

基于 GRAM 矩阵的粒感知机

吴少华¹ 陈玉明²

¹ 厦门美亚亿安信息科技有限公司 福建 厦门 361008

² 厦门理工学院计算机与信息工程学院 福建 厦门 361024

(myeant2024@163.com)

摘要 感知机是一种简单的线性分类器,也是 SVM 及深度学习的基石。然而,大部分复杂问题是非线性模型,感知机在处理这类问题时,分类效果不佳。因此,引入粒计算理论,以参考样本为模板,将训练样本粒化为特征粒子及特征粒向量,进而定义粒 GRAM 矩阵,提出一种基于 GRAM 矩阵的粒感知机模型。该模型优化感知机的对偶形式,构造新的粒感知机模型。为处理非线性分类问题,引入核函数,构造基于粒向量的核 GRAM 矩阵,并给出 GRAM 粒感知机的损失函数和学习方法。最后,从收敛性、非线性处理能力、参考样本的数量以及模型分类效果 4 方面进行实验分析,结果表明了 GRAM 粒感知机的有效性与正确性。

关键词 粒计算;感知机;GRAM 矩阵;非线性分类;核函数

中图分类号 TP181

Granular Perception Machine Based on GRAM Matrix

WU Shaohua¹ and CHEN Yuming²

¹ Xiamen Meiya eAnt Information Technology Co., Ltd., Xiamen, Fujian 361008, China

² College of Computer and Information Engineering, Xiamen University of Technology, Xiamen, Fujian 361024, China

Abstract The perceptron is a simple linear classifier and is also the cornerstone of SVM and deep neural networks. However, most complex problems are often nonlinear, and the perceptron performs poorly in handling such issues. Therefore, this paper introduces granular computing theory, whereby training samples are granulated into feature granules and feature granular vectors based on reference samples. The paper defines a granular GRAM matrix and proposes a granular perceptron model based on the GRAM matrix. This model optimizes the dual form of the perceptron to construct a new granular perceptron model. To better handle nonlinear classification problems, a kernel function is introduced to construct a kernel GRAM matrix based on granular vectors, and the loss function and learning method of the GRAM granular perceptron are provided. Finally, experiments analyze the model's convergence, nonlinear processing capability, the number of reference samples, and the classification performance of models. The result shows the effectiveness and correctness of the model of GRAM granular perceptron.

Keywords Granular computing, Perceptron, GRAM matrix, Nonlinear classification, Kernel function

1 引言

1979 年,美国学者 Zadeh^[1] 发表论文“Fuzzy sets and information granularity”,认为信息粒子存在于现实世界多个领域,进而推动信息粒子成为研究热点^[2]。1996 年,香港学者 Lin^[3] 基于信息粒子与邻域系统,发表了一系列相关论文,并首次将粒化问题的研究称作粒计算。粒计算含义广泛,涵盖了粒的理论和研究方法等^[4]。通常,粒计算被认为是 Fuzzy 集和 Rough 集^[5-6] 等领域的合集,它的精髓在于如何构建信息粒,并在问题求解的过程中使用粒子^[7]。Yao 等^[8] 提出了邻域粗糙集,将粒计算应用于知识发现、机器学习和数据分析^[9] 等领域。Miao 等^[10] 将信息熵引入,对知识的粒计算进行研究。Hu 等^[11] 解析了邻域约简,提出邻域关系粒化方法。

Chen 等^[12-13] 在各种机器学习分类中应用邻域粒化方法^[14],并在卷积分类器中引入模糊粒化,取得了良好效果。

1957 年, Rosenblatt^[15] 提出了感知机模型。感知机作为机器学习领域第一个意义深远的算法,其结构简单,易于训练。针对二分类问题,感知机分类性能较好^[16]。但现实世界中的问题更多是非线性的^[17],而它作为一个面向二分类的线性分类器,并不能直接解决一些线性不可分^[18-19] 的问题。近年来,学者们对感知机不断进行优化,提出了用核技巧^[20-21] 的方式来优化感知机线性不可分的问题。感知机算法是机器学习和深度学习的基础性算法,通过改进升级,多层感知机也出现在大众视野里,用来处理更复杂的分类问题。然而,多层感知机存在较多的嵌套层数、极高的模型复杂度、较长的训练时间以及梯度消失问题。粒子的结构性较强,单特征粒化形

基金项目:福建省自然科学基金(2024J011192);厦门市自然科学基金(3502Z202473069)

This work was supported by the Natural Science Foundation of Fujian Province, China(2024J011192) and Natural Science Foundation of Ximen, China(3502Z202473069).

通信作者:陈玉明(myeant2024@163.com)

成粒子,多特征粒化形成粒向量^[22]与粒矩阵^[23-24],而且粒向量包含更多的特征信息,所以粒向量可以通过多个特征角度来构建,让模型的分类效果更好。传统感知机难以处理非线性数据,为此提出一种基于 GRAM 矩阵的粒感知机模型。该模型对感知机的对偶形式进行优化,构造对偶版本的粒感知机模型,为了能够更好地处理非线性分类问题,引入核函数,定义基于粒向量的核 GRAM 矩阵,并给出 GRAM 粒感知机的损失函数和学习方法。本文从收敛性、非线性分类能力和参考样本系上对模型进行分析,通过实验表明粒感知机模型的有效性与正确性。

