

【工程与技术研究】

基于张量数据二分类问题的 RT-PCF 算法

许娜¹, 王敏²

(1. 曲阜师范大学 公共外语教学部; 2. 曲阜师范大学 管理学院, 山东 日照 276800)

摘要:考虑将多面体圆锥函数(PCF)同张量数据结合,对原有 PCF 算法进行改进,构建一类解决张量数据二分类问题的 RT-PCF 算法。在目标函数中引入正则化项,并对约束数据集进行松弛,解决过拟合问题。最后,通过实际数据验证算法的有效性和实用性。

关键词:二分类;多面体圆锥函数;张量数据

中图分类号: O 221.2 **文献标识码:** A **DOI:**10.13486/j.cnki.1673-2618.2022.06.011

监督分类是机器学习、数据挖掘的重要任务之一,许多现实问题可被定义为监督分类任务,如目标检测、癌症诊断、文档索引、图像识别、医学诊断等^[1-3]。监督分类问题可通过优化方法简化为数学规划模型,对模型的决策变量或参数进行优化可得到一个分离函数^[4],将输入数据映射到一组标签。迄今为止,多数优化方法采用了凸规划模型,与非凸模型相比^[5],凸模型容易求解。然而,具有特殊结构的大数据集,即使是凸模型,要获得最优解决方案也较为困难。因此,研究者致力于开发更加有效、准确的优化方法。支持向量机(svm)是解决分类问题的高效优化算法之一,它通过训练输入的向量数据产生分离函数,并对未知数据进行预测,进而获得分类结果。

张量是多维数据表达的有效工具,比如灰度图片可以用二阶张量(矩阵)表示,彩色图片用三阶张量表示。当应用支持向量机方法处理高阶张量数据时,往往将张量数据转化成向量数据,再对向量数据进行分类研究,最终再恢复成张量数据。研究表明,这种简单直接的转化不仅忽略了数据的原始结构及其内在关联^[6],还可能带来维数灾难问题,计算时间大大增加,分类效果也大打折扣。为了解决此类问题,基于高阶张量数据的分类问题开始被广泛研究^[7-8]。张量数据作为输入数据,不仅可以保持数据本身结构信息,还可以有效避免过拟合现象。

多面体圆锥函数(PCF)最早由 Gasimov 和 Ozturk 定义,文献[9]提出了一种名为 PCF 算法的二分类问题的迭代方法。PCF 算法主要解决源于实际问题的线性不可分向量数据分类问题。该算法的主要思想是通过求解有限个线性规划(LP)问题将给定的两个向量数据集分离。核函数方法是解决线性不可分问题的主要方法,即通过核函数将样本数据映射到高维空间中,使样本数据在高维空间中是线性可分的,然后再使用常规线性分类器进行数据分类。与核函数^[10-11]方法的最大区别在于 PCF 算法不必将样本数据映射到高维空间^[12]。笔者拟对原有 PCF 算法进行改进,并将其与张量数据相结合,构建一类解决张量数据二分类问题的 RT-PCF 算法。

收稿日期:2022-05-01

基金项目:国家自然科学基金项目(12071249);山东省自然科学基金杰出青年基金项目(ZR2021JQ01)

第一作者简介:许娜(1981—),女,山东日照人,助教,主要从事张量大数据分析 with 优化研究。

E-mail: promise202@126.com

1 预备知识

(1) 二分类问题。假设已知训练数据样本集 $T = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_l, y_l)\} \in (X \times Y)^l$, 其中, 输入数据 $\mathbf{x}_i \in X \subseteq \mathbf{R}^n, y_i \in Y = \{+1, -1\}, i = 1, \dots, l$ 代表输出标签。传统二分类问题实质就是找到一个可以将 \mathbf{R}^n 上的点分成两部分的标准。

(2) PCF 函数及算法。文献[9]首先提出了 PCF 函数的概念并将其定义为

$$g(\mathbf{w}, \boldsymbol{\xi}, \gamma, c)(x) = \langle \mathbf{w}, (\mathbf{x} - c) \rangle + \boldsymbol{\xi} \| \mathbf{x} - c \|_1 - \gamma, \tag{1}$$

