An Unsupervised Neural Network Algorithm SOMO for Continuous Optimization Problems

Anzhi Qi
Liaoning Jianzhu Vocational College, Liaoyang, Liaoning, 111000

Keywords: SOMO, Continuous Optimization Problems, Unsupervised Neural Network Algorithm

Abstract: The application of self-organizing neural network in customer classification is discussed. The concept of customer classification, index selection, classification method selection and SOM (Self Organization Map) clustering method are discussed, and a customer classification method based on SOM , Namely, the indexes of RFM (Recency; Frequency, Value, Monentary) are given, and the customers are classified according to the calculation of the comprehensive index and the relative learning results of each index, and the simulation calculation is performed, The simulation results are categorized to validate the algorithm.

1. Introduction

Because most economic and humanistic studies have a hard time getting a complete data set, the sources of information may be incomplete and illusory, even there are exceptions and counter-examples. Some traditional clustering methods have strict requirements on the required data. Therefore, Difficult to do these jobs. Due to its strong ability of nonlinear mapping, self-learning and fault tolerance, artificial neural network has become a hot research topic in recent years because of its application to pattern classification. The self-organizing feature mapping (SOMO) neural network can reasonably make decisions on complex problems based on the knowledge gained from learning. It uses a clustering method without teacher demonstration to cluster the sample data At the same time, it has the property of maintaining topological orderliness and feature extraction, and is especially suitable for solving a variety of classification and identification problems. However, there are not many structural optimization problems in SOMO networks involved in the current research results. In this paper, SOMO neural network is applied to the pattern classification of vectors on two-dimensional plane. Based on MATLAB7.0 platform, SOMO neural network Network, the SOMO network classification model structure optimization of a class of problems, that is, the output node shape and topology of the classification results, tested under different training steps, given the SOMO neural network Weight vector adjustment process and SOMO model classification effect.

2. SOM neural network structure

Self-organizing feature mapping algorithm is a teacherless teaching clustering method, which can map any input mode into one-dimensional and two-dimensional discrete graphics at the output layer, and keep its topology unchanged. That is, under the condition of no teacher's teaching, through the self-organized learning of the input mode, the classification results are expressed at the competition level. Kohonen has proved that the network can iteratively study the input mode to make the probability density distribution of the connection weight vector consistent with the input mode probability distribution, that is, the spatial distribution of the connection weight vector can reflect the statistical characteristics of the input mode. For the input mode, different regions of the neural network have different response characteristics. Normally only one or some of the neurons in the area have a positive response to the input pattern. As shown in Figure 2, a two-dimensional array of distributed self-organizing feature maps the input pattern X = (x1, x2, ..., xi, ..., xn) T in parallel to each neuron in the network, Corresponding to a weight vector M, it is an adjustable parameter of the network.

For input mode X, each neuron's weight vector is compared to it, and the nearest weight vector is automatically adjusted until it coincides with the direction of a certain maximum component of input pattern X. For input pattern X, the winning neuron J * is first determined to satisfy JX - * = min {JXii -}, and then to the surrounding Nj * (t) centered at J * The neuron's weight vector is adjusted as follows. Wij (t + 1) = Wij (t) + η (t) [xi (t) -Wij (t)], i = 1, 2, ..., n. η (t) is the learning coefficient, decaying with time, and can be defined as η (t) = 0.9 [1-t / 10000].

SOMO neural network algorithm iterative process is as follows:

1) Initialize Weights of the link weights of the N input neurons to the output neurons are given. The set Sj of the output neurons j adjoining neurons is selected. Sj (0) denotes a set of adjoining neurons of neuron j at t = 0, and Sj (t) denotes a set of adjoining neurons of neuron j at time t. The area Sj (t) decreases with time.

2) Provide a new input mode X.

3) Calculate Euclidean distance dj, as shown in equation (1), and calculate a neuron j * with minimum distance, that is, determine a neuron k such that for any j, there is dk = minj (dj)

4) Give a neighborhood of neighborhood Sk (t).

