一种新的基于MSC和ISOMAP的快速流形学习算法

雷迎科
(电子工程学院 合肥 230037)

摘 要 针对等距特征映射(ISOMAP)算法计算复杂度高的问题,提出了一种新的基于最小子集覆盖(MSC)策略的快速等距特征映射算法(Fast-ISOMAP)。与原始的ISOMAP算法相比,Fast-ISOMAP算法在不显著改变原始ISOMAP算法嵌入性能的条件下,大大提高了算法的计算效率,也适用于大规模流形学习问题。在标准数据集上的实验结果验证了该算法的有效性。

关键词 等距特征映射,最小子集覆盖,多维尺度分析,流形学习

中图法分类号 TP181 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2015.8.050

New Fast Manifold Learning Algorithm Based on MSC and ISOMAP

LEI Ying-ke
(Electronic Engineering Institute, Hefei 230037, China)

Abstract For the high complexity problem of the isometric feature mapping algorithm(ISOMAP), we designed a new fast isometric feature mapping(Fast-ISOMAP) method based on minimum set cover(MSC) strategy. It is found in experiments that Fast-ISOMAP can greatly improve the computational efficiency of the original ISOMAP and be used in large-scale manifold learning problems under the condition that it does not significantly change the performance of ISOMAP. Experimental results on many artificial benchmark datasets show the effectiveness of our proposed algorithm.

Keywords Isometric feature mapping, Minimum set cover, Multidimensional scaling, Manifold learning

1 引言

流形学习假定输入数据是嵌入在高维观测空间的低维流形上,流形学习方法的目的是找出高维数据中所隐藏的低维流形结构。经过十多年的研究与探索,人们提出了大量的流形学习理论与算法。经典的流形学习方法有等距特征映射算法(Isometric Feature Mapping, ISOMAP)[1]、局部线性嵌入算法(Locally Linear Embedding, LLE)[2]、Laplacian特征映射算法(Laplacian Eigenmaps, LE)[3]、Hessian特征映射算法(Hessian-based Locally Linear Embedding, HLLE)[4]、最大差异展开算法(Maximum Variance Unfolding, MVU)[5]、局部切空间排列算法(Local Tangent Space Alignment, LTSA)[6]、黎曼流形学习算法(Riemannian Manifold Learning, RML)[7]、柯西图嵌入算法(Cauchy Graph Embedding, CGE)[8]、自适应流形学习算法(Adaptive Manifold Learning)[9]、邻域保持多项式嵌入算法(Neighborhood Preserving Polynomial Embedding, NPPE)[10]等。流形学习方法的非线性本质、几何直观性和计算可行性,使得它在数据可视化[1,2]、信号处理[11]、模式识别[12]和图像处理[13]等领域得到了成功的应用。

等距特征映射(ISOMAP)是一种代表性的流形学习算法,它利用所有样本点对之间的测地距离矩阵来代替多维尺度分析(Multidimensional Scaling, MDS)算法中的欧氏距离矩阵,以保持嵌入在高维观测空间中的低维流形的全局几何特性,从而使所有样本点之间的流形距离在从高维到低维的流形学习过程中能够得到最大的重构。算法的关键是利用样本点之间的欧氏距离计算出所有样本点对之间的测地距离,真实地再现高维数据内在的非线性几何结构。对于近邻点,利用输入空间的欧氏距离直接得到其测地距离;对于非近邻点,利用近邻图上两点之间的最短路径来近似测地距离。然后使用经典的MDS算法在高维输入空间与低维嵌入空间之间建立等距映射,从而发现嵌入在高维空间的内在低维表示。ISOMAP算法的详细描述请参考文献[1]。

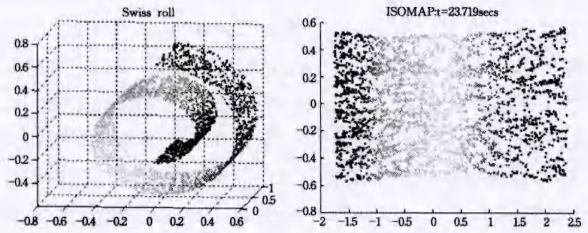
在ISOMAP算法中,影响计算复杂度的因素主要有两个[11]:(1)计算 $n\times n$ 最短路径距离矩阵 D_G。如果使用Floyd算法,计算复杂度为 $O(n^3)$;如果使用Dijkstra算法,计算复杂度降低为 $O(kn^2\log n)$(k 为近邻数,n 为样本个数)。(2)多维尺度分析中的特征值分解。需要对 $n\times n$ 稠密矩阵 $\tau(D_G)$ 进行特征值分解,计算复杂度为 $O(n^3)$,而在像LLE和LE等局部特性保持方法中,其特征值分解时所涉及到的矩阵是稀疏矩阵,它们的计算复杂度要比ISOMAP低得多。不难看出,随着样本个数 n 的增大,ISOMAP计算效率低下的问题将变得十分突出。为了减少ISOMAP算法的计算时间,de Silva和Tenenbaum提出了L-ISOMAP算法(ISOMAP with Landmark points)[14],即在样本集中选出 p 个样本点作为Landmark点,其中 $p\ll n$。在构造最短路径距离矩阵时,并不是计算所有样本点对之间的距离,而是仅仅计算样本点与

