Transactions of the Korean Nuclear Society Spring Meeting
Jeju, Korea, May 29-30, 2014

Development of Advanced Operation Validation System using Support Vector Regression Algorithm

Seung Geun Kim [a*], Poong Hyun Seong [a]
[a] *Dept. of Nuclear and Quantum Engineering, Korea Advanced Institute of Science and Technology, Daejeon, Korea*
[*] *Corresponding author: ksg92@kaist.ac.kr*

1. Introduction

Since the size and complexity of nuclear power plants (NPPs) are both increasing, the understanding of system problems and mitigation poses challenges to the operators [1]. In order to enhance the safety of NPPs, reducing operation errors committed by human operators is essential, and various operator support systems (OSSs) have been developed. Many researchers have made a huge effort to study about fault diagnosis systems (FDSs) which help operators to judge what kind of abnormalities is occurred. However, there seems to need further researches about operation validation systems (OVSs) which help operators to properly and effectively mitigate the abnormalities of NPPs.

In this study, a new OVS based on support vector regression (SVR) algorithm is proposed. SVR is one of the branches of support vector machine (SVM) applications. SVM is a machine-learning algorithm that has been successfully used in pattern recognition for cluster analysis [2]. It is widely used for various regression analysis fields because of its evident theoretical background, high performance in finding global optimum, and high performance in real application as well as artificial neural network (ANN), which is already widely used in various fields. With these advantages, it is expected that the new OVS based on SVR algorithm may provide a chance to conduct not only qualitative but also quantitative analysis about results of operators' actions.

The data for training the SVR algorithm was acquired by using MAAP5 (modular accident analysis program) code, which is world-widely used and is provisionally proved as reliable simulation code. Data was acquired based on APR1400 (advanced power reactor) reactor parameters. With the advantages of SVR and reliable data, the new OVS will be validated with enhanced performance in real NPP applications. Through the validation results, it is expected that human errors can be significantly reduced in implementation phase.

2. Brief Introduction of SVR Algorithm

2.1 Historic Background of Support Vector Machine

Support vector machine (SVM) is one of the optimization algorithms which are used for selecting best model in statistical measures, suggested by Vapnik in 1995 [3]. The structure of SVM is based on the structural risk minimization (SRM) principle, which has been shown to be superior to pre-developed empirical risk minimization (ERM) principle, employed by conventional neural networks [4]. Although SVM has shorter history compared to other algorithm such as neural networks, SVM algorithm gains popularity in various fields of research because of its underlying attractive features. For example, since SVM is based on evident mathematical background, analysis of the result from SVM algorithm is relatively easier than the result from other algorithms such as neural networks. In addition, SVM shows high performance in real-world applications as well as neural networks, which used for a long time in similar fields.

SVM algorithm was originally designated for solving classification problems, but it can be also extended to the domain of regression problems [5]. In order to classify these two applications, they are called support vector classification (SVC) and support vector regression (SVR) respectively. SVC and SVR shares same basic principles, but there are some differences in detail. In this study, SVR was used for regression analysis and its feature will be described in next section briefly.

2.2 The Basic Idea of Support Vector Regression

Relation between regression function $f(x)$ and training data set $\{(x_1, y_1), \cdots (x_n, y_n)\}$ can be represented in the form,

$$f(x) = w \cdot x + b \quad \text{with} \quad w, x \in R^N,\ b \in R \tag{1}$$

$$\text{where } \{(x_1, y_1), \cdots (x_n, y_n)\} \subset R^N \times R \tag{2}$$

By the principle of SRM, generalization performance of SVR is related to the 'flatness' of regression function, and this 'flatness' is obtained by minimizing w in Eq. (1) (i.e. minimizing $\|w\|^2$).

The regression process can be represented in the form of optimization problem, by adding some variables such as C (adjusting factor that adjusts trade-off between flatness and training error), ε (size of ε-tube), and slack variables ξ, ξ^* (error allowance in training process). Equations of optimization problem is represented as,

$$
\begin{aligned}
&\text{Minimize} && \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \\
&\text{Subject to} && y_i - w\cdot x_i - b \le \varepsilon + \xi_i \\
& && w\cdot x_i + b - y_i \le \varepsilon + \xi_i^* \\
& && \xi_i, \xi_i^* \ge 0
\end{aligned}
\tag{3}
$$

If the ε-loss function is ε-insensitive, SVR does not calculate training error of data set which is located inside of ε-tube, and as a result, SVR tends to cover the data as much as possible. In conclusion, SVR is trained to cover data as much as possible, and minimize $\|w\|^2$ [6].

