ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

SMOOTH SUPPORT VECTOR REGRESSION BASED ON MODIFICATION SPLINE INTERPOLATION

[1] BIN REN, [1] HUIJIE LIU, [1] LEI YANG, [2] LIANGLUN CHENG
[1] School of Electronic Engineering, Dongguan University of Technology, Dongguan, 523808, China
[2] School of Automation, Guangdong University of Technology, Guangzhou,510006, China

ABSTRACT

Regression analysis is often formulated as an optimization problem with squared loss functions. Facing the challenge of the selection of the proper function class with polynomial smooth techniques applied to Support Vector Regression models, this study takes three interplation points spline interpolation technology and modification interpolation value to generate a new polynomial smooth function $|x|_\varepsilon^2$ in ε-insensitive support vector regression. The experimental analysis shows that $S_{M\varepsilon}^2$-function is better than p_ε^2-function and S_ε^2-function in properties, and the approximation accuracy of the proposed smooth function is three order of higher than that of classical p_ε^2-function.

Keywords: *Support Vector Regression, ε-insensitive Loss Function, Smooth Polynomial Function, Modification Interpolation*

1. INTRODUCTION

Smooth function has been widely studied in numerical modeling[1-6], which, especially in the interest of the authors, has been successfully applied for classification of and regression model fittings in image processed and pattern recognition[2, 3, 7, 8]. Applying smooth function to regression models means to deal with square unsmoothed issue in ε-insensitive loss function while fitting the regression models [8]. According to the basic concept on how to solve classification problem, Lee et al, used p_ε^2 -function as to smoothly approach the target function, and brought forward the ε-insensitive support vector regression model (ε-SSVR) in 2005 [8]. Their results show that the effect of ε-SSVR is better than both LIBSVM[9] and SVM^{light} [10] in both regression property and efficiency.

It is, however, still an open and challenging issue to find a better smooth function [1,2,5,7,9]. Accordingly this paper is motive to present a study on using three interpolation points Cubic Spline Interpolation polynomial and modification interpolation value to improve this kind of smooth function in fitting support vector regression model. The proposed $S_{M\varepsilon}^2$ -function is better than p_ε^2 -function and S_ε^2 -function in property, and the approximation accuracy of the proposed smooth function is three order of higher than that of classical p_ε^2 -function and one order of higher than that of classical p_ε^2 -function S_ε^2 -function. The simulation case study shows that it improves the regression effect.

This paper is organized as follows: section 2 introduces regression problems and difficulties. section 3 introduces ε-insensitive loss function and support vector regression .In Section 4, we first introduce the principle and derive formula of Cubic Spline Interpolation polynomial, then use Modification Spline Interpolation polynomial to smooth single variable positive function, and we define $|x|_\varepsilon^2$'s polynomial approximation function $S_{M\varepsilon}^2(x,k)$. In Section 5, we analyze the performance of polynomial smooth approximation function $S_{M\varepsilon}^2(x,k)$. It is the 1st-order smooth function, and the approximation accuracy is 0.0081/k. Section 6, we run two numerical simulation experiments by using data sets from artificial database and UCI database to verify the validity of the model. Finally, we make a conclusion and foresee the future work in section 7.

2. REGRESSION BASED DATA FITTING

First, we discuss the simplest regression problem in 2-dimensional space: Let's suppose all values x_1 , x_2 , ... , x_m from 1 to m, each x_i is corresponding with an observed value y_i . The

Journal of Theoretical and Applied Information Technology
15th October 2012. Vol. 44 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

purpose is using the designated data set to generate interdependent function $y = f(x)$. We usually use this way to solve the problem as below: first, to restrict the function $y = f(x)$ in a simple function class in advance, then searching for $f(x)$ that can meet the following conditions in the function class as mush as possible:

$$y_i = f(x_i), \qquad i = 1, 2, \cdots, m \qquad (1)$$

In order to easy to deal with, we always use linear regression way, i. e., restricting $f(x)$ to be linear function $f(x) = wx + b$. Then search for f (x) which can meet the equation (1).

$$(y_i - f(x_i))^2 = (y_i - wx_i - b)^2 \qquad (2)$$

Equation (2) is often used to measure the deviation degree between $y = f(x) = wx + b$ and $y_i = f(x_i)$.The smaller value is, the less error is and higher efficient it is. So this process can be translated into the following optimized formula. So that we can define w and b in the function $f(x) = wx + b$:

$$\min_{w,b} \sum_{i=1}^{m} (y_i - wx_i - b)^2 \qquad (3)$$

Obviously, the regressive formula and solution above can be extended to a normal situation.