到稿日期:2014-09-25 返修日期:2014-12-19 本文受国家自然科学基金(61272333,61273302,61171170,61473237),安徽省自然科学基金(1208085MF94,1308085QF99,1408085MF129)资助。

雷迎科(1975—),男,博士,副研究员,主要研究方向为机器学习,E-mail:leiyingke@163.com。

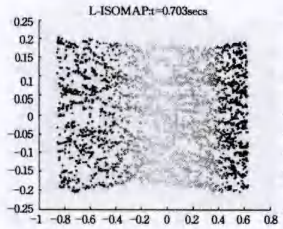
Landmark 点之间的距离，从而得到一个 $p\times n$ 距离矩阵 $D_{p\times n}$；然后将 Landmark MDS 方法[15]应用到 $D_{p\times n}$ 以得到低维嵌入的结果。首先只对 Landmark 点应用经典的 MDS 方法，将它们嵌入到 R^d 空间，然后对其余每个点，根据自身与 Landmark 点之间的距离约束在 R^d 空间进行定位。de Silva 等人认为，如果 $p\geqslant d+1$ 且 Landmark 点在数据集中分布均匀，则有充足的约束唯一地确定每个样本点的位置。通过这种选择 Landmark 点的方式可以将计算最短路径和低维嵌入时的计算复杂度分别减少到 $O(kpn\log n)$ 和 $O(np^2)$。

显然，L-ISOMAP 算法的计算复杂度远远低于原始的 ISOMAP 算法。但是，Landmark 点选取的好坏，将直接影响到 L-ISOMAP 算法最终的低维嵌入结果。de Silva 和 Tenenbaum 在 L-ISOMAP 算法中采用了随机选择的办法，但是当 Landmark 点过于集中导致所选取的 Landmark 点不足以反映数据的内在几何分布时，得到的低维嵌入效果会很差，使得该算法不稳定。图 1 显示了样本点个数 $n=3000$ 的 Swiss-roll 数据集及 ISOMAP 与 L-ISOMAP 算法的二维嵌入结果。从图 1(b)可以看出，原始的 ISOMAP 算法耗时为 23.719s；图 1(c)是随机选取 50 个 Landmark 点的 L-ISOMAP 算法获得的二维嵌入结果，与原始的 ISOMAP 相比，其性能没有显著的变化，但是算法的时间复杂度大大降低，仅为 0.703s；图 1(d)是所选择的 50 个 Landmark 点近似呈一直线分布时 L-ISOMAP 算法的嵌入结果，尽管所花费的时间大大减少，仅为 0.719s，但其二维嵌入结果发生了严重的变形。因此如何选择合理的 Landmark 点对 L-ISOMAP 算法的结果有着至关重要的影响。

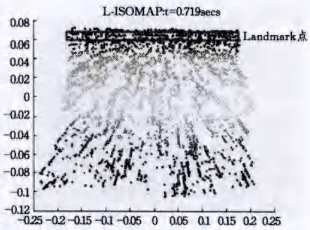


(a)Swiss-roll 数据集

(b)ISOMAP with $k=10$



(c)L-ISOMAP with $k=10$，随机选择 50 个 Landmark 点



(d)L-ISOMAP with $k=10$，50 个 Landmark 点近似位于一条直线上

图 1　$n=3000$ 的 Swiss-roll 数据集及 ISOMAP 与 L-ISOMAP 算法的二维嵌入结果

受 L-ISOMAP 算法的启发，本文提出一种基于最小子集覆盖(Minimum Set Cover，MSC)的快速等距特征映射算法(Fast-ISOMAP)，与原始的 ISOMAP 算法相比，Fast-ISOMAP 算法有如下特点：

