Exploration of Approach to Mining WDMS Spectra based on Laplacian Eigenmap and Neural Network

JIANG Bin[1,a], LI Zi-xuan[2,b] , WANG Wen-yu[3,c] and Qu mei-xia[3,d]

[1]School of Mechanical, Electrical and Information Engineering

Shandong University

Weihai, China

[a]jiangbin@sdu.edu.cn, [b]Lee_zix@163.com, [c]sdwangwenyu@163.com, [d]qumeixia@126.com

Keywords: Laplacian Eigenmap, Data mining, BPNN

Abstract. For the purpose of discovering White Dwarf +Main Sequence (WDMS) from massive spectra, in this paper, an unsupervised learning algorithm for Nonlinear Dimensionality Reduction named Laplacian Eigenmap is discussed. It turns out that, comparing with Principle Component Analysis (PCA), Laplacian Eigenmap maintains the information of nonlinear structure of high dimensional spectral data, which leads to a higher classification accuracy. In the feature space, backpropagation neural network is used to classify WDMS and non-WDMS spectra. Furthermore, Particle Swarm Optimization (PSO) is implemented to increase the classification accuracy via optimizing the parameters of the network. The results shows that the method in this paper can discover WDMS efficiently and accurately after training the neural network with low-dimensional data from Sloan Digital Sky Survey Data Release 10 (SDSS-DR10).

Introduction

With the rapid development in data science, the concept of big data attracts increasingly more attention. As the limitations of traditional algorithm used in data analysis arise, how to apply machine learning algorithm to big data has become a popular topic. Sloan Digital Sky Survey (SDSS), started in 2000, is a large redshift survey project, using a 2.5 m diameter telescope which is sited in Apache Peak Observatory in New Mexico to observe, recording nearly two millions spectral data, which include more than 80 million galaxies and more than 10 million quasar spectra data. WDMS is a very special binaries system, which is the progenitor star of *Ia* supernovae and cataclysmic variable star and is worthy of being studied. However, template matching methods based on the physical parameters, which generally were used to classify the spectrum, bring about great artificial intervention. Since astronomical spectra belong to high-dimensional data, how to find its structural features from the high dimensional data and furtherly use appropriate algorithms to reduce data dimensionality becomes a key problem in data preprocessing of the machine learning algorithm. Previously, Tan Dong-mei [1] used Principle Component Analysis (PCA) to classify stellar spectra rapidly. Connolly et al. [2] used PCA to extract the feature of known redshift galaxy spectra, discovering that some former main component spectra of galaxies have strong linear relation. Madgwick et al. [3] used PCA to classify emission line and absorption line spectra. However, although PCA can reconstruct linear-independent components from the data, it still uses the Euclidean distance to measure the sample space in essence. But the high dimensional spectral data have a strong non-linear structure so that the dimensionality reduced by PCA cannot accurately describe the distance between samples. In this paper, Laplacian Eigenmap is used to analysis and process WDMS spectral data and reconstruct them in a low dimension space, then backpropagation neural network is used to classify the date. It turns out that Laplacian Eigenmap performs better than PCA. Furthermore, the initial weights and threshold value of neural network have a great impact on the results. Using Particle Swarm Optimization to optimize the parameters of the neural network, the result shows that the accuracy of BP neural network reaches 88.97%, greatly improving the accuracy of discovering WDMS.

The rest of this paper is organized as follows. Section 1 introduces the fundamental idea and implementation tricks about Laplacian Eigenmap. Then demonstrates how to optimize controlling arguments of backpropagation neural network (BPNN) using particle swarm optimization (PSO) algorithm. Experimental results and analysis are presented in Section 2. Finally, conclusion and the future extension of the model are drawn at last.

Section 1. Introduction to the algorithm

Laplacian Eigenmap. Laplacian Eigenmap algorithm[4] is widely used for Dimensionality Reduction, especially for non-linear dimensionality reduction. The process to reconstruct the submanifold can be stated as follows:

(1) Construct a graph for the dataset, each node in the graph instead for a point in our data, there are two variations to connect two nodes if they are close:

a. ε - neighborhoods: calculate the Euclidean distance between node i and node j, two nodes are connected only if the distance below ε, $\varepsilon \in \Re$.

b. n -nearest neighbors: calculate the Euclidean distance between node i and node j, two nodes are connected if i is among the nearest neighbors of j and i is among the nearest neighbors of j.