First, extending data class (1) to data set S:

$$S = \{(x_1, y_1), \cdots, (x_m, y_m)\} \subseteq R^n \times R \qquad (4)$$

Secondly, the function class which restrict the function y=f (x) (1) above also can be extended to be a real function set F. Generally, there is not only criterion to measure the deviation of regression function $y = f(x)$ from $y_i = f(x_i)$. We call equation (3) above as quadratic loss function. Of course, other loss function also can be used. If we name loss function as $c(x, y, f)$. The optimized formula (3) will become minimization formula with empirical risk.

$$\min_{f \in F} \sum_{i=1}^{m} c(x_i, y_i, f(x_i)) \qquad (5)$$

Thus, the interdependent function $y = f(x)$ can be obtained, i. e., regression function.

When solving the optimized formula(5), the first issue is how to choose the function class set F. For the designated normal training data set S (4), we can not restrict F to be too small function class, such as linear functions will produce large regression error in a model in nature of nonlinearity. On the other hand, F cannot be too large otherwise the regression function will be meaningless. For example, we will obtain the following equation based on data set S (4) when F is the whole real function set.

$$f(x) = \begin{cases} y_i, x = x_i \\ 0, x \neq x_i \end{cases} \quad i = 1, 2 \cdots, m. \qquad (6)$$

Obviously, the regression function is too illogical. Accordingly the key point is how to choose the function class set F, neither too simple nor too complicated. Furthermore, it becomes difficult to choose the right one for the regression function.

3. SUPPORT VECTOR REGRESSION

For better analysis, we define the ε-insensitive loss function of independent variable X as $|x|_\varepsilon$, $|x|_\varepsilon = \max\{0, |x| - \varepsilon\}$, as shown in Figure 1. Definition the square of ε-insensitive loss function as $|x|_\varepsilon^2$, and the positive function x_+ as $(x_+)_i = \max\{0, x_i\}$.

Data set $S = \{(x_1, y_1), \cdots, (x_m, y_m)\} \subseteq R^n \times R$, define matrix A=[$x_1, x_2, \cdots x_m$], x_i is n dimensional vector, each x_i is corresponding with an observed value y_i, obviously $A \in R^{m \times n}$, that it is $S = \{(A_i, y_i) \mid A_i \in R^n, y_i \in R, \text{ for } i = 1, \cdots, m\}$. The purpose is using the designated data set S to generate a regression function $f(x)$, let $f(x)$ predict y more accurately according to the new input of x. The standard we use is ε-insensitive loss function:

$$|y - f(x)|_\varepsilon = \max\{0, |y - f(x)| - \varepsilon\} \qquad (7)$$

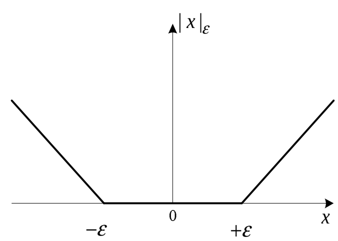

Figure 1: E-Insensitive Loss Function $|x|_\varepsilon$

For linear regression case, $f(x) = w^T x + b$, where $w \in R^n$ is a indeterminate vector, b is a

Journal of Theoretical and Applied Information Technology
15th October 2012. Vol. 44 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

indeterminate constant. ε-insensitive linear regression function is shown in Figure 2, we select two hyperplanes of the margin in a way and we call the distance between the two hyperplanes is ε zone. Only by there are no training points falling into the margin, we can have loss, and the loss is $|y - f(x)| - \varepsilon$.

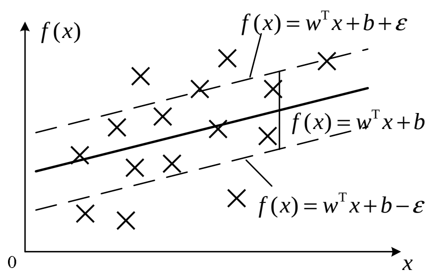

Figure 2: Ε-Insensitive Linear Regression Function

For nonlinear regression case $f(x)=\omega\varphi(x)+b$, where $\varphi(\cdot)$: nonlinear function. In theory, we can change it into linear regression ones to settle it according to kernel technique.

Standard regression problem is to solve the following minimum problem [11]:

$$\begin{cases} \min Q(\omega,b,\xi,\xi^*)=\frac{1}{2}\omega^T\omega+C\sum_{i=1}^{n}(\xi_i+\xi_i^*) \\ s.t.\ \ y_i-\omega\cdot\varphi(x_i)-b\le\varepsilon+\xi_i \\ \omega\cdot\varphi(x_i)+b-y_i\le\varepsilon+\xi_i^* \\ \xi_i\ge 0,\xi_i^*\ge 0, i=1,2,\cdots,n \end{cases} \quad (8)$$

Where

$\xi=(\xi_1,\xi_2,\cdots,\xi_n)^T$, $\xi^*=(\xi_1^*,\xi_2^*,\cdots,\xi_n^*)^T$, ε($\geqslant$0) is the maximum deviation allowed during the training and C(>0) represents the associated penalty for excess deviation during the training. The slack variables ξ_i and ξ_i^*, correspond to the size of this excess deviation for positive and negative deviations respectively. The first term, $\omega^T\omega$ is the regularized parameter; thus, it controls the function capacity; the second term $\sum_{i=1}^{n}(\xi_i+\xi_i^*)$, is the empirical error measured by the ε-insensitive loss function.

