文章编号:1007-6735(2017)05-0450-09 DOI:10.13255/j.cnki.jusst.2017.05.008

基于随机旋转集成的降维方法

王 椭, 肖庆宪
(上海理工大学 管理学院,上海 200093)

摘要: 对随机旋转集成方法提出了一种针对降维问题的改进,得到了新的降维算法框架进行随机变换降维,可以显著减少降维过程中造成的信息损失.采用随机变换降维后,训练监督学习算法时可以获得更高的准确率和更好的泛化性能.通过在模拟数据上进行的实验,证明了使用多重共线性数据进行回归分析时,与传统降维算法相比,经随机变换降维处理后可以保留更多的信息,获得更小的均方误差.对随机变换降维在手写数字识别数据集上的表现进行了研究,证明了与一般性的降维算法相比,随机变换降维在图像分类问题上可以获得更高的准确率.

关键词: 集成学习; 随机旋转集成; 降维; 多样性
中图分类号: TP 181 **文献标志码**: A

Dimension Reduction Method Based on the Random Rotation Ensemble

WANG Tuo, XIAO Qinxian
(*Business School, University of Shanghai for Science and Technology, Shanghai 200093, China*)

Abstract: For the random rotation ensemble method, a new dimension reduction algorithm named random transform dimension reduction was proposed, which can reduce the information loss caused by the reduction dimension. After the random transform dimension reduction, the training supervised learning algorithm can obtain higher accuracy and better generalization performance. Through the experiments on the simulated data, it is proved that the regression analysis using multiple collinearity data can retain more information and obtain smaller mean square error than the traditional dimensionality reduction method. The performance of the random transform dimension reduction in handwritten numeral recognition datasets was studied, and it is proved that, compared with the general dimensionality reduction algorithm, the random transform dimension reduction can achieve higher accuracy in image classification.

Keywords: *ensemble learning*; *random rotation ensemble*; *dimension reduction*; *diversity*

收稿日期: 2017-03-16
第一作者: 王 椭(1993-),男,硕士研究生.研究方向:金融工程.E-mail:1210724980@qq.com
通信作者: 肖庆宪(1956-),男,教授.研究方向:金融工程.E-mail:qxxiao@163.com

高维数据的处理一直是机器学习领域的重要问题，尤其是近年来机器学习的几个重要应用方向，如计算机视觉、自然语言处理等，都必须面对高维数据带来的挑战. 数据的维数越大，处理数据所需的时间消耗与储存空间消耗就越大，还可能存在许多与给定任务无关的特征甚至存在大量噪声，或是出现数据冗余、样本稀疏及多重共线性问题对数据分析造成干扰. 要想克服这些高维数据带来的问题，必须对数据进行一定的加工和处理.

在过去的几十年里，有大量的降维方法被不断地提出并被深入研究，其中常用的包括传统的线性降维算法如 PCA 和 LDA；核化线性降维算法如 KPCA，流形学习算法如 LE[1-4]，ISOMAP[5-7] 以及 LTSA[8]，度量学习算法如 NCA[9].

部分降维算法如线性降维与核化线性降维试图在降维过程中使噪声被消去，保留尽可能多的有效信息；而另外一部分降维算法，如流形学习与度量学习，则试图找到保留数据在高维下的结构特征的低维映射，以此来解决高维数据带来的问题.

但是在实践中，由于样本与总体间的差异，噪声有可能不会被正确的识别，样本数据的结构特征也可能与总体存在差异. 并且，将高维空间中的数据在低维空间中表示，这一数据压缩重构的过程不可避免地会造成信息的损失. 很多情况下，将降维后的数据用于预测，其准确性会低于甚至远低于采用原数据进行预测.

为解决这些问题，本文借鉴随机旋转集成 (random rotation ensemble) 的思想，并对降维问题作出了针对性改进，提出了一种新的降维算法框架随机变换降维 (RTDR). 通过对原数据集进行随机变换，生成多个子数据集后再应用现有的降维算法，然后通过 stacking 算法对降维后的结果进行集成，可以极大地减少降维过程中造成的信息损失. 这一过程本质上是在保留原数据集信息的情况下，对其在高维空间内进行扭曲变形，然后再生成对应的低维映射，最后将多个这样的低维映射进行集成. 由于每个低维映射都能包含原数据集在高维空间的结构信息，并且由于经过随机变换获得，这些信息彼此之间不完全重叠. 因此，在集成后，可以比单个低维映射保留更多的信息，并且在经过随机变换后，原数据集中可能被误判为噪声的一部分信息有可能在生成的子数据集中得到保留，同样能够得到减少信息损失的效果.

本文通过在模拟数据上的实验，验证了在使用具有多重共线性的数据进行回归分析时，随机变换降维可以有效克服多重共线性问题. 并且与传统的降维算法相比，随机变换降维提取信息更充分，得到的均方误差更小. 在经典的 MNIST 手写数字数据集上进行试验，证明了在图像分类问题中，与传统降维算法相比，采用随机变换降维准确率更高.

1　预备知识

1.1　集成学习

集成学习 (ensemble learning) 是通过构建并结合多个学习器来完成学习任务的一类算法，通过将多个学习器结合，常可获得比单一学习器显著优越的泛化性能[10-13]. Liu[14] 基于集成学习的特性提出了 EasyEnsemble，用于应对对数据集进行欠采样造成的信息损失.

