

# 结合切空间及特征空间校准的增量流形学习正则优化算法

谈超<sup>1,2</sup> 吉根林<sup>2</sup> 赵斌<sup>2</sup>

(1. 东南大学计算机科学与工程学院, 南京, 211189; 2. 南京师范大学计算机科学与技术学院, 南京, 210023)

**摘要:** 高维流式大数据的产生与发展对传统机器学习和数据挖掘算法提出了诸多挑战。本文结合流式大数据流式到达的特性, 首先建立自适应增量特征提取算法模型。然后, 针对噪声环境, 建立基于特征空间校准的增量流形学习算法模型, 解决小样本问题。最后, 构造流形学习的正则化优化框架, 解决高维数据流特征提取过程中产生的降维误差问题, 并得到最终的最优解。实验结果表明本文提出的算法框架符合流形学习算法的3个评价指标: 稳定性、提高性以及学习曲线能迅速增加到一个相对稳定的水平; 从而实现了高维数据流的高效学习。

**关键词:** 高维流式大数据; 自适应增量特征提取; 特征空间校准; 正则化优化

**中图分类号:** TP181      **文献标志码:** A

## Incremental Manifold Learning Regular Optimization Algorithm on Tangent Space and Feature Space Alignment

Tan Chao<sup>1,2</sup>, Ji Genlin<sup>2</sup>, Zhao Bin<sup>2</sup>

(1. School of Computer Science and Engineering, Southeast University, Nanjing, 211189, China; 2. School of Computer Science and Technology, Nanjing Normal University, Nanjing, 210023, China)

**Abstract:** The emergence and development of high dimensional big data streams have presented a great challenge to the traditional machine learning and data mining algorithms. Based on the characteristics of data flow, first we construct an adaptive incremental feature extraction algorithm model. Then, according to the environment with noise, we establish an incremental manifold learning algorithm model based on feature space alignment to solve the small size sample problem. Finally, the regularization optimization framework of manifold learning is constructed to solve the problem of dimensionality reduction errors of high-dimensional data flow in feature extraction process, and then the optimal solutions are obtained. Experimental results show that the proposed algorithm framework conforms to the three evaluation criteria of manifold learning algorithm: Stability, enhancement, and the learning curve can rapidly increase to a relative stable level. Thus the efficient learning of high-dimensional data streams can be realized.

**Key words:** high dimensional big data streams; adaptive incremental feature extraction; feature space alignment; regularization optimization

## 引 言

流式大数据<sup>[1]</sup>作为大数据的一种重要形态,在商务智能、市场营销和公共服务等诸多领域有着广泛的应用前景,并已在互联网、金融银行业和气象监测等场景的应用中取得了显著的成效<sup>[2]</sup>。各个行业产生的业务数据大多数情况下可以看作动态达到的流式数据。与传统数据相比,流式大数据呈现出实时性、易失性、突发性、无序性和无限性等特征<sup>[2]</sup>。文献[2]给出了流式大数据这几个特征的解释:(1)实时性。流式大数据是实时产生、实时计算,结果反馈往往也需要保证及时性;(2)易失性。在大数据流式计算环境中,大多数数据往往会被直接丢弃,数据的使用往往是一次性的、易失的;(3)突发性。在大数据流式计算环境中,由于不同的数据源在不同时空范围内的状态不统一且发生动态变化,导致数据流的速率呈现出了突发性的特征,前一时刻数据速率和后一时刻数据速率可能会有巨大的差异;(4)无序性。在大数据流式计算环境中,各数据流之间、同一数据流内部各数据元素之间是无序的;(5)无限性。在大数据流式计算中,数据是实时产生、动态增加的,只要数据源处于活动状态,数据就会一直产生和持续增加下去,可以说,潜在的数据量是无限的<sup>[2]</sup>。首先,大批量的数据源源不断地涌入,将这类数据完全存储下来几乎不可能。其次,数据具有时间属性,带有强烈时间特征,训练样本和测试样本的分布可能不同、样本的特征可能随时间变化(增加或者缺失),同时可能有新的类别产生,呈现动态变化的特点,也就是流式数据的动态特征<sup>[3]</sup>。这样的数据分布动态变化的问题给机器学习带来一些深刻的变化和挑战<sup>[4]</sup>。例如,对于呈流式到达的数据形态,传统批量式的学习方法不仅时间耗费大,学习效率低,更严重的是难以针对流式到达的增量数据进行学习模型的有效更新,导致模型难以适应动态变化的数据流。尤其是流式大数据呈现出的高维、实时、无序及突发等显著特征,使得其与传统批量大数据在数据处理的要求、方式等方面有着明显的不同,也使得当前诸多模式识别和机器学习等相关算法无法直接应用到流式大数据处理中<sup>[2]</sup>。在此背景下对高维流式大数据的分析和挖掘成为当前的热点研究课题<sup>[5]</sup>。

在对高维流式大数据进行分析和挖掘时,首先就是对数据集中连续到达的增量样本进行降维特征提取(Feature extraction),从而发现高维数据流内在的结构信息,为后续的分类、聚类等学习任务的高效顺利进行提供支撑。然而,针对流式达到的高维大数据集,在对其进行降维分析时,现有大多数的特征提取算法仍存在如下一些挑战:首先,如何对流式大数据特征呈动态变化的特性进行研究,设计具有自适应特性的增量特征提取算法,使系统可以弹性、动态地适应数据流的变化,是目前的主要挑战之一<sup>[6]</sup>。其次,如何在噪声环境下对异构流式大数据的特征进行研究,设计具有鲁棒性的增量特征提取算法,降低迭代运算带来的计算复杂度,是一个关键的挑战性问题。最后,如何在特征提取阶段,结合高维数据流的特性构建目标函数,设计带正则化项的损失函数优化降维误差,提高系统学习效率,是一个富有挑战性的问题。

在大数据环境中面对高维海量的情况下,目前的计算设备无法支持批处理模式高昂的计算代价。于是近来有学者利用分布并行的思想切分数据,从而实现分而治之的目的,例如 MapReduce 机制<sup>[7]</sup>,但是分而治之的手段割裂了样本之间的关联,尤其是对于数据流中蕴含的时间属性无法进行有效的刻画,而增量学习的学习原理正好符合数据流式到达的场景。故本文结合流式大数据流式到达的特性,构造增量学习的“损失函数+正则化项”的优化框架,最终得到问题的最优解。

因此,基于上述考虑,作者研究了高维数据流的特征,从而提高多种学习算法的有效性。本文的主要贡献如下:(1)研究具有自适应特性的增量特征提取算法,建立自适应增量特征提取算法模型;(2)针对噪声环境下异构流式大数据的机器学习问题,建立基于特征空间校准的增量流形学习算法模型,解决小样本问题;(3)针对高维数据流特征提取过程中不可避免产生的降维误差问题,研究基于正则化的增量特征提取优化算法,实现高维数据流的高效学习。

## 1 相关工作

高维流式大数据的产生与发展对传统机器学习和数据挖掘算法提出了诸多挑战。例如:存在动态特征难以处理、现实环境噪声干扰严重以及特征提取中不可避免产生误差等问题。本文从增量特征提取算法、基于特征空间校准的特征提取算法及特征提取算法的优化3个方面对相关的研究工作、研究现状进行了分析与总结。