2 多维数据粒化

感知机具有结构简单、参数较少、易于训练等特点,且难以拟合非线性数据。而现实中的问题往往是多维、复杂与非线性的。为使感知机模型能更好地处理复杂问题,引入粒计算理论,以样本间的相似性、距离度量方式等作为依据,进行特征粒化而构造特征粒子。在样本集中随机抽取若干样本作为参考样本集,一个样本与参考样本进行特征度量粒化为粒子,多维度上的特征度量则粒化为粒向量^[22]。

设 $U=(X,C,\{d\})$ 为分类系统,其中 $X=\{x_1,x_2,\dots,x_n\}$ 为样本集,从样本集中随机抽取 k 个样本作为参考样本集 $Q=(q_1,q_2,\dots,q_k)$, $C=\{c_1,c_2,\dots,c_m\}$ 为条件特征集合, d 为决策特征。对于任意样本 $x \in X$,其对应的决策值为 $y \in \{-1,+1\}$ 。

定义 1 设分类系统为 $U=(X,C,\{d\})$,对于样本 $x_i,x_j \in X$,单特征 $c \in C$,则 x_i 与 x_j 在单特征 c 上的曼哈顿距离为:

$$s_c(x_i,x_j) = |v(x_i,c) - v(x_j,c)|$$

其中, $v(x,c) \in R$ 表示样本 x 在特征 c 上的值。

定义 2 设 $U=(X,C,\{d\})$ 为分类系统,参考样本集为 $Q=(q_1,q_2,\dots,q_k)$,对于样本 $\forall x_i \in X$ 和条件特征 $\forall c \in C$,则 x_i 在参考样本 Q 和条件特征 c 进行粒化,构造条件粒子为:

$$g_c(x_i) = \{g_c(x_i)_j\}_{j=1}^k = \{r_j\}_{j=1}^k = \{r_1,r_2,\dots,r_k\}$$

其中,粒核为 $g_c(x_i)_j = r_j = s_c(x_i,x_j)$,是样本 x_i,x_j 在特征 c 上的曼哈顿距离。条件粒子由粒核组成。

定义 3 设在分类系统中存在样本 x_i ,该样本对应的决策值为 y_i ,则该样本 y_i 与参考样本 y_j 在决策特征进行度量,构造决策粒子为:

$$g_d(y_i) = \{g_d(y_i)_j\}_j^k = \{r_j\}_{j=1}^k = \{r_1,r_2,\dots,r_k\}$$

其中, $r_j = 1 - |y_i - y_j|$,由 $y_i,y_j \in \{-1,+1\}$,易知 $r_j \in \{-1,+1\}$ 。

分类系统中的任意训练样本 (x_i,y_i) 在参考样本上对特征进行粒化后, x_i 形成条件粒子, y_i 形成决策粒子。多个粒核组成一个粒子,条件粒子的粒核值为实数,决策粒子的粒核值为 ± 1 。

定义 4 设分类系统为 $U=(X,C,\{d\})$,对于任一样本 $x_i \in X$,任一特征子集 $P \subseteq C$,设 $P = \{a_1,a_2,\dots,a_m\}$,则 x_i 在特征子集 P 上的粒向量定义为:

$$G_P(x_i) = (g_{a_1}(x_i),g_{a_2}(x_i),\dots,g_{a_m}(x_i))^T$$

其中, $g_{a_m}(x_i)$ 是样本 x_i 在特征 a_m 上的粒子。

为方便计算,特征集 $P = \{a_1,a_2,\dots,a_m\}$ 用整数标记,则粒向量可表示成: $G(x_i) = (g_1(x_i),g_2(x_i),\dots,g_m(x_i))^T$ 。

多个粒核组合成一个粒子,多个粒子组合成一个粒向量。

由此,粒向量是一个矩阵,表示为:

$$G(x_i) = \begin{bmatrix} g_1(x_i)_1, g_1(x_i)_2, \dots, g_1(x_i)_k \\ g_2(x_i)_1, g_2(x_i)_2, \dots, g_2(x_i)_k \\ \dots \\ g_m(x_i)_1, g_m(x_i)_2, \dots, g_m(x_i)_k \end{bmatrix} = \begin{bmatrix} r_{11}, r_{12}, \dots, r_{1k} \\ r_{21}, r_{22}, \dots, r_{2k} \\ \dots \\ r_{m1}, r_{m2}, \dots, r_{mk} \end{bmatrix}$$

粒向量也可以用粒核向量来表示:

$$G(x_i) = (g(x_i)_1, g(x_i)_2, \dots, g(x_i)_k)$$

其中, $g(x_i)_j = (g_1(x_i)_j, g_2(x_i)_j, \dots, g_m(x_i)_j)^T$ 为粒核向量。

3 粒的运算与 GRAM 矩阵

3.1 粒的运算规则

给定分类系统为 $U=(X,C,\{d\})$,样本集合为 $X=\{x_1,x_2,\dots,x_n\}$,随机选取的参考样本集为 $Q=(q_1,q_2,\dots,q_k)$,条件特征集合为 $C=\{c_1,c_2,\dots,c_m\}$,决策特征为 d ,对于该分类系统,有以下运算规则^[22]。