其中 $\mathbf{w}, c \in \mathbf{R}^n, \boldsymbol{\xi}, \gamma \in \mathbf{R}, \| \mathbf{x} \|_1 = |x_1| + |x_2| + \dots + |x_n|$ 为 $\mathbf{x} \in \mathbf{R}^n$ 的 l_1 范数。文献[9]已经证明, 对于公式(1)给定的函数, 其图像是一个顶点为 $(c, -\gamma) \in \mathbf{R}^n \times \mathbf{R}$ 的多面体锥, 并且该函数的下水平集是凸多面体。PCF 函数可以将整个空间分成两个部分, 即内部区域和外部区域。设 A 和 B 是 \mathbf{R}^n 中的两个集合:

$$A = \{\mathbf{a}^1, \mathbf{a}^2, \dots, \mathbf{a}^m\}, \mathbf{a}^i \in \mathbf{R}^n, i \in I^n = \{1, 2, \dots, m\}; B = \{\mathbf{b}^1, \mathbf{b}^2, \dots, \mathbf{b}^p\}, \mathbf{b}^j \in \mathbf{R}^n, j \in J = \{1, 2, \dots, p\}.$$

假设 $\mathbf{a}^i, \mathbf{b}^j$ 对应的 PCF 函数值为 $g_i(\mathbf{a}^i)$ 和 $g_j(\mathbf{b}^j)$, 如果满足

$$\min\{g_1(\mathbf{a}^1), \dots, g_m(\mathbf{a}^m)\} \leq 0, i = 1, \dots, m; \min\{g_1(\mathbf{b}^1), \dots, g_p(\mathbf{b}^p)\} > 0, j = 1, \dots, p, \tag{2}$$

则 A 为正类, B 为负类。若存在有限个 PCF 函数满足式(2), 则称 A 和 B 为多面体圆锥可分离的。对于有限点集 A 和 B , 现有 PCF 算法步骤如下^[9]: (I) 设置初始条件 $k=1, I_k=I, A_k=A$ 。(II) 令 \mathbf{a}^k 是 A_k 中的任意一点, 求解子问题 (P_k)

$$\begin{aligned} \min_{\mathbf{w}, \boldsymbol{\xi}, \gamma} \frac{1}{m} \sum_{i=1}^m y_i, \text{ s. t. } & \langle \mathbf{w}, (\mathbf{a}^i - \mathbf{a}^k) \rangle + \boldsymbol{\xi} \| \mathbf{a}^i - \mathbf{a}^k \|_1 - \gamma + 1 \leq y_i, \forall i \in I_k, \\ & - \langle \mathbf{w}, (\mathbf{b}^j - \mathbf{a}^k) \rangle - \boldsymbol{\xi} \| \mathbf{b}^j - \mathbf{a}^k \|_1 + \gamma + 1 \leq 0, \forall j \in J, \boldsymbol{\xi} \geq 0, \gamma \geq 1, y_i \geq 0, \forall i \in I_k, \end{aligned}$$

设 $\omega_k, \xi_k, \gamma_k$ 是问题 (P_k) 的解。(III) 令 $g_k(\mathbf{x}) = g_{(\omega_k, \xi_k, \gamma_k, \mathbf{a}^k)}(\mathbf{x})$, 以及

$$I_{k+1} = \{i \in I_k : g_k(\mathbf{a}^i) + 1 > 0\}, A_{k+1} = \{\mathbf{a}^k \in A_k : k \in I_{k+1}\}, k = k + 1.$$

如果 $A_k \neq \emptyset$, 则返回步骤 (II)。(IV) 定义函数 $g(\mathbf{x})$ 分离集合 A 和 $B: g(\mathbf{x}) = \min_k g_k(\mathbf{x})$ 。算法终止。

PCF 算法的主要思想是从数据集 A 中随机选择一个 $\mathbf{a}^k \in A$ 作为中心, 并通过求解子问题 (P_k) , 获得 $\omega_k, \xi_k, \gamma_k$, 构成函数 $g_k(\mathbf{x})$ 。该函数的下水平集将整个空间分成两部分, 使集合 A 中尽可能多的数据点保持在下水平集的“内部”, 同时使集合 B 中所有的数据点保持在下水平集“外部”。接下来, 从集合 A 中去除正确分类的数据点, 算法进行下一次迭代, 并为修改后的集合生成一个新的函数 $g_k(\mathbf{x})$ 。循环上述过程, 直到集合 A 成为空集时停止。算法保存迭代过程中生成的所有函数 $g_k(\mathbf{x})$, 最终的分隔函数 $g(\mathbf{x})$ 由 $g_k(\mathbf{x})$ 的最小值组成。文献[9]已证明了该算法在有限步骤内终止。

