第 33 卷第 4 期
2016 年 4 月

控 制 理 论 与 应 用
Control Theory & Applications

Vol. 33 No. 4
Apr. 2016

DOI: 10.7641/CTA.2016.50303

精确增量式在线ν型支持向量回归机学习算法

顾斌杰†, 潘 丰

(江南大学 轻工过程先进控制教育部重点实验室, 江苏 无锡 214122)

摘要: 为了解决ν型支持向量回归机(ν-support vector regression, ν-SVR)对偶问题的目标函数中增加的额外线性项从而导致无法产生有效初始解的问题和在绝缘增量调整过程中可能存在的解路径不可行更新问题, 提出了精确增量式在线ν-SVR学习算法. 首先基于ν-SVR的等价形式, 利用提前调整, 宽松的绝缘增量调整和精确的恢复调整有效地解决了ν-SVR对偶问题存在的上述问题. 然后分别对算法的可行性和有限收敛性进行了理论分析. 最后在四个基准测试数据集上的仿真结果进一步验证了该算法的每一步调整都是可靠的, 经过有限次数调整最终收敛到最小化问题的最优解, 而且与批处理学习算法相比, 随着训练样本的增加, 算法在缩短学习时间上的优势显著.

关键词: 在线学习; ν型支持向量回归机; 机器学习; 学习算法; 可行性分析; 有限收敛性分析

中图分类号: TP181 **文献标识码**: A

Accurate incremental online ν-support vector regression learning algorithm

GU Bin-jie†, PAN Feng

(Key Laboratory of Advanced Process Control for Light Industry, Ministry of Education,
Jiangnan University, Wuxi Jiangsu 214122, China)

Abstract: In the ν-support vector regression learning algorithm (ν-SVR), to solve the two problems existing in the dual problem, i.e., the problem of unable to generate an effective initial solution due to the extra linear term introduced in the objective function and the problem of possible infeasible updating solution path in the adiabatic incremental adjustment process, we propose an accurate incremental online ν-SVR learning algorithm. First, based on the equivalent formulation of ν-SVR, the problems existing in the dual problem can be effectively solved by prior adjustments, relaxed adiabatic incremental adjustments and accurate restoration adjustments. Then, the feasibility and the finite convergence of the proposed algorithm are theoretically analyzed, respectively. Finally, the simulation results on four benchmark datasets further validate the reliability of each adjustment of the proposed algorithm, and the proposed algorithm will converge to the optimal solution of minimization problem within finite adjustments. Furthermore, the learning time of the proposed algorithm is much shorter than that of the batch learning algorithm when the number of training samples is increased.

Key words: online learning; ν-support vector regression; machine learning; learning algorithms; feasibility analysis; finite convergence analysis

1 引言(Introduction)

支持向量机(support vector machines, SVMs)[1]的核心问题之一就是样本的学习. 众所周知, 样本的学习需要求解一个与最小化问题样本个数相同的多参数凸二次规划问题, 而且学习难度随着训练样本集的规模增大而迅速增加.

然而, 在实际问题环境中, 例如: 谷氨酸发酵过程、网络数据监控、金融数据分析、时间序列预测等, 样本是在线环境下增量提供的, 最为极端的情况是每次只增加一个新的样本. 此时若采用批处理学习算法, 则需要从头开始重新进行学习, 效率低下.

与批处理学习算法相比, 在线学习算法的最大优势是当样本增加时, 无需从头开始重新进行学习[2]. Cauwenberghs 和 Poggio 提出了第1个精确增量和减量式在线C-支持向量分类机学习算法[3], 称为C&P算法. 在此基础上, Martin[4], Ma等人[5]将其推广到精确增量和减量式在线ε-支持向量回归机(support vector regression, SVR)学习算法; Laskov等人对C&P算法的

收稿日期: 2015−04−15; 录用日期: 2015−11−26.
†通信作者. E-mail: gubinjie1980@126.com; Tel.: +86 510-85910633.
本文责任编委: 陈杰.
国家自然科学基金资助项目(61273131, 61403168).
Supported by National Natural Science Foundation of China (61273131, 61403168).

增量式部分进行了理论分析[2]; 文献[6]将C&P算法推广到每次能够同时处理多个新增样本的精确多增量和减量式在线支持向量分类机学习算法; 顾彬等人将C&P算法推广到精确增量式在线ν型支持向量分类机学习算法[7], 并对算法的可行性和有限收敛性进行了理论分析[8].

然而, 截止目前为止还没有出现精确增量式在线ν-SVR学习算法. 主要问题如下: 一是ν-SVR对偶问题中的箱约束与惩罚参数和训练样本个数有关, 而且还多了一个额外的不等式约束(详见第2.1节); 二是与ν-SVC对偶问题相比, ν-SVR对偶问题的目标函数中增加了一个额外的线性项; 三是将C&P算法直接应用于ν-SVR可能会出现解路径的不可行更新问题[9–10]. 为此本文提出了精确增量式在线ν-SVR学习算法. 该算法基于ν-SVR的等价形式, 分别利用提前调整, 宽松的绝缘增量调整和精确的恢复调整解决上述三个问题. 通过理论分析和仿真结果表明该算法具有可行性和有限收敛性, 并体现算法在缩短训练时间上比批处理学习算法有明显改善.

2　ν-SVR及其等价形式(ν-SVR and its equivalent formulation)

2.1　ν-SVR简要回顾(Brief overview of ν-SVR)

在ε-SVR中, 参数ε的选取是比较困难的[11]. 为此, Schölkopf等人提出了一种新的支持向量回归机形式ν-SVR, 能够自动调节参数ε[12]. 在ν-SVR中, 参数$0\leqslant\nu\leqslant1$有直观上的意义, 可证: $m/l\leqslant\nu\leqslant n/l$, 其中: m是错误支持向量的个数; n是支持向量的个数; l是训练样本集$\boldsymbol{T}=\{(\boldsymbol{x}_1,y_1),\cdots,(\boldsymbol{x}_l,y_l)\}$中的样本个数, 其中: $\boldsymbol{x}_i\in\mathbb{R}^{\mathrm{d}}$是输入, $y_i\in\mathbb{R},\ i=1,\cdots,l$是输出. 因此, 参数$\nu$能用于控制$m$和$n$, 从而方便$\nu$值的选取. ν-SVR考虑如下的原始问题[13]:

$$\min_{\boldsymbol{w},b,\varepsilon,\xi_i^{(*)}} P=\frac{1}{2}\langle\boldsymbol{w},\boldsymbol{w}\rangle+C(\nu\varepsilon+\frac{1}{l}\sum_{i=1}^{l}(\xi_i+\xi_i^*)),\tag{1}$$

$$\begin{aligned}&\text{s.t. }(\langle\boldsymbol{w},\phi(\boldsymbol{x}_i)\rangle+b)-y_i\leqslant\varepsilon+\xi_i,\\&\quad y_i-(\langle\boldsymbol{w},\phi(\boldsymbol{x}_i)\rangle+b)\leqslant\varepsilon+\xi_i^*,\\&\quad\xi_i^{(*)}\geqslant0,\ i=1,\cdots,l,\varepsilon\geqslant0,\end{aligned}$$

其中: $(*)$是变量有或者无星号的缩写(下同), $\langle\cdot,\cdot\rangle$表示内积, 输入$\boldsymbol{x}_i$被函数$\phi$映射到一个高维再生核希尔伯特空间, $\xi_i^{(*)}$是非负松弛变量, C是惩罚参数, $\boldsymbol{w}$是权系数向量, b是偏置.

ν-SVR的对偶问题如下[13]:

$$\begin{aligned}&\min_{\alpha_i^{(*)}} D=\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}(\alpha_i-\alpha_i^*)H_{ij}(\alpha_j-\alpha_j^*)+\\&\qquad\sum_{i=1}^{l}(\alpha_i-\alpha_i^*)y_i,\\&\text{s.t. }\sum_{i=1}^{l}(\alpha_i-\alpha_i^*)=0,\sum_{i=1}^{l}(\alpha_i+\alpha_i^*)\leqslant C\nu,\\&\qquad0\leqslant\alpha_i^{(*)}\leqslant\frac{C}{l},\ i=1,\cdots,l,\end{aligned}\tag{2}$$

其中: $H_{ij}=K(\boldsymbol{x}_i,\boldsymbol{x}_j)=\langle(\phi(\boldsymbol{x}_i),\phi(\boldsymbol{x}_j)\rangle$, K是核函数(kernel function).

从式(2)可以看出ν-SVR对偶问题存在两个难题: 一是箱约束$0\leqslant\alpha_i^{(*)}\leqslant C/l$不仅与$C$有关, 而且还与$l$有关; 二是与标准的$\varepsilon$-SVR的对偶问题[13]相比, 多了一个额外的不等式约束$\sum_{i=1}^{l}(\alpha_i+\alpha_i^*)\leqslant C\nu$. 两个难题的存在不利于设计在线$\nu$-SVR学习算法, 下面研究如何解决上述两个难题.

2.2　ν-SVR的等价形式(Equivalent formulation of ν-SVR)

为了解决第1个难题, 首先把式(1)中的目标函数P乘l并利用Wolfe对偶得到对应的对偶问题, 然后将此对偶问题中的目标函数除C^2并令$\alpha_i^{(*)}\leftarrow\alpha_i^{(*)}/C$后得到如下的等价对偶问题:

$$\begin{aligned}&\min_{\alpha_i^{(*)}} D=\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}(\alpha_i-\alpha_i^*)P_{ij}(\alpha_j-\alpha_j^*)+\\&\qquad\sum_{i=1}^{l}(\alpha_i-\alpha_i^*)y_i'\\&\text{s.t. }\sum_{i=1}^{l}(\alpha_i-\alpha_i^*)=0,\sum_{i=1}^{l}(\alpha_i+\alpha_i^*)\leqslant\nu l,\\&\qquad0\leqslant\alpha_i^{(*)}\leqslant1,\ i=1,\cdots,l\end{aligned}\tag{3}$$

其中: $P_{ij}=H_{ij}/l$, $y_i'=y_i/C$.

为了解决第2个难题, 可以借助于文献[13]中的结论.

引理 1　对于式(3), 当$0\leqslant\nu\leqslant1$时, 总存在最优解使得不等式约束$\sum_{i=1}^{l}(\alpha_i+\alpha_i^*)\leqslant\nu l$可以用等式约束$\sum_{i=1}^{l}(\alpha_i+\alpha_i^*)=\nu l$来代替.

引理1的证明详见文献[13], 此处省略.

基于引理1, 考虑如下的对偶问题:

$$\begin{aligned}&\min_{\alpha_i^{(*)}} D=\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}(\alpha_i-\alpha_i^*)P_{ij}(\alpha_j-\alpha_j^*)+\\&\qquad\sum_{i=1}^{l}(\alpha_i-\alpha_i^*)y_i',\\&\text{s.t. }\sum_{i=1}^{l}(\alpha_i-\alpha_i^*)=0,\sum_{i=1}^{l}(\alpha_i+\alpha_i^*)=\nu l,\\&\qquad0\leqslant\alpha_i^{(*)}\leqslant1,\ i=1,\cdots,l.\end{aligned}\tag{4}$$

为了得到一个更为紧凑的最小化问题形式, 令z_i代表训练样本$(\boldsymbol{x}_i,y_i')$的标签, 定义新的训练样本集为$\boldsymbol{S}=\boldsymbol{S}^+\cup\boldsymbol{S}^-$, 其中: $\boldsymbol{S}^+=\{(\boldsymbol{x}_i,y_i',z_i=+1)|_{i=1}^l\}$, $\boldsymbol{S}^-=\{(\boldsymbol{x}_i,y_i',z_i=-1)|_{i=1}^l\}$. 在此基础上, 式(4)可以

改写成如下ν-SVR的等价形式:

$$\min_{\alpha_i} D=\frac{1}{2}\sum_{i=1}^{2l}\sum_{j=1}^{2l}\alpha_iQ_{ij}\alpha_j+\sum_{i=1}^{2l}z_i\alpha_iy_i', \quad (5)$$

$$\text{s.t.}\ \sum_{i=1}^{2l}z_i\alpha_i=0, \sum_{i=1}^{2l}\alpha_i=\nu l,$$
$$0\leqslant\alpha_i\leqslant 1,\ i=1,\cdots,2l,$$

其中$Q_{ij}=z_iz_jP_{ij}=\langle z_i\phi(\boldsymbol{x}_i), z_j\phi(\boldsymbol{x}_j)\rangle/l$.

根据凸优化理论[14], 式(5)等价于优化如下的凸二次目标函数:

$$\min_{0\leqslant\alpha_i\leqslant 1} W=\frac{1}{2}\sum_{i=1}^{2l}\sum_{j=1}^{2l}\alpha_iQ_{ij}\alpha_j+\sum_{i=1}^{2l}z_i\alpha_iy_i'+ b(\sum_{i=1}^{2l}z_i\alpha_i)+\rho(\sum_{i=1}^{2l}\alpha_i-\nu l), \quad (6)$$

其中: b和ρ是拉格朗日乘子, 而且参考文献[5]可以证明在最优的情况下, 式(6)中的b就是式(1)中的偏置b.

式(6)解的Karush-Kuhn-Tucker(KKT)条件如下:

$$g_i=\frac{\partial W}{\partial\alpha_i}=\sum_{j=1}^{2l}\alpha_jQ_{ij}+z_iy_i'+z_ib+\rho, \quad (7)$$

$$\sum_{i=1}^{2l}z_i\alpha_i=0, \quad (8)$$

$$\sum_{i=1}^{2l}\alpha_i=\nu l. \quad (9)$$

式(7)将新的训练样本集$\boldsymbol{S}$划分成如图1所示的3个相互独立的子集.

1) $\boldsymbol{S}_{\rm R}=\{g_i>0,\alpha_i=0,i=1,\cdots,p\}$集是被$\rho$-带覆盖的剩余样本, 其中: p是剩余样本的个数.

2) $\boldsymbol{S}_{\rm S}=\{g_i=0,\ 0<\alpha_i<1,\ i=1,\cdots,q\}$集是严格在$\rho$-带上的间隔支持向量集, 其中: q是间隔支持向量的个数.

3) $\boldsymbol{S}_{\rm E}=\{g_i<0,\ \alpha_i=1,i=1,\cdots,r\}$集是$\rho$-带外的错误支持向量集, 其中: r是错误支持向量的个数.

