Selection of the Suitable Parameter Value for ISOMAP

Li Jing and Chao Shao
School of Computer and Information Engineering, Henan University of Economics and Law, Zhengzhou 450002, China
Email: {jingli7762003, sc_flying0527}@yahoo.com.cn

***Abstract*—As a promising dimensionality reduction and data visualization technique, ISOMAP is usually used for data preprocessing to avoid "the curse of dimensionality" and select more suitable algorithms or improve the performance of algorithms used in data mining process according to No Free Lunch (NFL) Theorem. ISOMAP has only one parameter, i.e. the neighborhood size, upon which the success of ISOMAP depends greatly. However, it's an open problem how to select a suitable neighborhood size efficiently. Based on the unique feature of shortcut edges, introduced into the neighborhood graph by using the unsuitable neighborhood size, this paper presents an efficient method to select a suitable neighborhood size according to the decrement of the sum of all the shortest path distances. In contrast with the straightforward method with residual variance, our method only requires running the former part of ISOMAP (shortest path computation) incrementally, which makes it less time-consuming, while yielding the same results. Finally, the feasibility and robustness of this method can be verified by experimental results well.**

***Index Terms*—data visualization, ISOMAP, geodesic distance, shortest path distance, neighborhood size, residual variance**

I. INTRODUCTION

Nowadays, the explosive growth in the amount of data and their dimensionality makes data visualization more and more important in data mining process. According to No Free Lunch (NFL) Theorem, the data information should be taken into account to select more suitable data analysis/processing algorithms for data mining. For high-dimensional data, the useful distribution and structure information cannot be seen by eyes directly, but it can be obtained by data visualization approaches easily.

During the last hundreds of years, lots of approaches to visualize high-dimensional data have been emerged, most of which fall into the following five categories: 1) use several sub-windows to represent visually different subsets of the dimensions respectively, such as scatterplot matrices[1] and pixel-oriented techniques[2]; 2) rearrange the dimension axes in the low-dimensional space, such as parallel coordinates[3] and star coordinates[4]; 3) embed the dimensions to partition the low-dimensional space hierarchically, such as dimensional stacking[5] and treemap[6]; 4) use certain objects with several visual features, each of which stands for one dimension, such as Chernoff-faces[7] and stick figures[8]; 5) reduce the dimensionality of the data to two or three dimensions using dimensionality reduction techniques, such as PCA (Principal Component Analysis)[9], MDS (Multidimensional Scaling)[10], SOM (Self-Organizing Map)[11], ISOMAP (Isometric Mapping)[12][13][14], LLE (Locally Linear Embedding)[15][16] and Laplacian Eigenmap[17], *etc*.

Unlike other approaches, dimensionality reduction techniques try to preserve the high-dimensional relationship between the data in the low-dimensional space, which can represent visually the distribution and structure of the data very well. In addition, dimensionality reduction techniques can avoid "the curse of dimensionality" and improve the performance of algorithms used in data mining process. As one of non-linear dimensionality reduction techniques, ISOMAP, a variant of MDS (Multi-dimensional Scaling)[12], preserves global geodesic distances between the data in the low-dimensional embedded space, and thus can visualize the convex but intrinsically flat manifolds such as the Swiss-roll and S-curve data sets nicely. Its ability to preserve the global geometric structure of manifolds in the non-iterative way makes it more and more attractive in data preprocessing[16], in addition, one of its main advantages is that only one parameter, i.e. the neighborhood size upon which the success of ISOMAP depends greatly[18], is required.

Like other manifold learning techniques such as LLE[15] and Laplacian Eigenmap[17], it's very important for ISOMAP to select a suitable neighborhood size, which means a good trade-off between locality and globality, however, this is an open problem. It's well known that the neighborhood size should be neither so large that the neighborhood graph contains shortcut edges (which connect two data points with rather large distance along the manifold but relatively small Euclidean distance and thus don't lie on the manifold) so as not to represent the neighborhood structure of the data correctly, nor so small that the graph becomes disjoined or sparse so as not to approximate geodesic distances between the data accurately[18]. A straightforward method is to select a suitable neighborhood size through estimating the

Manuscript received November 1, 2010; revised December 15, 2010; accepted January 15, 2011.
Corresponding author: Li Jing, jingli7762003@yahoo.com.cn.

doi:10.4304/jsw.6.6.1034-1041

"quality" of the corresponding mapping measured by residual variance[12]. So this method requires running the whole ISOMAP algorithm with every possible neighborhood size, which makes it very time-consuming. In this paper, we present an efficient method to select a suitable neighborhood size, in which only the former part of the ISOMAP algorithm, i.e. shortest path computation, is required to run incrementally.

This paper is organized as follows: In Section II, ISOMAP and the straightforward method with residual variance are recalled briefly. In Section III, our method is described in detail. In Section IV and Section V, experimental results and conclusions are given respectively.

