云南大学学报（自然科学版），2011，33（6）：621～626　　CN 53－1045/N　ISSN 0258－7971
Journal of Yunnan University　　http://www.yndxxb.ynu.edu.cn

最小二乘双胞支持向量回归机的研究*[1]

胡光华，徐汝争
（云南大学 数学与统计学院，云南 昆明　650091）

摘要：利用最小二乘方法和临近支持向量机（PSVM）算法，并结合双胞支持向量机（TSVR），提出了最小二乘双胞支持向量回归机（LSTSVR）. 作为对照，TSVR 需要求解 2 个二次规划问题，而 LSTSVR 仅需求解 2 个线性方程组. 最后利用不同的实例验证了所提算法的可行性和有效性.

关键词：最小二乘支持向量机；支持向量回归机；双胞支持向量回归机
中图分类号：TP 274　　**文献标识码**：A　　**文章编号**：0258－7971(2011)06－0162－06

SVM（support vector machine 支持向量机）以统计学习理论的 VC 维理论和结构风险最小化原理为理论基础. 在实际的应用中，SVM 用求解一个二次规划问题，避免了神经网络的局部极值问题；对于非线性问题，通过一个映射，把低维空间变换成高维的特征空间，使在样本空间线性不可分的问题最终在高维的特征空间线性可分. 在模式识别、系统建模和时间序列预测等领域得到了广泛的应用.

虽然 SVM 与其他机器学习理论相比有许多自身的优点，然而 SVM 的理论并不够完善，比如训练就需要花费很多的时间，如何更快捷解决高维、大规模数据就是目前所研究的一个重要内容. 随着 SVM 越来越受到关注，对 SVM 的某些问题进行研究，提出了许多改进的 SVM. 比如我们所熟知的 SMO，LIBSVM，GEPSVM，等等. Mangasarian[1] 等提出了 lagrange SVM（LSSVM），该方法避免了用线性规划或二次规划去求解 SVM，当训练样本集规模很大时有很高的训练效率.

然而，上述工作大都只局限于支持向量分类机（SVC），对于支持向量回归（SVR），改进算法却很少，Suyken[2] 介绍了最小二乘支持向量回归机（LSSVR）. 在 LSSVR 算法中，优化指标采用平方项，将不等式约束转变为等式约束，从而将二次规划问题转变成线性方程组的求解问题. 与标准 SVR 算法相比，减少了一个调整参数，减少了多个优化变量，因此简化了计算的复杂性，提高了收敛速度. 但是在鲁棒性方面却不如 SVR. 在 2009 年，彭新俊[3] 把 TSVM 的方法应用到了 SVR. 提出了双胞支持向量回归（TSVR），实验证明双胞支持向量机的计算能力是 SVR 的 4 倍. 在本文中，把最小二乘和双胞支持向量机相结合，提出了最小二乘双胞支持向量回归机，用本文所提出的这种算法，用线性方程组就可以代替双胞支持向量机中的线性规划求解. LSTSVR 的线性方程组仅仅是要求解一个 $(n+1)\times(n+1)$ 的矩阵，n 远远小于训练样本的数量，所以我们的方法将会取得更快的计算速度.

1　最小二乘双胞支持向量回归机

1.1　线性 LSTSVR　Suykens[2] 提出的最小二乘支持向量机算法采用最小二乘线性系统代替标准 SVR 算法的二次规划方法解决模式识别和函数估计问题，减少了多个优化变量，提高了收敛速度，但是缺失了 SVR 中的稀疏性. 因此和 SVR 比起来，LSSVR 的“鲁棒性”欠佳，更多关于 SVR 和 LSSVR 可以参看文献[2,4－6]. Fung 和 Mangasarian[7] 提出了 PSVM，解决原始的二次规划问题可以由求解二次规划的对偶问

* **收稿日期**：2011－01－25
基金项目：国家自然科学基金资助项目(10961027).
作者简介：胡光华(1962－　)，男，云南人，教授，主要从事随机控制、智能学习算法方面的研究.