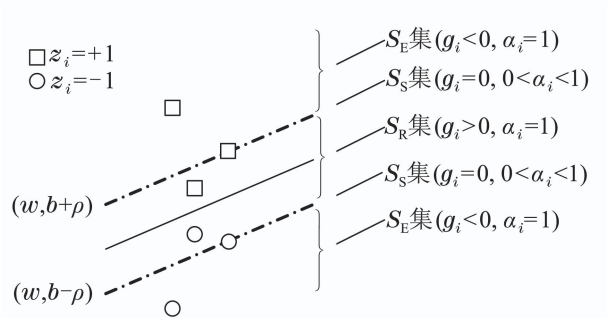


图 1 训练样本集$\boldsymbol{S}$划分成3个相互独立的子集

Fig. 1 The partition of $\boldsymbol{S}$ into three independent sets

3 精确增量式在线ν-SVR学习算法(Accurate incremental online ν-SVR learning algorithm)

式(4)目标函数中的$\sum_{i=1}^{l}(\alpha_i-\alpha_i^*)y_i'$将会导致在新样本$(\boldsymbol{x}_{\rm new}, y_{\rm new})$加入到训练样本集$\boldsymbol{T}$之前无法产生式(5)的有效初始解. 此外, C&P算法不能直接应用于式(5), 问题的根源在于在满足特定条件的情况下, 存在第3.2节中描述的矛盾1和第3.3节中描述的矛盾2.

为此本文提出了精确增量式在线ν-SVR学习算法(accurate incremental online ν-SVR learning algorithm), 称为AIOSVR算法. 该算法包含3大部分: 一是提前调整; 二是宽松的绝缘增量调整; 三是精确地恢复调整. 提前调整解决无法产生有效初始解的问题. 宽松的绝缘增量调整和精确的恢复调整能够分别有效地避免矛盾1和矛盾2的出现, 也就是说能够解决解路径不可行更新问题.

为了便于理解, 下面列出本文中用到的符号表示的含义: Δ表示变量在调整过程中的变化量; $[\![\cdot]\!]$表示集合中的样本个数; $\alpha_{\rm c}$表示候选样本的权系数; $\boldsymbol{0}$和$\boldsymbol{1}$分别表示恰当维数的全0和全1列向量; $\boldsymbol{I}$表示单位阵; $\det(\cdot)$表示方阵对应的行列式; $\rm rank(\cdot)$表示矩阵的秩; $\boldsymbol{Q}_{\boldsymbol{S}_{\rm S}\boldsymbol{S}_{\rm S}}$表示行列均是由$\boldsymbol{S}_{\rm S}$集中样本索引后得到的$[\![\boldsymbol{S}_{\rm S}]\!]\times[\![\boldsymbol{S}_{\rm S}]\!]$维方阵; 矩阵或向量的下标\表示删除; 当$\boldsymbol{S}_{\rm S}$集中的样本全部具有相同的标签时$\boxed{\Delta\rho}=0$表示$\Delta\rho=0$, 否则忽略$\Delta\rho$, 以此类推.

3.1 提前调整(Prior adjustments, PA)

PA的目的是为了在新样本$(\boldsymbol{x}_{\rm new}, y_{\rm new})$加入到训练样本集$\boldsymbol{T}$之前就准备好式(5)的有效初始解, 即当$Q\leftarrow\frac{l}{l+1}Q$时, 恢复式(5)的最优解. 本文采用的方法如下:

首先设置

$$g_i\leftarrow\frac{l}{l+1}g_i,\ i\in\boldsymbol{S},\ b\leftarrow\frac{l}{l+1}b,\ \rho\leftarrow\frac{l}{l+1}\rho.$$

然后在输出$y_i'(i=1,\cdots,2l)$上添加一个收缩因子μ, 初始时

$$\mu=\frac{l}{l+1}.$$

最后在保证所有样本严格满足KKT条件的情况下逐步增大μ直至1为止.

在PA中, 为了保证所有样本严格满足KKT条件, 式(7)到式(9)可以表示成如下的线性方程组:

$$\Delta g_i=\sum_{j\in\boldsymbol{S}_{\rm S}}\Delta\alpha_jQ_{ij}+\Delta\mu z_iy_i'+ z_i\Delta b+\Delta\rho=0,\ \forall i\in\boldsymbol{S}_{\rm S}, \quad (10)$$

$$\sum_{j\in\boldsymbol{S}_{\rm S}}z_j\Delta\alpha_j=0, \quad (11)$$

$$\sum_{j\in\boldsymbol{S}_{\rm S}}\Delta\alpha_j=0. \quad (12)$$

定义

$$\boldsymbol{\chi}_{\boldsymbol{S}_{\rm S}}=[z_1y_1'\ \ \cdots\ \ z_{[\![\boldsymbol{S}_{\rm S}]\!]}y_{[\![\boldsymbol{S}_{\rm S}]\!]}']^{\rm T},$$
$$\Delta\boldsymbol{\alpha}_{\boldsymbol{S}_{\rm S}}=[\Delta\alpha_1\ \ \cdots\ \ \Delta\alpha_{[\![\boldsymbol{S}_{\rm S}]\!]}]^{\rm T},$$
$$\boldsymbol{z}_{\boldsymbol{S}_{\rm S}}=[z_1\ \ \cdots\ \ z_{[\![\boldsymbol{S}_{\rm S}]\!]}]^{\rm T},$$

则线性方程组式(10)到式(12)可以表示成如下的矩阵形式:

$$\underbrace{\begin{bmatrix} 0 & 0 & \boldsymbol{z}_{\boldsymbol{S}_{\rm S}}^{\rm T} \\ 0 & 0 & \boldsymbol{1}_{\boldsymbol{S}_{\rm S}}^{\rm T} \\ \boldsymbol{z}_{\boldsymbol{S}_{\rm S}} & \boldsymbol{1}_{\boldsymbol{S}_{\rm S}} & \boldsymbol{Q}_{\boldsymbol{S}_{\rm S}\boldsymbol{S}_{\rm S}} \end{bmatrix}}_{\tilde{\boldsymbol{M}}} \begin{bmatrix} \Delta b \\ \Delta\rho \\ \Delta\boldsymbol{\alpha}_{\boldsymbol{S}_{\rm S}} \end{bmatrix} = -\begin{bmatrix} 0 \\ 0 \\ \boldsymbol{\chi}_{\boldsymbol{S}_{\rm S}} \end{bmatrix} \Delta\mu. \tag{13}$$

假设$\tilde{\boldsymbol{N}} = \tilde{\boldsymbol{M}}^{-1}$, 则有

$$\begin{bmatrix} \Delta b \\ \Delta\rho \\ \Delta\boldsymbol{\alpha}_{\boldsymbol{S}_{\rm S}} \end{bmatrix} = -\tilde{\boldsymbol{N}} \begin{bmatrix} 0 \\ 0 \\ \boldsymbol{\chi}_{\boldsymbol{S}_{\rm S}} \end{bmatrix} \Delta\mu \stackrel{\rm def}{=} \underbrace{\begin{bmatrix} \tilde{\gamma}_b \\ \tilde{\gamma}_\rho \\ \tilde{\boldsymbol{\gamma}}_{\boldsymbol{S}_{\rm S}} \end{bmatrix}}_{\tilde{\boldsymbol{\gamma}}} \Delta\mu, \tag{14}$$

其中$\tilde{\boldsymbol{\gamma}}_{\boldsymbol{S}_{\rm S}} = [\tilde{\gamma}_1 \ \cdots \ \tilde{\gamma}_{[\![\boldsymbol{S}_{\rm S}]\!]}]^{\rm T}$.

将式(14)代入式(10)可得

$$\Delta g_i = (\sum_{j\in\boldsymbol{S}_{\rm S}} \tilde{\gamma}_j Q_{ij} + z_i y_i' + z_i\tilde{\gamma}_b + \tilde{\gamma}_\rho)\Delta\mu \stackrel{\rm def}{=} \tilde{\varsigma}_i\Delta\mu,\ \forall i\in\boldsymbol{S}, \tag{15}$$

其中$\tilde{\varsigma}_i = 0,\ \forall i \in \boldsymbol{S}_{\rm S}$.

定理 1　AIOSVR算法在PA之后必定有$\rho\geqslant 0$.

证　PA的目的是为了恢复如下对偶问题的最优解:

$$\min_{\alpha_i}\ D = \frac{l}{l+1}\frac{1}{2}\sum_{i=1}^{2l}\sum_{j=1}^{2l}\alpha_i Q_{ij}\alpha_j + \sum_{i=1}^{2l} z_i\alpha_i y_i', \tag{16}$$

$$\text{s.t.}\ \sum_{i=1}^{2l} z_i\alpha_i = 0,\ \sum_{i=1}^{2l}\alpha_i = \nu l,$$
$$0\leqslant\alpha_i\leqslant 1,\ i=1,\cdots,2l.$$

可以验证式(16)是如下原始问题的对偶问题:

$$\min_{\boldsymbol{w},b,\varepsilon,\xi_i^{(*)}} P = \frac{l+1}{2}\langle\boldsymbol{w},\boldsymbol{w}\rangle + C(\nu l\varepsilon + \sum_{i=1}^{l}(\xi_i+\xi_i^*)), \tag{17}$$

$$\begin{aligned}\text{s.t.}\ & (\langle\boldsymbol{w},\phi(\boldsymbol{x}_i)\rangle + b) - y_i \leqslant \varepsilon + \xi_i, \\ & y_i - (\langle\boldsymbol{w},\phi(\boldsymbol{x}_i)\rangle + b) \leqslant \varepsilon + \xi_i^*, \\ & \xi_i^{(*)} \geqslant 0,\ i=1,\cdots,l,\ \varepsilon\geqslant 0.\end{aligned}$$

根据式(6)容易验证b和ρ也是式(16)的拉格朗日乘子. 进一步, 根据KKT条件可得式(16)和式(17)的最优解之间的关系为

$$\boldsymbol{w} = \frac{1}{l+1}\sum_{i=1}^{2l} z_i\alpha_i\phi(\boldsymbol{x}_i),\ \rho=\varepsilon.$$

进一步, 根据强对偶理论可得式(16)和式(17)最优解的对偶间隙为0. 假设在提前调整之后$\rho<0$, 那么通过设定$\varepsilon<0$, 式(17)中的目标函数P可以取更小的值, 这就意味着存在矛盾. 因此, 在PA之后必定有$\rho\geqslant 0$.

证毕.

3.1.1　处理特殊情况(Processing special cases)

在下述两个特殊情况下, 矩阵$\tilde{\boldsymbol{M}}$会变成奇异阵. 下面给出这两种特殊情况并讨论在$\tilde{\boldsymbol{M}}$是奇异阵的情况下, 如何确定Δb, $\Delta\rho$, $\Delta\alpha_i$, Δg_i与$\Delta\mu$之间的线性关系.

特殊情况1: $\boldsymbol{z}_{\boldsymbol{S}_{\rm S}} = \pm\boldsymbol{1}_{\boldsymbol{S}_{\rm S}}$, 即$\boldsymbol{S}_{\rm S}$ 集中的样本全部具有相同的标签+1或者−1时, 容易验证$\det(\tilde{\boldsymbol{M}}) = 0$, 此时$\tilde{\boldsymbol{M}}$是奇异阵.

特殊情况2: $[\![\boldsymbol{F}^+]\!] \geqslant 2$, 其中集合$\boldsymbol{F}^+$定义为

$$\boldsymbol{F}^+ = \{(\boldsymbol{x}_i, y_i', z_i = +1)\in\boldsymbol{S}_{\rm S}: (\boldsymbol{x}_i, y_i', z_i = -1)\in\boldsymbol{S}_{\rm S}\},$$

此时对应$\rho = 0$. 下面说明特殊情况2对应的矩阵$\tilde{\boldsymbol{M}}$是奇异阵. 为了方便起见, 假设$[\![\boldsymbol{F}^+]\!] = 2$, 此时存在$\boldsymbol{S}_{\rm S}$集中的4个样本, 其索引号分别为$i_1, i_2, j_1, j_2$, 且满足:

$$(\boldsymbol{x}_{i_1}, y_{i_1}') = (\boldsymbol{x}_{i_2}, y_{i_2}'),\ (\boldsymbol{x}_{j_1}, y_{j_1}') = (\boldsymbol{x}_{j_2}, y_{j_2}'),$$
$$z_{i_1} + z_{i_2} = 0,\ z_{j_1} + z_{j_2} = 0.$$

容易验证将矩阵$\tilde{\boldsymbol{M}}$的第i_2行加到第i_1行的结果与第j_2行加到第j_1行的结果相同, 这就意味着$\det(\tilde{\boldsymbol{M}}) = 0$, 此时$\tilde{\boldsymbol{M}}$也是奇异阵.

若$\boldsymbol{F}^+\neq\varnothing$, 则定义集合$\boldsymbol{F}$是从集合$\boldsymbol{F}^+$中删除任意一个样本后得到的缩减集合. 若$\boldsymbol{F}^+=\varnothing$, 则定义$\boldsymbol{F}=\varnothing$. 此外, 令集合$\boldsymbol{S}_{\rm S}'=\boldsymbol{S}_{\rm S}-\boldsymbol{F}$.

当$\boldsymbol{S}_{\rm S}$集中的样本全部具有相同的标签时, 定义集合$\tilde{\boldsymbol{F}}=\boldsymbol{F}$, 否则定义集合$\tilde{\boldsymbol{F}}=\boldsymbol{F}\cup\{i_\rho\}$. 此时, 定义从$\tilde{\boldsymbol{M}}$中删除集合$\tilde{\boldsymbol{F}}$对应的行和列后得到的缩减矩阵为$\tilde{\boldsymbol{M}}'=\tilde{\boldsymbol{M}}_{\setminus\tilde{\boldsymbol{F}}\tilde{\boldsymbol{F}}}$. 第4.1.1节中的定理2证明了$\tilde{\boldsymbol{M}}'$必定存在逆矩阵$\tilde{\boldsymbol{N}}$.