II. ISOMAP and the Straightforward Method with Residual Variance

A. ISOMAP

When the global geometric structure of high-dimensional data is unknown, we are not sure that the Euclidean distance metric is suitable to represent the dissimilarity relationship between the data. Fortunately, the Euclidean distance metric is trustworthy enough to represent the dissimilarity relationship between the data within a small enough neighborhood, which is also called the local Euclidean nature of the manifold. So the global geometric structure can be approximated using the local Euclidean distance information, as ISOMAP[12] does.

If the data lies on a single well-sampled manifold, it's proved that the unknown global geodesic distances between the data can be well approximated by the corresponding shortest path distances in the suitable neighborhood graph which represents the right neighborhood structure of the data[20]. After using the geodesic distance metric instead of the Euclidean distance metric, ISOMAP uses the classical MDS algorithm to map the data into the low-dimensional embedded space. So ISOMAP can be described briefly as follows[12][13]:

1) Select *n* representative data points randomly or using vector quantization (with better results[21]) for very large data sets to keep subsequent computation tractable;

2) Construct a suitable neighborhood graph (connected for data visualization) using the *K* nearest neighbors method with a suitable neighborhood size *K* (*K* is more natural to choose than ε [22]);

3) Compute all the shortest path distances in the neighborhood graph;

4) Use the classical MDS algorithm to map the data into the low-dimensional embedded space.

B. The Straightforward Method with Residual Variance

Given the data lying on a single well-sampled convex but intrinsically flat manifold such as the Swiss-roll and S-curve data sets, the success of ISOMAP depends upon selecting a suitable neighborhood size, with which the neighborhood graph can represent the right neighborhood structure of the data and thus geodesic distances can be approximated by the corresponding shortest path distances accurately[18].

A straightforward method is to find a suitable neighborhood size through estimating the "quality" of the corresponding mapping, i.e. how well the high-dimensional structure is represented in the low-dimensional embedded space[19], measured by residual variance[12]: $1-\rho^2_{\hat{D}_X(K)D_Y}$, where $\rho_{\hat{D}_X(K)D_Y}$ is the standard linear correlation coefficient, taken over all the entries of $\hat{D}_X(K)$ and D_Y , and where $\hat{D}_X(K)$ and D_Y are matrices of geodesic distances approximated by the shortest path distances in the high-dimensional data space *X*, which is the function of the neighborhood size *K*, and Euclidean distances in the low-dimensional embedded space *Y*, which is the low-dimensional mapping of ISOMAP, respectively.

The lower residual variance is, the better the high-dimensional structure is represented in the low-dimensional embedded space, and thus the more suitable the neighborhood size is. So the optimal *K* can be defined as follows[19]:

$$K_{opt} = \arg\min_K (1-\rho^2_{\hat{D}_X(K)D_Y}) \quad (1)$$

Due to residual variance's use of *Y*, i.e. the low-dimensional mapping of ISOMAP, and its multimodality, we must run the whole ISOMAP algorithm with every possible $K \in [K_{\min}, K_{\max}]$ ($K_{\min}$ is the minimal *K* which can make the corresponding neighborhood graph connected and $K_{\max}$ is the predefined maximal *K* which can be specified further in the next section), and select the optimal *K* with the minimal residual variance, which makes it very time-consuming. So, residual variance can only be used to estimate the intrinsic dimensionality of the data and compare the relative goodness of two neighborhood sizes, however, not to select a suitable neighborhood size practically.

III. Our Method

A suitable neighborhood size should be the one with which the neighborhood graph can represent the right neighborhood structure of the data and thus geodesic distances can be approximated by the corresponding shortest path distances accurately. To do this, the neighborhood graph should be connected and dense, but it cannot contain shortcut edges. Under these restrictions, the more the number of edges in the neighborhood graph is (that is, the denser the neighborhood graph is), the more accurately geodesic distances are approximated by the corresponding shortest path distances, which can be verified by experimental results in the next section. So, a suitable neighborhood size should be large enough, while not introducing shortcut edges into the neighborhood graph.

Obviously, under the restriction that the neighborhood graph is connected, as the neighborhood size increases and new edges are added into the neighborhood graph, the involved shortest path distances decrease

monotonously due to triangular inequality of the Euclidean distance metric and the definition of shortest path distance as in (2), limited by the corresponding geodesic distances before shortcut edges are introduced into the neighborhood graph.

$$\hat{d}_{ij} = \min(\hat{d}_{ij}, \hat{d}_{ik} + \hat{d}_{kj}), k = 1,2,\cdots \quad (2)$$

So, the sum of all the shortest path distances is a monotonously descent function of the neighborhood size *K*:

$$f(K) = \sum_{all\ the\ entries} \hat{D}_X(K) \quad (3)$$

Before shortcut edges are introduced into the neighborhood graph (when the neighborhood size is suitable), as the neighborhood size increases, the neighborhood graph is denser along the manifold, new edges added into the neighborhood graph will be longer and their influences on the involved shortest path distances will be weaker, and thus the decrement of the sum of all the shortest path distances $f(K)$ will decrease gradually.

