Advances and Applications in Mathematical Sciences
Volume 21, Issue 5, March 2022, Pages 2371-2382
© 2022 Mili Publications, India

AN IMPROVED ISOMAP ALGORITHM FOR NONLINEAR DIMENSIONALITY REDUCTION

S. GNANA SOPHIA, K. K. THANAMMAL* and S. S. SUJATHA

*Associate Professor
Affiliated by: Manonmaniam Sundaranar University
Abishekapatti, Tirunelveli-627012, Tamil Nadu, India
Department of Computer Science and Applications
S. T. Hindu College, Nagercoil
E-mail: gnanasophias@gmail.com
thanaravindran@gmail.com
sujaajai@gmail.com

Abstract

Dimensionality reduction is the seeking of small group of features to characterize a large set of reduction which is very useful in anticipating data, decline the processing time of calculation, etc. Linear PCA and classical MDS linear dimensionality reduction techniques are improper if the data set consists of nonlinear relationships amid the variables. Isomap is the meaning for isometric mapping. It is a nonlinear dimensionality reduction which follows spectral theory and the geodesic distances in lower dimensions. Isomap starts by constituting a neighborhood network and it can use the distance of graph to the estimated geodesic distance by associating all set of points. Isomap is the most popular nonlinear manifold dimensionality reduction. This can be applied to real world medical data sets. In this paper, we have formed the new proposed method called Secure Isomap. To secure the exactness of the graphs, globally optimal and non-iterative method is the main aim of this paper.

I. Introduction

Many machine learning technique may not be adequate for higher dimensional data. Dimensionality reduction is needed for visualization, i.e., it projects the high dimensional data onto 2D or 3D [3]. The valuable storage and regeneration are done in Data Compression and in Noise removal. Here, there will be the positive effect on the accuracy of query. Dimensionality

2020 Mathematics Subject Classification: 58J51.
Keywords: Dimensionality reduction, Linear Discriminant Analysis (LDA), Principal Component Analysis (PCA), Kernel Principal Component Analysis (KPCA), Feature selection.
Received May 9, 2021; Accepted August 22, 2021

Reduction is applicable in Customer Relationship Management, Text Mining, Image Retrieval, Face Recognition, Intrusion Detection etc.

Techniques of Dimensionality Reduction

There are two groups of techniques which can be used in dimensionality reduction: Feature selection and feature Extraction

1. Feature Selection method

It can be used to select which features to carry and which features to delete. This is the action in which we can choose the subgroup of applicable appearance for utilizing in the development of model i.e. it is used to select new features [14]. For example in anticipating the cost of a plot, an area, and the location are the important domain knowledge. Feature selection and dimensionality reduction are combined simultaneously to form the reduced number of features in a data set [4]. The common selection and the elimination of given features without changing is called feature selection. Dimensionality Reduction is used to reconstruct the features into a lower dimensions [6]. Some of the common features of feature selection are: detach features with missing values, separate features with low variance, take out deeply correlated features, etc., [13].

2. Feature Extraction

It is an activity of dimensionality reduction in which the beginning set of data is lessened to highly convenient set of transitioning. It does not select any subset and it creates a new feature set. After the creation, the features are unrelated among themselves [13].

The different techniques of Dimensionality Reduction: Linear Discriminant Analysis, Principal Component Analysis and Kernel Principal Component Analysis [7]

i. Linear Discriminant Analysis (LDA)

It is hands down to find the linear combination of attributes that can break two or more classes of objects. It is most commonly operated in pre-processing step for machine learning applications. It can target on maximizing the separability among known categories [12]

ii. Principal Component Analysis (PCA)

It is an attractive tool for linear dimensionality reduction and error detection method. The variables which can occur in divide ally are extremely tallied together. Variables inserted are metric level or nominal level. In nature features are low dimensional. In addition, independent variables are numeric in nature. PCA is the most familiar unsupervised machine-learning algorithm for the collection of applications [15]. They can be used in Dimensionality Reduction, compression, etc. The chief purpose of PCA is to observe the principal component sand it will narrate the data points with a set of principal components. These are vectors and cannot be chosen at random. The principal components can abolish the noise by lowering the huge number of features to a set of principal components and a feature vector and form the principal components [8], normalize the data, compute the covariance matrix, and compute eigen values and eigenvectors, selecting components, etc. It is easy to calculate because it is depended on linear algebra which is easily solved by computers. It boosts up the machine learning algorithms, its coverage is quick when applied on Principal components replacing actual data set.

