Computer and Information Science; Vol. 5, No. 4; 2012
ISSN 1913-8989 E-ISSN 1913-8997
Published by Canadian Center of Science and Education

Improved SOM-Based High-Dimensional Data Visualization Algorithm

Wang Zhisheng[1] & Xu Xiaobing[1]

[1] Management Science and Engineering, Business School, University of Shanghai for Science and Technology, Shanghai, China

Correspondence: Wang Zhisheng, Management Science and Engineering, Business School, University of Shanghai for Science and Technology, Shanghai 200093, China. E-mail: greatwnag@163.com

Received: May 10, 2012 Accepted: May 21, 2012 Online Published: June 15, 2012

doi:10.5539/cis.v5n4p110 URL: http://dx.doi.org/10.5539/cis.v5n4p110

Supported by Leading Academic Discipline Project of Shanghai Municipal Government, Project Number: S30504, and the Innovation Fund Project for Graduate Student of Shanghai Project Number: JWCXSL1102

Abstract

In this paper, a new high-dimensional data visualization algorithm based on the Self-Organizing Map (SOM) is proposed. It is named TDSOM (three-dimensional self-organizing map) to describe its special characteristics. TDSOM trains the high-dimensional data with SOM network and projects it into particular point sets in the three-dimensional coordinate system. In the three-dimensional coordinate system, the x axis represents attributes of the original data set; the y axis represents the weight of each attribute; the z axis represents different categories of the mapping result. The most important is that researchers can watch the three-dimensional model from different viewpoints by rotating it and gain some interesting patterns. Through the experiment, TDSOM is proved to be much more accurate and more analytical than the traditional methods in displaying the high-dimensional data. The main innovation of the new TDSOM algorithm is the presentation of large data in three-dimensional coordinate system which provides a much wider view than the two-dimensional one. What's more, users are able to discover some interesting patterns according to their own research areas through the model. The algorithm can be widely applied in areas such as data mining, pattern recognition and so on.

Keywords: self-organizing map, neural networks, data visualization, clustering analysis

1. Introduction

Data visualization, witch is used to display multi-dimensional data graphically, has been widely applied in pattern recognition, image processing and so on. There are many algorithms being used in data visualization, such as scatter plot matrices(Yu, Zhou, & Zhang, 2007) which displays the different combination of data items with many child graphics, parallel coordinates (Wegman, 1990) which rearrange the order of data items in two-dimensional space, stacked techniques (Feiner & Beshers, 1900) which embed one coordinate system in another one, cone trees which divide the n-dimensional data set to subspaces, Chernoff face (Soete & Corte, 1985) and star chart (Willis, 1992) which convert the multi-dimensional data set into specific graphics and so on. These methods have made great contribution to visualizing the multi-dimensional data set, but it is hard to find the characters through them when analyzing the data set with more dimensions. To solve the problem, researchers put forward dimension reduction methods which visualizing data sets after projecting the high-dimensional data to the low-dimensional space. These methods effectively extract the distribution characteristics of high-dimensional data set and reduce the complexity of final visualization result. However, this requires a higher performance of the mapping algorithm. Representative ones include the principal component analysis (Kambhatla & Leen, 1997), multidimensional scaling (Friedman & Tukey, 1974), self-organizing map (Kohonen, 1990; Vesanto, 1999) and so on. And researchers are keeping improve these methods continuously.

