World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:1, No:12, 2007

An Evolutionary Statistical Learning Theory

Sung-Hae Jun and Kyung-Whan Oh

***Abstract*—**Statistical learning theory was developed by Vapnik. It is a learning theory based on Vapnik-Chervonenkis dimension. It also has been used in learning models as good analytical tools. In general, a learning theory has had several problems. Some of them are local optima and over-fitting problems. As well, statistical learning theory has same problems because the kernel type, kernel parameters, and regularization constant C are determined subjectively by the art of researchers. So, we propose an evolutionary statistical learning theory to settle the problems of original statistical learning theory. Combining evolutionary computing into statistical learning theory, our theory is constructed. We verify improved performances of an evolutionary statistical learning theory using data sets from KDD cup.

***Keywords*—**Evolutionary computing, Local optima, Over-fitting, Statistical learning theory

I. Introduction

LEARNING and evolving have been used for the analytical tools of intelligent systems. Learning theory is based on objective function which minimizes the training errors. In many works, the local optima and over-fitting have been pointed out the problems to decrease the performance of learning methods[35].

Statistical learning theory(SLT) by Vapnik is a good learning theory based on Vapnik-Chervonenkis(VC) dimension. But also it has the local optima and over-fitting problems because the kernel type, kernel parameters, and regularization constant C are determined subjectively by the prior knowledge of researchers. To solve the problems of learning methods, we combine evolutionary computing into SLT. So, we propose an evolutionary SLT(ESLT). SLT has been originally developed for classification as pattern recognition. However, there is a growing empirical evidence of successful applications of the theory to prediction as regression[5]. Most recently, SLT makes possible to perform the clustering[1]. So, SLT is consisted of three types according to learning strategies. They are support vector machine(SVM), support vector regression(SVR), and support vector clustering(SVC) for classification, prediction, and clustering respectively. Among them we consider SVM and SVR in ESLT. We call SVM in ESLT an evolutionary SVM(ESVM) and SVR in ESLT an evolutionary SVR(ESVR). Using ESVM, we construct an efficient model for intrusion detection. To verify the performance of ESVM, the DARPA data set from KDD cup 1999 are used. Also, we show an effective model for web usage mining using our ESVR. Using web log data from KDD cup 2000, we verify the improved performance of our method for prediction.

Sung-Hae Jun is with Department of Bioinformatics & Statistics Cheongju University, Chungbuk, 360-764, Korea (corresponding author to provide phone: +82 43-229-8205; fax: +83 43-229-8432; e-mail: shjun@cju.ac.kr).

Kyung-Whan Oh is with Department of Computer Science Sogang University, Seoul, 121-742, Korea (phone: +82 2-703-7626; fax: +83 43-229-8432; e-mail: kwoh@sogang.ac.kr).

II. Evolutionary Computing and Statistical Learning Theory

A. Evolutionary Computing

Evolutionary computing(EC) is a special type of computing, which draws inspiration from the process of natural evolution. The fundamental of EC relates powerful natural evolution to a particular style of problem solving, that of trial and error[10]. Environment, individual, and fitness of the basic EC were linked respectively problem, candidate solution, and quality of the natural evolution to problem solving.

Evolutionary programming(EP) is another member of the EC family. EP was originally developed to simulated evolution as a learning process with the aim of generating artificial intelligence[13],[14]. The intelligence is viewed as the capability of a system to adapt its behavior in order to meet some specified goals in a range of environments. EP is typically used for continuous parameter optimization, meaning that the problem at hand can be given as an objective function $R^n \to R$. The genotype for a solution with k genes is now a vector ($x_1, \cdots, x_k$) with $x_i \in R$. The recombination is not used in EP. The mutation operator of EP is able to settle the local minima problems of machine learning algorithms.

B. Statistical Learning Theory

SLT effectively describes statistical estimation with small samples. Naturally, as a special case, this theory includes classical statistical methods which are developed for large samples and strict parametric assumptions. SLT is based on principle of empirical risk minimization(ERM) and VC dimension. The empirical risk is the average risk for the training data. This is minimized by choosing the appropriate parameters. For density estimation, the expected risk is given in the following[5].

$$R(w) = \int L(f(x,w))\,p(x)dx \tag{1}$$

This expectation is estimated by taking an average of the risk over the training data.

$$R_{emp}(w) = \frac{1}{n}\sum_{i=1}^{n} L(f(x_i, w)) \tag{2}$$

Then the optimal parameter values are found by minimizing the

empirical risk with respect to *w*. The theory of convergence of $R_{emp}(w)$ to $R(w)$ includes bounds on the rate of convergence, which are based on VC dimension. The VC dimension is a measure of the capacity or expressive power of the family of classification functions realized by the learning machine. VC theory explicitly takes into account the sample size and provides quantitative description of the trade-off between the model complexity and the available information.