### 1.1 增量特征提取算法

由于现在的流式大数据对处理分析的实时性要求很高,传统的特征提取方法如主成分分析(Principal component analysis, PCA)<sup>[8]</sup>,线性判别分析(Linear discriminant analysis, LDA)<sup>[8]</sup>和偏最小二乘法(Partial least squares, PLS)<sup>[9]</sup>等均不能满足流式大数据的实时性及要求。

数据流实时性问题的通常解决方法是以传统特征提取为基础,对其进行快速更新。在主成分分析方面,文献[10]提出了一种可以增量的更新数据特征的主成分分析法(Candid covariance-free incremental PCA, CCIPCA),该算法无需计算协方差矩阵,不仅提高了计算速度,而且面对高维数据时可快速收敛。在协方差矩阵学习方面,文献[11]阐述了协方差矩阵自适应学习的主要理论基础。在线性判别分析方面,文献[12]提出一种快速增量线性判别式分析算法,可以很好地处理新样本的更新问题,具备较低的空间及时间复杂度。

### 1.2 基于特征空间校准的特征提取算法

随着信息科技的发展,人类所收集到的信息越来越复杂,有语音、图像、文字和视频等多种形式,这些信息可以看成高维空间中向量模式的样本点。模式分类的研究目标就是对高维空间中的这些样本进行分类和识别。然而,当样本空间的维数很高时,分类和识别样本点的计算量很大,直接把原始数据用来做分类实际上是不可行的,所以必须提取图像的一些重要特征<sup>[13]</sup>。

Chen等<sup>[14]</sup>提出的一种基于广义线性模型的降维算法(Generalized linear principal component analysis, GLPCA),将原始数据协方差矩阵的最大的几个特征值对应的特征向量组成一个线性的子空间,该子空间中的数据点具有最小的均方误差,从而将原来的高维数据映射到低维的线性子空间中。CCIPCA<sup>[10]</sup>属于线性降维方法,它无需计算协方差矩阵就能获得主成分。CCIPCA的一个问题在于它应用的数据要求服从线性分布,不能检测出数据集的非线性结构。如果将它应用到非线性分布数据中,就暴露出其局限性。这是因为若数据集服从非线性分布,数据点之间的欧氏距离不能代表高维空间中它们的真实距离,只有用测地距离也就是分布在非线性曲面上的点沿着该曲面的最短路径距离才能表示<sup>[13]</sup>。

### 1.3 特征提取优化技术

目前大多数机器学习算法都使用正则化损失函数的设计框架来求解机器学习优化问题,一般的批处理学习算法可表述为损失函数+正则化项的一般范式,则

$$\min_{\omega} F(\omega) = L(\omega) + \lambda R(\omega) \quad \text{s. t.} \quad \omega \in \mathbf{R} \quad (1)$$

式中: $\omega$ 为特征要优化的系数, $R$ 为约束条件决定的 $\omega$ 的解空间, $L(\omega)$ 描述了损失函数,而 $R(\omega)$ 刻画了正则化项用来表达问题的先验信息对该问题的约束,例如,真实数据集中内蕴的结构特征,局部邻域关系和流形结构信息等特征。 $\lambda$ 是用于平衡损失函数和正则化项之间关系的标量参数。

一个增量学习的损失函数+正则化项的一般理论框架可以描述为

$$\min_{\omega} F(\omega) = L(\omega, x_i) + \lambda R(\omega, x_i) \quad \text{s. t.} \quad \omega \in \mathbf{R} \quad (2)$$

机器学习中,大部分训练样本集的冗余度通常较高,对于大规模数据集来说,往往只需要很少的训练样本就能得到所需泛化能力的分类器。另外,增量或在线优化算法的样本随时间序列产生,每次仅求解与单个样本有关的子优化问题,从而使每次迭代的计算复杂度较低,因此增量或在线优化方法特别适合处理大规模数据问题<sup>[3]</sup>。

在本文中,将该“损失函数+正则化项”的优化框架用于作者提出的增量学习算法中,进行优化求解。为了适应不同环境(例如增量切空间校准和特征空间校准的增量学习环境),本文将不同特征提取算法和损失函数项结合起来得到适应不同环境的增量学习优化模型。

## 2 正则化增量流形学习算法

本文基于流式大数据环境建立正则化增量流形学习算法的模型。首先建立自适应特征提取的增量学习模型;再构建基于特征空间校准的增量特征提取算法模型;最后建立增量的正则化学习框架,并讨论增量学习算法的性能评估问题。

### 2.1 自适应特征提取的增量学习模型

为了实现流式大数据的增量学习,首先要对实时数据流中的增量样本进行特征提取,从而发现高维数据集的内在结构信息,有利于后续的分类、聚类等学习任务的高效顺利进行。然而现有大多数的降维算法在处理增量样本时呈现出一定的局限性(例如难以针对增量数据有效地更新模型,导致学习时间长、学习效率低等),无法直接应用到流式大数据的降维中。基于以上所述,本文以作者之前的工作一种基于增量切空间校准的流形学习算法<sup>[15]</sup>为基础,建立自适应特征提取的增量学习模型。在下文中将给出该学习模型的主要理论和算法步骤,该模型由 2 个部分组成:

#### (1) 更新协方差矩阵并构造增量形式的局部切空间

首先,给定一组数据点 $\{x_i | i = 1, 2, \dots, N\}$ ,局部切空间校准算法(Local tangent space alignment, LTSA)<sup>[16]</sup>假设数据点采样于一个高维空间,即它们位于  $m$  维流形  $\mathbf{R}^m$  中。该算法将  $x_i$  映射到  $d$  维( $d < m$ ),并尽量保持  $x_i$  的局部几何信息。 $x_i$  的局部几何信息由其  $k$  个近邻点构成的流形来定义,生成关于  $x_i$  的切空间<sup>[16]</sup>。在所有数据点的切空间上通过校准的方法,来得到这些点在低维流形中的全局坐标。

本文作者之前提出的基于增量切空间校准的流形学习算法<sup>[15]</sup>根据增量 PCA 来发现样本点的局部主成分,不受样本点的协方差矩阵大小的限制,从而解决了 LTSA 中大规模矩阵特征分解的问题。此外,考虑了自适应系数以后该算法能够有效处理新到达的样本点。

当加入一个新点时,用  $x_{\text{new}}$  表示。为了解决增量非线性映射问题,设  $\mathbf{M}_N$  为  $m$  维空间中现有的  $N$  个样本点的平均值: $\mathbf{M}_N = \bar{x}_i e^T, i = 1, 2, \dots, N$ 。用  $\mathbf{M}_{\text{new}}$  来表示已有样本点协方差矩阵当新点插入以后的几何信息,其中,  $\alpha$  是自适应因子,控制当新点插入已有数据集后,如何影响  $\mathbf{M}_{\text{new}}$  的估计值。