题的 PSVM 方法来代替,Arun 和 Gopal[8] 也曾利用这样的方法提出了 LSTSVM. 在 2009 年,彭新俊[3] 又把 TSVM 的方法应用到了 SVR,提出了双胞支持向量回归(TSVR),但需求解 2 个二次规划问题.

对于给定的练样本点$(x_1,y_1),\cdots,(x_l,y_l)$,本文的 LSTSVR 的思想为:寻找 2 个超平面$f_1(x)=w_1^{\mathrm{T}}x+b_1$ 和$f_2(x)=w_2^{\mathrm{T}}x+b_2$,它们最多不超过样本点的上界和样本点的下界,而最终的回归估计函数定义为$f(x)=\frac{1}{2}(f_1(x)+f_2(x))$. 在这里,样本点的 ε 上界和样本点的 ε 下界组成了 ε 带,与标准 SVR 中的 ε 带相比较,个别误差较大的野点的影响降低了,从而提高了鲁棒性. 而要确定 2 个超平面$f_1(x)=w_1^{\mathrm{T}}x+b_1$ 和$f_2(x)=w_2^{\mathrm{T}}x+b_2$,只需要求解 2 个线性方程组即可,与 SVR 求解规模较大的二次规划问题及 TSVR 求解 2 个二次规划问题相比,极大地提高了运算的速度.

在本文中,记$X=(x_1^{\mathrm{T}},x_2^{\mathrm{T}},\cdots,x_l^{\mathrm{T}})^{\mathrm{T}}\in \mathrm{R}^{l\times n}$ 是给定样本组成的矩阵,$Y=(y_1;y_2;\cdots;y_l)$ 是相应的回归量,其中 $y_i\in \mathbf{R}$. 在实空间中,元素都是 1 的向量记为 e.

对于线性 LSTSVR,给定训练样本点$(x_1,y_1),\cdots,(x_l,y_l)$,目的是找到 2 个超平面$f_1(x)=w_1^{\mathrm{T}}x+b_1$ 和$f_2(x)=w_2^{\mathrm{T}}x+b_2$,最终得到回归函数$f(x)=\frac{1}{2}(f_1(x)+f_2(x))=\frac{1}{2}(w_1+w_2)^{\mathrm{T}}x+\frac{1}{2}(b_1+b_2)$. 寻找$f(x)$可以转化成求解下列最小问题:

$$\min \frac{1}{2}(Xw_1+eb_1)^{\mathrm{T}}(Xw_1+eb_1)+\frac{1}{2}C_1\xi, \tag{1}$$

$$s.t.\ Y-(Xw_1+eb_1)=e\varepsilon_1-\xi;$$

$$\min \frac{1}{2}(Xw_2+eb_2)^{\mathrm{T}}(Xw_2+eb_2)+\frac{1}{2}C_2\xi^{*\mathrm{T}}\xi^{*}, \tag{2}$$

$$s.t.\ (Xw_2+eb_2)-Y=e\varepsilon_2-\xi^{*};$$

其中 C_1,C_2 是惩罚系数,ξ,ξ^{*} 是松弛变量.

定理 1 记$\boldsymbol{E}=[X\quad e]$,若$\boldsymbol{E}^{\mathrm{T}}\boldsymbol{E}$ 可逆. 则问题(1) 和(2) 分别关于向量$[w_1^{\mathrm{T}}b_1]^{\mathrm{T}}$ 和向量$[w_2^{\mathrm{T}}b_2]^{\mathrm{T}}$ 的解为

$$[w_1^{\mathrm{T}}\quad b_1]^{\mathrm{T}}=\boldsymbol{Y}_1(\boldsymbol{E}^{\mathrm{T}}\boldsymbol{E})^{-1}\boldsymbol{E}^{\mathrm{T}}(Y-e\varepsilon_1), \tag{3}$$

$$[w_2^{\mathrm{T}}\quad b_2]^{\mathrm{T}}=\boldsymbol{Y}_2(\boldsymbol{E}^{\mathrm{T}}\boldsymbol{E})^{-1}\boldsymbol{E}^{\mathrm{T}}(Y-e\varepsilon_2), \tag{4}$$

其中$\boldsymbol{Y}_1=\frac{C_1}{C_1+1},\boldsymbol{Y}_2=\frac{C_2-1}{C_2}$.