III. Web Usage Mining

Web mining can be broadly defined as the discovery and analysis of useful information from the world wide web[6],[7],[12]. In general, web mining tasks can be classified into three categories which are web content mining, web structure mining, and web usage mining[23]. In this paper, we consider the web usage mining from web log data. The sparseness of web log file has been a problem of web usage mining. This is occurred by several reasons. Frequently it happens when the not visited web pages are much larger than the visited web pages in web sites. The click stream data of cleaned web log are very sparse. So, we have a difficulty of web log analysis for web usage mining which includes web information recommendation, next web page prediction, and web page duration time forecasting. The click stream data with sparseness is hard to analyze by general methods as regression, imputation methods, and others[23]. In this case the SVR is very useful tool for analyzing sparse data[20]. But SVR has had local minima problems[20]. So we have needed to solve local minima of SVR. To settle the problems, we propose ESVR in this paper. Because of the sparseness of web log file, the structure of click stream data is incomplete. These incomplete data have extremely many missing values. The missing data patterns from given data are shown in the following[23].

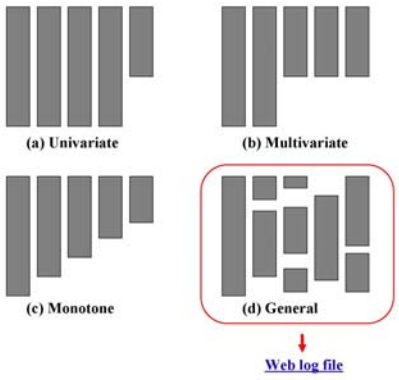

Fig. 1 typical examples of missing data

In the above, we show four missing data patterns. In (a) and (b) of Fig. 1, the missing data are eliminated by simple imputation method as mean and conditional mean methods. (c) shows monotone missing pattern. The method for this pattern is statistical missing data analysis models as multiple imputations. But (d) is a very difficult pattern for imputing missing values. In general, the missing data pattern of web log file is (d). In our work, ESVR is able to be considered for analyzing missing data of web log file.

IV. Intrusion Detection

A network intrusion called an attack is a sequence of related actions by a malicious adversary whose goal is to violate some stated or implied policy regarding appropriate use of a network. Examples include stealing protected data, denying service to a user or group of users, or performing probing actions in an attempt to gain information in preparation for an attack. Growing reliance on the internet and worldwide connectivity has greatly increased the potential damage that can be inflicted by such attacks[16]. Intrusion can be defined as any set of actions that attempt to compromise the integrity, confidentiality or availability of a resource. In the network systems, it refers to any unauthorized access, unauthorized attempt to access or damage, or malicious use of information resources[29]. Detection of anomaly patterns is computationally expensive because of the overhead of keeping track of, and possible updating several system profile metrics, as it must be tailored system to system, and sometimes even user to user, due to the fact behavior patterns and system usage vary greatly. Intrusion detection system(IDS) is designed to identify preferably in real time-unauthorized use, misuse and attacks on information systems. IDS maintains a set of historical profiles or recorded profiles for users, matches an audit record with appropriate profile, updates the profile whenever necessary, and reports any anomalies detected. An IDS does not usually perform any action to prevent intrusions; its main function is to alert the system administrators that there is a possible security violation; as such it is a proactive tool rather than a reactive tool.

V. An Evolutionary Support Vector Machine

A. Evolving SVM

SVM is learning machine that map the training vectors to high dimensional feature space, labeling each vector by its class. SVM views the classification problem as a quadratic optimization problem. It classifies data by determining a set of support vectors, which are members of the set of training inputs that outline a hyperplane in feature space[34]. SVM is based on the idea of structural risk minimization, which minimizes the generalization error, that is, true error on unseen examples. The number of free parameters used in SVM depends on the margin that separates the data points but not on the number of input features, thus SVM does not require a reduction in the number of features. SVM provides a generic mechanism to fit the surface of the hyper plane to the data through the use of a kernel function. The user may provide a function, such as a linear, polynomial, or sigmoid curve, to SVM during the training process, which selects support vectors along the surface of this function. This capability allows classifying a broader range of problems. The primary advantage of SVM is binary classification and regression that they provide to a classifier with a minimal VC dimension[34], which implies low expected probability of

generalization errors. In our paper, all intrusions are classified as +1, and normal data are classified as −1. In ESVM, the main element of classification algorithm is to construct the optimal separating hyperplane. To make this hyperplane, we maximize the quadratic form (3) subject to constraints (4).

