基于 LPP 和 Lasso 的 kNN 回归算法

龚永红[1,2],邓振云[1,3],孙 可[1,3],刘 越[1,3]

[1](广西师范大学 广西多源信息挖掘与安全重点实验室,广西 桂林 541004)
[2](桂林航天工业学院 图书馆,广西 桂林 541004)
[3](广西师范大学 计算机科学与信息工程学院,广西 桂林 541004)
E-mail: 597277287@qq.com

摘 要: 针对 kNN 回归算法中 k 值固定且未考虑样本相关性的影响,提出一种基于 LPP 和 Lasso 的最近邻算法.该算法通过局部保持投影与稀疏编码相结合,使训练样本对每一个测试样本都进行重构,重构过程中,LPP 用于保持原始数据的局部结构,l_1-范式确保每个测试样本被 k 个不同数目的最近邻样本预测,以此解决 kNN 算法中 k 值固定问题.在 UCI 数据集上得到的实验结果表明,改进算法在线性回归中的预测能力优于传统 kNN 算法.

关 键 词: kNN; 回归; 局部保持投影; 稀疏编码
中图分类号: TP181 **文献标识码:** A **文 章 编 号:** 1000-1220(2015)11-2604-05

kNN Regression Algorithm Based on LPP and Lasso

GONG Yong-hong[1,2], DEND Zhen-yun[1,3], SUN Ke[1,3], LIU Yue[1,3]
[1](Guangxi Key Lab of Multi-source Information Mining & Security, Guilin 541004, China)
[2](library, Guilin University of Aerospace Technology, Guilin 541004, China)
[3](College of Computer Science & Information Technology, Guangxi Normal University, Guilin 541004, China)

Abstract: This paper proposed a new nearest neighbor algorithm based LPP and Lasso, for solving the fixed k value problem and correlation among the samples wasn't considered of kNN algorithms. The proposed algorithm combined Locality Preserving Projections (LPP) with sparse coding (e.g., Lasso) to restructure test samples with the training data. During the reconstruction process, LPP was used to preserve the local structures of the data and the l_1-norm was used to learn different k value for various samples. Experiments results on UCI datasets showed that the proposed methods were superior to traditional kNN algorithm in terms of regression performance.
Key words: k-nearest neighbor; regression; locality preserving projections; sparse coding

1 引 言

数据挖掘是近年来数据库研究界最热的研究方向,其中回归分析是数据挖掘用于预测分析的重要手段之一.

回归分析作为一个统计预测模型,是处理多变量间相关关系的一种方法.常见的回归分析方法包括支持向量机(SVM)[1]、k-最近邻(kNN)[2]等.但这些方法都存在不足,如 SVM 训练时间长,难于处理非相关属性; k-最近邻方法没有模型创建,且易受噪声[3]和不相关属性的影响等.本文集中在 kNN 回归算法的研究.

kNN 算法是一种基于实例的学习方法[4-5].其思想是:给定一个样本空间,选取一个实例,在训练样本集中找出它的 k 个最近邻样本,并对这 k 个最近邻样本的目标值求均值,均值即作为算法的输出估计值.

但是 kNN 回归算法存在以下缺陷: kNN 算法对于 k 值的选择依赖于用户自定义或十折交叉验证方法,且确认 k 值后,kNN 算法对每个测试样本都采取固定的 k 值进行回归预测,这种固定 k 值的方法在实际应用中经常是不合理的.

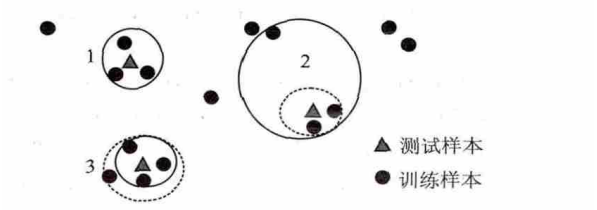

图 1 k=3 时测试样本的最近邻

Fig.1 Test samples and its 3 nearest neighbors

如图 1 中所示,该样本集有三个缺失测试样本.根据 KNN 算法,若令 k=3,则可得到图 1 中三个测试样本的最近邻样本空间(实线圈).对于第 2 个测试样本而言,最上方的样本离的较远,如果把它作为最近邻样本,很可能会影响缺失测试样本的预测[6-9].而对于第 3 个测试样本,距离该样本的最近邻样本数有四个,如果仅根据其中三个训练样本进行预测,很可能会影响预测的准确性.因此,kNN 算法对每个测试样本