受主成分分析算法 PCA 影响,作者在之前提出的一种基于增量切空间校准的自适应特征提取算法中,构造随着新样本点加入,不断更新的协方差矩阵  $\mathbf{C}_N$  的学习过程。

为了在低维空间增量地更新所有点的嵌入坐标,从之前  $\mathbf{C}_N$  获得的特征向量和当前观测的向量构造  $d+1$  个向量:  $y_i$  和  $y_{d+1}$ 。然后可以定义一个矩阵  $\mathbf{A}: \mathbf{A} = [y_1, y_2, \dots, y_{d+1}]$ , 大小为  $k \times (d+1)$ 。可知:  $\mathbf{C}_{\text{new}} = \mathbf{A}\mathbf{A}^T$ , 表示新样本的邻域点以及它自身的局部切空间矩阵。

接下来构造一个内积矩阵  $\mathbf{B}: \mathbf{B} = \mathbf{A}\mathbf{A}^T$ , 大小为  $k \times k$ , 比  $\mathbf{C}$  小得多。这样可以增量地构造局部切空间矩阵,而不是重复计算新的协方差矩阵。基于  $\mathbf{B}$  的特征向量  $u_i$ , 即可根据局部切空间中的信息构造全局坐标。

#### (2) 根据得到的局部几何信息获取低维全局坐标

考虑增量更新的情况,本文建立一种新的自适应特征提取的增量学习模型。新点  $x_{\text{new}}$  的局部坐标可以表示成矩阵形式为  $x_{\text{new}} = w_i u_i + \bar{X}_N = \mathbf{W}\mathbf{U} + \bar{X}_N$ 。

为了局部并线性地求解增量校准矩阵的特征值和特性向量,可以基于矩阵  $\mathbf{B}$  的特征向量  $u_1, \dots, u_d$  来构造  $\mathbf{U}_i$ 。根据定义  $\mathbf{X}_n = \mathbf{W}\mathbf{U} + \bar{X}$ ,  $u_1, \dots, u_d$  也是矩阵  $\mathbf{A}$  的  $d$  个最大右单位向量,或是对应于矩阵  $\mathbf{C}_n$  的  $d$  个最大特征值的特征向量。故定义矩阵:  $\mathbf{U}_i = [e/\sqrt{k}, u_1, \dots, u_d]$ , 那么,  $\mathbf{H}_i = \mathbf{I} - \mathbf{U}_i \mathbf{U}_i^T$  包含在  $\mathbf{W}\mathbf{U} + \bar{X}$  和  $e^T/\sqrt{k}$  张成的子空间中,  $\mathbf{I}$  为单位矩阵。

得到新的数据后,更新增量校准矩阵: $\mathbf{T}(\mathbf{I}_i, \mathbf{I}_i) \leftarrow \mathbf{T}(\mathbf{I}_i, \mathbf{I}_i) + \mathbf{I} - \mathbf{U}_i \mathbf{U}_i^T$ ,然后计算矩阵  $\mathbf{T}$  的第 2~第  $(d+1)$  个最小特征值和相应的特征向量  $\mathbf{t}_2, \dots, \mathbf{t}_{d+1}$ , 对应于数据点最优的低维全局坐标。

以上方法在本文中被命名为自适应特征提取的增量学习模型,从而实现了增量特征提取的功能。

## 2.2 基于特征空间校准的增量流形学习算法模型

流形学习算法在应用于数据分类时暴露出诸多缺陷,例如小样本问题,所以在对原始的数据进行处理之前,必须采取必要的措施来解决数据可能面对的小样本问题,否则后续的特征提取方法将不能应用<sup>[13]</sup>。LTSA 由于其构建局部子空间的局限性,当面临数据集有噪声的时候,对原始数据集降维以后不能很好地保留其本质特征,同时存在小样本问题。为了解决这些问题,作者之前提出过一种增量流形学习算法,称为基于特征空间校准的增量流形学习算法 (Feature space alignment learning algorithm, FSAL)<sup>[17]</sup>。

该算法利用了 CCIPCA 的思想,避免构造和重复计算数据集的协方差矩阵,来获得局部特征空间中构造的邻域点的坐标。当一个新点加入时,为保持子空间的维度,无需放弃主特征向量中的最小值,来寻找对应于降维空间中坐标的最优特征向量。

基于以上所述,本文基于 FSAL,建立基于特征空间校准的增量流形学习算法模型,该算法模型由两个部分组成:

### (1) 利用增量无偏主成分分析进行预处理

给出一组采样于非线性流形  $\mathbf{R}^m$  并带噪声的样本数据集:  $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_N]$ , 每个  $\mathbf{x}_i (i=1, \dots, N)$  是  $m$  维向量。作为预处理,本文用增量主成分方法来构造样本数据的特征空间。构造大小为  $m \times m$  的协方差矩阵  $\mathbf{E} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i \mathbf{x}_i^T$ , 设矩阵  $\mathbf{E}$  的特征向量为  $\mathbf{v}$ 。定义  $\mathbf{w}_j = \mathbf{x}_j \mathbf{x}_j^T \mathbf{v}_j$ ,  $E_v = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i \mathbf{x}_i^T \mathbf{v}_j$  可被认为是  $\mathbf{w}_j$  的平均值,  $\mathbf{v}_j$  是  $\mathbf{v}$  的第  $j$  步估计值。

根据统计有效性理论,已知标准误差的情况下,  $\mathbf{w}_j$  的均值  $\bar{\mathbf{w}} = \frac{1}{N} \sum_{i=1}^N \mathbf{w}_j$  是  $\mathbf{x}_j \mathbf{x}_j^T \mathbf{v}_j$  的有效估计<sup>[8]</sup>。

总之,导出矩阵  $\mathbf{E}$  的协方差矩阵第  $j$  个主特征向量的估计值。令  $\mathbf{u}_j$  表示  $E_v$  的第  $j$  次估计值。为了确定  $\mathbf{u}_j$  的下一步估计,将  $\mathbf{v}_j$  正规化的值定为  $\frac{\mathbf{u}_{j-1}}{\|\mathbf{u}_{j-1}\|}$ , 即,  $\mathbf{u}_j = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i \mathbf{x}_i^T \mathbf{v}_j$ 。那么由  $E_v$  的表达式就可以导出  $\mathbf{u}_j$  的增量表达式,  $\mathbf{u}_j = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i \mathbf{x}_i^T \frac{\mathbf{u}_{j-1}}{\|\mathbf{u}_{j-1}\|}$ 。

对  $\mathbf{u}_j$  改进的估计过程可以写成递归形式,通过对递归形式的  $\mathbf{u}_j$  进行迭代循环增量地计算其  $N$  个样本的主成分,而无需构造协方差矩阵,  $i, j=1, \dots, N$ 。循环结束后,将迭代中得到的特征向量  $\mathbf{u}_i$  存入矩阵  $\mathbf{U}$  中,  $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_N]$ 。因为这些特征向量都是互相正交的,故接下来计算的特征向量位于  $\mathbf{u}_j$  的互补空间,该空间由主特征向量构成。