The computation of Standard support vector regression is more complicated, because when solving the Optimization problem, you need to solve quadratic programming, especially when the training sample number is increased. The solution will face curse of dimensionality, in result that we can't train it. Suykens J.A.K [11] proposes least squares method-support vector machines (LS-SVM) to make the problem comes down to linear equations, and solving linear equations is easier and faster than the quadratic programming. Standard regression problem is to solve the following problem:

$$\begin{cases} \min Q(\omega,\xi)=\frac{1}{2}\omega^T\omega+\frac{C}{2}\sum_{i=1}^{n}\xi_i^2 \\ s.t.\ \ y_i=\omega\cdot\varphi(x_i)+b\xi_i, i=1,2,\cdots,n \end{cases} \quad (9)$$

In addition, Lee et al adds the parameter $\frac{1}{2}b^2$ into the objective function to induce strong convexity and to guarantee that the problem has a unique global optimal solution. The regression issue can be expressed by below unconstrained optimized issue formula [9]:

$$\min_{(w,b)\in R^{n+1}} \frac{1}{2}(w^Tw+b^2)+\frac{C}{2}\sum_{i=1}^{m}|A_iw+b-y_i|_\varepsilon^2. \quad (10)$$

Obviously, the $|x|_\varepsilon^2$ in formula (10) is not derivative, so this target function is not derivative.

4. POLYNOMIAL SMOOTH APPROXIMATION FUNCTION

Cubic spline function may generate smooth interpolation curve by combining the discontinuous cubes and the second derivative is continuous at the joint point, namely sampling point.

4.1 Mathematical Description

Assumption a set of nodes $a\le x_0<x_1<...<x_n\le b$ at $[a,b]$, if the function s(x) meet below term[11],

(1) $s(x)\in C^2[a,b]$;

(2) $s(x)$ is cubic polynomial at every region $[x_i,x_{i+1}]$ $(i=0,1,...,n-1)$.

If s(x) also meets the following spline term at node,

(3) $S(x_i)=f_i, i=0,1,...n$.

Then $s(x)$ is called cubic spline interpolation function, the second derivative of $s(x)$ at $[a,b]$ is continuous.

In this study, when using cubic spine interpolation polynomial approach positive function

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

x_+, at the end point a= x_0 ,b= x_n of region [a,b], using following boundary conditions: $S'(x_0) = f_0'$, $S'(x_n) = f_n'$, at region $[x_i, x_{i+1}]$, the formula of cubic spline function is

$$S(x) = M_i \frac{(x_{i+1}-x)^3}{6h_i} + M_{i+1}\frac{(x-x_i)^3}{6h_i} + (f_i - \frac{M_i h_i^2}{6})\frac{x_{i+1}-x}{h_i}$$
$$+(f_{i+1} - \frac{M_{i+1}h_i^2}{6})\frac{x-x_i}{h_i} \quad (i = 0,1,\cdots,n-1) \qquad (11)$$

To solve M_j we can write it in matrix form as following:

$$\begin{bmatrix} 2 & 1 & & & \\ \mu_1 & 2 & \lambda_1 & & \\ \cdots & \cdots & \cdots & & \\ & & 2 & \lambda_{n-1} \\ & & 1 & 2 \end{bmatrix} \begin{bmatrix} M_0 \\ M_1 \\ \cdots \\ M_{n-1} \\ Mn \end{bmatrix} = \begin{bmatrix} d_0 \\ d_1 \\ \cdots \\ d_{n-1} \\ d_n \end{bmatrix} \qquad (12)$$

Where: $d_0 = 6f[x_0, x_1, x_0]$ (13)

$$d_i = 6f[x_{i-1}, x_i, x_{i+1}] \qquad (14)$$

$$d_n = 6f[x_n, x_{n-1}, x_n] \qquad (15)$$

$$\mu_i = \frac{h_{i-1}}{h_{i-1}+h_i} \qquad (16)$$

$$\lambda_i = 1-\mu_i = \frac{h_i}{h_{i-1}+h_i} \qquad (17)$$

$$h_i = x_i - x_{i-1} \qquad (18)$$

4.2 The Derivation Of Smoothing Process

We use the method of cubic spline interpolation polynomial to smooth single variation function at region $[-\frac{1}{k}, \frac{1}{k}]$. Take 3 interpolation data from positive function x_+ at region x<0 x=0 and x>0m , point $x_j = -\frac{1}{k}$, $x_{j+1} = 0$ and $x_{j+2} = \frac{1}{k}$ (k>0), corresponding $f_j = 0$, $f_{j+1} = 0$, $f_{j+2} = \frac{1}{k}$.

Using cubic spline interpolation polynomial to smooth positive function x_+ at region $[-\frac{1}{k}, \frac{1}{k}]$, table 1 is interpolation point, corresponding function and the first derivative value.