在集成学习与降维算法结合时，集成学习可以从 3 方面提升降维算法的性能：a. 通过集成多个低维映射，可以降低有效信息被误判为噪声的可能；b. 通过集成多个低维投影，可以最大限度地保留数据在高维空间的性质；c. 通过集成多个低维投影，可以有效减少数据由高维向低维投影时造成的信息损失.

但是只有用于集成的学习器具有足够好的多样性时才能得到理想的效果. 多样性即个体学习器之间的差异，Krogh 等[15] 曾给出了误差 - 分歧分解 (error-ambiguity decomposition)，证明了对于回归任务，集成的泛化误差等于个体学习器泛化误差的加权均值减去个体学习器的加权分歧，说明了构造优秀的集成需要选择“好而不同”的个体学习器.

1.2　随机旋转集成

2016 年，Blaser 等[16] 发现将样本属性进行随机旋转，可以显著增强基学习器的多样性，并且基学习器的准确性仅会发生小幅下降，因此可以得到效果更好的集成学习算法，并由此提出了随机旋转集成.

假定 $\boldsymbol{x}=[\boldsymbol{x}_1,\boldsymbol{x}_2,\boldsymbol{x}_3,\cdots,\boldsymbol{x}_n]^{\mathrm{T}}$ 表示一个具有 n 个特征的样本，$D_1,D_2,D_3,\cdots,D_L$ 表示 L 个基学习器，随机旋转集成可由以下简单步骤来构造：

a. 对样本 $\boldsymbol{x}$ 的各特征作标准化处理；

b. 生成一个 $n\times n$ 的随机旋转矩阵 $\boldsymbol{R}_{n,i}$，$\boldsymbol{R}^{\mathrm{T}}=\boldsymbol{R}^{-1}$，且 $|\boldsymbol{R}|=1$；

c. 将 $\boldsymbol{x}$ 乘以 $\boldsymbol{R}_{n,i}$，使特征空间进行随机旋转，采用旋转后的样本训练基分类器 D_i；

d. 重复步骤 b，c；

e. 集成 L 个分类器的训练结果作为最终输出

结果.

1.3 stacking算法

stacking算法由Wolpert[17]提出.stacking算法使用多个不同类型的个体分类器,且将它们分为两层.首先使用多种不同的学习算法在同一训练集上进行训练,生成多个基分类器.基学习器分别对每个样本进行预测,预测结果作为新的属性,与样本的原始标签 y 结合起来作为新的训练集.第二层分类器在该训练集上进行训练,得到的结果作为最终的训练结果.

假定 $\boldsymbol{x}=[\boldsymbol{x}_1,\boldsymbol{x}_2,\boldsymbol{x}_3,\cdots,\boldsymbol{x}_n]^{\mathrm{T}}$ 表示一个具有 n 个特征的样本,$\boldsymbol{y}$ 为样本标签,$D_1,D_2,D_3,\cdots,D_L$ 表示 L 个初级学习器,D 表示次级学习器.stacking算法可经由以下简要步骤构造:

a. 采用$(\boldsymbol{x},\boldsymbol{y})$分别训练初级学习器 $D_1,D_2,D_3,\cdots,D_L$,得到的输出结果为 $\boldsymbol{Z}_1,\boldsymbol{Z}_2,\boldsymbol{Z}_3,\cdots,\boldsymbol{Z}_L$;

b. 构造新的数据集 $\boldsymbol{Z}=[\boldsymbol{Z}_1,\boldsymbol{Z}_2,\boldsymbol{Z}_3,\cdots,\boldsymbol{Z}_n]^{\mathrm{T}}$;

c. 采用$(\boldsymbol{Z},\boldsymbol{y})$训练次级学习器 D,得到的输出作为最终的输出结果.

1.4 主成分分析(PCA)

主成分分析(principal components analysis,PCA)[18-20]是一种经典的线性降维方法,它将一组观测变量 $\boldsymbol{X}_1,\boldsymbol{X}_2,\cdots,\boldsymbol{X}_n$ 通过线性变化,转化为一组互不相关的变量 $\boldsymbol{W}_1,\boldsymbol{W}_2,\cdots,\boldsymbol{W}_n$,称之为主成分.在变化过程中保持总方差不变,然后根据方差大小从 $\boldsymbol{W}_1,\boldsymbol{W}_2,\cdots,\boldsymbol{W}_n$ 中挑出 $k(k<n)$个方差最大的主成分,用这 k 个主成分替换原来的 n 个变量,实现降维的目的.

a. 求初始矩阵 $\boldsymbol{X}$ 的 $n\times n$ 阶相关系数矩阵 $\boldsymbol{R}$;

b. 求相关系数矩阵 $\boldsymbol{R}$ 的特征值 $\lambda_1\geqslant\lambda_2\geqslant\lambda_3\geqslant\cdots\geqslant\lambda_n$,以及与各特征值相对应的一组长度为1且相互正交的特征向量 $\boldsymbol{a}_1,\boldsymbol{a}_2,\cdots,\boldsymbol{a}_n$;

c. 计算提取的 q 个主成分,$\boldsymbol{W}_1=\boldsymbol{a}_1^{\mathrm{T}}\boldsymbol{X},\boldsymbol{W}_2=\boldsymbol{a}_2^{\mathrm{T}}\boldsymbol{X},\cdots,\boldsymbol{W}_n=\boldsymbol{a}_n^{\mathrm{T}}\boldsymbol{X}$.

本文在MNIST数据集上采用RTDR与PCA进行了对比,验证了RTDR减少信息损失的效果.