Once shortcut edges are introduced into the neighborhood graph (when the neighborhood size is unsuitable), the sum of all the shortest path distances $f(K)$ will drop sharply in contrast with the former gradually decreasing downtrend, because shortcut edges can influence much more shortest paths than other edges in the neighborhood graph and make them be no longer along the manifold.

Consequently, to select a suitable neighborhood size *K*, we can compute the sum of all the shortest path distances $f(K)$, which only requires running the former path of ISOMAP (shortest path computation) incrementally beginning with $K_{\min}$, and select the first one at which the sum of all the shortest path distances $f(K)$ begins to drop sharply in contrast with the former gradually decreasing downtrend as the suitable *K* (i.e. $K_{suitable}$) to be used in the subsequent ISOMAP algorithm.

For example, we can select $K_{suitable}$ as follows:

$$\begin{aligned} &IF\ f(K) - f(K+1) > 2*(f(K-1) - f(K)) \\ &\quad AND\ first == TRUE \\ &\qquad K_{suitable} = K; \\ &\qquad first = FALSE; \\ &\quad END \end{aligned} \quad (4)$$

Where *first* is a boolean variable whose initial value is TRUE.

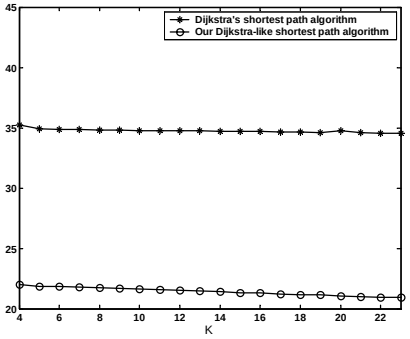
So, our method is less time-consuming than the straightforward method with residual variance. In addition, shortcut edges will be introduced into the neighborhood graph if a bit larger *K* is used, so the selected $K_{suitable}$ is also the maximal suitable *K*, which can also be used as the upper boundary of the suitable *K*s, i.e. $K_{\max}$, for the straightforward method with residual variance.

Although our method doesn't require running the latter part of ISOMAP, i.e. the classical MDS algorithm which includes the very time-consuming eigenvalue decomposition, our method still requires running the former part of ISOMAP, i.e. shortest path computation, which is still time-consuming.

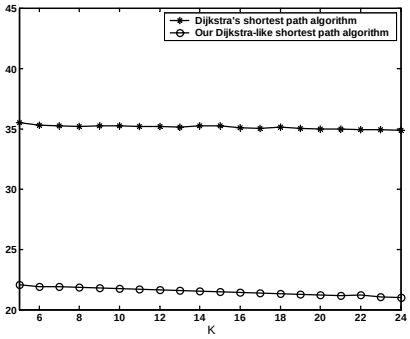
Note that weights of the edges in the neighborhood graph are specified as the corresponding Euclidean distances, and the Euclidean distance metric meets the symmetry and triangular inequality conditions, so we can greatly quicken Dijkstra's shortest path algorithm based on these two characteristics of the Euclidean distance metric, and the corresponding time costs of these two shortest path algorithms with different neighborhood size *K* over the well-known Swiss-roll and S-curve data sets are given in Fig. 1(a) and Fig. 1(b) respectively (different neighborhood size *K* means different neighborhood graph). As a Dijkstra-like shortest path algorithm, its differences from Dijkstra's shortest path algorithm are listed as follows:

1) The shortest paths found previously needn't be computed once again and can also be used to find other shortest paths (use the symmetry condition);

2) For each vertex, its edges existing in the neighborhood graph are the shortest paths themselves, needn't be computed once again, and can also be used to find other shortest paths (use the triangular inequality condition).



(a) The Swiss-roll data set.



(b) The S-curve data set.

Figure 1. Time costs of these two shortest path algorithms over different neighborhood graph of different data set.

We represent the *i*-th data point and its *K* nearest neighbors by x_i and $N_K(i)$ respectively, and then the

adjacent matrix of the corresponding neighborhood graph can be represented by $P=(p_{ij})_{n\times n}$, where

$$p_{ij}=\begin{cases}0, & j=i\\ \|x_i-x_j\|, & j\in N_K(i)\ or\ i\in N_K(j)\\ +\infty, & else\end{cases} \quad (5)$$

Consequently, our Dijkstra-like shortest path algorithm can be described as follows:

Initialize $\hat{D}_X(K)=(\hat{d}_{ij})_{n\times n}$ to be identical with $P=(p_{ij})_{n\times n}$;

For each data point i

$S=\{j\mid \hat{d}_{ij}<+\infty\}$; ($S=\{i\}$ in Dijkstra's shortest path algorithm)

