Proceedings of the Sixth International Conference on Machine Learning and Cybernetics, Hong Kong, 19-22 August 2007

SHRINKING THE TUBE: A NEW SUPPORT VECTOR REGRESSION ALGORITHM WITH PARAMETRIC INSENSITIVE MODEL

PEI-YI HAO

Department of Information Management, National Kaohsiung University of Applied Sciences, Kaohsiung, Taiwan
E-MAIL: haupy@cc.kuas.edu.tw

Abstract:

A new algorithm for Support Vector regression is described. For a priori chosen *v*, it automatically adjusts a flexible tube of arbitrary shape and minimal radius to include the data such that at most a fraction *v* of the data points lie outside. Moreover, it is shown how to use parametric tube shapes with non-constant radius. The algorithm is analysed theoretically and experimentally.

Keywords:

Support vector machines; Support vector regression; Interval regression; Insensitive model

1. Introduction

Support Vector (SV) machines comprise a new class of learning algorithms, motivated by results of statistical learning theory [1,5]. Originally developed for pattern recognition, they represent the decision boundary in terms of a typically small subset of all training examples, called the Support Vectors. In order for this property to carry over to the case of SV Regression, Vapnik devised the so-called ε-insensitive loss function [2, 5]:

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

which does not penalize errors below some ε>0, chosen a priori. His algorithm, which we will henceforth call ε-SVR, seeks to estimate functions $f(\mathbf{x}) = \langle \mathbf{w} \cdot \mathbf{x} \rangle + b$, $\mathbf{w}, \mathbf{x} \in R^n$, $b \in R$, based on data

$$(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N) \in R^n \times R \, ,$$

by minimizing the regularized risk functional

$$\frac{1}{2}\|\mathbf{w}\|^2 + C \cdot R_{emp}^{\varepsilon}$$

where C>0 is a constant determining the trade-off between minimizing training errors and minimizing the model complexity term $\|\mathbf{w}\|^2$, and

$$R_{emp}^{\varepsilon} := \frac{1}{N}\sum_{i=1}^{N} |y_i - f(\mathbf{x}_i)|_\varepsilon \, .$$

To motivate the new algorithm that shall be proposed, note that the parameter ε in ε-SVR can be useful if the desired accuracy of the approximation can be specified beforehand. In some cases, however, we want the estimate to be as accurate as possible without having to commit ourselves to a specific level of accuracy a priori [4]. Besides, the ε-insensitive zone in the ε-SVR is assumed to has a tube (or slab) shape. Namely, the radius of the insensitive zone is a user-predefined constant, and we do not care about the errors as long as they are inside the ε-insensitive zone. The selection of a parameter ε may seriously affect the modeling performance. In this paper, a new parametric insensitive loss function is proposed such that the corresponding insensitive zone of the proposed parametric-SVR can have arbitrary shape. This can be useful in situations where the noise is heteroscedastic, that is, where it depends on **x**. In addition, for a priori chosen *v*, the proposed parametric-SVR automatically adjusts a flexible insensitive zone of minimal radius to the data such that at most a fraction *v* of the data points lie outside the parametric-insensitive zone.

2. Support Vector Regression with Parametric Insensitive Model

By defining a new parametric-insensitive loss function, the *parametric*-SVR is derived to evaluate the interval regression model which automatically adjusts the interval to include all data. The parametric-insensitive loss function is defined by

$$|y - f(\mathbf{x})|_g := \max\{0, |y - f(\mathbf{x})| - g(\mathbf{x})\}$$

where f and g are real-valued functions on the a domain R^n, $\mathbf{x} \in R^n$ and $y \in R$. Following the concept of kernel-based learning, a non-linear function is learned by a linear learning machine in a kernel-introduced feature space while the capacity of the system is controlled by a parameter that does not depend on the dimensionality of the space. The basic idea is that a nonlinear regression function

1-4244-0973-X/07/$25.00 ©2007 IEEE

is achieved by simply mapping the input patterns $\mathbf{x}_i$ by Φ: $R^n \to$ F into a high-dimensional feature space F. Hence, the proposed parametric-SVR seeks to estimate the following two functions:

$f(\mathbf{x}) = \langle \mathbf{w} \cdot \Phi(\mathbf{x}) \rangle + b$, where $\mathbf{w} \in F$, $\mathbf{x} \in R^n$, $b \in R$,

$g(\mathbf{x}) = \langle \mathbf{c} \cdot \Phi(|\mathbf{x}|) \rangle + d$, where $\mathbf{c} \in F$, $\mathbf{x} \in R^n$, $d \in R$.

