Communications of the Korean Statistical Society
2010, Vol. 17, No. 2, 141–151

Hybrid Fuzzy Least Squares Support Vector Machine Regression for Crisp Input and Fuzzy Output

Jooyong Shim[a], Kyung Ha Seok[b], Changha Hwang[1,c]

[a]Department of Applied Statistics, Catholic University of Daegu
[b]Department of Data Science, Institute of Statistical Information, Inje University
[c]Department of Statistics, Dankook University

Abstract

Hybrid fuzzy regression analysis is used for integrating randomness and fuzziness into a regression model. Least squares support vector machine(LS-SVM) has been very successful in pattern recognition and function estimation problems for crisp data. This paper proposes a new method to evaluate hybrid fuzzy linear and nonlinear regression models with crisp inputs and fuzzy output using weighted fuzzy arithmetic(WFA) and LS-SVM. LS-SVM allows us to perform fuzzy nonlinear regression analysis by constructing a fuzzy linear regression function in a high dimensional feature space. The proposed method is not computationally expensive since its solution is obtained from a simple linear equation system. In particular, this method is a very attractive approach to modeling nonlinear data, and is nonparametric method in the sense that we do not have to assume the underlying model function for fuzzy nonlinear regression model with crisp inputs and fuzzy output. Experimental results are then presented which indicate the performance of this method.

Keywords: Fuzzy regression, hybrid regression, least squares support vector machine, nonlinear, weighted fuzzy arithmetic.

1. Introduction

Uncertainties pervade the analysis and modeling of data. There are usually two types of uncertainty. The first is randomness and the second is fuzziness. The theories of probability and statistics are used in dealing with randomness, whereas the theory of fuzzy sets are used in dealing with fuzziness. In many real applications, both randomness and fuzziness appear simultaneously in a system, and need to be considered in regression analysis. Fuzzy regression analysis handles fuzzy data which represents fuzziness. Hybrid regression analysis integrates both randomness and fuzziness into a regression model. This hybrid concept is discussed in this paper. The methods of fuzzy regression and hybrid regression analysis are used to develop prediction models based on fuzzy data. A detailed discussion of fuzzy regression is provided in Tanaka (1987), Tanaka *et al.* (1982) and Tanaka and Watada (1988). A major difference between fuzzy regression and statistical regression is in dealing with errors. Fuzzy regression deals with errors as fuzzy variables, whereas statistical regression deals with errors as random residuals. It has been noted that fuzzy regression could be more effective than statistical regression when the degree of fuzziness of systems is high. Chang (2001) proposed a hybrid fuzzy least squares regression approach. The method uses a new definition of weighted fuzzy arithmetic(WFA) and the well-known least squares criterion.

[1] Corresponding author: Professor, Department of Statistics, Dankook University, Gyeonggido 448-160, Korea.
E-mail: chwang@dankook.ac.kr

Support vector machine(SVM) has been very successful in pattern recognition and function estimation problems for crisp data. See for details Gunn (1998), Smola and Schoelkopf (1998) and Vapnik (1998). This paper focuses on a least squares support vector machine(LS-SVM) proposed by Suykens (2001), which is a reformulation to standard SVM. In this paper we propose a new method to evaluate hybrid fuzzy linear and nonlinear regression models with crisp inputs and fuzzy output using the definition of WFA and the principle of LS-SVM. The proposed method is computationally cheap since its solution is obtained by solving a simple linear equation system. LS-SVM allows us to perform fuzzy nonlinear regression analysis by constructing a fuzzy linear regression function in a high dimensional feature space. The proposed method here is a very attractive approach to evaluating hybrid fuzzy nonlinear regression model with crisp inputs and fuzzy output, and is model-free method in the sense that we do not have to assume the underlying model function for hybrid fuzzy nonlinear regression. This model-free method turns out to be a promising method which has been attempted to treat hybrid fuzzy nonlinear regression model.

The rest of this paper is organized as follows. In Section 2, we review some notions and definitions of WFA. In Section 3, we propose hybrid fuzzy LS-SVM regression method based on hybrid fuzzy least squares regression approach of Chang (2001). In Section 4, we describe some parameter selection methods. In Section 5, we illustrate our approach through examples. Finally, we conclude in Section 6.

2. Weighted Fuzzy Arithmetic

When conventional fuzzy arithmetic is used to deal with the regression problems, a large number of arithmetic operations needs to be involved and the fuzzy widths could add up to an unrealistic large number. Other problems with conventional fuzzy subtraction and conventional fuzzy division have been identified by Chao and Ayyub (1996). To make up these drawbacks of the conventional fuzzy arithmetic, Chang (2001) proposed a new definition of WFA. This WFA defines the arithmetic operations between two fuzzy numbers as operating two corresponding values in each fuzzy set at the same membership level, integrating each level operation weighted by the membership level for the entire fuzzy sets, and dividing the weighted integration by the total integral of the membership function. It has been noted that the WFA uses the concept of defuzzification to convert the operation of fuzzy sets into a crisp real number.