$$W(\alpha) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} \alpha_i \alpha_j K(x_i, x_j) y_i y_j \quad (3)$$

$$\sum_{i=1}^{l} y_i \alpha_i = 0, \quad 0 \le \alpha_i \le C, \quad i = 1, 2, \ldots l \quad (4)$$

where, *x* and *y* are the input vector and output label. $K(\cdot,\cdot)$ is a kernel function. The kernels for three common types of SVM are polynomial function, radial basis function(RBF), and sigmoid function[17]. The weights of SVM is defined in the following[34].

$$w = \sum_{i=1}^{l} \alpha_i y_i x_i \quad (5)$$

So, we are able to find w using α , y , and x from the above (3) and (4). In experimental results, we find that ESVM is not dependent upon the kernel types of SVM. EC including EP are search method for optimization problems, in which a mechanics of natural evolution principle is used to obtain the global optimal solution. They have been demonstrated considerable success in combination with other machine learning methods[24],[31],[35]. From them, we show ESVM algorithm in the following.

```
BEGIN
 Set t=0;
 Create an initial population
     ( w , C ) ∈ R^{p+1};
     p: the dimension of weights
     1: a regularization parameter
 Repeat Until (stop condition is satisfied)
 Do
    Mutation: draw z_i from N(0,1)
     y_i^t = x_i^t + z_i for all i ∈ {1, ..., n};
    If (f(x^t) ≤ f(y^t)) then x^{t+1} = x^t;
                         Else x^{t+1} = y^t;
    End if
  Set t=t+1;
 End do
End
```

Fig. 2 pseudo code of ESVM

The stop condition in ESVM has two cases of termination requirements. Firstly, the process of ESVM algorithm is stopped when the total number of fitness evaluations reaches a given limit. Secondly, for a given period of time, until the fitness improvement is remained under a threshold value, our algorithm has been processed. The mutation of ESVM is implemented by adding some random noise drawn from a standard normal distribution. $f(\cdot)$ is a fitness function used in ESVM algorithm. In ESVM, every member of the population creates exactly one offspring via mutation. The regularization parameter, C, which has been subjectively determined by researchers is suitably chosen by the mutation of ESVM. It is determined by the objective function. In this paper, we use misclassification ratio and lift value measures for the objective function. The performance of each method for intrusion detection is better according to deceasing misclassification rate[2].

B. Intrusion Detection using ESVM

In the beginning, though the internet was used at the limited purposes which were a part of national defense and a small group of research, currently, it has been used in the various fields of the whole world. Most information exchanges have been achieved in the internet environments. So, the technologies for information security which protect system from the intrusive attacks are needed. As time goes on, the techniques of intrusion have been cleverer than the skills of detection. Thoughtlessly the attack programs have been widespread by anonymous sources and thus individuals without related knowledge can do intrusion. This is a reason why the crimes of information securities have been increased recently. Anyone can be connected with internet because the usages of the internet have been rapidly increased. Anybody can do cracking, denial of service(DoS), and so forth, to do considerable damage to network systems using attack programs from the internet. The paradigm change of intrusion has been already begun. We find this seriousness from the cases which are distributed denial of service(DDoS) in Yahoo and Amazon web sites harmed by attacks. Most existing models for information securities are constructed by training only known intrusive data[11],[22]. But these have had a difficulty to detect new intrusive patterns which are unknown. So, novel researches for intrusion detection system have been studied by multiple disciplines[11],[19],[22],[37]. Many works for intrusion detection have been published using machine learning algorithms such as neural networks, fuzzy set theory, and support vector machine(SVM)[3],[8],[9],[15],[19],[21],[25], [29],[32]. But these models have the local optima problems [3],[17],[19],[20],[34]. So, we propose ESVM for intrusion detection. ESVM is constructed by combining evolutionary programming into SVM. Our proposed model is able to settle the problems because the global search of parameters by the mutation operator of evolutionary programming are performed in ESVM. The approach of ESVM is to make the detectable model for new attacks patterns as well as known attack patterns. Using ESVM, we are able to detect intrusive patterns which are known and unknown. In experimental results, we verify the performance of ESVM using KDD cup 1999 task data designed by the defense advanced research projects agency (DARPA) [37].