(2) Weighted each edges in the graph. Here we use the heat kernel to calculate the weight. If node i and j are connected, put

$$W_{ij} = e^{-\frac{\|x_i - x_j\|^2}{t}} \tag{1}$$

(3) Assume the graph constructed above, is connected. Otherwise, proceed with step 3 for each connected component. Compute eigenvalues an eigenvectors for the generalized eigenvector problem:

$$L\vec{f} = \lambda D\vec{f} \tag{2}$$

Where D is diagonal weight matrix, and its entries are column (or row, since W is symmetric) sums of W, $D_{ii} = \sum_j W_{ji}$. L = D – W is the Laplacian matrix. Let $\vec{f}_0, \vec{f}_1 ... \vec{f}_{k-1}$ be the solutions of Eq. 2, ordered according to their eigenvalues:

$$\vec{x}_i \to (\vec{f}_1(i), \vec{f}_2(i) ... \vec{f}_m(i)) \tag{3}$$

Particle Swarm Optimization. Particle Swarm Optimization [5] searches the global optimal solution via the cooperation and competition among the particle, inspired by social behavior of bird flocking or fish schooling. In PSO, the potential solutions, called particles, have a fitness which depends on the situation that the algorithm is applied. The particles fly through the problem space by following the current optimal particles it has achieved so far. This value is called *pbest*. When a particle takes all the population as its topological neighbors, the best value is globally optimal, called *gbest*. When it comes to the optimization of the BPNN, the particle's position is decided by the initial weights and thresholds in the backpropagation network. Using the error of prediction as the fitness of the particle, the algorithmic procedure is stated as follows:

(1) Generate 30 particles randomly as the initial distribution. Use the vector $x^{(i)} = (w,b), i = 1,2...n$ to represent the location of the particle. The parameters values are in the range of [-5, 5].

(2) Initialize the velocity of each particle as vector $v^{(i)} = (v_{w_1}{}^{(i)}...v_{w_k}{}^{(i)}, v_{b_1}{}^{(i)}...v_{b_p}{}^{(i)})$. In order to limit the range of the velocity, $v_{\max} = \alpha x_{\max}, v_{\min} = -v_{\max}$, where α is the constraint of the velocity .A valid α avoid the particle to be unlimited divergent and we choose α as 0.5.

(3) Regard the location of each particle as the control parameter of the backpropagation neural network. A total of 560 WDMS spectra and 2000 random spectra are selected as our data. Use 5-fold cross validation to estimate the solution.

(4) Begin the iterative calculation. Get the fitness of each particle via each iteration calculation. Finally, calculate the optimal fitness of the population. Take down the corresponding local optimal position of the fitness and the global optimal position of the population. In the end, update the velocity and the position of each particle. The equation is listed below:

$$v^{(i)} = \omega v^{(i)} + c_1 r_1 (p_{local}{}^{(i)} - x^{(i)}) + c_2 r_2 (p_{global} - x^{(i)}) \tag{4}$$

$$x^{(i)} = x^{(i)} + \eta v^{(i)} \tag{5}$$

ω is called inertia weight, r_1 , r_2 are the random numbers in the range of [0,1], called constriction factor, used to constraint the velocity. α is the constraint of the velocity. Calculate the fitness of each particle again with the equation above to update the local optimal position and the global optimal position.

(5) Self-adaptive mutation. Randomly changing the position of a particle among the popular, improves the possibility to find the global optimal value.

(6) End the argorithm if the global optimal value below the target value or the number of iteration exceeds the given bound. Or turn to the 4th step.

Backpropagation Neural Network. Backpropagation neural network trains the neural network using the back propagation algorithm, which leads to the network with a good capability of generalization. There are three layers of the network named input layer, hidden layer and output layer. The network spreads in two step:

(1) The sample is entered from the input layer, going through the hidden layer, reaching the output layer. The state of current layer only has impact on the next layer. Compare the output in the output layer with the expected output. If the current output is not expected, the step2 will be carried out.

(2) The residual error backpropagates, according to the forward propagation. At the same time, update the weights of each hide layer node with the purpose of decreasing the error. The neural network topologies is shown as fig 1.

The process of training can be stated as follows:

(1) Initialize the neural network. Desire the structure of the network, which includes the number of the nodes in the input layer, the number of the hidden layer and the number of the output layer. The weight and the threshold between the nodes. Besides, the learning rate and the activation function are given.

(2) Compute the output of the hidden layer, H, according to the input X.

(3) Compute the error according to the predicted output of the network and the given output.

(4) Update the weight and the threshold.

(5) Step 2 is carried out if the iterations is not enough.