进一步, 令$\Delta\boldsymbol{\alpha}_{\boldsymbol{F}}=\boldsymbol{0}$和$\boxed{\Delta\rho}=0$, 可以类似地得到如下的线性关系:

$$\begin{bmatrix} \Delta b \\ \Delta\rho \\ \Delta\boldsymbol{\alpha}_{\boldsymbol{S}_{\rm S}} \end{bmatrix}_{\setminus\tilde{\boldsymbol{F}}} = -\tilde{\boldsymbol{N}}\begin{bmatrix} 0 \\ 0 \\ \boldsymbol{\chi}_{\boldsymbol{S}_{\rm S}} \end{bmatrix}_{\setminus\tilde{\boldsymbol{F}}}\Delta\mu \stackrel{\rm def}{=} \begin{bmatrix} \tilde{\gamma}_b \\ \tilde{\gamma}_\rho \\ \tilde{\boldsymbol{\gamma}}_{\boldsymbol{S}_{\rm S}} \end{bmatrix}_{\setminus\tilde{\boldsymbol{F}}}\Delta\mu. \tag{18}$$

最后, 令$\tilde{\boldsymbol{\gamma}}_{\boldsymbol{F}}=\boldsymbol{0}$和$\boxed{\tilde{\gamma}_\rho}=0$, 将式(18)代入式(10)可以得到$\tilde{\boldsymbol{M}}$为奇异阵时, Δg_i与$\Delta\mu$之间的线性关系仍旧可以用式(15)表示.

3.1.2　计算最大调整量$\Delta\boldsymbol{\mu}^{\max}$(Computing the maximal adjustments $\Delta\mu^{\max}$)

随着$\Delta\alpha_i$的变化, $\boldsymbol{S}_{\rm R}$集, $\boldsymbol{S}_{\rm S}$集和$\boldsymbol{S}_{\rm E}$集的组成成分也要发生相应的变化, 因此PA的原则并不能直接应用于求得使所有训练样本满足KKT条件的新状态. 解决上述问题的方法是计算最大调整量$\Delta\mu^{\max}$, 使得在每次PA中, 只有一个特定的样本在$\boldsymbol{S}_{\rm R}$集, $\boldsymbol{S}_{\rm S}$集和$\boldsymbol{S}_{\rm E}$集之间迁移. 对于这种结构调整需要考虑如下3种情况:

情况 1 收缩因子μ达到上界1. 在μ达到上界1之前最大可能的调整量为$\Delta\mu^{\text{case1}}=1-\mu$.

情况 2 $\boldsymbol{S}_{\text{S}}$集中的某个样本的$\alpha_i$变成0或者1. 令$\tau$是一个非常小的正数, 计算集合: $\boldsymbol{I}^{+}_{\boldsymbol{S}_{\text{S}}}=\{\tilde{\gamma}_i>\tau,\forall i\in\boldsymbol{S}_{\text{S}}\}$和$\boldsymbol{I}^{-}_{\boldsymbol{S}_{\text{S}}}=\{\tilde{\gamma}_i<-\tau,\forall i\in\boldsymbol{S}_{\text{S}}\}$. 此时可能的权更新为

$$\Delta\alpha_i^{\max}=\begin{cases}1-\alpha_i, & i\in\boldsymbol{I}^{+}_{\boldsymbol{S}_{\text{S}}},\\ -\alpha_i, & i\in\boldsymbol{I}^{-}_{\boldsymbol{S}_{\text{S}}}.\end{cases}$$

最大可能的调整量为

$$\Delta\mu^{\text{case2}}=\min(\Delta\alpha_i^{\max}/\tilde{\gamma}_i,\forall i\in\boldsymbol{I}^{+}_{\boldsymbol{S}_{\text{S}}}\cup\boldsymbol{I}^{-}_{\boldsymbol{S}_{\text{S}}}).$$

情况 3 $\boldsymbol{S}_{\text{R}}$集或者$\boldsymbol{S}_{\text{E}}$集中某个样本的$g_i$变为0. 与情况2类似, 计算集合:

$$\boldsymbol{I}^{+}_{\boldsymbol{S}_{\text{E}}}=\{\tilde{\varsigma}_i>\tau,\forall i\in\boldsymbol{S}_{\text{E}}\},\ \boldsymbol{I}^{-}_{\boldsymbol{S}_{\text{R}}}=\{\tilde{\varsigma}_i<-\tau,\forall i\in\boldsymbol{S}_{\text{R}}\},$$

此时可能的权更新为$\Delta g_i=-g_i$, 最大可能调整量为

$$\Delta\mu^{\text{case3}}=\min(-g_i/\tilde{\varsigma}_i,\ \forall i\in\boldsymbol{I}^{+}_{\boldsymbol{S}_{\text{E}}}\cup\boldsymbol{I}^{-}_{\boldsymbol{S}_{\text{R}}}).$$

最后, 将3种情况的最小值作为$\Delta\mu^{\max}$, 即

$$\Delta\mu^{\max}=\min(\Delta\mu^{\text{case1}},\Delta\mu^{\text{case2}},\Delta\mu^{\text{case3}}).\tag{19}$$

得到$\Delta\mu^{\max}$之后, 可以更新$b,\rho,\alpha_i,i\in\boldsymbol{S}_{\text{S}},g_i,i\in\boldsymbol{S}_{\text{R}}\cup\boldsymbol{S}_{\text{E}}$, $\boldsymbol{S}_{\text{R}}$集,$\boldsymbol{S}_{\text{S}}$集和$\boldsymbol{S}_{\text{E}}$集.

3.1.3 高效更新逆矩阵$\tilde{\boldsymbol{N}}$(Efficiently updating the inverse matrix $\tilde{\boldsymbol{N}}$)

一旦$\boldsymbol{S}_{\text{S}}$集中的组成成分发生变化, 则$\boldsymbol{S}'_{\text{S}}$集也会随之改变, 同样地, $\boldsymbol{S}_{\text{S}}$集中样本的标签可能会变得不再全部相同. 相应地, 矩阵$\tilde{\boldsymbol{M}}'$及其逆矩阵$\tilde{\boldsymbol{N}}$也要随之变化. 借助于下面的引理2可以高效更新逆矩阵$\tilde{\boldsymbol{N}}$.

引理 2 若$(s+1)\times(s+1)$维分块矩阵$\boldsymbol{B}$可以分割成如下的形式:

$$\boldsymbol{B}=\begin{bmatrix}\boldsymbol{A} & \boldsymbol{\eta}_t\\ \boldsymbol{\eta}_t^{\text{T}} & Q_{tt}\end{bmatrix},$$

其中: $\boldsymbol{\eta}_t=[Q_{1t}\ \ \cdots\ \ Q_{st}]^{\text{T}}$, $Q_{tt}\neq 0$是常数, $\boldsymbol{A}$是$s\times s$维矩阵且可逆, 则$\boldsymbol{B}^{-1}$可以按照如下规则扩展为

$$\boldsymbol{B}^{-1}=\begin{bmatrix}\boldsymbol{A}^{-1} & \boldsymbol{0}\\ \boldsymbol{0}^{\text{T}} & 0\end{bmatrix}+\frac{1}{k}\begin{bmatrix}\boldsymbol{\beta}_t\\ 1\end{bmatrix}\begin{bmatrix}\boldsymbol{\beta}_t\\ 1\end{bmatrix}^{\text{T}},$$

其中: $\boldsymbol{\beta}_t=-\boldsymbol{A}^{-1}\boldsymbol{\eta}_t$, $k=\boldsymbol{\eta}_t^{\text{T}}\boldsymbol{\beta}_t+Q_{tt}$.

若$(\boldsymbol{B}^{-1})_{tt}\neq 0$, $(\cdot)_{tt}$表示矩阵的第t行和第t列的元素, 而且$\boldsymbol{B}^{-1}$存在且已知, 则$\boldsymbol{A}^{-1}$可以按照如下规则收缩为

$$\boldsymbol{A}^{-1}=(\boldsymbol{B}^{-1})_{\backslash tt}-\frac{\big((\boldsymbol{B}^{-1})_{*t}\cdot(\boldsymbol{B}^{-1})_{t*}\big)_{\backslash tt}}{(\boldsymbol{B}^{-1})_{tt}},$$

其中: $*\neq t$, $(\cdot)_{\backslash tt}$表示删除矩阵的第t行和第t列后得到的缩减矩阵.

容易验证: $\boldsymbol{B}\boldsymbol{B}^{-1}=\boldsymbol{I}_{s+1}$和$\boldsymbol{A}\boldsymbol{A}^{-1}=\boldsymbol{I}_s$, 引理2的证明详见文献[15], 此处省略.

1) 样本$(\boldsymbol{x}_t,y'_t,z_t)$移入$\boldsymbol{S}'_{\text{S}}$集. 根据引理2, 逆矩阵$\tilde{\boldsymbol{N}}$可以按照如下规则扩展为

$$\tilde{\boldsymbol{N}}\leftarrow\begin{bmatrix}\tilde{\boldsymbol{N}} & \boldsymbol{0}\\ \boldsymbol{0}^{\text{T}} & 0\end{bmatrix}+\frac{1}{\tilde{\omega}_t^t}\begin{bmatrix}\tilde{\boldsymbol{\beta}}_t\\ 1\end{bmatrix}_{\backslash\tilde{\boldsymbol{F}}}\begin{bmatrix}\tilde{\boldsymbol{\beta}}_t\\ 1\end{bmatrix}^{\text{T}}_{\backslash\tilde{\boldsymbol{F}}},\tag{20}$$

其中:

$$\tilde{\boldsymbol{\beta}}_t=-\tilde{\boldsymbol{N}}\begin{bmatrix}z_t\\ 1\\ \boldsymbol{Q}_{\boldsymbol{S}_{\text{S}}t}\end{bmatrix}_{\backslash\tilde{\boldsymbol{F}}}\overset{\text{def}}{=}\begin{bmatrix}\tilde{\beta}_{\text{b}}^t\\ \tilde{\beta}_\rho^t\\ \tilde{\boldsymbol{\beta}}^t_{\boldsymbol{S}_{\text{S}}}\end{bmatrix}_{\backslash\tilde{\boldsymbol{F}}},$$

$$\boldsymbol{Q}_{\boldsymbol{S}_{\text{S}}t}=[Q_1t\ \ \cdots\ \ Q_{[\![\boldsymbol{S}_{\text{S}}]\!]t}]^{\text{T}},$$

$$\boxed{\tilde{\beta}_\rho^t}=0,\ \tilde{\boldsymbol{\beta}}^t_{\boldsymbol{F}}=\boldsymbol{0},\ \tilde{\boldsymbol{\beta}}^t_{\boldsymbol{S}_{\text{S}}t}=[\tilde{\beta}_1^t\ \ \cdots\ \ \tilde{\beta}_{[\![\boldsymbol{S}_{\text{S}}]\!]}]^{\text{T}},$$

$$\tilde{\omega}_t^t=\sum_{j\in\boldsymbol{S}_{\text{S}}}\tilde{\beta}_j^tQ_{tj}+z_t\tilde{\beta}_{\text{b}}^t+\tilde{\beta}_\rho^t+Q_{tt}.$$

2) 样本$(\boldsymbol{x}_t,y'_t,z_t)$移出$\boldsymbol{S}'_{\text{S}}$集. 根据引理2, 逆矩阵$\tilde{\boldsymbol{N}}$可以按照如下规则收缩为

$$\tilde{\boldsymbol{N}}\leftarrow\tilde{\boldsymbol{N}}_{\backslash tt}-\frac{(\tilde{\boldsymbol{N}}_{*t}\cdot\tilde{\boldsymbol{N}}_{t*})_{\backslash tt}}{\tilde{\boldsymbol{N}}_{tt}},\tag{21}$$

其中$*\neq t$.

需要注意的是, 样本$(\boldsymbol{x}_t,y'_t,z_t)$移出$\boldsymbol{S}'_{\text{S}}$集时, 根据逆矩阵$\tilde{\boldsymbol{N}}$的定义, 可以求得$\tilde{\gamma}_t=0$, 这就是说样本$(\boldsymbol{x}_t,y'_t,z_t)$移出$\boldsymbol{S}'_{\text{S}}$集后并不会导致$\boldsymbol{S}_{\text{S}}$集中样本的标签从不全部相同变成全部相同.

3) $\boldsymbol{S}_{\text{S}}$集中样本的标签从全部相同变成不再全部相同. 根据引理2, 逆矩阵$\tilde{\boldsymbol{N}}$可以按照如下规则扩展为

$$\tilde{\boldsymbol{N}}\leftarrow\begin{bmatrix}\tilde{\boldsymbol{N}} & \boldsymbol{0}\\ \boldsymbol{0}^{\text{T}} & 0\end{bmatrix}+\frac{1}{a}\begin{bmatrix}\boldsymbol{A}\\ 1\end{bmatrix}\begin{bmatrix}\boldsymbol{A}\\ 1\end{bmatrix}^{\text{T}},\tag{22}$$

其中:

$$\boldsymbol{A}=-\tilde{\boldsymbol{N}}\,[0\ \ \boldsymbol{I}^{\text{T}}_{\boldsymbol{S}'_{\text{S}}}]^{\text{T}},\ a=[0\ \ \boldsymbol{I}^{\text{T}}_{\boldsymbol{S}'_{\text{S}}}]\boldsymbol{A}.$$

3.2 宽松的绝缘增量调整(Relaxed adiabatic incremental adjustments, RAIA)

设置$\boldsymbol{S}_{\text{new}}$中两个样本的初始权系数为0. 根据第3.1节中的定理1可推断出$\boldsymbol{S}_{\text{new}}$中至多存在一个候选样本$(\boldsymbol{x}_{\text{c}},y'_{\text{c}},z_{\text{c}})$不满足KKT条件, 即$\alpha_{\text{c}}<1$和$g_{\text{c}}<0$. 若存在一个候选样本$(\boldsymbol{x}_{\text{c}},y'_{\text{c}},z_{\text{c}})$不满足KKT条件, 则就需要借助于RAIA实现所有样本满足除式(9)之外的其他KKT条件. 在逐步调整增大α_{c}, 并保证所有样本满足KKT条件的过程中, 式(7)到式(9)可以表示成如下的线性方程组:

$$\Delta g_i=\sum_{j\in\boldsymbol{S}_{\text{S}}}\Delta\alpha_jQ_{ij}+\Delta\alpha_{\text{c}}Q_{i\text{c}}+z_i\Delta b+\Delta\rho=0,\ \forall i\in\boldsymbol{S}_{\text{S}},\tag{23}$$

$$\sum_{j\in\boldsymbol{S}_{\text{S}}}z_j\Delta\alpha_j+z_{\text{c}}\Delta\alpha_{\text{c}}=0,\tag{24}$$

$$\sum_{j\in\boldsymbol{S}_{\text{S}}}\Delta\alpha_j+\Delta\alpha_{\text{c}}=\nu.\tag{25}$$

当$\boldsymbol{S}_{\text{S}}$集中的样本全部具有相同的标签, 而且候选

样本$(\boldsymbol{x}_{\rm c}, y'_{\rm c}, z_{\rm c})$ 的标签$z_{\rm c}$与$\boldsymbol{S}_{\rm S}$集中的间隔支持向量的标签z_i全部相同时, 式(24)变成:

$$\sum_{j\in\boldsymbol{S}_{\rm S}} \Delta\alpha_j + \Delta\alpha_{\rm c} = 0. \tag{26}$$

当ν不为零时, 式(25)和式(26)显然不能同时得到满足, 也就是说存在如表1所示的矛盾1. 矛盾1的存在将会导致无法有效调整α_i, 即出现解路径不可行更新问题. 为了有效地避免矛盾1, 本文在绝缘增量调整过程中首先放弃式(9), 然后在下面的第3.3节中通过一定的策略精确的恢复调整式(9). 放弃式(9)意味着$\Delta\rho=0$, 故称为宽松的绝缘增量调整.