The problem of finding the **w**, **c**, b, and d that minimize the empirical risk

$$R_{emp}^{g}[f] = \frac{1}{N}\sum_{i=1}^{N} |y_i - f(\mathbf{x}_i)|_g$$

is equivalent to the following optimization problem:

$$\underset{\mathbf{w},\mathbf{c},b,d,\xi_i,\xi_i^*}{\text{minimize}} \ \frac{1}{2}\|\mathbf{w}\|^2 + C\left(v \cdot \left(\frac{1}{2}\|\mathbf{c}\|^2 + d \right) + \frac{1}{N}\sum_{i=1}^{N} (\xi_i + \xi_i^*) \right)$$

subject to

$$\left(\langle \mathbf{w} \cdot \Phi(\mathbf{x}_i) \rangle + b \right) + \left(\langle \mathbf{c} \cdot \Phi(|\mathbf{x}_i|) \rangle + d \right) \geq y_i - \xi_i \quad (1)$$

$$\left(\langle \mathbf{w} \cdot \Phi(\mathbf{x}_i) \rangle + b \right) - \left(\langle \mathbf{c} \cdot \Phi(|\mathbf{x}_i|) \rangle + d \right) \leq y_i + \xi_i^*$$

and $\xi_i, \xi_i^* \geq 0$ for i=1,…,N.

At each point $\mathbf{x}_i$, we allow an error of $g(\mathbf{x}_i)$. Everything above the parametric-insensitive zone $g(\mathbf{x}_i) = \langle \mathbf{c} \cdot \Phi(|\mathbf{x}_i|) \rangle + d$ is captured in slack variables ξ_i and ξ_i^*, which are penalized in the objective function via a regularization constant C, chosen a prior. The size of the parametric-insensitive zone, which is characterized by $\frac{1}{2}\|\mathbf{c}\|^2 + d$, is traded off against model complexity, which is characterized by $\|\mathbf{w}\|^2$, and slack variables via a constant $v > 0$, chosen a prior. We can find the solution of this optimization problem in dual variables by finding the saddle point of the Lagrangian:

$$L = \frac{1}{2}\|\mathbf{w}\|^2 + C\left(v \cdot \left(\frac{1}{2}\|\mathbf{c}\|^2 + d \right) + \frac{1}{N}\sum_{i=1}^{N} (\xi_i + \xi_i^*) \right)$$
$$- \sum_{i=1}^{N} \alpha_i \left[\left(\langle \mathbf{w} \cdot \Phi(\mathbf{x}_i) \rangle + b \right) + \left(\langle \mathbf{c} \cdot \Phi(|\mathbf{x}_i|) \rangle + d \right) - y_i + \xi_i \right]$$
$$- \sum_{i=1}^{N} \alpha_i^* \left[-\left(\langle \mathbf{w} \cdot \Phi(\mathbf{x}_i) \rangle + b \right) + \left(\langle \mathbf{c} \cdot \Phi(|\mathbf{x}_i|) \rangle + d \right) + y_i + \xi_i^* \right]$$
$$- \sum_{i=1}^{N} \beta_i \xi_i - \sum_{i=1}^{N} \beta_i^* \xi_i^*$$

where $\alpha_i, \alpha_i^*, \beta_i$, and β_i^* are the nonnegative Lagrange multipliers. Differentiating L with respect to **w**, **c**, b, d, ξ_i and ξ_i^* and setting the result to zero, we obtain:

$$\partial L / \partial \mathbf{w} = 0 \Rightarrow \quad \mathbf{w} = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) \Phi(\mathbf{x}_i), \quad (2)$$

$$\partial L / \partial \mathbf{c} = 0 \Rightarrow \quad \mathbf{c} = \frac{1}{C \cdot v} \sum_{i=1}^{N} (\alpha_i + \alpha_i^*) \Phi(|\mathbf{x}_i|), \quad (3)$$

$$\partial L / \partial b = 0 \Rightarrow \quad \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) = 0, \quad (4)$$

$$\partial L / \partial d = 0 \Rightarrow \quad \sum_{i=1}^{N} (\alpha_i + \alpha_i^*) = C \cdot v, \quad (5)$$

$$\partial L / \partial \xi_i = 0 \quad \Rightarrow \quad \alpha_i = \frac{C}{N} - \beta_i \text{ and } \alpha_i \leq \frac{C}{N}, \quad (6)$$

$$\partial L / \partial \xi_i^* = 0 \quad \Rightarrow \quad \alpha_i^* = \frac{C}{N} - \beta_i^* \text{ and } \alpha_i^* \leq \frac{C}{N}. \quad (7)$$

