

求解非正定核 Huber-SVR 的 SMO 算法

方益民¹ 张 玲¹ 孙为民² 徐保国¹

(江南大学物联网工程学院 无锡 214122)¹ (宝信软件协同商务软件事业部 上海 201900)²

摘 要 通过运用 SMO 分解思想和支持向量回归机 SVR 模型的约束条件,将 SVR 模型的求解问题转化成一系列的给定区间内抛物线的最小值求解问题,对于非正定核而言由于只改变其中部分抛物线的开口方向,因而可以求得其最小值。据此提出了一种可以求解非正定核的 Huber-SVR 的 SMO 方法,推导出了相应的迭代公式并设计了相应的算法。由于用该算法可以求解具有非正定核的 SVR,因此可用具有非正定核的 Huber-SVR 进行回归和预测实验,并与正定核的 Huber-SVR 的实验结果进行比较。实验表明,对于 Huber-SVR 而言,某些非正定核比正定核有更好的回归和预测性能,这说明了求解非正定核的 Huber-SVR 的 SMO 算法的有效性和必要性。这一算法也可以推广到其它 SVR 中。

关键词 非正定核,核方法,SMO 算法,支持向量回归机

中图分类号 TP183 **文献标识码** A

SMO Algorithm for Resolving Huber-SVR with Non-positive Kernels

FANG Yi-min¹ ZHANG Ling¹ SUN Wei-min² XU Bao-guo¹

(School of Things' Internet Engineering, Jiangnan University, Wuxi 214122, China)¹

(Department of Co-business, Baosight Software Co., Ltd, Shanghai 201900, China)²

Abstract A new SMO algorithm for SVR was proposed which can solve the SVR with non-positive kernels. In our SMO algorithm, the problem of solving SVR model is decomposed into a series of sub-problems of seeking the minimum of parabola within a limited range. Such minimum can be found because only the symmetry axis direction of some parabolas is changed as respect to non-positive kernels. Hence, we derived relevant iterative formula of SMO method for Huber-SVR and designed the relevant algorithm. Some necessary proofs about the algorithm were also given. Based on our SMO algorithm, we did both regression experiments and prediction experiments using Huber-SVR with non-positive kernels, and compared the experimental results with that of Huber-SVR with positive kernels. The experimental results showed that some non-positive kernels may have better regression performance and better prediction performance than positive kernels, and this confirmed the validity and necessity of our algorithm. This method can also be extended to the other SVR.

Keywords Non-positive kernel, Kernel method, SMO algorithm, SVR

1 引言

求解具有非正定核的 SVR 模型在理论上存在困难,因为该模型是非凸规划模型,一般认为是不可求解的。相应地,现在几乎所有的关于机器学习核方法的研究也都是基于核函数为正定的,而且核函数必需满足 Mercer 条件。但是在某些特定的应用领域中,所使用的核函数不能保证为正定核函数,如神经网络中的 Sigmoid 核,并且在实际应用中,除了已知的一些核函数外,往往难以检验核函数的正定性。所以希望找到一种方法来解决基于非正定核的机器学习问题,求解具有非正定核的 SVR 模型就是一类这样的问题。

近年来,关于非正定核的研究主要有以下几个方向:第

一,在支持向量分类机 SVM 中解决 Sigmoid 核的非正定问题。文献[1]在文献[5]的基础上,研究了 Sigmoid 核为非正定时训练 SVM 的问题,提出了 Sigmoid 核为非正定时训练 SVM 的 SMO 算法。文献[2]从理论上在欧几里得空间对非正定核的 SVM 分类问题作了几何解释。第二,从核方法的角度引入非正定核,以解决非正定核的核方法的机器学习问题。文献[3]研究了 RKKS 空间(Reproducing Kernel krein Spaces)中的核函数和基于此核函数的机器学习问题,提出了基于 RKKS 核的机器学习优化模型,最优解在目标函数的鞍点上,解决办法是通过近似转换将非凸规划问题变换为凸规划问题,然后加以求解。第三,从核函数的角度,通过在该函数中引入一些参数将非正定核转变为正定核。以上的研究成

到稿日期:2009-09-02 返修日期:2009-10-30 本文受国家 863 项目(2006AA10Z248)资助。

方益民(1962—),男,博士生,副教授,主要研究方向为建模与优化控制,E-mail:fang_yi_min@jiangnan.edu.cn;张 玲(1986—),女,硕士,主要研究方向为人工神经网络等;孙为民(1971—),男,硕士生,工程师,主要研究方向为软件工程、数据挖掘;徐保国(1954—),男,教授,博士生导师,主要研究方向为优化控制。

