文章编号: 1001-0920 (2011) 10-1511-04

基于 greedy GDA 的训练数据减少和非线性特征提取方法

刘小芳[1,2], 何彬彬[1], 李小文[1]

(1. 电子科技大学 地表空间信息技术研究所，成都 611731；2. 四川理工学院 计算机学院，四川 自贡 643000)

摘 要: 针对大数据集情况下标准广义判别分析 (GDA) 方法进行非线性特征提取时计算复杂度较高的问题, 提出了基于 GGDA (greedy GDA) 的训练数据减少和非线性特征提取方法. 该方法用 greedy 核主成分分析方法的 greedy 技术对训练数据选取子集; 然后用 GDA 方法对子集而不是全部训练数据训练特征提取模型; 并用几种特征提取的数据进行分类对比实验. 实验结果表明, GGDA 和 GDA 方法的特征提取性能优于其他对比方法, GGDA 方法不仅较好地保持了 GDA 方法的特征提取性能, 而且减少了大数据集进行非线性特征提取的计算复杂度.

关键词: greedy 广义判别分析；greedy 核主成分分析；训练数据减少；非线性特征提取；核矩阵；分类

中图分类号: TP391 **文献标识码:** A

Greedy GDA method for training data reduction and nonlinear feature extraction

LIU Xiao-fang[1,2], *HE Bin-bin*[1], *LI Xiao-wen*[1]

(1. Institute of Geo-spatial Information Science and Technology，University of Electronic Science and Technology of China，Chengdu 611731，China；2. Department of Computer Science and Technology，Sichuan University of Science and Engineering，Zigong 643000，China. Correspondent：LIU Xiao-fang，E-mail：lxf1969@163.com)

Abstract: Nonlinear feature extraction using standard generalized discriminant analysis(GDA) has high computational complexity in large datasets. Therefore, a greedy GDA(GGDA) is proposed to reduce training data and deal with the nonlinear feature extraction problem. Firstly, a subset is selected from the full training data by using the greedy technique of the greedy KPCA(GKPCA) method. Then, the feature extraction model is trained by using the GDA method with the subset instead of the full training data. Finally, classification experiments using data of several feature extraction methods are performed. The simulation results show that the feature extraction performance of both the GGDA and the GDA methods outperform that of other methods. In addition of retaining the performance of the GDA method, the GGDA method reduces the computational complexity of the nonlinear feature extraction in large datasets.

Key words: greedy generalized discriminant analysis；greedy kernel principal component analysis；training data reduction；nonlinear feature extraction；kernel matrix；classification

1 引 言

近年来, 核理论成为机器学习和模式识别的一个快速发展分支. 核理论利用满足 Mercer 条件的核函数巧妙地推导出线性算法的非线性形式[1], 如: 核主成分分析 (KPCA)[2]和广义判别分析 (GDA)[3]是通过核理论分别对主成分分析 (PCA) 和线性判别分析 (LDA) 方法进行非线性扩展而形成的, 显示了较强的非线性特征提取能力.

GDA 和 KPCA 方法应用核理论能够较好地提取非线性特征. 但两种方法进行特征提取时均需要计算和存储核矩阵, 并对核矩阵进行特征值分解. 核矩阵的维数等于样本数的平方, 核矩阵所需的计算时间和内存与输入空间的维数基本无关, 而与样本数目密切相关, 随着样本数目的增加, 内存与计算量存在问题, 使问题的求解更加困难, 甚至无法实现[4-5]. 因此, 有必要减少大数据集训练数据. Franc等人[6]在 GKPCA (greedy KPCA) 方法中提出用 greedy 技术寻找一个低维的数据表示近似再生核 Hilbert 空间的大数据集. 类似于 KPCA 方法, GKPCA 方法的目的是最小化近似数据的均方重构误差; 不同之处在于,

收稿日期: 2010-06-01；**修回日期:** 2010-10-07.

基金项目: 国家自然科学基金项目(40701146)；国家 863 计划项目(2007AA12Z227).