(1)Fast-ISOMAP 算法采用最小子集覆盖策略(MSC)从所有样本点中选择 p 个 Landmark 点($p\ll n$)。在构造最短路径距离矩阵时，用 $p\times n$ 距离矩阵 $D_{p\times n}$ 代替了原始的 $n\times n$ 距离矩阵 $D_{n\times n}$，从而显著提高了原始 ISOMAP 算法的计算效率，使它能够适用于大规模流形学习问题。

(2)Fast-ISOMAP 算法对如何选择 Landmark 点提出了明确的约束，使得所选择的 Landmark 点能够很好地反映内在数据分布，从而确保了算法的稳定性。

2　快速等距特征映射

已知高维观测空间 R^D 中 n 个样本点集合 $X=[x_1,x_2,\cdots,x_n]\in R^{D\times n}$，假定 X 中的样本点采样于光滑的低维流形。Fast-ISOMAP 算法的目的是在 ISOMAP 算法的基础上利用最小子集覆盖策略(MSC)既准确又快速地发现高维输入样本 X 在本征低维空间的嵌入结果 $Y=[y_1,y_2,\cdots,y_n]\in R^{d\times n}$，其中 $d\ll D$，y_i 是对应于 $x_i(i=1,2,\cdots,n)$ 的低维表示。

2.1　基于最小子集覆盖的 Landmark 点选择

众所周知，ISOMAP 算法第一步是为每个样本点确定一个邻域，根据这 n 个邻域 $N_1,\cdots,N_n$ 构造一个近邻图。然而在这个近邻图中，许多边对于低维嵌入来说是冗余的，即忽略其中一部分边对于低维嵌入结果几乎没有影响。因此可以从 n 个邻域 $N_1,\cdots,N_n$ 中选择一部分邻域来构造一个更简单的近邻图，从而在不影响算法低维嵌入结果的前提下，显著提高其计算效率。我们的目标是在满足下面两个约束条件下选择一个最小邻域子集：(1)所选择的邻域之间彼此有重叠，确保所构造的近邻图是连通的；(2)所选择的邻域子集的并集必须包含所有 n 个样本点。现在用数学的语言来描述上述问题：假设 $X=[x_1,x_2,\cdots,x_n]\in R^{D\times n}$ 表示一个有限的样本点集合，对每个样本点 x_i，可以确定一个邻域 N_i，用 $F=\{N_1,\cdots,N_n\}$ 表示所有 n 个邻域的集合。现在的问题是从 F 中找到一个最小的子集 $C\subseteq F$ 覆盖 X 中所有的样本点，并且在 C 中每个邻域之间有重叠，重叠系数不少于 $\alpha(0<\alpha<1)$，即对于每个邻域 $S\in C$，$|S\cap(\bigcup_{T\in C\setminus\{S\}}T)|\geqslant\alpha|S|$。

这个问题是经典的最小子集覆盖的变异问题[16,17]，当 $\alpha=0$ 时，它就退化成经典的最小子集覆盖问题。对于最小子集覆盖问题，传统的近似解法是采用贪婪算法，它通过迭代方式，每次选取一个包含剩下未被覆盖样本最多的子集，直到所选择的子集的并集包含了所有的样本点。我们可以对这个贪婪算法进行扩展，使它适合于求解我们所面临的问题。扩展后的贪婪算法如表 1 所列。集合 C 包含了待构造的覆盖，集合 R 包含剩下未被覆盖的样本点。表 1 中的第 4 行作为该贪婪算法中的判决条件，从 R 中选择与已覆盖样本点之间有重叠且包含未覆盖样本点最多的邻域 S。将所选择的邻域 S 插入到 C 中，同时从 R 中删除由邻域 S 所覆盖的样本点。当 R 变成空集时，算法迭代终止，此时集合 C 即为所求的最小子集覆盖。扩展后的算法与经典的贪婪最小子集覆盖算法之间唯一的区别是表 1 中第 4 行的判决条件 $|S\cap R|\leqslant(1-\alpha)|S|$($|S|$ 表示邻域 S 所包含样本点的个数)。

表 1　扩展后的贪婪最小子集覆盖算法

```
1. 选择包含样本数最多的邻域 S∈F;
2. C←{S};R←X−S;
3. while R≠∅;do
4.     选择邻域 S∈F,使|S∩R|达到最大且|S∩R|≤(1−α)|S|;
5.     C←C∪{S};R←R−S;
6. end while
```

扩展后的贪婪最小子集覆盖算法能够以时间复杂度 $O(kn)$ 进行高效求解[16,17]。一旦从 n 个邻域中找到了最小子集覆盖 $C=\{N_{a_1},\cdots,N_{a_p}\}$（$p$ 表示覆盖 C 所包含邻域的个数，$a_1,\cdots,a_p\in\{1,\cdots,n\}$），那么 Landmark 点即为 C 中确定每个邻域的样本点 $\{x_{a_1},\cdots,x_{a_p}\}$。