(3) 矩阵和张量的范数与内积等概念。张量是矩阵高阶推广。假设 $\mathbf{A} = (a_{ij}) \in \mathbf{R}^{m \times n}$ 为矩阵, $\mathbf{B} = (b_{i_1 i_2 \dots i_m}) \in \mathbf{R}^{n_1 \times \dots \times n_m}$ 为 m 阶张量。矩阵与张量的 l_1 范数定义为

$$\| \mathbf{A} \|_1 = \sum_{i=1}^m \sum_{j=1}^n | a_{ij} |, \| \mathbf{B} \|_1 = \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \dots \sum_{i_m=1}^{n_m} | b_{i_1 i_2 \dots i_m} |.$$

另一个有用的范数是矩阵与张量的 Frobenius 范数, 定义如下:

$$\| \mathbf{A} \|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n | a_{ij} |^2}, \| \mathbf{B} \|_F = \sqrt{\sum_{i_1=1}^{n_1} \dots \sum_{i_m=1}^{n_m} | b_{i_1 i_2 \dots i_m} |^2}.$$

假设 $\mathbf{A}, \bar{\mathbf{A}} \in \mathbf{R}^{m \times n}$ 为矩阵, 则其内积定义为 $\langle \mathbf{A}, \bar{\mathbf{A}} \rangle = \sum_{i=1}^m \sum_{j=1}^n a_{ij} \bar{a}_{ij}$ 。作为矩阵内积的推广, 假设 $\mathbf{B}, \bar{\mathbf{B}} \in \mathbf{R}^{n_1 \times \dots \times n_m}$ 为两个张量, 则其积定义为

$$\langle \mathbf{B}, \bar{\mathbf{B}} \rangle = \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \dots \sum_{i_m=1}^{n_m} b_{i_1 i_2 \dots i_m} \bar{b}_{i_1 i_2 \dots i_m}.$$

2 基于矩阵数据的 RT-PCF 算法

原始 PCF 算法易出现两方面不足:一是存在过拟合现象,二是因随机选择中心而导致 PCF 函数的计算复杂性无法确定。过拟合问题的主要原因是,通过最小化训练误差获得的分离函数在未知数据下可能表现不佳,使训练误差最小化的复杂分离函数并不一定能保证使测试误差达到最小。

为避免过拟合现象,经常采用最小化训练误差和正则化目标函数的方法。该技术通过降低分离函数的复杂性来最小化训练误差和测试误差的差距。因此,第一步就是在目标函数加入正则化项。原始问题中的目标函数只能使训练误差最小化,基于此,对于矩阵数据集,选择 F 范数的平方进行复杂度逼近,并利用其

可微性和凸的有利性质。故问题(P_k)的目标函数可改为 $\min_{W, \xi, \gamma} \|W\|_F^2 + \xi^2 + \gamma^2 + \sum_{i=1}^m y_i$ 。

另一个原因与问题(P_k)中的第二个约束条件有关。在每次迭代中,该约束不允许集合 B 出现任何错误分类的点,当集合 A 中的所有点都被正确分类时,算法就会终止。因此,该算法最终得到了一个训练精度为 100% 的分离函数。这也说明,仅仅完善目标函数并不足以处理过拟合问题。可通过添加非负松弛变量 z_j 来放松对集合 B 的约束,允许集合 B 存在错误分类的数据点。由于需要 z_j 是正的,因此将问题(P_k)的目

标函数改进为 $\min_{W, \xi, \gamma} \|W\|_F^2 + \xi^2 + \gamma^2 + \frac{1}{m} \sum_{i=1}^m y_i + \frac{1}{p} \sum_{j=1}^p z_j$, 相应的第二个约束条件可改为

$$-\langle W, (B^j - A^k) \rangle - \xi \|B^j - A^k\|_1 + \gamma + 1 \leq z_j, \forall j \in J.$$