In 1982, Professor Kohonen from Finland proposed the neural network model named SOM. SOM is an unsupervised neural network projecting high-dimensional data to low dimensional space with competition learning. No supervision means not pointing the output when the network was being trained. And competition learning is the selection rule of "survival of the fittest". SOM produce a two-dimensional graph by means of projecting the input space into output space. In the graph, input space records are projected to the neural nodes with which they have the maximum similarity. And this algorithm has high execution efficiency. However, the traditional SOM algorithms usually visualize data with consistent grid format, so the size of the network had to be designed in advance. It makes the presentation capability of the output graph largely limited. What's more, SOM does not reserve the distance information of network nodes. Although this fault can be made up by designing color map based on U-matrix, the structure and distribution of data were usually presented in a distortive format. Dynamic self-organizing map (Alahakoon, Halgamuga, & Srinivasan, 2000), an improved algorithm of the SOM, generates a two-dimensional graphic with a grid which grew dynamically. But the irregular network shape is not able to visualize the final result appropriately. The visualization-induced self-organizing map (Yin, 2002) preserves the distance information of data as well as the topological structure. But it cannot present the characteristics of different attributes. The polar self-organizing map (PolSOM) (L. Xu, Y. Xu, & Chow, 2010) and its improved probabilistic polar self-organizing map (PPoSOM) (L. Xu, Y. Xu, & Chow, 2011) which are based on SOM are recently put forward. They visualize data with the two-dimensional polar coordinates and project the radius and angle of the coordinates to the distance and characteristics of the data. The PolSOM and PPoSOM present the distance diversity between the network nodes and preserve the topology. What's more, they present the value weight and feature characteristic with polar coordinates' feature effectively. However, with the increase of record amounts and the growth of data dimensions, these methods based on polar coordinates may lead to the confusion of the visualization result.

A new algorithm based on SOM and used to visualize high-dimensional data is named as three-dimensional SOM, and we call it TDSOM for short. TDSOM discards the traditional model that projects data to the two-dimensional coordinates, but projects data to the three-dimensional coordinates to enforce the effect of the visualization. TDSOM projects the three coordinate variables of the three-dimensional coordinates to the attributes, values and category of the dataset.

2. SOM Algorithm and TDSOM Algorithm Principle

2.1 SOM Algorithm

Early SOM algorithm mainly projects the multi-dimensional to the two-dimensional space. The two-dimensional plane consists of many neural nodes. In the graph, each node has a weight vector with the same dimension and distribute on the grid made up of rectangles or hexagons. In SOM, an input datum x_i is represented as a d-dimensional feature vector $x_i=(x_{i1}, x_{i2},..., x_{id})^T$. And each neural node is represented as the corresponding vector $w_i=(w_{j1}, w_{j2},..., w_{jd})^T$. At the beginning of the training, the algorithm select one datum x randomly and compute the similarity of the datum x and the node j. The similarity is measured by Euclidian distances, and the node which has the minimum distance is the winning one:

$$\left\|x-w_c\right\| = min\left\|x-w_j\right\|, j=1,\ldots,N \tag{1}$$

where N is the number of neurons.

After getting the best matched node, the algorithm updates the weights of its neighbor nodes. Usually, the neighborhood function should be defined before updating them. And Gaussian function is usually chosen:

$$h_{jc} = exp\left\{-\left\|p_j - p_c\right\|^2 / \left[2\delta\left(t\right)^2\right]\right\}, j \in N_c \tag{2}$$

where p_j and p_c are the coordinates of neuron j and c, respectively, N_c is the neighboring set of winning neuron c, $||p_j-p_c||$ is the distance between j and c, $\delta(t)$ is the neighboring radius that monotonically decreases with time.

The weight updating formula is:

$$w_j\left(t+1\right) = w_j\left(t\right) + \zeta\left(t\right) h_{jc}\left[x\left(t\right) - w_j\left(t\right)\right], j \in N_c \tag{3}$$

Where $\zeta(t)$ is the learning rate that monotonically decreases with time.

After training, input data with similarity are projected onto adjacent neural nodes in the grid of the output space. Thus, SOM is able to reserve the topological relationship of the input space. But due to more than one datum are projected onto the same neural node, the inter-point distance is not preserved. What's more, the format of the output space usually is consistent, SOM is not able to display the intern-neuron distance.