### (2) 构造特征空间中的低维坐标

现在考虑如何构造特征空间中的  $d$  维坐标 ( $d < m$ )。首先确定每个特征向量  $\mathbf{u}_i$  的  $k$  个最近邻。令  $\mathbf{U}_i = [\mathbf{u}_1, \dots, \mathbf{u}_i]$  为由每个  $\mathbf{u}_i$  的  $k$  个最近邻所组成的矩阵 (包含其自身, 欧式距离意义)。对矩阵  $\mathbf{U}_i^T \mathbf{U}_i$  进行特征分解,由  $\mathbf{U}_i^T \mathbf{U}_i$  的  $d$  个最大特征值对应的特征向量组成矩阵  $\mathbf{V}_i$ 。

接下来构造矩阵  $\mathbf{F} = \mathbf{V}_i \mathbf{V}_i^T$ , 并提取其特征,计算矩阵  $\mathbf{F}$  的  $d$  个最小特征向量  $\mathbf{g}_1, \dots, \mathbf{g}_d$ 。

跟 2.1 节类似,令  $\mathbf{R}_i = \mathbf{I} - \mathbf{M}_i \mathbf{M}_i^T$ ,  $\mathbf{M}_i = [e/\sqrt{k}, \mathbf{g}_1, \dots, \mathbf{g}_d]$ , 并设置  $\mathbf{R}(\mathbf{I}_i, \mathbf{I}_i)$  的表达式为  $\mathbf{R}(\mathbf{I}_i, \mathbf{I}_i) \leftarrow \mathbf{R}(\mathbf{I}_i, \mathbf{I}_i) + \mathbf{I} - \mathbf{M}_i \mathbf{M}_i^T$ 。对  $\mathbf{R}$  进行特征分解,算法的最优解由  $\mathbf{R}(\mathbf{I}_i, \mathbf{I}_i)$  第 2~第  $(d+1)$  个最小特征值对应的特征向量给出,也就是特征空间中最优低维输出坐标。

这样就获得整个数据流包括新点的最优低维 ( $d$  维) 坐标,实现了基于特征空间校准的增量特征提取的功能。

### 2.3 正则化优化框架

目前大多数机器学习算法都使用正则化损失函数的设计框架,如前所述,一个增量学习的损失函数+正则化项的一般理论框架可以描述为

$$\min_{\mathbf{w}} F(\mathbf{w}) = L(\mathbf{w}, \mathbf{x}_i) + \lambda R(\mathbf{w}, \mathbf{x}_i) \quad \text{s. t.} \quad \mathbf{w} \in \mathbf{R} \quad (3)$$

增量学习非常适合于流式数据到达的环境,特别是对数据流中蕴含的时间属性能够很好地刻画。此时的目标函数可以表达为  $L(\mathbf{w}, \mathbf{x}_i) = L(\mathbf{w}, \mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N) = \sum_{i=1}^N L(\mathbf{w}, \mathbf{x}_i)$  和  $R(\mathbf{w}, \mathbf{x}_i) = R(\mathbf{w}, \mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_d) = \sum_{i=1}^d R(\mathbf{w}, \mathbf{x}_i)$  的形式。这里,  $L$  和  $R$  两个函数都具有了增量学习的性质。则本框架中的优化函数变为

$$\min_{\mathbf{w}} F(\mathbf{w}) = \sum_{i=1}^N L(\mathbf{w}, \mathbf{x}_i) + \lambda \sum_{i=1}^d R(\mathbf{w}, \mathbf{x}_i) \quad (4)$$

根据式(1)利用增量无偏主成分分析进行预处理,在本文中用增量主成分方法来构造样本数据的特征空间作为预处理。并且  $\mathbf{w}_j$  的均值  $\bar{\mathbf{w}} = \frac{1}{N} \sum_{j=1}^N \mathbf{w}_j$  是  $\mathbf{x}_i \mathbf{x}_i^T \mathbf{v}_j$  的有效估计。给出当前观测的已知样本点后,该估计值具有最小误差方差。故本文中将  $\mathbf{x}_i \mathbf{x}_i^T \mathbf{v}_j$  的估计值作为优化框架中目标函数中的损失函数项,用在本文提出的增量学习算法优化模型中。根据 2.2 节,  $\mathbf{u}_j$  可以写成

$$\mathbf{u}_j = \omega_1 \mathbf{u}_{j-1} + \omega_2 \mathbf{x}_i \mathbf{x}_i^T \frac{\mathbf{u}_{j-1}}{\|\mathbf{u}_{j-1}\|} \quad (5)$$

式中:  $\omega_1$  和  $\omega_2$  分别为  $\mathbf{u}_{j-1}$  和新数据点的权重,为了实现统计有效性,这里设  $\omega_2$  为  $\frac{1}{N}$ ,  $\omega_1$  为  $1 - \omega_2 = \frac{N-1}{N}$ ,可加速对式(5)的估计。对  $\mathbf{u}_j$  改进的估计过程为

$$\mathbf{u}_j = \frac{N-1}{N} \mathbf{u}_{j-1} + \frac{1}{N} \mathbf{x}_i \mathbf{x}_i^T \frac{\mathbf{u}_{j-1}}{\|\mathbf{u}_{j-1}\|} \quad (6)$$

为了估计接下来的特征向量,本文在计算下 1 项之前先从数据中减去第一项的映射,形成了如下的学习过程,从而达到对目标函数的每个子函数依次进行“首尾相连”的传递式梯度优化迭代,该梯度增量算法最终能得到问题的最优解,则

$$\mathbf{x}_i^{j+1} = \mathbf{x}_i^j - (\mathbf{x}_i^j)^T \frac{\mathbf{u}_j}{\|\mathbf{u}_j\|} \frac{\mathbf{u}_j}{\|\mathbf{u}_j\|} \quad (7)$$

接下来计算的特征向量位于  $\mathbf{u}_j$  的互补空间,这些特征向量都是互相正交的,这也是本项目中用预处理步骤来构建样本点的特征空间的原因,该空间由主特征向量构成。

前面提到,优化模型中的损失函数和正则化项的不同组合模型可以衍生出不同的模型,所以本文中使用前两节中描述的不同特征提取算法和本节中得到的损失函数项结合起来,就可以产生出适应不同环境(增量自适应和切空间校准)的增量学习优化模型。由于正则化项  $R(\mathbf{w})$  用来表达问题的先验信息对该问题的约束,故 2.1 和 2.2 中描述的两个算法的最优低维输出坐标即可作为真实数据集中内蕴的结构特征、局部邻域关系和流形结构信息,构成正则化项;而本节中的主特征向量  $\mathbf{u}_i$  即可构成损失函数,可由  $\sum_{i=1}^N L(\mathbf{w}, \mathbf{u}_i)$  得到。正则化项及目标函数的构造和最小化求解方法见 2.4 节。

### 2.4 正则化增量流形学习算法步骤

本文提出的正则化增量流形学习算法步骤为

IMLA 算法:

输入 位于高维空间( $m$  维)中的数据集  $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_N]$ ,  $k$  近邻参数,降维维数  $d$ , 新样本点  $\mathbf{x}_{\text{new}}$ 。