表 1 RAIA中存在的矛盾1

Table 1 Contradiction 1 existed in RAIA

$z_{\rm c}$	$\boldsymbol{S}_{\rm S}$集中间隔支持向量的标签z_i	是否存在矛盾
$+1$	$+1$	是
$+1$	-1	否
-1	$+1$	否
-1	-1	是

式(23)和式(24)可以表示成如下的矩阵形式:

$$\begin{bmatrix} 0 & \boldsymbol{z}_{\boldsymbol{S}_{\rm S}}^{\rm T} \\ \boldsymbol{z}_{\boldsymbol{S}_{\rm S}} & \boldsymbol{Q}_{\boldsymbol{S}_{\rm S}\boldsymbol{S}_{\rm S}} \end{bmatrix} \begin{bmatrix} \Delta b \\ \Delta\boldsymbol{\alpha}_{\boldsymbol{S}_{\rm S}} \end{bmatrix} = -\begin{bmatrix} z_{\rm c} \\ \boldsymbol{Q}_{\boldsymbol{S}_{\rm S}{\rm c}} \end{bmatrix} \Delta\alpha_{\rm c}, \tag{27}$$

其中$\boldsymbol{Q}_{\boldsymbol{S}_{\rm S}{\rm c}} = [Q_1 c \ \cdots \ Q_{[\![\boldsymbol{S}_{\rm S}{\rm c}]\!]}]^{\rm T}$.

与PA类似, 令$\hat{\boldsymbol{F}} = \boldsymbol{F} \cup \{i_\rho\}$, 则可以定义缩减矩阵$\boldsymbol{M} = \tilde{\boldsymbol{M}}_{\backslash \hat{\boldsymbol{F}}\hat{\boldsymbol{F}}}$, 第4.1.2节中的定理5证明了$\boldsymbol{M}$必定存在逆矩阵$\boldsymbol{N}$, 因此有

$$\begin{bmatrix} \Delta b \\ \Delta\boldsymbol{\alpha}_{\boldsymbol{S}'_{\rm S}} \end{bmatrix} = -\boldsymbol{N} \begin{bmatrix} z_{\rm c} \\ \boldsymbol{Q}_{\boldsymbol{S}'_{\rm S}{\rm c}} \end{bmatrix} \Delta\alpha_{\rm c} \stackrel{\rm def}{=} \underbrace{\begin{bmatrix} \gamma_{\rm b}^{\rm c} \\ \boldsymbol{\gamma}_{\boldsymbol{S}'_{\rm S}}^{\rm c} \end{bmatrix}}_{\boldsymbol{\gamma}^{\rm c}} \Delta\alpha_{\rm c}, \tag{28}$$

其中$\boldsymbol{\gamma}_{\boldsymbol{S}'_{\rm S}}^{\rm c} = [\gamma_1^{\rm c} \ \cdots \ \gamma_{\boldsymbol{S}_{\rm S}}^{\rm c}]^{\rm T}$.

令$\boldsymbol{\gamma}_{\boldsymbol{F}}^{\rm c} = \boldsymbol{0}$, 将式(28)代入式(23), 可得

$$\Delta g_i = (\sum_{j\in\boldsymbol{S}_{\rm s}} \gamma_j^{\rm c} Q_{ij} + Q_{i{\rm c}} + z_i\gamma_{\rm b}^{\rm c})\Delta\alpha_{\rm c} \stackrel{\rm def}{=} \varsigma_i^{\rm c}\Delta\alpha_{\rm c},\ \forall i \in \boldsymbol{S}, \tag{29}$$

其中: $\varsigma_i^{\rm c} = 0,\ \forall i \in \boldsymbol{S}_{\rm S}$.

当$\Delta\rho=0$时, RAIA就是C&P算法, 故在RAIA中, 可以首先计算出最大调整增量$\Delta\alpha_{\rm c}^{\max}$, 然后更新$b, \rho, \alpha_i,\ i\in\boldsymbol{S}_{\rm S},\ g_i,\ i\in\boldsymbol{S}_{\rm R}\cup\boldsymbol{S}_{\rm E}$, $\boldsymbol{S}_{\rm R}$集, $\boldsymbol{S}_{\rm S}$集, $\boldsymbol{S}_{\rm E}$集并基于引理2高效更新逆矩阵$\boldsymbol{N}$, 此处不再赘述.

3.3 精确的恢复调整(Accurate restoration adjustments, ARA)

在RAIA之后, 需要采用一定的策略精确的恢复调整式(9), 故称为精确的恢复调整. 在调整$\sum_{i\in\boldsymbol{S}_{\rm S}}\alpha_i$的过程中, 为了保证所有样本满足KKT条件, 式(7)到式(9)可以表示成如下的线性方程组:

$$\Delta g_i = \sum_{j\in\boldsymbol{S}_{\rm S}} \Delta\alpha_j Q_{ij} + z_i\Delta b + \Delta\rho = 0, \forall i \in \boldsymbol{S}_{\rm S}, \tag{30}$$

$$\sum_{j\in\boldsymbol{S}_{\rm S}} z_j\Delta\alpha_j = 0, \tag{31}$$

$$\sum_{j\in\boldsymbol{S}_{\rm S}} \Delta\alpha_j + \Delta\vartheta = 0. \tag{32}$$

当$\boldsymbol{S}_{\rm S}$集中的样本全部具有相同的标签, 而且$\Delta\vartheta\neq 0$时, 式(31)和式(32)显然不能同时得到满足, 也就是说存在类似于RAIA中的矛盾1, 称为矛盾2. 矛盾2同样也会导致解路径不可行更新问题.

为了有效地避免矛盾2, 将式(32)变成如下的形式:

$$\sum_{j\in\boldsymbol{S}_{\rm s}} \Delta\alpha_j + \kappa\Delta\rho + \Delta\vartheta = 0, \tag{33}$$

其中: κ是任意的负数, $\kappa\Delta\rho$是附加项, 与$\Delta\vartheta$合在一起能够有效地避免ARA中矛盾2的出现, 详情参见第4.2.3节中的定理12后面的分析.

式(30)–(31)和式(33)可以表示成如下的矩阵形式:

$$\underbrace{\begin{bmatrix} 0 & 0 & \boldsymbol{z}_{\boldsymbol{S}_{\rm S}}^{\rm T} \\ 0 & \kappa & \boldsymbol{1}_{\boldsymbol{S}_{\rm S}}^{\rm T} \\ \boldsymbol{z}_{\boldsymbol{S}_{\rm S}} & \boldsymbol{1}_{\boldsymbol{S}_{\rm S}} & \boldsymbol{Q}_{\boldsymbol{S}_{\rm S}\boldsymbol{S}_{\rm S}} \end{bmatrix}}_{\hat{\boldsymbol{M}}} \begin{bmatrix} \Delta b \\ \Delta\rho \\ \Delta\boldsymbol{\alpha}_{\boldsymbol{S}_{\rm S}} \end{bmatrix} = -\begin{bmatrix} 0 \\ 1 \\ \boldsymbol{0} \end{bmatrix} \Delta\vartheta. \tag{34}$$

与PA类似, 定义缩减矩阵$\hat{\boldsymbol{M}}' = \hat{\boldsymbol{M}}_{\backslash\mathbf{FF}}$, 第4.1.3节中的定理6证明了$\hat{\boldsymbol{M}}'$必定存在逆矩阵$\hat{\boldsymbol{N}}$. 因此有

$$\begin{bmatrix} \Delta b \\ \Delta\rho \\ \Delta\boldsymbol{\alpha}_{\boldsymbol{S}_{\rm S}} \end{bmatrix} = -\hat{\boldsymbol{N}} \begin{bmatrix} 0 \\ 1 \\ \boldsymbol{0} \end{bmatrix} \Delta\vartheta \stackrel{\rm def}{=} \underbrace{\begin{bmatrix} \hat{\gamma}_b \\ \hat{\gamma}_\rho \\ \hat{\boldsymbol{\gamma}}_{\boldsymbol{S}'_{\rm S}} \end{bmatrix}}_{\hat{\boldsymbol{\gamma}}} \Delta\vartheta, \tag{35}$$

其中$\hat{\boldsymbol{\gamma}}_{\boldsymbol{S}'_{\rm S}} = [\hat{\gamma}_1 \ \cdots \ \hat{\gamma}_{[\![\boldsymbol{S}'_{\rm S}]\!]}]^{\rm T}$.

令$\hat{\boldsymbol{\gamma}}_{\boldsymbol{F}} = \boldsymbol{0}$, 将式(35)分别代入式(31)和式(33)可得

$$\Delta g_i = (\sum_{j\in\boldsymbol{S}_{\rm s}} \hat{\gamma}_j Q_{ij} + z_i\hat{\gamma}_b + \hat{\gamma}_\rho)\Delta\vartheta \stackrel{\rm def}{=} \hat{\varsigma}_i\Delta\vartheta,\ \forall i \in \boldsymbol{S}, \tag{36}$$

$$\sum_{j\in\boldsymbol{S}_{\rm S}} \Delta\alpha_j = -(\kappa\hat{\gamma}_\rho + 1)\Delta\vartheta. \tag{37}$$

式(36)中: $\hat{\varsigma}_i = 0,\ \forall i \in \boldsymbol{S}_{\rm S}$. 式(37)意味着通过$\Delta\vartheta$可以达到逐步调整$\sum_{i\in\boldsymbol{S}_{\rm S}}\alpha_i$的目的.

由于ARA不能直接应用于求解式(5)的最优解, 故可以采用类似在PA中确定$\Delta\mu^{\max}$的方法, 计算临界调整量$\Delta\vartheta^*$, 使得在每一步恢复调整过程中, 只有一个样本在$\boldsymbol{S}_{\rm R}$集, $\boldsymbol{S}_{\rm S}$集和$\boldsymbol{S}_{\rm E}$集之间迁移. 若

$$\sum_{i\in\boldsymbol{S}_{\rm S}} \alpha_i > \nu(l+1),$$

则需要计算最大调整量$\Delta\vartheta^{\max}$, 并令$\Delta\vartheta^* = \Delta\vartheta^{\max}$; 若$\sum_{i\in\boldsymbol{S}_{\rm S}}\alpha_i < \nu(l+1)$, 则计算最小调整量$\Delta\vartheta^{\min}$, 并令$\Delta\vartheta^* = \Delta\vartheta^{\min}$. 由于篇幅限制, 此处省略$\Delta\vartheta^*$的计算过程.

计算得到$\Delta\vartheta^*$之后, 根据$\Delta\vartheta^*$可以更新$b,\rho,\alpha_i,i\in\boldsymbol{S}_{\rm S},g_i,i\in\boldsymbol{S}_{\rm R}\cup\boldsymbol{S}_{\rm E}$, $\boldsymbol{S}_{\rm R}$集, $\boldsymbol{S}_{\rm S}$集, $\boldsymbol{S}_{\rm E}$集并基于引理2高效更新逆矩阵$\hat{\boldsymbol{N}}$, 此处不再赘述.

3.4 AIOSVR算法(AIOSVR algorithm)

AIOSVR算法的整体步骤如下:

步骤 1 读入一个新样本$(\boldsymbol{x}_{\rm new},y_{\rm new})$, 将其转换成$(\boldsymbol{x}_{\rm new},y'_{\rm new})$, 其中$y'_{\rm new}=y_{\rm new}/C$, 并令

$$\boldsymbol{S}_{\rm new}=\{(\boldsymbol{x}_{\rm new},y'_{\rm new},+1),(\boldsymbol{x}_{\rm new},y'_{\rm new},-1)\}.$$

步骤 2 更新

$$g_i\leftarrow\frac{l}{l+1}g_i,\ i\in\boldsymbol{S},\ b\leftarrow\frac{l}{l+1}b,$$
$$\rho\leftarrow\frac{l}{l+1}\rho,\ \mu=\frac{l}{l+1}.$$

步骤 3 计算$\tilde{\gamma},\tilde{\varsigma}_i,\ i\in\boldsymbol{S}_{\rm R}\cup\boldsymbol{S}_{\rm E}$, $\Delta\mu^{\max}$并更新$b,\rho,\alpha_i,i\in\boldsymbol{S}_{\rm S},g_i,i\in\boldsymbol{S}_{\rm R}\cup\boldsymbol{S}_{\rm E}$, $\boldsymbol{S}_{\rm R}$集, $\boldsymbol{S}_{\rm S}$集, $\boldsymbol{S}_{\rm E}$集和逆矩阵$\tilde{\boldsymbol{N}}$直至$\mu=1$(详见第3.1节中的PA).

步骤 4 基于逆矩阵$\tilde{\boldsymbol{N}}$计算逆矩阵$\boldsymbol{N}$(详见第3.5节).

步骤 5 更新$\boldsymbol{S}\leftarrow\boldsymbol{S}\cup\boldsymbol{S}_{\rm new}$.