2.2 基于 Landmark MDS 的低维嵌入

根据最小子集覆盖策略获得 p 个 Landmark 点 $\{x_{a_1},\cdots,x_{a_p}\}$（$p\ll n$），可以计算每个 Landmark 点与其他样本点之间的最短路径，从而得到一个 $p\times n$ 最短路径距离矩阵 $D_{p\times n}$，其取代了原始 ISOMAP 算法的 $n\times n$ 距离矩阵。下面应用 Landmark MDS 算法[15]求解输入样本 X 在本征低维空间的嵌入结果 $Y=[y_1,y_2,\cdots,y_n]\in R^{d\times n}$。Landmark MDS 首先将经典的 MDS 方法应用到只包含 Landmark 点的最短路径距离矩阵 $D_{p\times p}$，计算 Landmark 点的低维嵌入坐标，其求解过程如表 2 所列。

表 2 Landmark 点的低维嵌入

1. 构建双中心化的内积矩阵 $B_p=-H_p\Delta_pH_p/2$，其中 Δ_p 是所有 Landmark 点之间的最短路径平方距离矩阵 $(\Delta_p)_{ij}=(D_{p\times p})_{ij}^2$，$H_p$ 是中心化矩阵 $\{(H_p)_{ij}=\delta_{ij}-1/p\}$；
2. 计算 B_p 的前 d 个最大特征值 $\lambda_1,\cdots,\lambda_d$ 及其所对应的特征向量 $v_1,\cdots,v_d$；
3. p 个 Landmark 点对应的 d 维嵌入坐标由 $L=[\sqrt{\lambda_1}v_1,\cdots,\sqrt{\lambda_d}v_d]^T$ 矩阵的列向量给出。

将 Landmark 点嵌入到低维空间 R^d 后，Landmark MDS 利用剩余样本点与 Landmark 点之间的距离约束，计算剩余样本点的低维嵌入坐标。其求解过程如表 3 所列。

表 3 其余样本点的低维嵌入

1. 令 $\beta_1,\cdots,\beta_p$ 表示 Δ_p 的 p 个列向量，计算均值向量 $\beta_u=(\beta_1+\cdots+\beta_p)/p$；
2. 计算所有 Landmark 点的 d 维嵌入坐标矩阵 L 的伪逆转置 $L^{\#}=[v_1/\sqrt{\lambda_1},\cdots,v_d/\sqrt{\lambda_d}]^T$；
3. 对于一个样本点 x，其低维嵌入坐标 $y=-L^{\#}(\beta_x-\beta_u)/2$，其中 β_x 表示样本点 x 与所有 Landmark 点平方距离列向量。

关于 Landmark MDS 算法的详细描述参见文献[15]。利用最小子集覆盖策略从所有样本点中选择 p 个 Landmark 点（$p\ll n$），用 $p\times n$ 最短路径距离矩阵 $D_{p\times n}$ 代替了原始的 $n\times n$ 距离矩阵 $D_{n\times n}$，从而有效解决了原始 ISOMAP 算法存在的两个计算瓶颈问题，显著提高了算法的运行效率。而且算法在选择 Landmark 点时提出了明确的约束，使得所选择的 Landmark 点能够很好地反映内在数据分布，从而确保了算法的稳定性。本文把该算法称为快速等距特征映射算法（Fast-ISOMAP）。

2.3 Fast-ISOMAP 算法的计算复杂度分析

表 4 列举了 Fast-ISOMAP 算法主要操作的计算复杂度。从表中可以看出，Fast-ISOMAP 算法的计算复杂度主要与样本个数 n、输入空间维数 D、近邻数 k 和 Landmark 点数 p 等参数有关。Fast-ISOMAP 算法由于采用最小子集覆盖策略，有效解决了原始 ISOMAP 算法的两个计算瓶颈问题：(1)计算最短路径距离矩阵的时间复杂度由 $O(kn^2\log n)$ 降低为 $O(kpn\log n)$；(2)计算低维嵌入的时间复杂度由 $O(n^3)$ 降低为 $O(np^2)$。同时也注意到选择 Landmark 点所需要的时间开销是非常低的。在实际应用中，样本个数 n 和输入空间维数 D 一般比较大，因此 Fast-ISOMAP 算法主要的时间开销在于选择 k 近邻、计算最短路径距离矩阵 $D_{p\times n}$ 和计算低维嵌入。