5) Modify the weights of output neurons j * and their neighboring neurons according to (2). wij (t + 1) = wij (t) + η (t) [xi (t) - wij (t)]

3. Self-organizing neural network application simulation

Using the neural network toolbox in Matlab software, the computer simulation of normal distribution shows the behavior of SOMO algorithm, the input vector of network X (n) = [x1 (n), x2 n), x2 (n) is a standard normal distribution random number, the number of learning samples is n = 1000, the output of the network is a grid of 100 neurons arranged in a 10-line and a 10-line two-Grid, learning rate η (t) = 0.2 (1-t / T), training times T = 800.

SOMO neural network learning process in three stages, Figure 2 shows the learning sample data distribution, Figure 3 and Figure 4, respectively, before the output layer neuron distribution topology and neuronal connections between the initial weight, Figure 5 shows the distribution of neurons in the output layer after network training.From the results shown in Figure 2 to Figure 5 shows the learning process of SOMO algorithm, after its network learning is completed, in addition to some edge effects, the output layer neuron weights The statistical properties of the well-mapped mapping input vector can be seen from the basic features in Figure 5. The final feature mapping state reflects the standard normal distribution and the output weight vector is basically concentrated between [-1, 1] The distribution of weight vector is a normal distribution, but there is a certain gap between it and the standard normal distribution. The main reason is that the learning rate η (t) has a great influence on the statistical accuracy in the network learning process. Kohonen's learning rate Is described as follows: initial value of learning rate is η0, decreasing with time t, but never 0; for good statistical accuracy, η (t) should keep a small value for a long time in the algorithm convergence phase (0.01 or less) [. The gap between the real results and the standard normal distribution is a reflection of learning statistical accuracy is not good enough.

SOMO algorithm is a mentor-free clustering method, which can extract important statistical features of samples through self-organized learning of learning samples and classify them according to unsupervised guidance. Randomly generate 4 types of 100 groups of samples, and input of each class Are two-dimensional vector, the output of the network is one-dimensional grid, the output node 4, the network learning rate η (t) = 0.2 (1-t / T), training times T = 800.

T. Kohonen's self-organizing feature mapping neural network, which is built on a one-dimensional or two-dimensional neuron grid, has the competition, cooperation and newer learning rules of the algorithm that can capture the data contained in the input (data) space The characteristics of interest make the adjustment of a complex system from the initial complete chaos to the final overall order.The current SOMO neural network needs further research and discussion in the following three aspects because it can provide the convergence of SOMO self-organizing algorithm Theoretical basis, and to provide any revised form of evaluation criteria, in turn, indicate the correct direction to improve the performance of the algorithm.

When the iteration times T → ∞, whether the network weights converge to an optimal solution or not, will the network converge to a unique state? The research on this problem is currently limited to one-dimensional grids. In the case of one-dimensional grids, Strictly speaking, the SOMF algorithm converges to a unique state "almost certain" for a large class of neighborhood functions after the self-organizing learning ends, and the multidimensional problem has not yet been solved.

How to determine the best learning rate and neighborhood radius for the convergence of the algorithm and the learning rate η (t), the neighborhood radius Nc (t), and the weights? The current research on this problem is also only Limited to one-dimensional grids, there is no solution for multi-dimensional.

Some modifications to the SOMO algorithm, as the SOMO algorithm can not provide a credible representation of the inherent distribution of input data, has prompted the revision of the algorithm and the development of new self-organizing algorithms such as modifying the competition process, during the competition Join to obtain the probability of success; modify the update process, mainly to adjust the neighborhood function of each neuron vector weight update rules to control the nature of the magnification of feature mapping.

4. Two-Dimensional Self-Organizing Feature Mapping Network Application Design

There are 1000 two-dimensional vectors, which are categorized using a two-dimensional self-organizing feature mapping network of 5 × 6 = 30 neurons. Use the Euclidean function to calculate the distance. It is hoped that each neuron responds to a certain area in the rectangle corresponding to each neuron, and the neighboring neurons correspond to the adjacent area. In this paper, MATLAB6.5 to design the above two-dimensional self-organizing feature mapping network. The experimental environment for the CPU P4 2.0A, OS for Windows 2K Sp3. The following is the design process:

Use the rand function to generate 1000 two-dimensional random vectors and use newsom to build the SOM network.

