{L}(\wedge) &= -\frac{1}{2}\wedge^T (K + \mathbf{1})\wedge + \wedge^T y, \quad (19) \\ \bar{\underline{L}}(\bar{\wedge}) &= -\frac{1}{2}\bar{\wedge}^T (K + \mathbf{1})\bar{\wedge} + \bar{\wedge}^T y. \end{aligned}$$

由式(18) 和广义逆有关的知识, 有

$$\begin{aligned} \wedge &= AA^+ = (AA^+)^T = (A^+)^T A^T, \\ \bar{\wedge} &= (I - AA^+)^T = (I - A^+)^T A^T, \end{aligned} \quad (20)$$

则将式(20) 代入(19) 得

$$\underline{L}(\wedge) = -\frac{1}{2}\wedge^T A^+ (K + \mathbf{1}) (A^+)^T \wedge + \wedge^T A^+ y.$$

定理 2 对于足够小的 δ , 按式(15) 调节的 δ , 使 \underline{L} 产生非负的变化, 即

$$\Delta \underline{L} = \underline{L}(\wedge + \delta \wedge) - \underline{L}(\wedge) \geq 0.$$

证明

$$\begin{aligned} \partial(\underline{L}(\wedge))/\partial\wedge &= -\tilde{K} - \mathbf{1}A^+ (A^+)^T \wedge + A^+ y = -\tilde{K} - \mathbf{1}A^+ (A^+)^T A^T + A^+ y = -\tilde{K} - A^+ (A^+)^T A^T \tilde{e} + A^+ y. \end{aligned}$$

因为

$$\begin{aligned} A^+ (A^+)^T \wedge &= A^+ (A^+)^T A^T = A^+ (A^+)^T A^T \tilde{e}, \end{aligned} \quad (21)$$

由式(14) 有

$$\tilde{K} = A^+ f. \quad (22)$$

由式(21) 和(22) 有

$$\begin{aligned} \partial(\underline{L}(\wedge))/\partial\wedge &= -A^+ f - A^+ (A^+)^T A^T \tilde{e} + A^+ y = A^+ (y - f) - A^+ (A^+)^T A^T (y - f) = A^+ (y - f) (I - (A^+)^T A^T), \\ \Delta \underline{L} &= \delta \wedge^T (\partial(\underline{L}(\wedge))/\partial\wedge) - \frac{1}{2} \wedge^T (K + \mathbf{1}) \wedge. \end{aligned}$$

由式(15), 当 δ 足够小时, \wedge 就足够小, 这样便可略去 \wedge 的高阶项, 因此 $\Delta \underline{L} \approx \delta \wedge^T (\partial(\underline{L}(\wedge))/\partial\wedge)$, 则有

$$\begin{aligned} \Delta \underline{L} &\approx \delta \wedge^T (\partial(\underline{L}(\wedge))/\partial\wedge) = \delta (y - f)^T / \delta \wedge^T (y - f)^T AA^+ (y - f) (I - (A^+)^T A^T) = (y - f)^T / \delta \wedge^T ((y - f)^T AA^+ (y - f) (I - AA^+)) \geq 0, \end{aligned}$$

因为 AA^+ 是半正定的, 所以 $\Delta \underline{L} \geq 0$ 成立.

在每次学习期间, 只有权值 \wedge 被修改, 而 $\bar{\wedge}$ 没有被修改, 因此 \underline{L} 的变化相应于 L 的变化, 即 $\Delta \underline{L} \geq 0$, 从而能保证 L 收敛到局部最大值.

在式(17) 中, 因为 $(K + \mathbf{1}I)$ 是半正定的, 所以 $L(\wedge)$ 是一个线性约束下的凸二次优化函数.

定理 3 线性约束下凸二次优化函数的局部最大就是全局最大.

证明 假定式(17) 有一个局部最优解 \wedge_1, \wedge_2 是全局最优解, 则 $L(\wedge_2) > L(\wedge_1)$. 构造一条直线, 将 \wedge_1, \wedge_2 连接起来, 设 $\wedge(\lambda) = (1 - \lambda)\wedge_1 + \lambda\wedge_2$, 因为约束是线性的, 所以在直线上的点都是凸二次优化函数的解, 则有

$$L(\wedge(\lambda)) > L(\wedge_1).$$

与原假设矛盾, 所以原命题成立.