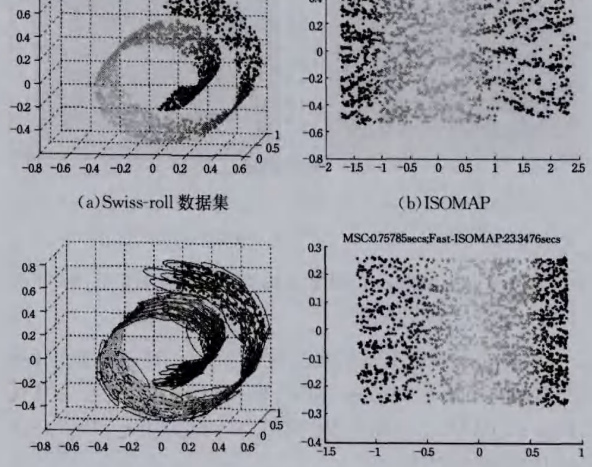
表 4 Fast-ISOMAP 算法的计算复杂度

计算过程	计算复杂度
选择 k 近邻	$O(Dn^2)$
计算最短路径距离矩阵 $D_{p\times n}$	$O(kpn\log n)$
选择 Landmark 点	$O(kn)$
计算 X 的低维嵌入坐标 Y	$O(np^2)$

3 实验结果

为了评估快速等距特征映射算法（Fast-ISOMAP）的整体性能，在大小分别为 $n=2000$、5000 和 10000 的 Swiss-roll 数据集上进行了大量实验，并与原始 ISOMAP 方法就低维嵌入结果和计算时间进行了比较。实验的目标是把嵌入在三维空间的人工数据集映射到二维空间。ISOMAP 和 Fast-ISOMAP 算法的近邻数 k 除了在 $n=2000$ 的 Swiss-roll 数据集为 25 外，在其它情况均固定为 30，除此，Fast-ISOMAP 算法的重叠系数 α 固定为 0.1。所有的实验均在一台 CPU 为 4×AMD Opteron2220 2.80GHz，内存为 4.0GB 的 HP xw9400 Workstation 的电脑上运行。

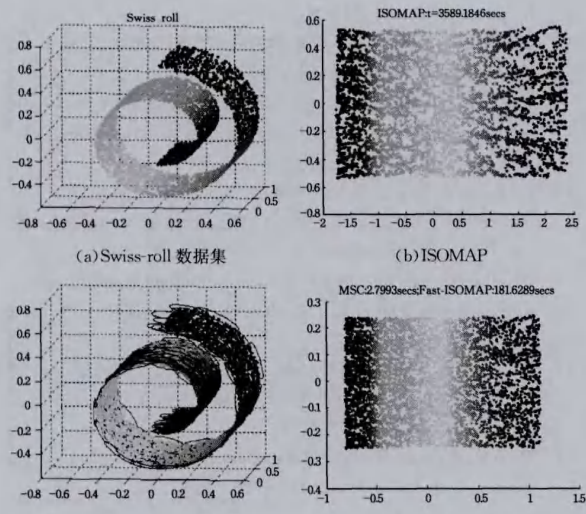
图 2—图 4 分别显示了 ISOMAP 和 Fast-ISOMAP 算法在 $n=2000$、5000 和 10000 的 Swiss-roll 数据集上的二维嵌入结果。图 2(a)、图 3(a)和图 4(a)分别为 $n=2000$、5000 和 10000 的 Swiss-roll 数据集，几何上，Swiss-roll 曲面的本征结构是二维的矩形区域。图 2(b)、图 3(b)和图 4(b)为原始 ISOMAP 算法的二维嵌入结果。图 2(c)、图 3(c)和图 4(c)为带有最小子集覆盖的 Swiss-roll 数据集。从图 2(d)、图 3(d)和图 4(d)中可以看出，相比原始 ISOMAP 算法而言，尽管 Fast-ISOMAP 算法采用了 Landmark 点，但其二维嵌入结果并没有显著变化，然而其计算时间却大大降低（见表 5），以 $n=5000$ 的 Swiss-roll 数据集为例，ISOMAP 算法计算时间为 3589.1846s，而 Fast-ISOMAP 算法仅为 181.6289s（计算最小子集覆盖时间为 2.7993s），是原始 ISOMAP 算法的 1/20（$n=2000$ 时为 1/15，$n=10000$ 时为 1/22）。



