ORIGINAL RESEARCH

Interpretable support vector regression

Tamás Kenesei, János Abonyi

Department Process Engineering, University of Pannonia, Veszprém, Hungary

Correspondence: Tamás Kenesei and János Abonyi. Address: Department Process Engineering, University of Pannonia, P.O. Box 158, H-8201, Veszprém, Hungary. E-mail: abonyij@fmt.uni-pannon.hu.

Abstract

This paper deals with transforming Support vector regression (SVR) models into fuzzy systems (FIS). It is highlighted that trained support vector based models can be used for the construction of fuzzy rule-based regression models. However, the transformed support vector model does not automatically result in an interpretable fuzzy model. Training of a support vector model results a complex rule base, where the number of rules are approximately 40-60% of the number of the training data, therefore reduction of the support vector model initialized fuzzy model is an essential task. For this purpose, a three-step reduction algorithm is used based on the combination of previously published model reduction techniques, namely the reduced set method to decrease number of kernel functions, then after the reduced support vector model is transformed into fuzzy rule base similarity measure based merging and orthogonal least-squares methods are utilized. The proposed approach is applied for nonlinear system identification, the identification of a Hammerstein system is used to demonstrate accuracy of the technique with fulfilling the criteria of interpretability.

Key words

Support vector regression, Fuzzy regression, Interpretability, Reduction, Visualization

1 Introduction

As universal approximator for any multivariate function; support vector machine model is widely-used to capture highly nonlinear, unknown or partially known complex systems plants or processes. However, it should be pointed out that simplicity and complexity of this type of models are challenging tasks to perform, since these black box systems don't give any insight to the model structure. Hence, incorporating robustness of support vector models with the transparency and interpretability of fuzzy models is a straightforward directional development step, confirmed by the fact that connection between support vector regression and fuzzy regression has been widely and extensively researched lately. Numerous techniques were reported to exploit the above listed synergies. Nowadays complexity of regression problems is increasingly growing, therefore it is necessary to deal with structural issues and linguistic interpretability in order to capture more information from nature of the modeled system, data or process.

Fuzzy logic helps to improve interpretability of knowledge based regression models through its semantics that provide insight in the model structure and decision making process. Application of support vector methods for the initialization of fuzzy models is not a completely new idea. Numerous methods have been proposed to build the connection between the SVR and the FIS. Chen and Wang [3, 4] propose a positive definite fuzzy system (PDFS). In the proposed fuzzy model, the

PDFS is equivalent to a Gaussian-kernel SVM [3] if Gaussian membership functions are adopted. Antecedent of a fuzzy rule is obtained by a support vector (SV). Therefore the number of fuzzy rules is the same as the number of SVs. As the number of SVs is generally large, the size of the FIS based on an SVM is also large. To solve this problem researcher [5] proposed a learning algorithm to remove the irrelevant fuzzy rules. In spite of this, the generalization performance is degraded. The above methods are for the zero order FIS, which has one fuzzy singleton in the consequent of a fuzzy rule. For the first order FIS, Leski [18] described a method for obtaining a FIS by means of the SVM with data-independent kernel matrix. Moreover Juang et al. used a combination of fuzzy clustering and the linear SVM to establish a fuzzy model with less parameter number and better generalization performance. However, negligible effort has been done to establish a HFIS (high order FIS) with kernel methods. It was presented a HFIS with high accuracy and good generalization performance [11]. It was shown how to obtain the formulation of the nonlinear function for the consequent part. Furthermore, Catala used prototype vectors to combine with the support vectors using geometric methods to define ellipsoids in the input space, which are later transformed to if then rules [2]. Special operator was utilized to achieve equivalency between support vector machines and fuzzy rule based system [6]. Utilization of support vector models is described to solve the convex optimization problem for multivariate linear regression models and it is also shown how multivariate fuzzy nonlinear regression model can be formalized for numerical inputs and fuzzy output [8]. Multiple types of kernels [6, 8] can be used to solve crisp nonlinear regression problems [1]. Juang and Hsieh [7] used a combination of fuzzy clustering and linear support vector regression to obtain Takagi-Sugeno type fuzzy rules. Support vector machines can be applied to determine the support vectors for each fuzzy cluster obtained by fuzzy c-means clustering algorithm [16]. Visualization of fuzzy regression models is also discussed lately. Interpretation of fuzzy regression is provided with an insight into regression intervals so that regression interval analysis, data type analysis and variable selections are analytically performed [15]. A visualization and interpretation tool is presented. Feature space is visualized with highlighting the corresponding variables in the original input data to show how they are associated to the output variable [17]. It is shown that which part of the input data can be utilized to estimate the output value. This technique also describes which input variable are responsible for the performance of the support vector regression. With the combination of visualization and interpretation the black-box support vector regression is identified in one step.