In this section we review well-known notions and definitions of WFA. Let $\tilde{A}$ and $\tilde{B}$ be triangular fuzzy numbers that can be expressed as (a, l_a, r_a) and (b, l_b, r_b), respectively. At μ membership level, the intervals of $\tilde{A}$ and $\tilde{B}$ can be expressed as

$$\tilde{A}^\mu = \left[A_L^\mu, A_R^\mu\right] = [a-(1-\mu)l_a, a+(1-\mu)r_a] \tag{2.1}$$

$$\tilde{B}^\mu = \left[B_L^\mu, B_R^\mu\right] = [b-(1-\mu)l_b, b+(1-\mu)r_b]\,. \tag{2.2}$$

According to the definition of WFA, the weighted fuzzy addition of $\tilde{A}$ and $\tilde{B}$ is then defined by

$$\tilde{A}+\tilde{B} = \left[\int_\mu \left(A_L^\mu + B_L^\mu\right)\mu d\mu\right]_L + \left[\int_\mu \left(A_R^\mu + B_R^\mu\right)\mu d\mu\right]_R. \tag{2.3}$$

Substituting the formulas for A_L^μ, A_R^μ, B_L^μ and B_R^μ into the weighted integrations in the Equation (2.3) yields the following integraions:

$$\left[\int_\mu \left(A_L^\mu + B_L^\mu\right)\mu d\mu\right]_L = \int_0^1 \{[a-(1-\mu)l_a]+[b-(1-\mu)l_b]\}\mu d\mu = \frac{1}{2}(a+b) - \frac{1}{6}(l_a+l_b) \tag{2.4}$$

and

$$\left[\int_{\mu}\left(A_R^{\mu}+B_R^{\mu}\right)\mu d\mu\right]_R = \int_0^1 \{[a+(1-\mu)r_a]+[b+(1-\mu)r_b]\}\mu d\mu = \frac{1}{2}(a+b)+\frac{1}{6}(r_a+r_b). \quad (2.5)$$

Thus, the sum of Equations (2.4) and (2.5) yields

$$\tilde{A}+\tilde{B}=(a+b)+\frac{1}{6}[(r_a+r_b)-(l_a+l_b)]. \quad (2.6)$$

When both $\tilde{A}$ and $\tilde{B}$ are symmetric fuzzy numbers, *i.e.*, $r_a = l_a$ and $r_b = l_b$, a special case of the Equation (2.6) can be obtained as

$$\tilde{A}+\tilde{B}=a+b. \quad (2.7)$$

However, $\tilde{A}+\tilde{B}=\tilde{A}+\tilde{C}$ does not imply that $\tilde{B}=\tilde{C}$. Similarly, weighted fuzzy subtraction, weighted fuzzy multiplication and weighted fuzzy division can be derived as follows:

$$\tilde{A}-\tilde{B} = (a-b)+\frac{1}{6}[(r_a-r_b)-(l_a-l_b)] \quad (2.8)$$

$$\tilde{A}\tilde{B} = (ab)+\frac{1}{6}[(br_a+ar_b)-(bl_a+al_b)]+\frac{1}{12}(l_al_b+r_ar_b) \quad (2.9)$$

$$\tilde{A}/\tilde{B} = \int_0^1 \frac{[a-(1-\mu)l_a]}{[b-(1-\mu)l_b]}\mu d\mu + \int_0^1 \frac{[a+(1-\mu)r_a]}{[b+(1-\mu)r_b]}\mu d\mu. \quad (2.10)$$

We notice that when arithmetic operations involve a fuzzy number and a crisp number, weighted fuzzy arithmetic becomes ordinary arithmetic between the fuzzy center value of the fuzzy number and the crisp number.

3. Hybrid Fuzzy LS-SVM Regression

Chang (2001) proposed a hybrid fuzzy least squares regression approach using the definition of WFA in order to integrate both randomness and fuzziness into a regression model. In this section, we propose hybrid fuzzy LS-SVM regression to evaluate fuzzy linear and nonlinear regression models with crisp inputs and fuzzy output data using the above definition of WFA and the principle of LS-SVM.

3.1. Fuzzy least squares criterion

In this subsection, we briefly describe the fuzzy least squares criterion of Chang (2001) and then reexpress it for our purpose. The definition of WFA is used to formulate the summation of the squares of errors between the predicted and observed values.

Suppose that we are given training data $\{(\boldsymbol{x}_1,\tilde{Y}_1),\ldots,(\boldsymbol{x}_n,\tilde{Y}_n)\}\subset R^d\times T(R)$. Here, $T(R)$ represents the space of all triangular fuzzy numbers. Each observed value $\tilde{Y}_i$ is expressed as $\tilde{Y}_i=(y_i,e_{i,L},e_{i,R})$. Let x_{ij} be element of $\boldsymbol{x}_i$. Then, we assume $x_{ij}\geq 0$ by simple translation of all vectors. From now on we use input vector $\boldsymbol{x}=(1,x_1,\ldots,x_d)^t$ instead of $\boldsymbol{x}=(x_1,\ldots,x_d)^t$ for our purpose. Here, the superscript t denotes the transpose of matrix. For pedagogical reasons, we begin by describing the case of fuzzy linear regression function $\hat{Y}_i$, taking the form

$$\begin{aligned}\hat{Y}_i &= \tilde{A}_0+\tilde{A}_1x_{1i}+\cdots+\tilde{A}_dx_{di}\\ &= (a_0,l_{a_0},r_{a_0})+(a_1,l_{a_1},r_{a_1})x_{1i}+\cdots+(a_d,l_{a_d},r_{a_d})x_{di} \quad (3.1)\\ &= \left(\boldsymbol{a}^t\boldsymbol{x}_i,\boldsymbol{l}_a^t\boldsymbol{x}_i,\boldsymbol{r}_a^t\boldsymbol{x}_i\right),\end{aligned}$$

where $\boldsymbol{a} = (a_0, a_1, \ldots, a_d)^t, \boldsymbol{l}_a = (l_{a_0}, l_{a_1}, \ldots, l_{a_d})^t, \boldsymbol{r}_a = (r_{a_0}, r_{a_1}, \ldots, r_{a_d})^t$.

