

基于局部线性嵌入算法的流量矩阵流形结构分析

陈 玄,殷保群,石 浩

(中国科学技术大学 自动化系,合肥 230027)

摘 要: 利用经典流形学习算法研究流量矩阵中的流形结构,能够获得流量矩阵的本征维度。然而局部线性嵌入(LLE)算法依赖于近邻点的选取,传统近邻选取个数往往通过实验试凑法得到最优解,不能完全揭示流量矩阵的流形结构。针对上述缺点,提出一种改进的局部线性嵌入算法,该算法利用改进的 LLE 算法探索流量矩阵的流形结构,并对实际骨干网络中的流量矩阵进行分析。实验结果证明,改进算法具有较小的重构误差,相对于标准 LLE 算法,能更为准确地揭示流量矩阵的低维流形结构。

关键词: 流量矩阵;局部线性嵌入算法;本征维度;流形结构;骨干网络

中文引用格式: 陈 玄,殷保群,石 浩. 基于局部线性嵌入算法的流量矩阵流形结构分析[J]. 计算机工程,2017,43(3):40-45.

英文引用格式: Chen Xuan, Yin Baoqun, Shi Hao. Manifold Structure Analysis of Traffic Matrix Based on Local Linear Embedding Algorithm[J]. Computer Engineering, 2017, 43(3):40-45.

Manifold Structure Analysis of Traffic Matrix Based on Local Linear Embedding Algorithm

CHEN Xuan, YIN Baoqun, SHI Hao

(Department of Automation, University of Science and Technology of China, Hefei 230027, China)

[Abstract] The manifold structure of the traffic matrix is studied by using classical manifold learning algorithms, and the intrinsic dimension of the flow matrix can be got. However, the Locally Linear Embedding (LLE) algorithms rely on the selection of neighbor points. The traditional method is often obtained by experiment. Even so, this does not fully reveal the manifold structure of traffic matrix. Aiming at the above shortcomings, this paper improves the LLE algorithm by improving the LLE algorithm. It applies the improved LLE algorithm to real OD traffic matrix taken from the backbone network (Abilene). Experimental result shows the improved LLE algorithm has a smaller reconstruction error, compared with standard LLE algorithm, it can reveal a low-dimensional manifold structure exactly.

[Key words] traffic matrix; Locally Linear Embedding (LLE) algorithm; intrinsic dimensionality; manifold structure; backbone network

DOI: 10.3969/j.issn.1000-3428.2017.03.007

0 概述

随着计算机技术以及计算机网络的飞速发展,涌现出大量关于网络流量的相关研究,涉及到网络流量的监测、分析、预测和控制。流量矩阵(Traffic Matrix)表示了网络中任意 OD (Origin-Destination) 对(流或节点)之间的流量,描述了网络流量在各个 OD 对间的分布情况。OD 节点可以是链路(Link-to-Link)、路由器(Router-to-Router),也可以是 POP (Point-to-Point)。针对不同的 OD 对,流量矩阵可以

是基于链路、路由器或者 PoP 的不同流量矩阵,从而流量矩阵可以提供不同的应用,为人们进行与网络流量相关的管理、监控工作提供有力的参考依据。文献[1-2]先后阐述了流量矩阵获取的各种方法并指明了其发展方向,本文工作的重点是流量矩阵的分析。迄今为止,文献[3]使用主成分分析(Principal Component Analysis, PCA)的方法来分析流量矩阵,利用 PCA 对一个庞大的流量矩阵进行降维处理,通过舍去较小的重构误差来还原原始流量矩阵。利用 PCA 的特性,从原始流量矩阵中提取出主要信息、噪

基金项目: 高等学校博士点专项科研基金(20123402110029);安徽省高等学校自然科学基金重点项目(KJ2012A286)。

作者简介: 陈 玄(1990—),男,硕士研究生,主研方向为流量矩阵分析;殷保群,教授、博士;石 浩,博士研究生。

收稿日期: 2016-01-29 **修回日期:** 2016-03-28 **E-mail:** cxzuoer@mail.ustc.edu.cn

声信息等,进一步比较了这些不同的信息在原始流量矩阵中所占的比例,并通过这些信息来推测流量的异常。另外一种方法是多维尺度分析法(MRA)^[4],即基于扩散小波变换的分析方法,基于特定的扩散小波算子,能够利用小波变换的多尺度建模能力,将原始的流量矩阵分解为大小不同的系数矩阵。