PCF 算法存在的另一个问题是中心随机选择会影响所产生 PCF 的复杂性,进而影响训练时间。基于此,为打造更好的分类器,产生性能更好的 PCF,在目标函数中引入了一个标量 $\lambda > 0$,称为正则化参数。这种改进可以降低生成的 PCF 函数的复杂性,同时保持模型的可凸性。则目标函数被改进为

$$\min_{W, \xi, \gamma} \lambda (\|W\|_F^2 + \xi^2 + \gamma^2) + \frac{1}{m} \sum_{i=1}^m y_i + \frac{1}{p} \sum_{j=1}^p z_j. \quad (3)$$

根据上述改进,对于矩阵数据集

$$A = \{A^1, \dots, A^l\}, A^i \in \mathbf{R}^{n \times m}, i \in I = \{1, \dots, l\},$$

$$B = \{B^1, \dots, B^p\}, B^j \in \mathbf{R}^{n \times m}, j \in J = \{1, \dots, p\}$$

可以通过求解线性约束的二次凸子问题(Q_k)

$$\begin{aligned} & \min_{W, \xi, \gamma} \lambda (\|W\|_F^2 + \xi^2 + \gamma^2) + \sum_{i=1}^m y_i + c \sum_{j=1}^p z_j, \\ \text{s. t. } & \langle W, (A^i - A^k) \rangle - \xi \|A^i - A^k\|_1 - \gamma + 1 \leq y_i, \forall i \in I_k, \\ & -\langle W, (B^j - A^k) \rangle - \xi \|B^j - A^k\|_1 + \gamma + 1 \leq z_j, \forall j \in J_k, \\ & y_i \geq 0 (\forall i \in I_k), z_j \geq 0 (\forall j \in J_k), \xi \geq 0, \gamma \geq 1, \end{aligned} \quad (4)$$

来获得分离函数。其中, $W \in \mathbf{R}^{n \times m}, c, \lambda > 0$ 。在每次迭代中,RT-PCF 算法以 c 和 λ 作为参数输入,然后随机选择中心 A^k 并求解问题(Q_k),得到相对应的 W_k, ξ_k, r_k ,从而生成 $g_k(\mathbf{X}) = g_{(W_k, \xi_k, r_k, A^k)}(\mathbf{X})$ 。在迭代过程中,保持对集合 A 和 B 的更新,从集合 A 中删除正确分类的点,从集合 B 中删除错误分类的点。直到给定的集合 A 或 B 成为空集时,算法终止。

另外,目标函数(3)中相关项的权重系数为 $1/m$ 和 $1/p$,而目标函数(4)中相应的权重发生了变化。原因如下:在子问题中,参数 c 只惩罚由集合 B 产生的错误,因为集合 A 的分类错误取决于迭代中产生的 PCF 函数。设有数据集 $A = A_1 \cup A_2$ 和 B 可用至少两个 PCF 来分离。比如 $A_1 = \{A^i : g_1(A^i) \leq 0, g_2(A^i) > 0, i = 1, \dots, l\}, A_2 = \{A^i : g_1(A^i) > 0, g_2(A^i) \leq 0, i = l + 1, \dots, m\}, B = \{B^j : g_1(B^j) > 0, g_2(B^j) > 0, j = 1, \dots, p\}$ 。然后,假设 g_1 和 g_2 分别是第一次和第二次迭代中生成的 PCF。显然,在第一次迭代时, g_1 对 A_2 中的点分类错,而这个错误并不会导致最终训练结果的错误,因为 g_2 在第二次迭代中会正确地分类这些点。故只通过取 $1/m = 1$ 和 $c = 1/p$ 来控制集合 B 相对于集合 A 的相对训练误差即可。

作为训练阶段的结果, 算法保存生成的所有函数, 其中, 由最小的 $g(\mathbf{X}) = \min\{g_1(\mathbf{X}), \dots, g_k(\mathbf{X})\}$ 作为最终的分离函数来分开给定的数据集。在测试阶段, 对于未知类别的数据 \mathbf{X} , 可使用 RT-PCF 算法生成的 $g(\mathbf{X})$ 判断其类别。如果 $g(\mathbf{X}) \leq 0$, 则 \mathbf{X} 属于 A , 否则属于 B 。

