Research on Adaptive ISOMAP Algorithm and Application in Intrusion Detection

Xiang Liu¹*, Peng Ma¹, Gaoming Li¹

¹ Postgraduate Battalion, Engineering University of PAP, Xi’an, Shanxi, 710086 code, China
² Department of Foundation, Engineering University of PAP, Xi’an, Shanxi, 710086 code, China
*Corresponding author’s e-mail: liuxiangwqx@163.com

Abstract. In view of the low accuracy and many redundant attributes in the detection algorithms, an intrusion detection algorithm based on adaptive Isomap is proposed. The algorithm uses the sparse representation theory to adaptively select the neighborhood of data points. Then the sparse coefficients are used as the distance weight to improve the data discrimination ability. Finally, the improved isometric feature mapping algorithm is introduced to the intrusion detection as the feature extraction module. The algorithm not only overcomes the difficulty of manual parameter adjustment, but also has strong robustness. Experimental results show that using this method to extract intrusion detection features can effectively improve the detection accuracy, and at the same time improve the detection accuracy of Probe, U2R, R2L.

1. Introduction

With the development of information technology, there are more and more kinds of network attacks and malicious intrusions, and the means of attacks show a complex and diversified trend. Traditional intrusion detection technology can not achieve the expected results. In recent years, with the development of artificial intelligence, intelligent intrusion detection has become a research hotspot. Whether it is host-based intrusion detection or network-based intrusion detection, it can be regarded as a classification problem. In the classification module of the intrusion detection model, some machine learning models such as svm[1], DNN[2], RF[3] and intelligent bionic algorithms have achieved certain results. However, most of the feature extraction modules of intrusion detection researches are limited to feature selection methods[4] and linear feature extraction methods[7]. It is well known that feature selection only selects some features that are useful for classification and clustering from the original features, and cannot produce some new comprehensive features; linear feature extraction methods assume that the data distribution is linear or approximately linear, but with the development of computer technology, the data extracted for intrusion detection often has high-dimensional, non-linear characteristics. These methods cannot obtain a good feature space.

As a dimensionality reduction tool, manifold learning can be regarded as the promotion of classic linear dimensionality reduction methods such as PCA. In recent years, manifold learning has been successfully used in many fields such as machine learning, pattern recognition, medical imaging, intrusion detection, etc.. In 2000, with the publication of the LLE[8] and ISOMAP[9] algorithms, a series of manifold learning algorithms such as LE, LPP, and NPE have achieved major breakthroughs.
All these methods assume that the samples in these high-dimensional spaces are actually located on a low-dimensional manifold. The core of manifold learning is to learn hidden low-dimensional data structures from high-dimensional data. When learning hidden structures, different manifold learning algorithms try to maintain different geometric information. For example, the Isomap algorithm maintains the distance information between data points, the LLE algorithm maintains the neighbor relationship between the data points, and the LE algorithm makes the distance in the high-dimensional space relatively small. The near points are still close in the low-dimensional embedding. These algorithms start from different perspectives to obtain low-dimensional embedding. Various algorithms seem to be very different. However, they all have a common feature. They all use the local Euclidean characteristics of the manifold to construct the geometric information that needs to be retained from the local Euclidean. By optimizing an objective function, the desired low-dimensional representation is obtained. Therefore, most manifold learning needs to construct the neighborhood information of the data. The construction of the neighborhood graph is the first step of most manifold learning, and also the key factor restricting the effect of manifold learning.

In recent years, experts have made outstanding contributions to the neighborhood selection problem of manifold learning. The neighborhood selection methods can be roughly divided into three types: fixed parameter method, dynamic parameter method, and no parameter method. Early neighborhood map construction is based on fixed parameters, such as the KNN algorithm and $\varepsilon$-ball algorithm. Both of these algorithms require manual setting of parameters, and the same parameters are used globally, which is not suitable for unevenly distributed data sets. Literatures[10][11] use residuals after dimensionality reduction. The difference value adaptively selects the k value, but this method still uses the same k value. The dynamic parameter method and the no-parameter method have been the focus of research in recent years. The dynamic parameter method mainly selects parameters adaptively according to the nature of the data itself[12], and the no-parameter method mainly uses sparse representation of theoretically adaptive parameters. For example, SPP, DSPP and other algorithms, most of these algorithms use sparse representation theory to improve the local manifold learning algorithm. Therefore, this paper mainly uses the sparse representation theory to improve the global manifold learning algorithm ISOMAP. While adaptively selecting the neighborhood, it also uses the sparse coefficient as the weight of the distance to improve the ability to discriminate the data.