(a)Swiss-roll 数据集　(b)ISOMAP

(c)带有最小子集覆盖的 Swiss-roll 数据集　(d)Fast-ISOMAP

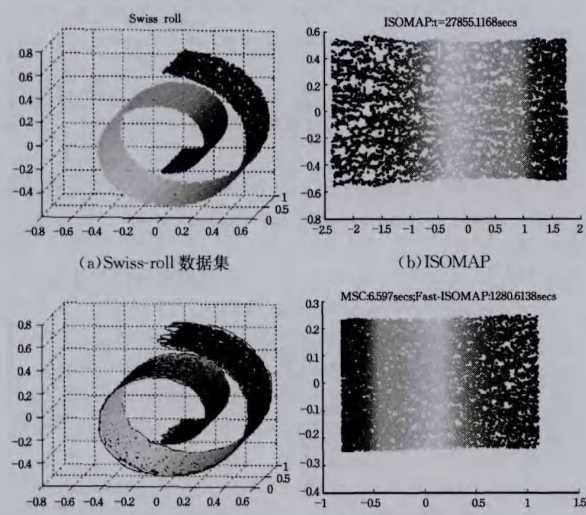
图 2 $n=2000$ 的 Swiss-roll 数据集和二维嵌入结果



(a)Swiss-roll 数据集　(b)ISOMAP

(c)带有最小子集覆盖的 Swiss-roll 数据集　(d)Fast-ISOMAP

图 3　$n=5000$ 的 Swiss-roll 数据集和二维嵌入结果



(a)Swiss-roll 数据集　(b)ISOMAP

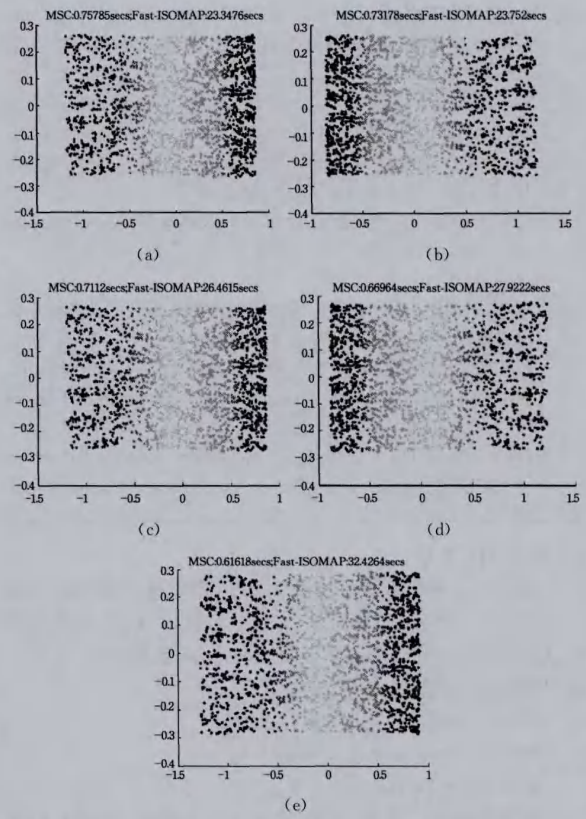
(c)带有最小子集覆盖的 Swiss-roll 数据集　(d)Fast-ISOMAP

图 4　$n=10000$ 的 Swiss-roll 数据集和二维嵌入结果

表 5　ISOMAP 和 Fast-ISOMAP 算法在不同规模 Swiss-roll 数据集上的计算时间比较

算法	n=2000 Swiss-roll	n=5000 Swiss-roll	n=10000 Swiss-roll
ISOMAP	346.8525s	3589.1846s	27855.1168s
Fast-ISOMAP	23.3476s	181.6289s	1280.6138s

下面通过实验来分析最小子集覆盖的重叠系数 α 对算法嵌入结果的影响。图 5 是重叠系数 α 取不同数值条件下，Fast-ISOMAP 算法应用于 $n=2000$ 的 Swiss-roll 数据集产生的二维嵌入结果。在图 5(a)－图 5(e)中，α 依次取 0.1,0.2,0.3,0.4 和 0.5；近邻数 k 固定为 25。从图中可以看出，重叠系数 α 的大小影响所选择的 Landmark 点的多少，随着 α 从 0.1 变化到 0.5，Landmark 点分别为 130、130、140、154 和 180，从而导致 Fast-ISOMAP 算法的计算时间从 23.3476s 提高到 32.4264s。但其二维嵌入结果并没有随重叠系数 α 取值的不同而发生显著变化。



(a)　(b)

(c)　(d)

(e)

图 5　重叠系数 α 取不同数值时 Fast-ISOMAP 算法在 $n=2000$ 的 Swiss-roll 数据集上的二维嵌入结果，从(a)－(e)α 依次取 0.1,0.2,0.3,0.4,0.5；$k=25$