The principle of the least squares technique is to minimize the sum of the squares of residual errors. Using the definition of WFA, we formulate the sum of the squares of residual errors between the predicted values $\hat{Y}_i$ and the observed values $\tilde{Y}_i$ as follows:

$$
\begin{aligned}
E &= \sum_{i=1}^{n} \left(\hat{Y}_i - \tilde{Y}_i\right)^2 \\
&= \sum_{i=1}^{n} \left[\int_0^1 \left(\hat{Y}_{i,L}^{\mu} - \tilde{Y}_{i,L}^{\mu}\right)^2 \mu d\mu\right]_L + \left[\int_0^1 \left(\hat{Y}_{i,R}^{\mu} - \tilde{Y}_{i,R}^{\mu}\right)^2 \mu d\mu\right]_R \\
&= \sum_{i=1}^{n} \int_0^1 \left[\left(\boldsymbol{a}^t\boldsymbol{x}_i - y_i\right) - (1-\mu)\left(\boldsymbol{l}_a^t\boldsymbol{x}_i - e_{i,L}\right)\right]^2 \mu d\mu \\
&\quad + \sum_{i=1}^{n} \int_0^1 \left[\left(\boldsymbol{a}^t\boldsymbol{x}_i - y_i\right) - (1-\mu)\left(\boldsymbol{r}_a^t\boldsymbol{x}_i - e_{i,R}\right)\right]^2 \mu d\mu \\
&= \sum_{i=1}^{n} \left[\frac{1}{2}\left(\boldsymbol{a}^t\boldsymbol{x}_i - y_i\right)^2 - \frac{1}{3}\left(\boldsymbol{a}^t\boldsymbol{x}_i - y_i\right)\left(\boldsymbol{l}_a^t\boldsymbol{x}_i - e_{i,L}\right) + \frac{1}{12}\left(\boldsymbol{l}_a^t\boldsymbol{x}_i - e_{i,L}\right)^2\right] \\
&\quad + \sum_{i=1}^{n} \left[\frac{1}{2}\left(\boldsymbol{a}^t\boldsymbol{x}_i - y_i\right)^2 + \frac{1}{3}\left(\boldsymbol{a}^t\boldsymbol{x}_i - y_i\right)\left(\boldsymbol{r}_a^t\boldsymbol{x}_i - e_{i,R}\right) + \frac{1}{12}\left(\boldsymbol{r}_a^t\boldsymbol{x}_i - e_{i,R}\right)^2\right] \\
&= \sum_{i=1}^{n} \left(\boldsymbol{a}^t\boldsymbol{x}_i - y_i\right)^2 + \frac{1}{3}\sum_{i=1}^{n} \left(\boldsymbol{a}^t\boldsymbol{x}_i - y_i\right)\left[\left(\boldsymbol{l}_a^t\boldsymbol{x}_i - e_{i,R}\right) - \left(\boldsymbol{l}_a^t\boldsymbol{x}_i - e_{i,R}\right)\right] \\
&\quad + \frac{1}{12}\sum_{i=1}^{n} \left[\left(\boldsymbol{l}_a^t\boldsymbol{x}_i - e_{i,R}\right)^2 + \left(\boldsymbol{l}_a^t\boldsymbol{x}_i - e_{i,L}\right)^2\right].
\end{aligned}
\tag{3.2}
$$

With a view to derive hybrid fuzzy LS-SVM we rewrite E as follows:

$$
E = \frac{1}{3}\sum_{i=1}^{n}\left(y_i - \boldsymbol{a}^t\boldsymbol{x}_i\right)^2 + \frac{1}{3}\sum_{i=1}^{n}\left(y_i - \frac{1}{2}e_{i,L} - \boldsymbol{a}^t\boldsymbol{x}_i + \frac{1}{2}\boldsymbol{l}_a^t\boldsymbol{x}_i\right)^2 + \frac{1}{3}\sum_{i=1}^{n}\left(y_i + \frac{1}{2}e_{i,R} - \boldsymbol{a}^t\boldsymbol{x}_i - \frac{1}{2}\boldsymbol{r}_a^t\boldsymbol{x}_i\right)^2.
$$

We notice that the rewritten E is partitioned into three equally weighted sums of the squares of errors so that LS-SVM can be appropriately applied.

3.2. Hybrid fuzzy LS-SVM regression

We now propose hybrid fuzzy LS-SVM to evaluate fuzzy linear and nonlinear regression models with crisp input and fuzzy output by applying LS-SVM to the preceding fuzzy least squares criterion. In the case of noisy learning data, the use of traditional neural network, often leads to poor generalization and overfitting. The SVM, whose foundations have been established by Vapnik, has been designed to overcome these problems. Hong and Hwang (2003, 2005), Hwang *et al.* (2005, 2006) and Shim *et al.* (2009) proposed the use of SVM for fuzzy and interval regression analysis. The standard SVM uses ϵ-insensitive loss function for function estimation. In this paper, we consider LS-SVM which uses quadratic loss function and is simpler and faster than the standard SVM.