Use the function train to train the established network. Select the training steps were 10,40,70 and 100, the network training. Figures 3, 4, 5, and 6 correspond to the distribution of weights for each neuron with training steps of 10, 40, 70, and 100, respectively.

It can be seen from Fig. 2 to Fig. 6 that the neurons in the competing layer can be distinguished from each other in the input space by the initial overlap of the neurons into a self-organized distribution.

After the network is trained, the weight is fixed. For each input value, the network will output the corresponding classification value. Set a specific point, use the sim function can be classified. Such as p = [0.5; 0.3]; a = sim (net, p) Get the corresponding output in the MATLAB command line as: a = (8,1) 1 NN tool 4.0 provided by MATLAB6.5. 2 added a graphical user interface (GUI), the use of GUI neural network design is very easy.

Is a SOM network diagram of the above problems that were obtained using GUI design.

The following is the initial weight for each neuron prior to training: [0.49979 0.49996; 0.49979 0.49996; 0.49979 0.49996; 0.49979 0.49996; 0.49979 0.49996; 0.49979 0.49996; 0.49979 0.49996; 0.49979 0.49996; 0.49979 0.49996; 0.49979 0.49996; 0.49979 0.49996; 0.49979 0.49996 0.49979 0.49996 0.49979 0.49996 0.49979 0.49996 0.49979 0.49996 0.49979 0.49996 0.49979 0.49996 0.49979 0.49996 0.49979 0.49996 0.49979 0.49996 0.49979 0.49996 0.49979 0.49996 0.49979 0.49996 0.49979 0.49979 0.49979 0.49979 0.49996; 0.49979 0.49996]

The following is the actual value of training weights for each neuron after 100 steps: [0.1626 0.85777; 0.28855 0.85731; 0.51199 0.87433; 0.68688 0.8738; 0.82138 0.83552; 0.20549 0.74558; 0.39332 0.75133; 0.58023 0.75652; 0.76652 0.74176; 0.86817 0.71439; 0.15875 0.57995 ; 0.27287 0.57334; 0.46005 0.57423; 0.64371 0.56542; 0.81437 0.54935; 0.19301 0.4473; 0.3504 0.43486; 0.51924 0.41931; 0.73669 0.40718; 0.85004 0.40707; 0.12488 0.26326; 0.23802 0.25944; 0.42162 0.25226; 0.60458 0.23667; 0.79947 0.23405; 0.16659 0.14827; 0.31161 0.13399; 0.49891 0.13742; 0.72193 0.13209; 0.84697 0.12338]

Use the Simulate command in the GUI and enter the data q = [0.5; 0.3]; The result is: [0; 0; 0; 0;

0; 0; 0; 0; 0; 0; 0; 0; 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0]

Can be seen, SOM network has the characteristics of automatic recognition of clustering features. Through competing learning, the network makes the weight vector more or less deviate from the input vector and finally forms a kind of pattern of the input space each represents. SOM networks can not only learn the distribution of inputs, but also learn the topology of input vectors. SOM networks can classify input vectors based on their distribution in the input space. This method can be used as a non-logical induction, implementation of information extraction feature extraction, and then clustering.

5. Conclusion

In this paper, we mainly study a method of continuous optimization problem using SOMO. By setting the indices of nearness, frequency and value,

References

[1] Dasgupta C G, Dispensa G S, Ghose S. Comparing the predictive performance of a neural network model with some traditional market response models [J]. International Journal of Forecasting, 1994, 10(2): 235-244.

[2] Haemoon Oh. Service quality, customer satisfaction, and customer value: a holistic perspective [J]. InternationalJournal of Hospitality Management, 1999, 18(1): 67-82.

[3] Niren Sirohi, EW Mclaughlin, DR Wittink. A model of consumer perceptions and store loyalty intentions for a su-permarket retailer [J]. Journal of Retailing, 1998, 74(2): 223-245.

[4] Kent Eriksson, A L Vaghult. Purchasing behavior and relationship substance in professional services [J]. Industrial Marketing Management, 2000, 29(4): 362-372.

[5] M Kiely. Word-of-mouth marketing [J]. Marketing, 1993, 9: 6-11.

[6] Mazanec J A. Classifying tourists into market segments: a neural network approach. Journal of Travel& Toruism Marketing, 1992, 1(1): 39-59.