证明 关于(1) 的证明如下,(2) 有类似的证明.

把式(1) 中的约束条件代入目标函数得到下面的无约束优化问题

$$\min \frac{1}{2}\|Xw_1+eb_1\|^2+\frac{C_1}{2}\|(Xw_1+eb_1)+e\varepsilon_1-Y\|^2, \tag{5}$$

求(5) 式关于 w_1 和 b_1 的梯度,并令其为 0,可得

$$X^{\mathrm{T}}(Xw_1+eb_1)+C_1X^{\mathrm{T}}(Xw_1+eb_1+e\varepsilon_1-Y)=0e, \tag{6}$$

$$e^{\mathrm{T}}(Xw_1+eb_1)+C_1e^{\mathrm{T}}(Xw_1+eb_1+e\varepsilon_1-Y)=0; \tag{7}$$

为了得到 w_1 和 b_1,我们把(6) 和(7) 式写成矩阵形式

$$\frac{1}{C_1}\begin{bmatrix}X^{\mathrm{T}}X & X^{\mathrm{T}}e\\ e^{\mathrm{T}}X & e^{\mathrm{T}}e\end{bmatrix}\begin{bmatrix}w_1\\ b_1\end{bmatrix}+\begin{bmatrix}X^{\mathrm{T}}X & X^{\mathrm{T}}e\\ e^{\mathrm{T}}X & e^{\mathrm{T}}e\end{bmatrix}\begin{bmatrix}w_1\\ b_1\end{bmatrix}=\begin{bmatrix}X^{\mathrm{T}}(Y-e\varepsilon_1)\\ e^{\mathrm{T}}(Y-e\varepsilon_1)\end{bmatrix},$$

$$\begin{bmatrix}w_1\\ b_1\end{bmatrix}=\frac{C_1}{C_1+1}\begin{bmatrix}X^{\mathrm{T}}X & X^{\mathrm{T}}e\\ e^{\mathrm{T}}X & e^{\mathrm{T}}e\end{bmatrix}\begin{bmatrix}X^{\mathrm{T}}(Y-e\varepsilon_1)\\ e^{\mathrm{T}}(Y-e\varepsilon_1)\end{bmatrix}=\frac{C_1}{C_1+1}\left(\begin{bmatrix}X^{\mathrm{T}}\\ e^{\mathrm{T}}\end{bmatrix}[X\quad e]\right)^{-1}\begin{bmatrix}X^{\mathrm{T}}\\ e^{\mathrm{T}}\end{bmatrix}(Y-e\varepsilon_1);$$

定义$\boldsymbol{E}=[X\quad e]$ 和$\boldsymbol{Y}_1=\frac{C_1}{C_1+1}$,方程(1) 的解为$[w_1^{\mathrm{T}}\quad b_1]^{\mathrm{T}}=\boldsymbol{Y}_1(\boldsymbol{E}^{\mathrm{T}}\boldsymbol{E})^{-1}\mathbf{E}^{\mathrm{T}}(Y-e\varepsilon_1)$. 证毕.

注意到实际计算中$\boldsymbol{E}^{\mathrm{T}}\boldsymbol{E}$ 可能不可逆,本文可以引入正则化项$\varepsilon\boldsymbol{I}(\varepsilon\geqslant 0)$ 以避免$\boldsymbol{E}^{\mathrm{T}}\boldsymbol{E}$ 不可逆,则(3) 和