作者简介: 刘小芳(1969−), 女, 副教授, 博士生, 从事智能信息处理、模式识别等研究；李小文(1947−), 男, 中国科学院院士, 教授, 博士生导师, 从事定量遥感反演等研究.

GKPCA 方法用于表示数据的低维空间的基向量是从训练数据集中选取的向量, 而不是全部训练数据的线性组合, 并找出了描述原始数据的简单模型. 对比 KPCA 方法, GKPCA 方法根据重构误差找到了数据表示的次优解, 该方法可用于降低核方法的计算量和存储需求, 同时也可以用于降低用核方法进行学习的复杂度, 还能进行在线特征提取. 由于 GKPCA 方法的计算以内积的形式出现, 可以引入核方法解决非线性问题.

KPCA 和 GDA 方法均是将输入空间的数据映射到具有非线性投影方向的再生核 Hilbert 特征空间. KPCA 方法是提取互不相关的特征向量, 对于分类问题而言不一定是最佳的. 而 GDA 方法则考虑了类内与类间的信息, 这种特征提取方法对于分类问题而言是较好的[7]. 因此, 本文将 GKPCA 方法中提取子集的 greedy 技术与 GDA 方法相结合, 形成 GGDA (Greedy GDA) 方法, 用于训练数据的减少和非线性特征提取. GGDA 方法是在使用 GDA 方法之前, 先用 GKPCA 方法中提取子集的 greedy 技术对训练数据选取子集, 子集能较好地表示原始数据, 用子集而不是全部训练数据训练特征提取模型.

2 Greedy GDA 方法

2.1 greedy 技术对训练数据的减少

GKPCA 方法是标准 KPCA 的一种有效实现. 给定训练数据集 $\boldsymbol{X}=\{\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_N\}\subset\boldsymbol{R}^q$, GKPCA 方法的目的是从训练数据集 $\boldsymbol{X}$ 中找到一个子集 $\boldsymbol{X}_S=\{\boldsymbol{s}_1,\boldsymbol{s}_1,\boldsymbol{s}_2,\cdots,\boldsymbol{s}_n\}\subset\boldsymbol{R}^q$ (即 $\boldsymbol{X_S}\subseteq\boldsymbol{X}$), 使 $\mathrm{span}(\boldsymbol{X_S})\approx\mathrm{span}(\boldsymbol{X})$. $\boldsymbol{I}=\{i_1,i_2,\cdots,i_N\}$ 是训练数据集 $\boldsymbol{X}$ 的索引矩阵, $\boldsymbol{J}=\{j_1,j_2,\cdots,j_n\}$ 是子集 $\boldsymbol{X_S}$ 的索引矩阵, 且 $\boldsymbol{J}\subseteq\boldsymbol{I}$.

GKPCA 方法是在再生核 Hilbert 空间实现, 训练数据集 $\boldsymbol{X}$ 通过一个非线性映射 $\boldsymbol{\Phi}$ 变换到特征空间 $\boldsymbol{F}$. $\{\boldsymbol{\Phi}(x_1),\ \boldsymbol{\Phi}(x_2),\ \cdots,\ \boldsymbol{\Phi}(x_N)\}$ 和 $\{\boldsymbol{\Phi}(s_1),\ \boldsymbol{\Phi}(s_2),\ \cdots,\ \boldsymbol{\Phi}(s_n)\}$ 分别是训练数据集 $\boldsymbol{X}$ 和子集 $\boldsymbol{X_S}$ 在特征空间的非线性映射. GKPCA 方法是在特征空间 $\boldsymbol{F}$ 中找到一种新的核扩展, 能较好地近似原始的核扩展, 即 $\mathrm{span}(\boldsymbol{\Phi}(\boldsymbol{X_S}))\approx\mathrm{span}(\boldsymbol{\Phi}(\boldsymbol{X}))$. 原始训练数据 $\boldsymbol{X}$ 在近似特征空间表示为

$$\tilde{\boldsymbol{\Phi}}(x_i)=\sum_{j\in\boldsymbol{J}}[\boldsymbol{\beta}_i]_j\,\boldsymbol{\Phi}(x_j),\ i\in\boldsymbol{I},\tag{1}$$