收稿日期: 2014-09-01 收修改稿日期: 2014-11-27 基金项目: 国家自然基金项目(61170131,61263035,61363009)资助; 国家"八六三"高技术研究发展计划项目(2012AA011005)资助; 国家"九七三"重点基础研究发展计划项目(2013CB329404)资助; 广西自然科学基金项目(2012GXNSFGA060004)资助; 广西多源信息挖掘与安全重点实验室开放基金项目(MIMS13-08)资助. 作者简介: 龚永红,女,1970 年生,本科,馆员,研究方向为图书情报学、数据挖掘的研究; 邓振云(通信作者),男,1991 年生,硕士研究生,研究方向为机器学习和数据挖掘; 孙 可,男,1987 年生,硕士研究生,研究方向为模式识别与智能系统; 刘 越,男,1989 年生,硕士研究生,研究方向为数据挖掘、机器学习.

都取固定的 k 值是不合理的,k 值的选取应该由样本的分布或特点决定[10],即 k 值应该是从样本数据中学习而来,较为合理的最近邻样本的选择应如虚线圈所示.

对此,本文从样本间的相关性出发对其进行了分析.由于每个样本之间都存在相关性,若对每个测试样本都用与之相关的最近邻样本预测,那么预测的准确率将有很大程度的提高.本文通过训练样本重构[11]每个测试样本的方法,来得到样本之间的相关性.在重构过程中,最小二乘损失函数[12]保证了回归前后损失的信息最小; 局部保局投影(Locality Preserving Projections,LPP[13])保证了重构前后原始样本局部结构不变; Lasso(the Least Absolute Shrinkage and Selection Operator)通过控制稀疏性来实现 k 值不固定问题,从而使得每个测试样本都能够通过对样本数据的自学习来确定 k 值,有效的解决了传统 kNN 算法在固定 k 值的情况下预测不准确的情况.本文将改进算法简称为 LL-kNN(LPP-Lasso-kNN)算法[14-15].

2　相关理论

2.1　局部保持投影(LPP)

LPP 是一种新的子空间分析方法,它是非线性方法拉普拉斯特征映射(Laplacian Eigenmap)的线性近似,其原理是通过一定的性能目标去寻找线性变换矩阵 **W**,以实现对高维数据的 x 降维 y:

$$y_j=\mathbf{W}^T x_i, i=1,2,\cdots,l \tag{1}$$

其中 x_i 为训练集 $\mathbf{X}=\{x_i\}_{i=1}^{l}\in R^{D\times L}$ 中的训练样本.变换矩阵 **W** 可以通过最小化如下目标函数得到[16-17]:

$$\min_{\mathbf{W}}\sum_{i,j}(\mathbf{W}^T x_i-\mathbf{W}^T x_j)^2 S_{ij} \tag{2}$$

其中,对于权值矩阵 **S**: 可以由一个基于热核的估计定义 $S_{ij}=\exp(-\|x_i-x_j\|^2/\sigma)$($\sigma$ 是一个调优参数常量)建立.S_{ij} 的值用来测量 x_i、x_j 两个点之间的近邻程度.

通过公式(2)操作后,可发现特征空间能够保持原始的局部结构不变.公式(2)代数变换操作如下:

$$\begin{aligned}&\frac{1}{2}\sum_{i,j}(\mathbf{W}^T x_i-\mathbf{W}^T x_j)^2 S_{ij}\\&=\sum_i \mathbf{W}^T x_i D_{ij} x_i^T\mathbf{W}-\sum_{i,j}\mathbf{W}^T x_i S_{ij} x_i^T\mathbf{W}\\&=tr(\mathbf{W}^T\mathbf{X}\mathbf{D}\mathbf{X}^T\mathbf{W})-tr(\mathbf{W}^T\mathbf{X}\mathbf{S}\mathbf{X}^T\mathbf{W})\\&=tr(\mathbf{W}^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{W})\end{aligned} \tag{3}$$

D 表示为一个对角矩阵,**D** 中第 i 个元素被计算为 **S** 的第 i 列的总和,也就是 $D_{ii}=\sum_j S_{ij}$.显然,$\mathbf{L}=\mathbf{D}-\mathbf{S}$ 是一个拉普拉斯矩阵.