2. Algorithm models

2.1. ISOMAP algorithm

In 2000, Tenenbaum JB proposed ISOMAP (Isometric feature mapping) and published it in Science. This algorithm is a representative global manifold learning algorithm based on the MDS algorithm. The MDS algorithm mainly maintains the Euclidean distance between the points in the high-dimensional and low-dimensional space is unchanged, so as to achieve the purpose of dimensionality reduction; however, for complex high-dimensional data, the Euclidean distance does not well express the dissimilarity of the global data; at the same time, due to the manifold In terms of local Euclidean, only the geodesic distance is a meaningful global measure on the manifold [13]. Therefore, the ISOMAP algorithm uses the shortest path to approximate the geodesic distance, and uses the geodesic distance matrix to replace the Euclidean distance matrix in the MDS. The specific steps of the ISOMAP algorithm are as follows:

Input: original data set $X = \{x_1, ..., x_N\}$, neighborhood parameter $k$ or $\varepsilon$, target dimension $d$; neighborhood of sample points is $N_{E_i} = \{1, 2, ..., N\}$.

Output: Data set $Y = \{y_1, ..., y_N\}$, where $y_i = (y_{i1}, ..., y_{id})$.

1. Generate neighborhood graph $G$

First calculate the Euclidean distance matrix between sample points $D_e$. The distance between any two points is expressed as $d_e(x_i, x_j)$. The following two methods can be used to construct a neighborhood graph.
a) K-NN: find the k nearest neighbors of each sample point through the matrix \( D_E \), \( i=1,2,\cdots N \).

b) \( \epsilon \)-ball: This method calculates the neighborhood set of samples by setting a fixed threshold \( \epsilon \) \( \{ x_j \mid j \neq i, d_i(x_i, x_j) \leq \epsilon \} \) \( i=1,2,\cdots N \).

Among them, the edge of the neighborhood graph is set to \( d_i(x_i, x_j) \) or \( \infty \), when two points are neighbors of each other, it is set to \( d_i(x_i, x_j) \), otherwise it is set to \( \infty \).

2. Calculate the geodesic distance matrix \( DG \). Use Dijkstra or Floyd algorithm to calculate the shortest path between any two points in the neighborhood graph. The formula(1) is to use Dijkstra to calculate the shortest path and use the shortest path to approximate the geodesic distance.

\[
d_{G_{ij}}(x_i, x_j) = \min \left\{ d_i(x_i, x_j), d_j(x_i, x_j) + d_j(x_j, x_i) \right\}
\]

\[
DG = \begin{bmatrix}
  d_{G_{1,1}}(x_1, x_1) & d_{G_{1,2}}(x_1, x_2) & \cdots & d_{G_{1,N}}(x_1, x_N) \\
  d_{G_{2,1}}(x_2, x_1) & d_{G_{2,2}}(x_2, x_2) & \cdots & d_{G_{2,N}}(x_2, x_N) \\
  \vdots & \vdots & \ddots & \vdots \\
  d_{G_{N,1}}(x_N, x_1) & d_{G_{N,2}}(x_N, x_2) & \cdots & d_{G_{N,N}}(x_N, x_N)
\end{bmatrix}
\]

Constructing d-dimensional Euclidean space embedding
First calculate the square matrix of the geodesic distance matrix: \( D \)

\[
D = \begin{bmatrix}
  d^2_{G_{1,1}}(x_1, x_1) & d^2_{G_{1,2}}(x_1, x_2) & \cdots & d^2_{G_{1,N}}(x_1, x_N) \\
  d^2_{G_{2,1}}(x_2, x_1) & d^2_{G_{2,2}}(x_2, x_2) & \cdots & d^2_{G_{2,N}}(x_2, x_N) \\
  \vdots & \vdots & \ddots & \vdots \\
  d^2_{G_{N,1}}(x_N, x_1) & d^2_{G_{N,2}}(x_N, x_2) & \cdots & d^2_{G_{N,N}}(x_N, x_N)
\end{bmatrix}
\]

