Evolutionary Computation for Statistical Learning Theory

*Sung-Hae Jun, **Mitsuo Gen

*Dept. of Statistics, Cheongju University

Chungbuk (360-740) Korea

email:shjun@cju.ac.kr

**Grad. School of Inform., Prod. & Sys., Waseda University

Kitakyushu (808-0135) Japan

email:gen@waseda.jp

***Abstract* – In the intelligent systems including statistical analysis, the learning and evolving have been studied as the approaches of knowledge discovery in given data sets. As compared with classical learning theory based on objective function with minimizing training errors, the recent evolving approaches have played an important role for constructing optimal model without the minimizing training errors. These made an attempt to settle the local minimum problem of learning approach. But the evolving based models also had the local minimum problem in complex data sets. In our research, using competitive co-evolving based on host and parasites of the natural world, we proposed a competitive co-evolving model of statistical learning theory for overcoming local minimum problems. The proposed model was compared with classical learning and recent evolving models using the given data from UCI machine learning repository. Our experimental results showed the improved performance of proposed model.**

I. INTRODUCTION

The learning and evolving have been studied on the intelligent system of computer and information science[8][9]. The methods of evolutionary artificial neural networks have been popular researches as combining learning and evolution[18][19]. The goal is to find optimal neural network structure using genetic algorithm[3]. Some works shown that the evolving of artificial neural networks had more chance to find global optima than the learning of artificial neural networks[3][18][19]. But the possibility of local optima still existed in the evolutionary artificial neural networks using genetic algorithm[9].

To solve local optima problem of artificial neural networks using genetic algorithm, Hills proposed a host-parasites co-evolution approach from biology[6]. The model of host-parasites co-evolution is the organism that evolves defenses to parasites from their attacks. With the parasites evolves ways to circumvent the defense, the hosts' evolving new defense. Ultimately, the increase of fitness of the hosts can be expected. So the approach of host-parasites co-evolution can avoid local optima. The good fitted structures were found in our research.

In this paper, we use the competitive co-evolution as advanced evolving computing to construct evolutionary computation for statistical learning model (SLT). The traditional SLT model was fitted using Lagrange multipliers[15]. But our model was optimized by the competitive co-evolutionary computation. The Lagrange multipliers were not used in our

SLT. Using the data sets of UCI machine learning repository and KDD cup 2000, we verified our proposed model compared to the existing models.

II. RELATED WORKS

Our world is changing too fast for us to keep up with based only on our logics. Due to digitalization, amount of data is increasing very fast. Most of information in huge data remains undiscovered. So we need tools for discovering knowledge in data. One of these tools is Statistics. Statistics is the art of learning from data[11]. The learning is to construct a claim by observing data. The learning procedure contains from this till performing experiments and making conclusion. SLT developed by Vapnik[15][16]. SLT is perhaps the best currently available theory for finite sample statistical estimation and predictive learning[1]. It has three types which are support vector machine(SVM), support vector regression(SVR), and support vector clustering(SVC). SVM, SVR, and SVC are respectively classification, prediction, and clustering tools [16][17]. All types of SLT are based on support vector. The approaches of support vector are projection instances into high dimensional spaces, learning linear separators with maximum margin, and learning as optimizing upper bound on expected error. The classification problem of SLT can be restricted to consideration of the two-class problem. In this problem the goal is to separate the two classes by a function which is induced from available examples. The goal is to produce a classifier that will work well on unseen examples, that is, it generalizes well. Consider the problem of separating the set of training vectors belonging to two separate classes,

$$D = \{(x_1, y_1), \cdots, (x_l, y_l)\}, \quad x \in R^n, \quad y \in \{-1, 1\} \quad (1)$$

with the hyperplane,

$$\langle w, x \rangle + b = 0 \quad (2)$$

The set of vectors is said to be optimally separated by the hyperplane if it is separated without error and the distance between the closest vector to the hyperplane is maximal. There is some redundancy in (2), and without loss of generality it is appropriate to consider a canonical hyperplane[15]. Where the parameters w, b are constrained by,

$$\min_i \left| \langle w, x_i \rangle + b \right| = 1 \quad (3)$$

This incisive constraint on the parameterization is preferable to alternatives in simplifying the formulation of the problem. In words it states that: the norm of the weight vector should be equal to the inverse of the distance, of the nearest point in the data set to the hyperplane. A separating hyperplane in canonical form must satisfy the following constraints,

$$y_i \left[\langle w, x_i \rangle + b \right] \geq 1, \quad i = 1, \ldots, l \quad (4)$$