2.2　LL-kNN 回归算法

由于 kNN 算法在 k 值固定问题上经常导致应用上的不合理,以及研究已经证明: 保持样本的局部结构能使得学习算法取得稳定的好的学习能力.对此,本节将详细介绍 LL-kNN 回归算法.假设训练集 $\mathbf{X}\in R^{n\times d}$,测试集 $\mathbf{Y}\in R^{m\times d}$,其中 d,n,m 分别是样本的维数、训练样本个数和测试样本的个数.LL-kNN 回归算法希望通过训练样本重构测试样本,从而找到一投影矩阵 **W**,使得 **Y** 与 $\mathbf{W}^T\mathbf{X}$ 差距尽可能最小.这可以通过最小二乘损失函数实现[18-20]:

$$\min_{\mathbf{W}}\|\mathbf{Y}-\mathbf{W}^T\mathbf{X}\|_F^2 \tag{4}$$

其中 $\|\mathbf{Y}-\mathbf{W}^T\mathbf{X}\|_F^2=\sum_i\sum_j(y_{ij}-w_{ij}^T x_{ij})^2$ 为 *Frobenius* 范数.目标函数(4)是凸函数,易知其解 $\mathbf{W}^*=(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$.

但实际应用中 $\mathbf{X}^T\mathbf{X}$ 不一定可逆,对此,通常情况下会考虑引入一个 l_2-范式使其可逆,即:

$$\min_{\mathbf{W}}\|\mathbf{Y}-\mathbf{W}^T\mathbf{X}\|_F^2+\rho\|\mathbf{W}\|_2^2 \tag{5}$$

其中 $\|\mathbf{W}\|_2^2=\sum_{i=1}^{m}\sum_{j=1}^{n}|w_{ij}|^2$,$\rho$ 为 l_2-范式的正则化因子参数.目标函数(5)的优化解为 $\mathbf{W}^*=(\mathbf{X}^T\mathbf{X}+\rho I)^{-1}\mathbf{X}^T\mathbf{Y}$,但研究已经证明,该函数得到的 **W** 不一定稀疏,利用到 kNN 回归问题中,即令 k 取 n,这也根本没解决 kNN 算法中 k 值固定问题.对此,LL-kNN 算法利用 LPP 正则化因子和 l_1-范式代替公式中 l_2-范式,得到如下目标函数:

$$\min_{\mathbf{W}}\frac{1}{2}\|\mathbf{Y}-\mathbf{W}^T\mathbf{X}\|_F^2+\rho_1 tr(\mathbf{W}^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{W})+\rho_2\|\mathbf{W}\|_1 \tag{6}$$

其中 $\|\mathbf{W}\|_1=\sum_{i=1}^{m}\sum_{j=1}^{n}|w_{ij}|$. ρ_1 是保局投影的参数,其作用是调整矩阵 **W** 的结构,使 $tr(\mathbf{W}^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{W})$ 数量级保持与 $\|\mathbf{Y}-\mathbf{W}^T\mathbf{X}\|_F^2$ 一致.ρ_1 越大,LPP 比重越大; 反之越小.对 ρ_1 进行调优处理即可使原始的局部结构在重构过程中保持不变.而 l_1-范式[21]的基本思想是: 在回归系数的绝对值之和小于一个常数的约束条件下,使其残差平方和最小化,从而能够产生某些严格等于 0 的回归系数.ρ_2 作为 l_1-范式的参数,可以很好的控制矩阵 **W** 的稀疏程度,ρ_2 越大,产生的矩阵稀疏性越大; 反之越小.