It must be taken into account that fuzzy logic does not guarantee interpretability as a prerequisite, because obtaining fuzzy models from support vector based training often result fuzzy models with high number of fuzzy rules. This phenomenon makes interpretability much more difficult, therefore aim of this paper is to describe a combination-of-tools three-step technique how to use reduction techniques on trained SVR models to acquire transparent, but accurate fuzzy rule based regression models. The steps are the following:

1) Application of the Reduced Set method

The identification of the SVM is followed by the application of the Reduced Set (RS) method to decrease the number of kernel functions. Originally, this method has been introduced [9] to reduce the computational complexity of SVMs. The obtained SVM is subsequently transformed into a fuzzy rule-based regression model.

2) Similarity-based fuzzy set merging

The Gaussian membership functions of the fuzzy rule based regression model are derived from the Gaussian kernel functions of the SVM. The interpretability of a fuzzy model highly depends on the distribution of the membership functions. Hence, the next reduction step is achieved by merging fuzzy sets based on a similarity measure [12].

3) Rule-base simplification by orthogonal transformation

Finally, orthogonal least-squares method is used to reduce the number of rules and re-estimate the consequent parameters of the regression model. The applications of orthogonal transformations for reducing the number of rules have received much attention in the recent literature [13, 14]. These methods evaluate the output contribution of the rules

to obtain the order of importance. The less important rules are then removed according this ranking to further complexity reduction and increase transparency.

This article is organized as follows. Firstly basic notations of support vector machines and the connection between the fuzzy regression are described. After detailed description of the three step reduction algorithm examples indicate the power and the usage of the described techniques on regression problems.

2 FIS interpreted SVR

SVM has been recently introduced for solving pattern recognition and function estimation problems. SVM is a nonlinear generalization of the Generalized Portrait algorithm developed in Russia in the 1960s. In its present form, the SVM was developed at AT&T Bell Laboratories by Vapnik and coworkers [19]. Due to this industrial context, SVM learning has now evolved into an active area of research. Moreover, the technique belongs to the standard methods toolbox of machine learning. The basic idea behind SVM is that with a kernel function: $k(x_i x_j)$, which for all data pairs $(x_1, \ldots, x_{N_d}) \subset \chi R_i^N$ give rise to positive matrices $K_{i,j} := k(x_i x_j) \times R_i^N$. Using k instead of dot product in R_i^N, this will correspond to map the data into a possibly high dimensional space F, by a usually nonlinear map $\phi : R_i^N \to F$ and take the dot product there

$$k\ (z_i; x)\ =\ (\phi\ (z_i)\ ;\phi\ (x)) \tag{1}$$

2.1 Support vector regression models

Suppose we have training data, $\{(x_1;y_1),\ldots(x_{Nd};y_{Nd})\} \subset \chi \times R_i^N$, where χ denotes the space of input patterns. Our goal is to find function $f(x)$ that has at most ε deviation from the targets with the obtained y_i, for all the training data. In other words we do not care about the errors as long as they are less than ε, but any larger deviation than ε won't be accepted. SVR can be formulated as follows:

$$\min_{w,b,\xi_i,\xi_i} \frac{1}{2}\left|\left|W\right|\right|^2 + C\sum_{i=l}^{N_d}(\xi_i + \xi_i^*)$$

$$s.t.\ yi - \mathrm{w}^\mathrm{T}\ \varPhi\ (x_i) - b\ \leq\ \varepsilon + \xi_i \tag{2}$$

$$w^T\ \varPhi\ (x_i) + b \leq\ \varepsilon + \xi_i^*$$

$$\xi_i\ ,\xi_i^* \geq 0$$

where ϕ is the feature mapping for kernel k, ε is the tolerance error, are ξ_i, ξ_i^* slack variables and C > 0 is a cost coefficient, which determines the trade-off between the model complexity and the degree of tolerance to the errors larger than ε. The dual form of the optimization problem eq. 2 becomes a quadratic programming (QP) problem:

$$\min_{\alpha,\alpha^*}\frac{1}{2}\sum_{i,j=1}^{N_d}(\alpha_i - \alpha_i^*)\ (\alpha_j - \alpha_j^*)\ k(x_i x_j)$$

$$+\varepsilon\sum_{i=1}^{N_d}(\alpha_i + \alpha_i^*) - \sum_{i=1}^{N_d}y_i\ (\alpha_i - \alpha_i^*) \tag{3}$$

$$s.t. \sum_{i=1}^{N_d} y_i \, (\alpha_i - \alpha_i^*) = 0 \; \alpha, \alpha^* \in [0, C]$$

where α and $\alpha*$ are the Lagrange multipliers. As an outcome of solving the QP problem eq. 3 can be rewritten to the following form:

$$f(x) = \sum_{i,j=1}^{N_d} (\alpha_i - \alpha_i^*) \, k(x_i, x) + b \tag{4}$$

Let $\gamma_i = \alpha_i - \alpha_i^*$ In case $\gamma_i \neq 0$ the corresponding training pattern xi can be noted as support vector.

2.2 Structure of the fuzzy rule-based regression model

To get a fuzzy rule-based regression model from the support vector regression model the following interpretation is needed:

$$y = \sum_{i=1}^{N_R} \beta_i(x) \, \delta_i + b \tag{5}$$

where β_i is the firing strength and δ_i is the rule consequent. The output of the regression model is calculated via this equation. In case of fuzzy systems, fuzzy rules can be formulated as follows

$$R_i if \; x_1 \; is \; A_{i1} \; and \; ... x_n \; is \; A_{in} \; then \; y_i = \delta_i, \qquad i = 1, ... N_R \tag{6}$$

where R_i is the i-th rule in the fuzzy rule-based regressor and N_R denotes the number of rules. $A_i, ..., A_{N_i}$ denote the antecedent fuzzy sets that define operating region of rule in the N_i dimensional input space. The rule consequent δ_i is a crisp number. The connective is modeled by the product operator. Hence the degree of activation of the ith rule is calculated as

$$\beta_i(x) = \prod_{j=i}^{N_i} A_{ij} \left(x_j \right), \qquad i = 1, ..., N_R \tag{7}$$

Main principle of kernel-based support vector regressors is the identification of a linear decision boundary in this high dimensional feature space. The link to the fuzzy model structure is the following: the fuzzy sets are represented in this paper by Gaussian membership functions

$$A_{ij}(x_j) = \exp\left(\frac{||x_j - z_{ij}||^2}{2\sigma^2}\right) \tag{8}$$

The degree of fulfillment $\beta_i(x)$ can be written through eq. 7-8 in a more compact form by using Gassuian kernels.

$$\beta_i(x) = \exp\left(\frac{||x - z_j||^2}{2\sigma^2}\right) \tag{9}$$

This kernel interpretation of fuzzy systems shows that fuzzy models are effective in solving nonlinear problems because they map the original input space into a nonlinear feature space by using membership functions similarly to the support vector machine that utilize kernel functions for this purpose.

