



A LAZY SUPPORT VECTOR REGRESSION MODEL FOR PREDICTION PROBLEMS WITH SMALL SAMPLE SIZE

X.Y. Fu*, H. Luo[†], G.Y. Zhang*, S.S. Zhong[†]

Abstract: Prediction problems with small sample size are problems which widely exist in engineering application. Because lazy prediction algorithms can utilize the information of predicted individual, it is often possible for them to achieve better predictive effect. Traditional lazy prediction algorithms generally use sample information directly, and therefore the predictive effect still has room for improvement. In this paper, we combine support vector regression (SVR) with lazy prediction algorithm, and propose a lazy support vector regression (LSVR) model. The insensitive loss function in LSVR depends on the distance between the individual in training sample set and the predicted individual. The smaller the distance, the smaller the lossless interval of the individual in training sample set, which means that the individual in training sample set has a great impact on the predicted individual. To solve the LSVR model, a generalized Lagrangian function is introduced to obtain the dual problem of the primal problem, and the solution to the primal problem is obtained by solving the dual problem. Finally, three numerical experiments are conducted to validate the predictive effect of LSVR. The experimental results show that the predictive effect of LSVR is better than those of ε -SVR, neural network (NN) and random forest (RF), and it is also better than that of k-nearest neighbor (k-NN) algorithm when the sample size is not too small and the distance between the predicted individual and the individual in training sample set is not too large. Therefore, LSVR not only has the advantage of good generalization ability of traditional SVR, but also has the advantage of good local accuracy of lazy prediction algorithm.

Key words: *lazy algorithm, support vector regression, prediction, small sample size*

Received: June 1, 2017

DOI: 10.14311/NNW.2019.29.003

Revised and accepted: March 11, 2019

1. Introduction

The prediction of the output for future input based on the limited number of input with known output is a problem which widely exists in engineering application.

*Xuyun Fu – Corresponding author; Guangyao Zhang; School of Naval Architecture and Ocean Engineering, Harbin Institute of Technology at Weihai, Weihai, China, E-mail: fuxuyun@hit.edu.cn, zhangguangyao23@163.com

[†]Hui Luo; Shisheng Zhong; School of Mechatronics Engineering, Harbin Institute of Technology, Harbin, China, E-mail: 15922551548@163.com, zhongss@hit.edu.cn

The input with known output is often referred to as a training sample set. For the problem, the current approaches can be divided into two categories. The first category is based on training sample set to establish a mapping between input space and output space, and then apply this mapping to predict the output for future input, such as a variety of regression in statistics [1, 2], neural network (NN) [3, 4], support vector regression (SVR) [5, 6] and random forest (RF) [7, 8] in machine learning, etc. Such methods are called eager prediction algorithms. The second category does not establish any mapping based on training sample set in advance. Only when the output for future input need to be predicted, a mapping between input space and output space will be established according to the relationship between the training sample set and future input, and then the output for future input will be predicted based on this mapping, such as k-nearest neighbors (k-NN) algorithm [9], local weighted regression [10, 11], etc. Those methods are called lazy prediction algorithms. It should be noted that the mapping established by lazy prediction algorithms is only applicable to the prediction of a single individual, and it is necessary to reestablish the mapping when other individuals need to be predicted.

The sample size of many prediction problems is small because of the natural sparseness of samples or the difficulty in sample collection. The essence of prediction problem with small sample size is lack of information, and the samples cannot fully reflect the mapping between input space and output space. In another word, the sample space is not complete. At present, there are mainly three approaches to deal with the prediction problem with small sample size. The first approach is directly using eager prediction algorithms with good generalization performance, such as SVR [12], extreme learning machines (ELM) [13], etc. These methods are limited by incomplete sample space, and there exists a large difference between the established global mapping and the actual mapping, and thus the predictive error is large. The second one is using virtual sample generation (VSG) technique to extend the sample size and then to predict [14, 15]. However, this method may produce wrong results because of the random data distribution. Moreover, when the data only has a small number of features, VSG may not be able to improve the prediction accuracy effectively. The third one is using lazy prediction algorithms to predict. When the predicted individual is close to the individual in training sample set, the lazy prediction algorithms are often able to obtain better results than the eager prediction algorithms. However, existing lazy prediction algorithms generally use sample information directly and simply.