果虽然都是行之有效的方法,但是也存在一些问题。文献[3]从核方法的角度引入非正定核的机器学习方法,这种方法通用性较好,但是仅对总体回归误差进行了约束,而没有针对各个样本点的回归误差进行约束,所以回归精度较低,泛化性能较差。而通过引入一些参数将非正定核转变为正定核这一方法,目前还未见有重要的研究成果公开发表。

文献[1]提出的用于解决非正定核的 SVC 优化问题的方法,可以保持 SVC 的泛化性能,这引起了我们的兴趣。这一方法的基本思想是,通过运用 SMO 分解思想和支持向量分类器 SVC 模型的约束条件,将 SVC 模型的求解问题转化成一系列的给定区间内抛物线线段的最小值求解问题。而对于给定区间内抛物线线段的最小值求解问题,就非正定核而言由于只改变其中抛物线的开口方向,且抛物线自变量的取值范围是被限定的,因而可以求得其所对应抛物线的最小值,进而可以求解具有非正定核的 SVC 的优化模型。本文在文献[4]的基础上,将文献[1]的方法引入 SVR 模型求解中,以解决具有非正定核的 SVR 求解问题。本文在 SVR 中使用 Huber 损失函数,是因为 Huber-SVR 与其它 SVR 相比,模型比较简单,较易推导,且同样的方法也适用于其它损失函数的 SVR。

本文证明了 Huber-SVR 的 SMO 方法的一些性质,据此推导了用于求解非正定核 Huber-SVR 的 SMO 的迭代公式,证明了迭代公式的正确性,并设计了相应的算法。最后利用本文提出的算法,用具有非正定核的 Huber-SVR 进行了回归和预测实验,并与具有正定核的 Huber-SVR 的回归结果进行了比较。实验表明,就 Huber-SVR 而言,某些非正定核比正定核有更好的回归精度和泛化性能,说明了本文提出的求解非正定核的 Huber-SVR 的 SMO 算法的有效性和必要性。这一算法也可以推广到其它 SVR 中。

本文第 2 节简要介绍 Huber-SVR 模型和非正定核的构造;第 3 节提出求解核函数为非正定的 Huber-SVR 的 SMO 算法,并给出有关的推导和证明;第 4 节分别用非正定核的 Huber-SVR 与标准的正定核的 Huber-SVR 进行回归和预测比较;最后是结论。

2 Huber-SVR 模型与非正定核的构造

2.1 Huber-SVR 模型

当 SVR 中的损失函数取 Huber 函数时,构成 Huber-SVR。其模型如下:

$$\min \Phi(w, \xi) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^l (\xi_i^- + \xi_i^+) \quad (1)$$

$$\text{s. t. } \begin{cases} L_{\text{Huber}}(y_i - f(x_i)) \leq \xi_i^-, y_i \geq f(x_i) \\ L_{\text{Huber}}(f(x_i) - y_i) \leq \xi_i^+, f(x_i) > y_i, i=1, \dots, l \\ \xi_i^-, \xi_i^+ \geq 0 \end{cases}$$

将 Huber 函数

$$L_{\text{Huber}}(f(x) - y) = \begin{cases} (f(x) - y)^2 / 4\mu, & |f(x) - y| \leq 2\mu \\ |f(x) - y| - \mu, & \text{其他} \end{cases}$$

代入上述约束条件,则上述模型可以进一步化为如下的二次规划问题:

$$\min F(\alpha) = \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l \alpha_i \alpha_j K(x_i, x_j) - \sum_{i=1}^l \alpha_i y_i + \frac{1}{2C} \sum_{i=1}^l \alpha_i^2 \mu \quad (2)$$

$$\text{s. t. } \begin{cases} -C \leq \alpha_i \leq C, i=1, \dots, l \\ \sum_{i=1}^l \alpha_i = 0 \end{cases}$$

式中, $K(\cdot, \cdot)$ 是核函数。在求出上式中的参数 α 后,即可求得回归函数 $f(x)$ 。

2.2 Krein 空间与非正定核的构造^[3]

文献[3]从理论上提出了非正定核的构造,这是目前对非正定核研究的较完整的成果,现作简要介绍。

定义 1(内积空间) 设 K 是一个 $R^n \rightarrow R$ 的向量空间。一个 K 上的内积空间 $\langle \cdot, \cdot \rangle_K$, 对于 $\forall f, g, h \in K, \alpha \in \mathbb{R}$, 具有如下双线性形式:

$$\begin{aligned} \langle f, g \rangle_K &= \langle g, f \rangle_K \\ \langle \alpha f + g, h \rangle_K &= \alpha \langle f, h \rangle_K + \langle g, h \rangle_K \\ \langle f, g \rangle &= 0, \forall g \in K \Rightarrow f = 0 \end{aligned}$$