3 基于高阶张量数据的 RT-PCF 算法

设两类训练集

$A = \{\mathbf{A}^1, \dots, \mathbf{A}^l\}, \mathbf{A}^i \in \mathbf{R}^{n_1 \times \dots \times n_m}, i \in I = \{1, \dots, l\}, B = \{\mathbf{B}^1, \dots, \mathbf{B}^p\}, \mathbf{B}^j \in \mathbf{R}^{n_1 \times \dots \times n_m}, j \in J = \{1, \dots, p\}$, 通过求解问题 (Q_k)

$$\begin{aligned} \min_{\mathbf{W}, \xi, \gamma} \quad & \lambda (\|\mathbf{W}\|_F^2 + \xi^2 + \gamma^2) + \sum_{i=1}^m y_i + c \sum_{j=1}^p z_j \\ \text{s. t.} \quad & \langle \mathbf{W}, (\mathbf{A}^i - \mathbf{A}^k) \rangle - \xi \|\mathbf{A}^i - \mathbf{A}^k\|_1 - \gamma + 1 \leq y_i (\forall i \in I_k), \\ & -\langle \mathbf{W}, (\mathbf{B}^j - \mathbf{A}^k) \rangle - \xi \|\mathbf{B}^j - \mathbf{A}^k\|_1 + \gamma + 1 \leq z_j (\forall j \in J_k), \\ & y_i \geq 0 (\forall i \in I_k), z_j \geq 0 (\forall j \in J_k), \xi \geq 0, \gamma \geq 1, \end{aligned}$$

获得分离函数。其中, $\mathbf{W} \in \mathbf{R}^{n_1 \times n_2 \times \dots \times n_m}, c, \lambda > 0$ 。当测试数据 \mathbf{T} 时, 其分离函数为 $g(\mathbf{T}) = \min\{g_1(\mathbf{T}), \dots, g_k(\mathbf{T})\}$ 。

RT-PCF 算法具体步骤如下: 输入 $A = \{\mathbf{A}^1, \dots, \mathbf{A}^l\}, \mathbf{A}^i \in \mathbf{R}^{n_1 \times \dots \times n_m}, i \in I = \{1, \dots, l\}, B = \{\mathbf{B}^1, \dots, \mathbf{B}^p\}, \mathbf{B}^j \in \mathbf{R}^{n_1 \times \dots \times n_m}, j \in J = \{1, \dots, p\}$ 。(I) 令 $k=1, I_k = I, J_k = J$ 。(II) 从 I_k 中随机选择一个指标 e , 令 $\mathbf{A}^k = \mathbf{A}^e$ 。(III) 求解问题 (Q_k) 得到 $\mathbf{W}_k, \xi_k, \gamma_k$ 。(IV) 构造函数 $g_k = g_{(\mathbf{W}_k, \xi_k, \gamma_k, \mathbf{A}^k)}(\boldsymbol{\chi})$, 并保存。(V) 如果 $I_k = \emptyset$ 或者 $J_k = \emptyset$, 进入下一步骤; 若否, 更新指标集, 令 $I_{k+1} = \{i \in I_k : g_k(\mathbf{A}^i) > 0\}, J_{k+1} = \{j \in J_k : g_k(\mathbf{B}^j) > 0\}$, 令 $k=k+1$, 返回步骤 (II)。(VI) 获得分离函数 $g(\boldsymbol{\chi}) = \min\{g_1(\boldsymbol{\chi}), \dots, g_k(\boldsymbol{\chi})\}$, 算法终止。

4 数值实验

通过源于实际数据库的训练数据对 RT-PCF 算法的有效性 with 准确性进行测试。选取了 USPS 手写体数字识别库和 Kaggle 的猫狗彩色图片数据集。其中, USPS 手写体数据库中由数字 0~9 的手写数字图片构成, 该库共有 11 000 张图片, 每个数字对应 1100 张图片, 每张灰度图片的大小均为 16×16 矩阵。Kaggle 猫狗数据集共包含训练集 25 000 张大小不一的猫狗图片, 测试集 12 500 张不同的猫狗图片。实验时, 在训练集中选取猫和狗各 200 张照片进行训练, 在测试集中随机选取猫狗各 1000 张照片进行测试, 并不影响图片清晰度的前提下将照片大小统一为 $128 \times 128 \times 3$ 。