Table 1: Interpolation Point And Function Value

j	x_j	f_j	f_j'
0	-1/k	0	0
1	0	0	0
2	1/k	1/k	1

$$h_0 = h_1 = \frac{1}{k},\ \mu_1 = \frac{1}{2},$$

$$\mu_2 = 1,\ \lambda_0 = 1,\ \lambda_1 = \frac{1}{2},$$

$$d_0 = 6\frac{1}{h_0}(f[x_0, x_1] - f_0') = 0$$
$$d_1 = 6f[x_0, x_1, x_2] = 3k$$

$$d_2 = 6\frac{1}{h_1}(f_2' - f[x_1, x_2]) = 0$$

$$\begin{bmatrix} 2 & 1 & 0 \\ \frac{1}{2} & 2 & \frac{1}{2} \\ 0 & 1 & 2 \end{bmatrix} \begin{bmatrix} M_0 \\ M_1 \\ M_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 3k \\ 0 \end{bmatrix} \qquad (19)$$

Then have the answer $M_0 = -k$, $M_1 = 2k$, $M_2 = -k$, so the cubic spline interpolation polynomial for the smoothing of single variable function at region $[-\frac{1}{k}, \frac{1}{k}]$ is as below:

$$S(x) = \begin{cases} -\frac{1}{2}k^2x^3 + kx^2 + \frac{1}{2}x, & x \in [0, \frac{1}{k}] \\ \frac{1}{2}k^2x^3 + kx^2 + \frac{1}{2}x, & x \in [-\frac{1}{k}, 0] \end{cases} \qquad (20)$$

$S(x) \le x_+$, in $x = \pm 1/3k$, the difference value is the largest, $\max(x_+ - S(x)) = 2/27k$. In order to make $S(x)$ greater than x_+, the $S(x)$ as a whole moves up to $2/27k$ [12].

Then the approaching function of the positive function x_+ is

Journal of Theoretical and Applied Information Technology
15th October 2012. Vol. 44 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$$S_M(x,k)=\begin{cases} x & x\ge \frac{1}{3k} \\ -\frac{1}{2}k^2x^3+kx^2+\frac{1}{2}x+\frac{2}{27k}, & 0\le x<\frac{1}{3k} \\ \frac{1}{2}k^2x^3+kx^2+\frac{1}{2}x+\frac{2}{27k}, & -\frac{1}{3k}<x<0 \\ 0 & x\le -\frac{1}{3k} \quad k>0 \end{cases} \quad (21)$$

4.3 The polynomial approaching function $S^2_{M\varepsilon}$ of $|x|^2_\varepsilon$

Define $S^2_{M\varepsilon}(x,k)$ is the square of polynomial approaching function $S_{M\varepsilon}(x,k)$ for ε-insensitive loss function $|x|_\varepsilon$, from formula (21) we have following conclusion:

$$S^2_{M\varepsilon}(x,k)=\begin{cases} (x-\varepsilon)^2, & x\ge \frac{1}{3k}+\varepsilon \\ (-\frac{1}{2}k^2(x-\varepsilon)^3+k(x-\varepsilon)^2+\frac{1}{2}(x-\varepsilon)+\frac{2}{27k})^2, & \varepsilon<x<\frac{1}{3k}+\varepsilon \\ (\frac{1}{2}k^2(x-\varepsilon)^3+k(x-\varepsilon)^2+\frac{1}{2}(x-\varepsilon)+\frac{2}{27k})^2, & -\frac{1}{3k}+\varepsilon<x<\varepsilon \\ 0, & \frac{1}{3k}-\varepsilon\le x\le -\frac{1}{3k}+\varepsilon \\ (-\frac{1}{2}k^2(x+\varepsilon)^3+k(x+\varepsilon)^2-\frac{1}{2}(x+\varepsilon)+\frac{2}{27k})^2, & -\varepsilon<x<\frac{1}{3k}-\varepsilon \\ (\frac{1}{2}k^2(x+\varepsilon)^3+k(x+\varepsilon)^2-\frac{1}{2}(x+\varepsilon)+\frac{2}{27k})^2, & -\frac{1}{3k}-\varepsilon<x<-\varepsilon \\ (-x-\varepsilon)^2, & x\le -\frac{1}{k}-\varepsilon \end{cases} \quad (22)$$

5. PROPERTY ANALYSIS

Lemma 1[8] p^2_ε -function

$$p^2_\varepsilon(x,k)=(p(x-\varepsilon,k))^2+(p(-x-\varepsilon,k))^2,$$

where $p(x,k)=x+\frac{1}{k}\ \ln(1+e^{-kx})$, $k>0$, e is the base of natural logarithm, it has the following properties:

(1) p^2_ε -function is any-order smooth w.r.t. x;;

(2) $p^2_\varepsilon(x,k)\ge |x|^2_\varepsilon$;

(3) For $x\in R$ and $|x|<\rho+\varepsilon$:

$$p^2_\varepsilon(x,k)-|x|^2_\varepsilon\le 2(\frac{\log 2}{k})^2+\frac{2\rho}{k}\log 2 .$$