Using the new sum of the squares of errors of the Equation (3.3) the objective function without penalty terms can be considered as follows:

$$\text{minimize} \sum_{i=1}^{n} \left(e_{1i}^2 + e_{2i}^2 + e_{3i}^2\right) \tag{3.3}$$

$$\text{subject to} \begin{cases} e_{1i} = y_i - \boldsymbol{a}^t\boldsymbol{x}_i, & i = 1,\ldots,n, \\ e_{2i} = y_i - \frac{1}{2}e_{i,L} - \boldsymbol{a}^t\boldsymbol{x}_i + \frac{1}{2}\boldsymbol{l}_a^t\boldsymbol{x}_i, & i = 1,\ldots,n, \\ e_{3i} = y_i + \frac{1}{2}e_{i,R} - \boldsymbol{a}^t\boldsymbol{x}_i - \frac{1}{2}\boldsymbol{r}_a^t\boldsymbol{x}_i, & i = 1,\ldots,n. \end{cases}$$

We add the penalty terms and penalty parameter γ on the objective function above to construct hybrid fuzzy LS-SVM formulation as follows:

$$\text{minimize} \ \frac{1}{2}\boldsymbol{a}^t\boldsymbol{a} + \frac{1}{2}\boldsymbol{l}_a^t\boldsymbol{l}_a + \frac{1}{2}\boldsymbol{r}_a^t\boldsymbol{r}_a + \frac{\gamma}{2}\sum_{i=1}^{n}\left(e_{1i}^2 + e_{2i}^2 + e_{3i}^2\right) \tag{3.4}$$

$$\text{subject to} \begin{cases} e_{1i} = y_i - \boldsymbol{a}^t\boldsymbol{x}_i, & i = 1,\ldots,n, \\ e_{2i} = y_i - \frac{1}{2}e_{i,L} - \boldsymbol{a}^t\boldsymbol{x}_i + \frac{1}{2}\boldsymbol{l}_a^t\boldsymbol{x}_i, & i = 1,\ldots,n, \\ e_{3i} = y_i + \frac{1}{2}e_{i,R} - \boldsymbol{a}^t\boldsymbol{x}_i - \frac{1}{2}\boldsymbol{r}_a^t\boldsymbol{x}_i, & i = 1,\ldots,n. \end{cases}$$

Here, the penalty parameter γ is a positive real constant and should be considered as a tuning parameter in the algorithm. This controls the smoothness and degree of fit. The cost function with squared error and regularization corresponds to a form of ridge regression.

Introducing Lagrange multipliers α_{1i}, α_{2i} and α_{3i}, we construct a Lagrange function as follows:

$$\begin{aligned} L = {} & \frac{1}{2}\boldsymbol{a}^t\boldsymbol{a} + \frac{1}{2}\boldsymbol{l}_a^t\boldsymbol{l}_a + \frac{1}{2}\boldsymbol{r}_a^t\boldsymbol{r}_a + \frac{\gamma}{2}\sum_{i=1}^{n}\left(e_{1i}^2 + e_{2i}^2 + e_{3i}^2\right) \\ & - \sum_{i=1}^{n}\alpha_{1i}\left(\boldsymbol{a}^t\boldsymbol{x}_i + e_{1i} - y_i\right) \\ & - \sum_{i=1}^{n}\alpha_{2i}\left(\boldsymbol{a}^t\boldsymbol{x}_i - \frac{1}{2}\boldsymbol{l}_a^t\boldsymbol{x}_i + e_{2i} - y_i + \frac{1}{2}e_{i,L}\right) \\ & - \sum_{i=1}^{n}\alpha_{3i}\left(\boldsymbol{a}^t\boldsymbol{x}_i + \frac{1}{2}\boldsymbol{r}_a^t\boldsymbol{x}_i + e_{3i} - y_i - \frac{1}{2}e_{i,R}\right). \end{aligned} \tag{3.5}$$

Then, the conditions for optimality are given by

$$\frac{\partial L}{\partial \boldsymbol{a}} = \boldsymbol{0} \to \boldsymbol{a} = \sum_{i=1}^{n}(\alpha_{1i} + \alpha_{2i} + \alpha_{3i})\boldsymbol{x}_i$$

$$\frac{\partial L}{\partial \boldsymbol{l}_a} = \boldsymbol{0} \to \boldsymbol{l}_a = -\frac{1}{2}\sum_{i=1}^{n}\alpha_{2i}\boldsymbol{x}_i$$

$$\frac{\partial L}{\partial \boldsymbol{r}_a} = \boldsymbol{0} \to \boldsymbol{r}_a = \frac{1}{2}\sum_{i=1}^{n}\alpha_{3i}\boldsymbol{x}_i$$

$$\frac{\partial L}{\partial e_{ki}} = 0 \rightarrow e_{ki} = \frac{1}{\gamma}\alpha_{1i}, \quad k = 1,2,3, \ i = 1,\ldots,n$$
$$\frac{\partial L}{\partial \alpha_{1i}} = 0 \rightarrow y_i = \boldsymbol{a}^t\boldsymbol{x}_i + e_{1i}, \quad i = 1,\ldots,n$$
$$\frac{\partial L}{\partial \alpha_{2i}} = 0 \rightarrow y_i - \frac{1}{2}e_{i,L} = \boldsymbol{a}^t\boldsymbol{x}_i - \frac{1}{2}\boldsymbol{l}_a^t\boldsymbol{x}_i + e_{2i}, \quad i = 1,\ldots,n$$
$$\frac{\partial L}{\partial \alpha_{3i}} = 0 \rightarrow y_i + \frac{1}{2}e_{i,R} = \boldsymbol{a}^t\boldsymbol{x}_i + \frac{1}{2}\boldsymbol{r}_a^t\boldsymbol{x}_i + e_{3i}, \quad i = 1,\ldots,n.$$