其中 $\boldsymbol{\beta}_i\in R^n$ 是一组系数. 训练数据集 $\boldsymbol{X}$ 在特征空间的近似表示 $\{\tilde{\boldsymbol{\Phi}}(x_1),\tilde{\boldsymbol{\Phi}}(x_2),\cdots,\tilde{\boldsymbol{\Phi}}(x_N)\}$ 为原始数据空间子集 $\boldsymbol{X_S}$ 在特征空间 $\{\boldsymbol{\Phi}(s_1),\boldsymbol{\Phi}(s_2),\cdots,\boldsymbol{\Phi}(s_n)\}$ 内的线性组合.

寻找近似的核扩展问题可以表示为最优化问题. 当子集 $\boldsymbol{X_S}$ 的样本数 n 很小时, 最小化均方重构误差

$$\varepsilon_{\mathrm{MS}}(\boldsymbol{X}|\boldsymbol{J})=\frac{1}{N}\sum_{i\in\boldsymbol{I}}\left\|\boldsymbol{\Phi}(\boldsymbol{x}_i)-\sum_{j\in\boldsymbol{J}}[\boldsymbol{\beta}_i]_j\boldsymbol{\Phi}(\boldsymbol{x}_j)\right\|^2.\tag{2}$$

训练数据集 $\boldsymbol{X}$ 的重构误差 $\varepsilon_{\mathrm{MS}}(\boldsymbol{X}|\boldsymbol{J})$ 依赖于子集 $\boldsymbol{X_S}$ 和系数 $\boldsymbol{\beta}_i\in R^n(i\in\boldsymbol{I})$. 给定子集 $\boldsymbol{X_S}$, 系数 $\boldsymbol{\beta}_i$ 通过最小化均方重构误差 $\varepsilon_{\mathrm{MS}}(\boldsymbol{X}|\boldsymbol{J})$ 得到, 即

$$\boldsymbol{\beta}_i=\arg\min_{\boldsymbol{\beta}\in R^n}\varepsilon_{\mathrm{MS}}(\boldsymbol{X}|\boldsymbol{J}),\ i\in\boldsymbol{I}.\tag{3}$$

为了避免训练数据集 $\boldsymbol{X}$ 通过非线性映射函数 $\boldsymbol{\Phi}(\boldsymbol{x}_i)$ 显式地映射到特征空间 $\boldsymbol{F}$, 用核函数重新表示式 (2) 为

$$\varepsilon_{\mathrm{MS}}(\boldsymbol{X}|\boldsymbol{J})=\frac{1}{N}\sum_{i\in\boldsymbol{I}}(k(\boldsymbol{x}_i,\boldsymbol{x}_i)-2\boldsymbol{K_S}\boldsymbol{k_S}(\boldsymbol{x}_i)+\langle\boldsymbol{k_S}(\boldsymbol{x}_i),\boldsymbol{K_S}\boldsymbol{k_S}(\boldsymbol{x}_i)\rangle).$$

其中: $\boldsymbol{K_S}\in R^{n\times n}$ 是子集 $\boldsymbol{X_S}$ 的核矩阵; $\boldsymbol{k_S}(\boldsymbol{x}_i)=[k(\boldsymbol{x}_{j_1},x_i),k(\boldsymbol{x}_{j_2},\boldsymbol{x}_i),\cdots,k(\boldsymbol{x}_{j_n},\boldsymbol{x}_i)]^{\mathrm{T}}\in R^n$ 是样本 $\boldsymbol{x}_i$ 映射到子集 $\boldsymbol{X_S}$ 的特征空间 $\boldsymbol{F}$ 的投影.