Lemma 2[13] S^2_ε -function

$$S^2_\varepsilon(x,k)=\begin{cases} (x-\varepsilon)^2, & x\ge \frac{1}{k}+\varepsilon \\ (\frac{1}{4}k(x-\varepsilon)^2+\frac{1}{2}(x-\varepsilon)+\frac{1}{4k})^2, & -\frac{1}{k}+\varepsilon<x<\frac{1}{k}+\varepsilon \\ 0, & \frac{1}{k}-\varepsilon\le x\le -\frac{1}{k}+\varepsilon \\ (\frac{1}{4}k(x+\varepsilon)^2-\frac{1}{2}(x+\varepsilon)+\frac{1}{4k})^2, & -\frac{1}{k}-\varepsilon<x<\frac{1}{k}-\varepsilon \\ (-x-\varepsilon)^2, & x\le -\frac{1}{k}-\varepsilon \end{cases} \quad (23)$$

has the following properties:

(1) S^2_ε is 1st-order smooth w.r.t. x. That is, at interpolation points,

$$S^2_\varepsilon(\pm(\tfrac{1}{k}+\varepsilon),k)=\frac{1}{k^2} \qquad S^2_\varepsilon(\pm(-\tfrac{1}{k}+\varepsilon),k)=0$$

$$\nabla S^2_\varepsilon(\pm(\frac{1}{k}+\varepsilon),k)=\frac{2}{k} \quad \nabla S^2_\varepsilon(\pm(-\frac{1}{k}+\varepsilon),k)=0;$$

(2) $S^2_\varepsilon(x,k)\ge |x|^2_\varepsilon$

Journal of Theoretical and Applied Information Technology
15th October 2012. Vol. 44 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

(3) $S_\varepsilon^2(x,k) - |x|_\varepsilon^2 \le \frac{1}{11k^2}$

Theorem 1 $S_{M\varepsilon}^2$ is defined in(16), we have:

(1) $S_{M\varepsilon}^2$ is 1st-order smooth w.r.t. x. That is, at interpolation points,

$$S_{M\varepsilon}^2(\pm(\tfrac{1}{3k}+\varepsilon),k) = \frac{1}{9k^2} \quad (24)$$

$$S_{M\varepsilon}^2(\pm\varepsilon,k) = \frac{4}{729k^2} \quad (25)$$

$$S_{M\varepsilon}^2(\pm(-\tfrac{1}{3k}+\varepsilon),k) = 0 \quad (26)$$

$$\nabla S_{M\varepsilon}^2(\pm(\frac{1}{3k}+\varepsilon),k) = \frac{2}{3k} \quad (27)$$

$$\nabla S_{M\varepsilon}^2(\pm\varepsilon,k) = \frac{2}{27k} \quad (28)$$

$$\nabla S_{M\varepsilon}^2(\pm(-\frac{1}{3k}+\varepsilon),k) = 0 \quad (29)$$

(2) $S_{M\varepsilon}^2(x,k) \ge |x|_\varepsilon^2$ (30)

(3) $S_{M\varepsilon}^2(x,k) - |x|_\varepsilon^2 \le \frac{1}{124k^2}$ (31)

Proof:

(1) According to the definition, we can directly prove it.

(2) Verify $S_{M\varepsilon}^2(x,k) \ge |x|_\varepsilon^2$

For $x \ge \frac{1}{k}+\varepsilon$, $x \le -\frac{1}{k}-\varepsilon$ and $-\varepsilon \le x \le \varepsilon$, Conclusion is obviously correct.

For $\varepsilon < x < \frac{1}{k}+\varepsilon$, define:

$g(x) = S_\varepsilon^2(x,k) - |x|_\varepsilon^2$, then we have

$$g(x) = (-\frac{1}{2}k^2(x-\varepsilon)^3 + k(x-\varepsilon)^2 + \frac{1}{2}(x-\varepsilon) + \frac{2}{27k})^2 - (x-\varepsilon)^2 = (-\frac{1}{2}k^2(x-\varepsilon)^3 + k(x-\varepsilon)^2 + \frac{3}{2}(x-\varepsilon) + \frac{2}{27k}) \bullet (-\frac{1}{2}k^2(x-\varepsilon)^3 + k(x-\varepsilon)^2 - \frac{1}{2}(x-\varepsilon) + \frac{2}{27k})$$

Define:

$$h_1(x) = -\frac{1}{2}k^2(x-\varepsilon)^3 + k(x-\varepsilon)^2 + \frac{3}{2}(x-\varepsilon) + \frac{2}{27k}$$

$$h_2(x) = -\frac{1}{2}k^2(x-\varepsilon)^3 + k(x-\varepsilon)^2 - \frac{1}{2}(x-\varepsilon) + \frac{2}{27k}$$