定义 5 设任意样本 $\forall x_i,x_h \in X$ 和任意特征 $a \in C$,以及 x_i,x_h 在特征 a 上的两个粒子为 $g_a(x_i) = \{s_j\}_{j=1}^k, g_a(x_h) = \{t_j\}_{j=1}^k$,则定义粒子的加、减、乘、除运算为:

$$g_a(x_i) + g_a(x_h) = \{s_j + t_j\}_{j=1}^k$$

$$g_a(x_i) - g_a(x_h) = \{s_j - t_j\}_{j=1}^k$$

$$g_a(x_i) * g_a(x_h) = \{s_j * t_j\}_{j=1}^k$$

$$g_a(x_i) / g_a(x_h) = \{s_j / t_j\}_{j=1}^k$$

定义 6 设存在样本 $\forall x_i \in X$ 和实数 α ,有粒子 $g(x_i) = \{r_j\}_{j=1}^k$,则实数与粒子相乘的结果定义为:

$$\alpha * g(x_i) = \{\alpha * r_j\}_{j=1}^k = \{\alpha * r_1, \alpha * r_2, \dots, \alpha * r_k\}$$

定义 7 设存在两个样本 $\forall x_i,x_h \in X$ 和特征 $a \in C$, $g_a(x_i) = \{s_j\}_{j=1}^k, g_a(x_h) = \{t_j\}_{j=1}^k$,则粒子的欧氏距离度量为:

$$o(g_a(x_i), g_a(x_h)) = \sqrt{\sum_{j=1}^k (s_j - t_j)^2}$$

定义 8 设存在样本 $\forall x_i \in X$,有 $g(x_i) = \{r_j\}_{j=1}^k$,则该粒子的 sign 函数定义为:

$$\text{sign}(g(x_i)) = \{\text{sign}(r_j)\}_{j=1}^k = \{\text{sign}(r_1), \text{sign}(r_2), \dots, \text{sign}(r_k)\}$$

其中, $\text{sign}(r_j) = \begin{cases} +1, & r_j \geq 0 \\ -1, & r_j < 0 \end{cases}$ 。

定义 9 设粒子为 $g_c(x) = \{r_j\}_{j=1}^n$,其质量定义为:

$$q(g_c(x)) = \sum_{j=1}^n r_j$$

若 $q(g_c(x)) > 0$,则粒子质量为正;反之,粒子质量为负。

定义 10 设粒子为 $g_c(x) = \{g_c(x)_j\}_{j=1}^n = \{r_j\}_{j=1}^n$,若粒核 $\forall g_c(x)_j > 0$,则称 $g_c(x)$ 为正粒子;若 $\exists g_c(x)_j \leq 0$,则称 $g_c(x)$ 为负粒子。

定义 11 设存在两个样本 $\forall x_i,x_j \in X$,在其参考样本下被粒化的两个粒向量分别为 $G(x_i) = (g_1(x_i), g_2(x_i), \dots, g_m(x_i))^T, G(x_j) = (g_1(x_j), g_2(x_j), \dots, g_m(x_j))^T$,则这两个粒向量点乘结果为:

$$\begin{aligned} G(x_i) \cdot G(x_j) &= G(x_i)^T G(x_j) \\ &= g_1(x_i) * g_1(x_j) + g_2(x_i) * g_2(x_j) + \dots + \\ &\quad g_m(x_i) * g_m(x_j) \end{aligned}$$

实数是向量的基本元素,粒子为粒向量的基本元素。因此,两个向量的点乘结果是一个实数,那么两个粒向量的点乘结果为一个粒子。

定义 12 设存在两个样本 $\forall x_i, x_h \in X$,在其参考样本下被粒化的两个粒向量分别为 $G(x_i) = (g_1(x_i), g_2(x_i), \dots, g_m(x_i))^T, G(x_h) = (g_1(x_h), g_2(x_h), \dots, g_m(x_h))^T$,则这两个粒向量的距离为:

$$\begin{aligned} o(G(x_i), G(x_h)) &= \|G(x_i) - G(x_h)\| \\ &= \sqrt{\sum_{j=1}^m (o(g_j(x_i), g_j(x_h)))^2} \end{aligned}$$

其中, $o(g_j(x_i), g_j(x_h))$ 为粒子的欧氏距离。

$$\text{GRAM} = \begin{bmatrix} G(x_1) \cdot G(x_1), G(x_1) \cdot G(x_2), \dots, G(x_1) \cdot G(x_n) \\ G(x_2) \cdot G(x_1), G(x_2) \cdot G(x_2), \dots, G(x_2) \cdot G(x_n) \\ \dots \\ G(x_n) \cdot G(x_1), G(x_n) \cdot G(x_2), \dots, G(x_n) \cdot G(x_n) \end{bmatrix}$$

为解决非线性问题,可在数据空间中进行映射,将非线性分类问题转换为线性分类问题。使用映射函数 $\phi(x)$ 来完成特征空间的转换,将数据投影到另一空间中,使得粒向量中粒子的维度升高。

定义 14 设有 $x_i, x_k \in \chi, \vartheta$ 是一个特征空间,存在一个转换函数 $\phi(x)$ 使得 $\chi \rightarrow \vartheta$,对 $\forall x_i, x_k \in \chi$,存在相应的粒向量为

$$\text{GRAM} = \begin{bmatrix} K(G(x_1), G(x_1)), K(G(x_1), G(x_2)), \dots, K(G(x_1), G(x_n)) \\ K(G(x_2), G(x_1)), K(G(x_2), G(x_2)), \dots, K(G(x_2), G(x_n)) \\ \dots \\ K(G(x_n), G(x_1)), K(G(x_n), G(x_2)), \dots, K(G(x_n), G(x_n)) \end{bmatrix}$$