文献[3-4]采用的 PCA 方法和 MRA 方法均属于线性降维的方法,它们旨在寻找存在于高维输入空间中的最佳线性低维嵌入空间,然而真实的因特网流量矩阵高维数据本质上是否存在更为复杂的非线性流形结构成为众多学者的研究方向,采用的方法有 LLE^[5]算法和 E-Isomap^[6]算法等。不难发现, LLE 算法^[7]中近邻选择算法强制为每个数据点选择相同个数的近邻点或认为一定半径超球内的点全部算作近邻,这样做降维后的结果千差万别,错误的选择近邻不能正确揭示高维数据流形结构。本文针对以上问题拟采用改进的 LLE 算法来分析 OD 流量矩阵的非线性流形结构。改进 LLE 算法可以分为以下 3 种类别:1) 紧邻选取时用其他距离代替欧氏距离,文献[8]提出图像欧氏距离的概念(IMED),类似的改进还有文献[9];2) 近邻选择时将样本点进行分,分为一个个样本,考虑样本的近邻,变成有监督的局部算法,典型的有文献[10-11]提出的改进算法;3) 新样本降维方式的改进,文献[12]对此类方法及进行的改进作出了详细的说明。

本文提出一种有效的近邻选取算法,区别于固定的近邻个数和近邻范围,在给定的范围内根据数据本身之间的关系,剔除不属于样本近邻的点从而确定样本近邻,并给出改进的 LLE 算法,最终分析得出流量矩阵的低维流形结构。

1 系统建模

定义 1(流量矩阵) 假设某自治系统(Autonomous System,AS)有 n 个 PoP 点,以一定的时间间隔(周期)连续地被动测量任意一对 PoP 点之间的流量,然后将获得的测量值排列成一个 $T \times p$ (T 为测量的周期数, p 表示每个周期内获得的流量测量值的个数 $p = n \times n$) 的矩阵 X ,它表示所有这些流量测量值的时间序列。其中,第 i 行表示在第 i 个周期内流量测量值的向量;第 j 列表示第 j 个 PoP 点对之间流量测量值的时间序列;矩阵 X 即为自治系统 PoP 级流量矩阵。

本节将对骨干网络进行系统建模,构成流量矩阵。一条 OD 流包含了一个给定起始节点到一个给定目的节点间的所有流量。用 n 表示一片局域

网中所有的起始节点和目的节点个数,流量矩阵可以由一个三维的非负超矩阵 $X(t)$ 表示。矩阵中每一项为流量,单位是字节或者包,用 $X_{i,j}(t)$ 表示。 $X_{i,j}(t)$ 具体含义是从起始节点 i 到目的节点 j 在时间间隔 $[t, t + \Delta t) \subset T$ 内的流量 (T 表示总的测量时间), Δt 由应用程序和可用的测量方法决定^[13]。事实上,任意节点之间的实时流量难以测量,因此,测量值是离散时间间隔内的流量平均值。本文实验采用的数据集是从阿比林(Abilene)骨干网络中采样得到的,采样时间间隔为 5 min。如图 1 所示左边为抽象的流量矩,右边为一个简单的网络节点模型。

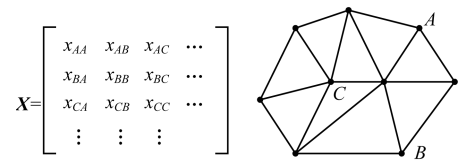


图 1 一个简单流量矩阵的抽象表示

本文将对实际骨干网络中流量数据进行分析,文中数据集下载的网址为:<http://www.cs.utexas.edu/~y Zhang/>。采样环境是一个 12 个 PoPs 的典型网络,因此网络中有 144 个 PoP-对以及 144 个 OD 流。本文选择了 X05, X11 以及 X14 作为最终的实验数据集,每一个数据集包含了 $12 \times 24 \times 7 = 2\ 016$ 个 5 min 的流量矩阵。

2 算法介绍

2.1 本征维度分析

文献[14]通过主成分分析法分析了 Internet 实测的 OD 流流量矩阵数据的维度和结构,证实其存在的低维特性,根据其特征流的类型和构成,解释了流量矩阵具有的空间特性。文献[5-6]也证实了这些流量矩阵低维特性的结论,并对流量矩阵进行维度分析,寻找流量矩阵本征维度。他们采用的方法是利用 Isomap 的残差分析给出残余方差和维数的曲线图,找出了数据的本征维度。Isomap 的主要思想是:计算流形上测地线距离,然后应用 MDS 算法,计算出嵌入在高维空间的低维坐标,这样就可以通过数据间的测地线距离,保留了数据固有几何分布。具体算法本文不再赘述。

