Quarterly Loss Support Vector Interval Regression Machine for Crisp Input–Output Data

Changha Hwang1)

Abstract

Support vector machine (SVM) has been very successful in pattern recognition and function estimation problems for crisp data. This paper proposes a new method to evaluate interval regression models for crisp input–output data. The proposed method is based on quadratic loss SVM, which implements quadratic programming approach giving more diverse spread coefficients than a linear programming one. The proposed algorithm here is model-free method in the sense that we do not have to assume the underlying model function. Experimental result is then presented which indicate the performance of this algorithm.

Keywords: Interval regression, Quadratic loss, Support vector machine

1. Introduction

In reality, information is often uncertain, imprecise and incomplete, which can be represented by fuzzy data or a generalization of interval data. For handling interval data, fuzzy regression analysis becomes an important tool. Fuzzy regression analysis can be simplified to interval regression analysis, where interval regression models are implemented. In interval linear regression, possibility and necessity models have been employed under given interval data. Coefficients in interval regression model are assumed to be interval. In fact, interval regression is regarded as the simplest version of possibilistic regression analysis. Possibilistic regression analysis has been first proposed by Tanaka et al.(1992) where a fuzzy linear system has been used as a regression model. To determine the interval parameters of interval regression models, a basic linear programming (LP) problem should be solved. However, the main difficulty in nonlinear interval regression

1) Professor, Dept. of Statistical Information, Catholic University of Daegu.
E–mail : chhwang@cu.ac.kr
with LP is to choose a nonlinear model from an infinite number of alternatives. Hence, several approaches with learning scheme were proposed to deal with this problem. Recently, Tanaka and Lee(1998) proposed quadratic programming(QP) approach to interval regression analysis. The interval regression analysis based on back-propagation neural networks(BPNNs) was originally proposed in Ishibuchi and Tanaka(1992). Jeng et al.(2003) proposed the support vector interval regression networks (SVIRNs), which consists of two radial basis function networks. Another direction of SVM approach to fuzzy regression was illustrated in Hong and Hwang(2003). For the details of SVM, see Vapnik(1995).

In this paper we propose a new method to evaluate interval regression models for crisp input-output data. The proposed method is based on quadratic loss support vector machine(SVM), which basically follows Tanaka and Lee(1998) and is much simpler than that in Jeng et al.(2003).

2. Interval Regression for Crisp Input–Output Data

In this section, we illustrate how to get solutions for interval regression models using QP approach proposed by Tanaka and Lee(1998).

Suppose that we are given training data \( \{(x_i, y_i), i = 1, \ldots, n\} \subset X \times R \), where \( X \) denotes the space of the input patterns. We begin by describing the case of interval linear regression functions \( Y(x) \), taking the form

\[
Y(x) = A_0 + A_1 x_1 + \cdots + A_m x_m = A^t x, \tag{1}
\]

where \( x = (1, x_1, \ldots, x_m)^t \) is a real input vector, \( A = (A_0, A_1, \ldots, A_m)^t \) is an interval coefficient vector, and \( Y(x) \) is the corresponding estimated interval. An interval coefficient \( A_i \) is denoted as \( A_i = (a_i, c_i) \) where \( a_i \) is a center and \( c_i \) is a radius. By interval arithmetic, the regression model (1) can be expressed as

\[
Y(x_i) = (a_0, c_0) + (a_1, c_1)x_{i1} + \cdots + (a_m, c_m)x_{im}
= (a_0 + a_1 x_{i1} + \cdots + a_m x_{im}, c_0 + c_1| x_{i1} | + \cdots + c_m | x_{im} | )
= (a^t x_i, c^t | x_i |),
\]

where \( a = (a_0, a_1, \ldots, a_m)^t, \ c = (c_0, c_1, \ldots, c_m)^t, \) and \( | x_i | = (1, | x_{i1} |, \ldots, | x_{im} |)^t \).

We now illustrate the formulation integrating central tendency and possibilistic property in Tanaka and Lee(1998). We consider a new objective function which reflects both properties of least squares and possibilistic approaches

\[
J = k_1 \sum_{i=1}^{n} (y_i - a^t x_i)^2 + k_2 \sum_{i=1}^{n} c^t |x_i||x_i|^t c
\]
where \( \sum_{i=1}^{n} |x_i| |x_i|^{t} \) is a symmetric positive definite matrix and \( k_1 \) and \( k_2 \) are weight coefficients. Interval regression analysis using this new objective function (3) is to determine the interval coefficients \( A_i = (a_i, c_i), i = 0, 1, \ldots, m \) by solving the following QP problem:

\[
\begin{align*}
\min_{a, c} \quad & J = k_1 \sum_{i=1}^{n} (y_i - a^T x_i)^2 + k_2 \sum_{i=1}^{m} c^T \frac{1}{x_i} |x_i| c \\
\text{subject to} \quad & a^T x_i + c^T |x_i| \geq y_i, \quad a^T x_i - c^T |x_i| \leq y_i, \quad i = 1, \ldots, n \\
& c_i \geq 0, \quad i = 0, 1, \ldots, m
\end{align*}
\]

The weight coefficients \( k_1 \) and \( k_2 \) in (4) have an important role in formulating fuzzy regression models. These coefficients can be assigned by considering a tradeoff between two terms in (4).