步骤 6 初始化$\boldsymbol{S}_{\rm new}$中两个样本的权系数为0并计算相应的$g_i,i\in\boldsymbol{S}_{\rm new}$.

步骤 7 若$\boldsymbol{S}_{\rm new}$中存在一个候选样本$(\boldsymbol{x}_{\rm c},y'_{\rm c},z_{\rm c})$不满足$\alpha_{\rm c}<1$和$g_{\rm c}<0$, 则重复执行步骤8直至满足KKT条件后转步骤9; 否则直接转步骤9.

步骤 8 计算$\boldsymbol{\gamma}^{\rm c},\varsigma_i^{\rm c},i\in\boldsymbol{S}_{\rm R}\cup\boldsymbol{S}_{\rm E}$, $\Delta\alpha_{\rm c}^{\max}$并更新$b,\rho,\alpha_i,i\in\boldsymbol{S}_{\rm S},g_i,i\in\boldsymbol{S}_{\rm R}\cup\boldsymbol{S}_{\rm E}$, $\boldsymbol{S}_{\rm R}$集, $\boldsymbol{S}_{\rm S}$集, $\boldsymbol{S}_{\rm E}$集和逆矩阵$\boldsymbol{N}$ (详见第3.2节中的RAIA).

步骤 9 计算$\hat{\gamma},\hat{\varsigma}_i,i\in\boldsymbol{S}_{\rm R}\cup\boldsymbol{S}_{\rm E}$, $\Delta\vartheta^*$并更新$b,\rho,\alpha_i,i\in\boldsymbol{S}_{\rm S},g_i,i\in\boldsymbol{S}_{\rm R}\cup\boldsymbol{S}_{\rm E}$, $\boldsymbol{S}_{\rm R}$集, $\boldsymbol{S}_{\rm S}$集, $\boldsymbol{S}_{\rm E}$集和逆矩阵$\hat{\boldsymbol{N}}$直至$\sum\limits_{i\in\boldsymbol{S}_{\rm S}}\alpha_i=\nu(l+1)$(详见第3.3节中的ARA).

步骤 10 基于逆矩阵$\hat{\boldsymbol{N}}$计算逆矩阵$\tilde{\boldsymbol{N}}$(详见第3.5节).

3.5 下一轮调整之前的准备工作(Preparations for the next round of adjustment)

AIOSVR算法中, PA之后, 在下一轮ARA之前需要用到$\boldsymbol{N}$; 同样地, ARA之后, 在下一轮PA之前需要用到$\tilde{\boldsymbol{N}}$.

根据引理2, 基于逆矩阵$\tilde{\boldsymbol{N}}$, 逆矩阵$\boldsymbol{N}$可以按照如下规则扩展为

$$\boldsymbol{N}\leftarrow\begin{bmatrix}\tilde{\boldsymbol{N}} & \boldsymbol{0}\\ \boldsymbol{0}^{\rm T} & 0\end{bmatrix}+\frac{1}{b}\begin{bmatrix}\boldsymbol{A}\\ 1\end{bmatrix}\begin{bmatrix}\boldsymbol{A}\\ 1\end{bmatrix}^{\rm T},\tag{38}$$

其中:

$$\boldsymbol{A}=-\tilde{\boldsymbol{N}}\,[0\ \ \boldsymbol{1}_{\boldsymbol{S}'_{\rm S}}^{\rm T}]^{\rm T},\ b=[0\ \ \boldsymbol{1}_{\boldsymbol{S}'_{\rm S}}^{\rm T}]\boldsymbol{A}+\kappa.$$

类似地, 基于逆矩阵$\hat{\boldsymbol{N}}$, 依次经过如下3步可得逆矩阵$\tilde{\boldsymbol{N}}$.

首先, 根据引理2, 基于逆矩阵$\hat{\boldsymbol{N}}$, 可以依次按照如下规则收缩得到逆矩阵$\boldsymbol{Q}_{\boldsymbol{S}'_{\rm s}\boldsymbol{S}'_{\rm s}}^{-1}\overset{\rm def}{=}\boldsymbol{P}$:

$$\boldsymbol{P}\leftarrow\hat{\boldsymbol{N}}_{\backslash\rho\rho}-\frac{(\hat{\boldsymbol{N}}_{*\rho}\cdot\hat{\boldsymbol{N}}_{\rho*})_{\backslash\rho\rho}}{\hat{\boldsymbol{N}}_{\rho\rho}},\ \boldsymbol{P}\leftarrow\boldsymbol{P}_{\backslash bb}-\frac{(\boldsymbol{P}_{*b}\cdot\boldsymbol{P}_{b*})_{\backslash bb}}{\boldsymbol{P}_{bb}}.$$

其次, 按照$\boldsymbol{P}\leftarrow\dfrac{l}{l+1}\boldsymbol{P}$更新逆矩阵$\boldsymbol{P}$.

最后, 根据引理2, 逆矩阵$\tilde{\boldsymbol{N}}$可以按照如下规则扩展为

$$\tilde{\boldsymbol{N}}\leftarrow\begin{bmatrix}\boldsymbol{P} & \boldsymbol{0}\\ \boldsymbol{0}^{\rm T} & 0\end{bmatrix}+\frac{1}{c}\begin{bmatrix}\boldsymbol{B}\\ 1\end{bmatrix}\begin{bmatrix}\boldsymbol{B}\\ 1\end{bmatrix}^{\rm T},\tag{39}$$

其中:

$$\boldsymbol{B}=-\boldsymbol{P}\boldsymbol{z}_{\boldsymbol{S}'_{\rm S}},\ c=\boldsymbol{z}_{\boldsymbol{S}'_{\rm S}}^{\rm T}\boldsymbol{B},\ \boldsymbol{z}_{\boldsymbol{S}'_{\rm s}}=[z_1\ \ \cdots\ \ z_{[\![\boldsymbol{S}'_{\rm s}]\!]}]^{\rm T}.$$

4 可行性和有限收敛性分析(Feasibility and finite convergence analysis)

通过可行性分析, 证明AIOSVR算法中的逆矩阵$\tilde{\boldsymbol{N}},\boldsymbol{N}$和$\hat{\boldsymbol{N}}$总是存在的, 即算法的每一步调整都是可靠的; 通过有限收敛性分析, 证明AIOSVR算法经过有限次数调整将会最终收敛到式(5)的最优解.

4.1 可行性分析(Feasibility analysis)

下面分3部分, 分别对PA, RAIA和ARA的可行性进行理论分析.

假设 1 矩阵$\boldsymbol{Q}_{\boldsymbol{S}_{\rm S}\boldsymbol{S}_{\rm S}}$是正定阵.

容易验证当且仅当$\{z_1\phi(\boldsymbol{x}_1),\cdots,z_{[\![\boldsymbol{S}_{\rm S}]\!]}\phi(\boldsymbol{x}_{[\![\boldsymbol{S}_{\rm S}]\!]})\}$线性无关时, 矩阵$\boldsymbol{Q}_{\boldsymbol{S}_{\rm S}\boldsymbol{S}_{\rm S}}$是正定阵.

4.1.1 PA的可行性分析(Feasibility analysis of PA)

引理 3 若$m\times n$维矩阵$\boldsymbol{A}$满足$\rm rank(\boldsymbol{A})=m$, $n\times n$维矩阵$\boldsymbol{B}$是正定阵, 则$\boldsymbol{A}\boldsymbol{B}\boldsymbol{A}^{\rm T}$是正定阵.

引理 4 设矩阵$\boldsymbol{A},\boldsymbol{B},\boldsymbol{C}$和$\boldsymbol{D}$分别是$m\times m$, $m\times n$, $n\times m$和$n\times n$维矩阵, 则有

$$\det\left(\begin{bmatrix}\boldsymbol{A} & \boldsymbol{B}\\ \boldsymbol{C} & \boldsymbol{D}\end{bmatrix}\right)=$$
$$\begin{cases}\det(\boldsymbol{A})\det(\boldsymbol{D}-\boldsymbol{C}\boldsymbol{A}^{-1}\boldsymbol{B}),\ \boldsymbol{A}^{-1}\text{存在},\\ \det(\boldsymbol{D})\det(\boldsymbol{A}-\boldsymbol{B}\boldsymbol{D}^{-1}\boldsymbol{C}),\ \boldsymbol{D}^{-1}\text{存在}.\end{cases}$$

引理3和引理4的证明详见文献[15], 此处省略.

定理 2 在PA中, 必有$\det(\tilde{\boldsymbol{M}}')\neq0$, 也就是说矩阵$\tilde{\boldsymbol{M}}'$必定存在逆矩阵$\tilde{\boldsymbol{N}}$.

证 首先考虑$\boldsymbol{F}^+=\varnothing$时的情况. 若$\boldsymbol{S}_{\rm S}$集中的样本全部具有相同的标签, 则令$\boldsymbol{E}=\boldsymbol{z}_{\boldsymbol{S}_{\rm S}}^{\rm T}$, 此时$\rm rank(\boldsymbol{E})=1$; 否则令$\boldsymbol{E}=[\boldsymbol{z}_{\boldsymbol{S}_{\rm S}}\ \ \boldsymbol{1}_{\boldsymbol{S}_{\rm S}}]^{\rm T}$, 此时$\rm rank(\boldsymbol{E})=2$. 根据假设1, $\boldsymbol{Q}_{\boldsymbol{S}_{\rm S}\boldsymbol{S}_{\rm S}}$是正定阵, 这就意味着$\boldsymbol{Q}_{\boldsymbol{S}_{\rm S}\boldsymbol{S}_{\rm S}}^{-1}$存在, 故根

据引理4可得:

$$\det(\tilde{\boldsymbol{M}}')=\det(\boldsymbol{Q}_{\boldsymbol{S}_{\rm S}\boldsymbol{S}_{\rm S}})\det(\boldsymbol{0}-\boldsymbol{E}\boldsymbol{Q}_{\boldsymbol{S}_{\rm S}\boldsymbol{S}_{\rm S}}^{-1}\boldsymbol{E}^{\rm T})=$$
$$(-1)^{{\rm rank}(\boldsymbol{E})}\det(\boldsymbol{Q}_{\boldsymbol{S}_{\rm S}\boldsymbol{S}_{\rm S}})\det(\boldsymbol{0}-\boldsymbol{E}\boldsymbol{Q}_{\boldsymbol{S}_{\rm S}\boldsymbol{S}_{\rm S}}^{-1}\boldsymbol{E}^{\rm T}).$$

进一步, $\boldsymbol{Q}_{\boldsymbol{S}_{\rm S}\boldsymbol{S}_{\rm S}}$是正定阵意味着$\boldsymbol{Q}_{\boldsymbol{S}_{\rm S}\boldsymbol{S}_{\rm S}}^{-1}$也是正定阵而且$\det(\boldsymbol{Q}_{\boldsymbol{S}_{\rm S}\boldsymbol{S}_{\rm S}})>0$. 根据引理3可得$\boldsymbol{E}\boldsymbol{Q}_{\boldsymbol{S}_{\rm S}\boldsymbol{S}_{\rm S}}^{-1}\boldsymbol{E}^{\rm T}$也是正定阵, 故$\det(\boldsymbol{E}\boldsymbol{Q}_{\boldsymbol{S}_{\rm S}\boldsymbol{S}_{\rm S}}^{-1}\boldsymbol{E}^{\rm T})>0$. 因此有$\det(\tilde{\boldsymbol{M}}')\neq 0$.

接着考虑$\boldsymbol{F}^{+}\neq\varnothing$时的情况. 分别将$\tilde{\boldsymbol{M}}'$的第$i_2$行加到第$i_1$行, 第$i_2$列加到第$i_1$列后可得如下的矩阵$\tilde{\boldsymbol{M}}''$:

$$\tilde{\boldsymbol{M}}''=\begin{bmatrix}0 & 0 & z'_{i_1} & \boldsymbol{z}_{\boldsymbol{S}''_{\rm S}}^{\rm T}\\ 0 & 0 & e'_{i_1} & \boldsymbol{1}_{\boldsymbol{S}''_{\rm S}}^{\rm T}\\ z'_{i_1} & e'_{i_1} & 0 & \boldsymbol{0}\\ \boldsymbol{z}_{\boldsymbol{S}''_{\rm S}} & \boldsymbol{1}_{\boldsymbol{S}''_{\rm S}} & \boldsymbol{0} & \boldsymbol{Q}_{\boldsymbol{S}''_{\rm S}\boldsymbol{S}''_{\rm S}}\end{bmatrix}_{\backslash\tilde{\boldsymbol{F}}^{-}\tilde{\boldsymbol{F}}^{-}},$$

其中: $i_1\in\boldsymbol{F}'=\boldsymbol{F}^{+}-\boldsymbol{F}$, $(\boldsymbol{x}_{i_2},y'_{i_2},z_{i_2})\in\boldsymbol{S}''_{\rm S}$ 且满足

$$(\boldsymbol{x}_{i_1},y'_{i_1})=(\boldsymbol{x}_{i_2},y'_{i_2}),\ z_{i_1}+z_{i_2}=0,$$
$$\boldsymbol{S}''_{\rm S}=\boldsymbol{S}_{\rm S}-\boldsymbol{F}^{+},\ \tilde{\boldsymbol{F}}^{-}=\tilde{\boldsymbol{F}}-\boldsymbol{F}'.$$

在假设1的条件下, 容易验证$\boldsymbol{Q}_{\boldsymbol{S}''_{\rm S}\boldsymbol{S}''_{\rm S}}$是正定阵. 若$\boldsymbol{S}_{\rm S}$集中的样本全部具有相同的标签, 则令$\boldsymbol{E}'=\boldsymbol{z}_{\boldsymbol{S}''_{\rm S}}^{\rm T}$, $\boldsymbol{E}''=z'_{i_1}$, 此时${\rm rank}(\boldsymbol{E}'')=1$; 否则令$\boldsymbol{E}'=[\boldsymbol{z}_{\boldsymbol{S}''_{\rm S}}\ \ \boldsymbol{1}_{\boldsymbol{S}''_{\rm S}}]^{\rm T}$, $\boldsymbol{E}''=[z'_{i_1}\ \ e'_{i_1}]^{\rm T}$, 此时${\rm rank}(\boldsymbol{E}'')=1$.