图 2 给出了应用 Isomap 算法得到的残余方差和维数的关系。当维数增加时,残余方差在不断地降低,而在 $d = 5$ 时增加维度,残余方差变化越来越缓慢,因此,可以得出的流量矩阵的本征维度为 5。

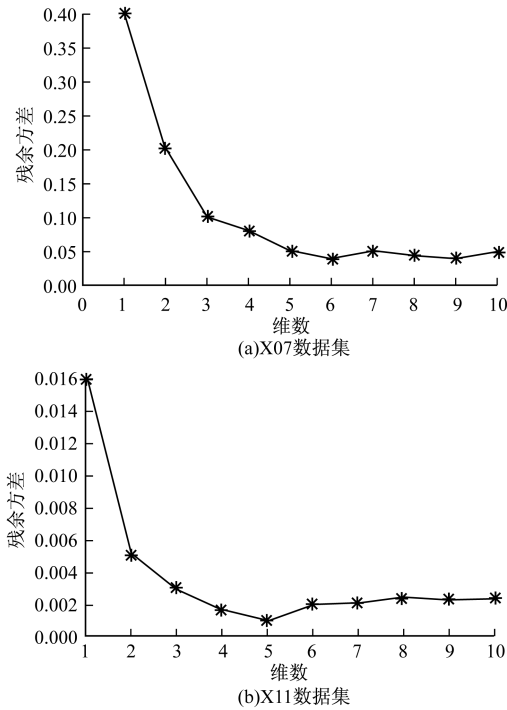


图2 X07和X11残余方差与维度关系

2.2 LLE基础算法

现有由 N 个数据点组成数据集 X , 假设该数据集位于某一高维非线性流形上, 每个数据都是 D 维的, 可以用实值向量 x_i 表示。LLE 算法认为每一个数据点以及该数据点的近邻位于或者近似位于一个局部的超平面上, 所以每一个数据点都可以由其近邻进行线性重构, 求出每个数据点的近邻, 并计算每个近邻的重构权重。如果该数据集位于或近似位于一个维度 $d \ll D$ 的非线性流形上, 那么就存在将高维数据映射到全局低维坐标系上映射。不难发现, 重构权重反映的是数据集的内在几何性质, 因此, 原始数据的局部几何特征在映射到低维流形上同样有效, 根据此约束可求得映射后的每一点的全局坐标。

算法1 LLE算法

步骤1 计算每个数据点 x_i 的近邻。

步骤2 根据每个数据点 x_i 的近邻, 计算重构每个数据点 x_i 的重构权重 W_{ij} 。

步骤3 根据每个数据点的重构权重 W_{ij} , 计算低维嵌入向量 y_i 。

在步骤1中, 求解 x_i 的近邻, 有2种方法: 1) 计算其他所有数据点与 x_i 的距离, 选取其中 K 个最小值对应的数据点, 即为 K 个近邻; 2) 选取半径为 δ 的超球内的所有数据点作为 x_i 的近邻。

在步骤2中, 重构权重 W_{ij} 必须使得重构误差最小化, 即必须使得式(1)最小。

$$\varepsilon(W) = \sum_i |x_i - \sum_j W_{ij} x_j|^2 \quad (1)$$

重构权重 W_{ij} 表示了数据点 x_i 的第 j 个近邻在重构时所占的权重, 同时为了最小化重构误差必须满足2个约束条件: 每个数据点 x_i 完全由其近邻重构, 首先若对 $\forall x_i \in X$, x_j 不是 x_i 的近邻, 则有 $W_{ij} = 0$; 其次, 对 $\forall x_i \in X$, 若 x_j 为 x_i 的近邻, 则有 $\sum_j W_{ij} = 1$ 。

在步骤3中, 每个高维向量 x_i 被映射到低维向量 y_i 上, 且 y_i 可用低维流形的全局坐标表示。可以通过最小化嵌入代价式(2)来计算。

$$\Phi(Y) = \sum_i |y_i - \sum_j W_{ij} y_j|^2 \quad (2)$$

式(1)、式(2)的依据同样是局部线性重构误差, 而 y_i 可以通过计算一个 $N \times N$ 的稀疏矩阵得到。具体计算步骤不在赘述, 可参考文献[7]。