2 随机变换降维算法的构造

2.1 原理

本文基于随机旋转集成的思想,提出了一种新的降维算法——随机变换降维RTDR(random transform dimensionality reduction).对于降维而言,通过集成学习可以综合多个高维数据的低维映射信息,减少降维过程中的信息损失.以主成分分析为例,主成分分析构建了一个具有最大可分性的超平面(样本点在这个超平面上的投影能尽可能分开),以此来尽可能保存样本在高维空间的信息.但是事实上,其他方向的超平面同样可以保存样本的信息,并且性能可以接近于具有最大可分性的超平面.集合多个不同投影方向的信息,可以尽可能地还原出原数据集的信息,几何中的三视图则是这一思想的典型案例.将三维立体图形投影成3个二维图形(俯视图、正视图、侧视图),通过观察对比这3个图形,能够了解这一立体图形的形状和结构.但是很多情况下,只用一个投影,无法反映一个立体图形的结构.

为集成多个高维数据的低维映射的信息,RTDR以stacking算法为基础进行集成,以多个低维投影上的学习器的输出作为次级学习器的输入,通过这种形式实现集成多个低维投影的信息.stacking算法通常可以获得比个体学习器更好的性能[17],并已在有监督学习(如回归[21]、分类及距离[22])和无监督学习(如密度估计[23])上都取得了巨大的成功.

但是,Wolpert曾指出stacking方法的一个难点是如何为每个特定领域数据集配置合适的基分类器和元分类器(第二层学习器).因为第一层学习器之间的差异需要尽可能的大(否则输出结果几乎一致,第二层学习器将面对严重的多重共线性问题),通常第一层学习器需要选择不同类型的学习器进行组合,如将随机森林和logistic回归进行组合.但是,对特定问题,不同类型的学习器往往性能差异较大,用随机森林取得较高的准确率,而用logistic回归取得的正确率则较低.这样一来,较差的学习器的性能将会拖累学习器的整体性能,甚至可能出现集成后的学习器的性能比单个的优秀学习器的性能要差.

随机旋转集成可以在不损失原数据集信息的情况下生产多个子数据集,显著提升集成效果.将随机旋转集成应用于stacking算法,可以解决stacking算法基学习器选择难的问题,只需选择最恰当一类的学习器,然后通过随机旋转集成技术,即可获得任意多个新的差异较大但准确率接近的学习器.

随机旋转集成通过旋转的方式保留了子数据集的空间结构信息,通过将样本乘 $\boldsymbol{R}^{\mathrm{T}}=\boldsymbol{R}^{-1}$,且$|\boldsymbol{R}|=1$的矩阵以随机角度进行旋转来生成子数据集.在监督学习问题上,这一方法在尽可能多地保留样本信息的情况下实现了多样性.但是,在降维问题上,由于子数据集的空间结构都是一致的,降维得到的结果反而失去了多样性,难以起到集成的效果和减

少降维造成的信息损失.

为此，本文对具有 n 个特征的样本生成 $n \times n$ 的随机矩阵，随机矩阵由 n 个标准化的随机向量组成，将样本乘以随机矩阵进行随机变换，得到的新样本的各个特征都是原样本特征的线性组合，因此保留了原样本的绝大部分信息. 并且由于新样本的每个特征都是原样本的特征乘以随机权重而获得，与随机旋转集成相比，对样本的空间结构进行一定程度的扭曲变形，实现了降维的多样性，保证了对降维进行集成的效果.

通过在模拟数据上的实验可以发现，在数据具有多重共线性问题时，进行回归估计会产生严重的偏差. 在经过降维算法处理后，可以有效减小多重共线性造成的负面影响. 而与一般性的降维算法相比，通过随机变换降维，可以获得更好的性能，进行预测得到的均方误差更小，并且残差更贴近于白噪声.

由于图像处理问题中，数据维度往往会远大于样本量，如果不进行降维处理，则难以对图片进行分类或者识别，因此本文在手写数字识别数据集上进行了对比实验. 从本文实验结果可以看到，在通过随机变换得到的新样本上进行降维，然后再应用多层感知机(MLP)，可以取得与直接在原数据上进行降维然后再应用 MLP 同等水平的准确性，甚至有些情况下准确性会更高. 并且将多个随机变换后的样本降维再进行集成，从实验结果来看，准确性可以得到巨大的提升，降维程度越大提升效果越明显. 这充分说明了随机变换降维(RTDR)生成的子数据集不仅保留了原数据集的绝大部分信息，并且具有多样性，获得了集成学习所需要的“好而不同”的基学习器，也证明了 RTDR 可以减少降维带来的信息损失.

2.2　步骤

假定 $\boldsymbol{x}=[x_1, x_2, x_3, \cdots, x_n]^{\mathrm{T}}$ 表示一个具有 n 个特征的样本，$\boldsymbol{y}$ 为样本标签，$D_1, D_2, D_3, \cdots, D_L$ 表示 L 个基学习器，D 为次级学习器(本文实验中基学习器与次级学习器都采用 MLP)，F 为降维算法(本文实验中采用 PCA)，RTDR 可由以下步骤进行构造，算法的流程如图 1 所示.