$T=\{1,\cdots,n\}-S$;

For each data point $j\in T$

$d_j=\min_{l\in S}\{\hat{d}_{il}+\hat{d}_{lj}\}$; ($d_j=\hat{d}_{ij}$ in Dijkstra's shortest path algorithm)

End

While $T\neq\Phi$

$j=\arg\min_{l\in T}\{d_l\}$;

$\hat{d}_{ij}=d_j$; $\hat{d}_{ji}=d_j$; (This two shortest paths can also be used to find other shortest paths)

$S=S+\{j\}$; $T=T-\{j\}$;

For each data point $l\in T$

If $d_l>\hat{d}_{ij}+\hat{d}_{jl}$

$d_l=\hat{d}_{ij}+\hat{d}_{jl}$;

End

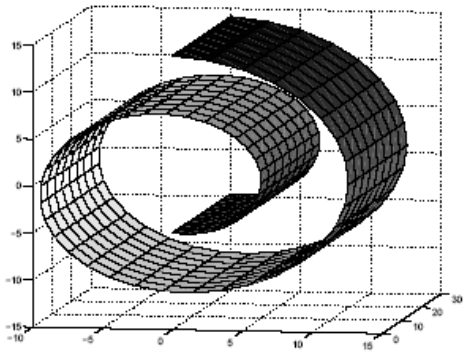
End

End

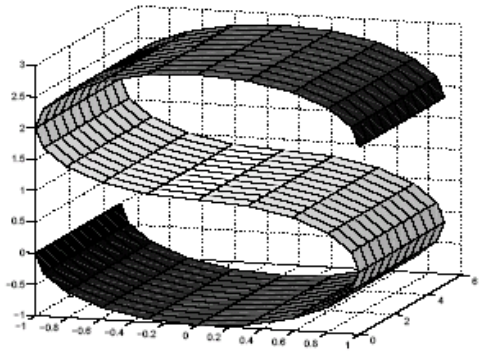
End

IV. Experimental Results

To contrast our method with the straightforward method with residual variance, we run ISOMAP, which uses our faster Dijkstra-like shortest path algorithm, with different neighborhood size *K* over two widely-used data sets, i.e. Swiss-roll and S-curve, with 2000 data points sampled uniformly from the corresponding intrinsic manifolds given in Fig. 2(a) and Fig. 2(b) respectively. In the experiments, we use the *k*-means algorithm in Matlab v6.5 toolboxes to select *n*=500 representative data points from these two data sets respectively. Residual variance and our function $f(K)$, i.e. the sum of all the shortest path distances obtained by our faster Dijkstra-like shortest path algorithm, with different *K* over these two data sets are given in Fig. 3 and Fig. 4 respectively.



(a) The intrinsic manifold of the Swiss-roll data set.



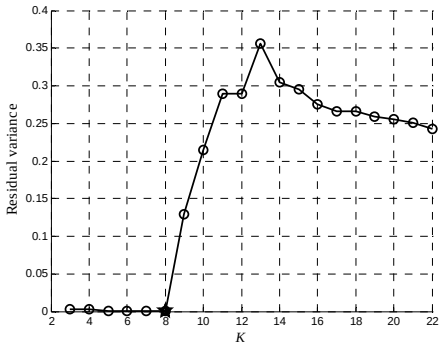
(b) The intrinsic manifold of the S-curve data set.

Figure 2. Two manifolds used in the experiments.

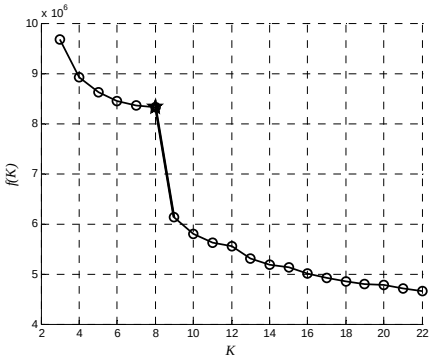
In the experiments, we specify the first *K* at which the next decrement of our function is one times larger than the last one as the suitable *K* to be used in the subsequent ISOMAP algorithm, as in (4).

From Fig. 3 and Fig. 4, We can see that our method selects the same neighborhood size with the straightforward method with residual variance, which means what our method selects, i.e. $K_{suitable}$, is also optimal, for example, $K_{suitable}$=7 over the Swiss-roll data set, represented by five-pointed stars in Fig. 3(a) and Fig. 3(b), but with 144.156 seconds in our method vs. 292.204 seconds in the straightforward method with residual variance, and $K_{suitable}$=18 over the S-curve data set, represented by five-pointed stars in Fig. 4(a) and Fig. 4(b), but with 314.031 seconds in our method vs. 668.266 seconds in the straightforward method with residual variance.