2.3 LLE改进算法

实际上, LLE 算法并不要求原始高维数据集能被一个单一坐标系表示出来, 只要求数据集中的每一个数据点能由它的近邻点重构。LLE 仅仅依赖于数据之间的几何特性, 并将数据间几何特性保留下来, 与数据之间的关系无关。

LLE 算法通过局部线性拟合来获得内在的全局线性结构。可以看出 LLE 有2个不定参数: 第1个是最近的 K 个近邻点, 或者半径为 δ 的超球; 第2个是本征维度 d 。通过前面的分析, 本文找出高维非线性流量矩阵数据的本征维度, 即已确定了 d , 所以只剩下一个需要设定的参数。

选择半径为 δ 的超球作为参数比选择 K 个近邻点作为参数, 时间复杂度和空间复杂度都要复杂得多, 并且比较难以计算。关于 K 值的选择, 如果 K 值过小, 可能会导致孤立点的存在, 即有些数据点没有近邻; 如果 K 值过大, 会将距离较远的数据点强行看作局部线性, 不能得到理想的降维效果(在本文实验部分能清楚地看到不同的 K 值对实验结果的影响)。实际上, K 值的选取往往是通过实验试凑出来的, 不难想象, 固定的 K 不能很好地发现数据的本质结构, 结合相关研究, 本文提出一种改进的 LLE 算法, 为每个数据点自动选取合适的近邻再进行降维分析, 具体算法介绍如下所示。

在为某数据点 x_i 选择 K 个近邻以后, 则 x_i 可通过如下公式进行重构: $x_i = \sum_j^k W_{ij} x_j$ 。进一步选择近邻之前, 规定一个近邻判定函数 $y = f(d_{ij})$, 判定 x_j 是否为 x_i 的近邻。此近邻判定函数自变量为数据点

之间的距离 $d_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|$, 且必须符合以下条件: $f(d_{ij})$ 必须单调递减, 当 d_{ij} 趋近于 0 时, $f(d_{ij})$ 越大; 当 d_{ij} 趋近于无穷大时, $f(d_{ij})$ 趋近于 0。也就是说当 \mathbf{x}_i 和 \mathbf{x}_j 距离越近, \mathbf{x}_j 在重构 \mathbf{x}_i 时的贡献越大, 反之, 当 \mathbf{x}_i 和 \mathbf{x}_j 距离越远, \mathbf{x}_j 贡献越小。因此, 选择距离的倒数作为近邻判定函数。

算法 2 近邻选择算法

步骤 1 为每个数据点 \mathbf{x}_i 选择 K 个近邻, 分别计算近邻与数据点的距离 $d_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\| = (\mathbf{x}_i - \mathbf{x}_j)^2$ 。

步骤 2 计算近邻判定函数值, $y = f(d_{ij}) = \frac{1}{d_{ij}}$ 。

步骤 3 令 $F = \frac{1}{N \times K} \sum_{i=1}^N \sum_{j=1}^K \frac{1}{d_{ij}}$ 作为所有样本的平均阈值。

即任意数据点 \mathbf{x}_j 与 \mathbf{x}_i 的近邻判定函数值小于 F , 则其为 \mathbf{x}_i 的近邻。在实验仿真时, 选择在最近的 $[K-5, K+5]$ 个近邻点中进行判定, 这样可以保证每个数据点至少有 $K_{\min} = K-5$ 个近邻点, 最多不超过 $K_{\max} = K+5$ 个近邻点。虽然控制了紧邻点的范围, 但是不同的数据点的近邻数不尽相同, 这就使得 LLE 基础算法求解 W_{ij} 不适用, 下面给出严格的数学推导证明来求解 W_{ij} 以及嵌入向量 \mathbf{y}_i 。

$$\begin{aligned} \varepsilon(\mathbf{W}_i) &= \|\mathbf{x}_i - \sum_{j \in K_i(1)}^{K_i(n_i)} W_{ij} \mathbf{x}_j\| \\ C &= \begin{bmatrix} C_i(K_i(1), K_i(1)) & C_i(K_i(1), K_i(2)) \cdots & C_i(K_i(1), K_i(n_i)) \\ C_i(K_i(2), K_i(1)) & C_i(K_i(2), K_i(2)) \cdots & C_i(K_i(2), K_i(n_i)) \\ \vdots & \vdots & \vdots \\ C_i(K_i(n_i), K_i(1)) & C_i(K_i(n_i), K_i(2)) \cdots & C_i(K_i(n_i), K_i(n_i)) \end{bmatrix} \end{aligned}$$