Centralize the D matrix:

\[
B = -HDH / 2
\]

\[
H = I_N - \frac{ee^T}{N}
\]

Decompose the eigenvalues of the matrix \( B \), and set the eigenvalues to be sorted from large to small \( \lambda_1 \geq \lambda_2 \cdots \lambda_{dN} \). Corresponding feature vectors \( \mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_e \) and take the largest \( d \) features and their corresponding feature vectors.

\[
Y = U \Sigma^{1/2}
\]

2.2. Adaptive neighborhood ISOMAP algorithm based on sparse representation (RP-ISOMAP)
Sparse representation theory was originally applied in signal processing, which can effectively reduce the cost of signal processing and improve the efficiency of source information compression. With the continuous improvement and development of sparse theory, and the sparse representation theory can automatically construct the relationship matrix between data, and the sparse representation has a natural discriminating ability; driven by these advantages, the sparse representation theory has been successfully applied to signals and images Neighborhoods such as processing, pattern recognition, and machine learning. Sparse representation has a concise mathematical model. Let a signal be a column vector \( x_i \in R^n \), containing an over-complete dictionary of all elements \( X \in R^{m \times e}, X = \{ x_1, x_2, \cdots, x_e \} \). The basic idea of sparse representation is to represent an element linearly with as few elements as possible in the dictionary. Assuming that the sparse representation coefficient is \( s_i = [s_{i,1}, s_{i,2}, \cdots, s_{i,1}, 0, s_{i,2}, \cdots, s_{i,n}] \), the optimized model is as follows:

\[
\min ||s_i||_0 \quad \text{s.t} \quad x_i = Xs_i
\]

The \( l_0 \) norm represented \( ||s_i||_0 \) by is equivalent to the number of non-zeros in the solution. Since it is non-convex, it is an NP-hard problem to solve the sparse coefficient. In order to solve this problem,
\( l_1 \) norms are introduced, and it is proved that when \( s_i \) is sparse, the solution to minimize the \( l_1 \) norm is equal to the solution to minimize the \( l_0 \) norm. The optimization model is given below

\[
\min ||s_i|| \quad \text{s.t} \quad x_i = Xs_i
\]  \hspace{1cm} (7)

The key to the success of the ISOMAP algorithm lies in the construction of the neighborhood map. Too large or too small \( k \) values will cause distortion of low-dimensional embedded manifolds. At the same time, the parameter adjustment process of \( k \) value is a very troublesome process. In this paper, the sparse representation can automatically construct the adjacency matrix and its natural discriminative features to improve the ISOMAP algorithm, and at the same time use the sparse coefficient as the weight of the distance to improve the classification performance of the data after dimensionality reduction.

Assuming input datas \( X \in \mathbb{R}^{m \times n}, X = \{x_1, x_2, \cdots, x_n\} \), calculate the sparse coefficient of each point \( s_{i,i}, i=1, \cdots, n \). Using the sparse representation coefficients instead of \( k \)-nearest neighbors, the calculated coefficients \( s_{i,j} \) reflect the geometric similarity between the datas to a certain extent. When \( s_{i,j} = 0 \), \( x_j \) is considered not in the neighborhood of \( x_i \), otherwise \( x_j \) is considered in the neighborhood of \( x_i \). For the time \( s_{i,j} \neq 0 \), not all the neighborhoods are in \( x_i \), because when \( s_{i,j} \) is very small, and the correlation is not strong, so the reciprocal of the coefficient is used to weight the distance to ensure that the smaller the coefficient, the greater the distance between the two points. The optimization function is as follows. Therefore, when calculating the neighborhood graph \( G \), the matrix elements are directly used for multiplication.