GKPCA 方法需要从训练数据集 $\boldsymbol{X}$ 寻找最佳的子集 $\boldsymbol{X_S}$, 即从 $\boldsymbol{X}$ 的索引矩阵 $\boldsymbol{I}$ 中找出子集 $\boldsymbol{X_S}$ 的索引矩阵 $\boldsymbol{J}$, 有

$$\boldsymbol{J}=\arg\min_{\boldsymbol{J}\subseteq\boldsymbol{I}}\varepsilon_{\mathrm{MS}}(\boldsymbol{X}|\boldsymbol{J}),\tag{4}$$

且 $\mathrm{card}(\boldsymbol{J})=n$. 如果给定了子集 $\boldsymbol{X_S}$ 的样本数 n, 通过最小化式 (4) 求解子集 $\boldsymbol{X_S}$ 的问题有 $\binom{N}{n}$ 组合, 这是不切实际的. 文献 [8] 提出求解最小化 $\varepsilon_{\mathrm{MS}}(\boldsymbol{X}|\boldsymbol{J})$ 上界, 即

$$\varepsilon_{\mathrm{MS}}(\boldsymbol{X}|\boldsymbol{J})\leqslant\frac{1}{N}(N-n)\sum_{i\in\boldsymbol{I}\backslash\boldsymbol{J}}\max||\boldsymbol{\Phi}(\boldsymbol{x}_i)-\tilde{\boldsymbol{\Phi}}(\boldsymbol{x}_i)||^2,\tag{5}$$

其中 $(N-n)$ 为尺度系数, 是由于子集 $\boldsymbol{X_S}$ 里的 n 个样本有零表示误差. 式 (5) 一定有效, 因为均方重构误差 $\varepsilon_{\mathrm{MS}}(\boldsymbol{X}|\boldsymbol{J})$ 不高于最大均值误差. 通过 greedy 技术实现式 (5), 具体实现过程见文献 [8], 每次迭代选择一个最大误差的样本进入子集 $\boldsymbol{X_S}$.

2.2 GDA 方法的非线性特征提取

GDA 方法是将原始训练数据通过非线性映射 $\boldsymbol{\Phi}$ 变换到特征空间 $\boldsymbol{F}$ 中, 然后在特征空间 $\boldsymbol{F}$ 中完成 LDA 变换. 这里, GDA 方法的原始训练数据为通过 GKPCA 方法中的 greedy 技术从训练数据集 $\boldsymbol{X}$ 中选取的子集 $\boldsymbol{X_S}$. 为了便于理解, 将子集 $\boldsymbol{X_S}$ 重新定义为 $\boldsymbol{T_{XY}}=\{(\boldsymbol{x}_1,y_1),(\boldsymbol{x}_2,y_2),\cdots,(\boldsymbol{x}_n,y_n)\}$, $\boldsymbol{x}_i\in\boldsymbol{R}^q$, $y\in\boldsymbol{Y}=\{1,2,\cdots,c\}$. GDA 方法的训练数据 $\boldsymbol{T_{XY}}$ 通过非线性映射函数 $\boldsymbol{\Phi}:\boldsymbol{X}\to\boldsymbol{F}$, 在特征空间 $\boldsymbol{F}$ 中对应 $\boldsymbol{T_{\Phi Y}}=\{(\boldsymbol{\Phi}(\boldsymbol{x}_1),y_1),(\boldsymbol{\Phi}(\boldsymbol{x}_2),y_2),\cdots,(\boldsymbol{\Phi}(\boldsymbol{x}_n),y_n)\}$. 训练数据集 $\boldsymbol{T_{\Phi Y}}$ 在特征空间 $\boldsymbol{F}$ 中的类内离散矩阵 $\boldsymbol{S}_w$ 和类间离散矩阵 $\boldsymbol{S}_b$ 分别为

$$\boldsymbol{S}_w = \frac{1}{n}\sum_{i=1}^{c}\sum_{j=1}^{n_i}(\boldsymbol{\Phi}(x_{ij})-\boldsymbol{\mu}_i)(\boldsymbol{\Phi}(x_{ij})-\boldsymbol{\mu}_i)^{\mathrm{T}}, \quad (6)$$

$$\boldsymbol{S}_b = \frac{1}{n}\sum_{i=1}^{c}\frac{n_i}{n}(\boldsymbol{\mu}_i-\boldsymbol{\mu})(\boldsymbol{\mu}_i-\boldsymbol{\mu})^{\mathrm{T}}. \quad (7)$$