由式(6)可以计算得到:

$$\overline{\mathbf{W}}_i^T = \overline{\mathbf{C}} - 1\overline{\mathbf{q}} \quad (7)$$

当近邻点的个数大于数据集本身的维度时, 即 $K > D$ 这一特殊情况时, 矩阵 \mathbf{C} 可能是奇异的, 必须通过: $\mathbf{C} \leftarrow \mathbf{C} + \eta_c \mathbf{I}$ 进行正则化, \mathbf{I} 为 $n_i \times n_i$ 的单位矩阵, η_c 为一个很小的常数。正则化后保证 $\overline{\mathbf{C}}$ 满秩。为了消除 η_c 对矩阵 $\overline{\mathbf{C}}$ 的影响, 与 η_c 矩阵 \mathbf{C} 的迹值相比较, 其取值必须很小。

最终通过使用 Rayleitz-Ritz 定理^[15] 计算低维嵌入向量 \mathbf{Y} , 如式(8)所示。

$$\mathbf{Y} = [\mathbf{y}_1 \quad \mathbf{y}_2 \quad \cdots \quad \mathbf{y}_N] = [\mathbf{v}_1 \quad \mathbf{v}_2 \quad \cdots \quad \mathbf{v}_{n_d}]^T \quad (8)$$

其中, $[\mathbf{v}_1 \quad \mathbf{v}_2 \quad \cdots \quad \mathbf{v}_{n_d}]$ 是正定对称矩阵 $\mathbf{M} = (\mathbf{I} - \mathbf{W})^T (\mathbf{I} - \mathbf{W})$ 的特征向量。

$$\begin{aligned} &= \left\| \sum_{j \in K_i(1)}^{K_i(n_i)} W_{ij} \mathbf{x}_i - \sum_{j \in K_i(1)}^{K_i(n_i)} W_{ij} \mathbf{x}_j \right\| \\ &= \sum_{j \in K_i(1)}^{K_i(n_i)} W_{ij} \sum_{m \in K_i(1)}^{K_i(n_i)} W_{im} (\mathbf{x}_i - \mathbf{x}_j)^T (\mathbf{x}_i - \mathbf{x}_m) \end{aligned} \quad (3)$$

其中, \mathbf{W} 是重构向量矩阵; W_i 表示 \mathbf{W} 的行向量; $\varepsilon(\mathbf{W}_i)$ 表示重构误差。

定义 $C_i(j, m) = (\mathbf{x}_i - \mathbf{x}_j)^T (\mathbf{x}_i - \mathbf{x}_m)$, 在式(3)中应用拉格朗日乘子, 得到式(4)。

$$\begin{aligned} \varepsilon(\mathbf{W}_i) &= \sum_{j \in K_i(1)}^{K_i(n_i)} W_{ij} \sum_{m \in K_i(1)}^{K_i(n_i)} W_{im} C_i(j, m) \\ &+ \eta_i \left(\sum_{j \in K_i(1)}^{K_i(n_i)} W_{ij} - 1 \right) \end{aligned} \quad (4)$$

其中, η_i 是拉格朗日相关系数, 对 $W_{iK(j)}$ 求偏导, 得:

$$\begin{aligned} \frac{\partial \varepsilon(\mathbf{W}_i)}{\partial W_{iK(j)}} &= 2 \sum_{m \in K_i(1)}^{K_i(n_i)} W_{im} C_i(K_i(1), m) \\ &+ \eta_i, \quad \forall j \in K_i \end{aligned} \quad (5)$$

令 $\frac{\partial \varepsilon(\mathbf{W}_i)}{\partial W_{iK(j)}} = 0$, 又因为对 $\forall j \in K_i$, 有 $\sum_{j \in K_i(1)}^{K_i(n_i)} W_{ij} = 1$, 则可以得到式(6)。

$$\overline{\mathbf{C}} \overline{\mathbf{W}}_i^T = \overline{\mathbf{q}} \quad (6)$$