核函数是原始空间到特征空间的映射函数。将粒子投影到高维空间后,使用核函数,无需显式地求出转换函数。以下为常用的核函数公式。

定义 15 设 $U = (X, C, \{d\})$ 为分类系统,对于两个样本 $\forall x_i, x_j \in X$,存在其相应的粒向量为 $G(x_i) = (g_1(x_i), g_2(x_i), \dots, g_m(x_i))^T, G(x_j) = (g_1(x_j), g_2(x_j), \dots, g_m(x_j))^T$,则粒向量的线性核函数、多项式核函数和高斯核函数如下。

1) 线性核函数

$$K(G(x_i), G(x_j)) = G(x_i) \cdot G(x_j) + c$$

2) 多项式核函数

$$K(G(x_i), G(x_j)) = (G(x_i) \cdot G(x_j) + c)^d$$

3) 高斯核函数

$$K(G(x_i), G(x_j)) = \exp\left(-\frac{\|G(x_i) - G(x_j)\|^2}{2\sigma^2}\right)$$

4 基于 GRAM 矩阵的粒感知机

感知机是一种二分类模型,其输入为样本的特征向量,输出为预测的样本类别,预测值为 1 或 -1,也称为正例或负例。粒感知机将输入样本的特征向量粒化为粒向量,经过一系列粒的运算规则,对其进行预测,输出粒核仅为 -1 或 1 的粒子,计算粒子大小,进而判断预测的样本类别。

定义 16 设 $U = (X, C, \{d\})$ 为分类系统,对于样本 $\forall x_i \in X$,存在相应的粒向量 $G(x_i) = (g_1(x_i), g_2(x_i), \dots,$

3.2 GRAM 矩阵与核函数

已知传统感知机的对偶模式在某些情形下准确率更高,从其对偶形式学习算法的过程可以看出,以内积形式存在的特征向量,在感知机对偶形式的训练算法中,也可以用核函数替代内积的形式来构造核感知机。为提高运算效率,会提前计算好所有粒向量之间的内积并存储于 GRAM 矩阵中,查表使用。

定义 13 给定训练数据集 $T = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$,其中 $x_i \in X \subseteq R^m, y_i \in \{-1, +1\}, i = 1, 2, \dots, n$ 。训练集 T 粒化为 $GT = \{(G(x_1), g(y_1)), (G(x_2), g(y_2)), \dots, (G(x_n), g(y_n))\}$,条件粒向量 $G(x_i) = (g_1(x_i), g_2(x_i), \dots, g_m(x_i))^T$ 两两向量点乘得到的矩阵,成为 n 个条件粒向量的 GRAM 矩阵,表示为:

$G(x_i) = (g_1(x_i), g_2(x_i), \dots, g_m(x_i))^T, G(x_k) = (g_1(x_k), g_2(x_k), \dots, g_m(x_k))^T$,有 $K(G(x_i), G(x_k))$ 满足条件 $K(G(x_i), G(x_k)) = G(\phi(x_i)) \cdot G(\phi(x_k))$,则 $K(G(x_i), G(x_k))$ 称为粒向量的核函数,其中 $G(\phi(x_i)) \cdot G(\phi(x_k))$ 为 $G(\phi(x_i))$ 和 $G(\phi(x_k))$ 的内积。粒感知机中则采用核函数进行映射,其 GRAM 矩阵为:

$g_m(x_i))^T$ 、权重粒向量 $W = (w_1, w_2, \dots, w_m)^T$ 和偏置粒子 b ,则粒感知机表示为:

$$f(G(x_i)) = \text{sign}(W \cdot G(x_i) + b)$$

4.1 粒感知机对偶形式及其损失函数

传统感知机的核心是通过训练集找到一个能对正反例完全正确分类的超平面。为找到这个超平面,需要定义模型的损失函数,确定一个学习策略,通过梯度下降的方式得到感知机分类超平面的参数 W, b 。感知机的损失函数为误分类点到超平面的总距离。粒计算将每个样本在单特征或多特征上进行粒化,构造粒子与粒向量。粒感知机的损失函数则为粒化的误分类点到超平面的总距离。

定义 17 给定训练数据集 $T = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$,其中 $x_i \in X \subseteq R^m, y_i \in \{-1, +1\}, i = 1, 2, \dots, n$ 。训练集 T 粒化为 $GT = \{(G(x_1), g(y_1)), (G(x_2), g(y_2)), \dots, (G(x_n), g(y_n))\}$,其中 $G(x_i) = (g_1(x_i), g_2(x_i), \dots, g_m(x_i))^T$ 为条件粒向量, $g(y_i)$ 为决策粒子。设权值粒向量为 $W = (w_1, w_2, \dots, w_m)^T$,偏置粒子为 b ,则粒感知机的原始损失函数为:

$$L(W, b) = - \sum_{G(x_i) \in M} g(y_i) * (W \cdot G(x_i) + b)$$

其中, M 为误分类粒向量的集合。

对于感知机原始形式,有损失函数:

$$L(W, b) = - \sum_{G(x_i) \in M} g(y_i) * (W \cdot G(x_i) + b)$$

在原始形式中,首先选取一个超平面,使用随机梯度下降

法求得损失函数极小化问题的解,则有:

$$\min_{W, b} L(W, b) = - \sum_{G(x_i) \in M} g(y_i) * (W \cdot G(x_i) + b)$$

对损失函数的 W 和 b 进行求导,得出:

$$\nabla_W L(W, b) = - \sum_{G(x_i) \in M} G(x_i) * g(y_i)$$

$$\nabla_b L(W, b) = - \sum_{G(x_i) \in M} g(y_i)$$

随机选择一个误分类粒子对参数 W, b 更新:

$$W \leftarrow W + \eta * G(x_i) * g(y_i)$$

$$b \leftarrow b + \eta * g(y_i)$$

选取一个误分类粒子的更新如上,而这种学习方式实际上是一种逐步学习方式。若假设样本 $(G(x_i), g(y_i))$ 的误分类次数为 n_i , 样本点总个数为 N , 则推导出 W 和 b 的表达式如下:

$$W = \sum_{i=1}^N n_i \eta * g(y_i) * G(x_i)$$

$$b = \sum_{i=1}^N n_i \eta * g(y_i)$$

设 $\alpha_i = n_i \eta$, 有 $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n)^T$, 其中 $n_i (i=1, 2, 3, \dots, N)$ 为每个样本的误分类次数, W 和 b 的表达式如下:

$$W = \sum_{i=1}^N \alpha_i * g(y_i) * G(x_i)$$

$$b = \sum_{i=1}^N \alpha_i * g(y_i)$$

只要找到所有样本的更新次数,即可求出 W 和 b 。同时,感知机的数学式可表示为:

$$f(G(x_i)) = (\sum_{j=1}^n \alpha_j * g(y_j) * G(x_j) \cdot G(x_i) + b)$$

其中,只需对 α_j 和 b 进行更新, $\alpha_j = \alpha_j + \eta, b = b + \eta * g(y_i)$, 即可找到最佳分离超平面。

针对核函数 GRAM 矩阵,感知机公式变换为:

$$f(G(x_i)) = (\sum_{j=1}^n \alpha_j * g(y_j) * K(G(x_j), G(x_i)) + b)$$

其中, $K(G(x_j), G(x_i))$ 为核函数 GRAM 矩阵。

4.2 基于 GRAM 矩阵的粒感知机算法

根据 4.1 节基于 GRAM 矩阵的粒感知机理论,可以设计其学习算法,包括粒感知机的训练算法(见算法 1)和推理算法(见算法 2)。

算法 1 GRAM 粒感知机的训练算法

输入:训练集为 $T = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$, 其中 x_i 为 m 维特征向量; $x_i \in X \subseteq \mathbb{R}^m, y_i \in \{-1, +1\}, i=1, 2, \dots, n$; 学习率为 η ($0 < \eta \leq 1$)

输出:参考样本集 Q , 权值粒向量 α 与偏置粒子 b

1. 随机从训练集中抽取 k 个样本,构造粒化的参考样本集 $Q = (q_1, q_2, \dots, q_k)$;
2. 训练集 T 粒化为 $GT = \{(G(x_1), g(y_1)), (G(x_2), g(y_2)), \dots, (G(x_n), g(y_n))\}$;
3. 初始化 $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n)^T$ 和偏置粒子 b , 其中 α_i 为样本 $(G(x_i), g(y_i))$ 被误分类的次数;
4. 条件粒向量 $G(x) = \{G(x_1), G(x_2), \dots, G(x_n)\}$ 两两之间进行点乘运算,形成 GRAM 矩阵;
5. 从粒化后的训练集中选取 $(G(x_i), g(y_i))$;
6. 若 $g(y_i) * (\sum_{j=1}^n \alpha_j * g(y_j) * K(G(x_j), G(x_i)) + b)$ 为负粒子,则 $\alpha_i = \alpha_i + \eta, b = b + \eta * g(y_i)$;
7. 转至步骤 3,直至 $g(y_i) * (\sum_{j=1}^n \alpha_j * g(y_j) * K(G(x_j), G(x_i)) + b)$ 结果为正粒子;

8. 输出参考样本集 $Q = (q_1, q_2, \dots, q_k)$ 、权值粒向量 $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n)^T$ 和偏置粒子 b 。

算法 2 GRAM 粒感知机的推理算法

输入:参考样本集 $Q = (q_1, q_2, \dots, q_k)$, 权值粒向量 $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n)^T$ 和偏置粒子 b , 测试样本 x

输出:推理样本 x 的类别

1. 测试样本 x 在参考样本 $Q = (q_1, q_2, \dots, q_k)$ 中粒化为 $G(x)$;
2. 计算 $f(G(x)) = (\sum_{j=1}^n \alpha_j * g(y_j) * K(G(x), G(x_j)) + b)$;
3. 计算粒子 q ($f(G(x))$) 的质量,若粒子质量为正,则样本 x 为正例,反之则为负例。

5 实验分析

实验数据集选自 UCI 机器学习仓库,如表 1 所列,其中 4 个二分类数据集,1 个三分类数据集。为了让所有数据集适应粒感知机的学习策略,将所有数据集类别标签都重置为 -1 和 1,其中 Wholesale 数据集取其中的两个类别进行分类。

表 1 UCI 数据集的描述

Table 1 Description of UCI datasets

| 数据集 | 特征数 | 类别数 | 样本数 |
|-------------|-----|-----|-----|
| Haberman | 3 | 2 | 306 |
| Ionosphere | 34 | 2 | 351 |
| Transfusion | 4 | 2 | 748 |
| Wholesale | 8 | 3 | 440 |
| Sonar | 60 | 2 | 208 |