结束语　本文提出了一种快速等距特征映射算法(Fast-ISOMAP)，该算法首先利用最小子集覆盖策略(MSC)从数据集中选择 p 个 Landmark 点；然后运用 Landmark MDS 方法计算所有样本的低维嵌入。该算法在不显著改变原始 ISOMAP 算法嵌入性能的条件下，大大提高了原始算法的计算效率，使它能够应用于大规模流形学习问题。在标准数据集上的实验结果验证了该方法的有效性。然而仍旧有一些问题值得进一步深入研究，如近邻数大小与最小子集覆盖之间的关系、算法的稳定性分析等问题。

参 考 文 献

[1] Tenenbaum J, de Silva V, Langford J. A global geometric framework for nonlinear dimensionality reduction[J]. Science, 2000, 290(5500): 2319-2323

[2] Roweis S, Saul L. Nonlinear dimensionality reduction by locally linear embedding[J]. Science, 2000, 290(5500): 2323-2326

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

[4] Donoho D, Grimes C. Hessian eigenmaps: new locally linear embedding techniques for high-dimensional data[J]. Proceedings of the National Academy of Sciences, 2003, 100(10): 5591-5596

[5] Weinberger K, Saul L. Unsupervised learning of image manifolds by semidefinite programming[C]// Proceedings of CVPR2004. 2004: 988-995

[6] Zhang Z, Zha H. Principal manifolds and nonlinear dimension reduction via local tangent space alignment[J]. SIAM Journal Scientific Computing, 2005, 26(1): 313-338

[7] Lin T, Zha H B. Riemannian manifold learning [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2008, 30 (5): 796-809

[8] Luo D J, Ding C, Nie F P, et al. Cauchy graph embedding[C]// Proceedings of ICML2011. 2011: 553-560

[9] Zhang Z Y, Wang J, Zha H Y. Adaptive manifold learning [J]. IEEE Trans. PAMI, 2012, 34(2): 253-265

[10] Qiao H, Zhang P, Wang D, et al. An explicit nonlinear mapping for manifold learning [J]. IEEE Trans. Cybernetics, 2013, 43 (1): 51-63

[11] 刘辉，杨俊安，王一. 基于流形学习的声目标特征提取方法研究 [J]. 物理学报，2011，60(7)：1-7
Liu H, Yang J A, Wang Y. A novel approach to research on feature extraction of acoustic targets based on manifold learning [J]. Acta Phys. Sin., 2011, 60(7): 1-7

[12] He X, Yan S, Hu Y, et al. Face recognition using Laplacianfaces [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005, 27(3): 328-340

[13] 刘利，韦佳，马千里. 基于流形学习的图像检索研究进展[J]. 北京交通大学学报，2010，34(5)：164-171
Liu L, Wei J, Ma Q L. State-of-the-art on image retrieval based on manifold learning[J]. Journal of Beijing Jiaotong University, 2010, 34(5): 164-171

[14] De Silva V, Tenenbaum J B. Global versus local methods in nonlinear dimensionality reduction[J]. Advances in Neural Information Processing Systems, 2003, 15: 721-728

[15] De Silva V, Tenenbaum J B. Sparse multidimensional scaling using landmark points[R]. Stanford, CA: Dept. Math., Stanford University, 2004

[16] Garey M R, Johnson D S. Computers and intractability: a guide to the theory of NP-completeness[M]. WH Freeman & Co. New York, NY, USA, 1979

[17] Karp R. Reducibility among combinatorial problems[M]// Complexity of Computer Computations. 1972: 85-103

(上接第 239 页)

由 where 子句的选择条件所得到的预选择结果关系是 $\{(0,[0.7,0.9])/t_1,(0,[0.5,0.7])/t_2,(1,1)/t_3\}$，这里元素的表示形式为 $(CN(t_i),CPoss(t_i))/t_i$。在此基础上，再进一步计算 Skyline 查询，结果如下：