其中:

$$\overline{\mathbf{C}} = \begin{bmatrix} 0 & 1 \\ 1 & 2C \end{bmatrix}, \quad \overline{\mathbf{q}}^T = [1 \quad 0 \quad 0 \quad \cdots \quad 0]^T$$

$$\begin{aligned} \overline{\mathbf{W}}_i^T &= [\eta_i \quad W_{iK_i(1)} \quad W_{iK_i(2)} \quad \cdots \quad W_{iK_i(n_i)}]^T \\ &= [\eta_i \quad \mathbf{W}_i^T]^T \end{aligned}$$

2.4 对比分析

根据上述算法给出标准 LLE 算法和改进算法的时间复杂度的对比, 如表 1 所示。

表 1 标准算法和改进算法的时间复杂度

方法	标准 LLE 算法	改进 LLE 算法
近邻点	$O(DN^2)$	$O(DN^2 + N^3)$
重构权重	$O(DNK^3)$	$O(DNK^3)$
低维嵌入向量	$O(dN^2)$	$O(dN^2)$

不难发现重构误差显然大于 0, 令:

$$r = \frac{\varepsilon(\mathbf{W}')}{\varepsilon(\mathbf{W})} \quad (9)$$

其中, r 表示重构误差比率; $\varepsilon(\mathbf{W})$ 表示标准 LLE 算法的重构误差; $\varepsilon(\mathbf{W}')$ 表示改进算法的重构误差。应用 LLE 标准算法和改进 LLE 算法对系统中流量矩阵进行降维分析, 表 2 给出了不同 K 值时, 重构误差比率的值。

表 2 不同数据集上降维分析的重构误差比率

K 值	X05	X11	X14
12	0.276 6	0.905 8	0.615 3
15	0.263 1	1.478 8	1.051 3
18	0.371 1	0.710 1	0.613 3
21	0.593 1	0.578 9	0.687 0
24	0.668 1	0.695 0	0.477 8

综上所述,可以发现改进的 LLE 算法相较于标准 LLE 算法,时间复杂度增加;改进算法的重构误差基本小于标准算法的重构误差,一般为标准算法重构误差的 60%。

LLE 的基本思想就是保持原流形中局部近邻间相互关系,将非线性流形的局部近似看作线性的,再

将高维数据集映射到一个低维的单一全局坐标系上。本文仍然是基于数据本身之间的距离关系,改进 K 值选取算法,提出改进的 LLE 算法,但是并没有作分类,因此它仍然是一种无监督的方法,与类标签局部线性嵌入算法 CLLE^[10] 和邻域线性嵌入算法 NLE^[11] 不同,CLLE 和 NLE 两者都是有监督的方法。

3 实验结果与分析

实验 1 对标准数据集 Twinpeaks 应用标准 LLE 算法进行降维分析,通过实验对比发现,选取不同的 K 值,降维效果不同。在实验中,在 Twinpeaks 标准数据集上取样 $N = 1\ 500$ 个数据点,实验结果如图 3 所示。