若${\rm rank}(\boldsymbol{E}')$等于$\boldsymbol{E}'$的列数, 则根据引理4可得:

$$\det(\tilde{\boldsymbol{M}}')=\det(\tilde{\boldsymbol{M}}'')=$$
$$(-1)^{{\rm rank}(\boldsymbol{E}'')}\det(\boldsymbol{Q}_{\boldsymbol{S}''_{\rm S}\boldsymbol{S}''_{\rm S}})\det(\hat{\boldsymbol{E}})\det(\boldsymbol{E}''^{\rm T}\hat{\boldsymbol{E}}^{-1}\boldsymbol{E}''),$$

其中: $\hat{\boldsymbol{E}}=\boldsymbol{E}'\boldsymbol{Q}_{\boldsymbol{S}''_{\rm S}\boldsymbol{S}''_{\rm S}}^{-1}\boldsymbol{E}'^{\rm T}$. 容易验证$\hat{\boldsymbol{E}}$和$\boldsymbol{E}''^{\rm T}\hat{\boldsymbol{E}}^{-1}\boldsymbol{E}''$都是正定阵, 因此有$\det(\tilde{\boldsymbol{M}}')\neq 0$.

若${\rm rank}(\boldsymbol{E}')$不等于$\boldsymbol{E}'$的列数, 即: $\boldsymbol{z}_{\boldsymbol{S}''_{\rm S}}=\pm\boldsymbol{1}_{\boldsymbol{S}''_{\rm S}}$, 则在$\tilde{\boldsymbol{M}}''$的基础上分别将$\tilde{\boldsymbol{M}}''$中$b$对应的行(列)加到(减去)$\rho$对应的行(列)后可得如下的矩阵$\tilde{\boldsymbol{M}}'''$:

$$\tilde{\boldsymbol{M}}'''=\begin{bmatrix}0 & 0 & z'_{i_1} & \boldsymbol{z}_{\boldsymbol{S}''_{\rm S}}^{\rm T}\\ 0 & 0 & e''_{i_1} & \boldsymbol{0}\\ z'_{i_1} & e''_{i_1} & 0 & \boldsymbol{0}\\ \boldsymbol{z}_{\boldsymbol{S}''_{\rm S}} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{Q}_{\boldsymbol{S}''_{\rm S}\boldsymbol{S}''_{\rm S}}\end{bmatrix}.$$

根据引理4可得

$$\det(\tilde{\boldsymbol{M}}')=\det(\tilde{\boldsymbol{M}}''')=$$
$$\det(\boldsymbol{Q}_{\boldsymbol{S}''_{\rm S}\boldsymbol{S}''_{\rm S}})\det(\hat{\boldsymbol{E}}')\det(\hat{\boldsymbol{E}}'')\det(e''_{i_1}\hat{\boldsymbol{E}}''^{-1}e''_{i_1}),$$

其中: $\hat{\boldsymbol{E}}'=\boldsymbol{z}_{\boldsymbol{S}''_{\rm S}}^{\rm T}\boldsymbol{Q}_{\boldsymbol{S}''_{\rm S}\boldsymbol{S}''_{\rm S}}^{-1}\boldsymbol{z}_{\boldsymbol{S}''_{\rm S}}$, $\hat{\boldsymbol{E}}''=z'_{i_1}\hat{\boldsymbol{E}}'^{-1}z'_{i_1}$.

由于${\rm rank}(\boldsymbol{z}_{\boldsymbol{S}''_{\rm S}})=1$, $z'_{i_1}\neq 0$和$e''_{i_1}\neq 0$, 容易验证$\hat{\boldsymbol{E}}'$, $\hat{\boldsymbol{E}}''$和$e''_{i_1}\hat{\boldsymbol{E}}''^{-1}e''_{i_1}$都是正定阵, 因此有$\det(\tilde{\boldsymbol{M}}')\neq 0$.

综上可得: 不管$\boldsymbol{F}^{+}$是否为$\varnothing$都必有: $\det(\tilde{\boldsymbol{M}}')\neq 0$. 矩阵存在逆矩阵的充要条件是该矩阵的行列式不为0, 据此可推断出矩阵$\tilde{\boldsymbol{M}}'$必定存在逆矩阵$\tilde{\boldsymbol{N}}$.　　证毕.

定理 3　在PA中, $\boldsymbol{S}'_{\rm S}$集总是非空的.

证　若$\boldsymbol{S}'_{\rm S}=\{(\boldsymbol{x}_t,y'_t,z_t)\}$, 根据式(18), 逆矩阵$\tilde{\boldsymbol{N}}$的定义和定理2容易证得$\tilde{\gamma}_t=0$. 因此, 样本$(\boldsymbol{x}_t,y'_t,z_t)$将不会移出$\boldsymbol{S}'_{\rm S}$集, 也就是说在PA中, $\boldsymbol{S}'_{\rm S}$集总是非空的.　　证毕.

4.1.2　RAIA的可行性分析(Feasibility analysis of RAIA)

引理 5　在RAIA中, 若$\boldsymbol{S}'_{\rm S}=\{(\boldsymbol{x}_t,y'_t,z_t)\}$, 则有$\gamma_t^{\rm c}=-z_tz_{\rm c}$.

证　若$\boldsymbol{S}'_{\rm S}=\{(\boldsymbol{x}_t,y'_t,z_t)\}$, 则根据逆矩阵$\boldsymbol{N}$的定义求得$\boldsymbol{N}$后代入式(28)容易证得: $\gamma_t^{\rm c}=-z_tz_{\rm c}$.

证毕.

定理 4　在RAIA中, $\boldsymbol{S}'_{\rm S}$集总是非空的.

证　根据引理5和式(7), 容易验证b值的改变方向必须为$z_{\rm c}$才能保证移入$\boldsymbol{S}'_{\rm S}$集中的样本不会在紧接着的下一轮调整中移出$\boldsymbol{S}'_{\rm S}$集.

若$\boldsymbol{S}'_{\rm S}$集为空, 则可以通过改变偏置b值的大小直至$\boldsymbol{S}_{\rm R}$集或者$\boldsymbol{S}_{\rm E}$集中的某个样本移入$\boldsymbol{S}'_{\rm S}$集, 从而保证$\boldsymbol{S}'_{\rm S}$集是非空的.　　证毕.

定理 5　在RAIA中, 必有$\det(\boldsymbol{M})<0$, 也就是说矩阵$\boldsymbol{M}$必定存在逆矩阵$\boldsymbol{N}$.

证　定理4表明在RAIA中, $\boldsymbol{S}'_{\rm S}$集总是非空的. 类似于定理2前半部分的证明可得

$$\det(\boldsymbol{M})=-\det(\boldsymbol{Q}_{\boldsymbol{S}'_{\rm S}\boldsymbol{S}'_{\rm S}})\det(\boldsymbol{z}_{\boldsymbol{S}'_{\rm S}}^{\rm T}\boldsymbol{Q}_{\boldsymbol{S}'_{\rm S}\boldsymbol{S}'_{\rm S}}^{-1}\boldsymbol{z}_{\boldsymbol{S}'_{\rm S}})<0.$$

据此可推断出矩阵$\boldsymbol{M}$必定存在逆矩阵$\boldsymbol{N}$.

证毕.

4.1.3　ARA 的可行性分析 (Feasibility analysis of ARA)

定理 6　在ARA中, 若$\kappa<0$, 则必有: $\det(\hat{\boldsymbol{M}}')>0$, 也就是说矩阵$\hat{\boldsymbol{M}}'$必定存在逆矩阵$\hat{\boldsymbol{N}}$.

证　将矩阵$\hat{\boldsymbol{M}}'$和$\tilde{\boldsymbol{M}}$的行列式按第2行展开后可得:

$$\det(\hat{\boldsymbol{M}}')=\det(\tilde{\boldsymbol{M}}_{\backslash\boldsymbol{F}\boldsymbol{F}})+\kappa\det(\boldsymbol{M}).$$

根据矩阵$\tilde{\boldsymbol{M}}$的定义, 引理4和柯西-施瓦兹不等式, 容易证得:

$$\det(\tilde{\boldsymbol{M}}_{\backslash\boldsymbol{FF}})=$$
$$\det(\boldsymbol{Q}_{\boldsymbol{S}'_{\rm S}\boldsymbol{S}'_{\rm S}})(\boldsymbol{z}_{\boldsymbol{S}'_{\rm S}}^{\rm T}\boldsymbol{Q}_{\boldsymbol{S}'_{\rm S}\boldsymbol{S}'_{\rm S}}^{-1}\boldsymbol{z}_{\boldsymbol{S}'_{\rm S}}\boldsymbol{1}_{\boldsymbol{S}'_{\rm S}}^{\rm T}\boldsymbol{Q}_{\boldsymbol{S}'_{\rm S}\boldsymbol{S}'_{\rm S}}^{-1}\boldsymbol{1}_{\boldsymbol{S}'_{\rm S}}-$$
$$\boldsymbol{z}_{\boldsymbol{S}'_{\rm S}}^{\rm T}\boldsymbol{Q}_{\boldsymbol{S}'_{\rm S}\boldsymbol{S}'_{\rm S}}^{-1}\boldsymbol{1}_{\boldsymbol{S}'_{\rm S}}\cdot\boldsymbol{1}_{\boldsymbol{S}'_{\rm S}}^{\rm T}\boldsymbol{Q}_{\boldsymbol{S}'_{\rm S}\boldsymbol{S}'_{\rm S}}^{-1}\boldsymbol{z}_{\boldsymbol{S}'_{\rm S}})\geqslant 0.$$

若$\kappa<0$, 则根据定理5可得$\kappa\det(\boldsymbol{M})>0$.

综上可得: $\det(\hat{\boldsymbol{M}}')>0$, 也就意味着矩阵$\hat{\boldsymbol{M}}'$必定存在逆矩阵$\hat{\boldsymbol{N}}$.　　证毕.

定理 7　在ARA中, $\boldsymbol{S}'_{\rm S}$集总是非空的.

证　若$\boldsymbol{S}'_{\rm S}=\{(\boldsymbol{x}_t,y'_t,z_t)\}$, 根据式(35), 逆矩阵$\hat{\boldsymbol{N}}$的定义和定理6容易证得: $\hat{\gamma}_t=0$. 因此, 样本$(\boldsymbol{x}_t,y'_t,z_t)$将不会移出$\boldsymbol{S}'_{\rm S}$集, 也就是说在ARA中$\boldsymbol{S}'_{\rm S}$集总是非

空的. 证毕.

4.2 有限收敛性分析(Finite convergence analysis)

4.2.1 PA的有限收敛性分析(Finite convergence analysis of PA)

引理 6 在PA中, 必有: $\tilde{\gamma}_i \neq \pm\infty$, $\forall i \in \boldsymbol{S}_{\rm S}$和$\tilde{\varsigma}_i \neq \pm\infty$, $\forall i \in \boldsymbol{S}_{\rm R} \cup \boldsymbol{S}_{\rm E}$.

证 根据式(18)和逆矩阵$\tilde{\boldsymbol{N}}$的定义可得

$$\tilde{\gamma}_i = -\frac{\sum\limits_{j\in \boldsymbol{S}_{\rm S}} (-1)^{i_i+i_j}\det(\tilde{\boldsymbol{M}}'_{\backslash ji})Q_{ji}}{\det(\tilde{\boldsymbol{M}}')}.$$

若$\boldsymbol{S}_{\rm S}$集中某个样本的$\tilde{\gamma}_i = \pm\infty$, 则必有$\det(\tilde{\boldsymbol{M}}') = 0$, 这显然与定理2矛盾, 故有$\tilde{\gamma}_i \neq \pm\infty, \forall i \in \boldsymbol{S}_{\rm S}$.

同理根据式(15)和逆矩阵$\tilde{\boldsymbol{N}}$的定义可证: $\tilde{\varsigma}_i \neq \pm\infty, \forall i \in \boldsymbol{S}_{\rm R} \cup \boldsymbol{S}_{\rm E}$. 证毕.

推论 1 在PA中, 必有$\Delta\mu^{\max} > 0$.

根据引理6, 结合第3.1.2节中关于最大调整量$\Delta\mu^{\max}$的描述, 容易验证推论1的结论是正确的.

定理 8 在PA中, 式(6)中的目标函数W是单调递增的.

证 假设前一次调整用索引号$k-1$表示, 紧接着的下一次调整用索引号k表示, 对应的目标函数W分别用$W^{[k-1]}$和$W^{[k]}$表示并令$\tilde{\boldsymbol{\gamma}}_{\boldsymbol{S}_{\rm R}} = \boldsymbol{0}$和$\tilde{\boldsymbol{\gamma}}_{\boldsymbol{S}_{\rm E}} = \boldsymbol{0}$.

在式(7)–(9), 式(11)–(12)和式(18)的基础上, 整理比较后目标函数W可以表示为

$$W^{[k]} = W^{[k-1]} + \frac{1}{2}\sum_{i\in \boldsymbol{S}^{[k]}}\sum_{j\in \boldsymbol{S}^{[k]}} \tilde{\gamma}_i^{[k-1]} Q_{ij} \tilde{\gamma}_j^{[k-1]} (\Delta\mu^{\max[k-1]})^2.$$

根据推论1和假设1, 容易验证:

$$\frac{1}{2}\sum_{i\in \boldsymbol{S}^{[k]}}\sum_{j\in \boldsymbol{S}^{[k]}} \tilde{\gamma}_i^{[k-1]} Q_{ij} \tilde{\gamma}_j^{[k-1]} (\Delta\mu^{\max[k-1]})^2 > 0.$$

因此有$W^{[k]} > W^{[k-1]}$, 这就意味着在PA中, 目标函数W是单调递增的. 证毕.

定理 9 在PA中, 经过有限次数调整, 式(6)中的目标函数W将会收敛到$\min\limits_{0\leqslant\alpha_i\leqslant 1} W$的最优解.

证 定义在PA中生成的能量函数序列为$(W^{[1]}, W^{[2]}, \cdots)$. 根据定理8可得该能量函数序列必定是无限的单调递增序列. 设$W^{\rm opt}$是$\min\limits_{0\leqslant\alpha_i\leqslant 1} W$的最优解, 容易验证$\lim\limits_{k\to\infty} W^{[k]} \leqslant W^{\rm opt}$, 即$W^{[k]}(k=1,\cdots,\infty)$是收敛的, 显然这与$W^{[k]} > W^{[k-1]}$相互矛盾, 故序列$W^{[k]}(k=1,\cdots,\infty)$是有限长序列, 表示为$(W^{[1]}, W^{[2]}, \cdots, W^{[m]})$.