目标函数(6)中的矩阵 **W** 可以理解为 **Y** 与 **X** 的相关性矩阵,则矩阵 **W** 中的元素 w_{ij} 表示第 i 个测试样本与和第 j 个训练样本之间的相关性大小.若 $w_{ij}>0$,表示 **Y** 的第 i 个样本与 **X** 的第 j 个训练样本正相关; 若 $w_{ij}<0$,则它们负相关; 而 $w_{ij}=0$,则说明它们不相关.应用此特性到 kNN 回归中,对第 i 个测试样本而言,我们当然希望选择只与它相关的样本进行回归预测,即满足 $w_{ij}\neq 0$.假设优化目标函数(6)得到如下矩阵 **W**:

$$\mathbf{W}=\begin{pmatrix}0.5 & 0.8 & 0 & 0\\0.6 & 0 & 0 & 0.1\\0 & 0 & 0.4 & 0\\0.7 & 0.1 & 0 & 0.3\end{pmatrix}$$

根据 **W** 知道,第一列有 3 个不为 0 的系数,那么说明此时有 k=3 个最近邻数,且数值越大,说明测试样本与训练样本的相关性越高; 第二列有 2 个不为 0 的系数,说明有 k=2 个最近邻数; 依次选择后可发现本算法使用了不固定的 k 值.可见考虑样本之间的相关性,能有效地解决固定 k 值的问题.综上可知,LL-kNN 算法有效的解决了 kNN 的缺陷,对样本进行回归预测时更加准确高效.注意,如果使用 ρ_2 范式就没这个功能,因为通过 ρ_2 范式得到的矩阵 **W** 不稀疏.

最后,给出 LL-kNN 算法的步骤,如下所示:

算法 1.LL-kNN 算法

输入: 样本集.

输出: 预测值.

1 对样本空间中的属性进行规范化处理,防止某个属性所占权重过大;
2 通过算法 2 得到最优化矩阵 W;
3 对 W 进行加权处理,根据加权后的 W 确定每个测试样本对应的最近邻样本数,即 k 值;
4 根据 k 值,求解测试样本的预测值.

3 LL-kNN 算法的优化

目标函数(6)是一个凸但非光滑的函数.本文通过设计一个加速近似梯度法[23]来解决这个问题.首先对目标函数(6)进行如下加速近似梯度操作:

$$f(\mathbf{W})=\frac{1}{2}\|\mathbf{Y}-\mathbf{W}^T\mathbf{X}\|_F^2+\rho_1 tr(\mathbf{W}^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{W}) \tag{7}$$

$$l(\mathbf{W})=f(\mathbf{W})+\rho_2\|\mathbf{W}\|_1 \tag{8}$$

注意 $f(\mathbf{W})$ 是凸且可微的.为了利用近似梯度方法去优化 W,首先将对 W 按如下优化规则进行迭代更新:

$$\mathbf{W}(t+1)=\arg\min_{\mathbf{W}} G_{\eta(t)}(\mathbf{W},\mathbf{W}(t)) \tag{9}$$

$$G_{\eta(t)}(\mathbf{W},\mathbf{W}(t))=f(\mathbf{W}(t))+\langle\nabla f\mathbf{W}(t),\mathbf{W}-\mathbf{W}(t)\rangle+\frac{\eta(t)}{2}\|\mathbf{W}-\mathbf{W}(t)\|_F^2+\rho_2\|\mathbf{W}\|_1 \tag{10}$$

$$\nabla f\mathbf{W}(t)=(\mathbf{X}\mathbf{X}^T+\rho_1\mathbf{X}\mathbf{L}\mathbf{X}^T)\mathbf{W}(t)-\mathbf{X}\mathbf{Y}^T \tag{11}$$

其中 $\eta(t)$ 是一个调优参数,$\mathbf{W}(t)$ 是 W 在 t 次迭代获得的值.通过忽略公式(9)中无关的 W,我们可以把它改写为:

$$\mathbf{W}(t+1)=\pi_{\eta(t)}(\mathbf{W}(t))=\arg\min_{\mathbf{W}}\frac{1}{2}\|\mathbf{W}-\mathbf{U}(t)\|_2^2+\frac{\rho_2}{\eta(t)}\|\mathbf{W}\|_1 \tag{12}$$