定义 2(Krein 空间) 一个内积空间是一个 Krein 空间,如果存在由 K 的线性组合构成的两个 Hilbert 空间 H_+ , H_- , 则使得:

$\forall f \in K, f$ 总可以分解为 $f = f_+ + f_-$, 其中 $f_+ \in H_+$, $f_- \in H_-$, $\forall f, g \in K, \langle f, g \rangle_K = \langle f_+, g_+ \rangle_{H_+} - \langle f_-, g_- \rangle_{H_-}$ 。

定义 3(Krein 空间的核函数) 设 K 是 RKKS, 且 $K = H_+ \oplus H_-$, 那么

1. H_+ 和 H_- 是 RKHS (Reproducing Kernel Hilbert Space);
2. 存在一个对称函数 $k(x, x'), k(x, \cdot) \in K$ 使得 $\forall f \in K, \langle f, k(x, \cdot) \rangle_K = f(x)$;
3. $k = k_+ + k_-$ 。

称满足上述条件的函数为 Krein 空间的核函数。

Krein 空间的核函数具有如下性质:

1. 存在一个 RKKS 和核 k ;
2. 核 k 可以分解出一个正定核 k_+ , 即存在两个正定核 k_+ 和 k_- , 使得 $k = k_+ - k_-$;
3. 如果 p 是正定核, 那么 $p - k$ 也是正定核。

3 求解非正定核 Huber-SVR 的 SMO 算法

3.1 Huber-SVR 的 SMO 模型

SMO 算法是一种求解大型优化问题的分解算法。分解算法是一个反复迭代的过程,就式(2)的优化模型而言,每一步可以把 α 的分量分成两部分:一部分属于工作集;另一部分属于非工作集。记它们对应的下标集分别为 B 和 N 。需要调整的是 α_B , 而 α_N 是固定不变的。如果用上述分解算法解决 Huber-SVR 的问题,则 Huber-SVR 的二次规划问题就变为:

$$\min_{\alpha_B} \frac{1}{2} \begin{bmatrix} \alpha_B^T & \alpha_N^T \end{bmatrix} \begin{bmatrix} K_{BB} & K_{BN} \\ K_{NB} & K_{NN} \end{bmatrix} \begin{bmatrix} \alpha_B \\ \alpha_N \end{bmatrix} - \begin{bmatrix} y_B^T & y_N^T \end{bmatrix} \begin{bmatrix} \alpha_B \\ \alpha_N \end{bmatrix} + \frac{\mu}{2C} \begin{bmatrix} \alpha_B^T & \alpha_N^T \end{bmatrix} \begin{bmatrix} \alpha_B \\ \alpha_N \end{bmatrix} \quad (3)$$

$$\text{s. t. } \begin{cases} -C \leq \alpha_i \leq C, i \in B \\ e_{B\alpha_B}^T = -e_{N\alpha_N}^T \end{cases}$$

当移去常数项后,即为:

$$\min_{\alpha_B} \frac{1}{2} \alpha_B^T K_{BB} \alpha_B + \alpha_B^T K_{BN} \alpha_N - y_{B\alpha_B}^T + \frac{\mu}{2C} \alpha_B^T \alpha_B \quad (4)$$

$$\text{s. t. } \begin{cases} -C \leq \alpha_i \leq C, i \in B \\ e_{B\alpha_B}^T = -e_{N\alpha_N}^T \end{cases}$$

而序列最小化(sequential minimal optimization, SMO)算法是分解算法中选取 $|B| = 2$ 的特殊情形,即每次迭代过程

中只调整相应于两个样本点 (x_i, y_i) 和 (x_j, y_j) 的 α_i 和 α_j ,它只求一个具有两个变量的最优化问题。实际上,这时工作集的规模已经减到最小,原因是原问题(2)受等式约束。只要变动一个变量 α_i ,就必须同时调整另一个变量来保证不违反该约束。

相对应于 SMO 算法的 Huber-SVR 二次规划问题为:

$$\min_{\alpha_i, \alpha_j} = \frac{1}{2} k_{ii} \alpha_i^2 + \frac{1}{2} k_{jj} \alpha_j^2 + k_{ij} \alpha_i \alpha_j + \alpha_i \sum_{t \neq i, j} \alpha_t k_{ti} + \alpha_j \sum_{t \neq i, j} \alpha_t k_{tj} - y_i \alpha_i - y_j \alpha_j + \frac{\mu}{2C} \alpha_i^2 + \frac{\mu}{2C} \alpha_j^2 \quad (5)$$