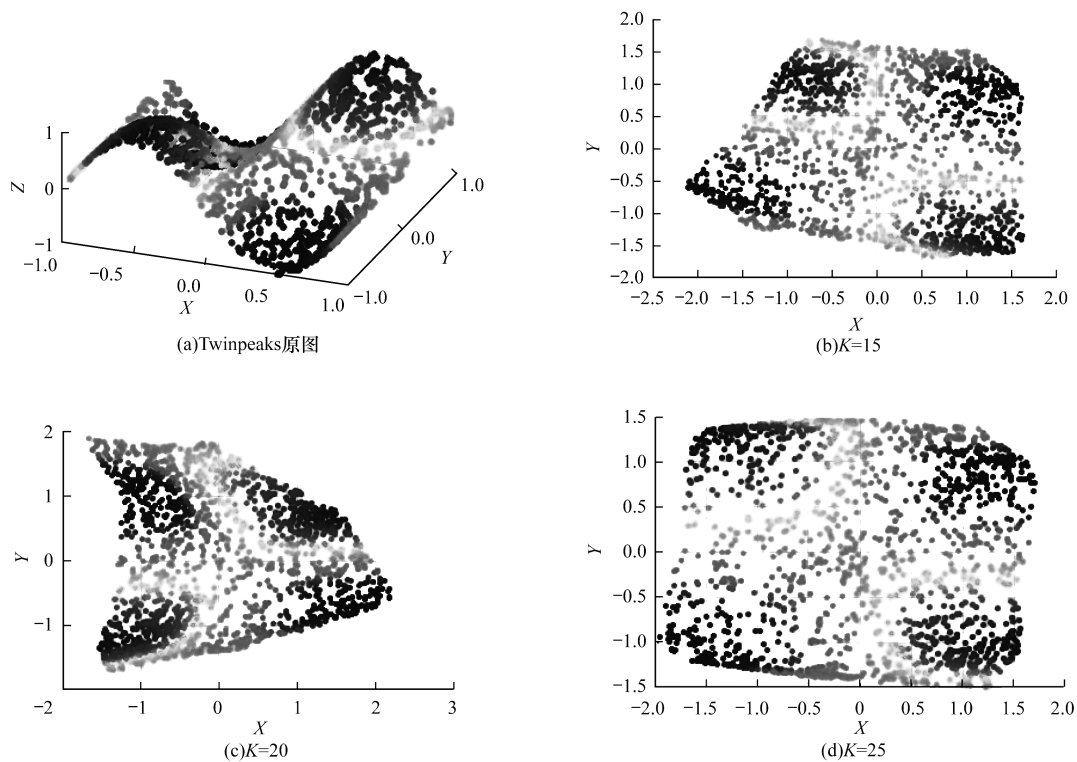


图 3 Twinpeaks 降维分析结果

实验 2 用标准 LLE 算法和改进 LLE 算法分别对 X05, X11 以及 X14 数据集上的流量矩阵进行降维分析。通过应用 LLE 算法和改进的 LLE 算法来探索流量矩阵中存在的流形结构,在

本文 2.1 节中已经得出流量矩阵的本征维度为 5,但是一个 5 维空间并不能被直观展示出来,因此,给出 2 维和 3 维的嵌入结果。结果如图 4、图 5 所示。

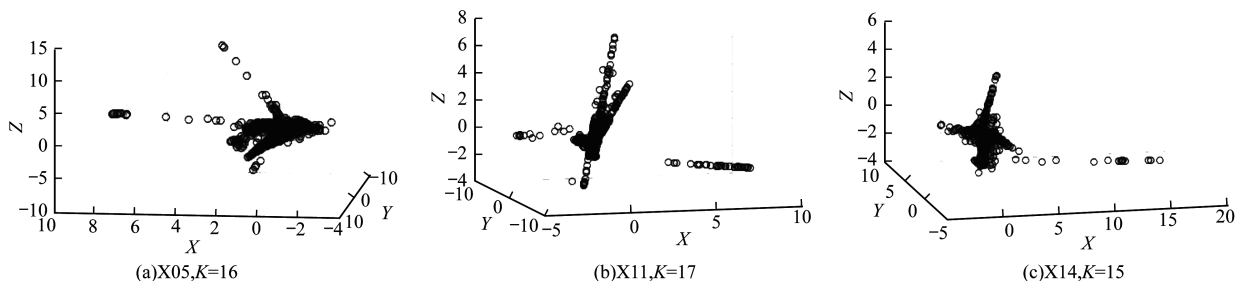


图 4 应用 LLE 标准算法对 X05, X11, X14 流量矩阵的降维分析

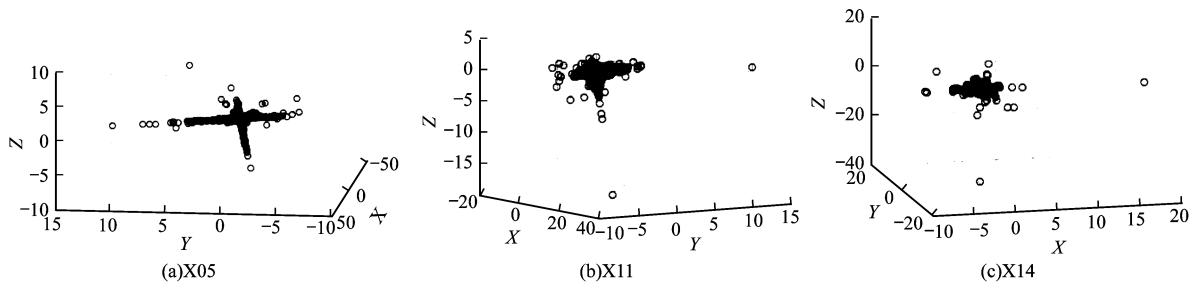


图 5 应用改进 LLE 算法对 X05, X11, X14 流量矩阵的降维分析

从实验结果中可以看出降维后孤立点的存在,分析可得到当前骨干网络中存在的异常。在一段时间内,对同一骨干网络不同的流量矩阵进行降维分析对比,可以优化网络结构。同时,实验中的几组流量矩阵数据集在低维嵌入空间中有着迥异结构,这说明了流量矩阵不同采样时间之间的区别。