3. Quadratic Loss SVM for Interval Regression

In this section, we propose a new method to evaluate interval linear and nonlinear regression models combining the possibility estimation formulation integrating the property of central tendency with the principle of SVM. We first need to look at how to get solutions for interval linear regression models by implementing the SVM approach. We follow the way of constructing objective function in SVM regression. Then, the objective function can be assumed as the following quadratic function:

\[
\begin{align*}
\min_{a, c} \quad & \frac{1}{2} (\| a \|^2 + \| c \|^2) + \frac{C}{2} \left( \sum_{i=1}^{n} \xi_{1i}^2 + \sum_{i=1}^{n} (\xi_{2i}^2 + \xi_{2i}^r) \right) \\
\text{subject to} \quad & c^T |x_i| \leq \xi_{1i}, \\
& y_i - a^T x_i \leq \xi_{2i}, \quad a^T x_i - y_i \leq \xi_{2i}^r, \quad i = 1, \ldots, n \\
& a^T x_i + c^T |x_i| \geq y_i, \quad a^T x_i - c^T |x_i| \leq y_i, \quad i = 1, \ldots, n.
\end{align*}
\]

Although it is possible to use two weight coefficients like Tanaka and Lee(1998), we use one weight coefficient. Here, \( \xi_{1i} \) represents spreads of the estimated outputs, and \( \xi_{2i}, \xi_{2i}^r \) are slack variables representing upper and lower constraints on the outputs of the model. Hence, we can construct a Lagrange function as follows:

\[
L = \frac{1}{2} (\| a \|^2 + \| c \|^2) + \frac{C}{2} \left( \sum_{i=1}^{n} \xi_{1i}^2 + \sum_{i=1}^{n} (\xi_{2i}^2 + \xi_{2i}^r) \right)
\]
\[
- \sum_{i=1}^{n} a_{1i}(\xi_{1i} - a^i|x_i|)
- \sum_{i=1}^{n} a_{2i}(\xi_{2i} - y_i + a^i x_i) - \sum_{i=1}^{n} a^*_2(\xi_{2i} - a^i x_i + y_i)
- \sum_{i=1}^{n} a^*_3(a^i x_i + c^i|x_i| - y_i) - \sum_{i=1}^{n} a^*_3(y_i - a^i x_i + c^i|x_i|)
\]

Here, \(a_{1i}, a_{2i}, a^*_2, a^*_3, a^*_3\) are Lagrange multipliers. It follows from the saddle point condition that the partial derivatives of \(L\) with respect to the primal variables \((a, c, \xi_{1i}, \xi_{2i}, \xi^*_2)\) have to vanish for optimality.

\[
\frac{\partial L}{\partial a} = 0 \rightarrow a = \sum_{i=1}^{n} (a_{2i} - a^*_2) x_i + \sum_{i=1}^{n} (a_{3i} - a^*_3) x_i
\]

\[
\frac{\partial L}{\partial c} = 0 \rightarrow c = -\sum_{i=1}^{n} a_{1i}|x_i| + \sum_{i=1}^{n} (a_{3i} + a^*_3)|x_i|
\]

\[
\frac{\partial L}{\partial \xi_{1i}} = 0 \rightarrow \xi_{1i} = \frac{1}{C} a_{1i}
\]

\[
\frac{\partial L}{\partial \xi_{2i}^*} = 0 \rightarrow \xi_{2i}^* = \frac{1}{C} a_{2i}^*
\]

Substituting (7)-(10) into (6) yields the dual optimization problem.

\[
\text{maximize} \left\{ -\frac{1}{2} \left( \sum_{i,j=1}^{n} (a_{2i} - a^*_2)(a_{2j} - a^*_2) x_i^j x_j \right. \\
+ 2 \sum_{i,j=1}^{n} (a_{2i} - a^*_2)(a_{3j} - a^*_3) x_i^j x_j + \sum_{i,j=1}^{n} (a_{3i} - a^*_3)(a_{3j} - a^*_3) x_i^j x_j \\
+ \sum_{i,j=1}^{n} (a_{3i} + a^*_3)(a_{3j} + a^*_3)|x_i|^j |x_j| \right) - \frac{1}{2C} \sum_{i=1}^{n} a^2_{2i} \\
- \frac{1}{2C} \sum_{i=1}^{n} (a_{2i} - a^*_2)(a_{3i} + a^*_3) y_i + \sum_{i=1}^{n} (a_{3i} - a^*_3) y_i \right\}
\]

subject to

\(a_{1i}, a_{ki}, a_{ki}^* \geq 0, k=2,3\).