这就是 Huber-SVR 的 SMO 模型。

3.2 求解非正定核 Huber-SVR 的 SMO 算法

当核函数为非正定时,为了解决上述的二次规划问题,我们提出用于求解非正定核 Huber-SVR 的 SMO 算法。

Huber-SVR SMO 算法

输入:初始可行权值 α ,样本 X, Y ,损失函数的参数 $\mu > 0$,惩罚系数 $C > 0$ —参见式(5),核函数 K ,精度 $\epsilon > 0$;

输出:已优化的 α

步骤:

1. 设定精度 $\epsilon, \alpha^0 = 0$,令 $k = 0$;

2. 按以下准则选取工作集 $B = \{i, j\}$;

$$i = \operatorname{argmax}_{i \in I_{UP}(\alpha, C)} -\nabla F(\alpha)_i, j = \operatorname{argmin}_{i \in I_{low}(\alpha, C)} -\nabla F(\alpha)_i.$$

式中, $I_{UP}(\alpha, C) = \{i | -C < \alpha_i < C \text{ or } \alpha_i = C\}$, $I_{low}(\alpha, C) = \{i | -C < \alpha_i < C \text{ or } \alpha_i = -C\}$, $F(\alpha)$ 即为(2);

3. 反复执行以下步骤:

3.1 计算 $d_j = \alpha_j^{k+1} - \alpha_j^k, L = \max(\alpha_i^k - C, -C - \alpha_j^k), H = \min(\alpha_i^k + C, C - \alpha_j^k)$

$$f(d_j) = \frac{1}{2} (k_{ii} - 2k_{ij} + k_{jj} + \frac{2\mu}{C}) + (-\nabla F(\alpha^k)_i + \nabla F(\alpha^k)_j) d_j$$

3.2 若 $k_{ii} - 2k_{ij} + k_{jj} + \frac{2\mu}{C} > 0$,则

$$d_j = \max(-\frac{-\nabla F(\alpha^k)_i + \nabla F(\alpha^k)_j}{k_{ii} - 2k_{ij} + k_{jj} + \frac{2\mu}{C}}, L)$$

否则,若 $f(H) < f(L)$,则

$$d_j = H$$

否则 $d_j = L$;

3.3 计算 $d_i = -d_j, \alpha_i^{k+1} = \alpha_i^k + d_i, \alpha_j^{k+1} = \alpha_j^k + d_j$;

3.4 若在精度 ϵ 内满足停机准则

$$-C \leq \alpha_i \leq C, i = 1, \dots, l, \sum_{i=1}^l y_i \alpha_i = 0$$

$$\sum_{i=1}^l |y_i - (\sum_{k=1}^n \alpha_k \cdot K(x_k, x_i) + b)| < \epsilon$$

则转第4步,否则,令 $k = k + 1$,转第2步;

4. 取近似最优解 $\alpha^* = \alpha^{k+1}$,算法结束。

这一算法的关键步骤是步骤2,即如何选择工作集,亦即变量 α_i^k, α_j^k 的选择准则;以及步骤3.2,即如何选择工作集变量的当前最优值,亦即如何计算 d_j 的当前最优值。下一节给出有关的推导和证明。本算法的计算用时取决于停机准则中的精度设置。实验表明,与标准 SVR 算法比较,本算法的用时较长。

需要说明的是,这里的 Huber-SVR SMO 算法,虽然是专门设计的用于求解非正定核的 Huber-SVR,但也可以用于求解正定核的 Huber-SVR。表1给出了 SMO 用于求解正定核和非正定核的 Huber-SVR 时的主要区别。

表1 SMO 用于求解正定核和非正定核的 Huber-SVR 的主要区别

求解正定核 Huber-SVR 的 SMO	求解非正定核 Huber-SVR 的 SMO
主要目的 对规模较大的问题提高求解速度	能够求解非正定核 Huber-SVR
3.2 $d_j = \max(-\frac{-\nabla F(\alpha^k)_i + \nabla F(\alpha^k)_j}{k_{ii} - 2k_{ij} + k_{jj} + \frac{2\mu}{C}}, L)$	3.2 若 $k_{ii} - 2k_{ij} + k_{jj} + \frac{2\mu}{C} > 0$, 则 $d_j = \max(-\frac{-\nabla F(\alpha^k)_i + \nabla F(\alpha^k)_j}{k_{ii} - 2k_{ij} + k_{jj} + \frac{2\mu}{C}}, L)$, 否则,若 $f(H) < f(L)$ 则 $d_j = H$ 否则 $d_j = L$;