其中 $\mathbf{U}(t)=\mathbf{W}(t)-\frac{\nabla f(\mathbf{W}(t))}{\eta(t)}$ 和 $\pi_{\eta(t)}(\mathbf{W}(t))$ 是 $\mathbf{W}(t)$ 在凸集 $\eta(t)$ 上的欧几里德投影.由于 $\mathbf{W}(t+1)$ 在每一行都可分,如 $\mathbf{w}^i(t+1)$;因此可以分别对每一行都进行权重更新:

$$\mathbf{w}^i(t+1)=\arg\min_{\mathbf{w}^i}\frac{1}{2}\|\mathbf{w}^i-\mathbf{u}^i(t)\|_2^2+\frac{\rho_2}{\eta(t)}\|\mathbf{w}^i\|_1 \tag{13}$$

其中 $\mathbf{u}^i(t)=\mathbf{w}^i(t)-\frac{1}{\eta(t)}\nabla f(\mathbf{w}^i(t))$ 和 $\mathbf{w}^i(t)$ 分别是 $\mathbf{U}(t)$ 和 $\mathbf{W}(t)$ 的第 i 行.根据公式(13),$\mathbf{w}^i(t+1)$ 将取一个如下形式的封闭解:

$$\mathbf{w}^{i*}=\max\{|\mathbf{w}^i|-\rho_2,0\}\cdot\mathrm{sgn}(\mathbf{w}^i) \tag{14}$$

其中 $\mathrm{sgn}(\mathbf{w}^i)$ 为符号函数.同时,为了在公式(14)中利用加速近似梯度法,将引入了辅助变量 $\mathbf{V}(t+1)$ 如下:

$$\mathbf{V}(t+1)=\mathbf{W}(t)+\frac{\alpha(t)-1}{a(t+1)}(\mathbf{W}(t+1)-\mathbf{W}(t)) \tag{15}$$

系数 $\alpha(t+1)$ 通常设定为 $\alpha(t+1)=\frac{1+\sqrt{1+4\alpha(t)^2}}{2}$.

最后给出优化算法 2 的伪代码及其收敛定理:

算法 2. LL-kNN 算法的优化

Input: $\eta(0)$, $\alpha(1)=1$, γ;

Output: W;

```
Initialize t=1;
Initialize W(1) as a random diagonal matrix;
repeat
    while L(W(t)) > G_{η(t-1)}(π_{η(t-1)}(W(t)), W(t)) do
        Set η(t) = γη(t-1);
    end
    Set η(t) = η(t-1);
    Compute W(t+1) = arg min_W G_{η(t)}(W, V(t))
    Compute α(t+1) = (1+√(1+4α(t)^2))/2
    Compute V(t+1) = W(t) + ((α(t)-1)/a(t+1))(W(t+1)-W(t))
until Eq.(6) converges;
```

收敛定理 1. 设$\{\mathbf{W}(t)\}$是由算法 1 所生成的序列,然后对于 $\forall t\geq 1$,下式成立:

$$l(\mathbf{W}(t))-l(\mathbf{W}^*)\leq\frac{2\gamma L\|\mathbf{W}(1)-\mathbf{W}^*\|_F^2}{(t+1)^2} \tag{16}$$

其中 γ 是一个正的预定义常数,L 是公式(6)中 $f(\mathbf{W})$ 一个渐变的 Lipschitz 常数,$\mathbf{W}^*=\arg\min_{\mathbf{W}} l(\mathbf{W})$.收敛定理 1 表明,该加速近似梯度法的收敛速度是 $O\left(\frac{1}{t^2}\right)$.其中 t 是算法 2 中迭代的次数.

4 实验结果分析

为了比较 LL-kNN 算法和 kNN 算法的性能,本文以均方根误差(Root Mean Square Error,RMSE)和相关系数两项作为评价指标.实验中的数据集来源于 UCI 和 LIBSVM(为了使数据特征之间具有可比性,已对数据集进行标准化处理),数据集有 bodyfat,pyrim,triazines 和 Concrete_Data.

评价指标中,均方根误差是算法效果的评判标准.多次实验结果的均方根误差可以反应算法的稳定性,均方根误差越小,稳定性越高;反之越低.相关系数是反应预测值与测试样本目标值之间的相关性,取值一般在(-1,1)范围之间,(-1,0)负相关,0 表示不相关,(0,1)表示正相关.相关系数越高,预测值越接近目标值,即预测的可信度越强;反之越弱.