由于数据集的特征都是实物世界的的数据,各个特征的数值范围不同,为了使得所有数据都映射到同一尺度,本实验将所有使用到的数据都进行最大最小值归一化,将所有特征的值域范围定在 $[0, 1]$ 之间,使得所有数据具有相同量纲。最大最小值归一化的公式如下所示:

$$X_{\text{norm}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}}$$

为了验证该算法的有效性,将其与经典的机器学习算法进行对比,分析了 GRAM 粒感知机的收敛性、非线性分类能力,并将参考样本数量对其分类结果的影响进行了比较。

5.1 收敛性分析

在收敛性分析实验中,对比了 Ionosphere 数据集在传统感知机与 GRAM 粒感知机上的收敛程度。在实验中,对两种分类器均采用十折交叉法,将数据集 10 等分,随机抽取 90% 的数据作为训练集,10% 的数据作为测试集,对其迭代 30 次,往复训练 10 轮,其中 GRAM 感知机的粒化参考全局样本,最终结果如图 1 和图 2 所示。

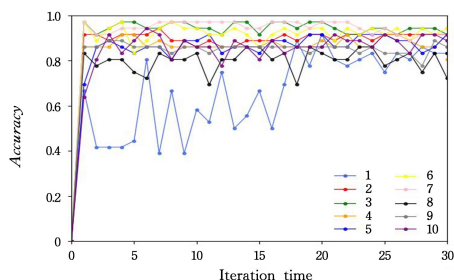


图 1 Ionosphere 在感知机上进行 10 次训练的准确率

Fig. 1 Accuracy of Ionosphere after 10 rounds of training on the perceptron

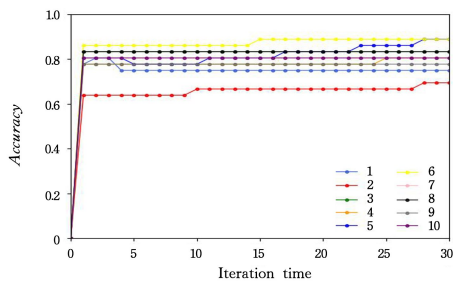


图2 Ionosphere 在 GRAM 粒感知机上进行 10 次训练的准确率
Fig.2 Accuracy of Ionosphere after 10 rounds of training on the GRAM granular perceptron

从图 1、图 2 可知,两个分类器经迭代训练后,均达到较为稳定的准确率,且在第 1 次迭代后就达到较好的分类精度。然而,在图 1 中,虽然每轮训练最终几乎都能达到 60%~90% 的准确率,但其波动较大。而对比图 2 可以发现,GRAM 粒感知机每轮训练,在第 1 次迭代以后就得到一个比较好的效果,且在经过 5 次迭代以后就已经趋于稳定,上下波动程度非常小。因此,在收敛性分析实验中可以看出,GRAM 粒感知机具有收敛快和稳定性好的特点。GRAM 粒感知机的收敛非常迅速,迭代次数少,分类效果明显,其稳定性明显优于传统感知机。

5.2 参考样本数量的影响

对偶模式的感知机需提前计算所有样本特征向量之间的

表 2 GRAM 粒感知机在不同参考样本数量下的准确率

Table 2 Accuracy of GRAM perceptron under different numbers of reference samples

| 数据集 | 1 | 5 | 10 | 20 | 30 | All | 样本总数 |
|-------------|---------------|---------------|--------|--------|--------|--------|------|
| Haberman | 0.7516 | 0.7484 | 0.7419 | 0.7419 | 0.7290 | 0.7419 | 306 |
| Ionosphere | 0.8472 | 0.8388 | 0.8388 | 0.8333 | 0.8250 | 0.8250 | 351 |
| Transfusion | 0.7653 | 0.7733 | 0.7693 | 0.7680 | 0.7653 | 0.7653 | 748 |
| Wholesale | 0.6538 | 0.6769 | 0.6769 | 0.6769 | 0.6769 | 0.6769 | 124 |
| Sonar | 0.7762 | 0.5619 | 0.5476 | 0.5476 | 0.5476 | 0.5476 | 208 |

5.3 非线性性能分析

感知机模型是一个二分类线性模型,异或问题由于具有非线性特征,一直都是感知机模型难以解决的问题之一。而 GRAM 粒感知机由于粒计算理论的支撑,比传统感知机拥有更高维度的权重向量,其可调整的参数更多,比传统感知机拥有更好地非线性拟合能力。此外,为了更好地解决非线性问题,本实验引进了核方法,并设计实验对其进行比较。本文涉及到的感知机模型只能预测正反例结果,因此本实验将异或问题中的 0 全部改为 -1,如表 3 所列。

表 3 XOR 异或函数

Table 3 XOR function

| U | X1 | X2 | Y |
|---|----|----|----|
| 1 | -1 | -1 | -1 |
| 2 | -1 | 1 | 1 |
| 3 | 1 | -1 | 1 |
| 4 | 1 | 1 | -1 |

图 3 展示了 GRAM 粒感知机算法在 3 种核函数上进行 10 次迭代训练后的准确率,其中线性核的非线性分类能力较弱,但最终也能达到 80% 以上的准确率。而高斯核与多项式核的非线性能力明显优于线性核,对于异或问题,两者都达到了 100% 的准确率。图 4 则对比了在 8 个异或样本中传统感知机与 GRAM 矩阵感知机的预测结果,可知传统感知机的非