It aims to distinguish the correlation between variables. When we want to establish that variables in data which are in dependent to each other, PCA is needed. From the data set if we wish to diminish the number of variables with many variables can clarify data and variables in a dataset, PCA is used. Suppose if we want nonlinear principal, standard PCA can find linear principal components to perform data in lower dimensions component, and if we administer standard PCA for the data, it cannot produce good representative direction. Kernel Principal Component Analysis (KPCA) is used Toa void this problem [8].

iii. Kernel Principal Component Analysis (KPCA)

The Kernel Principal Component Analysis (KPCA) achieves PCA in a new area. It can develop the PCA to a high dimensional feature space using "kernel Trick".

Linear: Something having to do with a line.

Nonlinear: It is a function where the graph is a straight line.

PCA is linear and it is inefficient to classify straight information adequately. KPCA is an approach to apprehend "non-linear data" sample. This is an expansion of PCA which is applied in non-linear applications by kernel trick [12]. It can be able to build non-linear mappings that can maximize the variance in data [9]. This is a random PCA expanded in applying kernel method. Also it makes use of Kernel trick to determine PCA in different space and executes PCA in a new space, and gets elevated difference.

$\varnothing$ is a nonlinear mapping function accordingly the map ping of a pattern X can be mentioned as

$X- > \varnothing(X)$ is known as kernel function. The phrase kernel interprets a function hate valuates the multiplication of the images of the patterns X based Ø

$$K(Xi, Yj) = \varnothing(Xi)\varnothing(Xj)T$$

Applications of KPCA

It applies kernel trick to note principal components in high dimensional space. It directs new path which is based on co variance matrix of original variables. The computational complexity for KPCA to extract principal components and it takes more time. KPCA is a nonlinear class of Principal component analysis which depends on kernel tricks. The major concept is that the input vector x can be activated into the higher dimensional feature space by the use of the non linear mapping which can be sorted out in advance. It can give a good re-encoding of the data when it lies along an on linear manifold.

1.1. Isomap

Isomap is the non linear dimensionality reduction technique which can preserve the geodesic distance and it is the alternative of Multidimensional Scaling (MDS). MDS is used to model the nonlinear data by the use of geodesic distance. The Isomap is used to obtain the best subspace and keep the geodesic distance bet ween the data points.

Procedure of Isomap.

- **Neighbourhood Graph.** It can produce an adjacency matrix and neighbourhood graph from the given data set.
- **Different matrix:** After searching the neighbourhood graph, we can use sparks graph *X* library to compute the geodesic distances between the points. One must check and confirm that the resultant graph is a single connected component or else, our similarity matrix will continue in adequate and the outcome will be incomprehensible. We have to emphasize over the various values of neighbourhood selection criterion to produce the completed linked graph.
- **Eigen value Decomposition.** First we have to square the distance and double center the squared resemblance matrix before eigenvalue decomposition. After that choose the first K eigenvectors with K maximum eigenvalues.

1.2. The *K* nearest Neighbor Algorithm.

This algorithm is simple and easy to implement machine-learning algorithm. This algorithm is used to recognize accurate nearest neighbors, The elasticity of such process is linear [for the users who close in Big-O notation, O(N), where N is the size of training set. In some images, same data points are close to each other.

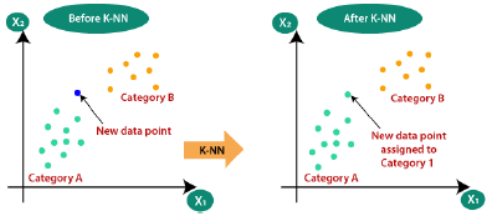

Figure 1. Before KNN and after KNN.

1.3. Stochastic Gradient Descent (SGD).

Stochastic gradient descent is a famous and general algorithm which is applied in many machine learning algorithms. Gradient is the meaning of go down a slope to attain the minimum point on the plane in plain terms [3]. This can be iterative algorithm which can begin from an arbitrary point on the function and proceed falling its slope in steps until it grasp that function which has the lowest point.

The procedure of the algorithm are

1. Calculate the slope of the function.

2. Choose random initial value for the frame work. (To make it clear, in the parabola example, differentiate "y" with respect to "x".