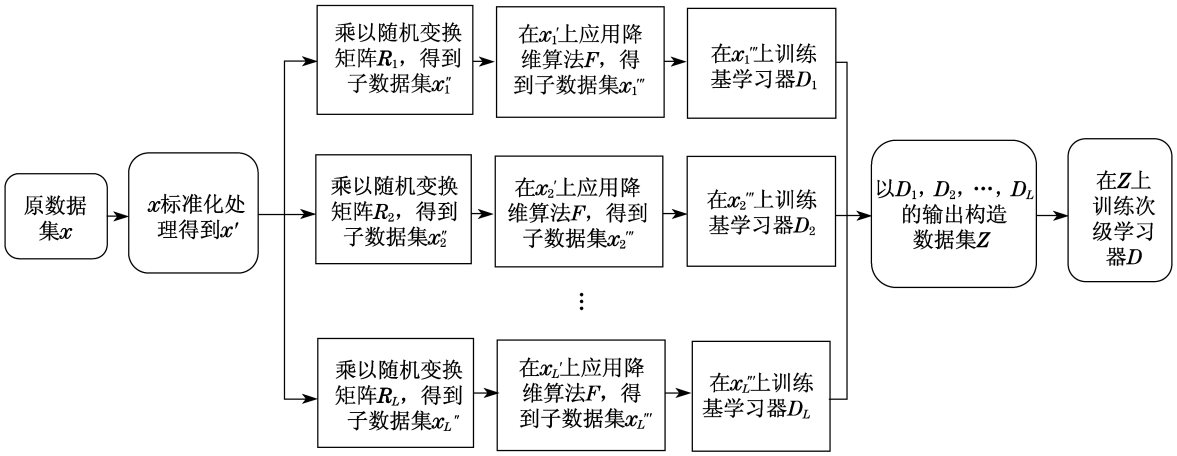

图 1　随机变换降维算法流程图

Fig.1　Flow chart of the random transform dimension reduction algorithm

a. 对样本 $\boldsymbol{x}$ 的各特征作标准化处理，$\boldsymbol{x}_i' = \dfrac{\boldsymbol{x}_i' - \min(\boldsymbol{x}_i')}{\max(\boldsymbol{x}_i') - \min(\boldsymbol{x}_i')}$；

b. 生成 L 个 $n \times n$ 随机变换矩阵 $\boldsymbol{R}$；

c. 将 $\boldsymbol{x}_i'$ 乘以 $\boldsymbol{R}_i$，进行随机变换，$\boldsymbol{x}_i'' = \boldsymbol{x}_i' \times \boldsymbol{R}_i = [x_{i1}'', x_{i2}'', x_{i3}'', \cdots, x_{im}'']^{\mathrm{T}}$；

d. 在变换后的样本上应用 F 进行降维处理，$\boldsymbol{x}'''_i = F(\boldsymbol{x}_i'') = [x'''_{i1}, x'''_{i2}, x'''_{i3}, \cdots, x'''_{im}]^{\mathrm{T}}$；

e. 采用降维后的样本分别训练初级学习器 $D_1, D_2, D_3, \cdots, D_L$，得到

$$\hat{\Omega}_i = \arg\min_{\Omega} Loss(\boldsymbol{y}, D_i(\boldsymbol{x}'''_i, \Omega_i))$$

其中，Ω_i 为 D_i 的参数，$Loss$ 为指定的损失函数；

f. D_i 训练完成后，$\boldsymbol{Z}_i = D_i(\boldsymbol{x}'''_i, \hat{\Omega}_i)$，$\boldsymbol{Z}_i$ 为 D_i 以 $\boldsymbol{x}'''_i$ 作为输入进行预测得到的输出结果，对于分类问题可以是预测的类概率，对于回归问题可以是估计值，构造新的数据集 $\boldsymbol{Z}=[\boldsymbol{Z}_1, \boldsymbol{Z}_2, \boldsymbol{Z}_3, \cdots, \boldsymbol{Z}_n]^{\mathrm{T}}$

g. 采用 $(\boldsymbol{Z}, \boldsymbol{y})$ 训练次级学习器 D，得到的输出作为最终的输出结果.

其中随机变换矩阵根据以下方式构造：

(a) 生成 n 个相互独立且服从标准正态分布 $N(0,1)$ 的随机数 $\{v_1, v_2, \cdots, v_n\}$，构造向量 $\boldsymbol{w}_i$，其

中 $w_{ij}=\dfrac{v_j}{\sqrt{v_1^2+v_2^2+\cdots+v_n^2}}$；

(b) 重复以上步骤，获得 n 个向量 $\boldsymbol{w}_1,\boldsymbol{w}_2,\cdots,\boldsymbol{w}_n$，$\boldsymbol{R}=[\boldsymbol{w}_1,\boldsymbol{w}_2,\cdots,\boldsymbol{w}_n]$.

3 模拟实验

3.1 实验介绍

多重共线性是指线性回归模型中的解释变量之间由于存在精确相关关系或高度相关关系，而使模型估计失真或难以估计准确的问题，在经济学研究中属于非常常见的现象. 经济变量往往存在共同的趋势，或是在模型中引入了滞后项，这些都会导致多重共线性问题. 通过降维算法对数据进行处理，是应对多重共线性的方法之一，经过降维处理后，可以获得更为稳定而准确的估计结果.

但是，在降维过程中，不可避免会抛弃一部分信息. 以 PCA 为例，除了方差最大的若干个方向上的信息被保留外，其他信息都被舍弃了. 通常假定这一部分信息是与因变量无关的，属于噪声，但这一假定并不总是成立. 如图 2 所示，最大方差方向上的信息对判断样本点属于“×”还是“+”毫无帮助.

随机变换降维相对于传统的降维算法而言，可以保留更多的信息，并且通过集成，可以拓展算法的假设空间. 因而在出现数据的特征不符合降维算法的假定时，采用随机变换降维处理后对数据进行回归分析得到的均方误差会更小，预测结果更准确.