The distance d(w,b; x) of a point x from the hyperplane (w,b) is,

$$d(w, b; x) = \frac{\left| \langle w, x_i \rangle + b \right|}{\|w\|} \quad (5)$$

The optimal hyperplane is given by maximizing the margin, subject to the constraints of (4). This approach of SVM is used for the prediction model, SVR.

III. COMPETITIVE CO-EVOLVING FOR STATISTICAL LEARNING THEORY

A. Support vector regression

In this paper, we applied competitive co-evolving to SVR. In SVR, our given training data consist of N pairs $(x_1, y_1), \ldots, (x_N, y_N)$, where x denotes the input patterns and y is target variable. In SVR with ε-insensitive loss function, our goal is to find a function f(x) that has at most ε-deviation from the actually obtained targets y_i for all the training data, and at the same time, is as flat as possible[13]. In other words, we do not care about errors as long as they are less than ε, but will not accept any deviation larger than this. The ε- insensitive loss function is defined as,

$$M(y, f(x,\alpha)) = L(| y - f(x,\alpha) |_\varepsilon) \quad (6)$$

where we denote,

$$| y - f(x,\alpha) |_\varepsilon = \begin{cases} 0, & if \ | y - f(x,\alpha) | \leq \varepsilon, \\ | y - f(x,\alpha) | - \varepsilon, & o.w. \end{cases} \quad (7)$$

and α is a positive constant. The loss is equal to 0 if the discrepancy between the predicted and the observed values is less than ε. The case of linear function f is described.

$$f(x) = \langle w, x \rangle + b \quad (8)$$

where $\langle \cdot, \cdot \rangle$ denotes the dot product. For SVR, the Euclidean norm $\| w \|^2$ is minimized. Formally this problem can be written as a convex optimization problem by requiring[15], Analogously to the loss function in [2], we introduce slack variables ξ_i, ξ_i^* to copy with otherwise infeasible constraints of the optimization problem.

$$\text{minimize } \tfrac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*) \tag{9}$$

$$\text{subject to } \begin{cases} y_i - \langle w, x_i \rangle - b \le \varepsilon + \xi_i \\ \langle w, x_i \rangle + b - y_i \le \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \ge 0 \end{cases} \tag{10}$$

The constant C(>0) determines the trade off between the flatness of f and the amount up to which deviations larger than ε are tolerated. Using a standard dualization method utilizing Lagrange multipliers, the parameters are determined from equation (9) and (10)[4].

B. Competitive co-evolving

Genetic algorithm(GA) provides a learning method motivated by an analogy to biological evolution[8]. General GA computes the fitness of given environment where is fixed. On the other side, the co-evolving On the other side, co-evolving is evolutionary mechanism of the natural world. The organism and the environment with it evolve together[9]. Our competitive co-evolving uses host-parasites co-evolutionary approach. The host and parasite are used for modeling SLT and training data set. The evolving of SLT follows the evolving of host. The initial parameters for SLT model were determined as uniform random number from -1 to 1. The genetic presentation of weights is shown by the type which is presented from target variable to input variables. The fitness function of SLT model is the inverse form of the squared error between real value and predict value as following.

$$f(x) = \frac{10}{\sum_{i=1}^{F}\sum_{j=1}^{N_{out}}(o_{ij}(x) - t_{ij})^2} \tag{11}$$

In above equation, t is the known value of target variable and o is computed output value for prediction.