Then we get $h_1(\varepsilon) = \frac{4}{729k^2}$, for $\varepsilon < x < \frac{1}{3k}+\varepsilon$

$$\nabla h_1(x) = -\frac{3}{2}k^2(x-\varepsilon)^2 + 2k(x-\varepsilon) + \frac{3}{2} = \frac{3}{2}(1-k^2(x-\varepsilon)^2) + 2k(x-\varepsilon) > 0 .$$

So $h_1(x)$ is strictly monotonic increasing at region $\varepsilon < x < \frac{1}{3k}+\varepsilon$. So for $\varepsilon < x < \frac{1}{3k}+\varepsilon$, $h_1(x) > h_1(\varepsilon) = \frac{4}{729k^2} > 0$, so, $h_1(x) > 0$ Then we get $h_2(\frac{1}{3k}+\varepsilon) = 0$

For $\varepsilon < x < \frac{1}{3k}+\varepsilon$

$$\nabla h_2(x) = -\frac{3}{2}k^2(x-\varepsilon)^2 + 2k(x-\varepsilon) - \frac{1}{2} = -\frac{1}{2}(1-k(x-\varepsilon))(1-3k(x-\varepsilon)) < 0$$

So $h_2(x)$ is strictly monotonic decreasing at region $\varepsilon < x < \frac{1}{k}+\varepsilon$. So for $\varepsilon < x < \frac{1}{k}+\varepsilon$, $h_2(x) > h_2(\frac{1}{k}+\varepsilon) = 0$, so, $h_2(x) > 0$.

then $S_{M\varepsilon}^2(x,k) \ge |x|_\varepsilon^2$ is correct for $\varepsilon < x < \frac{1}{3k}+\varepsilon$.

Similarly, for the case of $-\frac{1}{3k}-\varepsilon < x < -\varepsilon$, we have $S_{M\varepsilon}^2(x,k) \ge |x|_\varepsilon^2$.

Hence, $S_{M\varepsilon}^2(x,k) \ge |x|_\varepsilon^2$

(3) Verify $S_{M\varepsilon}^2(x,k) - |x|_\varepsilon^2 \le \frac{1}{124k^2}$

For $x \ge \frac{1}{3k}+\varepsilon$, $x \le -\frac{1}{3k}-\varepsilon$ and $\frac{1}{3k}-\varepsilon \le x \le -\frac{1}{3k}+\varepsilon$, the conclusion is obviously correct.

For $-\frac{1}{3k}+\varepsilon < x < \varepsilon$,we have

$S_{M\varepsilon}^2(x,k) - |x|_\varepsilon^2 = S_{M\varepsilon}^2(x,k)$,due to $S_{M\varepsilon}^2$ is a strictly monotone increasing function for $-\frac{1}{3k}+\varepsilon < x < \frac{1}{3k}+\varepsilon$, so

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$S^2_{M\varepsilon}(x,k) \le S^2_{M\varepsilon}(\varepsilon,k) = \dfrac{4}{729k^2} < \dfrac{1}{124k^2}$;For $-\varepsilon < x < \frac{1}{3k} - \varepsilon$,we have

$S^2_{M\varepsilon}(x,k) - |x|^2_\varepsilon = S^2_{M\varepsilon}(x,k)$,due to $S^2_{M\varepsilon}$ is a strictly monotone decreasing function for $-\frac{1}{3k} - \varepsilon < x < \frac{1}{3k} - \varepsilon$,so

$S^2_{M\varepsilon}(x,k) \le S^2_{M\varepsilon}(\varepsilon,k) = \dfrac{4}{729k^2} < \dfrac{1}{124k^2}$; For $\varepsilon < x < \frac{1}{3k} + \varepsilon$

$g(x) = S^2_{M\varepsilon}(x,k) - |x|^2_\varepsilon$,we have variable substitution for formula:

$$g(x) = (-\frac{1}{2}k^2(x-\varepsilon)^3 + k(x-\varepsilon)^2 + \frac{1}{2}(x-\varepsilon) + \frac{2}{27k})^2 - (x-\varepsilon)^2$$

, $a = k(x-\varepsilon) \in (0,1)$,then $g(a) = (-\dfrac{1}{2k}a^3 + \dfrac{a^2}{k} + \dfrac{1}{2k}a + \dfrac{2}{27k})^2 - \dfrac{a^2}{k^2}$ has maximum value point a=0.0751 at region 0<a<1,hence $g(x) \le g(0.0751) < \dfrac{1}{124k^2}$,therefore, the conclusion is correct.

Similarly, for the case of $-\frac{1}{3_k} - \varepsilon < x < -\varepsilon$, we have $g(x) \le g(0.0751) < \dfrac{1}{124k^2}$.

Hence, $S^2_{M\varepsilon}(x,k) - |x|^2_\varepsilon \le \dfrac{1}{124k^2}$.

6. EXPERIMENTAL RESULT

In the case of k=5,ε=0.3, the smooth function approximation comparison chart is as Figure 3, then we can see that, $S^2_{M\varepsilon}$ -function has higher approximation accuracy than p^2_ε -function and S^2_ε -function with the same K value.