均方根误差的计算公式如下:

$$\mathrm{RMSE}=\sqrt{\frac{\sum_{i=1}^{n}(y_i-\hat{y}_i)}{n}} \tag{17}$$

本文算法采用 MATLAB 编程,并在 Win7 系统下的 MATLAB 7.1 软件上进行运行测试.为了保证实验的公平性,所有算法都采用相同的样本集和测试集.

本文提出的 LL-kNN 改进算法有两个重要参数 ρ_1,ρ_2,其中参数 ρ_1 控制重构矩阵 W 的稀疏程度,参数 ρ_2 控制正则项 $tr(\mathbf{W}^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{W})$ 与损失函数 $\|\mathbf{Y}-\mathbf{W}^T\mathbf{X}\|_F^2$ 的数量级.经初步测试后,实验发现 $\rho_1,\rho_2\in\{10^{-2},\cdots,10^{-6}\}$ 能使得重构矩阵 W 产生稀疏,且正则项 $tr(\mathbf{W}^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{W})$ 与损失函数的 $\|\mathbf{Y}-\mathbf{W}^T\mathbf{X}\|_F^2$ 数量级可以使算法达到较好的结果.因此,我们对参数 ρ_1,ρ_2 在设定取值内进行了如下实验,如下页图 2 所示.

从图 2 可以看出,大部分情况下,相关系数随着 ρ_1,ρ_2 的增大而增加,*RMSE* 随着 ρ_1,ρ_2 的增大而减小.当参数 $\rho_1\in(10^{-5},10^{-4})$ 时,三个数据集随着参数 ρ_2 的增大相关系数和 RMSE 的增长都更加稳定.因此,为了使 LL-kNN 效果达到最

优,在经典 kNN 算法与 LL-kNN 算法的对比实验中,参数 ρ_1, ρ_2 在 $\rho_1 \in (10^{-5}, 10^{-4})$, $\rho_2 \in (10^{-5}, 10^{-4})$ 范围内随机取值.

在 kNN 与 LL-kNN 算法的对比实验中,我们对每个数据

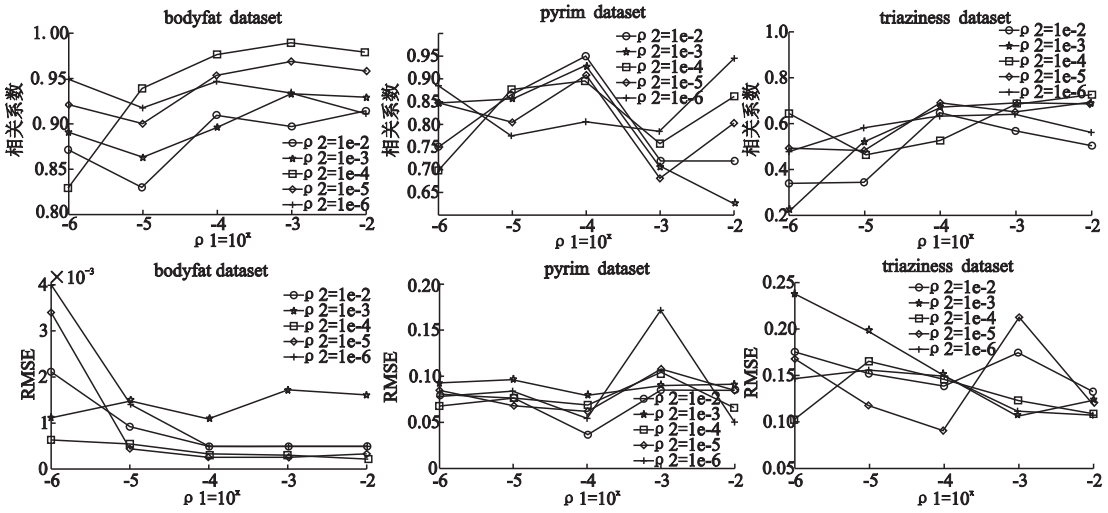

图 2　LL-kNN 算法在不同数据集不同参数下的相关系数和 RMSE(均方根误差)

Fig.2　Correlation and RMSE of LL-kNN on the different datasets with different parameters setting

集都重复 20 次,实验结果不但报告每次实验的结果而且报告 20 次结果的均值和方差,请见表 1 和表 2,以及对应的图 3、图 4 的实验结果.