Minimization problem in Eq. (3) is called the primal objective function. By introducing a dual set of variables, Lagrange function is constructed from the primal objective function (for details see e.g. [7], [8]). Eq. (4) shows the Lagrange function constructed from Eq. (3) and its constraints.

$$
\begin{aligned}
L = {} & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) - \sum_{i=1}^{n}(\lambda_i\xi_i + \lambda_i^*\xi_i^*) \\
& - \sum_{i=1}^{n}\alpha_i(\varepsilon + \xi_i - y_i + (w\cdot x + b)) \\
& - \sum_{i=1}^{n}\alpha_i^*(\varepsilon + \xi_i^* - y_i + (w\cdot x + b))
\end{aligned}
\tag{4}
$$

L is the Lagrangian and $\alpha_i, \alpha_i^*, \lambda_i, \lambda_i^*$ are Lagrange multipliers. Equation (5) represents the positivity constraints for Lagrange multipliers.

$$
\alpha_i, \alpha_i^*, \lambda_i, \lambda_i^* \ge 0 \tag{5}
$$

Partial derivatives of L respect to w, b, ξ_i, ξ_i^* yields these three equations respectively.

$$
\begin{aligned}
&\partial_b L = \sum_{i=1}^{n}(\alpha_i^* - \alpha_i) = 0 \\
&\partial_w L = \sum_{i=1}^{n}(\alpha_i^* - \alpha_i)x_i = 0 \\
&\partial_{\xi_i^{(*)}} L = C - \alpha_i^{(*)} - \lambda_i^{(*)} = 0
\end{aligned}
\tag{6}
$$

Note that $\alpha_i^{(*)}$ means α_i and α_i^*. By combining Eq. (5) and (6), dual optimization problem is obtained.

$$
\begin{aligned}
&\text{maximize} && -\frac{1}{2}\sum_{i,j=1}^{n}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)(x_i\cdot x_j) \\
& && -\varepsilon\sum_{i=1}^{n}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{n}x_i(\alpha_i - \alpha_i^*) \\
&\text{subject to} && \sum_{i=1}^{n}(\alpha_i - \alpha_i^*) = 0 \quad \text{and} \quad \alpha_i, \alpha_i^* \in [0, C]
\end{aligned}
\tag{7}
$$

In deriving Eq. (7), dual variables λ_i, λ_i^* are cancelled by the Eq. (6) which can be represented as $\lambda_i^{(*)} = C - \alpha_i^{(*)}$. First line of Eq. (6) can be rewritten as,

$$
\begin{aligned}
&w = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)x_i, \quad \text{thus:} \\
&g(x) = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)(x_i\cdot x) + b
\end{aligned}
\tag{8}
$$

This is called support vector machines regression expansion, and this means that w can be completely described as a linear combination of the training patterns x_i.

The Karush-Kuhn-Tucker (KKT) conditions [9, 10] are the basics for the Lagrangian solution. These conditions state that at the solution point, the product of dual variables and constraints has to vanish [11].

$$
\begin{aligned}
&\alpha_i(\varepsilon + \xi_i - y_i + w\cdot x_i + b) = 0 \\
&\alpha_i^*(\varepsilon + \xi_i + y_i - w\cdot x_i - b) = 0
\end{aligned}
\tag{9}
$$

$$
\begin{aligned}
&(C - \alpha_i)\xi_i = 0 \\
&(C - \alpha_i^*)\xi_i^* = 0
\end{aligned}
\tag{10}
$$

From these conditions, two kinds of conclusions can be made. First one is, only sample sets (x_i, y_i) with corresponding $\alpha_i^{(*)} = C$ are not covered by ε-tube. And second one is multiplication of α_i and α_i^* should be always zero. These conclusions can be represented as,

$$
\begin{aligned}
&\varepsilon - y_i + w\cdot x_i + b \ge 0 \ \text{and}\ \xi_i = 0 && \text{if}\ \alpha_i \le C \\
&\varepsilon - y_i + w\cdot x_i + b \le 0 && \text{if}\ \alpha_i > 0
\end{aligned}
\tag{11}
$$

Also, by the Eq. 9, if any sample set is covered by ε-tube, corresponding α_i, α_i^* has to be zero. The training sample sets that have corresponding non-zero α_i or α_i^* are called as support vectors.