For prediction problems with small sample size, the more information is used in modeling, the better the predictive effect may reach. Compared with eager prediction algorithms, lazy prediction algorithms not only use the sample information, but also use the predicted individual information. However, the existing lazy prediction algorithms cannot make full use of the sample information. The SVR model which is based on the VC dimension theory and the structural risk minimization can seek the best compromise between the model complexity and learning ability according to the limited sample information, and thus it has good generalization ability. Therefore, this paper introduces SVR into the lazy prediction algorithm, and proposes a lazy support vector regression (LSVR) model which is suitable for prediction problems with small sample size. By deducing the LSVR model, the

decision function and optimal solution of the model can be obtained. We test the model in several numerical experiments and compare its prediction accuracy with LSVR, ε -SVR, k-NN, NN and RF. We find that LSVR has the best prediction accuracy.

The remainder of the paper is organized as follows. Section 2 describes the establishment of the LSVR model. Section 3 introduces solution of the model. Section 4 sets up experiments to verify the predictive effect of LSVR and analyzes experimental results. Section 5 summarizes the results and concludes the paper.

2. Model Establishment

A training sample set $T = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_l, y_l)\}$ and a predicted individual (\mathbf{x}^*, y^*) are given, where l is the sample size, and \mathbf{x}_i is the input, $\mathbf{x}_i \in \mathbb{R}^n$, and y_i is the output, $y_i \in \mathbb{R}$, and $i = 1, 2, \dots, l$. The problem studied in this paper is how to predict y^* based on T and \mathbf{x}^* .

In this paper, we propose a LSVR model based on ε -SVR. In ε -SVR, ε -insensitive loss function is introduced to ensure its good generalization performance, denoted as

$$|y_i - f(\mathbf{x}_i)|_\varepsilon = \max\{0, |y_i - f(\mathbf{x}_i)| - \varepsilon\}, \quad (1)$$

where $f(\cdot)$ is the decision function, and ε is a small positive number. According to Eq. (1), ε -SVR allows ε deviation from the output y_i , as shown in Fig. 1.

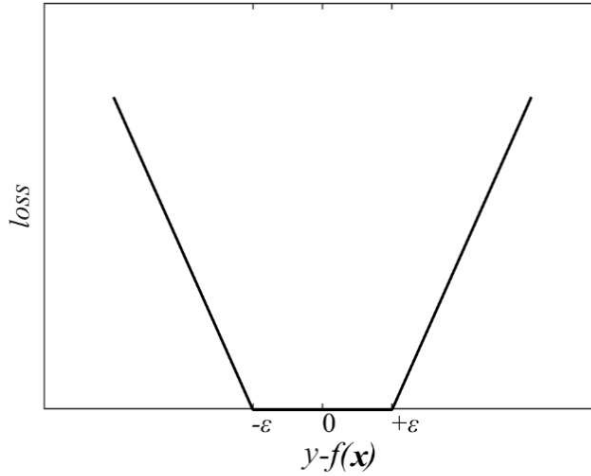


Fig. 1 ε -insensitive loss function.

The ε -SVR model aims at finding the appropriate decision function, denoted as

$$f(\mathbf{x}) = \omega \cdot \Phi(\mathbf{x}) + b, \quad (2)$$

fitting T and minimizing

$$\frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^l |y_i - f(\mathbf{x}_i)|_\varepsilon, \quad (3)$$

where ω is a normal vector, $\Phi(\cdot)$ represents the mapping function, b is denoted as the intercept, and $C(C > 0)$ is called penalty constant.

In ε -SVR, ε is a fixed number. This means that the allowable lossless intervals for all individuals in training sample set are equal, and that all individuals play exactly the same role in the process of building model. For a lazy prediction algorithm, its purpose is not to establish a perfect global mapping between input space and output space, but to establish a good local mapping near the predicted individual. In general, the closer the individual in sample is to the predicted individual, the better it can reflect the local mapping in vicinity of the predicted individual, and it should play a greater role in building the model, and vice versa. Thus, in LSVR, the allowable lossless intervals should be smaller for individuals in sample closer to the predicted individual. In order to establish a LSVR model, we firstly introduce an E-insensitive loss function, denoted as

$$|y_i - f(\mathbf{x}_i)|_E = \max\{0, |y_i - f(\mathbf{x}_i)| - E(\|\mathbf{x}_i - \mathbf{x}^*\|\}\}, \quad (4)$$

where $E(\cdot)$ is a function of the distance between the individual \mathbf{x}_i in sample and the predicted individual \mathbf{x}^* . The smaller $\|\mathbf{x}_i - \mathbf{x}^*\|$ is, the smaller $E(\|\mathbf{x}_i - \mathbf{x}^*\|)$ is, and vice versa. $E(\cdot)$ is generally defined as

$$E(\|\mathbf{x}_i - \mathbf{x}^*\|) = \frac{\varepsilon \|\mathbf{x}_i - \mathbf{x}^*\|}{\max_{i=1,2,\dots,l} \|\mathbf{x}_i - \mathbf{x}^*\|}, \quad (5)$$

where ε is a small positive number. According to Eq. (5), when $\|\mathbf{x}_i - \mathbf{x}^*\| = \max_{i=1,2,\dots,l} \|\mathbf{x}_i - \mathbf{x}^*\|$, $E(\|\mathbf{x}_i - \mathbf{x}^*\|) = \varepsilon$, and when $\|\mathbf{x}_i - \mathbf{x}^*\| = 0$, $E(\|\mathbf{x}_i - \mathbf{x}^*\|) = 0$.