2.2 TDSOM Algorithm Principle

TDSOM is the newly proposed algorithm in the article to exhibit the characteristics of high-dimensional input space. Unlike the traditional algorithms, its visualization model is constructed in the three-dimensional coordinate system which is much wider than the two-dimensional space. Records data are projected to the certain coordinates of the three-dimensional space after being processing by the algorithm. In the direction of *X* axis, the three-dimensional model is divided into *d* bar areas (where *d* is the dimensions of the input data) horizontally, and different *x* value represents different attribute of the data set. In the direction of *Y* axis, the vertical value of the point represents the weight of the corresponding attribute. In the direction of *Z* axis, different value means different class of the points. Each node of the neural network is represented with a vector of the same dimension $w=(w_1, w_2, \ldots, w_d)^T$ (where *d* is the dimension of the input data). After selecting the input datum $x=(x_1, x_2, \ldots, x_d)^T$ from the data set, the algorithm gets the winning neural node according to following formula:

$$\left\|w_c - x\right\| \le \| w_j - x \|, j = 1,\ 2,\ \ldots, n \tag{4}$$

After this, TDSOM update the winning node *c* and its neighbor nodes with Formula (2) following the below:

$$\begin{aligned} w_j\left(t+1\right) &= w_j\left(t\right) + \eta_1 h_{jc}\left[x\left(t\right) - w\left(t\right)\right],\ if\ j = c \\ w_j\left(t+1\right) &= w_j\left(t\right) + \eta_2 h_{jc}\left[x\left(t\right) - w\left(t\right)\right],\ else\ if\ j \in N_c \end{aligned} \tag{5}$$

where η_1 and η_2 are the learning rates of the *t-th* step, and $\eta_1 > \eta_2$. η_1 is in the region *[0, 0.3]*, and represents the learning rate of the winning node towards its current training sample. While η_1 is in the region *[0, 0.1]*, and η_2 represents the learning rate of the neighborhood nodes learning to the current training sample. The learning rates have effects on the convergence. While η_1 andη_2 increase, the training process is speeded up.

Set the training period, and start to train the network according to it. During the training, TDSOM save the times of each node chosen to be the winning one w_j *(j=1, ..., N)*. And during the last training period, the winning nodes of all input data J_c are recorded. After the training, the *U*-matrix that stores the distances information of the neural nodes in the network is computed. Then the neural network is visualized according to gray model generated from the *U*-matrix. In the result, the distance between similar neural nodes is closer, so their colors are similar. Presume that the distance between nodes is closer, their color is deeper. Finally, some dark areas come to being in the plane and each of them represents one clustering.

According to the information of input data projecting to clustering areas, the input data subset corresponding to certain clustering can be recorded. Follow the formula:

$$x \in C_k,\ k \in \left\{1, \ldots,\ m\right\},\ if\ x \in J_c\ and\ J_c \in C_k \tag{6}$$

where J_c is the winning node of datum *x*, *m* is the clustering number divided according to *U*-matrix and C_k is the *k-th* clustering.

After the above processing, TDSOM projects one datum onto *l* points of the coordinate system, and they are represented with *Pl(l=1,2,...,d)* (where *d* is the dimension of the datum *x*):

$$Pl_x = i,\ \left(i = 1, 2, \ldots, d\right) \tag{7}$$

$$Pl_y = x_i,\ \left(\ i = 1, 2, \ldots, d\right) \tag{8}$$

$$Pl_z = k,\ \left(x \in C_k\right) \tag{9}$$

where Pl_x, Pl_y and Pl_z are the *x*, *y* and *z* coordinates of the points.

The holistic executing steps of TDSOM algorithm are as follows:

Step(1): Normalize the input data. Initialize the weight of all neural nodes randomly, and initialize their coordinates in the output space. Set the training period *T*.

Step(2): Randomly select an input datum and find its corresponding winning neural node by *Equation (4)*.

Step(3): Update the weights of winning node and its neighborhood set by *Equation (5)* Record the winning times

W_j.

Step(4): If the training period reaches T, stop and go to *Step(5)*. Otherwise, reduce the learning rate η_1 ,η_2 and neighborhood radius $\delta(t)$ and go back to *Step(2)*.

Step(5): Draw the contour map according to W_j which is the numbers of records being projected to the corresponding neural node. And draw the gray graph according to the *U*-matrix. Get clustering according to *Equation (6)*.

Step(6): Update the coordinates by *Equation (7)*, *Equation (8)* and *Equation (9)* according to clustering result and winning neuron x_w, and draw the three-dimensional model.

Through the training above, every input datum have a points set in the three-dimensional coordinate system. Data with the similar attributes gather on the same group, and researchers can compare the distribution of all clusters' attributes and detect their weight from different viewpoints. The final model is made up of points set but not neural nodes, so the distance of points is reserved.