4.2 OS-LSSVMR 算法的特点与性能分析

OS-LSSVMR 算法具有如下特点:

1) 具有较好的稀疏性. 传统的 LS-SVM 回归不具有稀疏性, 文献[8] 改进的 LS-SVM 虽然具有稀疏性, 但其算法需要反复训练.

2) 该算法的复杂度是训练过程中时间的函数, 而其训练时间与参加训练的样本数目和算法结构有关. OS-LSSVMR 算法的结构与 LS-SVM 相类似, 只是训练样本不同, 因而两者复杂度的比较取决于训练样本的数目. LS-SVM 算法的复杂度为 $O(l)$, l 为样本的总数, 而 OS-LSSVMR 的复杂度为 $O(m)$, m 为样本字典的样本数目. 通常, $m \ll l$, 所以后者的复杂度较低.

3) 采用贪婪方式序贯加入新样本, 也就是后加入的样本不影响旧样本的矩阵 A, K, \tilde{K} 的参数, 适合在线训练.

5 仿真实验及分析

采用在线稀疏最小二乘支持向量机回归拟合 Sinc 函数 $\sin(x)/x$ 来说明 OS-LSSVMR 的特性, $x \in [-10, 10]$. OS-LSSVMR 的参数选择如下: RBF 核函数, $\gamma = 3, \sigma = 0.001, \tau = 0, \epsilon = 0.02$; 与其相比较的是稀疏在线贪吃支持向量回归 (SOG-SVR) 和序贯支持向量回归 (SEQ-SVR). SOG-SVR 的参数

选择如下:RBF核函数, $\gamma = 3$, $\sigma = 0.001$, $\epsilon = 0$, $\rho = 0.01$, $C = 0.001$. SEQ-SVR的参数选择如下:RBF核函数, $\gamma = 3$, $\sigma = 0.01$, $C = 0.001$. 采用两个实验来说明 OS-LSSVMR 拟合的效果.

第1个实验的样本数据输入 $x \in [-10, 10]$ 是随机产生的正态分布数据. 在没有噪声干扰的情况下, 样本数据集的大小从1个数据开始, 对于 SOG-SVR 和 OS-LSSVMR 数据集的大小逐渐增加到500, 对于 SEQ-SVR 数据集的大小逐渐增加到270. 每个训练集模拟在线训练时采用样本数据逐一加入的训练方式, 其训练结果如图1~图3所示. 图1表示的是随着样本数目的增加, 3种支持向量机回归的均方误差 MSE 的对数值 $\lg(\text{MSE})$ 变化趋势. 由图可知, 在没有噪声干扰的情况下, SEQ-SVR 具有最快的收敛速度和最好的收敛精度, 其次是 OS-LSSVMR, 最后是 SOG-SVR. 图2表示的是3种训练方式所用的时间. 可以看出, SEQ-SVR 训练需要的时间增长速度远远超过 SOG-SVR 和 OS-LSSVMR, OS-LSSVMR 和 SOG-SVR 都具有较快的训练速度, 但 OS-LSSVMR 比 SOG-SVR 须消耗的时间更少. 图3表示的是 SOG-SVR, OS-LSSVMR 和 SEQ-SVR 训练结果中支持向量数占总样本数的百分比. 由图可知, OS-LSSVMR 和 SOG-SVR 具有较好的稀疏性.

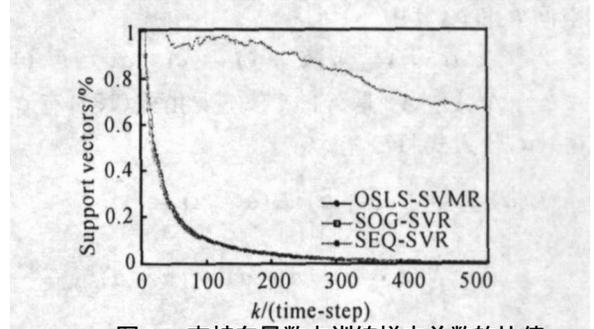


图3 支持向量数占训练样本总数的比值

500, 对于 SEQ-SVR 数据集的大小逐渐为270. 加入均值为0的白噪声, 噪声的偏差从0.01逐渐增加到1, 增量为0.01. 模拟在线训练时采用样本数据逐一加入的训练方式, 实验结果如图4和图5所示. 由图可知, 随着噪声偏差的增加, SEQ-SVR 的拟合效果比 OS-LSSVMR 和 SOG-SVR 的效果差, 训练时间更长, 说明 OS-LSSVMR 对 SOG-SVR 抗噪声能力较 SEQ-SVR 强, 即具有较好的鲁棒性.