VI. An Evolutionary Support Vector Regression

A. Evolutionary SVR

The 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[33]. 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 in the following.

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

This is denoted in the following.

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

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 in the following.

$$f(x) = < w, x > + b \tag{8}$$

where, $< \cdot, \cdot >$ 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[34]. Analogously to the loss function in [34], we introduce slack variables ξ_i, ξ_i^* to copy with otherwise infeasible constraints of the optimization problem.

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

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

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)[34]. In this section, we propose ESVR. Genetic algorithm(GA) has provided a analytical method motivated by an analogy to biological evolution[27]. General GA computes the fitness of given environment where is fixed. Distinguished from traditional GA, co-evolving approach is evolutionary mechanism of the natural world with competition or cooperation. The organism and the environment including organism evolve together[28]. We apply not cooperation but competition to our proposed co-evolutionary model. Our competitive co-evolving approach use host-parasites co-evolution. The host and parasites are used for modeling ESVR and training data set. Our ESVR and training data set are considered as the organism and the environment including it. That is, the evolving ESVR is followed the evolution of host. The initial parameters for ESVR model are determined as uniform random numbers from -1 to 1. The fitness function of ESVR is the inverse form of the squared error between real and predict values in the following.

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

In above equation, *t* is the value of known target variable and *o* is computed output value for prediction. *C* is a constant. *F* and N_{out} are the numbers of patterns and items. Next, the training of given data set is performed by evolving parasites. The evolution of training data is performed to retain larger training errors. So, the fitness function for training data set is inverse form of the fitness function of ESVR model in the following.

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

D and N_{out} are the numbers of patterns and items in the above equation. Our approache of ESVR and training data set are competitive. In other words, the model is two different groups' competitive co-evolving. One is the parasites' evolution of given training data set. Another is the host's evolution of ESVR. The following shows the process of proposed method.

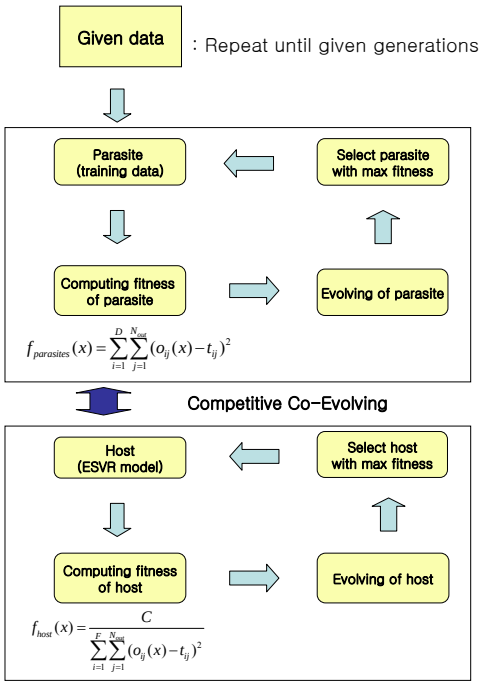

Fig. 3 process of ESVR

In the above, the ESVR model and training data set are respectively evolved. During evolution for weight optimization of ESVR, the competitive co-evolving is occurred between evolving SVM model and evolving training data set. In this place, our model use co-evolutionary computation instead of Lagrange multipliers of traditional SVM for parameter optimization. The following is a pseudo-code of ESVR.

```
BEGIN
INITIALIZE population with Uniform[-1,1]
EVALUATE
1. ESVR model by f_host(x) ;
2. training data set by f_parasites(x) ;
 REPEAT UNTIL
 (TERMINATION CONDITION is satisfied)
 DO
  1. SELECT parents;
  2. MUTATE the resulting offspring;
  3. EVALUATE new candidates;
  4. SELECT individuals for the next generation;
 Loop
END
```