3. Experimental Results

Wine recognition data set which was released in UC Irvine learning repository is one of the most widely applied data sets. These data are the results of a chemical analysis of wines grown in the region in Italy but derived from three different cultivars. The analysis determined the quantities of *13* constituents found in each type of wines. It is hard to represent characteristics of the data set with so many attributes clearly through traditional data visualization methods. So the effectiveness of TDSOM visualizing high-dimensional data is easy to be defected by this wine recognition data set. After being trained by the TDSOM, the results are listed as follows:

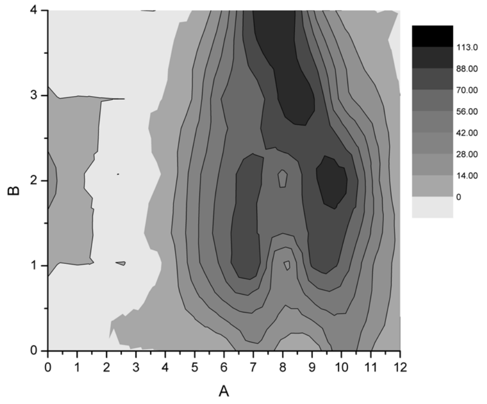

Figure 1. SOM contour map

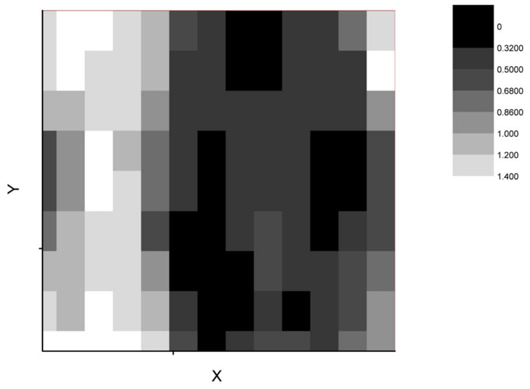

Figure 2. SOM U-matrix grayscale

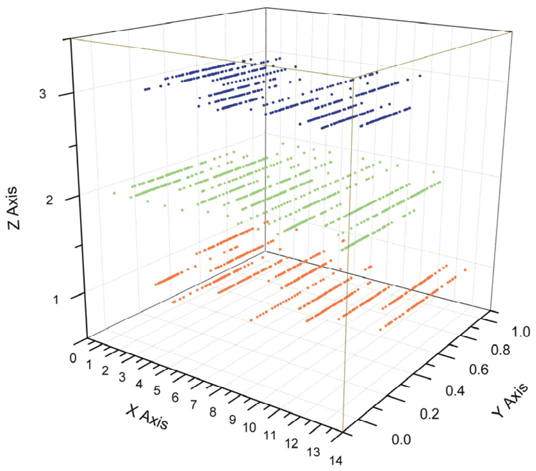

Figure 3. TDSOM projecting map (horizontally)

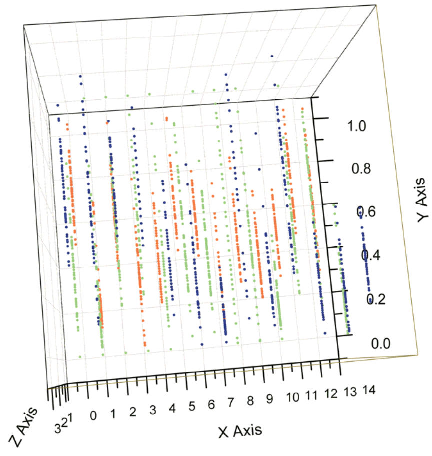

Figure 4. TDSOM projecting map (vertically)

In Figure 1, most data are mainly projected to three areas where their colors are darker than edges and gathered as three clusters. Figure 2 is the SOM U-matrix grayscale which is the traditional method of visualization. Although clusters' Edges are not clear enough, the result also contains three clusters.