Next, the training of given data set is the evolving of parasite. The evolving of training data is performed to retain larger training errors. So the fitness function of training of given data is inverse of the fitness function of SLT learning as following.

$$f(x) = \sum_{i=1}^{D}\sum_{j=1}^{N_{out}}(o_{ij}(x) - t_{ij})^2 \tag{12}$$

These evolving approaches of SLT model and training data is competitive.

C. The process of proposed model

Our proposed model was 2 groups evolving. One was the parasite evolving of given training data set. Another was the host evolving of SLT model.

Following figure showed the process of proposed model.

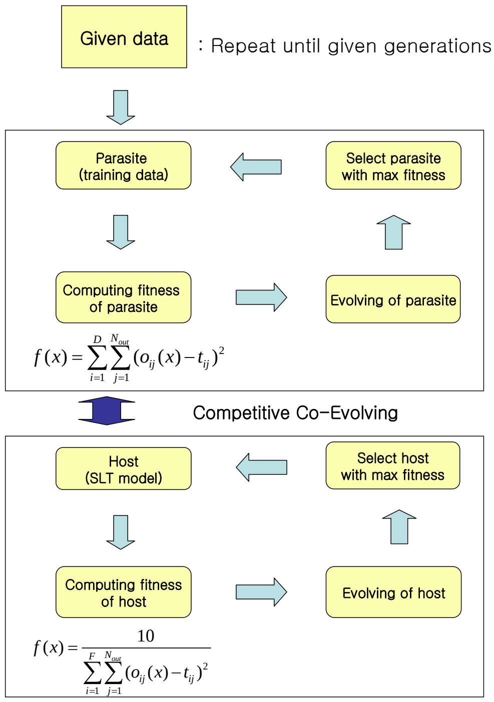

Fig. 1. Competitive co-evolving process of SLT

The SLT and training data were respectively evolved. During evolving for its weight optimization, the competitive evolving was occurred between SLT evolving and training data evolving. In above figure we did not show the Lagrange multipliers. Our model used competitive co-evolutionary computation instead of Lagrange multipliers.

IV. EEPERIMENTAL RESULTS

In this section, we want to show the experimental results of proposed model by abalone data set from UCI machine learning repository and the web log data from KDD cup 2000 data set[7][14]. In abalone data set, the number of instances of data is 4177. The 8 attributes, which are length, diameter, height, whole weight, shucked weight, viscera weight, shell weight, and rings, are abalone's physical state. Using this data set, we made data preprocessing experiment for data mining. Among data preprocessing methods, this experiment performed missing value imputation. The abalone data set is complete. For our experiments, we make complete abalone data to incomplete. The incomplete abalone data have 5%, 10%, 20%, and 30% missing ratios. Currently, since the tree imputations have been good preprocessing methods of missing data, we compared our model with general SLT, artificial neural networks, genetic neural networks and statistical regression model. The 2/3 of given data set used for training and 1/3 of given data set used for testing[8]. In our experiment, the evaluation measure of performance used mean squared error(MSE) as following.

$$MSE = \frac{1}{n}\sum_{j=1}^{n}(y_j - y_j^*)^2 \qquad (13)$$

where y_j represents the jth known value of target variable, y_j^* represents the jth predicted value of output, and n is the size of data set. The smaller the value of MSE is, the better the performance of model is. Following table shown the experimental result. In the table, CE-SLT was our competitive co-evolutionary computation for SLT. ANN, G-NN, SLT, and Regression were respectively artificial neural networks, genetic neural networks, traditional SLT, and statistical regression models[10][12].

Table 1. Result by abalone data set

Models	MSE			
	5%	10%	20%	30%
CE-SLT	0.39	0.42	0.48	0.59
ANN	0.56	0.63	0.69	0.73
G-NN	0.43	0.45	0.55	0.68
SLT	0.51	0.59	0.63	0.69
Regression	0.95	1.21	1.69	2.58

In above table, we knew that the MSEs of all models were increased as the missing ratio increased. With compared other models, the values of MSE of CE-SLT were smaller. So our model had a good performance.