假设该有限长序列不能收敛到$\min\limits_{0\leqslant\alpha_i\leqslant 1} W$的最优解, 根据强对偶理论可以得到第$m$次迭代将不满足$\min\limits_{0\leqslant\alpha_i\leqslant 1} W$的KKT条件, 显然这与增量调整终止在第$m$次迭代相互矛盾. 故经过有限次数调整$(W^{[1]}, W^{[2]}, \cdots, W^{[m]})$必将收敛到最优解. 证毕.

4.2.2 RAIA有限收敛性分析(Finite convergence analysis of RAIA)

引理 7 在RAIA中, 必有

$$\gamma_i^{\rm c} \neq \pm\infty, \forall i \in \boldsymbol{S}_{\rm S};\ \varsigma_i^{\rm c} \neq \pm\infty,\ \forall i \in \boldsymbol{S}_{\rm R} \cup \boldsymbol{S}_{\rm E}.$$

类似引理6, 根据式(28)–(29)和逆矩阵$\boldsymbol{N}$的定义容易验证引理7的结论是正确的.

推论 2 在RAIA中, 必有$\Delta\alpha_{\rm c}^{\max} > 0$.

根据引理6, 结合C&P算法[3]中关于$\Delta\alpha_{\rm c}^{\max}$的描述, 容易验证推论2的结论是正确的.

定理 10 在RAIA中, 式(6)中的目标函数W是单调递减的.

证 $W^{[k-1]}$和$W^{[k]}$的定义与证明定理8时相同, 并令$\boldsymbol{\gamma}_{\boldsymbol{S}_{\rm R}}^{\rm c} = \boldsymbol{0}$, $\boldsymbol{\gamma}_{\boldsymbol{S}_{\rm E}}^{\rm c} = \boldsymbol{0}$和$\gamma_{\rm c}^{\rm c} = 1$.

在式(7)–(8), 式(24)和式(28)–(29)的基础上, 整理比较后目标函数W可以表示为

$$W^{[k]} = W^{[k-1]} + (g_{\rm c}^{[k-1]} + \frac{1}{2}\varsigma_{\rm c}^{c[k-1]}\Delta\alpha_{\rm c}^{\max[k-1]}) \times \Delta\alpha_{\rm c}^{\max[k-1]}.$$

进一步, 根据推论2可得$\Delta\alpha_{\rm c}^{\max[k-1]} > 0$. 此外, 容易验证

$$g_{\rm c}^{[k-1]} + \frac{1}{2}\varsigma_{\rm c}^{c[k-1]}\Delta\alpha_{\rm c}^{\max[k-1]} < 0,$$

因此有$W^{[k]} < W^{[k-1]}$, 这就意味着在RAIA中, 目标函数W是单调递减的. 证毕.

定理 11 在RAIA中, 经过有限次数调整后, 候选样本$(\boldsymbol{x}_{\rm c}, y'_{\rm c}, z_{\rm c})$将会满足除式(9)以外的其他所有KKT条件.

证 令$(\alpha_{\rm c}^{[1]}, \alpha_{\rm c}^{[2]}, \cdots)$, $(g_{\rm c}^{[1]}, g_{\rm c}^{[2]}, \cdots)$, $(W^{[1]}, W^{[2]}, \cdots)$分别代表在RAIA中产生的3个无限长序列. 根据定理10可得$(W^{[1]}, W^{[2]}, \cdots)$必定是一个无限的单调递减序列. 此外容易证明每个$W^{[k]}$与$\boldsymbol{S}_{\rm S}$集是一一对应关系, 而且$\boldsymbol{S}_{\rm S}$集的组合数是有限的, 故序列$W^{[k]}(k=1,\cdots,\infty)$必定是有限长序列, 可以表示为$(W^{[1]}, W^{[2]}, \cdots, W^{[m]})$. 同理可证: $\alpha_{\rm c}^{[k]}(k=1,\cdots,\infty)$和$g_{\rm c}^{[k]}(k=1,\cdots,\infty)$也必定是有限长序列. 根据推论2和$\varsigma_{\rm c}^{\rm c} > 0$, 容易验证有限长序列$(\alpha_{\rm c}^{[1]}, \alpha_{\rm c}^{[2]}, \cdots, \alpha_{\rm c}^{[m]})$和$(g_{\rm c}^{[1]}, g_{\rm c}^{[2]}, \cdots, g_{\rm c}^{[m]})$是单调递增的.

假设$(\alpha_{\rm c}^{[1]}, \alpha_{\rm c}^{[2]}, \cdots, \alpha_{\rm c}^{[m]})$和$(g_{\rm c}^{[1]}, g_{\rm c}^{[2]}, \cdots, g_{\rm c}^{[m]})$不会收敛到$\alpha_{\rm c}^{m} < 1$和$g_{\rm c}^{m} < 1$, 显然这与绝缘增量调整终止在第$m$次调整相互矛盾. 因此, 有限长序列$(\alpha_{\rm c}^{[1]}, \alpha_{\rm c}^{[2]}, \cdots, \alpha_{\rm c}^{[m]})$和$(g_{\rm c}^{[1]}, g_{\rm c}^{[2]}, \cdots, g_{\rm c}^{[m]})$经过有限次数调整必将收敛到问题对应的KKT条件. 证毕.

4.2.3 ARA的有限收敛性分析(Finite convergence analysis of ARA)

引理 8　在ARA中, 必有$\hat{\gamma}_i \neq \pm\infty$, $\forall i \in \boldsymbol{S}_{\rm S}$和$\hat{\varsigma}_i \neq \pm\infty, \forall i \in \boldsymbol{S}_{\rm R} \cup \boldsymbol{S}_{\rm E}$.

类似引理6, 根据式(35), 式(36)和逆矩阵$\hat{\boldsymbol{N}}$的定义容易验证引理8的结论是正确的.

定理 12　在ARA中, 若$\kappa < 0$, 则必有$\kappa\hat{\gamma}_\rho + 1 \geqslant 0$, 等号当且仅当$\boldsymbol{S}'_{\rm S}$集中的样本全部具有相同的标签时成立.

证　根据式(35)和逆矩阵$\hat{\boldsymbol{N}}$的定义容易证得

$$\kappa\hat{\gamma}_\rho + 1 = \frac{\det(\hat{\boldsymbol{M}}') - \kappa\det(\boldsymbol{M})}{\det(\hat{\boldsymbol{M}}')}.$$

若$\kappa < 0$, 则根据定理6的证明过程容易验证$\kappa\hat{\gamma}_\rho + 1 \geqslant 0$. 等号当且仅当$\det(\tilde{\boldsymbol{M}}_{\backslash \boldsymbol{FF}}) = \det(\hat{\boldsymbol{M}}') - \kappa\det(\boldsymbol{M}) = 0$时成立. 容易验证此时必有$\boldsymbol{z}_{\boldsymbol{S}'_{\rm S}} = +\boldsymbol{1}_{\boldsymbol{S}'_{\rm S}}$或者$\boldsymbol{z}_{\boldsymbol{S}'_{\rm S}} = -\boldsymbol{1}_{\boldsymbol{S}'_{\rm S}}$, 也就意味着$\boldsymbol{S}'_{\rm S}$集中的样本全部具有相同的标签.　证毕.

定理12表明: 当$\boldsymbol{S}'_{\rm S}$集中的样本全部具有相同的标签时, 式(31)和式(33)也能同时得到满足, 因此能够有效地避免ARA中矛盾2的出现.

推论 3　在ARA中, 若$\sum\limits_{i \in \boldsymbol{S}_{\rm S}} \alpha_i > \nu(l+1)$, 则有$\Delta\vartheta^* > 0$; 若$\sum\limits_{i \in \boldsymbol{S}_{\rm S}} \alpha_i < \nu(l+1)$, 则有$\Delta\vartheta^* < 0$.

根据引理8和定理12, 结合第3.3节中关于临界调整量$\Delta\vartheta^*$的描述, 容易验证推论3的结论是正确的.

定理 13　在ARA中, 式(6)中的目标函数W是单调递增的.

证　$W^{[k-1]}$和$W^{[k]}$的定义与证明定理8时相同, 并令$\hat{\boldsymbol{\gamma}}_{\boldsymbol{S}_{\rm R}} = \boldsymbol{0}$和$\hat{\boldsymbol{\gamma}}_{\boldsymbol{S}_{\rm E}} = \boldsymbol{0}$.

在式(7)–(8)和式(35)–(37)的基础上, 整理比较后目标函数W可以表示为

$$W^{[k]} = W^{[k-1]} + \hat{\gamma}_\rho^{[k-1]}\Delta\vartheta^{*[k-1]}\Big(\sum_{i \in \boldsymbol{S}^{[k]}} \alpha_i^{[k-1]} - \nu(l+1) - \frac{1}{2}(\kappa\hat{\gamma}_\rho^{[k-1]} + 1)\Delta\vartheta^{*[k-1]}\Big).$$

进一步, 根据推论3和定理12容易验证

$$\Delta\vartheta^{*[k-1]}\Big(\sum_{i \in \boldsymbol{S}^{[k]}} \alpha_i^{[k-1]} - \nu(l+1) - \frac{1}{2}(\kappa\hat{\gamma}_\rho^{[k-1]} + 1)\Delta\vartheta^{*[k-1]}\Big) > 0.$$

容易验证$\hat{\gamma}_\rho^{[k-1]} > 0$. 因此有$W^{[k]} > W^{[k-1]}$, 这就意味着在ARA中, 目标函数$W$是单调递增的.

证毕.

定理 14　在ARA中, 经过有限次数调整, 式(6)中的目标函数W将会收敛到$\min\limits_{0 \leqslant \alpha_i \leqslant 1} W$的最优解.

类似定理9, 根据定理13和强对偶理论容易验证定理14的结论是正确的. 此处不再赘述.

5 仿真实验(Simulation experiments)

5.1 实验设计和参数设置(Design of experiments and setting of parameters)

仿真实验分成两部分: 一是展示AIOSVR算法的有效性和有限收敛性, 二是体现AIOSVR算法在学习时间效率上的优越性.

仿真软件采用的是MATLAB2010a, 电脑配置为主频3.10GHz, 处理器为Intel$^{\rm R}$Core$^{\rm TM}$i5–2400, 内存为4 GB. 实验中使用如表2所示的4个基准回归测试数据集.

表 2 实验中使用的基准测试数据集

Table 2 Benchmark datasets used in the experiments

数据集	最大训练样本个数	属性数
Triazines	186	60
Housing	506	13
Cpusmall	8192	12
Cadata	20640	8

本文基于MATLAB软件实现AIOSVR算法, 而且为了在同一平台上比较运行时间效率, ν-SVR学习算法采用的是Chen等人用MATLAB软件实现的热启动(warm start)序列最小最优化算法[16]. 实验中采用3种不同的核函数:

1) 线性核: $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle$;

2) 多项式核: $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle + 1)^2$;

3) 径向基核: $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-\|\boldsymbol{x}_{\rm i} - \boldsymbol{x}_{\rm j}\|/2\sigma^2)$, 其中$\sigma = 0.7071$.

惩罚参数C的作用是将y_i变成$y'_i = y_i/C$, 故设置$C = 100$. 此外, 在ARA中容易验证κ仅与$\det(\boldsymbol{Q}_{\boldsymbol{S}_{\rm S}\boldsymbol{S}_{\rm S}})$有关, 但不会导致$\boldsymbol{S}_{\rm R}$集, $\boldsymbol{S}_{\rm S}$集和$\boldsymbol{S}_{\rm E}$集的结构变化, 因此设置$\kappa = -1$. 参数ν设置为$\nu = 0.3$.

5.2 实验结果与分析(Experimental results and analysis)

为了展示AIOSVR算法的有效性, 在训练样本个数分别为10, 20, 30, 40和50的情况下, 经过200次实验, 统计得到的特殊情况1和特殊情况2的次数以及矛盾1和矛盾2的出现次数如表3所示.

从表3中可以看出, 出现特殊情况1的概率要高于出现特殊情况2的概率. 虽然出现特殊情况2的概率极低, 但是并不能完全排除其出现的可能性. 此外, 从表3中还可以看出, 虽然出现矛盾1和矛盾2的概率较小, 但是也不能完全排除它们出现的可能性. 因此, 如何处理两种特殊情况和两种矛盾是非常重要的, 幸运的是AIOSVR算法能够处理两种特殊情况并有效地避免两种矛盾的出现.

表 3 出现特殊情况1、特殊情况2、矛盾1和矛盾2的次数

Table 3 The number of occurrences of special case 1, special case 2, contradiction 1 and contradiction 2

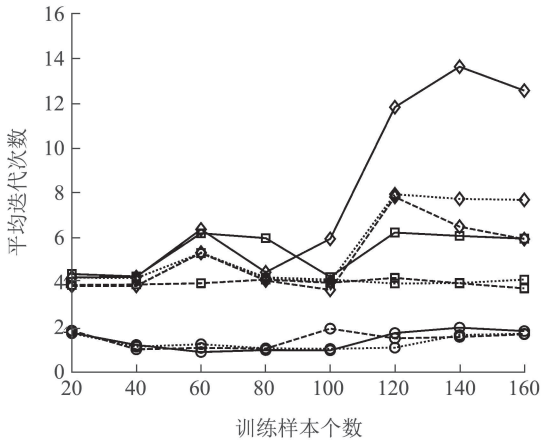
数据集	大小	特殊情况1			特殊情况2			矛盾1			矛盾2		
		线性核	多项式核	径向基核	线性核	多项式核	径向基核	线性核	多项式核	径向基核	线性核	多项式核	径向基核
Triazines	10	1	0	0	0	0	5	0	0	0	1	0	0
	20	0	3	0	0	0	0	0	6	0	0	0	0
	30	2	10	0	0	0	0	4	13	0	0	5	0
	40	0	0	0	0	0	0	0	0	0	0	1	0
	50	0	0	0	0	0	0	0	1	0	0	0	0
Housing	10	0	0	0	1	0	1	0	0	2	0	0	0
	20	0	0	0	0	0	0	0	2	5	5	0	2
	30	0	3	25	0	0	0	0	1	37	0	2	11
	40	0	0	2	0	0	0	0	0	5	0	0	3
	50	0	0	0	0	0	0	0	0	0	0	1	0
Cpusmall	10	0	0	0	0	0	1	3	5	3	1	1	1
	20	1	2	0	0	0	1	4	5	13	1	1	3
	30	5	19	44	0	0	0	11	24	57	2	4	13
	40	0	3	5	0	0	0	6	6	15	1	2	8
	50	0	2	0	0	0	0	8	5	3	1	1	2
Cadata	10	0	0	0	0	9	4	0	0	31	0	0	12
	20	0	1	4	0	2	0	0	0	56	0	0	13
	30	1	0	74	0	0	0	2	0	67	0	0	18
	40	1	3	7	0	0	0	0	0	43	0	0	20
	50	15	0	1	0	0	0	1	0	42	3	0	0

为了对AIOSVR算法的有限收敛性进行经验分析, 经过20次实验后, 统计了PA, RAIA和ARA的平均迭代次数, 结果如图2所示.