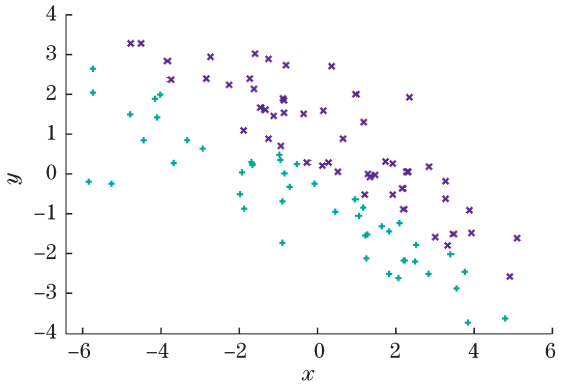

图 2 模拟数据散点图

Fig. 2 Scatter plot of simulated data

3.2 数据构造

模拟实验采用的数据通过以下方式构造：

a. 构造 n 维随机向量 $\boldsymbol{a}$，k 维随机向量 $\boldsymbol{b}$，向量中的元素相互独立且服从均值为 0，标准差为 σ 的正态分布.

b. 构造 n 维随机向量 $\boldsymbol{x}_{k+1},\boldsymbol{x}_{k+2},\cdots,\boldsymbol{x}_m$ ($m>k$)，$\boldsymbol{x}_{k+1},\boldsymbol{x}_{k+2},\cdots,\boldsymbol{x}_m$ 之间相互独立，且向量中的元素相互独立且服从标准正态分布. 令 $\boldsymbol{x}=\boldsymbol{a}^{\mathrm{T}}\times\boldsymbol{b}$，

$$\boldsymbol{X}=[\boldsymbol{x},\boldsymbol{x}_{k+1}^{\mathrm{T}},\boldsymbol{x}_{k+2}^{\mathrm{T}},\cdots,\boldsymbol{x}_{k+2}^{\mathrm{T}}]$$

由于完全的多重共线性在实际问题中较少出现，因此再对 $\boldsymbol{X}$ 中所有元素加上随机扰动，扰动项服从标准正态分布.

c. 对 $\boldsymbol{X}^{\mathrm{T}}$ 进行奇异值分解，得到 $\boldsymbol{X}^{\mathrm{T}}=\boldsymbol{W}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}}$，其中 $m\times m$ 矩阵 $\boldsymbol{W}$ 是 $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}$ 的本征矢量矩阵，$\boldsymbol{\Sigma}$ 是 $m\times n$ 的非负矩形对角矩阵，$\boldsymbol{V}$ 是 $\boldsymbol{X}\boldsymbol{X}^{\mathrm{T}}$ 的本征矢量矩阵.

d. 令 $\boldsymbol{Y}=\sum_{i=1}^{L}(\boldsymbol{X}\boldsymbol{W}_L)_i+\varepsilon$，其中 ε 为随机干扰项，服从标准正态分布. $\boldsymbol{W}_L$ 为保留了 $\boldsymbol{W}$ 上对应 $\boldsymbol{\Sigma}$ 中最小的 L 个特征值的列向量构造成的 $m\times L$ 矩阵，$\boldsymbol{Y}$ 各行的值等于 $\boldsymbol{X}\boldsymbol{W}_L$ 按行求和的值再加上随机干扰项.

e. 以 n 行 m 列矩阵 $\boldsymbol{X}$ 作为自变量，n 维向量 $\boldsymbol{Y}$ 作为因变量，构造成实验所需的数据集，在实际实验操作中，n 与 m 设置为 1 000.

3.3 实验设置

本文采用随机生成的 1 000 个样本进行实验，其中 600 个样本用于训练，400 个样本用于预测. 回归分析采用多元线性模型，RTDR 以多元线性模型为基学习器和次学习器，用于对照的降维算法为 PCA. 实验首先不作降维处理，直接应用多元线性回归并观察结果，然后分别应用 RTDR 与 PCA 将原数据由 1 000 维降至 200 维、190 维……直到降至 10 维，并分别在降维后的数据集上应用多元线性回归模型并观察结果.

3.4 实验结果

实验对具有多重共线性的模拟数据进行回归分析，分别采用了 PCA 和随机变换降维对数据进行了处理. 在未进行降维处理前，直接采用多元线性回归得到的均方误差(*MSE*)为 29.73，而经降维处理后，均方误差可以缩减为原来的 10% 左右，可以有效克服多重共线性对回归分析的影响. 并且，采用随机变换降维得到的 *MSE* 更小，说明经随机变换降维处理，可以得到更准确的预测结果. 均方误差对比结果如图 3 所示.

实验还对经 RTDR 与 PCA 处理后采用多元线性回归模型进行预测得到的残差进行比较. 图4为

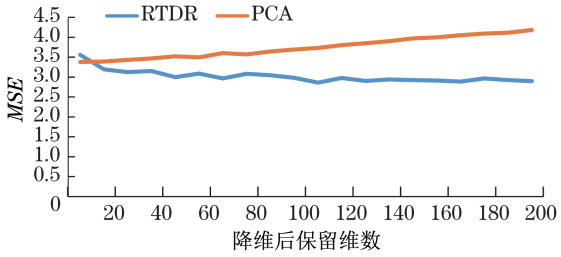

图 3　RTDR 与 PCA 的均方误差比较

Fig.3　Comparison between the MSEs of RTDR and PCA

通过 RTDR 与 PCA 将原数据降至 150 维时，对多元线性回归得到的残差采用 *P-P* 图进行检验，观察是否残差符合正态分布.