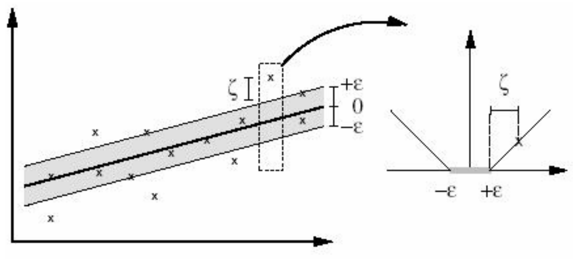

Fig. 1. Linear SVR with Vapnik's ε-insensitive loss function [12].

Computing of SVR algorithm will not described in this paper. For details about this subject, refer [13].

Non-linear regression can be also conducted with SVR algorithm by using kernel trick. This can be achieved by simple mapping of original feature space $\Re$ into some feature space $\Im$ with mapping function Ψ , and applying standard linear SVR algorithm.

For example, non-linear regression of 2-dimensional original feature space (x, y) can be achieved by mapping it into 3-dimensional feature space by utilizing mapping function Ψ .

$$\Psi : \mathbb{R}^2 \rightarrow \mathbb{R}^3$$
$$\Psi(x, y) = (x^2, \sqrt{2}xy, y^2) \tag{12}$$

When kernel trick is applied, SVM optimization problem and SVR expansion both change. If the mapping function denoted as K , instead of Ψ , SVM optimization problem can be represented as,

$$\begin{aligned} &\text{maximize} \quad -\frac{1}{2}\sum_{i,j=1}^{n}(\alpha_i-\alpha_i^*)(\alpha_j-\alpha_j^*)K(x_i\cdot x_j) \\ &\qquad\qquad -\varepsilon\sum_{i=1}^{n}(\alpha_i+\alpha_i^*)+\sum_{i=1}^{n}x_i(\alpha_i-\alpha_i^*) \\ &\text{subject to} \quad \sum_{i=1}^{n}(\alpha_i-\alpha_i^*)=0 \quad \text{and} \quad \alpha_i,\alpha_i^* \in [0,C] \end{aligned} \tag{13}$$

And similarly, SVR expansion can be represented as,

$$\begin{aligned} &w=\sum_{i=1}^{n}(\alpha_i-\alpha_i^*)K(x_i) \qquad \text{thus:} \\ &g(x)=\sum_{i=1}^{n}(\alpha_i-\alpha_i^*)K(x_i\cdot x)+b \end{aligned} \tag{14}$$

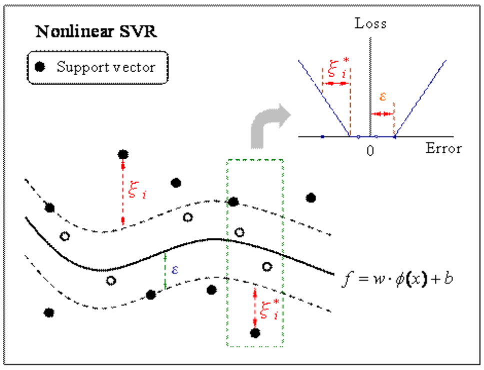

Fig. 2. Non-linear SVR with Vapnik's ε-insensitive loss function [14].

In non-linear regression, the optimization problem finds the 'flattest' regression function in new feature space, not in original feature space.

In normal, several kinds of mapping functions are used frequently. Table I lists frequently used mapping functions and own required specifications.

Table I: Frequently used mapping functions

Kernel function	Specifications
Linear	-
Polynomial	Degree of polynomial
RBF	Sigma (width)
Exponential-RBF	Sigma (width)
Spline	-
B-spline	Degree of B-spline
Anovaspline	Max. order of terms
Fourier	Degree of Fourier series
Sigmoid	Scale, offset

3. Application of SVR Algorithm

3.1 Training data acquisition: MAAP5 code

In order to develop the new OVS, it is necessary to collect data sets that indicate how plant status variables are changed when operators control something such as pumps and valves. These data sets are used for training SVR algorithm, and properly-trained SVR algorithm should have the capability of prediction of plant status variables change under various conditions. Since the quality of training data determines the quality of trained SVR algorithm, acquisition of reliable training data is necessary and very important.