To verify the suitability of the selected $K_{suitable}$, the neighborhood graphs and low-dimensional mappings of ISOMAP with the corresponding $K_{suitable}$ over these two data sets are given in Fig. 5 and Fig. 6 respectively, from which we can see that the intrinsic manifold structures of these two data sets are recovered nicely, which means that the selected *K* is suitable.

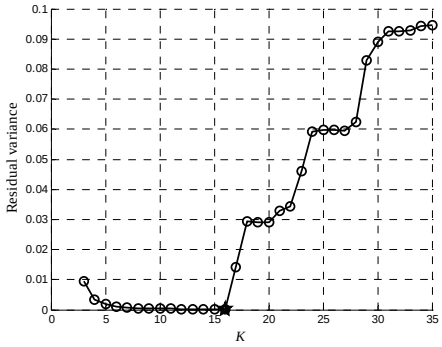


(a) Residual variance with different *K*.

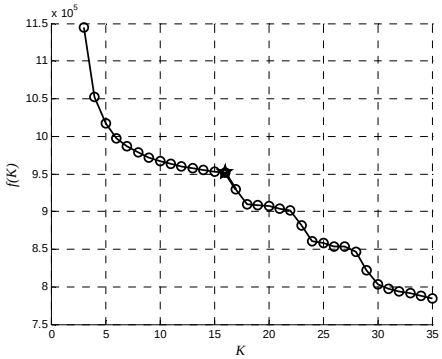


(b) The sum of all the shortest path distances ($f(K)$) with different *K*.

Figure 3. The contrast of these two methods over the Swiss-roll data set ($K_{suitable}$ is represented by the five-pointed star).

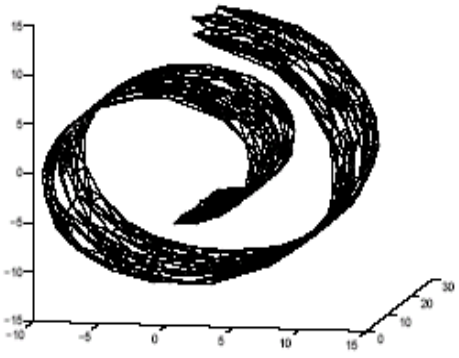


(a) Residual variance with different *K*.

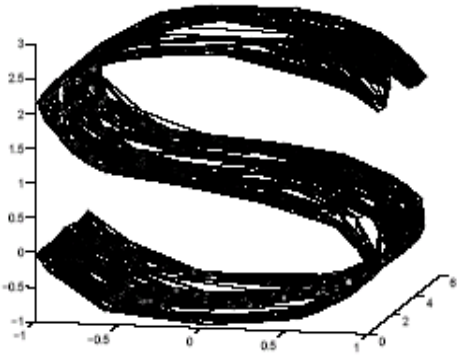


(b) The sum of all the shortest path distances ($f(K)$) with different *K*.

Figure 4. The contrast of these two methods over the S-curve data set ($K_{suitable}$ is represented by the five-pointed star).

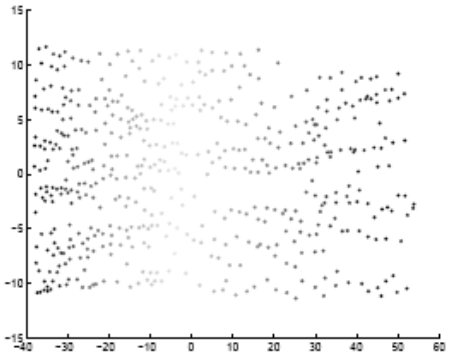


(a) The Swiss-roll data set, *K*=7.

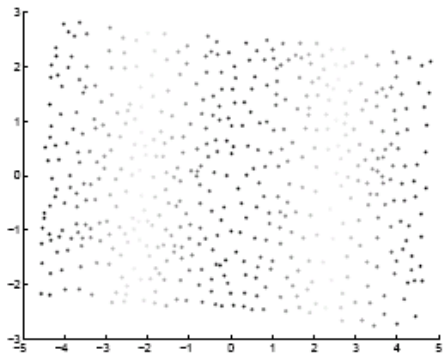


(b) The S-curve data set, *K*=18.

Figure 5. The neighborhood graphs with the corresponding $K_{suitable}$ over different data set.



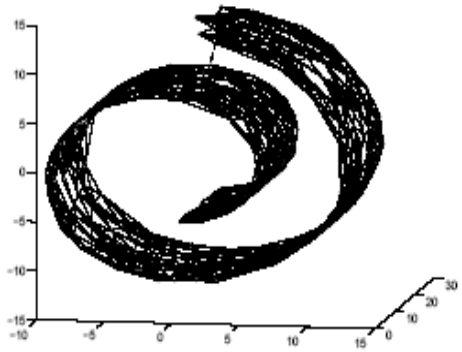
(a) The Swiss-roll data set, *K*=7.