可以看到，RTDR 的残差基本分布在 *P-P* 图的对角线上，与 PCA 得到的残差相比，与对角线的偏离程度要更小，说明 RTDR 的残差要更接近于正态分布.再对 RTDR 与 PCA 得到的残差进行观察，两者的标准差、偏度、峰度如表 1 所示.

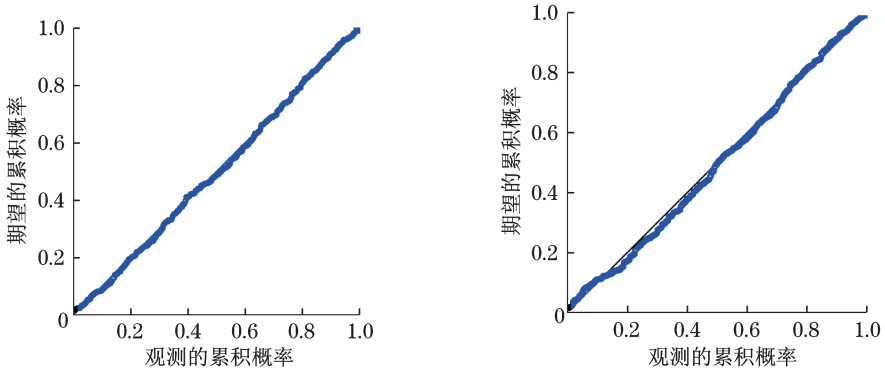

图 4　RTDR 与 PCA 的残差 *P-P* 图比较

Fig.4　Comparison between the *P-P* plots of RTDR's residual and PCA

表 1　RTDR 与 PCA 残差的标准差、偏度、峰度比较

Tab.1　Comparison between the standard deviations, skewnesses and kurtosises of RTDR's residual and PCA

维度	RTDR			PCA		
	标准差	偏度	峰度	标准差	偏度	峰度
200	1.700 393 3	0.144	−0.419	2.047 088 4	0.239	−0.418
190	1.703 213 8	0.133	−0.548	2.031 154 9	0.257	−0.417
180	1.724 267 8	0.154	−0.449	2.024 448 4	0.247	−0.408
170	1.697 758 2	0.164	−0.538	2.013 916 7	0.260	−0.379
160	1.707 274 2	0.148	−0.478	2.001 334 9	0.236	−0.395
150	1.712 524 7	0.110	−0.405	1.994 936 2	0.247	−0.407
140	1.715 745 3	0.118	−0.459	1.976 258 0	0.251	−0.414
130	1.698 635 5	0.080	−0.387	1.963 257 2	0.246	−0.411
120	1.724 062 7	0.063	−0.392	1.951 273 6	0.215	−0.438
110	1.688 331 9	0.189	−0.450	1.933 838 8	0.233	−0.432
100	1.724 251 2	0.174	−0.425	1.922 159 0	0.228	−0.432
90	1.740 480 0	0.174	−0.372	1.909 085 8	0.233	−0.433
80	1.749 246 2	0.106	−0.385	1.890 027 9	0.233	−0.409
70	1.715 588 0	0.153	−0.591	1.898 776 6	0.235	−0.416
60	1.759 432 0	0.168	−0.507	1.869 653 9	0.218	−0.430
50	1.724 665 6	0.116	−0.499	1.876 302 6	0.229	−0.424
40	1.764 827 3	0.216	−0.453	1.862 717 3	0.231	−0.384
30	1.770 061 0	0.192	−0.417	1.853 575 2	0.224	−0.430
20	1.780 215 0	0.222	−0.495	1.841 736 9	0.231	−0.431
10	1.867 508 3	0.140	−0.579	1.838 613 1	0.227	−0.421

可以看到,RTDR残差的标准差更小,说明残差的分布要更为集中.尽管峰度与PCA得到的残差峰度相似,但偏度明显要比PCA的偏度更小,进一步说明RTDR的残差更接近于正态分布.

从模拟数据的结果可以看到,经过RTDR处理后再进行多元线性回归,不仅有效克服了多重共线性问题,并且相对于PCA而言,RTDR得到的预测结果更为稳定、准确,并且对信息的提取更为充分.

4 手写数字识别数据集(MNIST)实验

4.1 实验数据

在图像处理问题中,由于图片可能具有千万级的像素,将图片转换为数据后,数据的维度也将是千万级的,远远大于可用于训练算法的样本数量,因而图像处理问题是降维算法的重要应用领域之一.因此,本文通过MNIST (mixed national institute of standards and technology database)数据集上的实验,检验了在图像分类问题中随机变换降维的性能,并与一般的降维算法进行了比较.

MNIST数据集是一个大型的手写数字识别数据集,是验证图像处理算法的常用数据集之一,也被广泛应用于训练和测试机器学习算法.MNIST数据集来源于NIST数据集,有训练集60 000例,测试集10 000例.

MNIST数据集中的每一张图片都是0~9的单个数字,每一张都是抗锯齿(anti-aliasing)的灰度图,图片大小28×28像素,数字部分被归一化为20×20的大小,位于图片的中间位置,保持了原来形状的比例.MNIST数据集中的图片如图5所示.