Substituting Eqs. (2)-(7) into L, we obtain the following dual problem

$$\max \quad \frac{-1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*) \langle \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j) \rangle$$
$$- \frac{1}{2Cv}\sum_{i=1}^{N}\sum_{j=1}^{N} (\alpha_i + \alpha_i^*)(\alpha_j + \alpha_j^*) \langle \Phi(|\mathbf{x}_i|) \cdot \Phi(|\mathbf{x}_j|) \rangle$$
$$+ \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) y_i$$

subject to

$$\sum_{i=1}^{N} (\alpha_i - \alpha_i^*) = 0, \quad \sum_{i=1}^{N} (\alpha_i + \alpha_i^*) = C \cdot v, \quad (8)$$
$$\alpha_i, \alpha_i^* \in \left[0, \frac{C}{N} \right].$$

Parameters b and d can be determined from the Karush-Kuhn-Tucker (KKT) conditions:

$$\alpha_i \left(\langle \mathbf{w} \cdot \Phi(\mathbf{x}_i) \rangle + b + \langle \mathbf{c} \cdot \Phi(|\mathbf{x}_i|) \rangle + d - y_i + \xi_i \right) = 0, \quad (9)$$

$$\alpha_i^* \left(-\langle \mathbf{w} \cdot \Phi(\mathbf{x}_i) \rangle - b + \langle \mathbf{c} \cdot \Phi(|\mathbf{x}_i|) \rangle + d + y_i + \xi_i^* \right) = 0, \quad (10)$$

$$\left(\frac{C}{N} - \alpha_i \right) \xi_i = 0 \quad \text{and} \quad \left(\frac{C}{N} - \alpha_i^* \right) \xi_i^* = 0. \quad (11)$$

For some $\alpha_i, \alpha_j^* \in (0, C/N)$, we have $\xi_i = \xi_j^* = 0$ and moreover the second factor in Eqs. (9) and (10) has to vanish. Hence, b and d can be computed as follows:

$$b = \frac{-1}{2}\left(\begin{array}{l} \langle \mathbf{w} \cdot \Phi(\mathbf{x}_i) \rangle + \langle \mathbf{w} \cdot \Phi(\mathbf{x}_j) \rangle \\ + \langle \mathbf{c} \cdot \Phi(|\mathbf{x}_i|) \rangle - \langle \mathbf{c} \cdot \Phi(|\mathbf{x}_j|) \rangle - y_i - y_j \end{array} \right), \quad (12)$$

$$d = \frac{-1}{2}\left(\begin{array}{l} \langle \mathbf{w} \cdot \Phi(\mathbf{x}_i) \rangle - \langle \mathbf{w} \cdot \Phi(\mathbf{x}_j) \rangle \\ + \langle \mathbf{c} \cdot \Phi(|\mathbf{x}_i|) \rangle + \langle \mathbf{c} \cdot \Phi(|\mathbf{x}_j|) \rangle - y_i + y_j \end{array} \right), \quad (13)$$

for some $\alpha_i, \alpha_j^* \in (0, C/N)$.

The functional form of mapping Φ does not need to be known since it is implicitly defined by the choice of kernel function $k(\mathbf{x},\mathbf{y}) \equiv \langle \Phi(\mathbf{x}) \cdot \Phi(\mathbf{y}) \rangle$. Hence it suffices to know and use $k(\mathbf{x}_i,\mathbf{x}_j) = \langle \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j) \rangle$ and $k(|\mathbf{x}_i|,|\mathbf{x}_j|) = \langle \Phi(|\mathbf{x}_i|) \cdot \Phi(|\mathbf{x}_j|) \rangle$ instead of defining $\Phi(\bullet)$ explicitly. Therefore, the upper bound and lower bound of the interval regression model are given as follows:

$$F^+(\mathbf{x}) = f(\mathbf{x}) + g(\mathbf{x}) = \left(\sum_{i=1}^{N} (\alpha_i - \alpha_i^*) k(\mathbf{x}_i \cdot \mathbf{x}) + b \right) + \left(\frac{1}{C \cdot v} \sum_{i=1}^{N} (\alpha_i + \alpha_i^*) k(|\mathbf{x}_i| \cdot |\mathbf{x}|) + d \right) \quad (14)$$

$$F_-(\mathbf{x}) = f(\mathbf{x}) - g(\mathbf{x}) = \left(\sum_{i=1}^{N} (\alpha_i - \alpha_i^*) k(\mathbf{x}_i \cdot \mathbf{x}) + b \right) - \left(\frac{1}{C \cdot v} \sum_{i=1}^{N} (\alpha_i + \alpha_i^*) k(|\mathbf{x}_i| \cdot |\mathbf{x}|) + d \right) \quad (15)$$

3. The Upper Bound on Number of Errors

The Karush-Kuhn-Tucker conditions make several useful conclusions to us. The training point $\mathbf{x}_i$ for which $\alpha_i^{(*)} > 0$ ($^{(*)}$ being a shorthand implying both the variables with and without asterisks) are termed support vectors (SVs) since only those points determine the final regression result among all training points. Here we have to distinguish the difference between the examples for which $0 < \alpha_i^{(*)} < C/N$, and those for which $\alpha_i^{(*)} = C/N$. In the first case, from condition (11), it follows $\xi_i^{(*)} = 0$ and moreover the second factor in Eq. (9) (or Eq. (10)) has to vanish. In other words, those examples $(\mathbf{x}_i, y_i)$ with corresponding $\alpha_i^{(*)} \in (0, C/N)$ lie on the upper bound (or lower bound) of the interval. In the second case, from condition (11), it follows $\xi_i^{(*)} > 0$. In other words, only examples $(\mathbf{x}_i, y_i)$ with corresponding $\alpha_i^{(*)} = C/N$ lie outside the interval around $f(\mathbf{x})$. Here, we will use the term *errors* to refer to the training points lying outside the data interval and the term fraction of error or SVs to denote the relative number of errors or SVs (i.e., divided by N). Now, let us analyze the theoretical aspects of the new optimization problem given in Eq. (8). The core aspect can be captured in the proposition stated below.

Proposition 1: *Suppose the parametric-SVR is applied to some data set, the following statements hold:*

v is an upper bound on the fraction of errors.

Proof.

The second and third constraints in the new optimization problem given in Eq. (8) imply that at most a fraction v of all training points can have $\alpha_i^{(*)} = C/N$. All training points with $\xi_i^{(*)} > 0$ certainly satisfy $\alpha_i^{(*)} = C/N$ (if not, $\alpha_i^{(*)}$ could grow further to reduce $\xi_i^{(*)}$).

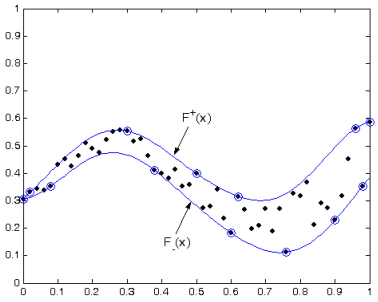
4. Experiments

In this section, two examples are used to verify the effectiveness of the proposed new support vector regression algorithm. The Gaussian kernel

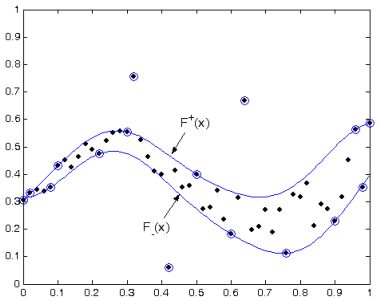
$$k(\mathbf{x},\mathbf{y}) = \exp\left(-\|\mathbf{x}-\mathbf{y}\|^2 / 2\sigma^2\right)$$

is used here. The optimal choice of parameters C, v and σ was tuned using a grid search mechanism. For the first example, the training data sets are generated by

$$y_k = 0.2\sin(2\pi x_k) + 0.2x_k^2 + 0.3 + (0.1x_k^2 + 0.05)e_k, \quad (16)$$
$$x_k = 0.02(k-1), \quad k = 1, 2, ..., 51,$$