When $E(\cdot)$ is defined as in Eq. (5), the E-insensitive loss function is shown in Fig. 2.

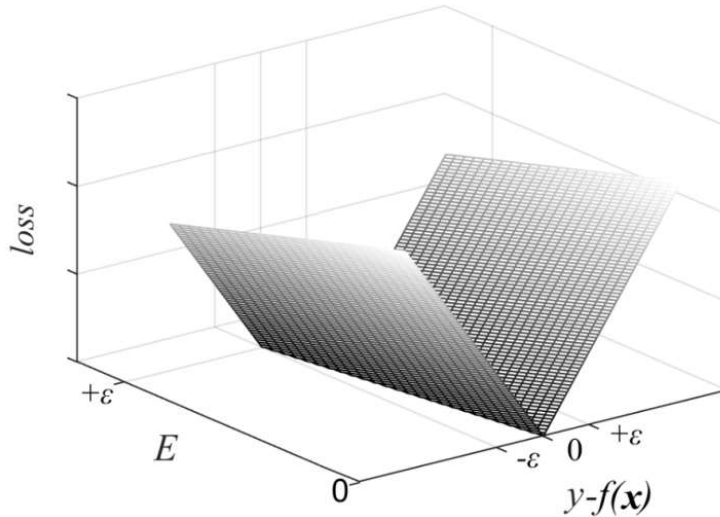


Fig. 2 E-insensitive loss function.

Similar to ε -SVR, the LSVR model also aims at finding an appropriate decision function, denoted as

$$f(\mathbf{x}) = \omega \cdot \Phi(\mathbf{x}) + b, \quad (6)$$

fitting T and minimizing

$$\frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^l |y_i - f(\mathbf{x}_i)|_{\mathbb{E}}. \quad (7)$$

Compared with Eq. (3), we can find that LSVR is exactly the same as ε -SVR except for the insensitive loss function.

3. Model Solution

Minimizing Eq. (7) is equivalent to solving the constrained optimization problem Eqs. (8)–(11).

$$\min_{\omega, \xi_i, \xi_i^*, b} \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^l (\xi_i + \xi_i^*) \quad (8)$$

$$\text{s.t. } y_i - \omega \cdot \Phi(\mathbf{x}_i) - b \leq \mathbb{E}(\|\mathbf{x}_i - \mathbf{x}^*\|) + \xi_i, \quad (9)$$

$$\omega \cdot \Phi(\mathbf{x}_i) + b - y_i \leq \mathbb{E}(\|\mathbf{x}_i - \mathbf{x}^*\|) + \xi_i^*, \quad (10)$$

$$\xi_i, \xi_i^* \geq 0, \quad i = 1, 2, \dots, l. \quad (11)$$

In order to solve this constrained optimization problem, a generalized Lagrange function is introduced, denoted as

$$\begin{aligned} L(\omega, \xi, \xi^*, b, \alpha, \alpha^*, \mu, \mu^*) &= \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^l (\xi_i + \xi_i^*) + \\ &\sum_{i=1}^l \alpha_i (y_i - \omega \cdot \Phi(\mathbf{x}_i) - b - \mathbb{E}(\|\mathbf{x}_i - \mathbf{x}^*\|) - \xi_i) + \\ &\sum_{i=1}^l \alpha_i^* (\omega \cdot \Phi(\mathbf{x}_i) + b - y_i - \mathbb{E}(\|\mathbf{x}_i - \mathbf{x}^*\|) - \xi_i^*) + \\ &\sum_{i=1}^l \mu_i (-\xi_i) + \sum_{i=1}^l \mu_i^* (-\xi_i^*) \end{aligned}, \quad (12)$$

where $\alpha_i, \alpha_i^*, \mu_i, \mu_i^* \geq 0, i = 1, 2, \dots, l$.

The minimax problem of Eq. (12) is denoted as

$$\min_{\omega, \xi_i, \xi_i^*, b} \max_{\alpha, \alpha^*, \mu, \mu^*} L(\omega, \xi, \xi^*, b, \alpha, \alpha^*, \mu, \mu^*) \quad (13)$$

$$\text{s.t. } \alpha_i, \alpha_i^*, \mu_i, \mu_i^* \geq 0, \quad i = 1, 2, \dots, l, \quad (14)$$

which is equivalent to Eqs. (8)–(11).