\[
\min ||s_i|| \quad \text{s.t} \quad x_i = Xs_i
\]  \hspace{1cm} (8)

among them \( S' = \begin{bmatrix} 0 & 1 & \cdots & 1 \\ s_{i,2} & 0 & \cdots & s_{i,n} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 0 \end{bmatrix} \), \( D_e = \begin{bmatrix} d_i(x_1, x_1) & d_i(x_1, x_2) & \cdots & d_i(x_1, x_n) \\ d_i(x_2, x_1) & d_i(x_2, x_2) & \cdots & d_i(x_2, x_n) \\ \vdots & \vdots & \ddots & \vdots \\ d_i(x_n, x_1) & d_i(x_n, x_2) & \cdots & d_i(x_n, x_n) \end{bmatrix} \) is euclidean distance matrix, \( \odot \) is to multiply element by element, and \( \text{And stipulate } 1/0 = \infty \). Therefore, when constructing the neighborhood graph, not only can the neighborhood be automatically identified, and the non-adjacent edges are set to infinity, but also the weight of the non-adjacent data can be enlarged by the reciprocal weighting of the sparse coefficient to improve the discriminability of the data. For the solution of Equation 7, this paper uses the base tracking algorithm to transform it into a linear programming problem. The remaining steps are the same as the ISOMAP algorithm. Table 1 gives a brief introduction to the RP ISOMAP algorithm.

| Table 1. RP_ISOMAP algorithm steps |
|-------------------------------------|
| S1: Solve the sparse coefficients \( s \) of each data using sparse representation theory to get the sparse coefficient matrix \( s \). |
| S2: Construct the neighborhood graph \( G \) by sparse coefficient matrix \( s \) and Euclidean distance matrix \( D \). |
| S3: Calculate the geodesic distance matrix \( DG \). |
| S4: Calculating low-dimensional embedding using classic MDS algorithm |

The following figure 1-4 preliminary explores the visualization effect of RP ISOMAP, which is compared with ISOMAP under the premise of \( k=8, 16, 24 \). Figure 1, using the S-shaped data set
commonly used in dimensionality reduction, can be seen from the figure, the RP-ISOMAP algorithm dimensionality reduction effect is closer to the potential distribution structure of the data. Figure 2, using the Balancescale data set in UCI, it can be seen from the figure that the RP-ISOMAP algorithm not only has a good visualization effect, but also the distribution of similar data is concentrated after dimensionality reduction, and the heterogeneous data is relatively discrete, which improves the ability to discriminate data. Figure 3 and Figure 4, using the IRIS and WINE data sets respectively, it can be seen that the same effect as in Figure 2 is obtained.

Figure 1. S-data using RP-ISOMAP and ISOMAP to reduce dimensionality

Figure 2. Balancescale using RP-ISOMAP and ISOMAP to reduce dimensionality

Figure 3. IRIS using RP-ISOMAP and ISOMAP to reduce dimensionality
3. Simulation experiments

In the experiments of intrusion detection, the KDD CUP 99 intrusion detection data source (add literature) provided by DARPA is often used for experiments. The DARPA intrusion detection data set is about 5 million pieces, and the data redundancy is high and the noise is high. In order to reduce the training time of the classifier and ensure the scientificity of the experiment, this article selects the enhanced version of the NSL-KDD dataset of KDDCUP 99. The software and hardware configuration of the experiment is as follows: The CPU is Intel Core i5-6300HQ, the memory is 8G, and the software platform matlab2017a used.

3.1. Data set introduction

The NSL-KDD dataset has a total of 42 attributes, the first 41 are feature attributes, and the last one is a tag attribute. Characteristic attributes include numeric data and character data. For character data, use numerical processing, such as attribute protocol_type, there are three possible values: TCP, UDP, ICMP, which are encoded as 0, 1, 2 in data preprocessing. Normalize the numerical data. The label attributes include 4 types of attacks and one type of normal behavior, respectively labeled Dos, Probe, R2L, U2R, Normal. The label attributes in the original data set include 40 categories, each of which belongs to the above four types of attacks One is shown in Table 2 below. In data preprocessing, all attacks belonging to the same category are represented by the same data value. The training data set is KDD Train+, and the test data is KDD Test+. The statistical data is shown in Table 3 below.