输出 低维空间( $d$ 维)中对应的嵌入坐标 $[\mathbf{y}_1, \dots, \mathbf{y}_N], \mathbf{y}_{\text{new}}$ 。

(1) 自适应特征提取的增量学习模型

1) 为每个点  $x_i$  确定  $k$  个近邻点(欧式距离), 组成矩阵  $\mathbf{X}_i, i=1, \dots, N$ ;

2) 构造 2.1 节中的协方差矩阵  $\mathbf{C}_N$ , 矩阵  $\mathbf{A}$  和矩阵  $\mathbf{B}$ ;

3) 通过计算矩阵  $\mathbf{B}$  的  $d$  个最小的特征值对应的特征向量  $\mathbf{u}_1, \dots, \mathbf{u}_d$ , 构造  $\mathbf{U}_i = [\mathbf{e}/\sqrt{k}, \mathbf{u}_1, \dots, \mathbf{u}_d]$ ;

4) 更新增量校准矩阵:  $\mathbf{T}(\mathbf{I}_i, \mathbf{I}_i) \leftarrow \mathbf{T}(\mathbf{I}_i, \mathbf{I}_i) + \mathbf{I} - \mathbf{U}_i \mathbf{U}_i^T$ , 然后计算矩阵  $\mathbf{T}$  的第 2 到第  $(d+1)$  个最小特征值和相应的特征向量  $\mathbf{t}_2, \dots, \mathbf{t}_{d+1}$ , 对应于数据点最优的低维全局坐标 $[\mathbf{t}_2, \dots, \mathbf{t}_{d+1}]$ 。

(2) 基于特征空间校准的增量流形学习算法模型

1) 对每个点  $x_i$  进行预处理, 通过迭代循环增量地计算这  $N$  个样本的主特征向量  $\mathbf{u}_i$  并存入矩阵  $\mathbf{U}$ ,  $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_N]$ , 而无需构造协方差矩阵,  $i=1, \dots, N$ 。

2) 用步骤(1)中得到的每个特征向量  $\mathbf{u}_i$  确定其欧式距离的  $k$ -近邻  $\mathbf{u}_j$ 。令  $\mathbf{U}_i = [\mathbf{u}_i, \dots, \mathbf{u}_j]$  为包含所有  $k$ -近邻的矩阵,  $j=1, \dots, N$ 。

对矩阵  $\mathbf{U}_i^T \mathbf{U}_i$  进行特征分解, 获得  $\mathbf{U}_i^T \mathbf{U}_i$  的  $d$  个最大特征值对应的特征向量  $\mathbf{V}_i$ 。

3) 构造矩阵  $\mathbf{F} = \mathbf{V}_i \mathbf{V}_i^T$ , 并计算矩阵  $\mathbf{F}$  的  $d$  个最小特征向量  $\mathbf{g}_1, \dots, \mathbf{g}_d$ 。

通过局部地构造矩阵  $\mathbf{R}$ , 计算  $\mathbf{R}$  的第 2 到第  $(d+1)$  个最小特征值对应的特征向量, 则特征向量组成的矩阵 $[\mathbf{r}_2, \dots, \mathbf{r}_{d+1}]$ 对应算法的最优低维输出坐标。

(3) 构造正则化优化目标函数

1) 构造损失函数

①对位于高维流形中的数据集  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N \subset \mathbf{R}^m$ , 计算  $N$  个样本点的主特征向量。对原始数据样本进行预处理, 提取高维空间( $m$ 维)的主特征向量。

②对  $n=1, \dots, N$ , 通过式(6,7)的迭代循环增量地计算这  $N$  个样本的主成分(无需构造协方差矩阵)。

③循环结束后, 将上述迭代中得到的主特征向量  $\mathbf{u}_i$  存入矩阵  $\mathbf{U}, \mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_N]$ 。则损失函数可由

$$\sum_{i=1}^N L(\mathbf{w}, \mathbf{u}_i) \text{ 获得。}$$

2) 构造正则化项

2.3 节中的正则化项  $\mathbf{R}(\mathbf{w}, \mathbf{x}_i)$  分为两个部分:  $\mathbf{R} = \mathbf{R}_1 + \mathbf{R}_2$ ,  $\mathbf{R}_1$  由步骤(1)中矩阵  $\mathbf{T}$  的第 2~第  $(d+1)$  个最小特征值对应的特征向量组成:  $\mathbf{R}_1 = \sum_{i=2}^{d+1} \mathbf{R}(\mathbf{w}, \mathbf{t}_i)$ 。  $\mathbf{R}_2$  由步骤(2)中矩阵  $\mathbf{R}$  的第 2~第  $(d+1)$  个最小特征值对应的特征向量组成:  $\mathbf{R}_2 = \sum_{i=2}^{d+1} \mathbf{R}(\mathbf{w}, \mathbf{r}_i)$ 。

3) 构造目标函数

本算法中的优化函数为  $\min_{\mathbf{w}} F(\mathbf{w}) = \sum_{i=1}^N L(\mathbf{w}, \mathbf{u}_i) + \lambda(\mathbf{R}_1 + \mathbf{R}_2)$ , 通过对该式进行最小化, 得到优化后的特征系数  $\mathbf{w}$ , 则  $\mathbf{w}$  构成的解空间即为正则化后的特征解。

本文提出的算法框架具有以下优点, 可以在后文的实验部分得到证实。

(1) 算法框架中增量学习的学习原理符合数据流式到达的场景, 对数据流中蕴含的时间属性能够很好地刻画。

(2) 将增量主成分分析扩展到基于特征空间校准的增量特征提取的预处理中, 直接从上一步提取到的特征向量和新观察到的样本进行更新, 避免了迭代地更新参数, 大大节约了算法的计算复杂度。

(3) 当数据集有噪声的时候, 建立基于特征空间校准的增量流形学习算法模型, 在面对噪声环境下的高维数据流进行特征抽取时具备对噪声的鲁棒性。

## 2.5 增量学习的一些评价标准

除了在一些基准测试数据集上的预测精度等一般准则,研究者们也根据增量学习的特性提出了其他具体的评价标准,如:(1)稳定性,即在测试集上的预测精度应在每一个增量学习步骤中保持稳定;(2)提高性,即学习算法的预测精度在第  $N+i$  步应比第  $N$  步有所提高;(3)可恢复性,即学习算法即使性能下降到一定精度,也应该能够恢复到之前最好的预测精度<sup>[18]</sup>。

此外,Geng 等<sup>[18]</sup>提出了另一种常用的标准,即学习曲线。由于增量学习算法通常随着训练样本量的增加而逐渐获得知识,该算法的学习精度随着时间的推移应该有一个逐渐改善的曲线,即学习曲线。Geng<sup>[18]</sup>提到,通常学习算法在学习曲线的早期阶段是不可靠的。一般来说,一个好的学习算法应该迅速增加到一个相对稳定的水平。这是延伸上述评价标准(1)和(2)得到的结论。

