Protein Secondary Structure Prediction Using AutoEncoder Network and Bayes Classifier

Leilei Wang, Jinyong Cheng*
College of Information, Qilu University of Technology(Shandong Academy of Sciences), Jinan, China

*Corresponding author e-mail: cjq@qlu.edu.cn

Abstract. Protein secondary structure prediction is belong to bioinformatics, and it's important in research area. In this paper, we propose a new prediction way of protein using bayes classifier and autoencoder network. Our experiments show some algorithms including the construction of the model, the classification of parameters and so on. The data set is a typical CB513 data set for protein. In terms of accuracy, the method is the cross validation based on the 3-fold. Then we can get the Q3 accuracy. Paper results illustrate that the autoencoder network improved the prediction accuracy of protein secondary structure.

1. Introduction
The human genome project is completed, the attention of people from the analysis of genome expression, turned to the prediction and function of protein structure. Our research about protein structure, it is helpful to study the function of protein. Protein space fold structure depends on the amino acid sequence that is used to build the protein[1], in other words, all the information in the space of the protein folding structure are hidden in the amino acid sequence. In the past few years, have many methods that the prediction of protein secondary structure, such as Artificial Neural Network[2], Bayes Classifier, K-nearest Neighbor, Support Vector Machine[3], hidden markov model (HMM)[4], maximum entropy model[5], and so on.

AutoEncoder is a kind of neural network that have a number of layers. It can pass the results forward and be used for making the high-dimensional data mapped to the low-dimensional, and get the low-dimensional feature vector. In recent years, AutoEncoder[6] that is represented by deep learning[7] has made breakthrough progress in the research for machine learning. The development of the AutoEncoder is mainly used for deep learning model’s initialization[8]. In order to prevent the AutoEncoder had learned into its own mapping, some regularization method about AutoEncoder is proposed, including sparse AutoEncoder[9], denoising AutoEncoder[10] and the decomposition of predicted coefficient[11].

2. AutoEncoder and Bayes
AutoEncoder is a structure of deep learning neural network. It can extract hierarchical characteristics of the input data from the data without a label and get the distributed characteristics of the original data, this process through a multistory and nonlinear networks which including supervised learning and non-supervised finely adjust. The AutoEncoder is composed of encoder, decoder and hidden layer. As shown in figure 1.
AutoEncoder is the mapping of input $x$ to implicit representation $h$, it is defined as follows:

$$h = f(x) = S_f(W + b_x)$$  \hspace{1cm} (1)

Among them, the $S_f$ is no-linear activation function, it is universal logic function, the expression is:

$$sigmoid(z) = \frac{1}{1 + e^{-z}}$$  \hspace{1cm} (2)

Decoder function $g(h)$ made hidden layer’s data map to reconstitution $y$, the expression is:

$$y = g(h) = S_g(W'h + by)$$  \hspace{1cm} (3)

In the expression of (3), $S_g$ is the activation function of decoder. It is usually linear function or sigmoid function. The process of training the AutoEncoder is find minimize the reconstruction error of parameter $\theta$ in the training set $D$. The reconstruction error expressed as:

$$J_{ae} = \sum_{x,y} L(x,g(f(x)))$$  \hspace{1cm} (4)

$L$ is the function of reconstruction error, it is used to Square-Error or the cross-entropy, respectively expressed as:

$$L(x,y) = \sum_{i=1}^{d} x_i \log y_i + (1 - x_i) \log (1 - y_i)$$  \hspace{1cm} (5)

The Square-Error function is used to linear $S_g$ and the cross-entropy cost function is used to sigmoid.

Bayes classifier is through the prior probability of class nodes and conditional probability of property to figure out the posterior probability of class node, and can be divided into different kinds of predicted object by setting threshold of posterior probability.

3. Protein database

It is important that choosing the appropriate training data set for protein secondary structure prediction. Now people use a number of training data sets, and we used the 25 PDB and CB513’s training and testing sample data set. The CB513 data set is from the Cuff and Barton, it has 513 protein sequences, the similarity between them is less than 25%. First of all, we have to process the data set. That is remove the sequence that length is less than 30 and contains the character of X, Z and B. The quantity of the aminophenol is 84119. When we calculate the accuracy, the method is the cross validation based on the 3-fold.

On the basis of the amino acid sequence, we can get the PSSM matrix by the PSI - BLAST program and NR database. Then we predict protein secondary structure by the PSSM method. Input sample is 84119 amino acids in the experiment, the PSSM matrix with 13 as the sliding window.
coding to get 260 characteristics of each amino acid. In this way, we can get a PSSM matrix, it’s number of rows is 260 and columns is 84119.

4. Experiment and results
First, we processed the CB513 data set, the amino acid sequence’s length is 84119 in the CB513 data set. We processed the data set used PSI-BLAST with 13 as the sliding window. In this way, we can get a PSSM matrix, it’s number of rows is 260 and columns is 84119. Then we divided the data into training set and testing set. Finally we used AutoEncoder and Bayes classifier to classify and predict the data. Figure 2 shows our proposed method.