Table 2. Subtypes of various attack types.

| Type | Attack Types                                      |
|------|---------------------------------------------------|
| DoS  | (10) Back, Land, Neptune, Pod, Smurf, Teardrop, Apache2, Udpstorm |
|      | Processesstable, Mailbomb                         |
| Probe| (6) Satan, Ipsweep, Nmap, Portsweep, Mscan, Sain |
| R2L  | (16) Guess_Password, Ftp_write, Imap, Phf, Multihop, Warezmaster, Warezclient, Spy, Xlock, Xsnoop, Snmpguess, Snmpgetattack, Http Tunnel, Sendmail, Named, Worm. |
| U2R  | (8) Buffer_overflow, Loadmodule, Rootkit, Perl, hseptunnel, ps, sqittack, xterm |

Table 3. Statistical results of the data sets

|                  | Normal | DOS    | Probe | R2L    | U2R    | Total  |
|------------------|--------|--------|-------|--------|--------|--------|
| KDD Train+       | 67343  | 45927  | 11656 | 995    | 52     | 125973 |
| KDD Test+        | 9711   | 7458   | 2421  | 2754   | 200    | 22544  |

Figure 4. Wine data using RP_ISOMAP and ISOMAP to reduce dimensionality
3.2. Comparative Experiment

The focus of the experiment in this paper is to explore the feature extraction effect of the improved ISOMAP algorithm on intrusion detection data. Therefore, this paper uses the SVM classifier to compare the RP_ISOMAP, ISOMAP processed features and unprocessed features. The main evaluation indicators of the experiment are There are four: accuracy rate AC, false positive rate (FPR), training time (Train T), test time (Test T). In order to avoid the influence of the amount of data on the experiment, 10%, 30%, 60% and 90% of the data were randomly selected for experiments. Each group of experiments randomly selected equal proportions of test data and training data, and for each group of experiments, Randomly draw the average multiple times. For the setting of the intrinsic dimension, this article explores the relationship between the detection rate and the dimension. As shown in the figure, the intrinsic dimension set in this article is 16, in the exploration experiment of the intrinsic dimension, the amount of data is randomly drawn 90% of the training set and the test set As training and testing respectively, it can be seen from the figure that the detection rate is the highest when d=16, so the eigendimension set in this paper is 16.

![Figure 5. Relationship between dimension and accuracy](image)

In the experiment, two parameters of SVM are set: sigma=1, c=1. The accuracy rate of the experiment is expressed by AC, the false alarm rate is expressed by FPR, and the formula is its expression (8). Where TN represents the correct number of abnormal behavior tests, TP represents the correct number of normal behavior tests, FP represents the number of normal behavior test errors, TP represents the correct number of normal behavior tests, and N is the total number of tests.

\[ AC = \frac{TN + TP}{N} \]
\[ FPR = \frac{FP}{TP + FP} \]

Figure 6 shows the reduction of 16 dimensions using the ISOMAP and RP_ISOMAP algorithms under different data volumes, and the comparison of classification experiments with data without dimensionality reduction. The experimental indicators mainly use the classification detection rate. It can be seen from the figure that with the same processing method, the detection rate increases slightly as the amount of data increases; under the premise of the same amount of data, the data detection rate after RP_ISOMAP dimensionality reduction is slightly higher, there are two main reasons: 1) Using the sparse representation theory to adaptively construct the neighborhood graph, compared with the neighborhood graph constructed by setting the K value, it can better adapt to the distribution characteristics of the data. 2) The use of sparse coefficients to weight the Euclidean distance can shorten the distance between points with strong spatial correlation and improve the performance of data discrimination. Therefore, the detection rate will be higher than that of ISOMAP processed and unprocessed data. Of course, from the figure, you can also get pure ISOMAP processing data, because of the loss of information, it will bring the price of the detection rate.
Table 4 shows the results of randomly extracting 90% of the training data and test data for classification experiments. The results in the table are the average of 5 experiments. It can be seen that the RP_ISOMAP processed data has a high detection rate and a low false alarm rate. At the same time, the training and testing time after dimensionality reduction is reduced, which fully proves the importance of dimensionality reduction processing in data analysis. The RP iSOMAP algorithm not only reduces the time consumption, but also improves the classification detection rate.