## 3 实验与结果分析

因为增量学习非常适合于流式数据到达的环境,算法框架中增量学习的学习原理正好符合数据流式到达的场景,特别是对数据流中蕴含的时间属性能很好地刻画,故本文设计优化框架中目标函数中的损失函数项用在本文提出的增量学习算法优化模型中。本文提出的算法将增量主成分分析扩展到基于特征空间校准的增量特征提取的预处理中,直接处理从上一步提取到的特征向量和新观察到的样本,避免了迭代地更新参数,大大节约了算法的计算复杂度。故本文算法可以有效地推广到高维大数据集,这是经过理论证明的。本节用一系列实验结果来展示本文所提算法的性能和优势,并与其他相关算法如局部线性嵌入(Locally linear embedding, LLE)<sup>[19]</sup>、拉普拉斯映射(Laplacian eigenmaps, LE)<sup>[20]</sup>、LT-SA<sup>[16]</sup>和等距映射(Isometric mapping, ISOMAP)<sup>[21]</sup>等算法进行比较<sup>[22]</sup>。所有算法用 Matlab 实现,实验环境为一台 Core-2 2.2 GHz CPU 的 PC,具有 2 GB 存储量。

### 3.1 数据集介绍及参数设置

实验选用了 ORL<sup>[23]</sup>、Yale<sup>[24]</sup>、YaleB<sup>[25]</sup>、AR<sup>[26]</sup>、Coil-20<sup>[27]</sup> 5 种数据集。ORL、Yale、YaleB 和 AR 是人脸图像数据集,取原数据集约 50% 做训练集,其余的做测试集。Coil-20 是哥伦比亚图像数据集,包含了 20 种物品,取大约 53% 的数据集做训练集,其余的做测试集。数据详细信息如表 1 所示。

本文提出的算法与 LLE 算法、LE 算法、LTSA 算法以及 ISOMAP 算法进行对比。改进的算法以及其他算法在计算重构权值矩阵都需要查找  $k$  近邻点,由于  $k$  近邻法的  $k$  值必须小于最小同类样本点总数,故将本文提出的算法查找样本点的  $k$  近邻值设置为 8,其他算法都设置为 8。本文提出的算法降维后采用  $K$ -NN 分类器,参数  $K$  设定为 10。其他比较算法均采用  $K$ -NN 分类器,且  $K=10$ 。

为了保证数据集不偏向任何算法,每次都随机划分训练集和测试集,连续运行 10 次取平均值作为最终结果。

表 1 实验数据集描述

Tab. 1 Description of experiment data set

数据集	个体数目	图像数目/个体	图像像素	训练/测试集数量
ORL	40	10	64×64	5/5
Yale	15	11	32×32	5/6
YaleB	38	64	32×32	32/32
AR	120	25	50×40	13/12
Coil-20	20	72	64×64	38/34

### 3.2 Olivetti face 人脸表情数据库实验

为了检验算法对实际数据的降维能力,利用 Olivetti face 人脸表情数据库<sup>[23]</sup>进行分类实验,此数据



集包含 40 个人的脸部图像,每人 10 张不同表情的图像,图像大小为 64 像素 $\times$ 64 像素<sup>[22]</sup>。将这 400 张人脸图像数据集用包括本文算法的 5 种不同方法(本文提出的针对流式大数据的增量流形学习优化算法(Incremental manifold learning optimization algorithm for data streams, IMLA),LLE,LE,LTSA 及 ISOMAP 算法)分别降维至 10,20, $\dots$ ,100 维,再进行分类,并计算分类精确度,如图 1 所示。由图 1 可知,IMLA 将人脸数据降维以后的分类精确度在大多数范围内优于其他算法,并符合 Geng 等提出的增量学习的下述评价标准<sup>[22]</sup>:(1)稳定性。本文算法 IMLA 在测试集上的识别精度于不同维度上都保持了稳定。(2)提升性。本文算法 IMLA 的识别精度在迭代后的每一步都比前面的步骤有了一定的提高。

### 3.3 Yale 和带噪声的 Yale 人脸表情数据库实验

利用 Yale 人脸表情数据库进行分类实验,Yale 人脸数据集选自 Yale 计算视觉及模式识别中心<sup>[24]</sup>,包含 15 个人共计 165 幅人脸图像,每个人具有 11 幅脸部图像。这些图像包括各种面部表情(快乐、一般、不快乐、惊讶、困倦和眨眼)和不同光照条件(从左到右、中间和右到右)。所有的图像都由原来的数据集被裁剪成分辨率为 32 像素 $\times$ 32 像素的图像。

实验随机选择每个个体的 5 个图像作为训练集,其余 6 个图像作为测试集。首先,将数据集降维到从 2~100 个不同维度,再使用  $K$ -NN 分类,图 2 显示了识别率与维度的关系。随着维数的增加,所有算法的识别率都在增加。可以看到,本文算法在低于 20 维的识别精度更高。从图 2 中可以看到,本文提出的方法降维以后的分类精度在绝大部分范围都可以达到良好的性能。

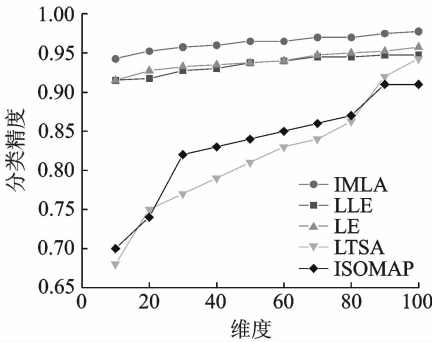


图 1 Olivetti face 人脸表情数据库实验

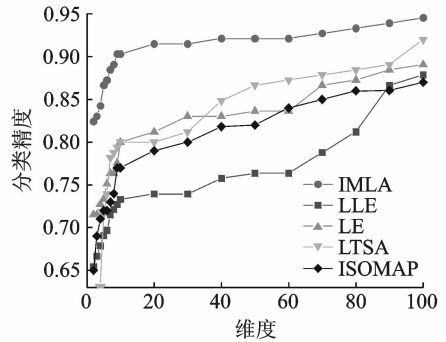


图 2 Yale 人脸图像数据集实验

Fig. 1 Experiments on Olivetti face expression database

Fig. 2 Experiments on Yale face image datasets

为了考察算法对不规则数据的效果,实验从 Yale 人脸数据库剩余的样本集中随机选取 1 人的 11 张面部表情图像,替换原训练集的一组训练样本,作为噪声样本<sup>[22]</sup>,在此情况下,IMLA 算法同其他线性算法约简到不同维数的识别率如图 3 所示。

可以看出,本文算法同其他算法约简到不同维数的识别率受噪声影响程度不同。对于含有噪声样本的训练集,IMLA 算法在本实验的各种算法中识别率最高;而对于其他算法,例如 ISOMAP 算法,由于其对图像信息的敏感性,一旦引入不规则数据,低维嵌入空间将受到较大影响<sup>[22]</sup>。

因此,本文算法在面对噪声环境下的高维数据流进行特征抽取时具备对噪声的鲁棒性,因此符合了前文中提到的增量学习算法第 3 个评价标准:可恢复性,即学习算法即使性能下降到一定精度,也能够恢复到之前最好的预测精度。从图 3 可以看出,本文算法能够迅速增加到一个相对稳定的水平。