Fig. 4 pseudo code of ESVR

B. Web Usage Mining using ESVR

In the web mining approaches, web usage mining is mainly to analyze the click stream data from web log file. The web log records contain many collections of hyperlink information and the usage transactions of web page access. The size of web log data is very large, but they are very sparse. So, we have had serious difficulties for web usage mining. It is very difficult to estimate the dependencies of all web pages in the spare web log data. We find SLT is a good approach for analyzing the sparse data because of its ε-insensitive loss function[34]. Using SLT as a missing value imputation, the spare data set is changed to complete data set[20]. Our previous research provided a useful strategy for analyzing sparse data like web log data. Our work was to use SVR among statistical learning models[20]. SVR is the regressive version of SLT[17]. SVM is able to be applied to the case of regression, maintaining all the main features that characterize the maximal margin. Using SVR, we made an efficient missing value imputation model to analyze sparse web log data[20]. But, including SLT, the learning approaches based on minimizing objective function of errors have local minima problems yet to be solved. The recent evolving researches have played an important role for constructing optimal models without minimizing training errors. So, combining the evolutionary computing into SVR, we propose ESVR. Our model is able to offer a good result in spare web log data. In experiments using KDD cup 2000 data[36], we verify the performance of our work. In this paper, to eliminate the sparseness in click stream data, a missing value imputation approach is used. Our imputation method is ESVR. This has a good performance for sparse data analysis because of its ε-insensitive loss function[34]. This satisfies conditions for consistency of risk minimization principle[34]. What is more, ESVR is an evolutionary approach to solve local optima of general SVR. Fig. 5 shows sparse click stream data from web log file. This must be complete for web usage mining.

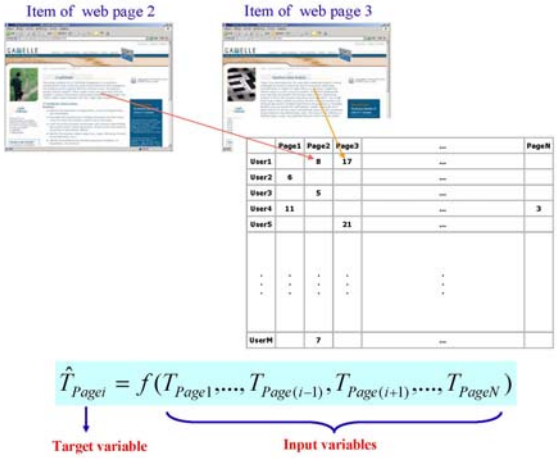

Fig. 5 incomplete click stream data

A cell of above table structure contains a duration time of each user accessing. The sparseness of cells is very serious. Therefore, general missing value imputation methods as multiple imputation method are not suitable to solve the problem. In our research, ESVR is used for sparseness elimination in web log file. Fig. 6 shows complete table without sparseness using ESVR.

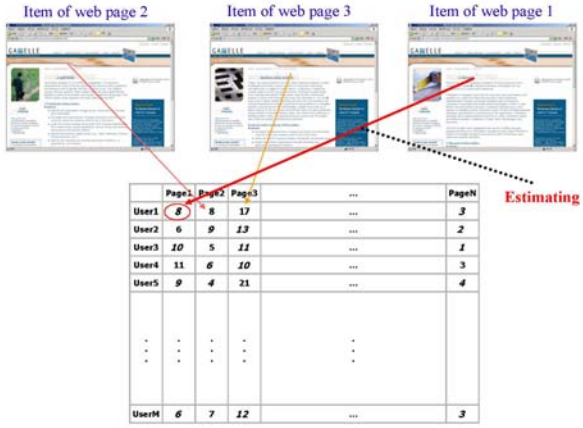

Fig. 6 complete data without missing values

The duration time of *i*th page is estimated as following equation in ESVR.

$$\hat{T}_{Pagei} = f(T_{Page1},\ldots,T_{Page(i-1)},T_{Page(i+1)},\ldots,T_{PageN}) \quad (13)$$

In the equation, T_{PageK} is defined the duration time of page *K* by user accessing and $\hat{T}_{Pagei}$ is defined the estimated duration time *i*th page. $\hat{T}_{Pagei}$ is computed by the ESVR method of (*N*-1) pages, $T_{Page1},\ldots,T_{Page(i-1)},T_{Page(i+1)},\ldots,T_{PageN}$ out of *i*th page. So, we are able to predict all pages' duration time as using estimating the time of missing cells. This approach changes sparse table of Fig. 5 into complete table of Fig. 6.

VII. Experimental Results

A. Intrusion Detection using KDD cup 1999

To verify the performance of ESVM, we use the network packet data from KDD cup 1999. The data were designed by DARPA for evaluating the results of intrusion detection[37]. Then, the original data were biased to specific types. So, we do random sampling from the given data to overcome the bias problem because it is difficult to construct an effective model in the unbalanced data set. The following table shows the bias original data and sampled data.