其中: n_i为第i类训练样本的个数,$\sum_{i=1}^{c} n_i = n$; $\boldsymbol{\Phi}(\boldsymbol{x}_{ij})$ 为特征空间 $\boldsymbol{F}$ 中第 i 类第 j 个训练样本; $\boldsymbol{\Phi}(\boldsymbol{x}_j)$ 为特征空间 $\boldsymbol{F}$ 中第 j 个训练样本; $\boldsymbol{\mu}_i = \frac{1}{n_i}\sum_{j=1}^{n_i}\boldsymbol{\Phi}(\boldsymbol{x}_j)$ 为特征空间 $\boldsymbol{F}$ 中第 i 类训练样本的均值; $\boldsymbol{\mu} = \frac{1}{n}\sum_{j=1}^{n}\boldsymbol{\Phi}(\boldsymbol{x}_j)$ 为特征空间 $\boldsymbol{F}$ 中全部训练样本的均值.

GDA 的目的在于找到一组向量 $\boldsymbol{u} = [\boldsymbol{u}_1, \boldsymbol{u}_2, \cdots, \boldsymbol{u}_n]$, 使类间离散矩阵 $\boldsymbol{S}_b$ 和类内离散矩阵 $\boldsymbol{S}_w$ 之比达到最大, 即

$$\boldsymbol{u} = \arg\max_{\boldsymbol{u}} \frac{|\boldsymbol{u}^{\mathrm{T}}\boldsymbol{S}_b\boldsymbol{u}|}{|\boldsymbol{u}^{\mathrm{T}}\boldsymbol{S}_w\boldsymbol{u}|}. \quad (8)$$

式 (8) 的解实际是找到矩阵 $\boldsymbol{S}_w^{-1}\boldsymbol{S}_b$ 的特征向量 $\boldsymbol{u}$ 对应的最大特征值 λ[9], 即

$$\boldsymbol{S}_w^{-1}\boldsymbol{S}_b\boldsymbol{u} = \lambda\boldsymbol{u}. \quad (9)$$

最大化商 $\boldsymbol{J}(\boldsymbol{u}) = |\boldsymbol{u}^{\mathrm{T}}\boldsymbol{S}_b\boldsymbol{u}|/|\boldsymbol{u}^{\mathrm{T}}\boldsymbol{S}_w\boldsymbol{u}|$, 即对 $\boldsymbol{u}$ 求导数, 有

$$\frac{\partial \boldsymbol{J}(\boldsymbol{u})}{\partial \boldsymbol{u}} = 0 \Rightarrow \boldsymbol{S}_w^{-1}\boldsymbol{S}_b\boldsymbol{u} = \Big(\frac{\boldsymbol{u}^{\mathrm{T}}\boldsymbol{S}_b\boldsymbol{u}}{\boldsymbol{u}^{\mathrm{T}}\boldsymbol{S}_w\boldsymbol{u}}\Big)\boldsymbol{u} \Rightarrow$$

$$\lambda = \Big(\frac{\boldsymbol{u}^{\mathrm{T}}\boldsymbol{S}_b\boldsymbol{u}}{\boldsymbol{u}^{\mathrm{T}}\boldsymbol{S}_w\boldsymbol{u}}\Big). \quad (10)$$

特征向量 $\boldsymbol{u} \in \boldsymbol{F}$ 在 $\boldsymbol{\Phi}(x_{ij})$ 张成的空间里, 存在系数 $\boldsymbol{\alpha} = (\boldsymbol{\alpha}_{ij})_{i=1,2,\cdots,c,\ j=1,2,\cdots,n_i}$, 使得

$$\boldsymbol{u} = \sum_{i=1}^{c}\sum_{j=1}^{n_i}\boldsymbol{\alpha}_{ij}\boldsymbol{\Phi}(\boldsymbol{x}_{ij}). \quad (11)$$