3. By packing in the parameter values, we can renovate the gradient function.

4. Compute the step sizes for every features as: step size=gradient* learn ingrate.

5. Evaluate new criteria=former criteria-step size.

6. Continue working procedures 3 to 5 until gradient is almost 0.

The quantity can be determined that they decline our cost function, we can employ optimization algorithm called gradient descent: In batch gradient descent, in order to reduce cost function, the gradient is evaluated from the whole training set (this is why this approach is also mentioned as "batch"). By graph, it shows the distance between the two points. The geode sic distance between two points in the images are estimated Therefore Euclidean distance will not be applied for estimating the distance between the two points in nonlinear manifolds [16]. An Isomap uses this concept to constitute the similarity matrix for eigenvalue de composition. Isomap estimates both global and local structure of the data set in the low dimensional embedding. Isomap is firmly relevant to the authentic multidimensional scaling algorithm. It is an expansion of the classical multidimensional scaling. It can bring out a closed form of result to the dimensionality reduction issue. The classical MDS uses the Euclidean distances as the comparison matrix and Isomap algorithm uses geodesic distances [3].

The stochastic defines a system or a process which is connected with a random probability. Therefore, some cases are chosen arbitrary which replaces the total data set for each iteration in stochastic gradient descent [17]. In gradient descent, the name "batch" mentions the whole number of patterns from a data set. In each iteration it will compute the gradient. In typical gradient descent optimization, like batch gradient descent, the batch is treated as full dataset. While processing the gradient descent, until the minima is reached it has to be done for every iteration when we have a much

more samples in the particular data set. Later, if we use a common gradient descent optimization technique, we use the samples of large dataset for accomplishing one iteration. Hence, it becomes computationally high priced to act on it. This issue is resolved by Stochastic Gradient Descent. In SGD, it uses only one sample, i.e., a batch size of one, to perform each iteration. The sample is chosen for performing the iteration after the random shuffle.

for i in range (m) : $\theta_j = \theta_j - \alpha(y^i - y^i)x^i j$

So, in SGD, at each iteration the cost function of the gradient is figured out of a single example instead of the sum of the gradient of the cost function of all the examples.

In SGD, though only one sample from the data set is chose Nat random for every iteration, the path taken by the algorithm to attain the minima is usually noisier than your typical Gradient Descent algorithm. But when analyzing it does not matter much because the path chosen by the algorithm is not concerned, as long as one extends the minima with naturally brief training time.

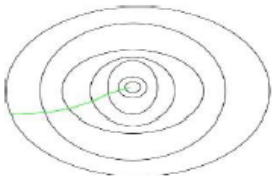

Figure 2. Path chosen by Batch Gradient Descent.

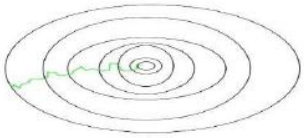

Figure 3. Path taken by Stochastic Gradient Descent.

SGD is usually noisier than typical Gradient Descent, and to reach the minima it generally takes larger iterations. Because in its descent it will be randomness. It is computationally less costly than the traditional Gradient Descent though it needs large number of iterations to attain the minima than traditional Gradient Descent. Consequently, in most cases, SGD is chosen

over Batch Gradient Descent for boosting a learning algorithm.

Algorithm. SGD
Initialize $W\hat{0}$
for $t = 1, 2, \ldots$
Draw (X_t, Y_t) randomly from D
Update $\hat{w}_{t-1}$ as
$W\hat{t} = \hat{w}_{t-1} - \eta_t S^{-1}(\lambda \hat{w}_{t-1} + L'(\hat{w}_{t-1}^{1} X_t, \underset{1}{Y_t}) X_t)$
End

II. Proposed Algorithm

In common, if given high-dimensional data sets $X = \{x \in Rd\}$, one focuses on every data set to denote xi with low dimensional vector $yi \in Rd$, in which the value of disuse ally 2 or 3. Most often the algorithm of Isomap cannot reconstruct from high-dimensional data into low dimensional space. So, our Secure Isomap can effectively clarify the said issue. KNN graphs need a metric of distances, Euclidean distance is used in the high-dimensional space replacing geodesic distances because *s*. Euclidean distance E is used to scheme the same data points through a square matrix, and it yields appropriate results than geodesic distances. The fast Isomap method depends on the two factors: randomized truncating (vantage point tree) algorithm (tree building algorithm) and NN-Descent algorithm which is depending on the KNN graph. Both algorithms can be used for wrong link problems. Both of these involves two methods. First, to find the K nearest neighbor via a vantage point tree algorithm. The data points are divided into small datasets to find the K nearest neighbor by the use of Vantage point tree algorithm. Secondly, the sub graphs data and purified graph data with methods of the NN-Descent are combined. Further, Vantage point-tree searcher method is used for calculating the overall efficiency of the designed algorithm.