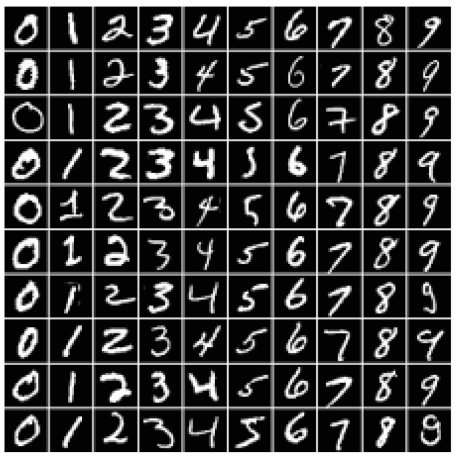

图5 MNIST数据集中的图片

Fig.5 Images in MNIST dataset

MNIST数据集的来源是两个数据集的混合:一个来自Census Bureau employees (SD-3);一个来自high-school students (SD-1).有训练样本60 000个,测试样本10 000个.训练样本和测试样本中,employee和student写的各占一半.60 000个训练样本一共由250个人写,训练样本和测试样本的来源人群没有交集,MNIST数据库也保留了手写数字与身份的对应关系.实验中,MNIST中的每张图片被表示成784维的向量.

4.2 实验设置

本文在MNIST数据集上进行实验,分别通过RTDR与PCA将MNIST数据集中的样本由784维降至100维、50维、25维、10维、3维、1维,并应用MLP在降维后的样本上进行训练,比较MLP在不同降维处理后的效果.

MLP的参数设置为,隐层数(hidden layer sizes)=50,最大迭代次数(max iter)=10,$L1$罚系数alpha=0.000 1,优化方法为sgd,学习率为0.1.

4.3 RTDR基学习器与PCA的比较结果

RTDR首先会通过随机变换生成多个子数据集,在子数据集上经过PCA处理后应用MLP作为基学习器,基学习器的输出作为次级学习器的输入.为验证随机变换不会损失原数据集的信息,本文通过实验对比了RTDR基学习器的训练准确率与直接通过PCA处理后的数据集上MLP的准确率,比较结果如表2所示.

可以看到,RTDR的基学习器(经过随机变换后采用PCA处理然后应用MLP)的训练准确率与在直接PCA降维后的数据上采用MLP的准确率非常接近,在部分情况下甚至会高于直接采用PCA降维的数据的准确率,说明经过随机变换的处理不会造成原数据集的信息损失.

4.4 RTDR次级学习器与PCA的比较结果

RTDR基学习器的输出作为次级学习器的输入,次级学习器的输出为最终的输出结果.本文将RTDR次级学习器的准确率与经过PCA处理后的数据集进行比较,验证RTDR可以显著地减少降维带来的信息损失.

图6和图7为通过PCA与RTDR将MNIST数据集中的数据降至100维、50维、25维、10维、5维、3维、1维时,在PCA处理过的数据上应用MLP和在RTDR处理过的数据上应用MLP作为次级学习器时,两者的准确率差异.可以看到,随着降维程度的增加,PCA与RTDR处理的数据上应用MLP的

表 2　RTDR 基学习器与 PCA 训练准确率比较

Tab.2　Comparison between the accuracies of the RTDR's base learner and the training of PCA

学习器	降至 1 维	降至 3 维	降至 5 维	降至 10 维	降至 25 维	降至 50 维	降至 100 维
RTDR 基学习器 1	0.295 15	0.522 93	0.750 45	0.912 70	0.970 60	0.981 20	0.987 03
RTDR 基学习器 2	0.305 35	0.516 63	0.746 90	0.920 26	0.967 43	0.981 26	0.987 00
RTDR 基学习器 3	0.295 56	0.492 60	0.746 45	0.915 06	0.969 11	0.980 11	0.986 76
RTDR 基学习器 4	0.311 63	0.516 41	0.746 61	0.907 70	0.969 16	0.981 23	0.987 80
RTDR 基学习器 5	0.302 11	0.494 75	0.750 61	0.914 18	0.970 88	0.980 20	0.987 38
RTDR 基学习器 6	0.291 20	0.514 48	0.736 51	0.909 58	0.970 86	0.979 98	0.986 26
RTDR 基学习器 7	0.303 85	0.531 85	0.745 56	0.914 81	0.969 55	0.981 33	0.987 01
RTDR 基学习器 8	0.302 18	0.525 78	0.745 35	0.907 91	0.969 53	0.979 73	0.986 96
RTDR 基学习器 9	0.303 66	0.516 55	0.746 41	0.916 51	0.968 78	0.980 08	0.986 18
RTDR 基学习器 10	0.305 63	0.522 40	0.752 21	0.924 55	0.974 36	0.981 01	0.989 00
PCA	0.305 91	0.520 63	0.753 25	0.925 13	0.974 38	0.981 15	0.988 48

准确率都在不断下降，说明降维的过程中必然会带来信息损失，影响在降维后的数据集上采用监督学习算法进行预测的准确率. 但是采用 RTDR 准确率的下降速率要明显低于采用 PCA，在最低端情况下，将原数据集降至 1 维，采用 PCA 仅有 0.3 左右的准确率，但是采用 RTDR 仍然能有接近 0.7 的准确率. 不同降维幅度下 PCA 与 RTDR 的次级学习器的准确率比较如图 6 和图 7 所示.