Defining $\boldsymbol{\alpha}_k = (\alpha_{k1},\ldots,\alpha_{kn})^t, k = 1,2,3$, $\boldsymbol{y} = (y_1,\ldots,y_n)^t$, $\boldsymbol{e}_R = (e_{1,R},\ldots,e_{n,R})^t$, $\boldsymbol{e}_L = (e_{1,L},\ldots,e_{n,L})^t$ and eliminating $\boldsymbol{a}$, $\boldsymbol{l}_a$, $\boldsymbol{r}_a$, $e_{ki}, k = 1,2,3$, we obtain

$$\begin{bmatrix} \boldsymbol{\Omega} + \frac{1}{\gamma}\boldsymbol{I} & \boldsymbol{\Omega} & \boldsymbol{\Omega} \\ \boldsymbol{\Omega} & \frac{5}{4}\boldsymbol{\Omega} + \frac{1}{\gamma}\boldsymbol{I} & \boldsymbol{\Omega} \\ \boldsymbol{\Omega} & \boldsymbol{\Omega} & \frac{5}{4}\boldsymbol{\Omega} + \frac{1}{\gamma}\boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha}_1 \\ \boldsymbol{\alpha}_1 \\ \boldsymbol{\alpha}_3 \end{bmatrix} = \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{y} - \frac{1}{2}\boldsymbol{e}_L \\ \boldsymbol{y} + \frac{1}{2}\boldsymbol{e}_R \end{bmatrix}, \tag{3.6}$$

where $\Omega_{ij} = \boldsymbol{x}_i^t\boldsymbol{x}_j, \ i,j = 1,\ldots,n$.

Hence, the prediction $\hat{Y}$ given by the LS-SVM procedure on the new unlabeled example $\boldsymbol{x}$ is

$$\hat{Y}_i = \left(\sum_{i=1}^{n}(\alpha_{1i} + \alpha_{2i} + \alpha_{3i})\boldsymbol{x}_i^t\boldsymbol{x}, -\frac{1}{2}\sum_{i=1}^{n}\alpha_{2i}\boldsymbol{x}_i^t\boldsymbol{x}, \frac{1}{2}\sum_{i=1}^{n}\alpha_{3i}\boldsymbol{x}_i^t\boldsymbol{x}\right). \tag{3.7}$$

Next, we will study LS-SVM to be used in estimating hybrid fuzzy nonlinear regression model. In contrast to hybrid fuzzy linear regression, there has been no article on hybrid fuzzy nonlinear regression. In this paper we treat hybrid fuzzy nonlinear regression for data of the form with crisp inputs and fuzzy output, without assuming the underlying model function. In the case where a linear regression function is inappropriate LS-SVM makes algorithm nonlinear. How can the above methods be generalized to the case where the regression function is not a linear function of the data? This could be achieved by simply preprocessing input patterns $\boldsymbol{x}_i$ by a map $\boldsymbol{\Phi} : R^{d+1} \rightarrow \mathcal{F}$ into some feature space $\mathcal{F}$ and then applying LS-SVM regression algorithm. This is an astonishingly straightforward way.

First notice that the only way in which the data appears in the training problem is in the form of dot products $\boldsymbol{x}_i^t\boldsymbol{x}_j$. The algorithm would only depend on the data through dot products in $\mathcal{F}$, *i.e.* on functions of the form $\boldsymbol{\Phi}(\boldsymbol{x}_i)^t\boldsymbol{\Phi}(\boldsymbol{x}_j)$. Hence it suffices to know and use $K(\boldsymbol{x}_i,\boldsymbol{x}_j) = \boldsymbol{\Phi}(\boldsymbol{x}_i)^t\boldsymbol{\Phi}(\boldsymbol{x}_j)$ instead of $\boldsymbol{\Phi}$ explicitly. The only difference between LS-SVMs for linear and nonlinear regression estimations is the use of mapping function $\boldsymbol{\Phi}$. The well used kernels for regression problem are given below.

$$K(\boldsymbol{x},\boldsymbol{y}) = (\boldsymbol{x}^t\boldsymbol{y} + 1)^p \quad : \text{Polynomial kernel}$$
$$K(\boldsymbol{x},\boldsymbol{y}) = e^{-\frac{\|\boldsymbol{x}-\boldsymbol{y}\|^2}{2\sigma^2}} \quad : \text{Gaussian kernel}$$