Therefore, the dual problem of Eqs. (8)–(11) is denoted as

$$\max_{\alpha, \alpha^*, \mu, \mu^*} \min_{\omega, \xi, \xi^*, b} L(\omega, \xi, \xi^*, b, \alpha, \alpha^*, \mu, \mu^*) \quad (15)$$

$$\text{s.t. } \alpha_i, \alpha_i^*, \mu_i, \mu_i^* \geq 0, \quad i = 1, 2, \dots, l. \quad (16)$$

The optimal solutions of the primal problem and dual problem are respectively represented as $(\bar{\omega}, \bar{\xi}, \bar{\xi}^*, \bar{b})$ and $(\bar{\alpha}, \bar{\alpha}^*, \bar{\mu}, \bar{\mu}^*)$. First, we solve $\bar{\alpha}$ and $\bar{\alpha}^*$. According to the Karush-Kuhn-Tucker (KKT) conditions, we can get

$$\nabla_{\omega} L(\bar{\omega}, \bar{\xi}, \bar{\xi}^*, \bar{b}, \bar{\alpha}, \bar{\alpha}^*, \bar{\mu}, \bar{\mu}^*) = \bar{\omega} - \sum_{i=1}^l \bar{\alpha}_i \Phi(\mathbf{x}_i) + \sum_{i=1}^l \bar{\alpha}_i^* \Phi(\mathbf{x}_i) = 0, \quad (17)$$

$$\nabla_{\xi_i} L(\bar{\omega}, \bar{\xi}, \bar{\xi}^*, \bar{b}, \bar{\alpha}, \bar{\alpha}^*, \bar{\mu}, \bar{\mu}^*) = C - \bar{\alpha}_i - \bar{\mu}_i = 0, \quad (18)$$

$$\nabla_{\xi_i^*} L(\bar{\omega}, \bar{\xi}, \bar{\xi}^*, \bar{b}, \bar{\alpha}, \bar{\alpha}^*, \bar{\mu}, \bar{\mu}^*) = C - \bar{\alpha}_i^* - \bar{\mu}_i^* = 0, \quad (19)$$

$$\nabla_b L(\bar{\omega}, \bar{\xi}, \bar{\xi}^*, \bar{b}, \bar{\alpha}, \bar{\alpha}^*, \bar{\mu}, \bar{\mu}^*) = - \sum_{i=1}^l \bar{\alpha}_i + \sum_{i=1}^l \bar{\alpha}_i^* = 0. \quad (20)$$

Eqs. (17)–(19) can be transformed into Eqs. (21)–(23), denoted as

$$\bar{\omega} = \sum_{i=1}^l \bar{\alpha}_i \Phi(\mathbf{x}_i) - \sum_{i=1}^l \bar{\alpha}_i^* \Phi(\mathbf{x}_i), \quad (21)$$

$$\bar{\mu}_i = C - \bar{\alpha}_i, \quad (22)$$

$$\bar{\mu}_i^* = C - \bar{\alpha}_i^*. \quad (23)$$

Put Eqs. (20)–(23) into Eqs. (15)–(16) and then simplify them, we can get

$$\min_{\alpha, \alpha^*} \left(\frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l (\alpha_i - \alpha_i^*) (\alpha_j - \alpha_j^*) \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j) - \sum_{i=1}^l (\alpha_i (y_i - E(\|\mathbf{x}_i - \mathbf{x}^*\|)) - \alpha_i^* (y_i + E(\|\mathbf{x}_i - \mathbf{x}^*\|))) \right) \quad (24)$$

$$s.t. \quad \sum_{i=1}^l (\alpha_i^* - \alpha_i) = 0, \quad (25)$$

$$0 \leq \alpha_i, \alpha_i^* \leq C, \quad i = 1, 2, \dots, l, \quad (26)$$

where $\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j) = K(\mathbf{x}_i, \mathbf{x}_j)$ is denoted as positive definite kernel function.

It is easy to prove that Eqs. (24)–(26) is a convex quadratic programming problem. The optimal solution $\bar{\alpha}$ and $\bar{\alpha}^*$ can be solved by interior point method or sequential minimal optimization method.