When p^2_ε -function and S^2_ε -function as smooth functions, define ρ=1/ k , from Lemma 1 and Lemma 2 we have $p^2_\varepsilon(x,k) - |x|^2_\varepsilon \le 1.3854/k^2$ and $S^2_\varepsilon(x,k) - |x|^2_\varepsilon \le 0.0909/k^2$, Table 2 list the approximation accuracy of three smooth functions, then we can see that the approximation accuracy of $S^2_{M\varepsilon}$ -function is three order of magnitude higher than that of the p^2_ε -function and one order of magnitude higher than that of the S^2_ε -function at the same K value.

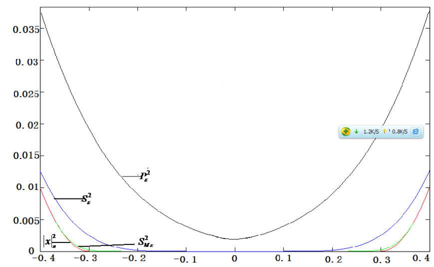

Figure 3: Smooth Function Approximation Comparison Chart In The Case Of K=5 And E=0.3

Table2: Approximation Accuracy Of Smooth Functions

smooth function	p^2_ε - function	S^2_ε - function	$S^2_{M\varepsilon}$ - function
approximation accuracy	$1.3854/k^2$	$0.0909/k^2$	$0.0081/k^2$

To further verify the property of this smooth function applying to support vector regression, Two simulated experiments were selected to demonstrate the analytical results, which were run at Matlab7.0 on a personal computer with an AMD X4 620 processor and 2GB memory. Based on the first order optimality conditions of unconstrained convex minimization problem, our stopping criterion was satisfied when the 2-norm of gradient of the objective function is less than 10^{-5} .For an observation vector y and the prediction vector $\hat{y}$, the 2-norm relative error of two vectors y and $\hat{y}$ was defined as follows:

$$\frac{\| y - \hat{y} \|_2}{\| y \|_2} \quad (32)$$

This relative error used to measure the accuracy of regression. In order to evaluate how well each method generalized to unseen data, we split the entire data set into two parts, the training set and testing set. The training data was used to generate the regression function that is learning from training data; the testing set, which is not involved in the training procedure, was used to evaluate the prediction ability of the resulting regression function. We also used a stratification scheme in splitting the entire data set to keep the "similarity" between training and testing data sets . That is, we tried to make the training set and the testing set have the similar observation distributions. A smaller testing error indicates better prediction

Journal of Theoretical and Applied Information Technology
15th October 2012. Vol. 44 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ability. We performed tenfold cross-validation on each data set and reported the average testing error in our numerical results. To generate a highly nonlinear function, a Gaussian kernel was used for all nonlinear numerical tests defined as below:

$$K(A_i, A_j^T) = e^{-\mu\|A_i - A_j\|_2^2}, i, j = 1, 2, 3, \cdots, m \quad (33)$$

The parameters μ and C were determined by a tuning procedure.

First, we selected 101 points evenly from [-1, 1] as the input data of the artificial data sets and the observation was generated from a simple function as follows:

$$f(x) = 0.5 \cdot \frac{\sin(\frac{30}{\pi}x)}{\frac{30}{\pi}x} + \rho, \quad (34)$$

Table 3: Numerical Result For Sin Function

Methods	#SVs	Train Error(%)	Test Error(%)	CPU (s)
ε-PSSVR	69	5.76	5.76	0.026
ε-MPSSVR	69	5.48	5.48	0.022

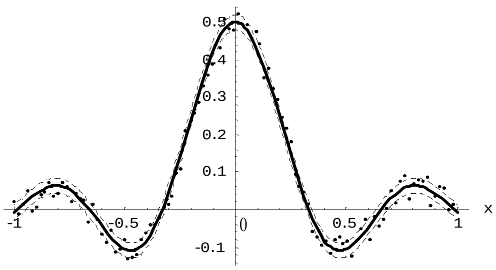

Figure 4: Regression Function Produced By Smooth Support Vector Regression

Where ρ is an additive Gaussian noise with mean=0 and standard deviation $\sigma = 0.04$ .We set $\varepsilon = 0.02$, which is one half standard deviation of the Gaussian noise. The rest of the parameters, $\mu = 33$ and C=6, were determined by a tuning procedure. The experimental results show that the ε-PSSVR has the smallest relative error. ε-PSSVR took 0.026 CPU seconds , while ε-MPSSVR took 0.022 seconds. We summarized the results in Figure 4 and Table 3.