(4) 转化为

$$[w_1^{\mathrm{T}} \quad b_1]^{\mathrm{T}} = Y_1(\boldsymbol{E}^{\mathrm{T}}\boldsymbol{E} + \varepsilon I)^{-1}\boldsymbol{E}^{\mathrm{T}}(Y - e\varepsilon_1), \tag{8}$$

$$[w_2^{\mathrm{T}} \quad b_2]^{\mathrm{T}} = Y_2(\boldsymbol{E}^{\mathrm{T}}\boldsymbol{E} + \varepsilon I)^{-1}\boldsymbol{E}^{\mathrm{T}}(Y - e\varepsilon_2), \tag{9}$$

最终的回归估计函数为

$$f(x) = \frac{1}{2}(f_1(x) + f_2(x)) = \frac{1}{2}(w_1 + w_2)^{\mathrm{T}}x + \frac{1}{2}(b_1 + b_2). \tag{10}$$

综上即可得到本文提出的线性最小二乘双胞支持向量的算法线性——LSTSVR 算法:

(1) 选择惩罚系数 C_1, C_2. 从而得到 $Y_1 = \dfrac{C_1}{C_1 + 1}, Y_2 = \dfrac{C_2 - 1}{C_2}$. 选取上下界参数 ε_1 及 ε_2,及正则化项参数 ε.

(3) 利用(8) 式和(9) 式得到 2 个超平面的参数$[w_1^{\mathrm{T}} \quad b_1]^{\mathrm{T}}$ 和$[w_2^{\mathrm{T}} \quad b_2]^{\mathrm{T}}$.

(4) 利用(10) 式计算最终回归估计函数$f(x)$.

注意到(8) 和(9) 式,本文提出的算法仅仅计算 2 个线性方程组,最终是计算 2 个维数是$(n + 1) \times (n + 1)$ 的逆矩阵,其中 n 远远小于训练集的样本数,所以本文的算法与 SVR,LSSVR 及 TSVR 相比将会取得更快的收敛速度. 另外,LSTSVR 算法由于引入了 ε_1 和 ε_2 2 个参数,使得我们所提出的算法和 SVR 及 LSSVR 相比"鲁棒性" 也将会更好.

1.2　非线性 LSTSVR　在本节中,利用含有核函数的非线性回归估计函数代替线性回归估计函数,我们把线性 LSTSVR 推广到非线性的情况. 类似于线性的情况,我们得到非线性回归估计函数:

$$f_1(x) = K(x, X^{\mathrm{T}})w_1 + b_1, f_2(x) = K(x, X^{\mathrm{T}})w_2 + b_2;$$

其中$K(x, X^{\mathrm{T}})$ 是核函数向量,定义为$K(x, U) = (k(x, u_1), k(x, u_2), \cdots, k(x, u_m))$,$U = (u_1, u_2, \cdots, u_m) \in \mathbf{R}^{n \times m}$.

非线性 LSTSVR 求解问题转化成下列最小问题

$$\min \frac{1}{2}(K(X, X^{\mathrm{T}})w_1 + eb_1)^{\mathrm{T}}(K(X, X^{\mathrm{T}})w_1 + eb_1) + \frac{1}{2}C_1\xi^{\mathrm{T}}\xi,$$

$$s.t. \quad Y - (K(X, X^{\mathrm{T}})w_1 + eb_1) = e\varepsilon_1 - \xi; \tag{11}$$

$$\min \frac{1}{2}(K(X, X^{\mathrm{T}})w_2 + eb_2)^{\mathrm{T}}(K(X, X^{\mathrm{T}})w_2 + eb_2) + \frac{1}{2}C_2\xi^{*\mathrm{T}}\xi^{*},$$

$$s.t. \quad (K(X, X^{\mathrm{T}})w_2 + eb_2) - Y = e\varepsilon_2 - \xi^{*}; \tag{12}$$

类似于线性的情况,有如下定理 2:

定理 2　记 $\boldsymbol{H} = [K(X, X^{\mathrm{T}})e]$,若 $\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H}$ 可逆,问题(11) 和(12) 分别关于向量$[w_1^{\mathrm{T}} \quad b_1]^{\mathrm{T}}$ 和向量$[w_2^{\mathrm{T}} \quad b_2]^{\mathrm{T}}$ 的解为

$$[w_1^{\mathrm{T}} \quad b_1]^{\mathrm{T}} = Y_1(\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}(Y - e\varepsilon_1), \tag{13}$$

$$[w_2^{\mathrm{T}} \quad b_2]^{\mathrm{T}} = Y_2(\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}(Y - e\varepsilon_2); \tag{24}$$

其中 $Y_1 = \dfrac{C_1}{C_1 + 1}, Y_2 = \dfrac{C_2 - 1}{C_2}$.