内积,将其存储于 GRAM 矩阵,而涉及到矩阵运算的运算量往往较为庞大,同时粒计算理论中构造粒子的方式,导致 GRAM 矩阵的规格会比传统感知机更高。为了提高运行速度,本实验对粒化参考样本的数量进行更改。参考样本的数量决定了粒子的维度,若能选择适当的参考样本,则粒向量之间点乘的运算量也会减少,从而大幅度提高实验的效率。

为了探究参考样本数量对实验结果的影响,分别随机选取 1,5,10,20,30 个样本和所有样本作为参考样本,对 5 个 UCI 数据集进行测试。由表 2 可知,大部分数据集在选取 1—10 个样本作为参照样本时,实验效果达到最好。对于 Haberman 数据集、Ionosphere 数据集以及 Sonar 数据集,在参考样本数量为 1 时就达到了比较好的效果;而对于 Transfusion 数据集和 Wholesale 数据集,当参考样本的数量达到 5 以后,才达到峰值。大部分数据集在当参考样本达到 10—20 以后,逐渐趋于稳定,其中 Wholesale 数据集由于样本数量较少,当参考样本达到 5 时,基本就趋于稳定。此后,无论参考样本的数目如何改变,对模型的准确性都基本没有太大的影响。由此看出,参考样本数量对实验效果有显著的影响,参考样本数量多,其包含的信息也就越多,但是特征粒子维度过多有时并不会给实验带来更好的效果,过多的特征反而会成为冗余的信息,对模型找到合适决策边界造成影响,从而可能降低模型的准确性。因此,适当选择参考样本的数量,不仅可以提高运算效率,也有利于提升模型的准确性。

线性处理能力较弱,而 GRAM 矩阵具有较强的非线性处理能力。

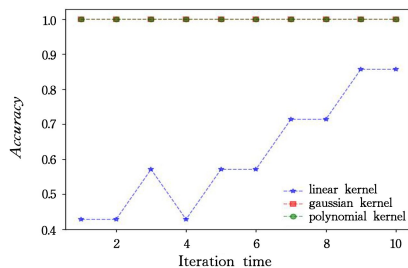


图 3 3 种核函数在异或问题上进行 10 次迭代训练的准确率
Fig.3 Accuracy of 10 iterations of training with three kernel functions on the XOR problem

5.4 经典算法比较

常用的机器学习模型包括感知机、逻辑回归、KNN、决策树、高斯贝叶斯、SVM、Adaboost 以及随机森林算法等。为测试 GRAM 粒感知机的分类能力,本文详细对比了 GRAM 粒感知机与这 8 个经典机器学习模型在 5 个 UCI 数据集上的分类效果。其中,Adaboost 分类使用了 100 个弱分类器,GRAM 粒感知机模型均随机选取了 5 个样本作为参考样本。本次实验均使用十折交叉法,以分类准确度为衡量标准进行比较。

表4 在常用机器学习模型上的准确率比较

Table 4 Accuracy comparison among commonly used machine learning models

| 数据集 | GRAM 粒感知机 | 感知机 | 逻辑回归 | KNN | 决策树 CART | 高斯 贝叶斯 | SVM | Adaboost | 随机森林 |
|-------------|---------------|--------|--------|--------|-------------|-----------|---------------|----------|---------------|
| Haberman | 0.7484 | 0.6499 | 0.7320 | 0.7124 | 0.6085 | 0.7519 | 0.7549 | 0.6158 | 0.6804 |
| Ionosphere | 0.8388 | 0.8406 | 0.8690 | 0.8377 | 0.8691 | 0.8775 | 0.9401 | 0.9087 | 0.9287 |
| Transfusion | 0.7733 | 0.7421 | 0.7721 | 0.6807 | 0.6258 | 0.7473 | 0.7580 | 0.7514 | 0.6753 |
| Wholesale | 0.6769 | 0.5487 | 0.6212 | 0.5199 | 0.5244 | 0.5981 | 0.6212 | 0.5974 | 0.5500 |
| Sonar | 0.5619 | 0.5540 | 0.6683 | 0.5952 | 0.5681 | 0.6071 | 0.6690 | 0.7012 | 0.7217 |

各个数据集在常用机器学习算法上的分类准确率比较如表4所列。可知,GRAM粒感知机的分类效果,除 Ionosphere 数据集外,在其他数据集上均优于传统感知机;而对于 Transfusion 和 Wholesale 数据集,其准确率明显优于其他算法。其中 Sonar 数据集和 Haberman 数据集由于参考样本数量选取的原因,虽 GRAM 粒感知机的准确率不为全部算法中的最优值,但若参考表2选取适当的参考样本与效果最佳的分类器相比,GRAM 粒感知机的分类效果也达到最优。总之,GRAM 粒感知机算法的分类能力明显高于传统感知机,在非线形分类方面表现优异。

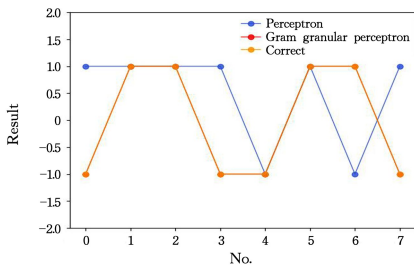


图4 两种算法在异或问题上的预测结果

Fig. 4 Prediction results of the two algorithms on the XOR problem

结束语 针对传统感知机的非线性处理能力弱的固有缺陷,提出一种基于 GRAM 矩阵的粒感知机模型。该模型通过构造粒的方式对数据进行处理,增加模型参数的维度。此外,该模型还基于传统感知机模型的对偶模式,将其中成对出现的点积运算替换成核函数,从而使得非线性分类能力更强。而由于粒 GRAM 矩阵的规模远远大于传统 GRAM 矩阵,为提升运行效率,本文通过改变参考样本的数量探究该参数对分类效果的影响。实验也从收敛性、非线性处理能力、参考样本的数量等方面将 GRAM 与传统机器学习模型对比分析,验证了 GRAM 粒感知机的有效性。感知机作为一种线性的二分类模型,是多层感知机和 SVM 的基础。在未来的研究中,可基于单层感知机对更复杂的算法的粒化方式进行探讨;同时,如何构造粒化参照样本,才能剔除粒子中的冗余信息,或是以其他更能够提取全局信息的方式构造粒子,这些都需要未来逐步完善和探究。