Table 4. Indicators of data processed using different processing methods

|               | AC     | FPR    | Train T | TestT   |
|---------------|--------|--------|---------|---------|
| Not Processed | 0.9707 | 0.020  | 4.2150s | 2.1259s |
| ISOMAP        | 0.9612 | 0.0441 | 2.7680s | 1.4590s |
| RP_ISOMAP     | 0.9840 | 0.0060 | 2.5638s | 1.592s  |

Table 5 shows the results of the classification experiment in which 90% of the training data and test data are randomly extracted. The results in the table are the average of the detection rate of each attack type in 5 experiments. It can be seen from the table that the data processed by RP_ISOMAP has a high detection rate. For the same algorithm, the Normal detection rate is the highest, and the U2R detection rate is the lowest. Although the U2R detection rate of the data processed by RP_ISOMAP is very low, the other two groups of experiments have increased by 0.071 and 0.122., respectively.

Table 5. Accuracy of various attacks

|               | Normal | Dos    | Probe  | R2L    | U2R    |
|---------------|--------|--------|--------|--------|--------|
| Not Processed | 0.980  | 0.9820 | 0.9790 | 0.910  | 0.444  |
| ISOMAP        | 0.9569 | 0.9960 | 0.9760 | 0.9000 | 0.566  |
| RP_ISOMAP     | 0.9940 | 0.9840 | 0.980  | 0.9400 | 0.637  |

4. Conclusion

In this paper, the sparse representation is used to adaptively select the neighborhood, which effectively overcomes the difficulties caused by artificial parameter selection, can adapt to unevenly distributed data, and uses sparse coefficients for weighting, which effectively improves the ability of data discrimination. The superiority of the algorithm is shown on the artificial data set and NSL-KDD data, which effectively improves the accuracy of intrusion detection. Of course, the algorithm still has deficiencies. For example, the incremental problem is not considered. In each experiment, the training data and test data are reduced in dimensionality using a batch mode. In the next step of research work, we should focus on incremental ISOMAP research.
References

[1] Wang H, Gu J, Wang S. (2017) An effective intrusion detection framework based on SVM with feature augmentation[J]. Knowledge-Based Systems, 136: 130-139.

[2] Devan, P., & Khare, N. (2020). An efficient XGBoost-DNN-based classification model for network intrusion detection system. Neural Computing and Applications, 1-16.

[3] Wei, J., Long, C., Wan, W., Zhang, Y., Zhao, J., & Du, G. (2019). An Effective RF-based Intrusion Detection Algorithm with Feature Reduction and Transformation. 3-5

[4] Sellappan, D., & Srinivasan, R. (2020). Association Rule-Mining-Based Intrusion Detection System With Entropy-Based Feature Selection: Intrusion Detection System. In Handbook of Research on Intelligent Data Processing and Information Security Systems (pp. 1-24).

[5] Thaseen, I. S., & Kumar, C. A. (2017). Intrusion detection model using fusion of chi-square feature selection and multi class SVM. Journal of King Saud University-Computer and Information Sciences, 29(4), 462-472.

[6] Aghdam, M. H., & Kabiri, P. (2016). Feature Selection for Intrusion Detection System Using Ant Colony Optimization. IJ Network Security, 18(3), 420-432.

[7] Serpen, G., & Aghaei, E. (2018). Host-based misuse intrusion detection using PCA feature extraction and kNN classification algorithms. Intelligent Data Analysis, 22(5), 1101-1114.

[8] Roweis, S. T., & Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. science, 290(5500), 2323-2326.

[9] Tenenbaum, J. B., De Silva, V., & Langford, J. C. (2000). A global geometric framework for nonlinear dimensionality reduction. science, 290(5500), 2319-2323.

[10] Kouropetova, O., Okun, O., & Pietikäinen, M. (2002). Selection of the optimal parameter value for the locally linear embedding algorithm. FSKD, 2, 359-363.

[11] Samko O, Marshall A D, Rosin P L (2006) Selection of the optimal parameter value for the Isomap algorithm[J]. Pattern Recognition Letters, 27(9): 968-979.

[12] İnkaya T, Kayalıgil S, Özdemirel N E (2015). An adaptive neighbourhood construction algorithm based on density and connectivity[J]. Pattern Recognition Letters, 52: 17-24.

[13] Saxena, Ashutosh, Abhinav Gupta, and Amitabha Mukerjee (2004). "Non-linear dimensionality reduction by locally linear isomaps." International Conference on Neural Information Processing. Springer, Berlin, Heidelberg, 23:13-24.