3.3 有关算法的两个定理

关于算法步骤2中工作集 $B = \{i, j\}$ 的存在性,有以下定理1^[4]。

定理1 当 α 是模型(2)的稳定点时,当且仅当 α 是可行的并且满足:

$$\max_{i \in I_{UP}(\alpha, C)} -\nabla F(\alpha)_i \leq \min_{i \in I_{low}(\alpha, C)} -\nabla F(\alpha)_i \quad (6)$$

证明:

将式(2)写成如下的形式:

$$\max_{\alpha} = \sum_{i=1}^l \alpha_i y_i - \frac{1}{2C} \sum_{i=1}^l \alpha_i^2 \mu - \frac{1}{2} w(\alpha) w(\alpha) \quad (7)$$

$$\text{s. t. } \begin{cases} -C \leq \alpha_i \leq C, i = 1, \dots, l \\ \sum_{i=1}^l \alpha_i = 0 \end{cases}$$

式中, $w(\alpha) = \sum_i \alpha_i z_i, k(x_i, x_j) = \phi(x_i)^T \phi(x_j), z_i = \phi(x_i)$ 。

引入 Lagrange 函数

$$\bar{L} = \frac{1}{2} w(\alpha) \cdot w(\alpha) - \sum_{i=1}^l \alpha_i y_i + \frac{1}{2C} \sum_{i=1}^l \alpha_i^2 \mu + \beta \sum_{i=1}^l \alpha_i - \sum_{i=1}^l \psi_i (\alpha_i + C) - \sum_{i=1}^l \delta_i (C - \alpha_i)$$

并令 $F_i = y_i - w(\alpha) z_i$,则有

$$\frac{\partial \bar{L}}{\partial \alpha_i} = -F_i + \frac{\mu}{C} \alpha_i + \beta - \psi_i + \delta_i = 0,$$

$$\psi_i (\alpha_i + C) = 0, \psi_i \geq 0, \alpha_i \geq -C,$$

$$\delta_i (C - \alpha_i) = 0, \delta_i \geq 0, \alpha_i \leq C.$$

根据 α_i 的取值范围,可出现以下互不相交的情形之一:

情形1: $\alpha_i = C, F - \beta - \mu \geq 0$;

情形2: $-C < \alpha_i < C, F - \beta - \frac{\mu}{C} \alpha_i = 0$;

情形3: $\alpha_i = -C, F - \beta - (-\mu) \leq 0$ 。

若定义:

$$I_0 = \{i: -C < \alpha_i < C\}, I_1 = \{i: \alpha_i = C\}, I_2 = \{i: \alpha_i = -C\}$$

并令

$$\tilde{F}_i = F_i - \frac{\mu}{C} \alpha_i, i \in I_0 \cup I_1$$

$$\bar{F}_i = F_i - \frac{\mu}{C} \alpha_i, i \in I_0 \cup I_2$$

则归纳上述3种情形可得:

$$\beta \leq \tilde{F}_i \forall i \in I_0 \cup I_1; \beta \geq \bar{F}_i \forall i \in I_0 \cup I_2$$

若再令:

$$b_{up} = \min\{\tilde{F}_i; i \in I_0 \cup I_1\}, b_{low} = \max\{\bar{F}_i; i \in I_0 \cup I_2\}$$

可得:

$$b_{low} \leq b_{up}$$

$$I_{UP}(\alpha, C) = \{i | -C < \alpha_i < C \text{ or } \alpha_i = C\}$$

$$I_{low}(\alpha, C) = \{i | -C < \alpha_i < C \text{ or } \alpha_i = -C\}$$

即

$$\max_{i \in I_{UP}(\alpha, C)} -\nabla F(\alpha)_i \leq \min_{i \in I_{kno}(\alpha, C)} -\nabla F(\alpha)_i$$

由定理 1 可知,若 α^k 是可行的但不是(2)的最优解,则条件(6)不满足。所以工作集 $B=\{i, j\}$ 存在:

$$i = \operatorname{argmax}_{i \in I_{UP}(\alpha^k, C)} -\nabla F(\alpha)_i, j = \operatorname{argmin}_{i \in I_{kno}(\alpha^k, C)} -\nabla F(\alpha)_i \quad (8)$$