引入正则化项 $\varepsilon\boldsymbol{I}(\varepsilon \geqslant 0)$ 以避免 $\boldsymbol{E}^{\mathrm{T}}\boldsymbol{E}$ 不可逆,(13) 和(14) 式转化为

$$[w_1^{\mathrm{T}} \quad b_1]^{\mathrm{T}} = Y_1(\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H} + \varepsilon\boldsymbol{I})^{-1}\boldsymbol{H}^{\mathrm{T}}(Y - e\varepsilon_1), \tag{15}$$

$$[w_2^{\mathrm{T}} \quad b_2]^{\mathrm{T}} = Y_1(\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H} + \varepsilon\boldsymbol{I})^{-1}\boldsymbol{H}^{\mathrm{T}}(Y - e\varepsilon_2); \tag{16}$$

最终的回归估计函数为

$$f(x) = \frac{1}{2}(f_1(x) + f_2(x)) = \frac{1}{2}(w_1 + w_2)^{\mathrm{T}}K(X, x) + \frac{1}{2}(b_1 + b_2). \tag{17}$$

非线性 LSTSVR 算法:

(1) 选择核函数,得到 $\boldsymbol{H} = [K(X, X^{\mathrm{T}})e]$;

(2) 选择惩罚系数 C_1, C_2,得到 $\Upsilon_1 = \dfrac{C_1}{C_1 + 1}, \Upsilon_2 = \dfrac{C_2 - 1}{C_2}$. 选取上下界参数 $\varepsilon_1, \varepsilon_2$,及正则化项参数 ε;

(3) 利用(15) 式和(16) 式得到 2 个超平面的参数 $[w_1^{\mathrm{T}} \quad b_1]^{\mathrm{T}}$ 和 $[w_2^{\mathrm{T}} \quad b_2]^{\mathrm{T}}$;

(4) 利用(17) 式计算最终回归估计函数 $f(x)$.

类似于线性 LSSVR,本文所提出的方法在解决非线性 LSTSVR 问题上也仅仅是要求解 2 个加入核函数的线性方程组,提高了计算速度. 同时也引入了不超过样本点的 ε_1 和 ε_2 上下界,组成了未知测试集的 ε 带,提高了"鲁棒性".

2 实验和讨论

为了测试 LSTSVR 算法的可行性和有效性,本节中我们利用一个构造的实际函数和 3 个 UCI 数据和 LSSVR 及 TSVR 相比较. 试验中我们只使用高斯函数 $K(u^{\mathrm{T}}, v) = \exp(-\theta \|u - v\|^2)$,其中参数 $\theta > 0$. 在没有特殊的说明下,l 表示训练样本的数目,y_i 表示训练样本 x_i 的真实对应输出值,$\hat{y}_i$ 是期望输出,$\bar{y} = \dfrac{1}{n}\sum_{i=1}^{n} y_i$ 表示 $y_1, \cdots, y_n$ 的平均值. 定义 $\mathrm{MAT} = \dfrac{1}{n}\left(\sum_{i=1}^{n} |\hat{y}_i - y_i|\right)$,$\mathrm{SSE/SST} = \dfrac{\sum_{i=1}^{n} (\hat{y}_i - y_i)^2}{\sum_{i=1}^{n} (\bar{y}_i - y_i)^2}$,$\mathrm{SSR/SST} = \dfrac{\sum_{i=1}^{n} (\hat{y}_i - \bar{y}_i)^2}{\sum_{i=1}^{n} (\bar{y}_i - y_i)^2}$. SSE 越小,估计值将会越好,但是过小,也将会造成过拟合的问题. 而 SSR 越大,估计值也将会越好,同样要避免过拟合.