求解特征向量 $\boldsymbol{u}$ 可转化为求相应的系数 $\boldsymbol{\alpha}$. 为了能够应用核函数 $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = k_{ij} = \boldsymbol{\Phi}^{\mathrm{T}}(\boldsymbol{x}_i)\boldsymbol{\Phi}(\boldsymbol{x}_j)$ 使训练数据集 $\boldsymbol{X}$ 通过非线性映射函数 $\boldsymbol{\Phi}$ 隐式地映射到特征空间, 用 $\boldsymbol{\Phi}^{\mathrm{T}}(\boldsymbol{x}_{ij})$ 乘以式 (9), 得到

$$\boldsymbol{\Phi}^{\mathrm{T}}(\boldsymbol{x}_{ij})\boldsymbol{S}_w^{-1}\boldsymbol{S}_b\boldsymbol{u} = \lambda\boldsymbol{\Phi}^{\mathrm{T}}(\boldsymbol{x}_{ij})\boldsymbol{u}. \quad (12)$$

式 (12) 和 (9) 有相同的特征向量[2]. 将式 (11) 代入 (12) 得

$$\lambda = \Big(\frac{\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{K}\boldsymbol{W}\boldsymbol{K}\boldsymbol{\alpha}}{\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{K}\boldsymbol{K}\boldsymbol{\alpha}}\Big). \quad (13)$$

其中

$$\boldsymbol{K} = (\boldsymbol{K}_{pq})_{p=1,2,\cdots,c,\ q=1,2,\cdots,c}, \quad (14)$$

$$\boldsymbol{W} = (\boldsymbol{W}_i)_{i=1,2,\cdots,c}. \quad (15)$$

核矩阵 $\boldsymbol{K}$ 是一个分块矩阵, 其子矩阵 $\boldsymbol{K}_{pq} = (k_{ij})_{p=1,2,\cdots,n_p,\ q=1,2,\cdots,n_q}$ 由第 p 类和第 q 类的数据经过内积 $(k_{ij})_{pq} = \boldsymbol{\Phi}^{\mathrm{T}}(\boldsymbol{x}_{pi})\boldsymbol{\Phi}(\boldsymbol{x}_{qj})$ 得到, $\boldsymbol{K}_{pq}$ 是 $n_p \times n_q$ 矩阵, $\boldsymbol{K}$ 是 $n \times n$ 的对称矩阵. 系数矩阵 $\boldsymbol{W}$ 是一个 $n \times n$ 的分块对角阵, 其子矩阵 $\boldsymbol{W}_i$ 是所有元素为 $1/n_i$ 的 $n_i \times n_i$ 矩阵.

对核矩阵 $\boldsymbol{K}$ 进行特征值分解, 求出 $\boldsymbol{\alpha}$, 具体求解过程见文献 [3]. 通过归一化 $\boldsymbol{\alpha}$ 使特征向量 $\boldsymbol{u}$ 归一化, 即将式 (11) 代入 $\boldsymbol{u}^{\mathrm{T}}\boldsymbol{u} = 1$, 得到

$$\boldsymbol{u}^{\mathrm{T}}\boldsymbol{u} = \boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{K}\boldsymbol{\alpha} = 1. \quad (16)$$

系数 $\boldsymbol{\alpha}$ 除以 $\sqrt{\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{K}\boldsymbol{\alpha}}$, 得到归一化的特征向量 $\boldsymbol{u}$. 测试数据 $\boldsymbol{z}$ 在特征向量 $\boldsymbol{u}$ 上的投影为

$$\boldsymbol{u}^{\mathrm{T}}\boldsymbol{\Phi}(\boldsymbol{z}) = \sum_{i=1}^{c}\sum_{j=1}^{n_i}\boldsymbol{\alpha}_{ij}\,\boldsymbol{\Phi}(\boldsymbol{x}_{ij})\boldsymbol{\Phi}(\boldsymbol{z}) =$$

$$\sum_{i=1}^{c}\sum_{j=1}^{n_i}\boldsymbol{\alpha}_{ij}\,k(\boldsymbol{x}_{ij}, \boldsymbol{z}). \quad (17)$$

算法要求非线性映射函数 $\boldsymbol{\Phi}(\boldsymbol{x}_j)$ 中心化, 由于 $\boldsymbol{\Phi}(\boldsymbol{x}_j)$ 是隐式映射到特征空间 $\boldsymbol{F}$, 其中心化是通过中心化特征空间 $\boldsymbol{F}$ 的核矩阵 $\boldsymbol{K}$ 实现的[3].