关于算法中的步骤 3.2,有以下引理和定理。

引理 1 式(5)中,若设 $\alpha_i = \alpha_i^k + d_i, \alpha_j = \alpha_j^k + d_j$,则式(5)可化为:

$$\min_{d_j} \frac{1}{2} (k_{ii} - 2k_{ij} + k_{jj} + \frac{2\mu}{C}) d_j^2 + (-\nabla F(\alpha^k)_i + \nabla F(\alpha^k)_j) d_j \quad (9)$$

$$L \leq d_j \leq H$$

式中, $L = \max(\alpha_i^k - C, -C - \alpha_j^k), H = \min(\alpha_i^k + C, C - \alpha_j^k)$ 。

证明:

将 $\alpha_i = \alpha_i^k + d_i, \alpha_j = \alpha_j^k + d_j$ 代入式(5)中,化简可得:

$$\min_{d_i, d_j} \frac{1}{2} [d_i, d_j] \begin{bmatrix} K_{ii} & K_{ij} \\ K_{ji} & K_{jj} \end{bmatrix} \begin{bmatrix} d_i \\ d_j \end{bmatrix} + [\nabla F(\alpha^k)_i \quad \nabla F(\alpha^k)_j] \begin{bmatrix} d_i \\ d_j \end{bmatrix} + \frac{\mu}{2C} [d_i, d_j] \begin{bmatrix} d_i \\ d_j \end{bmatrix} \quad (10)$$

$$d_i + d_j = 0$$

$$\text{s. t. } -C \leq \alpha_i^k + d_i, \alpha_j^k + d_j \leq C$$

再将约束 $d_i = -d_j$ 代入式(10)中,经推导可得:

$$\min_{d_j} \frac{1}{2} (k_{ii} - 2k_{ij} + k_{jj} + \frac{2\mu}{C}) d_j^2 + (-\nabla F(\alpha^k)_i + \nabla F(\alpha^k)_j) d_j$$

$$\text{s. t. } L \leq d_j \leq H$$

式中, $L = \max(\alpha_i^k - C, -C - \alpha_j^k), H = \min(\alpha_i^k + C, C - \alpha_j^k)$ 。

引理 1 的意义在于它将式(5)所描述的二次规划问题转化为式(9)所描述的约束条件下抛物线的最优值求解问题,使得可将二次规划的研究转化为对抛物线的研究,这实际上有效简化了研究对象。

现在只需利用抛物线的性质来研究模型(9)。

引理 2 模型(9)中,若 $k_{ii} + k_{jj} - 2k_{ij} + \frac{2\mu}{C} > 0$,则 d_j 的最优值为

$$d_j^* = \max \left\{ -\frac{-\nabla F(\alpha^k)_i + \nabla F(\alpha^k)_j}{k_{ii} - 2k_{ij} + k_{jj} + \frac{2\mu}{C}}, L \right\}$$

证明:

当 $k_{ii} + k_{jj} - 2k_{ij} + \frac{2\mu}{C} > 0$ 时,模型(9)的目标函数只是关于自变量 d_j 的开口朝上的抛物线,其顶点横坐标为 $-\frac{-\nabla F(\alpha^k)_i + \nabla F(\alpha^k)_j}{k_{ii} - 2k_{ij} + k_{jj} + \frac{2\mu}{C}}$ 。此式中,由定理 1 知, $-\nabla F(\alpha^k)_i + \nabla F(\alpha^k)_j > 0$,故有 $-\frac{-\nabla F(\alpha^k)_i + \nabla F(\alpha^k)_j}{k_{ii} - 2k_{ij} + k_{jj} + \frac{2\mu}{C}} < 0$,即顶点只能在 II, III 象限。

又由引理 1 知, $H > 0$ (只要选择足够大的 C 即可,实际应用中 $C \gg |d_j|$),所以该抛物线的最小值只能在顶点或下限 L 上,这取决于 L 在抛物线顶点横坐标的左则还是右侧。故有:

$$d_j^* = \max \left\{ -\frac{-\nabla F(\alpha^k)_i + \nabla F(\alpha^k)_j}{k_{ii} - 2k_{ij} + k_{jj} + \frac{2\mu}{C}}, L \right\}$$

模型(9)是 Huber-SVR 模型在其工作集 $B = \{i, j\}$ 上经简化后的表述。若 Huber-SVR 模型的核函数是正定的或半正定的,则对于所有的工作集 $B = \{i, j\}$,均有 $k_{ii} + k_{jj} - 2k_{ij} + \frac{2\mu}{C} > 0$ 。所以引理 2 给出了正定核情况下 d_j^* 的取法。但是若 Huber-SVR 模型的核函数是非正定的,则对某些工作集 $B = \{i, j\}$ 会出现 $k_{ii} + k_{jj} - 2k_{ij} + \frac{2\mu}{C} \leq 0$ 的情况。以下引理 3 给出了非正定核情况下 d_j^* 的取法。