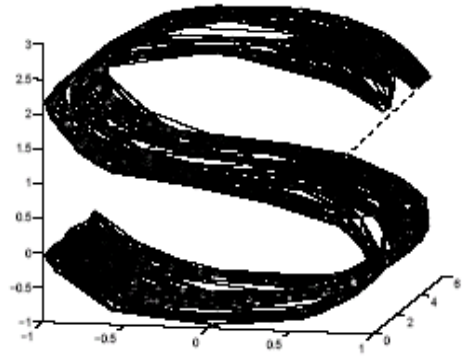
(b) The S-curve data set, *K*=18.

Figure 6. The low-dimensional mappings of ISOMAP with the corresponding $K_{suitable}$ over different data set.

When we increase $K_{suitable}$ by one correspondingly, shortcut edges (represented by dashed lines in Fig. 7(a) and Fig. 7(b) respectively) are introduced into the corresponding neighborhood graphs, and thus the corresponding low-dimensional mappings of ISOMAP worsen badly, as given in Fig. 8, which means what our method selects, i.e. $K_{suitable}$, is also the maximal suitable *K*.

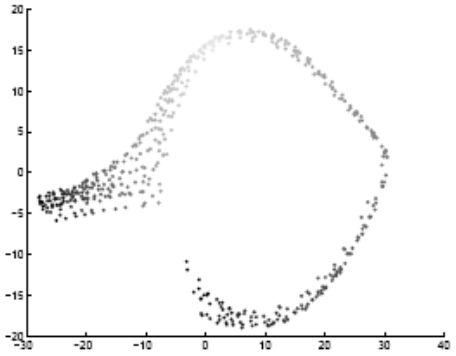


(a) The Swiss-roll data set, *K*=8.

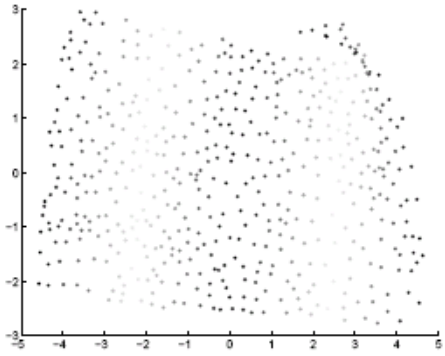


(b) The S-curve data set, *K*=19.

Figure 7. The neighborhood graphs with the corresponding $K_{suitable}$+1 over different data set (shortcut edges are represented by dashed lines).



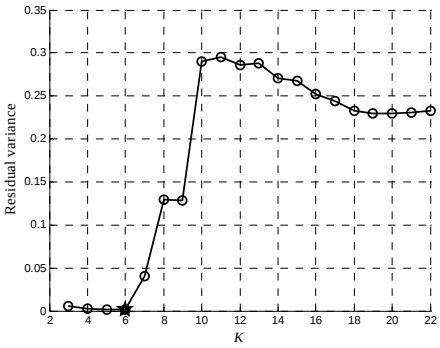
(a) The Swiss-roll data set, *K*=8.



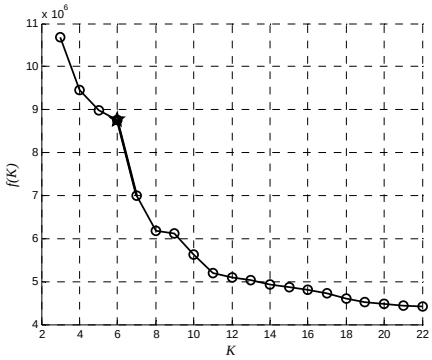
(b) The S-curve data set, *K*=19.

Figure 8. The low-dimensional mappings of ISOMAP with the corresponding $K_{suitable}$+1 over different data set.

To verify the robustness of our method, we run ISOMAP with different neighborhood size *K* over the noisy Swiss-roll and the noisy S-curve data sets, with zero-mean normally distributed noise added to each data point of the corresponding data set, where the standard deviation of the noise is chosen to be 2% of smallest dimension of the bounding box enclosing the data. Residual variance and our function $f(K)$ with different *K* over these two noisy data sets are given in Fig. 9 and Fig. 10 respectively.

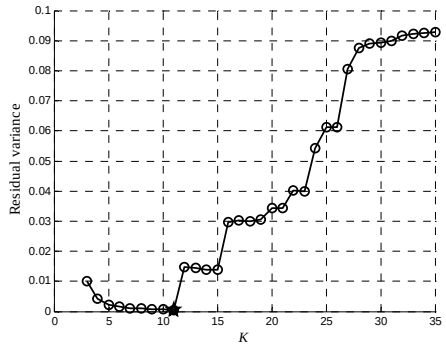


(a) Residual variance with different *K*.