根据上述的实验可以清楚地看到高维流量矩阵中确实存在着低维的流形结构,通过分析这些流形结构,实际网络流量中的一些重要的信息能被观测到,这些信息包含了流量变化的趋势以及网络异常的发生。LLE 算法具有旋转、变换、平移的不变性,虽然改进 LLE 算法和标准 LLE 算法降维结果迥然,实际上结构类似,并且改进 LLE 算法降维后相对于标准算法结构更加清晰。

4 结束语

流量矩阵本身的维数比较高,本文利用改进的 LLE 算法对流量矩阵进行降维分析。该算法拥有较小的重构误差,更为准确地揭示了流量矩阵的低维流形结构。实验结果表明,该算法易于观测流量矩阵中出现流量异常的节点,同时相对于标准 LLE 算法更能清晰地表明流量矩阵的低维流形结构和流量发展的趋势。骨干网异常或者拥塞引起的流量变化因流量矩阵维数非常高而比较细微,难以察觉,通过降维分析而变得清晰。本文改进算法发现了不同流量矩阵数据集在低维投影空间展示的非线性结构。流量矩阵利用流量特征为研究互联网提供了一个整体的角度。因此,下一步将在了解网络特性和网络状态的情况下,排除网络故障,改善网络拥塞,优化网络拓扑结构。

参考文献

- [1] 张宏莉,方滨兴,胡铭曾,等. Internet 测量与分析综述[J]. 软件学报,2003,14(1):110-116.
- [2] 蒋定德,胡光岷. 流量矩阵估计研究综述[J]. 计算机科学,2008,35(4):5-9.

- [3] Lakhina A, Papagiannaki K, Crovella M, et al. Structural Analysis of Network Traffic Flows [M]. New York, USA: ACM Press, 2004.
- [4] Rincon D, Roughan M, Willinger W. Towards a Meaningful MRA of Traffic Matrices [C] // Proceedings of ACM SIGCOMM Conference on Internet Measurement. New York, USA: ACM Press, 2008: 331-336.
- [5] 钱叶魁,陈 鸣. 因特网流量矩阵的流形结构 [J]. 电子与信息学报, 2010, 32(12): 2981-2986.
- [6] Shi Hao, Yin Baoqun, Qian Yequi, et al. Manifold Structure Analysis of Internet Traffic Matrix Based on E-Isomap [C] // Proceedings of CCC' 14. Nanjing, China: [s. n.], 2014: 5498-5503.
- [7] Roweis S T, Saul L K. Nonlinear Dimensionality Reduction by Locally Linear Embedding [J]. Science, 2000, 290(5500): 2323-2326.
- [8] Wang Liwei, Zhang Yan, Feng Jufu. On the Euclidean Distance of Images [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005, 27(8): 1334-1339.
- [9] Zhang Lijing, Wang Ning. Locally Linear Embedding Based on Image Euclidean Distance [C] // Proceedings of International Conference on Automation and Logistics. Jinan, China: [s. n.], 2007: 1914-1918.
- [10] de Ridder D, Kouropteva O, Okun O, et al. Supervised Locally Linear Embedding [M]. Berlin, Germany: Springer, 2013.
- [11] Zhang Shiqing. Enhanced Supervised Locally Linear Embedding [J]. Pattern Recognition Letters, 2009, 30(13): 1208-1218.
- [12] Saul L K, Roweis S T. Think Globally, Fit Locally: Unsupervised Learning of Low Dimensional Manifolds [J]. Journal of Machine Learning Research, 2004, 4(2): 119-155.
- [13] Zhang Yin, Roughan M, Willinger W, et al. Spatio-temporal Compressive Sensing and Internet Traffic Matrices [J]. ACM SIGCOMM Computer Communication Review, 2009, 39(4): 267-278.
- [14] Tenenbaum J B, de Silva V, Langford J C. A Global Geometric Framework for Nonlinear Dimensionality Reduction [J]. Science, 2000, 290(5500): 2319-2323.
- [15] Horn R A, Johnson C R. Matrix Analysis [M]. Cambridge, UK: Cambridge University Press, 2012.

编辑 索书志