表 1、图 3 中 4 个数据集的 20 次实验结果表明: LL-kNN 算法比 kNN 算法得到的预测值和测试样本的目标值相关度更高,这说明算法 LL-kNN 预测值更接近真实值,可信度高.

表 1　二种算法在四个数据集上的相关系数

Table 1　Correlation of kNN and LL-kNN on four datasets

Dataset	LL-kNN	kNN
bodyfat	0.9582±0.0013	0.9379±0.0017
pyrim	0.9213±0.0025	0.7874±0.0102
triazines	0.6726±0.0113	0.5136±0.0104
Concrete_Date	0.9082±2.6501e-04	0.8636±3.2841e-04

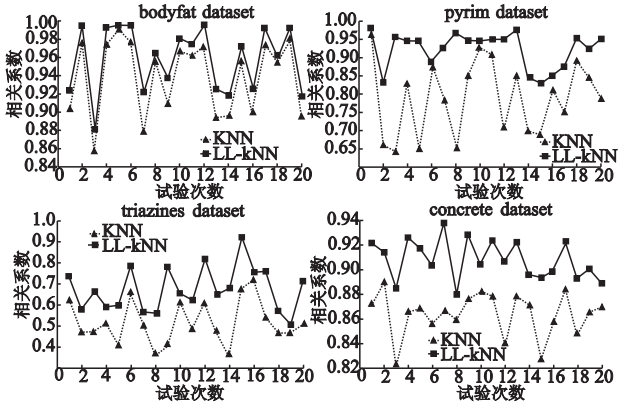

图 3　kNN 和 LL-kNN 在四个数据集上的相关系数

Fig.3　Correlation of kNN and LL-kNN on four datasets

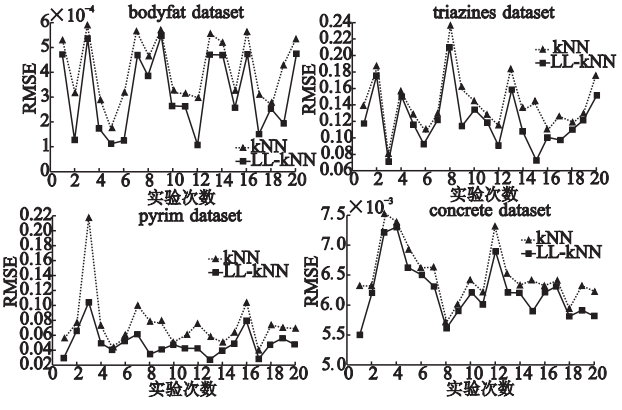

图 4　kNN 和 LL-kNN 在四个数据集上的均方根误差

Fig.4　RMSE of kNN and LL-kNN on four datasets

而从表 2、图 4 中可发现: LL-kNN 算法得到的 RMSE 值都比 kNN 低,这说明 LLk-NN 算法的稳定性更高.综上所述,

表 2　二种算法在四个数据集上的均方根误差

Table 2　RMSE of kNN and LL-kNN on four datasets

Dataset	LL-kNN	kNN
bodyfat	3.1562e-04±2.5786e-08	4.1004e-04±1.7360e-08
pyrim	0.0488±0.0003	0.0733±0.0014
triazines	0.1215±0.0012	0.1411±0.0012
Concrete_Date	0.0062±2.3355e-07	0.0065±2.2513e-07

LL-kNN 算法在稳定性以及可信度方面,都比传统 kNN 算法更加适合做回归分析.

5　结　论

本文提出的基于 LPP 和 Lasso 的 kNN 改进算法 LL-kNN,充分考虑了样本之间的相关性,并通过重构方法,使得测试样本能够根据样本的分布和特点确定最近邻样本数 k,解决了传统 kNN 算法中 k 值固定问题.此外,传统 kNN 算法 k 值的选定通常是由用户自定义得到,而本文算法的 k 值由样本数据驱动学习选定.因此本文算法可用于用户无法通过经验选定 k 值的情况,大大的减少了测试 k 值的时间.综上,

LL-kNN 算法在准确性和效率上都优于传统 kNN 算法.