![Method](image)

**Fig.2 Method**

In the first experiment of this paper, the value of the sliding window is set to 13, because it is a good empirical value. Setting the hidden layer of the autoencoder, we found that the prediction is best when the hidden layers of the autoencoder is 400. To avoid over-fitting, we use the L2 regularization term to adjust the weights. After several experiments, we found that after we tried some different values of the L2 regularization term to 0.04, we can get the best prediction. In the autoencoder, the range of the sparsity proportion is from 0 to 1, so we compared the results of the sparsity proportion from 0.1 to 0.9, and we found that when the sparsity proportion is 0.3, the prediction is best. Another parameter of sparsity is sparsity regularization, when the value of the sparsity regularization is 4, we can get the best prediction.

In order to evaluate predict of accuracy, we are using Q3 accuracy.

Only use Bayes classifier to predict training set and testing set, we get the Q3 accuracy are in the table1 that:

|      | train | 2   | 3   | average |
|------|-------|-----|-----|---------|
| train| 72.81 | 71.81 | 71.79 | 72.13   |
| testing| 70.52 | 71.96 | 72.14 | 71.54   |

Using Bayes classifier and AutoEncoder to extract the feature and predict training set and testing set, we get the Q3 accuracy are in the table2:

| data | 1    | 2    | 3    | average |
|------|------|------|------|---------|
| train | 73.17 | 72.33 | 72.21 | 72.57   |
| testing| 70.59 | 72.28 | 72.23 | 71.70   |

We used Bayes classifier and AutoEncoder to extract the feature and predict training set and testing set when the windows are different. We get the Q3 accuracy are in the table3:

| data | 9    | 11   | 13   | 15   | 17   | 19   |
|------|------|------|------|------|------|------|
| train | 72.12 | 72.24 | 72.57 | 72.55 | 72.63 | 72.69 |
| testing| 71.29 | 71.48 | 71.70 | 71.74 | 72.00 | 71.98 |
From the above-mentioned results, we can find that only using the Bayes classifier to classify and predict the data, we can get $Q_3$ accuracy 71.54%, and using Bayes classifier and AutoEncoder to extract the feature and predict training set and testing set, we can get the $Q_3$ accuracy 71.70%. We have chosen six numbers as a sliding window to process PSSM data, the five numbers are 9, 11, 13, 15, 17, 19. So we get six corresponding PSSM data. We found that when the value of the sliding window is set to 17, we can get the best prediction. The best prediction result is 72.00%. AutoEncoder can adjust the parameters of each layer to get the weight of each layer, so it can extract the most important feature that represent the input data, it also is a kind of neural network that reproduce entered signals, so the result increased by 0.46%. But AutoEncoder doesn’t have global optimization, the reconstruction of the input may not be learning general characterization of the ideal measure, so the improvement of accuracy is not obvious.

5. Conclusion
It is successful that we predicted protein secondary structure using AutoEncoder network and Bayes classifier. The input data is encoded when the input through the hidden layer of AutoEncoder, and next it is decoded when the code arrived at the output layer. At this time, we must ensure it is impossible that the output results consistent with the input data. In other words, the data-out is is the data-in to be reproduced, so the result was improved. But it is ideal that data-out is similar to the data-in in AutoEncoder. The truth is that there is error in each layer of input and each layer of output. The error is got by compare the data of reconstituted with input data. Therefore, it is not obvious that the accuracy was improved.

Acknowledgment
The research work is supported by the National Natural Science Foundation of China (Grant No. 61375013), and Natural Science Foundation of Shandong Province (ZR2013FM020), China.

References
[1] PRIBYLO, GOULIAS K G. Simulation of daily activity patterns incorporating interactions within households: algorithm overview and performance[J]. Journal of Trans on Research Board, 2005, 1926 (1) : 135 2 141.
[2] KAUFMAN L, ROUSSEEUW P J. Finding groups in data: an introduction to cluster analysis[M]. New York: Wiley, 1990.
[3] WANG H, XUE D. An intelligent zone 2 based delivery scheduling approach[J]. Computers in Industry, 2002, 48 (2) : 109 2 125.
[4] HUANG Zhexue. Clustering large data sets with mixed numeric and categorical values[C]//Proc of the 1st Pacific-Asia Conference on Knowledge Discovery and Data Mining. 1997: 283 2 304.
[5] MAULIKU, BANDYOPADHYAY S. Genetic algorithm 2 based clustering technique[J]. Pattern Recognition, 2000,33 (9) : 1883 2 1893.
[6] HintonG, Osinder S, Teh Y M. A fast learning algorithm for deep belief nets[J]. Neural computation, 2006, 18(7):1527-1554.
[7] Bengio Y. Learning deep architectures for AI[J]. Foundations and trends in Machine Learning, 2009, 2(1):1-127.
[8] Bengio Y, Courville A C, Vincent P. Unsupervised feature learning and deep learning: A review and new perspectives[J]. CoRR, abs/1206.5538, 2012,1.
[9] Bengio Y, Lamblin P, Popovici D, et al. Greedy layer-wise training of deep networks[J]. Advances in neural information processing systems, 2007, 19:153.
[10] Vincent P, Larochelle H, Bengio Y, et al. Extracting and composing robust features with denoising autoencoders[C]//Proceedings of the 25th international conference on Machine learning. ACM, 2008: 1096-1103.
[11] Kavukcuoglu K, Ranzato M A, LeCun Y. Fast inference in sparse coding algorithms with applications to object recognition[J]. arXiv preprint arXiv: 1010.3467, 2010.