The process of solving \bar{b} is as follows. According to the KKT conditions, we can get

$$\bar{\alpha}_i (y_i - \bar{\omega} \cdot \Phi(\mathbf{x}_i) - \bar{b} - E(\|\mathbf{x}_i - \mathbf{x}^*\|) - \bar{\xi}_i) = 0, \quad (27)$$

$$\bar{\alpha}_i^* (\bar{\omega} \cdot \Phi(\mathbf{x}_i) + \bar{b} - y_i - E(\|\mathbf{x}_i - \mathbf{x}^*\|) - \bar{\xi}_i^*) = 0, \quad (28)$$

$$\bar{\mu}_i \bar{\xi}_i = 0, \quad (29)$$

$$\bar{\mu}_i^* \bar{\xi}_i^* = 0. \quad (30)$$

When $0 < \bar{\alpha}_i < C$, according to Eqs. (27) and (22), we can get

$$y_i - \bar{\omega} \cdot \Phi(\mathbf{x}_i) - \bar{b} - E(\|\mathbf{x}_i - \mathbf{x}^*\|) - \bar{\xi}_i = 0, \quad (31)$$

$$\bar{\mu}_i = C - \bar{\alpha}_i > 0. \quad (32)$$

And according to Eq. (29) we can also get

$$\bar{\xi}_i = 0. \quad (33)$$

Put Eqs. (21) and (33) into Eq. (31) and transform it, we can get

$$\bar{b} = y_i - \sum_{j=1}^l (\bar{\alpha}_j - \bar{\alpha}_j^*) K(\mathbf{x}_i, \mathbf{x}_j) - E(\|\mathbf{x}_i - \mathbf{x}^*\|). \quad (34)$$

Put $(\bar{\alpha}, \bar{\alpha}^*)$, Eqs. (21) and (34) into Eq. (6), we can obtain the final decision function

$$f(\mathbf{x}) = \sum_{j=1}^l (\bar{\alpha}_j - \bar{\alpha}_j^*) K(\mathbf{x}_j, \mathbf{x}) + \left[y_i - \sum_{j=1}^l (\bar{\alpha}_j - \bar{\alpha}_j^*) K(\mathbf{x}_i, \mathbf{x}_j) - E(\|\mathbf{x}_i - \mathbf{x}^*\|) \right]. \quad (35)$$

4. Numerical Experiment

In order to verify the predictive effect of LSVR, the sinc function $\text{sinc}(x) = \sin(x)/x$, which is shown in Fig. 3, is selected for three numerical experiments.

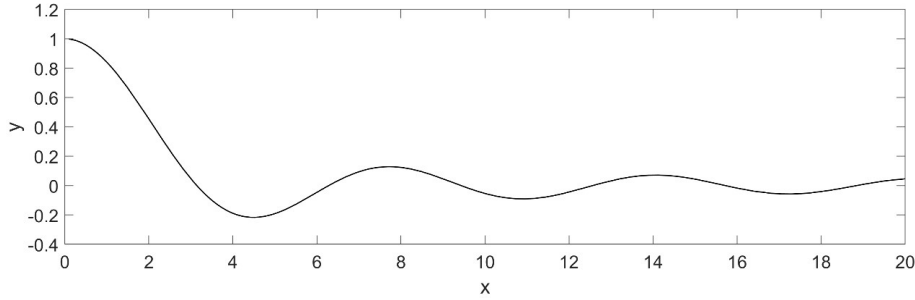


Fig. 3 Sinc function.

Experiment 1 the training sample set is $\{(x, \sin c(x)) | x = 1, 2, \dots, 20\}$, and its size is 20. The test sample set is $\{(x, \sin c(x)) | x = 0.1, 0.2, \dots, 2.0\}$, and its size is 200. LSVR, ε -SVR, k-NN, NN and RF are all used in the numerical experiment. For LSVR and ε -SVR, their penalty constant $C = 50000$, and the kernel function is Gaussian kernel function $K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right)$, where $\sigma = 0.25$. The ε in LSVR and ε -SVR is 0.1, 0.01, 0.001 and 0.0001 respectively. The k of k-NN is 1, 2, 3 and 4 respectively. The structure of NN is a multi-layer perceptron, which has 2 hidden layers with 10 nodes, and its activation function is the rectified linear unit (ReLU). For RF, the number of trees in the forest is 200, and the minimum number of samples at a leaf node is 8. In order to observe the impact of the distance d between the predicted individual and the individual in training

sample set on the predictive effect, we calculate the mean absolute error (MAE) for predicted individuals of $d \leq 0.2$, $d \leq 0.5$ and all the individuals in test sample set respectively.

Experiment 2 the training sample set is $\{(x, \sin c(x)) \mid x = 1, 3, 5, \dots, 19\}$, and its size is 10. The minimum number of samples at a leaf node in RF is 4. The rest is similar to Experiment 1.

Experiment 3 the training sample set is $\{(x, \sin c(x)) \mid x = 1, 5, 9, 13, 17\}$, and its size is 5. The structure of NN is also a multi-layer perceptron, which has 2 hidden layers with 9 nodes. The rest is similar to Experiment 2.