对于 USPS 手写体数据库(见表 1), 将其分成 9 个二分类问题, 前后相邻数字依次成为一组, 比如 0 和 1 为第一个二分类问题, 1 和 2 为第二个二分类问题, 依次组合, 直至 8 和 9 组成最后一个二分类问题。在每一个二分类问题中, 随机抽取 100 张图片为训练集, 剩下的 1000 张照片为测试集。参数 c 和 λ 选取范围为 $\{10^{-3}, 10^{-2}, 10^{-1}, 10^0, 10^1, 10^2, 10^3\}$ 。9 个二分类问题的测试精度 $p = (t_p + t_n)/n$, 其中 n 为全体测试样本个数, 其值见表 1, 并将其与经典的支持张量机交替投影迭代算法(ALP-STM)^[13] 测试精度进行比较。在 9 个二分类的问题中, RT-PCF 算法有 6 个优于 ALP-STM 算法。

表 1 不同算法下手写体数字识别数据库 p 的平均结果

训练集	ALP-STM	RT-PCF	训练集	ALP-STM	RT-PCF	训练集	ALP-STM	RT-PCF
0-1	0.8962	0.9133	3-4	0.9219	0.9163	6-7	0.9471	0.9237
1-2	0.8781	0.8996	4-5	0.9346	0.9096	7-8	0.8955	0.9282
2-3	0.8284	0.9315	5-6	0.8696	0.9202	8-9	0.8522	0.9103

为了探究参数 c 和 λ 对 p 的影响, 以猫狗数据集为例进行研究。令 $\lambda=1$, 参数 c 取 $\{10^{-3}, 10^{-2}, 10^{-1},$

$10^0, 10^1, 10^2, 10^3$ }, 结果见图 1。可以直观地看出, 当 c 增加时, p 也会有所增加。但是, 在 c 的取值大于 10 之后, p 并没有在进行明显的提升。原因在于当 c 足够大, 大到主导问题(Q_k)时, 其对集合 B 的约束就会变成强约束, 导致出现过拟合问题。令参数 $c=1$, 参数 λ 取 $\{10^{-3}, 10^{-2}, 10^{-1}, 10^0, 10^1, 10^2, 10^3\}$, 结果见图 2, 可以看出 λ 对 p 会产生负面影响。

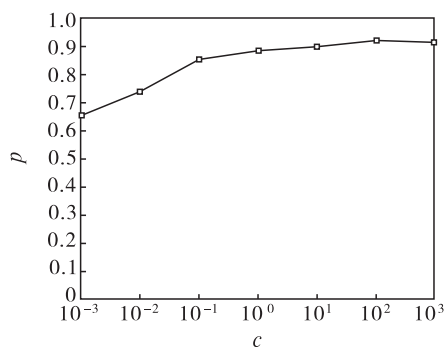


图 1 参数 c 对 p 的影响

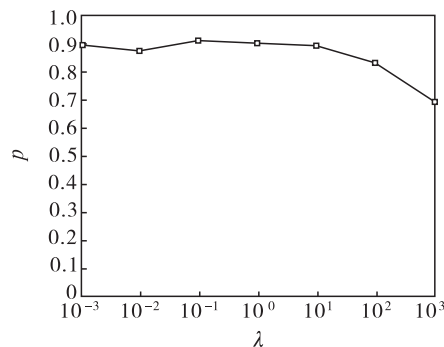


图 2 参数 λ 对 p 的影响

接下来, 探究参数 c 和 λ 对需要求解子问题个数产生的影响。以猫狗数据集和 USPS 中第 5 个二分类问题为例。令 λ 取固定值 10, c 取 $\{10^{-3}, 10^{-2}, 10^{-1}, 10^0, 10^1, 10^2, 10^3\}$, 其结果展示于图 3。能够明显看出, 当 λ 取固定值时, 参数 c 对于需要求解子问题的个数并没有明显影响。令 c 取固定值 10, λ 取 $\{10^{-3}, 10^{-2}, 10^{-1}, 10^0, 10^1, 10^2, 10^3\}$, 其结果展示于图 4。可以发现参数 c 固定后, λ 的不同取值对需要求解子问题的个数有负面影响。当 λ 逐渐增大时, 需要求解的问题越来越少。究其原因当 λ 足够大, 大到主导问题(Q_k)时, RT-PCF 算法倾向于求解更少的子问题。