TABLE I
Summary of Original and Sampled Data Sets

Types	Attack or Normal	Number of instances (original data)	Number of instances (sampled data)
DoS	Attack	3,883,370	194,169
R2L	Attack	1,126	56
U2R	Attack	42	42
Probing	Attack	41,102	2,055
Normal	Normal	972,780	48,639

In the above table, Dos, R2L, U2R, and Probing are denial of service, unauthorized access from a remote machine, unauthorized access to local super user privileges, and surveillance(other probing). The sampled data for verifying the performance of ESVM are used. Therefore, to construct an effective model for intrusion detection, we use all U2R instances and sampled DoS, R2L, Probing, and Normal instances with proportional probability to each size. Firstly we make an experiment on the comparative performances of SVMs with polynomial, RBF(radial basis function), and sigmoid kernels. The following table shows the experimental result.

TABLE II
Performance Evaluation according to Kernels in ESVM

Kernel type	Misclassification rate	Lift value(LV)
Polynomial	0.0398	3.01
RBF	0.0365	3.42
Sigmoid	0.0371	3.23

In the experiment, the misclassification rate and lift value(LV) are used for evaluating the performances of comparative models. Lift is a measurement of how much better the data mining model predicted results for a given case set over what would be achieved through random selection. Lift is typically calculated by dividing the percentage of expected response predicted by the data mining model by the percentage of expected response predicted by a random selection. For example, suppose that 2% of the customers mailed a catalog without using the model would make a purchase. However, using the model to select catalog recipients, 10% would make a purchase. Then the lift is 10/2 or 5. Lift may also be used as a measure to compare different intrusion detection models. Since lift is computed using a data table with actual outcomes, lift compares how well a model performs with respect to this data on predicted outcomes. Lift indicates how well the model improved the predictions over a random selection given actual results. Lift allows a researcher to infer how a model will perform on new data. Generally the *LV* is defined as the following[18].

$$LV = \frac{\%response}{LV_{BL}} \tag{14}$$

In the above equation, *%response* is percentage of the number of correctly predicted attacks using constructed model and LV_{BL} is the base line lift value which is the predicted result by random selection without modeling. The model has twice improved performance when its *LV* is 2.

In the above table, the misclassification rates and the LVs are nearly not changed according to the types of kernels. So, we know that the performance of ESVM is not dependent upon the kernel functions. But original SVM is commonly dependent on the kernel types[17],[34]. From the experiment, we find that ESVM can be converged on global optimum because of its independence on the kernel types. Consequently the EP approach of ESVM using Gaussian mutation overcomes the local optima problems of SVM. In the following, the performance evaluations of the comparative models are shown. We compare ESVM with popular classification methods which are decision tree, logistic regression, Gaussian mixture model, and SVMs with different kernels[2],[5],[17],[18],[26],[30],[34].

TABLE III
Performance Evaluation of Comparative Methods

Kernel type		Misclassification rate	Lift value
ESVM		0.0365	3.42
SVM	Polynomial	0.2154	1.11
	RBF	0.0619	2.41
	Sigmoid	0.1956	1.35
Decision tree		0.2143	1.24
Logistic		0.1017	2.09
Gaussian mixture		0.0956	2.33

We find that the performance of each SVM is severely varied according to its kernel type in the above results. The RBF kernel of SVM is the best among three kernels in this data set. Generally the kernels of SVM are dependent on the specific of given training data set. From the result of the table, the improved performance of ESVM is shown. Therefore, we are able to verify our ESVM.

B. Web Usage Mining using KDD cup 2000

Using the web log data from KDD cup 2000, we show the improved performance of ESVR. In this experiments, we use mean squared error(MSE) which are the mean of the difference between the observed and predicted values as the performance measure of each model. It is defined in the following[4].

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{15}$$

where, y_i is the observed value and $\hat{y}_i$ is the predicted value. The smaller the value of MSE is, the better the performance of the method is.

The deployment of ESVR is to construct an efficient model for web usage mining using the click stream data of web log file. The data set is web log file of real internet shopping mall. We use the one-third of given data for the validation and the

other two-thirds for training [27]. After data cleaning, we get the data set as the following summary.