In this study, training data was acquired by using MAAP5 code developed by Fauske associates, which is world widely used for transient analysis and is provisionally proved as reliable simulation code. Data was acquired based on APR1400 reactor parameters.

In the process of data acquisition, there were some assumptions and considerations. First was the kind of transient scenarios. In this study, for simplicity, only LOCA scenario is considered. Instead, 3 break sizes (0.01 ft radius, 0.1 ft radius, and 0.1 ft radius) and 8 break locations (hot leg, cold leg, S/G hot tube, S/G cold tube) were assumed (total 12 kinds of LOCA scenarios). Table II shows the list of assumed transient scenarios.

Table II: Assumed transient scenarios

Transient type	Break size (ft^2)	Break location
LOCA	0.02 0.2 1	Hot leg, Cold leg, S/G hot tube, S/G cold tube

Second consideration was the kind of control modes. Considered control modes were selected by reviewing MAAP5 code manual, with the criteria of accessibility to the code. Before data acquisition, total 23 kinds of control modes were considered, but meaningful data was obtained only from several control modes (will be described in detail in later part). Table III lists 23 considered control modes.

Table III: Considered 23 control modes

HPI switch	MFWP	Cavity injection pump
LPI switch	RX vent switch	Primary system makeup flow
Accum. block valve	UHI accum.	Letdown switch
MCP switch	Charging pump switch	PRHR valve
Recirculation switch	S/G relief valve	S/G MSIV
PZR spray	S/G safety valve	Turbine stop valve
PZR heater	RHR spray valve	Turbine bypass valve
Motor-driven AFWP	RWST valve	

The last consideration was the kind of plant status variables. Considered plant status variables were selected by reviewing procedures for LOCA transient scenario and reviewing MAAP5 code manual, with the criteria of relations with safety and accessibility to the code. Table IV lists plant status variables that considered.

Table IV: Considered plant status variables

S/G 1 pressure	PRZ level	Hot leg temp. (4 nodes)
S/G 1 level	PRZ valve water flow rate	Cold leg temp. (4 nodes)
S/G 2 pressure	PRZ valve gas flow rate	Avg. RCS pressure
S/G 2 level	Accum. pressure	Avg. RCS void fraction
PRZ pressure	Loop 1 flow rate	RWST level
PRZ water temp.	Loop 2 flow rate	Max. core temp.
PRZ gas temp.	Boric acid mass	RV water level

Input files for MAAP5 code were constructed with these 3 kinds of considerations. In this process, control actions that automatically conducted were normally operated. Detailed training data specifications are introduced in table V.

Table V: Training data specifications

1. LOCA occurs at $t = 0$ min.
2. No manual actions are conducted before $t = 30$ min.
3. At $t = 30$ min., operator actuates one of the selected control modes.
4. At $t = 90$ min., simulation ends.
5. Data was collected for every 36 seconds

3.2 Data smoothing method: Savitzky-Golay filter

Applying data smoothing will be helpful to enhance the performance of regression analysis. There are various data smoothing methodologies, and in this study, Savitzky-Golay filter was applied for data smoothing. Savitzky-Golay filter is one kind of digital filters which developed by A. Savitzky and M. J. E. Golay in 1964 [15]. This filter was originally developed for the analytic chemistry.

With determined span N and degree x, Savitzky-Golay filter starts to approximate N sequential data with polynomial of degree x (i.e. making local polynomial), by the method of least squares. For example, if Savitzky-Golay filter with span 10 and degree 2 is applied to the data set which has 100 data points, there would be 90 local polynomials of degree 2. After obtaining all local polynomials, next step is just averaging these local polynomials to get new data points (if the span is N, one original data point would be included in N local polynomials). Fig. 3 and Fig. 4 show one of the training data before filtering and after filtering respectively.

Mathematical representation of Savitzky-Golay filter will not be covered in this paper. For details about this subject, refer [15].