引理 3 模型(9)中,若 $k_{ii} + k_{jj} - 2k_{ij} + \frac{2\mu}{C} \leq 0$,则 d_j 的最优值只能为 $d_j^* = L$ 或 $d_j^* = H$ 。

证明:

当 $k_{ii} + k_{jj} - 2k_{ij} + \frac{2\mu}{C} \leq 0$ 时,模型(9)的目标函数只能是

关于自变量 d_j 的开口朝下的抛物线($k_{ii} + k_{jj} - 2k_{ij} + \frac{2\mu}{C} < 0$ 时),或者是一条直线($k_{ii} + k_{jj} - 2k_{ij} + \frac{2\mu}{C} = 0$ 时)。此时目标函数的最小值只能在边界点上,即 d_j 只能取 $d_j^* = L$ 或 $d_j^* = H$ 。

归纳引理 2 和引理 3 的结论,自然得到定理 2。定理 2 给出了当核函数为正定、半正定或非正定情况下, d_j 最优值的取法。

定理 2 模型(9)中, d_j 的最优值为:

$$d_j^* = \begin{cases} \max(-\frac{-\nabla F(\alpha^k)_i + \nabla F(\alpha^k)_j}{k_{ii} - 2k_{ij} + k_{jj} + \frac{2\mu}{C}}, L), k_{ii} - 2k_{ij} + k_{jj} + \frac{2\mu}{C} > 0 \\ H, k_{ii} - 2k_{ij} + k_{jj} + \frac{2\mu}{C} \leq 0 \text{ 且 } f(H) < f(L) \\ L, k_{ii} - 2k_{ij} + k_{jj} + \frac{2\mu}{C} \leq 0 \text{ 且 } f(H) > f(L) \end{cases}$$

式中, $f(\cdot)$ 是目标函数。

由引理 1、引理 2 和引理 3 不难证明定理 2,故证明从略。

定理 2 实际说明了算法步骤 3.2 的正确性。

4 实验及其结果

4.1 实验目的

验证本文提出的求解非正定核 Huber-SVR 的 SMO 算法的可行性,并借助于这一算法,比较正定核 Huber-SVR 与非正定核 Huber-SVR 的回归性能和预测性能,同时也与文献[3]的方法作一比较。

4.2 实验设计

实验 1:取函数 $y = \cos(\exp(x))$, $x \in [-4, 2]$,为基准信号,令间隔为 0.1 产生训练样本。分别用正定核 Huber-SVR、非正定核 Huber-SVR、文献[3]的方法,对这组数据进行训练,然后进行回归。非正定核 Huber-SVR 的求解使用本文的算法。这一实验主要考察本文算法的可行性以及非正定核 Huber-SVR 的回归性能。

实验 2:在实际的 4000 个鲍鱼样本的数据中,随机抽取 50 个作为训练样本,分别用正定核 Huber-SVR、非正定核 Huber-SVR、文献[3]的方法对这组数据进行训练。再在剩余

的样本数据中随机抽取 3 组(每组 50 个样本数据)作为预测测试样本,利用训练的结果,分别用正定核 Huber-SVR、非正定核 Huber-SVR、文献[3]的方法对这组数据进行预测。每一个鲍鱼样本的数据结构如表 2 所列(鲍鱼数据取自 <http://archive.ics.uci.edu/ml/datasets/Abalone>)。要求根据鲍鱼的性别、长度等 8 项数据预测鲍鱼的年龄。这一实验主要考察非正定核 Huber-SVR 的泛化性能。

表 2 鲍鱼样本数据结构

性别 (M,F, and I)	长度 (mm)	直径 (mm)	厚度 (mm)	总重量 (g)	去外壳 的重量 (g)	内脏的 重量 (g)	外壳的 重量(g)	年龄 (+1.5 gives the age in years)
-----------------------	------------	------------	------------	------------	-------------------	------------------	--------------	---

4.3 实验结果与分析

4.3.1 实验 1 的结果与分析

实验 1 中,Huber-SVR 的参数 C, μ 分别取为 $C=100, \mu=0.5$ 。

非正定核 Huber-SVR 的核函数取组合高斯核 $\exp\left(-\frac{\|s-t\|^2}{\sigma_1}\right) + \exp\left(-\frac{\|s-t\|^2}{\sigma_2}\right) - \exp\left(-\frac{\|s-t\|^2}{\sigma_3}\right)$, 标准差分别取为 $\sigma_1=0.8, \sigma_2=1.2, \sigma_3=4$ 。SMO 算法的设置预定精度为 $\epsilon=1.7$ 。回归的结果如图 1 所示。图中符号“+”为样本点,虚线为回归线。