3 Ensuring interpretability with three step algorithm

In the previous sections, it has been shown how a SVM, that is structurally equivalent to a fuzzy model, can be identified. Unfortunately, this identification method cannot be used directly for the identification of interpretable fuzz systems because the number of the support vectors is usually very large. Typical values are 40-60% of the number of training data which is in our approach equal to the number of rules in the fuzzy system. Therefore, there is a need for an interpretable approximation of the support vector expansion. For this purpose a step-wise algorithm will be introduced, where the first step is based on the recently published Reduced Set (RS) method developed for reducing the computational demand of the evaluation of SVMs [9].

3.1 Model simplification by reduced set method

The aim of the RS method is to approximate the highdimensional feature space given by the support vectors

$$\Psi = \sum_{i=1}^{N_x} \gamma_i \phi(x_i) \tag{10}$$

by a reduced set expansion

$$\Psi' = \sum_{i=1}^{N_R} \delta_i \phi(z_i) \tag{11}$$

with $N_R < N_x < N_d$, where N_x denotes the number of support vectors (the number of x_i vectors for those $\gamma_i \neq 0$) and NR represents the number of the desired rules in the fuzzy rule based regressor that we would like to identify and z_i denotes the centers of the new kernel functions that are not necessarily training samples. NR should be as small as possible because it determines the number of fuzzy rules. In practice it turns out that the RS method is often able to deliver a one-tenth reduction, so NR can be chosen as $N_R = N_x/10$. For this model reduction, the squared error $\|\Psi - \Psi'\|/2$ has to be minimized. For this purpose, the 'kernel trick' has to be applied because ϕ is not given explicitly

$$\left\|\Psi - \Psi'\right\|^2 = \sum_{ij=1}^{N_x} \gamma_i \gamma_j k(x_i x_j) + \sum_{ij=1}^{N_R} \delta_i \delta_j (z_i z_j) - 2 \sum_{i=1}^{N_x} \sum_{j=1}^{N_R} \gamma_i \delta_j k(x_i, z_j) \tag{12}$$

The cost function 12 is minimized in a step-wise manner while the feature space is approximated by the following iterative algorithm:

Repeat for $m : 2; \ldots \ldots N_R$

- Step 1: Obtain the residual space

    Let Ψ_m mean the residual of the feature space approximation generated at the (m-1)–th step

$$\Psi_m = \sum_{i=1}^{N_x} \gamma_i \phi(x_i) - \sum_{i=1}^{m-1} \delta_i \phi(z_i)$$

$$= \sum_{i=1}^{N_m} \varepsilon_i \phi(v_i) \tag{13}$$

where

$(\varepsilon_1,\ldots,\varepsilon_{N_m})=(\gamma_1,\ldots,\gamma_{N_x},-\delta_1,\ldots,-\delta_{m-1}),$

$(v_1,\ldots,v_{N_m})=(x_1,\ldots x_{N_x},z_1,\ldots z_{m-1})$

$N_m=N_x+m-1$

- Step 2: Inner iteration step for determining z_m

This residual function is approximated by the determination of z_m and δ_m in the iterative procedure, where the following cost function has to be minimized

$$\min_{\delta_m,z_m} \|\Psi_m - \delta_m \phi(z_m)\|^2 \tag{14}$$

This can be done by standard techniques or using fixed point iteration [9].

$$Z_m^{n+1} = \frac{\sum_{i=1}^{N_m} \varepsilon_i \exp\left(-\|v_i - z_m^n\|^2/(2\sigma^2)\right)v_i}{\sum_{i=1}^{N_m} \varepsilon_i \exp\left(-\|v_i - z_m^n\|^2(2\sigma^2)\right)} \tag{15}$$

where the superscript n denotes the n–th inner iteration step; eq. 15 is iterated till it converges to $\|z_m^{n+1} - z_m^n\|^2 < \varepsilon$. Interestingly, eq. 15 can be interpreted in the context of clustering [10]. It determines the center of a single Gaussian cluster, trying to capture as many of the v_i with positive δ_i as possible, and simultaneously avoiding those v_i with negative δ_i.