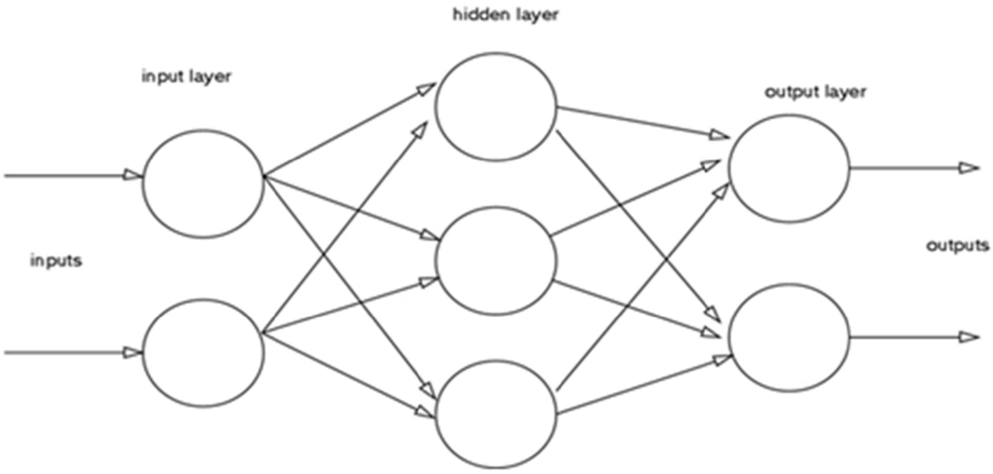

Fig 1 Neural network topologies

Section 2. Experimental result

The experimental data. The experiment of this paper is carried out on SDSS-DR10. A total of 1140 WDMS spectra and 6000 random spectra are selected as our data. We randomly choose 75% of them as our training set and 25% as testing set.

(1) Scale spectral data into the range (0, 1) using the equation below as a preprocessing step.

$$\mathrm{x}_i = \mathrm{x}_i \Bigg/ \sqrt{\sum_{j=1}^{M} x_j^2} \tag{6}$$

(2) Reconstruct spectral data into 3-D space using Laplacian Eigenmap and PCA separately.

(3) Input low dimensional data into BPNN and run 5-fold cross validation to estimate the model.

The experimental results and analysis. We use Laplacian Eigenmap and PCA separately to reduce the dimension of the spectra and visualize them in Fig.2 and Fig.3.