(a)



(b)

Figure 1. (a) The training data generated by (16) without outliers and the result of proposed parameter-SVR, (b) The training data generated by (16) with three outliers and the result of proposed parameter-SVR.

where e_k represents a real number randomly generated in

the interval [−1; 1]. This example was also used in [3]. Figure 1(a) shows those 51 training data generated by Eq. (16) without outliers. Figure 1(b) shows the same 51 training data except that 3 of them are randomly selected and moved away from their original locations as outliers. Figures 1(a) and 1(b) show the curves of $F^{+}(\mathbf{x})$ and $F_{-}(\mathbf{x})$ obtained by the proposed parametric-SVR where the parameters (C, v, σ) were chosen as (300, 0.01, 0.18) and (20, 0.21, 0.18), respectively. The support vectors are marked with circles. As shown in figure 1(b) the estimates of the upper and lower bounds are not affected by outliers, and the proposed parametric-SVR performs very well.

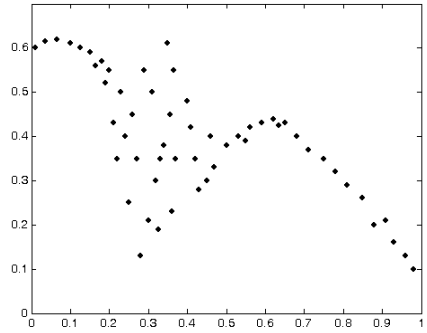


Figure 2. The data with heteroscedastic error structure

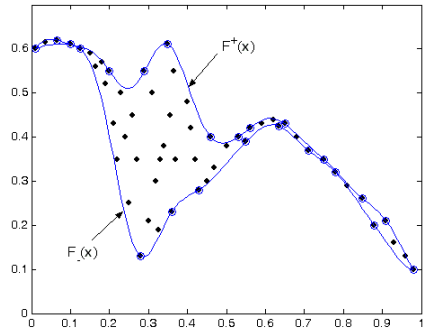


Figure 3. The result of proposed parameter-SVR for the data with heteroscedastic error structure

For the second example, the training data set is shown in Figure 2. This example was also used in [3]. As seen from Figure 2, the data have heteroscedastic error structure, i.e., the noise strongly depends on the input value **x**. Figure 3 shows the curves of $F^{+}(\mathbf{x})$ and $F_{-}(\mathbf{x})$ obtained by the proposed parametric-SVR where the parameters (C,v,σ) were chosen as (200, 0.1, 0.08). The experimental results show that parametric-SVR derives the satisfying solution to estimating interval bounds and captures well the characteristics of the data set.

5. Conclusions

In this paper, the new parametric-SVR is proposed to evaluate interval linear and nonlinear regression models for crisp input and output data. By utilizing a new parametric-insensitive loss function, the proposed parametric-SVR automatically adjusts a flexible parametric-insensitive zone of arbitrary shape and minimal radius to include all data. Moreover, the proposed method can achieve automatic accuracy control in the interval regression analysis task. For a priori chosen v, at most a fraction v of the data points lie outside the interval constructed by the proposed parametric-SVR. Experimental results have demonstrated the simplicity and effectiveness of the proposed method.

Acknowledgements

This work was partially supported by National Science Council Taiwan under grant. NSC 95-2221-E-151-037.

References

[1] C. Cortes, and V. N. Vapnik, “Support Vector Network,” *Machine learning*, Vol. 20, pp. 1-25, 1995.

[2] H. Drucker, C. Burges, L. Kaufman, A. Smola, and V. N. Vapnik, “Support Vector regression machines,” In *Advances in Neural Information Processing Systems* 9, vol. 9, pp. 155-161. The MIT Press, 1996.

[3] J.-T. Jeng, C.-C. Chuang, and S.-F. Su, “Support vector interval regression networks for interval regression analysis,” *Fuzzy Sets Syst.*, vol. 138, pp. 283–300, 2003.

[4] B. Schölkopf, A. J. Smola, R. Williamson, and P. L. Bartlett. “New support vector algorithms”, *Neural Computation*, vol. 12, no. 5, pp. 1207-1245, 2000.

[5] V.N. Vapnik, *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, 1995.