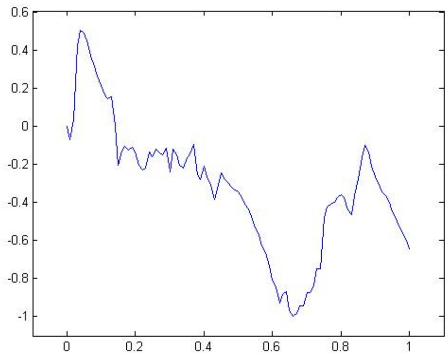

Fig. 3. Training data before Savitzky-Golay filtering

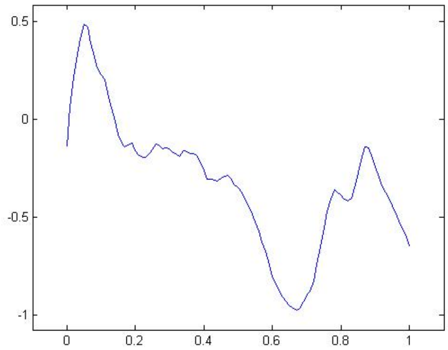

Fig. 4. Training data after Savitzky-Golay filtering (span=10, degree=2, # of data points=100)

3.3 Training and Optimization

Even if operator actuates the same control mode, its result could be different according to the state of NPP at the moment of control. Therefore, to predict the change of plant status variables more accurately in transient condition, obtaining training data under transient condition (not in normal state) is necessary.

By using MAAP5 code, two kinds of LOCA transient data were acquired. First one is transient data without any manual control and second one is transient data with one manual control. By subtracting these two results, transition of plant status variables due to corresponding control action can be obtained. For every selected control modes, this process was applied, and SVR algorithm was trained by using these data.

Among 23 control modes, only several kinds of control modes show significant change of plant status variables. This means that except these controls, the other controls did not affect significantly to safety-related plant status variables. Therefore, the control modes that did not affect significantly were excluded from optimization and training.

To optimize, various kernel functions and various values for parameters such as C, degree of polynomial (in case of the kernel function is polynomial) were applied, and set of kernel function and parameter values that shows minimum error was selected. Table VI lists the applied kernel functions and parameter values to optimization process, and Fig. 5 shows one of the optimized training results of SVR algorithm.

Table VI: Applied kernel functions and parameter values to optimization process

Kernel functions	Linear Polynomial (deg 2 to 6) RBF (sigma 1 to 3) Exponential RBF (sigma 1 to 3) Spline B-Spline (deg 1 to 3) Anovaspline (Max. term 1 to 5)
Loss function	ε-insensitive (fixed)
C	10^n (n=-6,-4,-2,0,2,4,6,8,10,12), ∞
ε	0.03 (fixed)

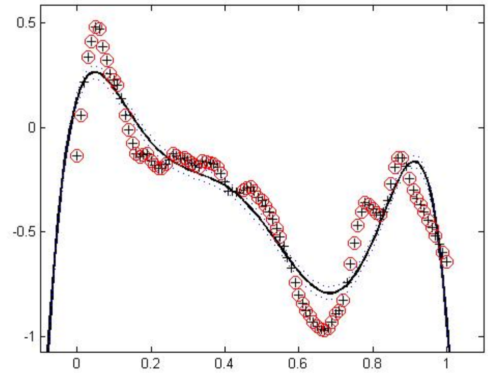

Fig. 5. Training result of SVR algorithm (kernel function: polynomial, deg 6, loss function: ε-insensitive, C=10 E10, ε=0.03, data normalized, data smoothen by Savitzky-Golay filter, span 10, deg 2)

4. Conclusions

In this paper, new OVS using SVR algorithm is proposed. Under simplified cases, SVR algorithm is trained to be capable of plant status variables prediction.

After optimization and training for each data set, all trained data are supposed to be verified. To verify, it is necessary to check whether the trained SVR algorithm predicts the change of plant status variables accurately when operators actuate some control modes (the condition should not be same with the training data). If the output of trained SVR algorithm and real data (obtained from reliable source) show an error in acceptable range, trained SVR algorithm could be verified. If the result is successfully verified, it is expected that development of OVS using SVR algorithm is not only possible but also quite reliable.

After verification, all trained data will be integrated, and the prototype of new OVS would be proposed. Prototype will be constructed by using MATLAB GUI.