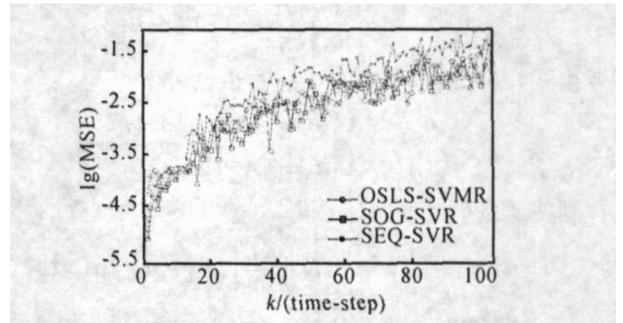


图4 有噪声的均方误差 MSE 的对数值

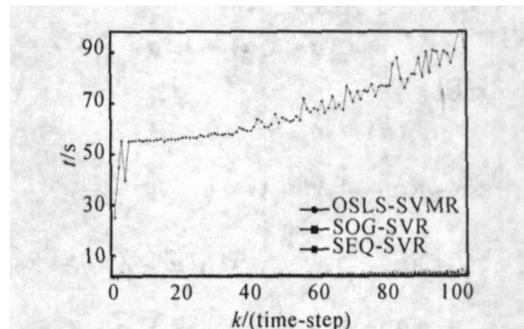


图5 有噪声训练的时间

图6和图7是关于回归的实际效果图. 从图中可以看出, 在噪声较小的情况下3种算法的效果相近, 但在噪声较大的情况下 SEQ-SVR 效果较差,

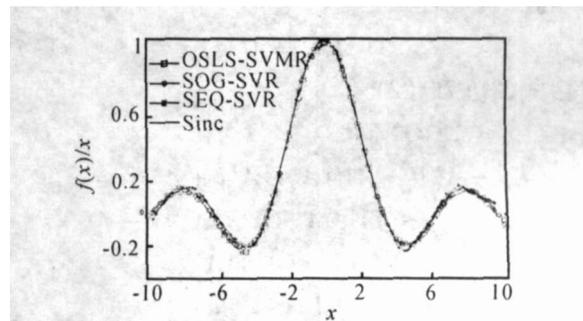


图6 噪声方差为0.1回归的实际效果图

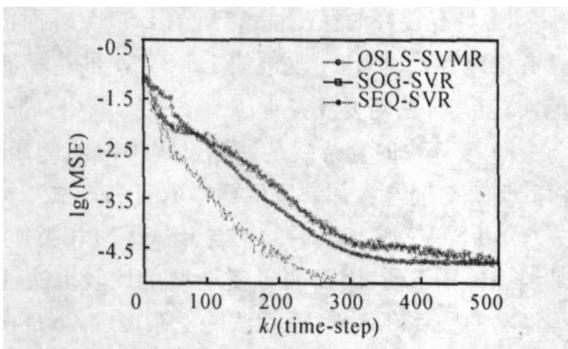


图1 无噪声的 MSE 的对数值

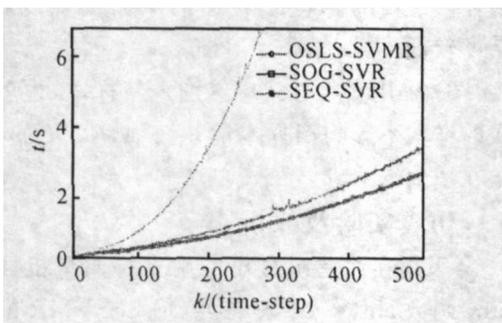


图2 无噪声训练所消耗的时间

第2个实验的样本数据输入 $x \in [-10, 10]$ 是随机产生的正态分布数据. 固定样本数据集的大小, 对于 SOG-SVR 和 OS-LSSVMR 数据集的大小为

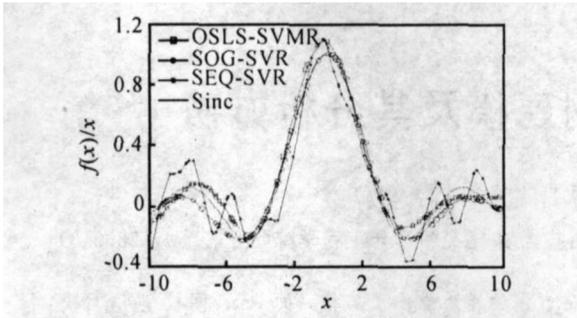


图 7 噪声方差为 0.5 回归的实际效果图

OS-LSSVMR 相对 SOG-SVR 效果较好.