Figure 3 and Figure 4 are screenshots of the three-dimensional model which are taken horizontally and vertically. In the two figures, the numbers of 1 to 13 in x axis represent thirteen attributes of the wine recognition data set; the value of y coordinate represent the weight of attributes; 1, 2 and 3 on the z axis represent three different areas. In Figure 3, researchers can distinguish three different clusters projected from original data set, and different layers contain the corresponding points set of its clustering. Rotating the three-dimensional model and observing the Figure 4, researchers can find the distribution of front attributes easily. For example, observe the '3' attribute of three clusters, researchers can easily find out that although the distribution of Cluster 1 is disperse, the values of the set mainly focus on the area which values are smaller; that Cluster 2 middle; and that Cluster 3 bigger. According the above illustration, researchers may get knowledge of the distribution of all the attributes of different clusters by rotating the model and detecting it from different view point.

Based on the experiment results, it is noted that obvious improvement has been made in visualizing high-dimensional data set by TDSOM, and that the distribution of each attribute of different clusters is effectively represented.

4. Conclusion

In this paper, a new improved mapping method, TDSOM, is proposed for visualization and projection of high-dimensional data. The categories, attributes and weight of the data set are represented clearly, and the distribution differences of clusters are also obvious. What's more, researchers can find the characteristics of each cluster. However, TDOM may be modified for further improvement. During the experiment, one accurate model usually contains thousands of millions of points. And this requires high-performance computer system. With the development of computer technology, the problem may be solved in the near further. At last, what's worth mentioning, due to the upstanding visualization result, TDSOM is worth widely applying in relevant areas.

References

Alahakoon, D., Halgamuga, S., & Srinivasan, B. (2000). Dynamic self-organizing maps with controlled growth for knowledge discovery. *IEEE transactions on neural networks, 11*(3), 601-614. http://dx.doi.org/10.1109/72.846732

Feiner, S., & Beshers, C. (1900). Visualizing n-dimensional virtual worlds with n-vision. *Computer Graphics, 24*(2), 37-38.

Friedman, J. H., & Tukey, J. W. (1974). A projection pursuit algorithm for exploratory data analysis. I*EEE Transactions on Computers, C-23*(9), 881-890. http://dx.doi.org/10.1109/T-C.1974.224051

Kambhatla, N., & Leen, T. (1997). Dimension reduction by local principal component analysis. *Neural Computation, 9*(7), 1493-151. http://dx.doi.org/10.1162/neco.1997.9.7.1493

Kohonen, T. (1990). The Self-organizing map. *Proceedings of the IEEE, 78*(9), 1464-1480.

Soete, G., & Corte, W. (1985). On the perceptual salience of features of chernoff faces for representing multivariate data. *Applied Psychological Measurement, 9*(3), 275-280. http://dx.doi.org/10.1177/014662168500900305

Vesanto, J. (1999). SOM-based data visualization methods. *Intelligent Data Analysis, 3*(2), 111-126. http://dx.doi.org/10.1016/S1088-467X(99)00013-X

Wegman, E. J. (1990). Hyperdimensional data-analysis using parallel coordinantes. *American Statistical Association, 85*, 664-675.

Willis, J. P. (1992). *Star plots diagrams with strong visual impact for simultaneously displaying variations in 4 to 9 sample characteristics.* pp. 75-79. Barren River Resort, KY: World Scientific Publ Copteltd.

Xu, L., Xu, Y., & Chow, S. (2010). PolSOM: A new method for multi-dimensional data visualization. *Pattern recognition, 43*(4), 1668-1675. http://dx.doi.org/10.1016/j.patcog.2009.09.025

Xu, Y., Xu, L., & Chow, S. (2011). PPoSOM: A new variant of PolSOM by using probabilistic assignment for multidimensional data visualization. *Neurocomputing, 74*(11), 2018-2027. http://dx.doi.org/10.1016/j.neucom.2010.06.028

Yin, H. (2002). ViSOM-A novel method for multivariate data projection and structure visualization. *IEEE transactions on neural networks, 13*(1), 237-243. http://dx.doi.org/10.1109/72.977314

Yu, X. S., Zhou, N., & Zhang, F. F. (2007). Research on Methods of High-dimensional Data Visualization. *Information Science, 25*(1), 117-120.