Here, p and σ^2 are kernel parameters. The kernel approach is again employed to address the curse of dimensionality. In final, the solution of the hybrid fuzzy nonlinear LS-SVM regression is given by

$$\hat{Y}_i = \left(\sum_{i=1}^{n}(\alpha_{1i} + \alpha_{2i} + \alpha_{3i})K(\boldsymbol{x}_i,\boldsymbol{x}), -\frac{1}{2}\sum_{i=1}^{n}\alpha_{2i}K(\boldsymbol{x}_i,\boldsymbol{x}), \frac{1}{2}\sum_{i=1}^{n}\alpha_{3i}K(\boldsymbol{x}_i,\boldsymbol{x})\right).$$

Here, $\boldsymbol{\alpha}_k = (\alpha_{k1},\ldots,\alpha_{kn})^t$, $k = 1,2,3$ can be obtained from the corresponding linear equation system constructed by replacing $\Omega_{ij} = \boldsymbol{x}_i^t\boldsymbol{x}_j$ in the Equation (3.7) with $\Omega_{ij} = K(\boldsymbol{x}_i,\boldsymbol{x}_j)$.

Table 1: Symmetric triangular fuzzy numbers

x_i	$\tilde{Y}_i$
2	(14, 1)
4	(16, 1)
6	(14, 1)
8	(18, 1)
10	(18, 1)
12	(22, 1)
14	(18, 1)
16	(22, 1)

4. Model Selection Method

When we use the proposed LS-SVM approach for hybrid fuzzy linear model, we still have to determine an optimal choice of the penalty parameter γ. In particular, when we use this algorithm for hybrid fuzzy nonlinear model, we have to determine in addition kernel parameter, which is the polynomial degree p for polynomial kernel and the kernel width σ for Gaussian kernel. There could be several parameter selection methods such as cross-validation methods, bootstrapping and Bayesian learning methods. In this paper, we use cross-validation methods.

If data is not scarce, then the set of available input-output measurements can be divided into two parts. One of them is used to train a model while the other, called the test set, is used for testing the model. In this way several different models, all trained on the training set, can be compared on the test set. According to their performance on the test set, we try to infer the proper values of parameters. This is the basic form of cross-validation. A better method is to partition the original set in several different ways and to compute an average performance over the different partitions.

An extreme variant of this is to split the n measurements into a training set of size $n-1$ and a test set of size 1 and average the squared error on the left-out measurements over the n possible ways of obtaining such a partition. This is called leave-one-out(LOO) or 1-fold cross-validation. LOO is computationally more demanding. For large data sets we typically prefer ten-fold cross-validation. The advantage of LOO is that all the data can be used for training - none has to be held back in a separate test set. In this paper, we actually use LOO in order to select penalty and kernel parameters, which is defined as follows:

$$\mathrm{LOO}(\lambda) = \frac{1}{n}\sum_{i=1}^{n}\left(\tilde{Y}_i - \hat{Y}_i^{(-i)}\right)^2, \tag{4.1}$$

where λ is the set of parameters and $\hat{Y}_i^{(-i)}$ is the predicted value of $\tilde{Y}_i$ obtained from the training data with the ith measurement $(\boldsymbol{x}_i, \tilde{Y}_i)$ removed.

5. Numerical Experiments

In this section, we use the same two data sets as in Chang (2001) and two simulated data sets to verify the effectiveness of hybrid fuzzy LS-SVM for crisp input and fuzzy output data. The experiments were conducted in MATLAB environment. Here, the Gaussian kernel $K(\boldsymbol{x},\boldsymbol{y}) = e^{-\|\boldsymbol{x}-\boldsymbol{y}\|^2/(2\sigma^2)}$ is used for the nonlinear model. The related parameters are determined by LOO method.

For the first example, a data set of symmetric triangular fuzzy numbers is given below.

The parameter γ for hybrid fuzzy linear LS-SVM is chosen as 400. Figure 1 shows the data set and hybrid fuzzy linear regression estimates by our proposed method and Chang (2001). In Figure 1, the solid and dotted lines illustrate the fuzzy linear regression estimates by our proposed method

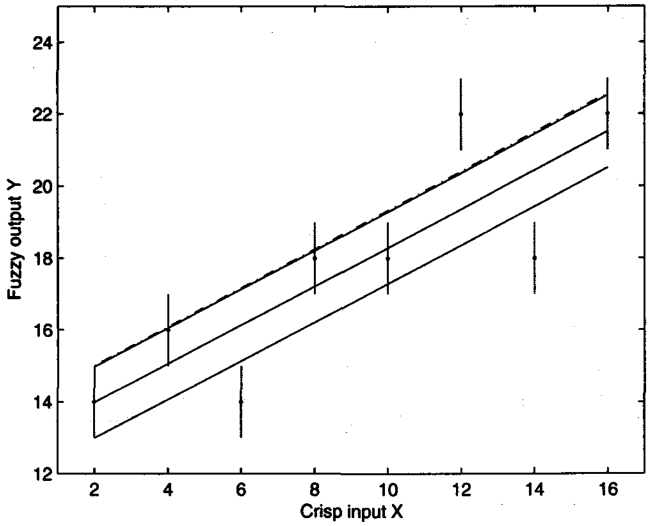

Figure 1: *Hybrid fuzzy linear regression estimates for the first example*

and Chang (2001) for $\mu = 0.0$, respectively. From Figure 1 we recognize that the estimates of both methods overlap so that we can not distinguish between them, and thus our proposed method performs quite well.