6 结 语

最小二乘支持向量机回归将传统支持向量机回归的不等式约束变成等式约束,使传统支持向量机的二次规划优化方法变成了解线性方程的优化方法,然而这种方法不具有稀疏性.在线稀疏最小二乘支持向量机回归方法采用样本字典的方法大大减少了支持向量数目,使最小二乘支持向量机具有较好的稀疏性.与 SOG-SVR 相比,在线稀疏最小二乘支持向量机回归在收敛速度上具有稍好的效果,但需调整的参数较少,使用起来相对简单,因而可进一步应用于在线实时建模与控制等方面的研究.

参考文献(References)

[1] Syed N A, Liu H, Sung K K. Incremental learning with support vector machines [C]. Proc of Workshop on Support Vector Machines at the Int Joint Conf on Artificial Intelligence. Sweden, 1999: 313-321.
 [2] Cauwenberghs G, Poggio T. Incremental and decremental support vector machines learning [C]. Advances in Neural Information Systems. Cambridge: MIT Press, 2001: 409-415.

[3] Ma Junshui, Theiler James, Perkins Simon. Accurate on-line support vector regression [J]. Neural Computation, 2003, 15: 2683-2703.
 [4] Martin Mario. On-line support vector machine regression [C]. European Conf on Machine Learning. Berlin: Springer-Verlag, 2002: 282-294.
 [5] Frie T T, Cristianini N, Campbell C. The kernel adatron algorithm: A fast and simple learning procedure for support vector machines [C]. Proc of 15th Int Conf Machine Learning. Morgan: Kaufmann Publishers, 1998: 188-196.
 [6] Vijayakumar S, Wu S. Sequential support vector classifiers and regression [C]. Proc Int Conf on Soft Computing. Genca, 1999: 610-619.
 [7] Jyrki Kivinen, Alexander J Smola, Robert C Williamson. Online learning with kernels [J]. IEEE Trans on Signal Processing, 2004, 52(8): 2165-2176.
 [8] Suykens J A K, Lukas L, Vandewalle J. Sparse approximation using least squares support vector machines [C]. IEEE Int Symposium on Circuits and Systems. Geneva, 2000: 11757-11760.
 [9] Yaakov E, Shie M, Ron M. Sparse online greedy support vector regression [C]. Proc of European Conf on Machine Learning. Berlin: Springer-Verlag, 2002: 84-96.
 [10] Down T, Gates K E, Masters A. Exact simplification of support vector solutions [J]. J of Machine Learning Research, 2001, 2: 293-297.
 [11] 黄琳. 系统与控制理论中的线性代数 [M]. 北京: 科学出版社, 1984.
 (Huang L. Linear algebra of system and control theory [M]. Beijing: Science Press, 1984.)

(上接第 131 页)

[37] Bendtsen C N, Krink T. Phone routing using the dynamic memory model [C]. Proc of the 2002 Congress on Evolutionary Computation. Piscataway: IEEE Service Center, 2002: 992-997.
 [38] Branke J, Kaubler T, Schmidt C, et al. A multi-population approach to dynamic optimization problems [C]. Adaptive Computing in Design and Manufacturing 2000. Berlin: Springer-Verlag, 2000: 299-308.
 [39] Oppacher F, Wineberg M. The shifting balance genetic algorithm: Improving the GA in a dynamic environment [C]. Proc of Genetic and Evolutionary Computation Conf. San Francisco: Morgan Kaufmann Publisher, 1999: 504-510.

[40] Wineberg M, Oppacher F. Enhancing the GA's ability to cope with dynamic environments [C]. Proc of Genetic and Evolutionary Computation Conf. San Francisco: Morgan Kaufmann Publisher, 2000: 3-10.
 [41] Ursem R K. Mutinational GA optimization technique in dynamic environments [C]. Proc of Genetic and Evolutionary Computation Conf. San Francisco: Morgan Kaufmann Publisher, 2000: 19-26.
 [42] Oh S K, Lee C Y, Lee J J. A new distributed evolutionary algorithm for optimization in nonstationary environments [C]. Proc of the 2002 Congress on Evolutionary Computation. Piscataway: IEEE Service Center, 2002: 1875-1882.