The results of all the experiments are listed in Tab. I. The best results of each method are shown in Fig. 4.

Based on the results, firstly, the prediction results of LSVR and ε -SVR are compared. For LSVR, when $\varepsilon = 0.01$, the MAE in experiments 1 and 2 is minimal, and when $\varepsilon = 0.1$, the MAE in experiment 3 is minimal. For ε -SVR, when $\varepsilon = 0.001$, the MAE in experiments 1 and 2 is minimal, and when $\varepsilon = 0.01$, the MAE in experiment 3 is minimal. It can be seen from Tab. I that LSVR obtains better predictive effect than ε -SVR for both the predicted individuals in the vicinity of training sample set and all predicted individuals in three experiments. This means that the LSVR model not only has good generalization ability of ε -SVR, but also improves the local accuracy because of using the information of predicted individuals.

Then, compared with k-NN, LSVR shows obvious advantages in the first two experiments. It can be clearly seen from Figs. 4(g) and (h) that k-NN cannot fit the sinc function exactly in experiments 1 and 2. In experiment 3, it can be seen from Figs. 4(c) and (i) that LSVR and k-NN cannot accurately track the trend of sinc function due to the serious shortage of training sample. For $d \leq 0.2$ and $d \leq 0.5$, the predictive effect of LSVR is better than k-NN. But for all the individuals in test sample set, the predictive effect of LSVR is worse than k-NN. This means that LSVR is more sensitive to the training sample size and the distance between predicted individual and training sample than k-NN. Therefore, when the training sample size is not too small or the distance between predicted individual and training sample is not too large, LSVR is more suitable. If not, k-NN may be more suitable.

Finally, compared with NN and RF, LSVR shows obvious advantages in three experiments. RF relies on a large number of different decision trees, but the problem we are discussing is a prediction problem with small sample size, and thus many decision trees are almost the same. Therefore, RF may not well deal with the prediction problem with small sample size. The model space of NN is directly related to the number of model parameters. In order to fit complex functions, NN must have many neurons, which means many model parameters need to be trained. Therefore, NN often requires a certain number of samples. When the sample size is small, NN is often difficult to achieve good prediction effect.

Prediction model	Parameter	MAE of experiment 1		MAE of experiment 2		MAE of experiment 3				
		d≤0.2	d≤0.5	all	d≤0.2	d≤0.5	all	d≤0.2	d≤0.5	all
LSVR	$\varepsilon = 0.1$	0.00098	0.00179	0.00253	0.00430	0.00976	0.01696	0.01314	0.03051	0.09447
	$\varepsilon = 0.01$	0.00042	0.00077	0.00161	0.00112	0.00263	0.00518	0.01484	0.03360	0.10780
	$\varepsilon = 0.001$	0.02209	0.02520	0.02729	0.00163	0.00395	0.00767	0.01502	0.03397	0.10926
	$\varepsilon = 0.0001$	0.00903	0.01349	0.01624	0.00169	0.00393	0.00772	0.01504	0.03401	0.10941
ε -SVR	$\varepsilon = 0.1$	0.06621	0.06588	0.06626	0.08786	0.08564	0.08246	0.09931	0.09703	0.12044
	$\varepsilon = 0.01$	0.00740	0.00730	0.00731	0.00993	0.00993	0.01029	0.01725	0.03381	0.10487
	$\varepsilon = 0.001$	0.00570	0.00639	0.00775	0.00193	0.00357	0.00669	0.01518	0.03395	0.10896
	$\varepsilon = 0.0001$	0.00874	0.00919	0.01372	0.00209	0.00404	0.00715	0.01506	0.03401	0.10938
k-NN	$k = 1$	0.01317	0.02527	0.02768	0.01360	0.03068	0.05319	0.01393	0.03114	0.10122
	$k = 2$	0.00554	0.00996	0.01499	0.01472	0.02472	0.03467	0.01715	0.03765	0.09196
	$k = 3$	0.01427	0.02369	0.03061	0.02520	0.04182	0.05692	0.01927	0.03937	0.07712
	$k = 4$	0.01760	0.02882	0.03733	0.02868	0.04843	0.06703	0.01962	0.03919	0.07063
NN	0.03986	0.04234	0.04439	0.03262	0.03937	0.04512	0.07169	0.08136	0.10784	
RF	0.12812	0.13106	0.14798	0.13952	0.14419	0.14808	0.22681	0.22837	0.14851	

Tab. I Predictive effect of each method in each experiment.