### 3.4 YaleB 人脸图像数据库实验

扩展的 YaleB 人脸数据库<sup>[25]</sup>包含许多人脸图像。每个人都有不同的姿势、面部表情和光照条件。在这里,选择了包含 38 个人的子集,每个人有大约 64 张大小为 32 像素 $\times$ 32 像素的人脸图像。实验中,随机选取 32 个图像个体作为训练集,其余 32 个图像作为测试集。图 4 显示了识别率随着维度变化的

趋势。随着维数的增加,不同算法的识别费率均有改善。可以看到所提算法的分类精度达到最高后趋于恒定。其他算法也随维数的增加而增大,直到趋向于它们的极值。总的来说,算法在大部分维度方面效果都很好,这意味着本文的算法可以在较为广阔的范围内发现原始数据的内在结构。

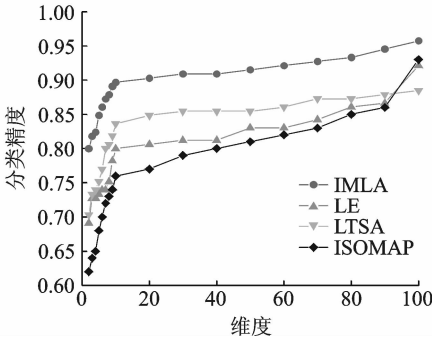


图3 带噪声 Yale 人脸图像数据集实验

Fig. 3 Experiments on Yale face image datasets with noise

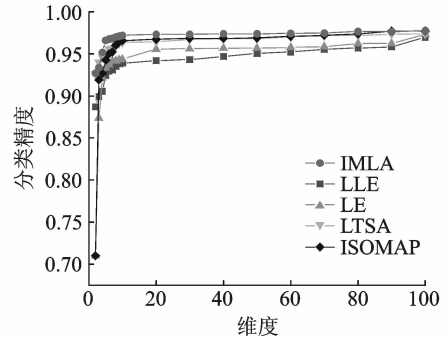


图4 YaleB 人脸图像数据集实验

Fig. 4 Experiments on YaleB face image datasets

### 3.5 AR 人脸表情数据库实验

AR 数据库<sup>[26]</sup>包含 126 人的 4 000 多张人脸图像,其中有 70 个男性和 56 个女性。每个人的图像包括面部表情变化、不同光照和遮挡的状态。这些人的照片在相隔两周的时间被获取。选择 120 个个体,每个个体的图像被裁剪为 50 像素 $\times$ 40 像素。实验中,每个人随机选取 13 张图像作为训练集,12 张图像作为测试集。将 AR 数据集约简到不同维数以后,本文算法同其他几种算法降维以后计算得到的重构误差比较如图 5 所示。

可见当维数约简到比较低时,与 LE、LTSA、ISOMAP 这几个主流降维算法相比,本文算法体现出较为明显的优势,说明它在降到较低维以后能最大程度上保持原始高维数据集的内在结构,而其他算法由于异常值的影响而迅速破坏<sup>[22]</sup>。

### 3.6 Coil-20 哥伦比亚图像数据库实验

COIL-20 哥伦比亚图像数据库<sup>[27]</sup>包含 20 个对象,每个对象在水平上旋转 360°,每隔 5°拍摄一张照片,因此每个对象共 72 幅图,大小是 64 像素 $\times$ 64 像素。降维以后的分类精度实验结果如图 6 所示。

可以看出,LLE 和 ISOMAP 这两个主流降维算法在降到 10 维以内的分类精度已经远远低于其他算法,故这些方法未能捕获由降维引起的重构误差;与之相比,本文所提出的算法在低维空间执行得非常好,异常值被滤除,这在模式识别领域具有很重要的意义和应用价值。

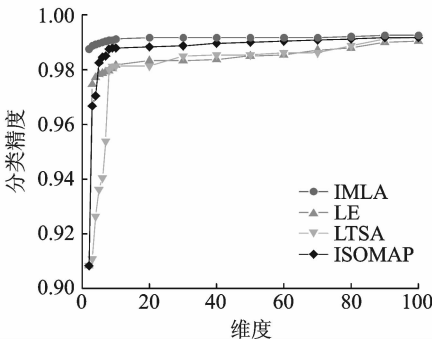


图5 AR 人脸表情数据库实验

Fig. 5 Experiments on AR face expression database

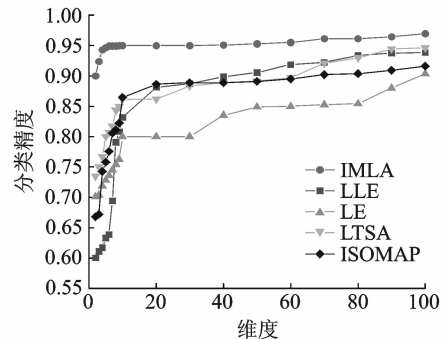


图6 COIL-20 图像数据库实验

Fig. 6 Experiments on COIL-20 Image database

### 3.7 时间复杂度分析

与 LTSA 算法相比,  $\mathbf{A}$  和  $\mathbf{B}$  这两个矩阵在本文算法的第 1 步起到重要的作用。LTSA 与本文算法的时间效率近似, 首先它们都是基于协方差矩阵来构造切空间, 然后这两个算法均是按相近的方式在  $(d+1)$  阶的矩阵上进行特征分解。矩阵  $\mathbf{A}$  的大小为  $k \times (d+1)$ , 远远小于矩阵  $\mathbf{C}$ 。接着基于矩阵  $\mathbf{B}$  的特征向量构造切空间, 并在大小为  $(d+1) \times (d+1)$  的矩阵上做特征分解, 矩阵  $\mathbf{B}$  的大小为  $k \times k$ , 故需要的时间复杂度为  $O(k(d+1))^2$ 。

在本文算法接下来的步骤中, 核心运算过程包括式(6,7)。式(6)的迭代过程节省的计算复杂度为  $(k-1)/2$ , 故可以得出式(6)的复杂度为  $(k^2+1)/2$ ,  $k$  是近邻点的数目。经过  $n$  次迭代相加之后, 计算复杂度为  $O(nk^2)$ 。式(7)的收敛速度快, 计算复杂度低。故算法总的时间复杂度可以等价于  $O(k(d+1))^2 + O(nk^2) = O(n+d^2)k^2$ 。

## 4 结束语

本文结合流式大数据流式到达的特性, 首先, 研究具有自适应特性的增量特征提取算法, 建立自适应增量特征提取算法模型。然后, 当数据集有噪声的时候, 建立基于特征空间校准的增量流形学习算法模型, 解决小样本问题。最后, 构造增量学习的“损失函数+正则化项”的优化框架, 最终得到问题的最优解。该学习算法可以随时在线更新样本点, 这对于状态可变的动态系统尤其重要。将增量主成分分析扩展到基于特征空间校准的增量特征提取的预处理中, 直接从上一步提取到的特征向量和新观察到的样本进行更新, 避免了迭代地更新参数, 大大节约了算法的计算复杂度。当数据集有噪声的时候, 建立基于特征空间校准的增量流形学习算法模型, 在面对噪声环境下的高维数据流进行特征抽取时具备对噪声的鲁棒性。实验结果表明了本文提出的算法框架符合增量学习算法的下列评价指标: 稳定性、提高性以及学习曲线能够迅速增加到一个相对稳定的水平, 以上证明了本算法优化框架的有效性。本文有待进一步研究的方面包括: (1) 使用先前的特征向量和特征值近似标准偏差; (2) 鲁棒函数的选择。这些可以有助于进一步提高性能。