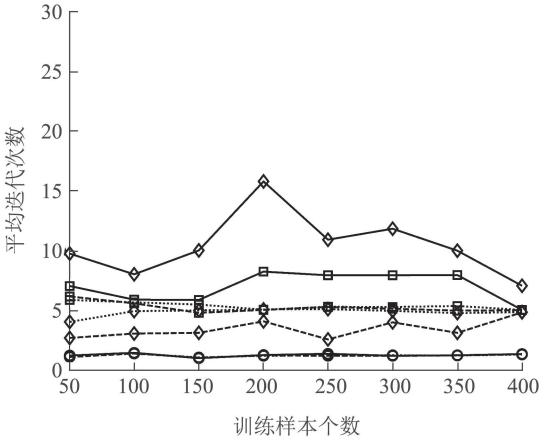
从图2中可以看出, AIOSVR算法在不同类型核函数和不同基准测试数据集上均具有快速的有限收敛性, 尤其是PA. 结合表3可以看出, AIOSVR算法能够有效地避免在绝缘增量调整过程中可能出现的解路径不可行更新问题, 而且经过有限次数调整能够以较快的速度收敛到式(5)的最优解.

为了体现AIOSVR算法在计算效率上的优越性, 本文与批处理热启动ν-SVR算法进行了比较, 对两种算法的训练样本规模与运行时间之间的关系进行经验分析, 结果如图3所示.

从图3中可以看出不管采用哪种类型的核函数, AIOSVR算法的学习时间总是要比批处理热启动ν-SVR算法的要短, 而且对于大规模数据集, 样本越多, 优势越明显, 原因是AIOSVR算法无需全部从头开始重新进行学习.



(a) Triazines



(b) Housing

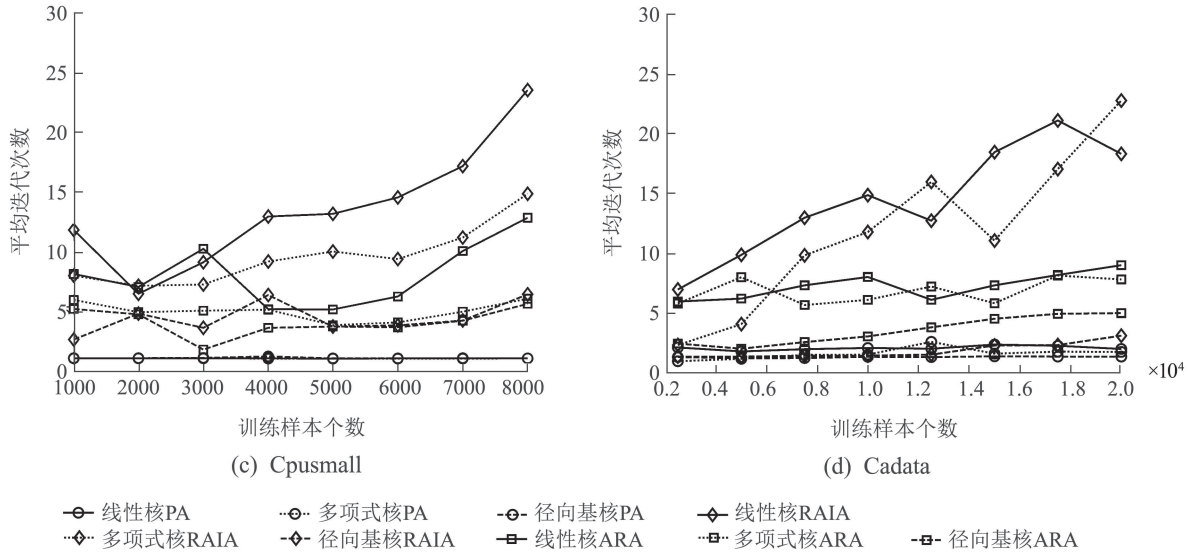


图 2 PA, RAIA和ARA的平均迭代次数

Fig. 2 Average numbers of iterations of PA, RAIA and ARA

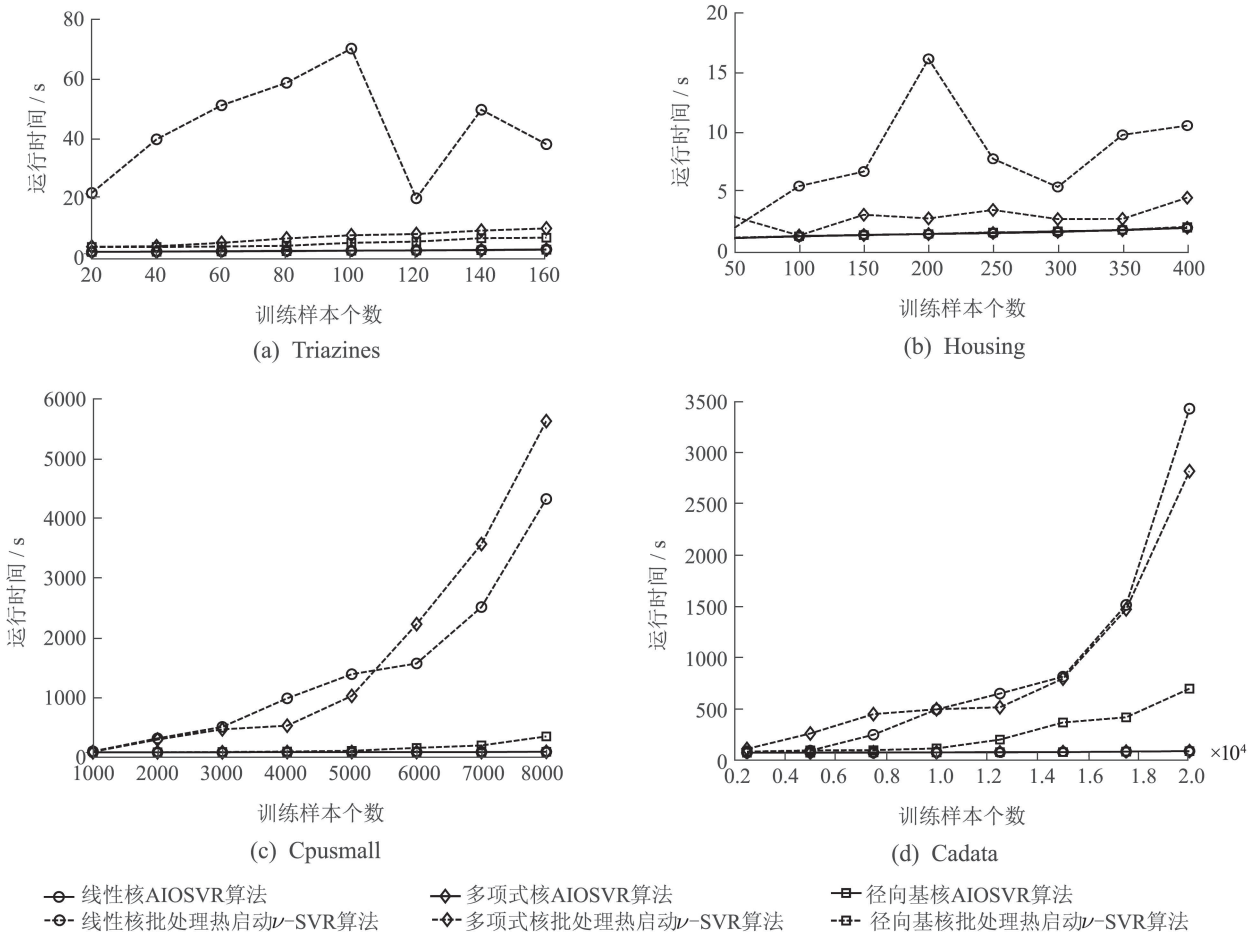


图 3 AIOSVR算法和批处理热启动ν-SVR算法的运行时间

Fig. 3 Run-time of AIOSVR and warm start batch ν-SVR algorithm

6 结论与展望(Conclusion and prospect)

针对ν-SVR的精确增量式在线学习问题, 提出了AIOSVR算法并对其进行了详细的理论分析, 解决了ν-SVR对偶问题存在的两个难题, 无法产生有效初始解的问题以及解路径的不可行更新问题. 算法的核心是PA, RAIA和ARA, 三者的可行性分析保

证了算法的每一步调整都是可靠的; 有限收敛性分析保证了算法经过有限次数调整后将会收敛到最小化问题的最优解. 与批处理热启动ν-SVR算法相比, 随着训练样本的增加, AIOSVR算法在缩短学习时间上具有明显优势, 因此该算法适合于在线学习场合.

AIOSVR算法及其可行性和有限收敛性分析能够推广到其他在线学习算法, 例如: 有序支持向量回归机[17]和SVMs的解路径算法[18–19]等.

然而, 为了证明本文的结论需要假设1作为前提条件. 此外, 在最差的情况下, SVMs解路径算法的复杂度随着训练样本个数呈指数级数增长[20]. 因此, 在进一步的工作中, 希望能够解决AIOSVR算法的收敛速度与训练样本个数之间的关系, 并研究精确减量式在线ν-SVR学习算法.

参考文献(References):

[1] VAPNIK V N. *The Nature of Statistical Learning Theory* [M]. New York: Springer, 1999: 3 – 30.

[2] LASKOV P, GEHL C, KRÜGER S, et al. Incremental support vector learning: analysis, implementation and applications [J]. *Journal of Machine Learning Research*, 2006, 7(3): 1909 – 1936.

[3] CAUWENBERGHS G, POGGIO T. Incremental and decremental support vector machine learning [C] //*Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press, 2001: 409 – 415.

[4] MARTIN M. On-line support vector machines for function approximation [C] //*Proceedings of the 13th European Conference on Machine Learning*. London, UK: Springer-Verlag, 2002: 282 – 294.

[5] MA J, THEILER J, PERKINS S. Accurate on-line support vector regression [J]. *Neural Computation*, 2003, 15(11): 2683 – 2703.

[6] KARASUYAMA M, TAKEUCHI. Multiple incremental decremental learning of support vector machines [J]. *IEEE Transactions on Neural Networks*, 2010, 21(7): 1048 – 1059.

[7] GU B, WANG J D, YU Y C, et al. Accurate on-line ν-support vector learning [J]. *Neural Networks*, 2012, 27(3): 51 – 59.

[8] GU B, SHENG V S. Feasibility and finite convergence analysis for accurate on-line ν-support vector machine [J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2013, 24(8): 1304 – 1315.

[9] GU Bin, WANG Jiandong. Effective ν-path algorithm for ν-support vector regression [J]. *Journal of Software*, 2012, 23(10): 2643 – 2654. (顾彬, 王建东. 有效的ν支持向量回归机的ν解路径算法 [J]. 软件学报, 2012, 23(10): 2643 – 2654.)

[10] GU Bin, ZHENG Guansheng, WANG Jiandong. Analysis for incremental and decremental standard support vector machine [J]. *Journal of Software*, 2013, 24(7): 1601 – 1613.
(顾彬, 郑关胜, 王建东. 增量和减量式标准支持向量机的分析 [J]. 软件学报, 2013, 24(7): 1601 – 1613.)

[11] DENG Naiyang, TIAN Yingjie. *Support Vector Machines: a New Method in Data Mining* [M]. Beijing: Science Press, 2004: 254 – 259.
(邓乃扬, 田英杰. 数据挖掘中的新方法——支持向量机 [M]. 北京: 科学出版社, 2004: 254 – 259.)

[12] SCHÖLKOPF B, SMOLA A J, WILLIAMSON R C, et al. New support vector algorithms [J]. *Neural Computation*, 2000, 12(5): 1207 – 1245.

[13] CHANG C C, LIN C J. Training ν-support vector regression: theory and algorithms [J]. *Neural Computation*, 2002, 14(8): 1959 – 1977.

[14] BERTSEKAS D P. *Convex Optimization Theory* [M]. Beijing: Tsinghua University Press, 2011: 167 – 181.

[15] GOLUB G H, VAN LOAN C F. *Matrix Computations* [M]. London: John Hopkins University Press, 1996: 174 – 182.

[16] CHEN P H, LIN C J, SCHÖLKOPF B. A tutorial on ν-support vector machines [J]. *Applied Stochastic Models in Business and Industry*, 2005, 21(2): 111 – 136.

[17] CHU W, KEERTHI S S. Support vector ordinal regression [J]. *Neural Computation*, 2007, 19(3): 792 – 815.

[18] HASTIE T, ROSSET S, TIBSHIRANI, et al. The entire regularization path for the support vector machine [J]. *Journal of Machine Learning Research*, 2004, 5(4): 1391 – 1415.

[19] GUNTER L, ZHU J. Computing the solution path for the regularized support vector regression [C] //*Advances in Neural Information Processing Systems 18*. Cambridge, MA: MIT Press, 2005: 483 – 490.

[20] GÄRTNER B, JAGGI M, MARIA C. An exponential lower bound on the complexity of regularization paths [J]. *Journal of Computational Geometry*, 2012, 3(1): 168 – 195.

作者简介:

顾斌杰 (1980–), 男, 讲师, 博士, 目前研究方向为机器学习、工业过程建模, E-mail: gubinjie1980@126.com;

潘 丰 (1963–), 男, 教授, 博士生导师, 目前研究方向为工业过程建模及优化控制, E-mail: pan_feng_63@163.com.