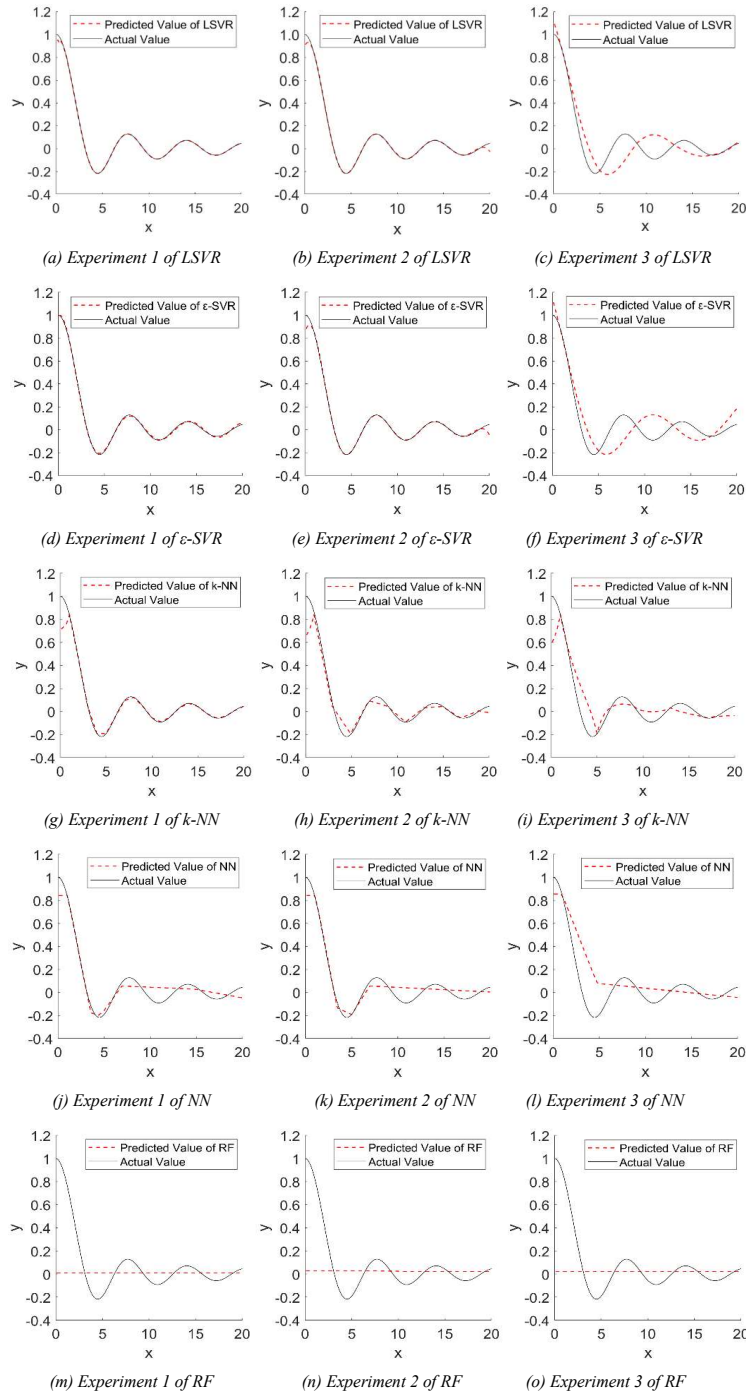


Fig. 4 Best Prediction results of each method in each experiment.

5. Conclusion

In order to improve the predictive effect with small sample size, this paper combines ε -SVR and lazy prediction algorithm to establish a LSVR model. The insensitive loss function in LSVR depends on the distance between the individual in sample and the predicted individual. To solve the LSVR model, a generalized Lagrangian function is introduced to obtain the dual problem of the primal problem. The solution to the primal problem is obtained by solving the dual problem. Finally, three numerical experiments were conducted using sinc function. The experimental results show that the predictive effect of LSVR is better than those of ε -SVR, NN and RF. When training sample size is not too small and the distance between the predicted individual and the individual in training sample set is not too large, the predictive effect of LSVR is also better than that of k-NN. Therefore, the LSVR model can integrate the advantages of traditional SVR and lazy prediction algorithm. It not only has better generalization ability, but also improves local accuracy.

Despite the era of big data is coming, many small enterprises are still far from being overwhelmed by big data. The proposed LSVR model can well deal with the prediction problems such as insufficient accumulation of historical data, high data acquisition cost and long data acquisition time, etc.

Acknowledgement

The authors would like to thank the editors and the anonymous reviewers for their valuable comments and constructive suggestions. This work is supported by the Key National Natural Science Foundation of China (No. U1533202) and the Fundamental Research Funds for the Central Universities (No. HIT.NSRIF.201704).