表 1 LSSVR 和 LSTSVR 在各种噪声下的 sinex 数据的比较结果

Tab. 1 Result comparisons of LSSVR and LSTSVR on sinex datas with different types of noises

噪声类型	回归机	SSE/SST	SSR/SST	MAT	CPU 时间/s
A	LSSVR	0.034 2	0.790 4	0.025 3	2.295 761
	LSTSVR	0.017 5	0.906 1	0.021 9	2.239 423
B	LSSVR	0.073 1	0.852 3	0.046 3	2.307 575
	LSTSVR	0.0654	1.004 9	0.042 9	2.238 287
C	LSSVR	0.063 9	1.003 7	0.032 9	2.294 776
	LSTSVR	0.030 6	1.096 8	0.028 5	2.209 751
D	LSSVR	0.097 6	1.013 7	0.053 9	2.300 951
	LSTSVR	0.091 5	1.190 6	0.053 7	2.222 543

例1 有训练样本 (x_i, y_i),$i = 1, \cdots, l$,l 是样本的个数. Sinex 函数可定义为 $y = \mathrm{sinex}(x) = \mathrm{e}^{-\frac{x}{2.5}}\sin 3x$,$x \in [0, \ 4\pi]$,在训练样本中,分别加入不同的零平均值高斯噪声和均匀分布噪声,加入噪声 Sinex 后的函数为:$y_i = \mathrm{sinex}(x_i) + \xi_i$,(a) $\xi_i \sim N(0, \ 0.1^2)$,(b) $\xi_i \sim N(0, \ 0.2^2)$,(c) $\xi_i \sim U[-0.2, \ 0.2]$,(d) $\xi_i \sim U[-0.4, \ 0.4]$.

其中 $x_i \sim U[0, 4\pi]$,$N(c, d^2)$ 和 $U[a, b]$ 分别是均值为 0 和方差为 d^2 的高斯分布随机数及区间为 (a, b) 的均匀分布随机数. 在不同噪声下的 LSSVR 和 LSTSVR 仿真图像见图 1. 表 1 显示了在不同噪声下的平均比较结果. 可以看出无论是 CPU 时间上,还是各种衡量标准上都充分体现出 LSTSVR 的有效性.