TABLE IV
SUMMARY OF CLEANING DATA

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

In the above table, the cookie-id is the index of user accessing to web site. The assortment-id represents each web page containing the descriptive contents of each item in the shopping mall and the duration-time of web page has the value between 0 and 1000 seconds. To compare with ESVR, we use multiple regression, multi layer perceptron(MLP), nearest neighbor, and SVR which are established methods for predictive models. The following table shows the experimental result.

TABLE V
EVALUATION RESULTS

Methods	MSE_T	MSE_V	$(MSE_T+ MSE_V)/2$	$\| MSE_V-MSE_T \|$
Multiple Regression	3.81	5.35	4.58	1.54
MLP	2.99	4.19	3.59	1.2
Nearest neighbor	2.78	4.08	3.43	1.3
SVR	1.55	1.83	1.69	0.28
ESVR	1.34	1.53	1.42	0.16

In the above table, MSE_T and MSE_T are MSE of training and validation data sets respectively. $(MSE_T+ MSE_V)/2$ is the average value of MSE of training and validation data. We also see the difference between MSE_T and MSE_T in $\left|MSE_V - MSE_T\right|$. The MSE_T and MSE_T of ESVR is smaller than other methods. Also, the value of $\left|MSE_V - MSE_T\right|$ of ESVR is 0.16. So, we know that ESVR settles the over fitting problem of learning models, because the value is represented the performance difference of models between training and validation data. In general, according to this value is decreased, the performance of learning model is able to be good. Therefore, using objective KDD cup data set, we verify an improved performance of ESVR.

VIII. CONCLUSIONS AND FUTURE WORKS

In this paper, we propose an evolutionary statistical learning theory. Our theory is consisted of two methods which are ESVM and ESVR for classification and prediction respectively. We find that ESVM and ESVR are improved approaches to settle the local optima and over fitting problems of learning models. Using data sets from KDD cup 1999 and 2000, we verify the performances of our research.

In future works, we will combine more diverse evolving into learning methods to overcome the problems of learning models.