Solving (11) with above constraints determines the Lagrange multipliers, \(a_{1i}, a_{ki}, a_{ki}^*\). Hence, if \(c^i|x| \geq 0\), then the linear interval regression function is as follows:

\[
Y(x) = (a^i x, c^i x)
\]

Next, we will consider nonlinear interval regression model. In contrast to linear interval regression, there have been no articles on nonlinear interval regression. In this paper we treat nonlinear interval regression, without assuming the underlying model function. In the case where a linear regression function is inappropriate
SVM makes algorithm nonlinear. This could be achieved by simply preprocessing input patterns \( x_i \) by a map \( \Phi : \mathbb{R}^d \rightarrow E \) into some feature space \( E \) and then applying SVM regression algorithm. This is an astonishingly straightforward way.

First notice that the only way in which the data appears in (11) is in the form of inner products \( x_i^T x_j, \, |x_i|^2 |x_j| \). The algorithm would only depend on the data through dot products in \( E \), i.e. on functions of the form \( K(x_i, x_j) = \Phi(x_i)^T \Phi(x_j) \), \( K(|x_i|, |x_j|) = \Phi(|x_i|)^T \Phi(|x_i|) \). The well used kernels for regression problem are given below:

\[
K(x, y) = (x^T y + 1)^p, \quad K(x, y) = e^{-\frac{||x-y||^2}{2\sigma^2}}.
\]

Here, \( p \) and \( \sigma^2 \) are kernel parameters. In final, the nonlinear interval regression solution is given by

\[
\begin{align*}
\text{maximize} & \quad -\frac{1}{2} \left( \sum_{i,j=1}^{n} (a_2i - a_2^*) (a_2j - a_2^*) K(x_i, x_j) \\
& \quad + 2 \sum_{i,j=1}^{n} (a_2i - a_2^*) (a_3j - a_3^*) K(x_i, x_j) \\
& \quad + \sum_{i,j=1}^{n} (a_3i - a_3^*) (a_3j - a_3^*) K(x_i, x_j) + \sum_{i,j=1}^{n} a_1i a_1j K(|x_i|, |x_j|) \\
& \quad - 2 \sum_{i,j=1}^{n} a_1i (a_3j + a_3^*) K(|x_i|, |x_j|) \\
& \quad + \sum_{i,j=1}^{n} (a_3i + a_3^*) (a_3j + a_3^*) K(|x_i|, |x_j|) \right) - \frac{1}{2C} \sum_{i=1}^{n} a_i^2 \\
& \quad - \frac{1}{2C} \sum_{i=1}^{n} (a_2i + a_2^*) y_i + \sum_{i=1}^{n} (a_3i - a_3^*) y_i \right) \tag{13}
\end{align*}
\]

subject to

\[
a_1i, a_3i, a_3^* \geq 0, \, k = 2, 3.
\]

4. Empirical Studies

In this section, two examples are used to verify the effectiveness of the proposed SVM for the interval regression. These simulations were conducted in the Matlab environment. In the first example, the training data sets are generated by

\[
y_i = 0.2 \sin(2\pi x_i) + 0.2x_i^2 + 0.3 + (0.1 x_i^3 + 0.05) \text{rand}[-1,1],
\]

\[
x_i = 0.02(i - 1), \quad i = 1, 2, \cdots, 51,
\]

where \( \text{rand}[-1,1] \) represents a real number randomly generated in the interval \([-1,1]\). In the second example, we consider the function given by
$y = \exp(-x^2) + 0.5 \text{md}[-1,1], \ x \in [-2, 2].$

For the second example, we generate 51 data points. Fig. 1 and 2 show the interval regression results for the first and second example, respectively. The parameters \((C, \sigma)\) were determined by leave-one-out method, and chosen as \((30, 0.18)\) and \((35, 0.70)\), respectively.

**Fig. 1** Results for the first example **Fig. 2** Results for the second example

In Figs 1 and 2, the solid and the dotted lines represent the maximum and the minimum limitations, respectively. As seen from Figs 1 and 2, the proposed method works quite well. In fact, this method is much simpler and computationally less expensive than SVIRN in Jeng et al. (2003), since SVIRN uses two independent neural networks for estimating intervals after applying SVM to determining the initial structure. Compared with the method in Tanaka and Lee (1998), our procedure has an advantage that we do not need assume the underlying model function.

**References**


[ received date : Dec. 2003 , accepted date : Mar. 2004 ]