2.3 GGDA 方法的实现步骤

GGDA 方法的训练数据减少和非线性特征提取的实现步骤如下:

1) 通过 GKPCA 方法中的 greedy 技术实现式 (5), 从训练数据集 $\boldsymbol{X}$ 中选取子集 $\boldsymbol{X_S}$ 作为 GDA 方法的训练数据 $\boldsymbol{T_{XY}}$;

2) 用式 (14) 和 (15) 分别计算核矩阵 $\boldsymbol{K}$ 和系数矩阵 $\boldsymbol{W}$;

3) 在特征空间对核矩阵 $\boldsymbol{K}$ 进行特征值分解, 计算其特征值并标准化特征向量;

4) 找出最大特征值 λ 和对应的特征向量 $\boldsymbol{u}$;

5) 根据式 (17) 计算测试样本 $\boldsymbol{z}$ 在特征向量 $\boldsymbol{u}$ 上的投影.

对比 GDA 方法中核矩阵 $\boldsymbol{K}$ 的大小 $(N \times N)$, GGDA 方法核矩阵 $\boldsymbol{K}$ 的大小为 $(n \times n)$; GDA 方法和 GGDA 方法的时间复杂度分别为 $\boldsymbol{O}(N^3)$ 和 $\boldsymbol{O}(Nn^2)$; 当 $n \ll N$ 时, GGDA 方法大大降低了 GDA 方法的内存需求和计算时间.

3 实验结果及分析

实验平台为 Intel 酷睿 2 双核 T6500 2.10 GHz 处理器, 2 G 内存, Windows XP 操作系统. 实验算法用 C 和 Matlab 混合编程实现. 实验数据选取 UCI数据库数据集 Iris 和 pendigits[10]. Iris 数据个数较少, pendigits 数据个数较多. 对于数据集 Iris, 数据特点为一类数据与其他类数据离得较远, 另外两类数据离得较近, 而且有部分数据交叠. 数据集 pendigits 为 0∼9 数字的钢笔字识别数据库. 表 1 给出了数据集的描述.

为了验证 GGDA 方法的性能, 对于几种特征提取方法提取的特征数据, 用 k- 最近邻分类算法[11]进

表 1 数据集的描述

数据集	样本数	类别	属性
Iris	150	3	4
pendigits	10992	9	16

行分类对比分析. k-近邻分类器(kNN)是从测试样本 $\boldsymbol{x}_i$ 开始生长, 不断地扩大区域, 直到包含进 k 个距离最近的训练样本为止, 并将测试样本 $\boldsymbol{x}_i$ 的类别归为最近的 k 个训练样本中出现频率最大的类别. 这里用欧氏距离作为 k-近邻分类器设计中衡量样本之间距离的度量函数.

对于给定的训练数据集 $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N\} \subset \boldsymbol{R}^q$, 通过GKPCA方法的greedy技术从训练数据集 $\boldsymbol{X}$ 中找到子集 $\boldsymbol{X}_S = \{\boldsymbol{s}_1, \boldsymbol{s}_2, \cdots, \boldsymbol{s}_n\} \subset \boldsymbol{R}^q$. LDA和PCA方法从 $\boldsymbol{X}$ 中最多提取 q 个特征; GDA和KPCA方法从 $\boldsymbol{X}$ 中最多提取 N 个特征; GGDA和GKPCA方法从 $\boldsymbol{X}_S$ 中最多提取 n 个特征, 有 $q < n \leqslant N$. 本文主要目的是验证子集 $\boldsymbol{X}_S$ 的非线特征提取性能, 没有考虑不同特征个数对分类的影响, 因此, 分类实验统一选取变换后的 q 个特征.