- Step 3: Least-squares estimation of the δ_i coefficients

The δ_m coefficient is calculated by recalculating the whole $\delta = [\delta_1, \ldots, \delta_m]^T$ vector by minimizing eq. 12

$$\delta = (K^z)^{-1} K^{zx} \gamma \tag{16}$$

where the element of the matrices are expressed by the kernel functions $K_{ij}^z(z_i, z_j)$ and $K_{ij}^{zx} = k(z_i, x_j)$.

3.2 Reduction of the number of fuzzy sets

In the previous section, it has been shown how kernel-based regression model with a given number of kernel functions N_R, can be obtained. Because the number of the rules in the transformed fuzzy system is identical to the number of kernels, it is extremely important to get a moderate number of kernels in order to obtain a compact fuzzy rule-based regression model. From eq. 9 it can be seen that the number of fuzzy sets in the identified model is $N_s = N_R N_i$. The interpretability of a fuzzy model highly depends on the distribution of these membership functions. With the simple use of eq. 8, some of the membership functions may appear almost undistinguishable. Merging similar fuzzy sets reduces the number of linguistic terms used in the model and thereby increases the transparency of the model. This reduction is achieved by a rule-base simplification method [12, 20] based on a similarity measure $S(A_{ij}, A_{kj})$ and i≠j. If $S(A_{ij}, A_{kj})$, then the two membership functions A_{ij} and A_{kj} are equal. $S(A_{ij}, A_{kj})$ becomes 0 when the membership functions are non-overlapping. During the rule-base simplification procedure similar fuzzy sets are merged when their similarity exceeds a user-defined threshold θ $\in [0;1]$. The set-similarity measure can be based on the set theoretic operations of intersection and union [12].

$$S\left(A_{ij}, A_{kj}\right) = \frac{\left|A_{ij} \cap A_{kj}\right|}{\left|A_{ij} \cup A_{kj}\right|} \tag{17}$$

where |.| denotes the cardinality of a set, and the $\cap$ and $\cup$ operators represent the intersection and union, respectively, or it can be based on the distance of the two fuzzy sets. Here, the following expression was used to approximate the similarity between two Gaussian fuzzy sets [20].

$$S\left(A_{ij}, A_{kj}\right) = \frac{1}{1 + d\left(A_{ij}, A_{kj}\right)}$$

$$S\left(A_{ij}, A_{kj}\right) = \frac{1}{1 + \sqrt{\left(z_{ij} - z_{kj}\right)^2 \left(\sigma_{ij} - \sigma_{kj}\right)^2}} \tag{18}$$

3.3 Reduction of the number of rules by orthogonal transforms

By using the previously presented SVM identification and reduction techniques, the following fuzzy rule-based regression model has been identified

$$y = \sum_{i=1}^{N_R} \prod_{j=1}^{N_i} \frac{\left(x_j - z_{ij}\right)^2}{2\sigma^2} \delta_i + b \tag{19}$$

Because the application of the RS method and the fuzzy set merging procedure the obtained membership functions only approximate the original feature space identified by the SVM. Hence, the $\delta = \left[\delta_1, \dots, \delta_{N_R}\right]^T$ consequent parameters of the rules have to be re-identified to minimize the difference between the decision function of the support vector machine eq. 4 and the fuzzy model eq. 19.