We now consider the hybrid standard error HS_e to evaluate both our proposed method and the method of Chang (2001). The HS_e is used to measure the goodness of fit between the hybrid regression model and the observed fuzzy data, which is given as follows:

$$HS_e = \sqrt{\frac{1}{n-p-1}\sum_{i=1}^{n}(\hat{Y}_i - \tilde{Y}_i)^2},$$

where $n - p - 1$ is the degree of freedom. In this example, p equals 1. The smaller the HS_e value, the better goodness of fit and the better accuracy of predictions. The HS_e's of both the proposed method and the method of Chang (2001) are calculated to be $HS_e = 1.7929$ and $HS_e = 1.79$, respectively. Hence, from Figure 1 and HS_e we recognize that both methods work similarly well for the linear case.

The parameter γ for hybrid fuzzy linear LS-SVM is chosen as 200. Figure 2 shows the data set and hybrid fuzzy linear regression estimates by our proposed method and Chang (2001). As in Figure 1, the solid and dotted lines illustrate the fuzzy linear regression estimates by our proposed method and Chang (2001) for $\mu = 0.0$, respectively. From Figure 2 we recognize that the estimates of both methods almost overlap and thus our proposed method performs quite well. The HS_e's of both the proposed method and the method of Chang (2001) are calculated to be $HS_e = 1.8519$ and $HS_e = 1.85$, respectively. Hence, we recognize that both methods work similarly well for the linear case.

In contrast to hybrid fuzzy linear regression, there has been no article on hybrid fuzzy nonlinear regression. We now treat hybrid fuzzy nonlinear regression for data of the form with crisp inputs and fuzzy output, without assuming the underlying model function. For the second example, a data set of asymmetric triangular fuzzy numbers is given below.

In order to illustrate the performance of the hybrid fuzzy nonlinear regression prediction for crisp inputs and fuzzy outputs, two examples are considered. In both examples, the 25 centers of x_i's are randomly generated in $[0, 0.25, \ldots, 10.0]$. The spreads of $\tilde{Y}_i$'s are randomly generated in

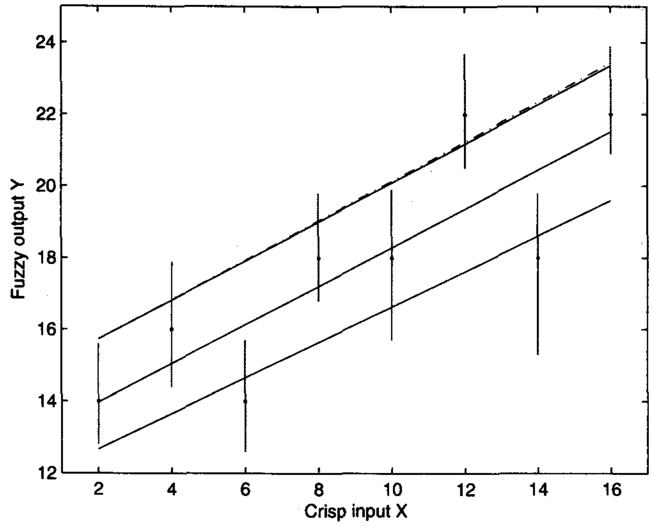

Figure 2: *Hybrid fuzzy linear regression estimates for the second example*

Table 2: Asymmetric triangular fuzzy numbers

x_i	$\tilde{Y}_i$
2	(14, 1.2, 1.6)
4	(16, 1.6, 1.9)
6	(14, 1.4, 1.7)
8	(18, 1.2, 1.8)
10	(18, 2.3, 1.9)
12	(22, 1.5, 1.7)
14	(18, 2.7, 1.8)
16	(22, 1.1, 1.9)

$[0.3, 0.4, \ldots, 1.0]$. The centers of $\tilde{Y}_i$'s of the third and fourth examples are generated as follows:

$$y_i = 1.1 + 2.5\log(1 + x_i) + \epsilon_i$$
$$y_i = 2.1 + \exp(0.2x_i) + \epsilon_i,$$

respectively, where $\epsilon_i, i = 1, 2, \ldots, 25$, is a random error from the normal distribution with mean 0 and variance 0.01.

The parameters (γ, σ^2) for the third and fourth examples are chosen as $(1000, 20)$ and $(1000, 100)$, respectively. Figure 3 and 4 show the data set and hybrid fuzzy nonlinear regression estimates by our proposed method for the third and fourth examples, respectively. In Figure 3 and 4, the dotted, solid and dashed curves illustrate the true centers and the hybrid fuzzy nonlinear regression estimates for $\mu = 0.0$ and $\mu = 0.7$, respectively. The HS_e's for both examples are calculated to be $HS_e = 0.1050$ and $HS_e = 0.1186$, respectively. These values are quite small. Hence, we recognize that our proposed method achieves satisfying results for the nonlinear case, too.