GGDA, GDA, GKPCA和KPCA方法的核函数取高斯核函数 $\boldsymbol{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-||\boldsymbol{x}_i - \boldsymbol{x}_j||^2/(2\sigma^2))$. 通过5-重交叉验证法[12], 对于数据集Iris和pendigits, k-最近邻分类算法的 k 分别取9和3; 核参数 σ 分别取4和60; 用GKPCA方法提取的子集 $\boldsymbol{X_S}$ 的样本数 n 分别为17和200. 分类结果用训练分类器的时间 T_c, 训练数据识别率以及测试数据的识别率进行了评价. 对于Iris和pendigits数据, 不同特征提取方法得到的特征提取数据用 k-近邻分类算法分类的结果分别如表2和表3所示, 表中的评价结果是利用5-重交叉验证法得到的平均值.

表 2 几种方法对Iris数据的分类性能比较

方法	$T_c(s)$	识别率/%	
		训练	测试
kNN	0.009 050	93.33	90.00
PCA+kNN	0.009 245	95.37	90.16
KPCA+kNN	0.009 494	95.36	90.33
GKPCA+kNN	0.000 696	95.33	90.26
LDA+kNN	0.001 237	96.67	96.33
GDA+kNN	0.001 542	98.78	96.67
GGDA+kNN	0.000 323	98.73	96.59

表 3 几种方法对pendigits数据的分类性能比较

方法	$T_c(s)$	识别率/%	
		训练	测试
kNN	5.49	98.17	94.97
PCA+kNN	5.53	98.35	95.17
KPCA+kNN	5.87	99.35	96.57
GKPCA+kNN	0.59	99.18	96.29
LDA+kNN	3.12	98.71	95.45
GDA+kNN	3.18	99.77	97.83
GGDA+kNN	0.38	99.58	97.71

2个数据集和6种特征提取数据的分类结果表明: 用特征提取的数据进行分类比不用特征提取的数据进行分类的性能要好; 用线性特征提取方法LDA特征提取的数据分类结果好于PCA方法; 用非线性特征提取方法GGDA和GDA的特征提取性能优于GKPCA和KPCA方法; 核空间的非线性特征提取方法GGDA, GDA, GKPCA和KPCA优于输入空间的线性特征提取方法LDA和PCA; GKPCA方法的性能接近于KPCA方法的性能; GGDA方法的性能接近于GDA方法的性能. 总之, 在实验中不管对于大数据集还是小数据集, GGDA和GDA方法的特征提取性能均优于其他方法. GGDA方法不仅较好地保持了GDA方法特征提取性能, 而且减少了大数据集进行非线性特征提取的内存和计算时间. 实验也表明, 由于GGDA和GDA方法考虑了类内与类间的信息, 这种特征提取方法对于分类问题而言是较好的. 因此, 尽管GGDA方法是根据重构误差找到数据表示的次优解, 但该方法可用于降低核方法的计算量和存储需求, 同时也可以用于降低用核方法进行学习的复杂度, 还能进行在线非线性特征的提取.

4 结 论

本文提出将一种有效的特征提取方法GGDA方法用于训练数据减少和非线性特征的提取, 该方法有效地结合了GKPCA方法中提取子集的greedy技术和非线性特征提取GDA方法. 对比实验结果表明, GGDA和GDA方法的特征提取性能优于其他方法. GGDA方法不仅较好地保持了GDA方法的非线性特征提取性能, 而且减少了大数据集进行非线性特征提取的计算复杂度. 由于GGDA和GDA方法考虑了类内与类间的信息, 这种特征提取方法对于分类问题而言好于提取互不相关的特征向量的GKPCA和KPCA方法. 下一步研究方向是用GGDA方法对大数据集的在线非线性特征进行提取.

参考文献(References)

[1] Schölkopf B, Smola A J. Learning with kernels[M]. Cambridge: MIT Press, 2002: 53-55.

[2] Schölkopf B, Smola A, Muller K R. Nonlinear component analysis as a kernel eigenvalue problem[J]. Neural Computation, 1998, 10(5): 1299-1319.

[3] Baudat G, Anouar F. Generalized discriminant analysis using a kernel approach[J]. Neural Computation, 2000, 12(10): 2385-2404.

(下转第1519页)