$$MSE = \sum_{j=1}^{N_d} \left(\sum_{i=1}^{N_x} \gamma_i k\left(x_j, x_i\right) - \sum_{i=1}^{N_R} \delta_i \beta_i\left(x_j\right)\right)^2 \tag{20}$$

$$= \|y_s - B\delta\|^2$$

where the matrix $B = \left[b_1, \dots, b_{N_R}\right] \in R^{N_d \times N_R}$ contains the firing strength of all NR rules for all the inputs x_i, where $b_j = \left[\beta_j(x_1), \dots, \beta_j(x_{N_d})\right]^T$. As the fuzzy rule-based regression model in eq. 19 is linear in the parameters δ, eq. 20 can be solved by a least-squares method

$$\delta = B^+ y_s \tag{21}$$

where B^+ denotes the Moore-Penrose pseudo inverse of B. The application of orthogonal transforms for the above mentioned regression problem eq. 20 for reducing the number of rules has received much attention in recent literature [13, 14]. These methods evaluate the output contribution of the rules to obtain an importance ordering. For modeling purposes, the Orthogonal Least Squares (OLS) is the most appropriate tool [13]. The OLS method transforms the columns of B into a set of orthogonal basis vectors in order to inspect the individual contribution of each rule. To do this, Gram-Schmidt orthogonalization of B=WA is used, where W is an orthogonal matrix WTW = I and A is an upper triangular matrix with unity diagonal elements. If w_i denotes the i–th column of W and g_i is the corresponding element of the OLS solution vector g = Aδ, the output variance $y^T s$ y_s=N_d can be explained by the regressors

$\sum_{i=1}^{N_r} g_i w_i^T w_i / N_d$. Thus, the error reduction ratio, ρ, due to an individual rule i can be expressed as

$$p^i = \frac{g_i^2 w_i^T w_i}{y_s^T y_s} \tag{22}$$

This ratio offers a simple mean for ordering the rules, and can be easily used to select a subset of rules in a forward regression manner. Evaluating only the approximation capabilities of the rules, the OLS method often assigns high importance to a set of redundant or correlated rules. To avoid this, some extension for the OLS method were proposed [14, 21].

4 Application examples

4.1 Illustrative example

To demonstrate the potential of Support Vector Regression techniques two examples were introduced. Firstly an illustrative regression problem is solved with a simple dataset containing 51 samples (see Figure 1). The SVR technique obtained 14 support vectors. This model has been reduced by the RS method (Step 1), by which we tried to reduce the model to operate with 10 rules. Modeling results can be seen in Table 1. Utilization of all the three steps of the algorithm reduced number of fuzzy rules to 6, however this indicated slight increase in modeling error.
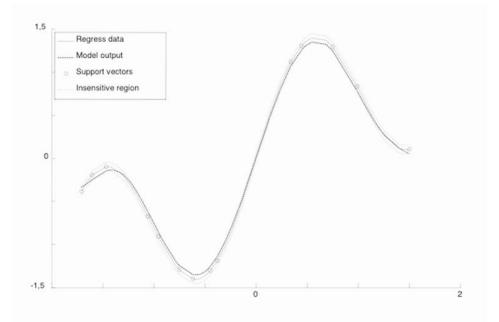

Figure 1. Illustrative example with model output, support vectors and the insensitive region

Table 1. Application of the proposed three step algorithm

Method	RMSE	#Rules
SVR Identification	0.084	14
Step 1. Reduction	0.0919	10
Step 2. Reduction	0.2415	9
Step 3. Reduction	0.3361	8

4.2 Identification of a hammerstein system

In this example, the support vector regression is used to approximate a Hammerstein system that consists of a series connection of a memory less nonlinearity, f, and linear dynamics, G, as it is shown in Fig. 2., where v represents the transformed input variable. For transparent representation the Hammerstein system consist of a first-order linear part $y(k + 1) = 0,9y(k) + 0,1v(k)$ and a static nonlinearity is represented by a polynomial, $v(k) = u(k)^2$. The dataset contains 500 input-output data. Support vector regression model was identified with efficiency summarized in Table 2.
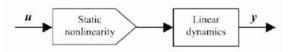

Figure 2. Hammersten system

Table 2. Results of Hammerstein system identification

Method	RMSE	#Rules
SVR Identification	0-0533	22
Step 1. Reduction	0.0604	15
Step 2. Reduction	0.0.65	13
Step 3. Reduction	0.0792	12