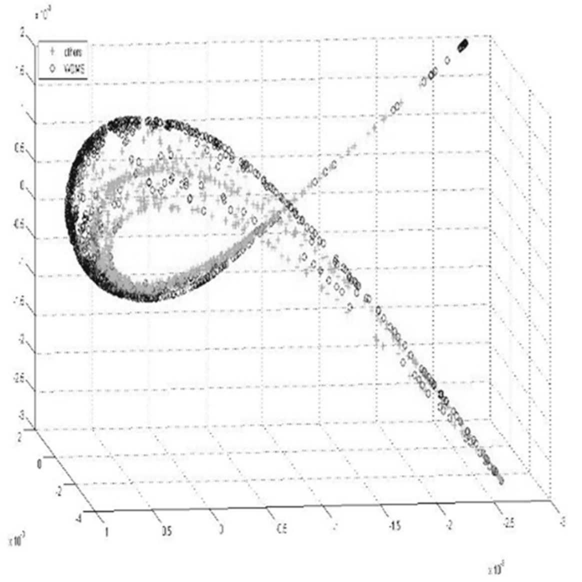

Fig 2 Three-dimensional projection by Laplacian Eigenmap

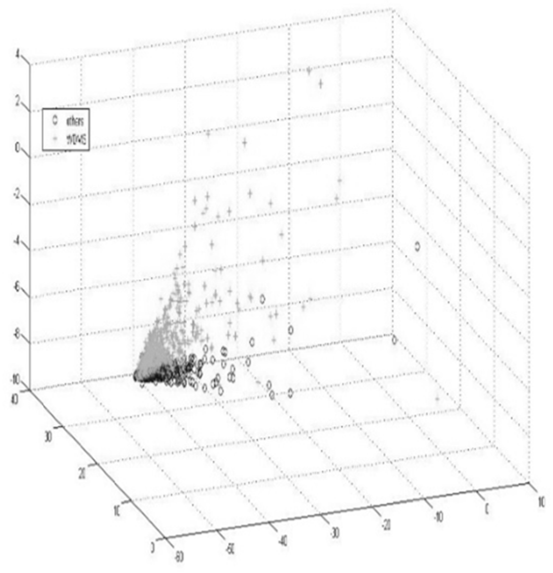

Fig 3 Three -dimensional projection by PCA

It turns out that the overlapping rate of Laplacian Eigenmap is smaller than PCA, which has an effect on the accuracy of the classification. As shown in Fig 3, the accuracy of BPNN reaches the peak when the input data is reconstructed to 20 dimension. And in every dimension, Laplacian Eigenmap performs better than PCA, the accuracy reaches 88.97% in the 20 dimension with Laplacian Eigenmap. Table 1 shows that Particle Swarm Optimization (PSO) performs better than Grid Search(GS). And Fig.4 shows that the PSO can search for the optimal initial parameters and reduce the classification error obviously.

Table 1 Experiment Result

	PSO-BP		GS-BP	
ALGORITM	[LE]	[PCA]	[LE]	[PCA]
ACCURAY (%)	[88.97]	[84.25]	[86.23]	[84.25]

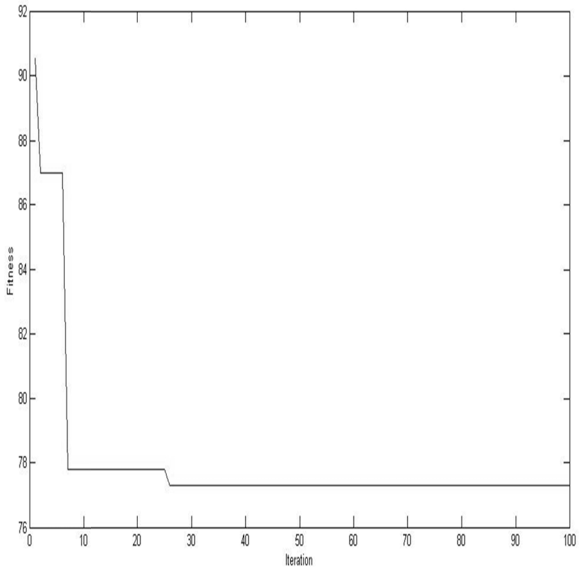

Fig 4 Fitness curve of PSO

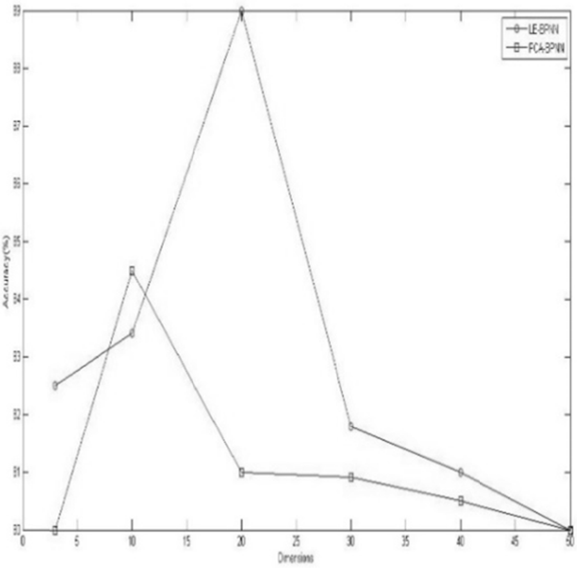

Fig 5 Accuracy of LE-BPNN and PCA

Summary

(1) The initial parameters of the Backpropagation neural network have effect on the accuracy of the classification directly. PSO algorithm can determine the optimal initial parameters, which can improve the performance of the classification and reduce the time.

(2) The manifold learning algorithm, Laplacian Eigenmap, can keep the geodesic distance between different samples, which has a good dimension reduction effect in our high dimension, nonlinearity spectral data.

(3) The approach proposed in this paper, Laplacian Eigenmap and Backpropagation neural network, can be efficiently used to discover new WDMS automatically.

Acknowledgment

This work is supported by National Natural Science Foundation of China #11473019, Natural Science Foundation of Shandong #ZR2014AM015 and Basal Research Fund of Shandong University #2015ZZXM002 .

References

[1] Tan Dong-mei,Hu Zhan-yi,Zhao Yong-heng. Spectroscopy and Spectral Analysis. (2003)

[2] Connolly A J, Szalay A S, Bershady M A, et al. Astron. J. , (1995), 110:1071.

[3] Madgwick D S, Coil A L, et al. ApJ, (2003), 599:997

[4] Mikhail Belkin, Partha Niyogi.Neural Computation. (2004)

[5] Kennedy, J, Eberhart, R. Proceedings of IEEE International Conference on Neural Networks. (1995)