r–SVR 的 PSO 求解方法

方益民 [1],徐保国 [2]

FANG Yi-min[1],XU Bao-guo[2]

1.江南大学 信息学院,江苏 无锡 214063

2.江南大学 信控学院,江苏 无锡 214063

1.The School of Information Technology at Jiangnan University,Wuxi,Jiangsu 214063,China

2.The School of Communications and Control Engineering at Jiangnan University,Wuxi,Jiangsu 214063,China

E-mail:fang_yi_min@jiangnan.edu.cn

FANG Yi-min,XU Bao-guo.PSO method for *r*-SVR.Computer Engineering and Applications,2009,45(29):41-42.

Abstract: When r in r-SVR is less than 1,previous Newton descent method cannot be used for r-SVR.Hence,PSO algorithm is proposed to be used to solve r-SVR.In this paper,the fitness function is used in PSO for r-SVR and the relevant PSO algorithm is derived.Experimental results confirm that this method is effective.

Key words: support vector regression;r-loss function;particle swarm optimization

摘 要:针对 r 接近 1 时牛顿法不可求解 r-SVR 的问题,提出了 r-SVR 的 PSO 求解方法,推导出了 PSO 求解 r-SVR 的适应函数,给出了 PSO 求解 r-SVR 的算法和实验结果。结果表明,当 r 接近 1 时,牛顿法求解 r-SVR 失效,而 PSO 求解 r-SVR 的方法是有效的。

关键词:支持向量回归机;r 范数损失函数;粒子群优化算法

DOI:10.3778/j.issn.1002-8331.2009.29.012 **文章编号**:1002-8331(2009)29-0041-02 **文献标识码**:A **中图分类号**:TP301

1 引言

近年来,支持向量回归机 SVR 在各种回归问题中得到广泛应用。各种 SVR 通常基于三类损失函数,即 ε-不敏感损失函数、Huber 损失函数和 r 范数损失函数[1],而由此构成的SVR分别称为 ε-SVR、Huber-SVR 和 r-SVR。最近的研究结果表明,为了提高 r-SVR 的鲁棒性,r-SVR 中参数 r 的选取与样本的输入噪声有关,即当输入噪声逐渐变大时,r 的取值逐渐接近 1[1-3]。文献[1-2]推导出了 r-SVR 的模型并用牛顿迭代法加以求解。但是,当 r-SVR 中 r 接近 1 时,由于损失函数在 0 点不光滑,其一、二阶导数均不存在,不可用牛顿迭代法求解 r-SVR,甚至无法用梯度下降法求解。而文献[1-2]并未解决当 r 接近 1 时 r-SVR 的求解问题,目前也未见解决这一问题的方法。

为了解决 r 接近 1 时 r-SVR 的求解问题,提出了 r-SVR 的 PSO 求解方法。由于在求解优化问题时,PSO[4]算法不需要相应的目标函数的导数信息,因此当牛顿迭代法受到限制时,用 PSO 算法求解 r-SVR 是一个合理的选择。推导了用于求解 r-SVR 的 PSO 算法的适应函数,给出了相应的 PSO 算法和实验结果,并与牛顿迭代法的实验结果作了比较。结果表明,对于 r 接近 1 的情形,当牛顿法求解 r-SVR 失效时,用 PSO 算法求解 r-SVR 是有效的。

2 r–SVR 及其牛顿迭代法

2.1 r–SVR 简介

先介绍 r-SVR[1]的概念。

考虑用函数

$$f(x)=\langle w,x\rangle+b^1$$

来逼近下列数据集:

$D=\{(x_1,y_1),\cdots,(x_n,y_n)\},x\in R^d,y\in R$,这里 n 为样本数目。

若使用 r 范数损失函数:

$$L_r(f(x)-y)=|f(x)-y|^r$$

最优回归函数可通过求解下式得到:

$$\min\quad \Phi(w,\xi)=\frac{1}{2}\|w\|^2+C\sum_i\xi_i \tag{1}$$

$$\text{s.t.}\quad \begin{cases}|f(x_i)-y_i|^r\leqslant\xi_i,i=1,\cdots,l\\ \xi_i\geqslant 0\end{cases}$$