The second artificial data set was obtained by using MATLAB command "peaks (170)" to generate 28900 data points in R2. Just like our first experiment, the Gaussian noises (mean=0 and standard deviation $\sigma = 0.4$) were added. Similar to the first experiment, we set $\varepsilon = 0.02$, $\mu = 1$ and C = 1000. Because of storing the fully dense kernel matrix required in nonlinear ε-PSSVR, will exceed the memory capacity and the reduced kernel technique was applied here. We randomly selected 300 points which are slightly over 10 percent of the entire training data set to form a reduced set and used the reduced kernel formulation to generate the nonlinear regression function. The resulting function of ε-PSSVR (a) and the original function (b, without noises) were shown in Figure 5. The dots that were shown in Figure 5 form the reduced set. This result was generated in 11.362 seconds with 0.01 relative error. We also tested ε-SSVR on his artificial data set. However, they took a much longer time to get the solution with the same level of accuracy. We summarized the numerical results of these two artificial data sets in Table4.

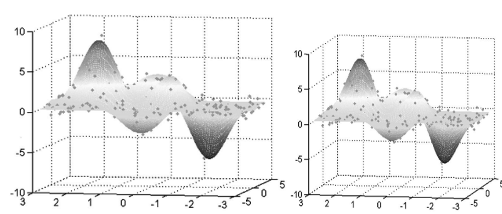

Figure 5: The Regression Of 3D Artificial Data Sets(A)(B)

Table4: Numerical Result For 3D Artificial Data Sets

Methods	#SVs	Train Error(%)	Test Error(%)	CPU(s)
ε-PSSVR	17820	1.08	1.12	12.086
ε-MPSSVR	17818	1.01	1.01	11.362

7. CONCLUSIONS

In this paper, we successfully obtain the polynomial smoothing function which approaches the square of ε-insensitive loss function by using three interpolation points cubic Spline interpolation method, that is $S^2_{M\varepsilon}$-function, and proved that this function has better properties, the approximation accuracy is three order of higher than P^2_ε -function and one order of higher than S^2_ε -function. As a result, to apply $S^2_{M\varepsilon}$ -function in support vector regression, the number of support vector is less, CPU time. Therefore, we can provide a new, better polynomial smooth function in smooth support

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

vector regression model fitting and other related fields.

ACKNOWLEDGEMENTS

This work was supported by the Guangdong Natural Science Foundation Project(No.S2011 010002144),The Ministry of Education, Guangdong Province, Production and Research Projects(No.2010B090400457, No.2011B090400 269 , No.2011A091000028),Guangdong Province Enterprise Laboratory Project(No.2011A091000 0 46) , the Natural Science Foundation of Dong Guan University of Technology Project (No.2010ZQ04).

REFERENCES:

[1] C. Chen and O. L. Mangasarian, "A class of smoothing functions for nonlinear and mixed complementarity problems", *Computational Optimization and Application*, Vol.5, No.5, 1996, pp.97-138.

[2] Y. B. Yuan, J. Yan and C. X. Xu, "Polynomial smooth support vector machine (PSSVM)", *Chinese Journal of Computers* , Vol. 28, No. 1, 2005, pp. 9-17.

[3] C. Chen and O. L. Mangasarian, "Smoothing methods for convex inequalities and linear complementarity problems", *Mathematical Programming*, Vol. 71, 1995, pp. 51-69.

[4] C. astroNeto, M. J. Youngseon, J. Myong K. et al, "AADT prediction using support vector regression with data-dependent parameters", *Expert Systems with Applications*, Vol. 36, No. 2: PART 2 , 2009, pp. 2979-2986.

[5] C. Andreas and V. M. Arnout, "Bouligand derivatives and robustness of support vector machines for regression", *Journal of Machine Learning Research*, Vol. 9, 2008, pp. 915-936.

[6] L. Mangasarian and D. R. Musicant, "Successive overrelaxation for support vector machines", *IEEE Transactions on Neural Networks*, Vol. 10, No. 8, 1999, pp. 1032-1037. Y. J. Lee and O. L. Mangasarian, "SSVM: A smooth support vector machine for classification", *Computational Optimization and Applications*, Vol. 22, No. 1, 2001, pp. 5-21.

[7] Y. J. Lee, W. F. Hsieh and C. M. Huang, "ε-SSVR: A smooth support vector machine for ε-insensitive regression", *IEEE Transactions on Knowledge and Data Engineering* , Vol. 17, No. 5, 2005, pp. 5-22.

[8] C. C. Chang and C. J. Lin, LIBSVM:A library for support vector machines, software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm, 2012.

[9] J Platt. Sequential minimal optimization: A fast algorithm for training support vector machines, *Advances in Kernel Methods-Support Vector Learning*. MIT Press, Cambridge, MA, 1999, PP.185-208.

[10] N. Y. Deng and Y. J. Tian. The new method in data mining-support vector machine, Press of science, 2004, pp. 244.

[11] H. Q. Yuan,W. G. Tu,J,Z. Xiong, et al, new polynomial smooth support vector machine, *Computer Science*, Vol. 38, No. 3, 2011, pp. 243-247.

[12] Ren Bin, Cheng Lianglun, "Polynomial smoothing support vector regression", *Journal of Control Theory and Applications*. Vol. 28, No. 2, 2011, pp. 261-265.