正定核 Huber-SVR 的核函数取 Gauss 径向基核 $\exp\left(-\frac{\|s-t\|^2}{\sigma}\right), \sigma=2$ 。回归的结果如图 2 所示。图中符号“+”为样本点,虚线为回归线。

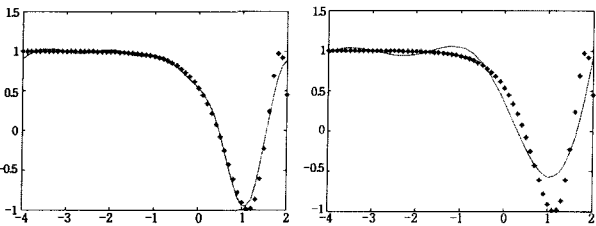


图 1 非正定核 Huber-SVR 的回归结果

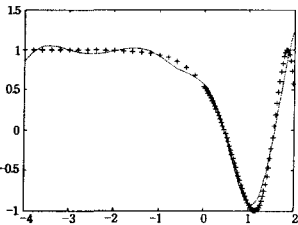


图 3 用文献[3]方法的回归结果

从图 1 与图 2 的比较中可以明显看出,非正定核 Huber-SVR 的回归结果优于正定核 Huber-SVR 的回归结果。从图 1 与图 3 的比较中可以看出,非正定核 Huber-SVR 与文献[3]方法的回归结果相近。这说明本文提出的求解非正定核

Huber-SVR 的 SMO 算法是可行的。

4.3.2 实验 2 的实验结果与分析

实验 2 中,Huber-SVR 的参数 C, μ 分别取为 $C=10, \mu=0.2$ 。SMO 算法的预设精度为 $\epsilon=90$ 。其它参数设置同实验 1。表 3 给出了实验 2 的实验结果。

表中数据是预测结果与实际结果之差的绝对值之和,即 $err = \sum_{i=1}^{50} |f(x_i) - y_i|$ 。误差和越小则预测性能越好,泛化性能也越好。

表 3 3 种方法预测结果的比较

	非正定核 Huber-SVR	正定核 Huber-SVR	文献[3]的方法
第 I 组	91.0842	103.5811	70.872
第 II 组	114.0396	123.3711	133.3218
第 III 组	114.1521	120.3517	140.2803

从表 3 中可知,非正定核 Huber-SVR 与正定核 Huber-SVR 相比,预测的效果有明显的改善,泛化性能较好,与文献[3]的方法比较,泛化性能也较好。

结束语 本文提出了用于求解非正定核 Huber-SVR 的 SMO 算法,并对算法作了理论的和证明,通过实验验证了算法的正确性和可行性。实验同时还表明,在一些应用中,对于同样的 Huber-SVR,一些非正定核的回归性能可能比正定核的要好。因此,有两方面值得进一步研究,一是从理论上,本文的方法仅仅是局部可行还是有一般意义需要探讨,因为一般认为非正定核的 SVR 模型是无法求解的;二是从应用上看,由于本文的算法为 SVR 使用非正定核铺平了道路,因此进一步的问题是如何根据实际应用环境选择好合适的核函数,包括正定的和非正定的。此外,与一般的用于求解 SVR 的 SMO 算法相比,本文的 SMO 算法的运行时间相对较长,这也是需要进一步研究改进的。

参考文献

- [1] Lin H-T, Lin C-J. A Study on Sigmoid Kernels for Svm and The Training of non-psd Kernels by Smo-Type Methods[R]. <http://www.work.caltech.edu/~htlin/publication/doc/tanh.pdf>, March 2003
- [2] Hassdonk B. Feature Space Interpretation of SVMs with non Positive Definite Kernels[J]. IEEE transactions on pattern analysis and machine intelligence, 2005, 27(4): 482-492
- [3] Cheng Soon Ong, Mary X, Canu S, et al. Learning with non-positive kernels[C]// International Conference on Machine Learning. Banff, Canada, July 2004
- [4] Shevade S K, Keerthi S S, Bhattacharyya C, et al. Improvements to SMO Algorithm for SVM Regression[J]. IEEE Transactions on Neural Networks, 2000, 11(5): 1188-1193
- [5] Keerthi S S, Shevade S K, Bhattacharyya C, et al. Improvements to Platt's SMO algorithm for SVM classifier design[J]. Neural Computation, 2001, 13: 637-649