As figure 3 and Table 2 conclude, support vector regression is able to give accurate models for Hammerstein system identification. Extracted, non-distinguishable rules from this system are represented on Figure 4, therefore the three-step reduction algorithm is used to acquire interpretable models.
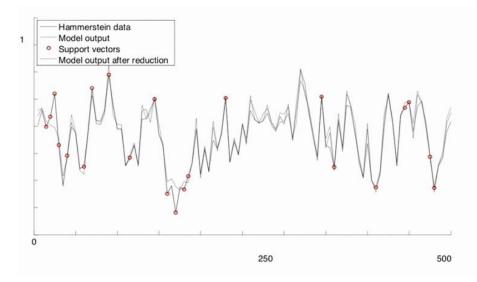

Figure 3. Identified Hammerstein System, support vectors and model output after reduction

Number of rules could be reduced to 15 without any major modeling error increase after applying the RS method (Step 1), Using further reductions with the second and third step of the proposed algorithm interpretable model (see Figure 5) and accurate model could be extracted.
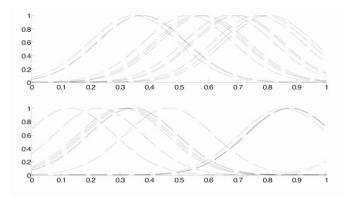
Figure 4. Non-distinguishable membership functions obtained after the application of RS method
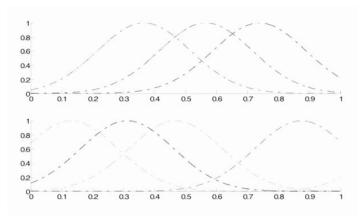

Figure 5. Interpretable membership functions of the reduced fuzzy model

5 Conclusions

Support vector based techniques and fuzzy rule-based models work in a similar manner as both models maps the input space of the problem into a feature space with the use of either nonlinear kernel or membership functions. The main difference between support vector based and fuzzy rule-based systems is that fuzzy systems have to fulfill two objectives simultaneously, i.e., they must provide a good modeling performance and must also be linguistically interpretable, which is not an issue for support vector systems. However, as the structure identification of fuzzy systems is a challenging task, the application of kernel-based methods for model initialization could be advantageous because of the high performance and the good generalization properties of this type of models.

Accordingly, support vector based initialization of fuzzy rule-based model is used in this paper. First, the initial fuzzy model is derived by means of the support vector learning algorithm. Then the support vector model is transformed into an initial fuzzy model that is subsequently reduced by means of the reduced set method, similarity based fuzzy set merging, and orthogonal transform-based rule-reduction. Because these rule base simplification steps do not utilize any nonlinear optimization tools, it is computationally cheap and easy to implement them.

The application of the proposed approach was shown on simple one dimensional function identification data and Hammerstein system identification. The obtained models are very compact but their accuracy is still adequate. Besides, it might be clear that still real progress can be made in the development of novel methods for feature selection.

We intend this paper also as a case study for further developments in the direction of a combination-of-tools methodology for modeling and identification. We are seeking for techniques that perform well on multiple criteria, considering here different soft-computing tools, namely support vector machines and fuzzy techniques are combined to achieve a predefined trade-off between performance and transparency.

Acknowledgement

References

[1] S. Gunn. Support Vector Machines for Classification and Regression, ISIS Tech. Report, University of Southampton. 1998.

[2] H. Nunez, C. Angulo, A. Catala. Rule extraction from support vector machines, European Symposium on Artificial Neural Networks Proceedings. 2002; 107-112.

[3] Y.X. Chen, J.Z. Wang. Support vector learning for fuzzy rule-based classification systems, IEEE Trans. Fuzzy Syst. 2003; 11(6): 716-728. http://dx.doi.org/10.1109/TFUZZ.2003.819843

[4] Y.X. Chen, J.Z. Wang. Kernel machines and additive fuzzy systems: classification and function approximation, Proceedings of IEEE International Conference on Fuzzy Systems, St. Louis, Mo. 2003; 789-795.