For simplification, in optimization process, ε value was fixed and C value varies only 11 kinds. Under this condition of optimization, some data sets show poor regression result. For more perfect optimization, since determining appropriate parameter values is trial-and-error process, more number of cases should be considered. In some researches, this kind of problem was solved by applying another algorithm such as genetic algorithm for searching appropriate parameter values [16, 17]. If these kinds of methodologies applied for selecting appropriate parameter values, overall regression quality is expected to be enhanced.

Another problem of this research is that there are only small numbers of control modes that have meaningful data. To collect the data for other control modes which were not originally coded in MAAP5 code, it is necessary to construct custom codes for every control modes. For simplicity, this paper did not cover un-coded control modes, but in real OVS application, OVS should has capability to cover every single valves and pumps in nuclear power plants. Since there are so many components in nuclear power plants, this work would requires a lot of time and labor. Thus, finding of proper common ground between simplification and performance should be achieved by further studies.

Main purpose of this study is to check that operation validation system could be developed by using support vector regression algorithm. Currently, this study involves many problems described above, but by conducting further studies, mentioned problems are expected to be solved.

REFERENCES

[1] Z. Yangping, Z. Bingquan, W. Dongxin, 'Application of genetic algorithms to fault diagnosis in nuclear power plants', Reliability engineering & system safety, vol. 67, pp. 153-160, 2000.

[2] C. M. Rocco S., E. Zio, 'A support vector machine integrated system for the classification of operation anomalies in nuclear components and systems', Reliability engineering & system safety, vol. 92., pp. 593-600, 2007.

[3] C. Cortes, V. Vapnik, 'Support-vector networks', Machine learning, vol. 20, pp. 273-297, 1995.

[4] S. R. Gunn, M. Brown, K. M. Bossley, 'Network performance assessment for neurofuzzy data modelling', In X. Liu, P. Cohen, M. Berthold, editors, Intelligent data analysis, vol. 1208 of Lecture notes in computer science, pp. 313-323, 1997.

[5] V. Vapnik, S. Golowich, A. Smola, 'Support vector method for function approximation, regression estimation and signal processing', In M. Mozer, M. Jordan, T. Petsche, editors, Advances in neural information processing systems, vol. 9, pp. 281-287, Cambridge, MA, 1997. MIT Press.

[6] 김동일, 조성준, 'ε-tube based pattern selection for support vector regression', 대한산업공학회, 2005

[7] O. L. Mangasarian, Nonlinear programming. McGraw-Hill, New York, 1969.

[8] R. J. Vanderbei, 'LOQO users manualversion 3.10', Technical report SOR-97-08, Princeton university, Statistics and operations research, 1997.

[9] W. Karush, 'Minima of functions of several variables with inequalities as side constraints', Master's thesis, Dept. of Mathematics, University of Chicago, 1939.

[10] H. W. Kuhn, A. W. Tucker, 'Nonlinear programming', 2nd Berkeley symposium on mathematical statistics and probabilistics, pp. 481-492, Berkeley, 1951.

[11] A. Farag, R. M. Mohamed, 'Regression using support vectgor machines: basic foundations', Technical report, University of Louisville, 2004.

[12] A. Smola, B. Scholkopf, 'A tutorial on support vector regression', Neuro-COLT technical report NC-TR-98-030, Royal Holloway college, University of London, UK, 2002.

[13] S. R. Gunn, 'Support vector machines for classification and regression', Technical report, University of Suthoampton, School of electronics and computer science, 1998.

[14] P. S. Yu, S. T. Chen, I. F. Chang, 'Support vector regression for real-time flood stage forecasting', Journal of hydrology, vol. 328, issues 3-4, pp. 704-716, 2006.

[15] A. Savitzky, M. J. E. Golay, 'Smoothing and differentiation of data by simplified least squares procedures', Analytical chemistry, vol. 36, no.8, pp. 1627-1639, 1964.

[16] N. Xin, X. Gu, H. Wu, Y. Hu, Z. Yang, 'Application of genetic algorithm-support vector regression (GA-SVR) for quantitative analysis of herbal medicines', Journal of chemometrics, vol. 26, issue 7, pp. 353-360, 2012.

[17] K. Chen, C. Wang, 'Support vector regression with genetic algorithms in forecasting tourism demand', Tourism management, vol. 28, issue 1, pp. 215-226, 2007.