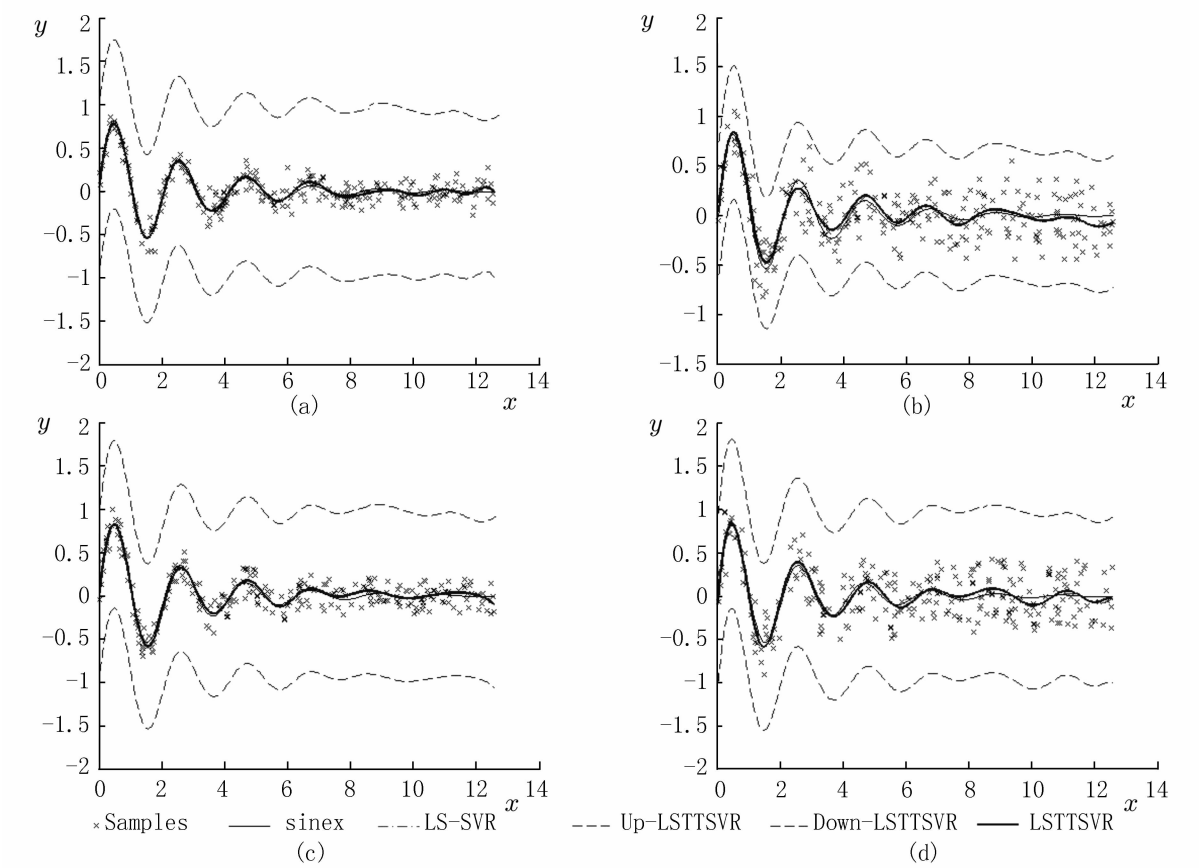

图 1　LSSVR 和 LSTSVR 在不同噪声下的 Sinex 回归曲线,其中(a) $\xi_i \sim N(0,\ 0.1^2)$,(b) $\xi_i \sim N(0,\ 0.2^2)$,(c) $\xi_i \sim U[-0.2,\ 0.2]$,(d) $\xi_i \sim U[-0.4,\ 0.4]$

Fig. 1 Predictions of LSTSVR, LSSVR and TSVR on Sinex function with different types of noises, where type (a) $\xi_i \sim N(0,\ 0.1^2)$, (b) $\xi_i \sim N(0,\ 0.2^2)$, (c) $\xi_i \sim U[-0.2,\ 0.2]$, (d) $\xi_i \sim U[-0.4,\ 0.4]$

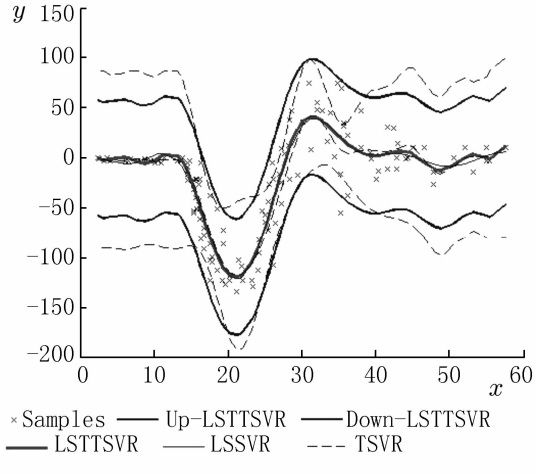

图 2　LSSVR, TSVR 及 LSTSVR 在 Motorcycle 数据下的回归曲线

Fig. 2 Predictions of LSTSVR, LSSVR and TSVR on the motorcycle dataset

例 2　为了更好地评价 LSTSVR 的有效性,我们选取了 4 个 UCI 数据进行比较. 这几个数据分别是 Motorcycle, Boston housing, Auto – MPG 和 Glassidentification[8]. 对于 Motorcycle 我们选择了 3 种方式进行比较对照,包括了 LSSVR, TSVR 和我们的 LSTSVR. 图 2 给出了模拟图像,表明了 LSTSVR 较为优越. 对于其他 3 个数据,我们采用了 LSSVR 和 LSTSVR 进行对照,表 2 记录了各种评价体系下的比较结果,很好地说明了 LSTSVR 算法的可行性及有效性.