[5] J.H. Chiang, P.Y. Hao. Support vector learning mechanism for fuzzy rulebased modeling: a new approach, IEEE trans. Fuzzy Syst. 2004; 12 (1): 1-12. http://dx.doi.org/10.1109/TFUZZ.2003.817839

[6] J. L. Castro, L. D. Flores-Hidalgo, C. J. Mantas, J. M. Puche . Extraction of fuzzy rules from support vector machines, Fuzzy Sets and Systems. 2007; 158 (18): 2057-2077. http://dx.doi.org/10.1016/j.fss.2007.04.014

[7] Chia-Feng Juang, Cheng-Da Hsieh, TS-fuzzy system-based support vector regression, Fuzzy Sets and Systems. 2009; 160(17): 2486-2504. http://dx.doi.org/10.1016/j.fss.2008.11.022

[8] Dug Hun Hong, Changha Hwang. Support vector fuzzy regression machines, Fuzzy Sets and Systems. 2003; 138 (2): 271-281. http://dx.doi.org/10.1016/S0165-0114(02)00514-6

[9] B. Schölkopf, S. Mika, C. J. C. Burges, P. Knirsch, K.-R. Mller, G. Ratsch, A. Smola. Input space vs. feature space in kernel-based methods, IEEE Trans. on Neural Networks. 1999; 10(5): 1000-1017. PMid:18252603 http://dx.doi.org/10.1109/72.788641

[10] B. Schölkopf and P. Knirsch and A. Smola and C. Burges. Fast approximation of support vector kernel expansion, and an interpretation of clustering as approximation in feature spaces, DAGM-Symposium, Informatik aktuell. 1998; 124-132.

[11] Q. Cai, Z. Haom X. Yang. Gaussian kernel-based fuzzy inference system for high dimensional regression, Neuorocomputing. 2012; 77(1): 197-204. http://dx.doi.org/10.1016/j.neucom.2011.09.005

[12] M. Setnes, R. Babuska, U. Kaymak, H. R. van Nauta Lemke. Similarity measures in fuzzy rule base simplification, IEEE Trans. SMC-B. 2000; 28: 376-386

[13] J. Yen and L. Wang. Simplifying fuzzy rule-based models using orthogonal transformation methods, IEEE Trans. SMC-B. 1999; 29: 13-24.

[14] M. Setnes and H. Hellendoorn. Orthogonal transforms for ordering and reduction of fuzzy rules, in FUZZ-IEEE. 2000; 700-705.

[15] Hsiao-Fan Wang, Ruey-Chyn Tsaur. Insight of a fuzzy regression model, Fuzzy Sets and Systems. 2000; 112 (3): 355- 369. http://dx.doi.org/10.1016/S0165-0114(97)00375-8

[16] Asli Celikyilmaz, I. Burhan Trksen. Fuzzy functions with support vector machines, Information Sciences. 2007; 177 (23): 5163-5177. http://dx.doi.org/10.1016/j.ins.2007.06.022

[17] B. ÜStün, W.J. Melssen, L.M.C. Buydens. Visualisation and interpretation of Support Vector Regression models, Analytica Chimica Acta. 2007; 595 (1-2): 299-309. PMid:17606013 http://dx.doi.org/10.1016/j.aca.2007.03.023

[18] Jacek M. Leski. On support vector regression machines with linguistic interpretation of the kernel matrix, Fuzzy Sets and Systems. 2006; 157 (8): 1092-1113. http://dx.doi.org/10.1016/j.fss.2005.09.012

[19] Corrina Cortes and Vladimir Vapnik. Support-Vector Networks, AT&T Research Labs, USA. 1995.

[20] Y. Jin. Fuzzy Modeling of High-Dimensional Systems, IEEE Transactions on Fuzzy Systems. 2000; 8: 212-221. http://dx.doi.org/10.1109/91.842154

[21] M. Setnes and R. Babuska. Rule base reduction: some comments on the use of orthogonal transforms, Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on Fuzzy Systems. 2001; 31(2): 199-206.