这里 C 是设定常数,$\xi=(\xi_1,\xi_2,\cdots,\xi_l)$,其中 $\xi_i(i=1,2,\cdots,l)$表示约束系统输出的松弛变量。上式中也可用核函数 $K(w,x)$代替点积$\langle w,x\rangle$。

r 范数损失函数与式(1)一起即构成 r 范数-支持向量回归机 r-SVR。

2.2 r–SVR 解的牛顿迭代法

为对 r-SVR 用牛顿迭代法求解,需要有 r-SVR 的解的形式。文献[2]利用 KKT 条件推导出 r-SVR 解的形式为:

基金项目:国家高技术研究发展计划(863)(the National High-Tech Research and Development Plan of China under Grant No.2006AA10Z248)。

作者简介:方益民(1962-),男,副教授,博士生,主研方向:过程控制与模式识别;徐保国(1950-),男,教授,博士生导师,主研方向:过程控制与优化。

收稿日期:2008-05-28 **修回日期**:2008-07-29

$$\min\ L(\alpha,\alpha^*)=\frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i-\alpha_i^*)(\alpha_j-\alpha_j^*)\langle x_i,x_j\rangle-\sum_{i=1}^{l}(\alpha_i-\alpha_i^*)y_i+\frac{r-1}{r^{\frac{r}{r-1}}C^{\frac{1}{r-1}}}\sum_{i=1}^{l}(\alpha_i+\alpha_i^*)^{\frac{r}{r-1}} \tag{2}$$

$$\text{s.t.}\quad\begin{cases}\alpha_i\geqslant 0,\\ \alpha_i^*\geqslant 0,\ \ i=1,\cdots,l\\ \sum\limits_{i=1}^{l}(\alpha_i-\alpha_i^*)=0\end{cases}$$

注意在式(2)中可以用核函数 $K(x_i,x_j)$代替点积$\langle x_i,x_j\rangle$。

通过构造障碍因子可将上述条件极值问题式(2)转化为无条件极值问题

$$\min_{\alpha,\alpha^*}\quad \varphi(\alpha,\alpha^*)=L(\alpha,\alpha^*)+P(\alpha,\alpha^*)$$

其中 $P(\alpha,\alpha^*)=-N\left(\sum\limits_{i=1}^{l}\log\alpha_i+\sum\limits_{i=1}^{l}\log\alpha_i^*\right)+M\left[\sum\limits_{i=1}^{l}(\alpha_i-\alpha_i^*)\right]^2$($M$、$N$为障碍因子)。

用牛顿法(即二阶梯度法)求解之,迭代公式为:

$$\begin{bmatrix}\alpha_k\\ \alpha_k^*\end{bmatrix}=\begin{bmatrix}\alpha_{k-1}\\ \alpha_{k-1}^*\end{bmatrix}-\left[\nabla^2\phi(\alpha_{k-1},\alpha_{k-1}^*)\right]^{-1}\nabla\phi(\alpha_{k-1},\alpha_{k-1}^*) \tag{3}$$

其中一阶、二阶梯度的推导结果见文献[2]。

3　*R*–SVR 的 PSO 解法

3.1　PSO 算法

粒子群优化算法 PSO[4]是一种仿生优化算法,其主要的计算公式为:

$$\begin{cases}p[k+1]=p[k]+v[k]\\ v[k+1]=v[k]+c1\times r1\times(pbest[k]-p[k])+c2\times r2\times(gbest[k]-p[k])\end{cases} \tag{4}$$

其中,$pbest[k]$是第 k 时刻的局部最优解;$gbest[k]$是第 k 时刻的全局最优解;$c1$ 和 $c2$ 是两个常数,$r1$ 和 $r2$ 是两个[0,1]范围内的随机数。

PSO 算法框架如下:

输入:有待优化的适应函数式;

输出:使适应函数值达到最小时的解;

步骤 1　根据适应函数的特性,确定群体规模 m 和搜索空间维数 D,并初始化群体中粒子的速度和位置;

步骤 2　根据公式(4)计算群体中每一个粒子新的速度和位置;

步骤 3　根据适应函数式计算粒子的适应度;

步骤 4　对每个粒子,将其适应值与其经历过的最好位置 $pbest(i)$作比较,如果较好,则将其作为当前的最好位置 $pbest(i)$,否则继续执行下一步;

步骤 5　对每个粒子,将其适应值与全局所经历的最好位置 $gbest$ 作比较,如果较好,则将其作为当前全局的最好位置$gbest$,否则继续执行下一步;