REFERENCES

[1] A. Ben-Hur, A. D. Horn, H. Siegelmann, V. Vapnik, "Support Vector Clustering," *Journal of Machine Learning Research 2*, 2001, pp. 125-137.
[2] L. Breiman, J. H. Friedman, R. A. Olshen, C. J. Stone, *Classification and Regression Trees*, Wadsworth Inc., 1984.
[3] J. Cannady, "Artificial Neural Networks for Misuse Detection. National Information Systems," *Proceedings of Security Conference*, 1998.
[4] G. Casella, R. L. Berger, *Statistical Inference*, Duxbury Press, 1990.
[5] V. Cherkassky, F. Mulier, *Learning From Data Concepts, Theory, and Methods*, John Wiley & Sons, 1998.
[6] R. Cooley, B. Mobasher, J. Srivastava, "Web Mining: Information and Pattern Discovery on the World Wide Web," *Proceeding of the 9th IEEE International Conference on Tools with Artificial Intelligence*, 1997.
[7] R. Cooley, P. N. Tan, J. Srivastava, "Discovery of interesting usage patterns from web data," *Technical Report TR 99-022*, University of Minnesota, 1999.
[8] H. Debar, M. Becke, D. Siboni, "A Neural Network Component for an Intrusion Detection System," *Proceedings of the IEEE Computer Society Symposium on Research in Security and Privacy*, 1992, pp. 240-250.
[9] H. Debar, B. Dorizzi, "An Application of a Recurrent Network to an Intrusion Detection System," *Proceedings of the International Joint Conference on Neural Networks*, 1992, pp 78-483.
[10] A. E. Eiben, J. E. Smith, *Introduction to Evolutionary Computing*, Springer, 2003.
[11] S. M. Emran, M. Xu, N. Ye, Q. Chen, X. Li, "Probabilistic techniques for intrusion detection based on computer audit data," *IEEE Transactions on Systems, Man and Cybernetics, Part A, vol.31*, 2001, pp.266-274.
[12] D. Fisher, K. Hildrum, J. Hong, M. Newman, M. Thomas, R. Vuduc, "SWAMI: A Frame-work for Collaborative Filtering Algorithm Development and Evaluation," *Proceeding of SIGIR 2000*, ACM Press, 2000.
[13] D. B. Fogel, *Evolutionary Computation*, IEEE Press, 1995.
[14] L. J. Fogel, A. J. Owens, M. J. Walsh, *Artificial Intelligence through Simulated Evolution*, Wiley, Chichester, UK, 1996.
[15] A. K. Ghosh, *Learning Program Behavior Profiles for Intrusion Detection*, USENIX, 1999.
[16] J. W. Haines, R. P. Lippmann, D. J. Fried, M. A. Zissman, E. Tran, S. B. Boswell, "1999 DARPA Intrusion Detection Evaluation: Design and Procedures," *Technical Report 1062*, Lincoln Laboratory, MIT, 2001.
[17] S. Haykin, *Neural Networks*, Prentice Hall, 1999.
[18] S. Huet, A. Bouvier, M. A. Poursat, E. Jolivet, *Statistical Tools for Nonlinear Regression*, Springer Series in Statistics, Springer, 2003.
[19] S. H. Jun, "Hybrid Statistical Learning Model for Intrusion Detection of Networks," *The KIPS Transaction: Part C*, vol. 10-C, no. 6, 2003, pp. 705-710.
[20] S. H. Jun, "Web Usage Mining Using Support Vector Machine," Lecture Note in Computer Science, vol. 3512, 2005, pp. 349-356.
[21] S. Kumar, E. H. Spafford, "An Application of Pattern Matching in Intrusion Detection," *Technical Report CSD-TR-94-013*, Purdue University, 1994.
[22] W. Lee, S. J. Stolfo, K. W. Mok, "A data mining framework for building intrusion detection models," *Proceedings of the 1999 IEEE Symposium on Security and Privacy*, 1999, pp.120-132.
[23] R. J. A. Little, D. B. Rubin, *Statistical Analysis with Missing Data*, Wiley Inter-Science, 2002.
[24] B. Liu, "Fuzzy Random Chance-Constrained Programming," *IEEE Transactions on Fuzzy Systems*, vol. 9, Issue 5, 2001, pp. 713-720.
[25] J. Luo, S. M. Bridges, "Mining Fuzzy Association Rules and Fuzzy Frequency Episodes for Intrusion Detection," *International Journal of Intelligent Systems*, John Wiley & Sons, 2000, pp. 687-703.
[26] G. Mclachlan, D. Peel, *Finite Mixture Models*, John Wiley & Sons, Inc., 2000.
[27] T. M. Mitchell, *Machine Learning*, McGraw-Hill, 1997.
[28] T. M. Mitchell, *An introduction to Genetic Algorithms*, MIT Press, 1998.
[29] S. Mukkamala, G. Janoski, A. Sung, "Intrusion Detection Using Neural Networks and Support Vector Machines," *Proceedings of International Symposium on Applications and the Internet Technology*, 2000, pp. 209-216
[30] R. H. Myers, *Classical and Modern Regression with Applications*, Duxbury Press, 1990.

[31] A. T. Quang, Q. L. Zhang, X. Li, "Evolving Support Vector Machine Parameters," *Proceedings of the First International Conference on Machine Learning and Cybernetics*, 2002, pp. 548-551.
[32] J. Ryan, M. J. Lin, R. Miikkulainen, "Intrusion Detection with Neural Networks," *Advances in Neural Information Processing Systems 10*, Cambridge, MA: MIT Press, 1998.
[33] A. J. Smola, *Regression estimation with support vector learning machines*, Master's thesis, Technische University, 1996.
[34] V. Vapnik, *Statistical Learning Theory*, John Wiley & Sons, Inc., 1998.
[35] X. Yao, "Evolving Artificial Neural Networks," *Proceedings of the IEEE*, vol. 87, Issue 9, 1999, pp. 1423-1447.
[36] http://www.ecn.purdue.edu/KDDCUP
[37] http://www.ll.mit.edu/IST/ideval/data

Sung-Hae Jun received the BS, MS, and PhD degrees in department of Statistics, Inha University, Incheon, Korea, in 1993, 1996, and 2001. He is currently Assistant Professor in department of Bioinformatics & Statistics, Cheongju University, Chunkbuk, Korea. He is also PhD candidate of Computer Science, Sogang University, Seoul, Korea. He has researched statistical data analysis, machine learning and evolutionary computing.

Kyung-Whan Oh received the BS degree in mathematics from Sogang University, Seoul, Korea, in 1978, and the MS and PhD degrees in computer science from Florida State University, Tallahassee, USA in 1985 and 1988, respectively. He is currently with the department of computer science at Sogang University, Seoul, Korea, where he is a Professor. His research and teaching interests include fuzzy system, cognitive science, knowledge discovery and data mining, intelligent agents and multi-agent systems, expert system and statistical learning.