References:

[1] Joachims T.Text categorization with support vector machines: learning with many relevant features [C] .In: European Conference on Machine Learning, Berlin: Springer, 1998: 137-142.

[2] Yang Y, Lin X. A re-examination of text categorization methods [C] .In: Special Interest Groupon Information Restrieval, 1999.

[3] Liu H, Zhang S. Noisy data elimination using mutual k-nearest neighbor for classification mining [J] .Journal of Systems and Software, 2012, 85(5): 1067-1074.

[4] Wu X, Zhang C, Zhang S. Efficient mining of both positive and negative 5 association rules [J] .ACM Trans.Inf.Syst.2004, 22(3): 381-405.

[5] Wu Xin-dong, Zhang Cheng-qi, Zhang Shi-chao. Database classification for multi-database mining [J] .Inf.Syst.2005, 30(1): 71-88.

[6] Zhang S, Jin Z, Zhu X. Missing data imputation by utilizing information within incomplete instances [J] .The Journal of Systems and Software, 2011, 84(3): 452-459.

[7] Qin Y, Zhang S, Zhu X, et al. Semi-parametric optimization for missing data imputation [J] .Appl.Intell, 2007, 27(1): 79-88.

[8] Zhang S, Qin Z, Ling C, et al. Missing Is Useful': missing values in cost-sensitive decision trees [J] . IEEE Trans, Knowl, Data Eng, 2005, 17(12): 1689-1693.

[9] Zhu X, Zhang S, Jin Z, et al. Missing value estimation for mixed-attribute data sets [J] .IEEE Trans, Knowl, Data Eng, 2011, 23(1): 110-121.

[10] Zhu X, Zhang L, Huang Z. A sparse embedding and least variance encoding approach to Hashing [J] . IEEE Transactions on Image Processing, 2014, 23(9): 3737-3750.

[11] Covert, Hartp. Nearest neighbor pattern classification [J] . IEEE Transactions on Information Theory, 1967, 13: 21-27.

[12] Zhu X, Huang Z, Cheng H, et al. Sparse hashing for fast multimedia search [J] .ACM Tran Inf Syst, 2013, 31(2): 9.

[13] Wu X, Zhang S. Synthesizing high-frequency rules from different data sources [J] .IEEE Trans, Knowl, Data Eng, 2003, 15(2): 353-367.

[14] Zhu X, Huang Z, Shen H, et al. Dimensionality reduction by mixed kernel canonical correlation analysis [J] .Pattern Recognition, 2012, 45(8): 3003-3016.

[15] Zhang S. Nearest neighbor selection for iteratively knn imputation [J] .Journal of Systems and Software, 2012, 85(11): 2541-2552.

[16] Zhu X, Huang Z, Shen H, et al. Linear cross-modal hashing for efficient multimedia search [C] .ACM Multimedia, 2013: 143-152.

[17] Zhu X, Huang H, Yang Y, et al. Self-taught dimensionality reduction on the high-dimensional small-sized data [J] .Pattern Recognition, 2013, 46(1): 215-229.

[18] Zhang S, Zhang C, Yan X. Post-mining: maintenance of association rules by weighting [J] .Inf, Syst, 2003, 28(7): 691-707.

[19] Zhu X, Suk H, Shen D. A novel matrix-similarity based loss function for joint regression and classification in ad diagnosis [J] .Neuro Image, 2014, 100: 91-105.

[20] Zhu X, Suk H, Shen D. Matrix-similarity based loss function and feature selection for alzheimer's disease diagnosis [C] .In Computer Vision and Pattern Recognition, 2014: 3089-3096.

[21] Zhao Y, Zhang S. Generalized dimension-reduction framework for recent-biased time series analysis [J] . IEEE Trans. Knowl. Data Eng, 2006, 18(2): 231-244.

[22] Zhu X, Huang Z, Cui J, et al. Video-to-shot tag propagation by graph sparse group lasso [J] . IEEE Transactions on Multimedia, 2013, 15(3): 633-646.