步骤 6　判断算法是否满足终止条件。若达到终止条件,则算法停止,当前粒子的位置就是所求的适应函数的最优解;否则,返回步骤 2,继续下一循环。

3.2　*r*–SVR 的 PSO 的适应函数设计

用 PSO 求解 r–SVR,关键是设计出可用的适应函数。由于当 $r<1$ 时,式(2)中的最后一项不存在最小值,原来用于牛顿迭代的目标函数不是凸函数,不能用作这里的适应函数。为此,从 r–SVR 的模型式(1)入手,重新设计适应函数。

在 r–SVR 的模型式(1)中,利用

$$w=\sum_{i=1}^{l}(\beta_i-\beta_i^*)\varphi(x_i)$$

可以推导出

$$\frac{1}{2}\|w\|^2=\frac{1}{2}\sum_{i,j=1}^{l}(\beta_i-\beta_i^*)(\beta_j-\beta_j^*)K(x_i,x_j)$$

上式中令$(\beta_i-\beta_i^*)=\alpha_i$,再将 r–SVR 的模型式(1)中的约束条件转化为障碍因子,可得与 r–SVR 的模型式(1)近似等价的适应函数如下:

$$P(\alpha,\xi)=\frac{1}{2}\sum_{i,j=1}^{l}\alpha_i\alpha_jK(x_i,x_j)+C\sum_{i=1}^{l}\xi_i-\sum_{k=1}^{n}(\log(\xi_k)+\log(\xi_k-L_r(f(x_k)-y_k))) \tag{5}$$

式(5)中,$-\log(\cdot)$用于构造障碍项,意即当自变量接近或小于 0 时,函数值会很大。程序实现此障碍项时需略加改造,以免自变量接近或小于 0 时,$-\log(\cdot)$出现错误。

4　PSO 解法的实验结果

实验目的在于比较 r–SVR 的牛顿解法和 PSO 解法对典型样本的回归性能。

取函数 $y=\sin(x)/x$,$x\in[-10,10]$为基准信号,令 x 从−10 至 10,步长为 0.5 产生训练样本集。对此样本集,分别用牛顿解法和 PSO 解法的 r–SVR 进行回归,比较回归结果。实验中,核函数为高斯核函数,标准差为 1。分别取 r 为 1.5、1.005 和 0.5 对应三组实验。

第一组实验旨在比较正常情形下($r>1$)PSO 解法和牛顿解法的回归性能。图 1 给出了这一实验结果。其中左图为牛顿法的求解结果,右图为 PSO 法的求解结果。结果表明两种方法都可以进行求解,而牛顿迭代法的求解速度明显较快。

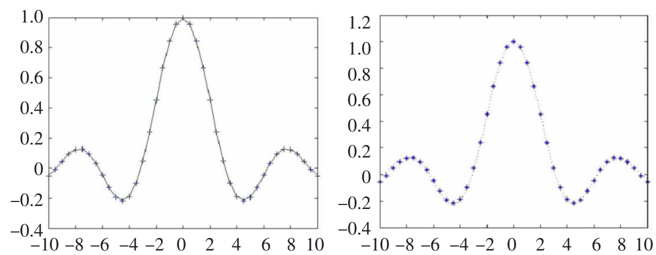

图 1　r=1.5 时牛顿法和 PSO 算法对 r–SVR 的求解结果

第二组实验旨在比较当 r 接近 1 时(r=1.005),PSO 解法和牛顿解法的回归性能。图 2 给出了这一实验结果。同样左图为牛顿法的求解结果,右图为 PSO 法的求解结果。实验结果表明当 r 接近 1 时,牛顿法已经失效,而 PSO 方法可以有效进行回归。

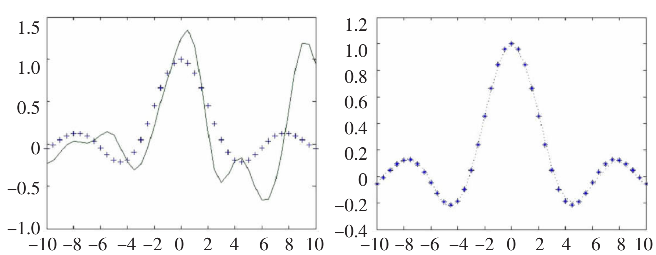

图 2　r=1.005 时牛顿法和 PSO 算法对 r–SVR 的求解结果

(下转 59 页)