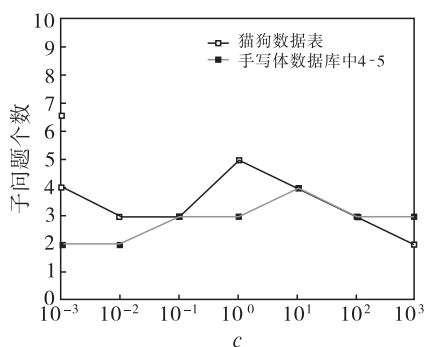


图 3 参数 c 对子问题个数影响

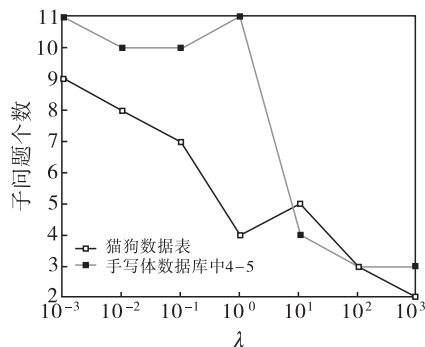


图 4 参数 λ 对子问题个数影响

5 结论

对传统 PCF 算法进行分析, 并在目标函数中加入正则项, 放宽对集合 B 的约束条件与加入正则化参数 λ , 改进了原有 PCF 算法, 提出了基于张量数据二分类问题的 RT-PCF 算法。通过对实际数据集进行训练测试, 并与现有算法进行比较, 验证了 RT-PCF 算法的有效性。为保证测试精度, 尽可能减少子问题的求解次数, 后续将进一步研究如何找到一个带自适应性的最佳参数 λ 和 c , 并将两个参数调整到合适的比例。

参 考 文 献:

[1] WANG H, HUANG G. Application of support vector machine in cancer diagnosis[J]. Medical oncology, 2011, 28(1): 613 - 618.
 [2] GU Z, ZHANG Z, SUN J, et al. Robust image recognition by L1-norm twin-projection support vec-

- tor machine[J]. Neurocomputing, 2017, 223: 1 - 11.
- [3] CHIANG H M, WANG T Y. One-against-one fuzzy support vector machine text categorization classifier[C]. 2008 IEEE International Conference on Industrial Engineering and Engineering Management, 2008: 1519 - 1523.
- [4] BENNETT K P, PARRADO-HERNANDEZ E. The interplay of optimization and machine learning research[J]. Journal of machine learning research, 2006, 7: 1265 - 1281.
- [5] CARRIZOSA E, MORALES D R. Supervised classification and mathematical optimization[J]. Computers operations research, 2013, 40(1): 150 - 165.
- [6] GLASSER I, PANCOTTI N, CIRAC J I. From probabilistic graphical models to generalized tensor networks for supervised learning[J]. IEEE access, 2020, 8: 68169 - 68182.
- [7] HUGGINS W, PATIL P, MITCHELL B, et al. Towards quantum machine learning with tensor networks[J]. Quantum science and technology, 2019, 4(2): 024001.
- [8] CHEN Y, ZHONG P. Linear one-class support tensor machine[J]. International journal of signal processing, image processing and pattern recognition, 2016, 9(9): 379 - 388.
- [9] GASIMOV R N, OZTURK G. Separation via polyhedral conic functions[J]. Optimization methods and software, 2006, 21(4): 527 - 540.
- [10] GÖNEN M, ALPAYDIN E. Multiple kernel learning algorithms[J]. Journal of machine learning research, 2011, 12: 2211 - 2268.
- [11] HÖFMANN T, SCHLKOPF B, SMOLA A J. Kernel methods in machine learning[J]. The annals of statistics, 2008, 36(3): 1171 - 1220.
- [12] CIMEN E, OZTURK G, GEREK O N. Incremental conic functions algorithm for large scale classification problems[J]. Digital signal processing, 2018, 77: 187 - 194.
- [13] 杨兵. 基于张量数据的机器学习方法研究与应用[M]. 北京: 中国农业大学, 2014.

RT - PCF Algorithm Based on Tensor Data for Binary Classification Problem

XU Na¹, WANG Min²

(1. Department of College English Teaching, Qufu Normal University;

2. School of Management Science, Qufu Normal University, Rizhao 276800, China)

Abstract: Polyhedral conic function (PCF) and its related theory are analyzed to improve it. Then, a new RT - PCF algorithm is established for tensor data binary classification problem. A regularized term is added to the objective function and the constrained data set is relaxed to solve the over-fitting problem. Finally, the validity and practicability of the algorithm are verified by actual data sets.

Keywords: binary classification; polyhedral conic functions; tensor data

(责任编辑: 贾晶晶)