### 参考文献:

- [1] Zeng X, Li G. Incremental partial least squares analysis of big streaming data[J]. Pattern Recognition, 2014, 47(11):3726-3735.
- [2] 孙大为, 张广艳, 郑纬民. 大数据流式计算: 关键技术及系统实例[J]. 软件学报, 2014, 25(4): 839-862.  
Sun Dawei, Zhang Guangyan, Zheng Weimin. Big data stream computing: Technologies and instances[J]. Journal of Software, 2014, 25(4): 839-862.
- [3] 潘志松, 唐斯琪, 邱俊洋, 等. 在线学习算法综述[J]. 数据采集与处理, 2016, 31(6): 1067-1082.  
Pan Zhisong, Tang Siqi, Qiu Junyang, et al. Survey on online learning algorithms[J]. Journal of Data Acquisition and Processing, 2016, 31(6): 1067-1082.
- [4] 张长水, 张见闻. 演化数据的学习[J]. 计算机学报, 2013, 36(2): 310-316.  
Zhang Changshui, Zhang Jianwen. Learning on time-evolving data[J]. Chinese Journal of Computers, 2013, 36(2): 310-316.
- [5] 张钢, 谢晓珊, 黄英, 等. 面向大数据流的半监督在线多核学习算法[J]. 智能系统学报, 2014, 9(3): 355-363.  
Zhang Gang, Xie Xiaoshan, Huang Ying, et al. An online multi-kernel learning algorithm for big data[J]. CAAI Transactions on Intelligent Systems, 2014, 9(3): 355-363.
- [6] 孙大为. 大数据流式计算: 应用特征和技术挑战[J]. 大数据, 2015, 3(2): 99-105.  
Sun Dawei. Big data stream computing: Features and challenges[J]. Big Data Research, 2015, 3(2): 99-105.
- [7] 王桂玲, 韩燕波, 张仲妹, 等. 基于云计算的流数据集成与服务[J]. 计算机学报, 2017, 40(1): 107-125.  
Wang Guiling, Han Yanbo, Zhang Zhongmei, et al. Cloud-based integration and service of streaming data[J]. Chinese Journal of Computers, 2017, 40(1): 107-125.
- [8] Wang X. A summary of LDA, PCA and relative work[J]. Journal of the Graduates Sun Yat-Sen University: Natural Sci-

ences, *Medicine*, 2007, 28(4): 50-61.

- [9] Rosipal R, Kramer N. Overview and recent advances in partial least squares[C]//International conference on Subspace, Latent Structure and Feature Selection. Heidelberg, Berlin; Springer Press, 2006: 34-51.
- [10] Weng J Y, Zhang Y L, Hwang W S. Candid covariance-free incremental principal component analysis[J]. *IEEE Trans on Pattern Analysis and Machine Intelligence*, 2003, 25(8): 1034-1040.
- [11] 李焕哲, 吴志健, 汪慎文, 等. 协方差矩阵自适应演化策略学习机制综述[J]. *电子学报*, 2017, 45(1): 238-245.  
Li Huanzhe, Wu Zhijian, Wang Shenwen, et al. The overview of learning mechanism of covariance matrix adaptation evolution strategy[J]. *Acta Electronica Sinica*, 2017, 45(1): 238-245.
- [12] Chu D, Liao L, Ng K, et al. Incremental linear discriminant analysis: A fast algorithm and comparisons[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2015, 26(11): 2716-2735.
- [13] 李波. 基于流形学习的特征提取方法及其应用研究[D]. 合肥: 中国科学技术大学, 2008.
- [14] Chen M, Li W, Zhang W, et al. Dimensionality reduction with generalized linear models[C]//Proceedings of the International Joint Conference on Artificial Intelligence. San Jose, CA, USA: IEEE Computer Society Press, 2013: 1267-1272.
- [15] Tan C, Ji G. A manifold learning algorithm based on incremental tangent space alignment[C]//International Conference on Cloud Computing and Security. Heidelberg, Berlin; Springer Press, 2016: 541-552.
- [16] Zhang Z Y, Zha H Y. Principal manifolds and nonlinear dimensionality reduction via tangent space alignment[J]. *SIAM Journal of Scientific Computing*, 2004, 26(1): 313-338.
- [17] Tan C, Guan J. A feature space alignment learning algorithm [C]//Pacific Rim International Conference on Artificial Intelligence. Heidelberg, Berlin; Springer Press, 2012: 795-800.
- [18] Geng X, Smith-Miles K. Encyclopedia of biometrics[M]. New York; Springer, 2015: 912-917.
- [19] Roweis S, Saul L. Nonlinear dimensionality reduction by locally linear embedding[J]. *Science*, 2000, 290(5500): 2323-2326.
- [20] Belkin M, Niyogi P. Laplacian eigenmaps for dimensionality reduction and data representation[J]. *Neural Computation*, 2003, 15(6): 1373-1396.
- [21] Tenenbaum J, Silva de V, Langford J. A global geometric framework for nonlinear dimensionality reduction[J]. *Science*, 2000, 290(5500): 2319-2323.
- [22] 谈超, 关侗红, 周水庚. 基于等角映射的多样本增量流形学习算法[J]. *模式识别与人工智能*, 2014, 27(2): 127-133.  
Tan Chao, Guan Jihong, Zhou Shuigeng. Multi-sample incremental manifold learning algorithm based on isogonal mapping [J]. *Pattern Recognition and Artificial Intelligence*, 2014, 27(2): 127-133.
- [23] Roweis S. Research: Data for MATLAB [EB/OL]. <http://www.cs.nyu.edu/~roweis/data.html>, 2017-08-06.
- [24] Yale University. Yale face database [EB/OL]. <http://cvc.cs.yale.edu/cvc/projects/yalefaces/yalefaces.html>, 2017-08-06.
- [25] Li B, Li J, Zhang X. Nonparametric discriminant multi-manifold learning for dimensionality reduction[J]. *Neurocomputing*, 2015, 152: 121-126.
- [26] Martinez A, Benavente R. The AR face database[R]. Computer Vision Center, Technical Report. Barcelona, Spain; [s. n.], 2007, 3: 5.
- [27] Nene S, Nayar S, Murase H. Columbia object image library (COIL-20) [R]. Technical Report CUCS-005-96. New York: Columbia University, 1996.

#### 作者简介:



谈超 (1983-), 女, 博士, 讲师, 研究方向: 机器学习和数据挖掘, E-mail: tutu\_tanchao@163.com。



吉根林 (1964-), 男, 教授, 博士生导师, 研究方向: 数据挖掘及其应用。



赵斌 (1978-), 男, 博士, 副教授, 研究方向: 数据挖掘及其应用。