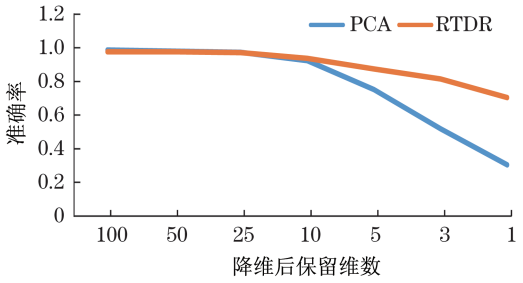

图 6　RTDR 次级学习器与 PCA 训练准确率比较

Fig.6　Comparison between the accuracies of the RTDR's secondary learner and the training of PCA

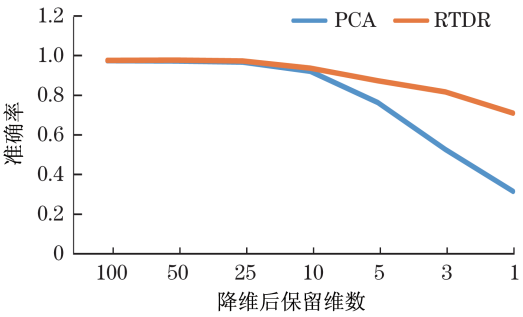

图 7　RTDR 次级学习器与 PCA 测试准确率比较

Fig.7　Comparison between the accuracies of the RTDR's secondary learner and the testing of PCA

5　结束语

从实验结果可以看到，采用 RTDR 可以显著减少降维过程中带来的信息损失，经过 RTDR 处理后，可以有效解决多重共线性问题，并且用于预测时，相对于传统的降维算法可以获得更高的准确性和稳定性，并且提取信息更充分. 采用 RTDR 进行降维的计算复杂度要高于直接采用 PCA，但是 RTDR 的结构可以使其自然地采用分布式计算，减少计算所需要消耗的时间. 但是，RTDR 仍然存在有待改进的地方，主要包括两点：a. 只适用于定量数据，不适合定性数据，对定性数据采用 RTDR 难以取得良好的效果也缺乏可解释性；b. RTDR 只适用于将数据进行降维然后进行预测的问题，即只适用于监督学习的场景，对于无监督学习的情况不能采用 RTDR.

参考文献：

[1] BELKIN M, NIYOGI P. Laplacian eigenmaps for dimensionality reduction and data representation[J]. Neural Computation, 2003, 15(6): 1373－1396.

[2] FRANZ T, ZIMMERMANN R, GÖRTZ S. Adaptive sampling for nonlinear dimensionality reduction based on manifold learning[M]// BENNER P, OHLBERGER M, PATERA A, et al. Model Reduction of Parametrized Systems. Cham: Springer, 2017.

[3] LIU Y, GU Z L, CHEUNG Y M, et al. Multi-view manifold learning for media interestingness prediction [C]// Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval. Bucharest,

Romania:ACM,2017:308-314.
[4] YANG L, WANG X. Online Appearance manifold learning for video classification and clustering[C]// International Conference on Computational Science and Its Applications. Beijing: Springer, 2016: 551-561.
[5] TENENBAUM J B, DE SILVA V, LANGFORD J C. A global geometric framework for nonlinear dimensionality reduction [J]. Science, 2000, 290 (5500):2319-2323.
[6] YANG B, XIANG M, ZHANG Y P. Multi-manifold discriminant Isomap for visualization and classification [J]. Pattern Recognition,2016,55:215-230.
[7] QU T, QU T, CAI Z X, et al. An improved Isomap method for manifold learning[J]. International Journal of Intelligent Computing and Cybernetics, 2017, 10 (1):30-40.
[8] ZHANG Z Y, ZHA H Y. Principal manifolds and nonlinear dimensionality reduction via tangent space alignment[J]. Journal of Shanghai University (English Edition),2004,8(4):406-424.
[9] GOLDBERGER J, ROWEIS S, HINTON G E, et al. Neighbourhood components analysis[C]// Proceedings of the Conference on Neural Information Processing Systems. Cambridge:MIT Press,2004.
[10] OPITZ D, MACLIN R. Popular ensemble methods: an empirical study[J]. Journal of Artificial Intelligence Research,1999,11:169-198.
[11] POLIKAR R. Ensemble based systems in decision making[J]. IEEE Circuits and Systems Magazine, 2006,6(3):21-45.
[12] ROKACH L. Ensemble-based classifiers[J]. Artificial Intelligence Review,2010,33(1-2):1-39.
[13] ZHOU Z H. Ensemble methods: foundations and algorithms[M]. Boca Raton:CRC press,2012.
[14] LIU T Y. Easy ensemble and feature selection for imbalance data sets [C] // International Joint Conference on Bioinformatics, Systems Biology and Intelligent Computing. Shanghai: IEEE, 2009: 517-520.
[15] KROGH A, VEDELSBY J. Neural network ensembles, cross validation, and active learning[C]// Proceedings of the 7th International Conference on Neural Information Processing Systems. Denver: MIT Press, 1995:231-238.
[16] BLASER R, FRYZLEWICZ P. Random rotation ensembles[J]. Journal of Machine Learning Research, 2016,17(4):1-26.
[17] WOLPERT D H. Stacked generalization [J]. Neural Networks,1992,5(2):241-259.
[18] 徐雅静,汪远征.主成分分析应用方法的改进[J].数学的实践与认识,2006,36(6):68-75.
[19] 马庆国.管理统计:数据获取、统计原理 SPSS 工具与应用研究[M].北京:科学出版社,2008:308-315.
[20] 林海明.对主成分分析法运用中十个问题的解析[J].统计与决策,2007(8):16-18.
[21] BREIMAN L. Stacked regressions [J]. Machine Learning,1996,24(1):49-64.
[22] OZAY M, VURAL F T Y. A new fuzzy stacked generalization technique and analysis of its performance [Z]. arXiv preprint arXiv: 1204.0171,2013.
[23] SMYTH P, WOLPERT D. Linearly combining density estimators via stacking[J]. Machine Learning, 1999, 36(1/2):59-83.

(编辑:丁红艺)