References

- [1] FERNANDO V. A comparative analysis between two techniques for the prediction of software defects: fuzzy and statistical linear regression. *Innovations in Systems and Software Engineering*. 2015, 11(4), pp. 277–287, doi: [10.1007/s11334-015-0256-4](https://doi.org/10.1007/s11334-015-0256-4).
- [2] LIN T., TSAI G. C. A Simple Linear Regression Approach to Modeling and Forecasting Mortality Rates. *Journal of Forecasting*. 2015, 34(7), pp. 543–559, doi: [10.1002/for.2353](https://doi.org/10.1002/for.2353).
- [3] MADHAVAN P. G. New recurrent neural network learning algorithm for time series prediction. *Journal of Intelligent Systems*. 1997, 7(1-2), pp. 103–116, doi: [10.1515/JISYS.1997.7.1-2.103](https://doi.org/10.1515/JISYS.1997.7.1-2.103).
- [4] DING G., ZHONG S. Time series prediction by parallel feedforward process neural network with time-varied input and output functions. *Neural Network World*. 2005, 15(2), pp. 137–147.
- [5] ZHAO W., TAO T., DING Z., ZIO E. A dynamic particle filter-support vector regression method for reliability prediction. *Reliability Engineering and System Safety*. 2013, 119, pp. 109–116, doi: [10.1016/j.ress.2013.05.021](https://doi.org/10.1016/j.ress.2013.05.021).
- [6] ZHAO X., WANG G., ZHAO K., TAN D. On-line least squares support vector machine algorithm in gas prediction. *Mining Science and Technology*. 2009, 19(2), pp. 194–198, doi: [10.1016/S1674-5264\(09\)60037-5](https://doi.org/10.1016/S1674-5264(09)60037-5).
- [7] GONG H. R., SUN Y. R., SHU. X., HUANG B. S. Use of random forests regression for predicting IRI of asphalt pavements. *Construction and Building Materials*. 2018, 189, pp. 890–897, doi: [10.1016/j.conbuildmat.2018.09.017](https://doi.org/10.1016/j.conbuildmat.2018.09.017).

- [8] PEYMAN Z., SAEID P., ALIREZA A., BRENTON S. M., SIMACK A. S., BRETT A. M., Random forest regression prediction of solid particle Erosion in elbows. *Powder Technology*. 2018, 338, pp. 983–992, doi: [10.1016/j.powtec.2018.07.055](https://doi.org/10.1016/j.powtec.2018.07.055).
- [9] ALTMAN N. S. An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression. *The American Statistician*. 1992, 46(3), pp. 175–185, doi: [10.1080/00031305.1992.10475879](https://doi.org/10.1080/00031305.1992.10475879).
- [10] CENTNER V., MASSART D. L. Optimization in locally weighted regression. *Analytical Chemistry*. 1998, 70(19), pp. 4206–4211, doi: [10.1021/ac980208r](https://doi.org/10.1021/ac980208r).
- [11] ZHANG J., CHUNG C. Y., HAN Y. Online Damping Ratio Prediction Using Locally Weighted Linear Regression. *IEEE Transactions on Power Systems*. 2016, 31(3), pp. 1954–1962, doi: [10.1109/TPWRS.2015.2448104](https://doi.org/10.1109/TPWRS.2015.2448104).
- [12] AL-ANAZI A. F., GATES I. D. Support vector regression to predict porosity and permeability: Effect of sample size. *Computers and Geosciences*. 2012, 39, pp. 64–76, doi: [10.1016/j.cageo.2011.06.011](https://doi.org/10.1016/j.cageo.2011.06.011).
- [13] SOVILJ D., BJÖRK K., LENDASSE A. Comparison of combining methods using Extreme Learning Machines under small sample scenario. *Neurocomputing*. 2016, 174, pp. 4–17, doi: [10.1016/j.neucom.2015.03.109](https://doi.org/10.1016/j.neucom.2015.03.109).
- [14] LI D., WEN I. A genetic algorithm-based virtual sample generation technique to improve small data set learning. *Neurocomputing*. 2014, 143, pp. 222–230, doi: [10.1016/j.neucom.2014.06.004](https://doi.org/10.1016/j.neucom.2014.06.004).
- [15] CHEN H., LI D. Generating Information-diffusion-based Virtual Samples to Improve Small Data Set Prediction for Ceramic Powder: A Case Study. In: *Proceedings – 3rd International Conference on Applied Computing and Information Technology and 2nd International Conference on Computational Science and Intelligence*, Okayama, Japan. New York: Institute of Electrical and Electronics Engineers Inc, 2015, pp. 374–378, doi: [10.1109/ACIT-CSI.2015.71](https://doi.org/10.1109/ACIT-CSI.2015.71).