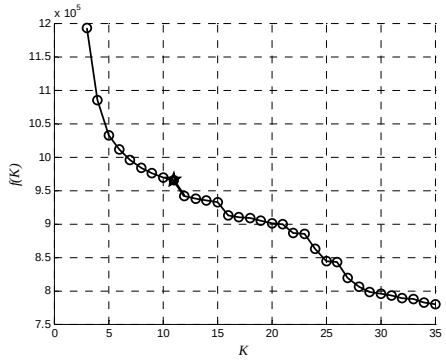


(b) The sum of all the shortest path distances ($f(K)$) with different *K*.

Figure 9. The contrast of these two methods over the noisy Swiss-roll data set ($K_{suitable}$ is represented by the five-pointed star).



(a) Residual variance with different *K*.



(b) The sum of all the shortest path distances ($f(K)$) with different *K*.

Figure 10. The contrast of these two methods over the noisy S-curve data set ($K_{suitable}$ is represented by the five-pointed star).

From Fig. 9 and Fig. 10, We can see that our method still can select the same neighborhood size with the straightforward method with residual variance over the noisy data sets, that is, what we selects is still the maximal suitable *K*.

From Fig. 3(a) and Fig. 4(a), under the restriction that shortcut edges are not introduced into the neighborhood graph (that is, *K* is less than or equal to the selected $K_{suitable}$), as more edges are introduced into the neighborhood graph, indicated by the larger *K*, geodesic distances can be approximated more accurately, indicated by the smaller residual variance. This fact can also be verified by the comparison among three methods to construct the neighborhood graph in ISOMAP: ISOMAP with the *K* nearest neighbors method[12], ISOMAP with the *K*-edge connected neighborhood graph[23], and ISOMAP with the *K* mutual neighborhood graph[24]. In these three methods with the same suitable neighborhood sizes *K*=6 and *K*=7 respectively over the Swiss-roll data set, ISOMAP with the *K*-edge connected neighborhood graph has the most edges in the corresponding neighborhood graph and the smallest residual variance, and ISOMAP with the *K* mutual neighborhood graph has the least edges in the corresponding neighborhood graph and the largest residual variance, as given in Table I.

TABLE I. THE RESIDUAL VARIANCE AND THE NUMBER OF EDGES IN THE CORRESPONDING NEIGHBORHOOD GRAPH OBTAINED BY THREE METHODS WITH *K*=6 AND *K*=7 OVER THE SWISS-ROLL DATA SET

Three methods		Residual variance	The number of edges in the corresponding neighborhood graph
ISOMAP with the *K* nearest neighbors method	K=6	0.0006	3432
	K=7	0.0004	3970
ISOMAP with the *K*-edge connected neighborhood graph	K=6	0.0002	5988
	K=7	0.0001	6986
ISOMAP with the *K* mutual neighborhood graph	K=6	0.0010	2568
	K=7	0.0008	3030

In this sense, what our method selects, i.e. the maximal suitable *K*, is optimal, which can also be proved by the fact that what our method selects is same to the one with the minimal residual variance, as given in Fig. 3-4 and Fig. 9-10.

In addition, from Fig. 3-4 and Fig. 9-10, we can see that the sum of all the shortest path distances ($f(K)$) decreases monotonously and its decrement decreases gradually until a sharp drop emerges, at which shortcut edges begin to emerge in the neighborhood graph and the corresponding low-dimensional mapping of ISOMAP begins to worsen badly, represented by Fig. 5-8 respectively, by which the feasibility and effectiveness of our method can be verified.

V. CONCLUSIONS

It's an open problem for those manifold learning techniques such as ISOMAP and LLE to select a suitable neighborhood size, with which a right local connectivity can be constructed and thus these techniques can be applied successfully. In this paper, we present an efficient and robust method to select the suitable neighborhood size *K* according to the decrement of the sum of all the shortest path distances. Our method is less time-consuming than the straightforward method with residual variance, while yielding the same results. What's more, what our method selects is also the maximal suitable *K*, which can be provided as the upper boundary of the suitable *K*s, i.e. $K_{\max}$, for the straightforward method with residual variance.

ACKNOWLEDGMENT

This work was supported in part by Research Programme of Henan Fundamental and Advanced Technology of China (No. 082300410110), the Key Technologies R&D Programme of Henan Province, China (No. 102102210400).

REFERENCES

[1] W. S. Cleveland, *Visualizing Data*. New Jersey, U.S.A.: Hobart Press, 1993.

[2] D. A. Keim, "Designing pixel-oriented visualization techniques: theory and applications," *IEEE Transactions on Visualization and Computer Graphics*, vol. 6, no. 1, pp. 59-78, 2000.

[3] A. Inselberg and B. Dimsdale, "Parallel coordinates: a tool for visualizing multidimensional geometry," in Proc. IEEE Visualization 1990, San Francisco, California, October 23-25, 1990, pp.361-378.

[4] E. Kandogan, "Visualizing multi-dimensional clusters, trends, and outliers using star coordinates," in Proc. 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, California, August 26-29, 2001, pp.107-116.