参考文献

[1] ZADEH L A. Fuzzy sets and information granularity [J]. Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems, 1996, 8: 433-448.

[2] WANG G Y, ZHANG Q H, HU J. An overview of granular computing [J]. CAAI Transactions on Intelligent Systems, 2016, 29(6): 927-933.

[3] LIN T Y. Granular computing on binary relations I: data mining

and neighborhood systems [J]. Rough Sets in Knowledge Discovery, 1998, 1(1): 107-121.

[4] XUE R X, YI S C, WANG P X. GBDEn: a fast clustering algorithm for large-scale data based on granular ball [J]. Computer Science, 2024, 12: 166-173.

[5] MIAO D Q, WANG J. On the relationships between information entropy and roughness of knowledge in rough set theory [J]. PR & AI, 1998, 11(1): 34-40.

[6] MIAO D Q, WANG J. An information representation of the concepts and operations in rough set theory [J]. Journal of Software, 1999, 10(2): 113-116.

[7] MIAO D Q, ZHANG Q H, et al. From human intelligence to machine implementation model: theories and applications based on granular computing [J]. CAAI Transactions on Intelligent Systems, 2016, 11(6): 743-757.

[8] YAO Y Y. Granular computing using neighborhood systems [M] // Advances in Soft Computing: Engineering Design and Manufacturing. London: Springer, 1999: 539-553.

[9] PEDRYCZ W. Granular computing for data analytics: a manifesto of human-centric computing [J]. IEEE/CAA Journal of Automatica Sinica, 2018, 5(6): 1025-1034.

[10] MIAO D Q, FAN S D. The calculation of knowledge granulation and its application [J]. Systems Engineering—Theory & Practice, 2002, 22(1): 48-56.

[11] HU Q H, YU D R, XIE Z X. Numerical attribute reduction based on neighborhood granulation and rough approximation [J]. Journal of Software, 2008, 19(3): 640-649.

[12] CHEN Y M, ZHU S Z, LI W, et al. Fuzzy granular convolutional classifiers [J]. Fuzzy Sets and Systems, 2022, 426: 145-162.

[13] CHEN Y M, QIN N, LI W, et al. Granule structures, distances and measures in neighborhood systems [J]. Knowledge-Based Systems, 2019, 165: 268-281.

[14] ZHENG C Y, CHEN Y Y, HOU X Y, et al. A neighbourhood granular fuzzy c-means clustering algorithm [J]. Journal of Shandong University(Natural Science), 2024, 59(5): 35-44.

[15] ROSENBLATT F. The perception: a probabilistic model for information storage and organization in the Brain [J]. Psychological Review, 1958, 65(6): 111-127.

[16] LI W, YANG H. A non-linear blind source separation method based on perceptron structure and conjugate gradient algorithm [J]. Circuits, Systems, and Signal Process, 2014, 33: 3573-3590.

[17] XU J H, ZHANG X G, LI Y D. A nonlinear perceptron algorithm based on kernel functions [J]. Chinese Journal of Computers, 2002, 25(7): 689-695.

[18] CHOLAQUIDIS A, FRAIMAN R, KALEMKERIAN J, et al. A nonlinear aggregation type classifier [J]. Journal of Multivariate Analysis, 2016, 146: 269-281.

- [19] NIE F P,ZHU W,LI X L. Decision tree svm : an extension of linear svm for non-linear classification [J]. Neurocomputing, 2020,401:153-159.
- [20] KWAK N. Nonlinear projection trick in kernel methods;an alternative to the kernel trick [J]. IEEE Transactions on Neural Networks and Learning Systems,2013,24(12):2113-2119.
- [21] KEMPFER K C,WANG Y,et al. A comparison study on nonlinear dimension reduction methods with kernel variations;visualization, optimization and classification [J]. Intelligent Data Analysis,2020,24(2):267-290.
- [22] FU X Y,CHEN Y Y,CHEN Y M,et al. A classification method of fully connected granular neural network [J]. Journal of Shanxi University(Natural Science Edition), 2023, 46 (1) : 91-100.
- [23] YANG T,ZHONG X R,LANG G M,et al. Granular matrix;a new approach for granular structure reduction and redundancy evaluation [J], IEEE Transactions on Fuzzy Systems, 2020, 28(12):3133-3144.
- [24] LIU W X,LI J J,WANG H W. A dynamic attribute reduction method for formal context based on matrix information entropy [J]. Journal of Nanjing University (Natural Science), 2025, 61(1):117-128.



WU Shaohua, born in 1985, senior engineer. His main research interests include machine learning and granular computing.



CHEN Yuming, born in 1977, Ph.D, professor, Ph.D supervisor, is a member of CCF(No. 60007M). His main research interests include deep learning and granular computing.