Next, for another experiment, we used the KDD Cup 2000 data. The data set had web log file of real internet shopping mall(gazelle.com). The capacity of given data was 1.2GB. With similar to previous experiment, we used the one-third of given data for the testing and the other two-thirds for training. After data cleaning, we showed the basic information of given data in the following table.

Table 2. Summary of KDD Cup data set

Attributes	Value range
cookie-id	13,109 (users)
assortment-id	269 (web pages)
duration-time	0~1000 (second(s))

In table 2, the cookie-id was the index of user accessing to web site. The assortment-id represented each web page containing the descriptive contents of each item in the shopping mall and the duration-time of web page had the value between 0 and 1000 seconds. This data had originally some missing values. So we used KDD cup data as they were. We can show the result in the following table.

Table 3. Result by KDD Cup data set

Models	MSE
CE-SLT	4.56
ANN	6.11
G-NN	5.32
SLT	5.98
Regression	10.43

From above result, we also knew the CE-SLT was the better than others.

V. CONCLUSION AND FUTURE WORK

To settle the local problem of learning models, we used the competitive co-evolving. In our research, we selected SLT as applied learning model. We applied competitive co-evolutionary computation to SLT. So our SLT did not used Lagrange multipliers. Using data sets from UCI machine learning repository and KDD cup 2000, our model was verified. In the experimental results, our model was the better than other learning and evolving models.

For the future works, we will apply competitive co-evolving to other SLTs as SVM and SVC.

REFERENCES

[1] V. Cherkassky, F. Mulier. *Learning From Data Concepts, Theory, and Methods*, John Wiley & Sons, 1998.

[2] C. Cortes, V. Vapnik. Support Vector Networks, *Machine Learning*, vol. 20, 273-297, 1995.

[3] D.J. Montana, L. Davis. Training Feedforward Neural Network Using Genetic Algorithm, Proceeding of the International Conference on Artificial Intelligence, pp. 763-767, 1989.

[4] R. Fletcher. *Practical Methods of Optimization*, John Wiley & Sons, 1989.

[5] J. Han, M. Kamber. *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers, 2001.

[6] W.D. Hill. Co-evolving parasites improve simulated evolution as an optimization procedure, Physica, vol. D 42, pp. 228-234, 1992.

[7] KDD Cup 2000, http://www.ecn.purdue.edu /KDDCUP

[8] T.M. Mitchell. *Machine Learning*, WCB McGraw Hill, 1997.

[9] T.M. Mitchell. *An introduction to Genetic Algorithms*, MIT Press, 1998.

[10] A.S. Pandya, R.B. Macy. *Pattern Recognition with Neural Networks in C++*, CRC Press, 1995.

[11] S.H. Ross. *Introductory Statistics*, McGraw-Hill, 1996.

[12] L. Rreiman, J.H. Friedman, R.A. Olshen, C.J. Stone. *Classification and Regression Trees*, Wadsworth International Group, 1984.

[13] A.J. Smola. *Regression estimation with support vector learning machines*, Master's thesis, Technische University, 1996.

[14] UCI Machine Learning Repository, http://www.ics.uci.edu/~mlearn

[15] V.N. Vapnik. *The Nature of Statistical Learning Theory*, Springer, 1995.

[16] V.N. Vapnik. *Statistical Learning Theory*, John Wiley & Sons, 1998.

[17] V.N. Vapnik, S. Golowich, A. Smola. Support vector method for function approximation, regression estimation, and signal processing, *Advanced in Neural Information Processing Systems*, Cambridge, MA, 1997.

[18] X, Yao. A review of evolutionary artificial neural networks, *International Journal Intelligent System*, vol. 8, no. 4, pp. 539-567, 1993.

[19] X. Yao. Evolving Artificial Neural Networks, *Proceeding of the IEEE*, vol. 87, issue 9, pp. 1423-1447, 1999.