在本文中,我们分析了标准 SVR 的改进算法 LSSVR 和改进的回归算法 TSVR,2 种方法的优良性和欠缺不足,提出了 LSTSVR,把以上 2 种方法加以巧妙的融合,用 2 个线性方程组代替了 2 个二次规划问题,把复杂的求解运算转化成了求解 2 个维数是 $(n+1)\times(n+1)$ 的逆矩阵求解问题,n 只是训练样本的个数,要远远的小于样本数 l,使计算的复

杂性大大降低,不但提高了运算收敛的速度,而且提高了回归问题的"鲁棒性".理论和实验均充分证明了改进的 LSTSVR 算法的可行性.然而,LSTSVR 并没有摆脱参数选择的问题,如何把参数的选择个数进一步的降低,比如用 v – SVM 代替标准的 SVM,将是未来要研究的一个方向.

表 2 LSSVR 和 LSTSVR 在 Boston housing,Auto – MPG 和 Glassidentification 数据下的比较结果

Tab. 2 Result comparisons of LSSVR and LSTSVR Boston housing, Auto. Mpg, Glassidentification

数据	回归机	SSE	SSR	CPU 时间/s
Boston housing	LSSVR	475.777 8	3.643 0 e +004	5.906 414
	LSTSVR	0.015 6	4.271 5 e +004	5.491 188
Auto – MPG	LSSVR	5.827 2	202.887 9	4.480 216
	LSTSVR	0.006 1	250.277 9	3.007 056
Glass Identification	LSSVR	1.662 7	920.345 2	0.727 628
	LSTSVR	0.004 7	942.660 0	0.694 658

参考文献:

[1] MANGASARIAN O L, MUSICANT D R. Lagrangian support vector machines[J]. Machine Learning 2001, 1(3): 161-177.

[2] SUYKENS J A K, VANDEWALLE J. Least squares support vector machine classifiers[J]. Neural Process Letter, 1999, 9(3): 293-300.

[3] PENG X. TSVR: An efficient Twin Support Vector Machine for regression[J]. Neural Network, 2010, 23: 365-372.

[4] VAPNIK V. An overview of statistical learning theory[J]. IEEE Trans Neural Networks, 1999, 10(5): 988-999.

[5] CORTES C, VAPNIK V. Support vector networks[J]. Machine Learning, 1995, 20(3): 273-297.

[6] 彭新俊,胡光华. 密度函数估计的修正 SVM 法[J]. 云南大学学报:自然科学版, 2004, 26(4): 284-287.

[7] FUNG G, MANGASARIAN O L. Proximal support vector machine classifiers, in: D. Lee, et al. (Eds.), Proceedings of KDD – 2001: Knowledge Discovery and Data Mining, San Francisco, CA, Association for Computing Machinery, New York, 2001, 77-86.

[8] ARUN K M, GOPAL M. Least squares twin support vector machines for pattern classification[J]. Expert Systems with Applications, 2009, 36: 7 535-7 543.

[9] NEWMAN D J. UCI Repository of machine learning database [DB/OL]. http://www.ics.uci.edu/ ~ mleam/MLRepository.html.

Least squares twin support vector machines for regression

HU Guang-hua, XU Ru-zheng

(Department of Mathematics, Yunnan University, Kunming 650091, China)

Abstract: Using least squares methods and PSVM, combining with twin support vector machine (TSVR), the least squares twin SVR for regression (LSTSVR) is proposed. For comparison, the LSTSVR is only need to solve two linear equations instead of two quadratic programming problems in TSVR. Finally, the simulated experiments confirm the validity and the efficiency of LSTSVR.

Key words: least squares support vector machine; support vector regression; twin support vector machine for regression