$Poss(t_1)=[0.6,0.8]$，$N(t_1)=0$

$Poss(t_2)=[0.5,0.7]$，$N(t_2)=0$

$Poss(t_3)=1$，$N(t_3)=[0.3,0.5]$

将结果中的元素用 $(N(t),Poss(t))/t$ 表示。排序后的结果为：

$([0.3,0.5],1)/t_3 \succ (0,[0.6,0.8])/t_1 \succ (0,[0.5,0.7])/t_2$

结束语 本文基于 Vague 关系数据模型，讨论了 Vague 数据库 Skyling 查询的处理方法，该方法用于查询给定 Vague 关系中的任意元组确定不被该关系中的任意其它元组所支配的程度，并给出了相关的计算公式和实现算法。该实现算法直接对 Vague 数据库而不是分别对 Vague 数据库所对应的所有可能性状态进行操作，避免了模糊数据库所对应的可能性状态数量“组合爆炸”问题的发生，在很大程度上降低了 Skyling 查询计算过程的复杂性。在此基础上，还进一步给出了带有 where 子句的 Skyline 查询的计算方法。一些用于经典数据库[16]或概率数据库[12]或不完备数据库[13]的 Skyline 查询技术是否可以用于 Vague 数据库框架还需要进一步的研究。另外，在后续的研究中，我们计划以本文的研究工作为基础，采取批量查询的形式，在大规模模拟实例的基础上测试 Vague 数据库 Skyline 查询处理方法的有效性和效率。

参 考 文 献

[1] Zadeh L A. Fuzzy sets[J]. Information and Control, 1965, 8(3): 338-353

[2] Ma Z M, Mili F. Handling fuzzy information in extended possibility-based fuzzy relational databases [J]. International Journal of Intelligent Systems, 2002, 17(10): 925-942

[3] Bosc P, Pivert O. Modeling and Querying Uncertain Relational Databases: a Survey of Approaches Based on the Possible Worlds Semantics [J]. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 2010, 18(5): 565-603

[4] Gau W L, Buehrer D J. Vague sets[J]. IEEE Transactions on Systems, Man, and Cybernetics, 1993, 23(2): 610-614

[5] Lu A, Ng W. Vague sets or intuitionist fuzzy sets for handling vague data: which one is better [M]// Conceptual Modeling-Ek 2005. Springer, 2005: 401-416

[6] 郝忠孝，李松. Vague 时间段关系与 Vague 区域关系的表示和复合推理[J]. 计算机学报，2014，37(8)：1743-1753
Hao Zhong-Xiao, Li Song. Representation and Compound Reasoning of the Vague Temporal Interval Relations and the Vague Region Relations [J]. Chinese Journal of Computers, 2014, 37 (8): 1743-1753

[7] 欧阳春娟，李斌，李霞，等. 基于 Vague 集相似度量的图像隐写系统安全性测度[J]. 计算机学报，2012，35(7)：1510-1521
Ouyang Chun-juan, Li Bin, LI Xia, et al. A New Security Evaluation for Steganographic System Based on Vague Set Similarity Measure[J]. Chinese Journal of Computers, 2012, 35(7): 1510-1521

[8] 赵法信，马宗民，吕艳辉. 基于 Vague 数据库的代数查询语言 [J]. 小型微型计算机系统，2008，29(10)：1893-1899
Zhao Fa-xin, Ma Zong-min, Lv Yan-hui. Vague Databases Based Algebraic Query Language [J]. Journal of Chinese Computer Systems, 2008, 29(10): 1893-1899

[9] 赵法信，金义富. 基于异构双极信息的模糊查询研究[J]. 计算机科学，2013，40(7)：153-156，181
Zhao Fa-xin, Jin Yi-fu. Study on Fuzzy Query of Heterogeneous Bipolarity information[J]. Computer Science, 2013, 40(7): 153-156, 181

[10] Borzsonyi S, Kossmann D, Stocker K. The skyline operator [C]// Proc of the Int Conf on Data Engineering. Los Alamitos, CA: IEEE Computer Society, 2001: 421-430

[11] 王意洁，李小勇，杨永滔，等. 不确定 Skyline 查询技术研究[J]. 计算机研究与发展，2012，49(10)：2045-2053
Wang Yi-jie, Li Xiao-yong, Yang Yong-tao, et al. Reasearch on Uncertain Skyline Query Processing Technique[J]. Journal of Computer Research and Development, 2012, 49(10): 2045-2053

[12] Pei J, Jiang B, Lin X, et al. Probabilistic skylines on uncertain data[C]// Proc. of VLDB 2007. New York: ACM, 2007: 15-26

[13] Khalefa M E, Mokbel M F, Levandoski J J. Skyline query processing for incomplete data[C]// Proc. of ICDE 2008. Piscataway, NJ: IEEE, 2008: 556-565

[14] Pivert O, Prade H. Skyline Queries in an Uncertain Database Model Based on Possibilistic Certainty[C]// SUM 2014. 2014: 280-285

[15] Kießling W, Kostler G. Preference SQL- Design, implementation, experiences[C]// Proc. of VLDB 2002. 2002: 990-1001

[16] Bartolini I, Ciaccia P, Patella M. Efficient sort-based skyline evaluation[J]. ACM Transaction on Database Systems, 2008, 33 (4): 1-49