6. Conclusions

Through numerical experiments, we realize that the proposed algorithm derive the satisfying solutions and are the attractive approaches to modeling the data with crisp inputs and fuzzy output. In particular,

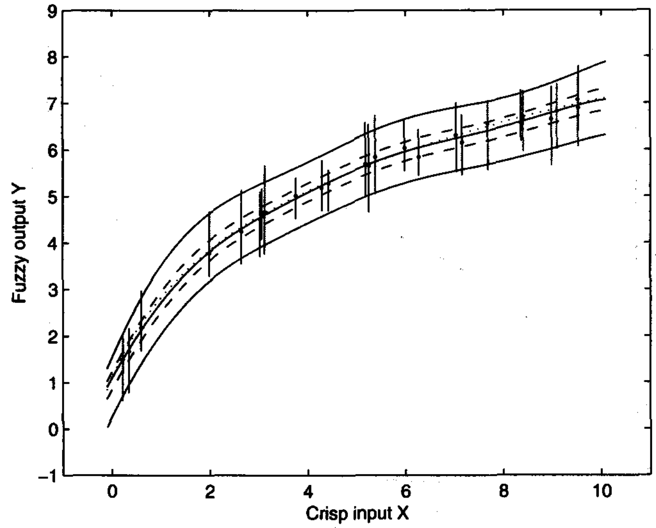

Figure 3: *Hybrid fuzzy nonlinear regression estimates for the third example*

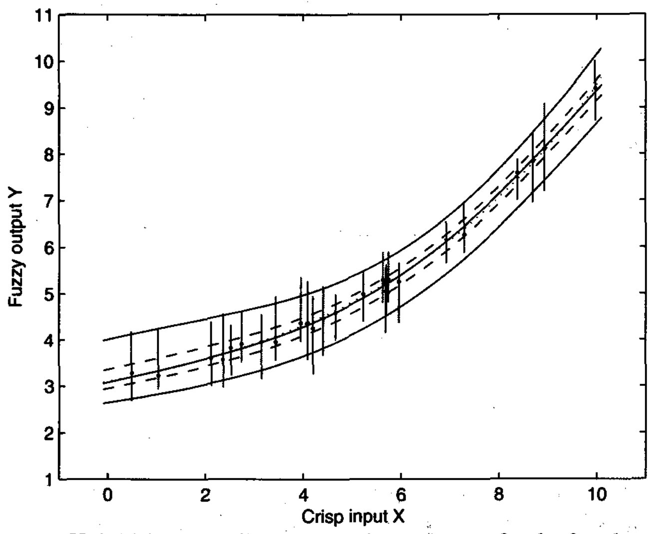

Figure 4: *Hybrid fuzzy nonlinear regression estimates for the fourth example*

we realize that we can use this algorithm successfully when a linear model is inappropriate. One of advantages of this algorithm is that we do not need to assume the underlying structure for the fuzzy nonlinear regression model used in this paper. This nonparametric method turns out to be a promising method which has been attempted to treat hybrid fuzzy nonlinear regression model.

The algorithm combine generalization control with a technique to address the curse of dimensionality. The main formulation results in solving a simple linear equation system. Hence, this is not a computationally expensive way.

In this paper, we focus on proposing a new method to evaluate hybrid fuzzy linear and nonlinear regression models using WFA and LS-SVM. However, we can derive, in a straightforward manner, similar algorithms by using standard SVM which uses ϵ-insensitive loss function. The penalty and kernel parameters of the proposed algorithm have been tuned using LOO cross-validation and a grid search mechanism.

References

Chang, Y. -H. O. (2001). Hybrid fuzzy least-squares regression analysis and its reliability measures, *Fuzzy Sets and Systems*, **119**, 225–246.

Chao, R. and Ayyub, B. M. (1996). Structural analysis with fuzzy variables, *Microcomputers in Civil Engineering*, **11**, 47–58.

Gunn, S. (1998). Support vector machines for classification and regression, *ISIS Technical Report*, University of Southampton.

Hong, D. H. and Hwang, C. (2003). Support vector fuzzy regression machines, *Fuzzy Sets and Systems*, **138**, 271–281.

Hong, D. H. and Hwang, C. (2005). Interval regression analysis using quadratic loss support vector machine, *IEEE Transactions on Fuzzy Systems*, **13**, 229–237.

Hwang, C., Hong, D. H., Na, E., Park, H. and Shim, J. (2005). Interval regression analysis using support vector machine and quantile regression, *Lecture Notes in Computer Science*, **3613**, 100–109.

Hwang, C., Hong, D. H. and Seok, K. H. (2006). Support vector interval regression machine for crisp input and out data, *Fuzzy Sets and Systems*, **157**, 1114–1125.

Shim, J., Hwang, C. and Hong, D. H. (2009). Fuzzy semiparametric support vector regression for seasonal time series analysis, *Communications of the Korean Statistical Society*, **16**, 335–348.

Smola, A. J. and Schoelkopf, B. (1998). A tutorial on support vector regression, *Neuro-COLT2 Technical Report*, NeuroCOLT.

Suykens, J. A. K. (2001). Nonlinear modelling and support vector machines, *Proceeding of the IEEE International Conference on Instrumentation and Measurement Technology*, Budapest, Hungary, 287–294.

Tanaka, H. (1987). Fuzzy data anlysis by possibilistic linear models, *Fuzzy Sets and Systems*, **24**, 363–375.

Tanaka, H., Uejima, S. and Asai, K. (1982). Linear regression analysis with fuzzy model, *IEEE Transactions on Systems, Man, and Cybernetics* , **12**, 903–907.

Tanaka, H. and Watada, J. (1988). Possibilistics linear systems and their applications to linear regression model, *Fuzzy Sets and Systems*, **27**, 275–289.

Vapnik, V. N. (1998). *Statistical Learning Theory*, John Wiley & Sons, New York.

Received January 2010; Accepted February 2010