A vantage-point tree can be used to obtain the nearest neighbor of a point *x*. The search algorithm is recursive. At any given step this can be worked

with a node of the tree which has a vantage point v and a threshold distance t. The point of interest x will be some distance from the vantage point v. If the distance d is less than t then recursively use the algorithm to find the sub tree of the node which contains the points closer to the vantage point than the threshold t, otherwise recourse to the sub tree of the node which contains the points that are beyond than the vantage point than the threshold t. If the recursive use of the algorithm catches a neighboring point n with distance to x that is less than $|t - d|$ then it cannot help to examine the other subtree of this node; the revealed n is returned. Otherwise, the other subtree also needs to be examined recursively.

Vantage point tree algorithm.

Function vantage point tree (list of points p List, in t depth)

{

//Select axis based on depth so that axis cycles through all valid values

var int axis :=depth **mod** k;

//Sort point list and choose median as pivot element

select median by axis from pList;

//Create node and construct subtree

node.location :=median;

node.leftChild := vantagepointtree(points in pList before median, depth+1);

node.rightChild := vantagepointtree(points in pList after median, depth+1); returnnode;

}

NN Descent Algorithm 1. Outline of NN-Descent algorithm. input: data set D, distance function dist, neighborhood size K

output: K-NNG G 1

1. **for each** data point $u \in D$ do

2. **Initialize** G by randomly generating tentative K-NN list

for u with an assigned distance of $+\infty$;

3. **end**

4. **repeat**

5. **for each** data point $u \in D$ do

6. **Check** different pairs of u's neighbors (v, w) in u's K-NN and R-NN(reverse nearest neighbor)lists, and compute dist(v, w);

7. **Use** ⟨v, dist (v, w)⟩ to update w's K-N Nlist, and use ⟨w, dist (v, w)⟩ to update v's K-NN list;

8. **end**

9. **until** G converges;

10. return G

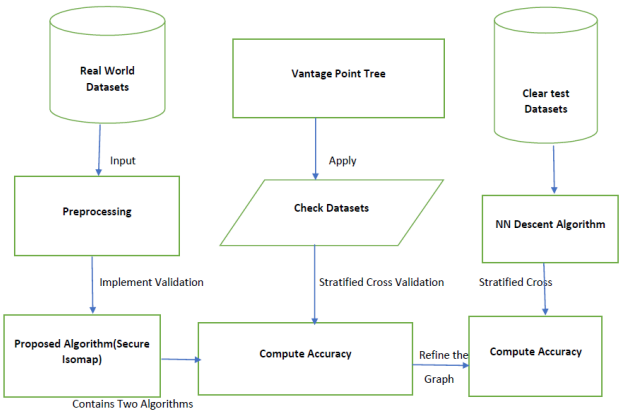

Figure 4. General steps of the secure Isomap.

III. Related Work

In the concept of Dimensionality Reduction, the methods do an important role to evaluate and anticipatory high-dimensional multi-source data. In proposed study, the author gauge 87 methods to incorporate Dimensionality Reduction of mRNA expression and DNA methylation data. This is a general issue in biology and medicine where their ranking is depended on four

feature quality of Dimensionality Reduction i.e. local, global and local-global, clustering quality and sensitivity to input parameters on multiple datasets making through inter SIR. The outputs are certified on real datasets [3].

The aim of data reduction technique in this paper is to leak out inappropriate appearance and strident data samples [4] and to decrease the high dimensional data. They produced it to a subspace by the use of PCA m disintegration and a narrative access based on autoencoder neural network. So it can reduce the dimensionality of original data. Some of the issues which are applicable in dimensionality reduction based data visualization are: conservation of adjacency relationships, treating data on a non-linear manifold, efficiency of anticipating extensions for the new test data [4]. The uniformity of the proposed model acts and achieves by experiments.

The authors used the algorithm: K-means Clustering and it was used on both original and reduced dataset. They performed many internal measures which can be used to review the clustering for various number of dimensions to assess how the reduction method collide the method of clustering [5].