[5] J. LeBlanc, M. O. Ward, and N. Wittels, "Exploring n-dimensional databases," in Proc. IEEE Visualization 1990, San Francisco, California, October 23-25, 1990, pp.230-239.

[6] B. Johnson, *Visualizing hierarchical and categorical data*. Ph.D. Thesis, Department of Computer Science, University of Maryland, 1993.

[7] E. R. Tufte, *The visual display of quantitative information*. Cheshire, CT: Graphics Press, 1983.

[8] R. M. Pickett and G. G. Grinstein, "Iconographic displays for visualizing multidimensional data," in Proc. 1988 IEEE Conference on Systems, Man and Cybernetics, Beijing and Shenyang, China, August 8-12, 1988, pp.514-519.

[9] J. C. Mao and A. K. Jain, "Artificial neural networks for feature extraction and multivariate data projection," *IEEE Transactions on Neural Networks*, vol. 6, no. 2, pp. 296-317, 1995.

[10] A. Naud and W. Duch, "Interactive data exploration using MDS mapping," in Proc. 5th Conference on Neural Networks and Soft Computing, Zakopane, Poland, June 5-10, 2000, pp.255-260.

[11] C. Shao and H. K. Huang, "Improvement of data visualization based on SOM," in Proc. International Symposium on Neural Networks, Dalian, China, August 19-21, 2004, pp.707-712.

[12] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 22, pp. 2319-2323, 2000.

[13] V. de Silva and J. B. Tenenbaum, "Unsupervised learning of curved manifolds," in Proc. MSRI Workshop on Nonlinear Estimation and Classification, Berkeley, California, March 19-29, 2001, pp.453-466.

[14] M. Vlachos, C. Domeniconi, D. Gunopulos, G. Kollios, and N. Koudas, "Non-linear dimensionality reduction techniques for classification and visualization," in Proc. 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Edmonton, Alberta, Canada, July 23-26, 2002, pp.645-651.

[15] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 22, pp. 2323-2326, 2000.

[16] A. Hadid and M. PietikÄainen, "Efficient locally linear embeddings of imperfect manifolds," in Proc. 3rd International Conference on Machine Learning and Data Mining in Pattern Recognition (MLDM'03), Leipzig, Germany, July 5-7, 2003, pp.188-201.

[17] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Computation*, vol. 15, no. 6, pp. 1373-1396, 2003.

[18] M. Balasubramanian, E. L. Schwartz, J. B. Tenenbaum, V. de Silva, and J. C. Langford, "The Isomap algorithm and topological stability," *Science*, vol. 295, no. 5552, pp. 7a-7, 2002.

[19] O. Kouropteva, O. Okun, and M. PietikÄainen, "Selection of the optimal parameter value for the locally linear embedding algorithm," in Proc. 1st International Conference on Fuzzy Systems and Knowledge Discovery (FSKD'02), Singapore, November 18-22, 2002, pp.359-363.

[20] M. Bernstein, V. de Silva, J. C. Langford, and J. B. Tenenbaum, "Graph approximations to geodesics on embedded manifolds," Technical Report, Department of Psychology, Stanford University, 2000.

[21] J. A. Lee, A. Lendasse, and M. Verleysen, "Nonlinear projection with curvilinear distances: Isomap versus curvilinear distance analysis," *Neurocomputing*, vol. 57, pp. 49-76, March 2000.

[22] O. Kouropteva, O. Okun, A. Hadid, M. Soriano, S. Marcos, and M. Pietikäinen, "Beyond locally linear embedding algorithm," Technical Report MVG-01-2002, University of Oulu, Machine Vision Group, Information Processing Laboratory, 2002.

[23] Y. Li, "K-edge connected neighborhood graph for geodesic distance estimation and nonlinear data projection," in Proc. 17th International Conference on Pattern Recognition (ICPR'04), Cambridge, UK, August 23-26, 2004, pp.196 – 199.

[24] Y. Qian and K. Zhang, "Discovering spatial patterns accurately with effective noise removal," in Proc. 9th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery (DMKD'04), Paris, France, June 13, 2004, pp.43-50.

Li Jing, born in Henan Province, China on February 20, 1971. She received her PH.D. degree in computer software theory from Information Engineering University, Zhengzhou, China in 2009. Now she is an ASSOCIATE PROFESSOR of School of Computer and Information Engineering, Henan University of Economics and Law, Zhengzhou, Henan Province, China. His research fields include artificial neural network,, data mining and image process, *etc*. Dr. Jing is also a member of China Computer Federation.

Chao Shao, born in Henan Province, China on December 12, 1977. He received his PH.D. degree in computer application technology from Beijing Jiaotong University, Beijing, China in 2006.

He is currently an ASSOCIATE PROFESSOR of School of Computer and Information Engineering, Henan University of Economics and Law, Zhengzhou, Henan Province, China. His research fields include artificial neural network, machine learning, data mining and data visualization, *etc*.

Dr. Shao is also a member of China Computer Federation.