IV. Conclusion

This paper introduced the novel framework called secure Isomap. This method provides very well-organized and active algorithm for the creation of vantage point tree algorithm and NN Descent algorithm. The experimental results demonstrated that Secure Isomap can keep the security of graphs and produced the optimal solution for the particular class of data manifolds. It can measure the distance on the integral geometric surface. The proposed work conserves the global data structure and it can achieve global optimization.

References

[1] S. Gnana Sophia and K. K Thanammal, An improved homomorphic encryption technology for the surveillance of cloud based medical data, Studies in Indian Place Names (UGC Care Journal) 40(18) (2020).

[2] S. Gnana Sophia and K. K Thanammal, An improved homomorphic encryption technology for the surveillance of cloud data, Solid State Technology 63(2s) (2020).

[3] M. Yousaf, T. U. Rehman and L. Jing, An extended isomap approach for nonlinear dimension reduction, SN COMPUT. SCI. 1, 160 (2020). https://doi.org/10.1007/s42979-020-00179-y

[4] A. Gisbrecht, B. Hammer, B. Mokbel and A. Sczyrba, Nonlinear dimensionality reduction for cluster identification in metagenomic samples, 17th International Conference on Information Visualisation, London (2013), 174-179, doi:10.1109/IV.2013.22.

[5] Thippa Gadekallu, et al., Thar Analysis of Dimensionality Reduction Techniques on Big Data, March 2020, IEEE Access PP(99):1-1, DOI:10.1109/ACCESS.2020.2980942, (2020).

[6] Suchismita Das and Nikhil Pal, Nonlinear Dimensionality Reduction for Data Visualization: An Unsupervised Fuzzy Rule-based Approach, IEEE Transactions on Fuzzy Systems (Early Access 2021) DOI:10.1109/TFUZZ.2021.3076583, (2020).

[7] M. Alkhayrat, M. Aljnidi and K. Aljoumaa, A comparative dimensionality reduction study in telecom customer segmentation using deep learning and PCA, J. Big Data 7(9) (2020).

[8] Hadi Fanaee-T and Magne Thoresen, Performance evaluation of methods for integrative dimension reduction, Information Sciences 493, (2019). DOI:10.1016/j.ins.2019.04.041.

[9] Y. Wang, H. Yao and S. Zhao, Auto-encoder based dimensionality reduction, Neurocomputing 5 (2016), 232-242.

[10] D. Fabre, V. Citro, D. Ferreira Sabino, P. Bonnefis, J. Sierra, F. Giannetti and M. Pigou, A practical review on linear and nonlinear global approaches to flow instabilities, Applied Mechanics Reviews, American Society of Mechanical Engineers 70(6) (2018), 060802. DOI:10.1115/1.4042737.

[11] J. Yang, H. Wang, H. Ding, et al., Nonlinear dimensionality reduction methods for synthetic biology biobricks' visualization, BMC Bioinformatics 18(47) (2017).

[12] Portnova-Fahreeva AA, Rizzoglio F, Nisky I, Casadio M, Mussa-Ivaldi FA, Rombokas E., Linear and Non-linear Dimensionality-Reduction Techniques on Full Hand Kinematics, Front Bioeng Biotechnol. 2020; 8: 429. Published 2020 May 5.doi:10.3389/fbioe.2020.00429

[13] R. Ramachandran, G. Ravichandran and A. Raveendran, Evaluation of dimensionality reduction techniques for big data, 2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC), 2020, pp. 226-231, doi: 10.1109/ICCMC48092.2020.ICCMC-00043.

[14] M. Karg, R. Jenke, W. Seiberl, K. Kühnlenz, A. Schwirtz and M. Buss, A comparison of PCA, KPCA and LDA for feature extraction to recognize affect in gait kinematics, 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops, Amsterdam, Netherlands, (2009), 1-6, doi:10.1109/ACII.2009.5349438.

[15] X. Zheng, S. Pawar and D. F. M. Goodman, Graph drawing by stochastic gradient descent, IEEE Transactions on Visualization and Computer Graphics 25(9) (2019), 2738-2748, doi: 10.1109/TVCG.2018.2859997.

[16] H. Chen, L. Zheng, R. Kontar and G. Raskutti, Stochastic gradient descent in correlated settings, A Study on Gaussian Processes, NeurIPS, (2020).