A working likelihood approach to support vector regression with a data-driven insensitivity parameter

Jinran Wu and You-Gan Wang[*]

School of Mathematical Sciences, Queensland University of Technology, Australia.

February 7, 2020

Abstract

The insensitive parameter in support vector regression determines the set of support vectors that greatly impacts the prediction. A data-driven approach is proposed to determine an approximate value for this insensitive parameter by minimizing a generalized loss function originating from the likelihood principle. This data-driven support vector regression also statistically standardizes samples using the scale of noises. Nonlinear and linear numerical simulations with three types of noises (ϵ-Laplacian distribution, normal distribution, and uniform distribution), and in addition, five real benchmark data sets, are used to test the capacity of the proposed method. Based on all of the simulations and the five case studies, the proposed support vector regression using a working likelihood, data-driven insensitive parameter is superior and has lower computational costs.

Keywords: Approximate loss function; Parameter estimation; Prediction; Working likelihood

1 Introduction

In the machine learning field, support vector regression (SVR) has been popular in management and engineering applications (Trafalis & Ince, 2000; Mohandes et al., 2004; Vrablecová et al., 2018; Li et al., 2017), due to its solid theoretical foundation (Vapnik et al., 1997; Chang & Lin, 2011, 2002) and insensitivity to the dimensionality of the samples (Drucker et al., 1997). As recommended by Vapnik (2013), the parameter settings in SVR modeling contribute the generalization of the predictive performance. However, practitioners applying SVR in real-world applications often cannot obtain the most effective model. There are two key approaches to setting the hyper-parameter. One option is to use the k-cross validation to choose the parameters for SVR (Hastie et al., 2005; Ito & Nakano, 2003). The other approach is to set the parameter as a constant, based on the empirical practice developed by Chang & Lin (2011). In particular, the researchers suggested that the regularization parameter C and the insensitive parameter ϵ be set at 1.0 and 0.1, respectively.

[*] Corresponding author.

E-mail addresses: jinran.wu@hdr.qut.edu.au (J. Wu); you-gan.wang@qut.edu.au (Y-G. Wang).

However, although the tuning parameter setting provides an acceptable generalization in most conditions, there is still a huge gap between this solution and the best SVR using the optimal parameters.

For the insensitive parameter ϵ that controls the number of support vectors Schölkopf et al. (1999), Schölkopf et al. (2000) used the parameter ν to effectively control the number of support vectors to eliminate the free parameter, ϵ. However, one drawback is that the choice of ν has an impact on the generalization of the model (Schölkopf et al., 1998). Furthermore, insensitive parameter estimation methods that consider the noises in observations have been developed. Jeng et al. (2003) proposed to estimate the insensitive parameter in two steps. The first step is to estimate the regression errors by the SVR at $\epsilon = 0$. Then, the ϵ value is updated by $c\hat{\sigma}$ with an empirical constant c and the estimated standard deviation of the noise $\hat{\sigma}$. In the absence of outliers, the standard deviation can be calculated based on all the regression errors, and c is set as 1.98. Otherwise, a trimmed estimator is obtained by removing $5 - 10\%$ of samples at both ends to achieve robustness, and c is recommended to be fixed at 3. Obviously, although Jeng et al. (2003)'s method aims to incorporate data size in the estimation, the empirical settings make the method unable to recognize the noise level to estimate the insensitive parameter ϵ. Similar to Jeng et al. (2003)'s method, Cherkassky & Ma (2004) incorporated sample size into the insensitive parameter estimation. As explored by them, the empirical formulation for $\hat{\epsilon}$ is calculated by the product of the empirical constant 3, the standard deviation of the noise, and an empirical coefficient $\sqrt{\ln n/n}$ (n is the sample size). However, when the sample size increases, this $\hat{\epsilon}$ would approach to 0, so this method does not recognize the noise level for the insensitive parameter estimation.

As explained by Vapnik (2013), the insensitive loss function consists of the least modulus (LM) loss and the special Huber loss function when $\epsilon = 0$. Hence, in our study, considering the insensitive Laplacian distribution loss function inspired by Vapnik et al. (1997) and Bartlett et al. (2002), we focused on the insensitive parameter ϵ and propose a novel SVR with a data-driven (D-D) insensitive parameter. Similar to Jeng et al. (2003) and Cherkassky & Ma (2004)'s work, our method is developed on the theoretical background of SVR instead of parameter estimation based on re-sampling. Motivated by Wang et al. (2007), we propose designating the working likelihood to estimate the insensitive parameter for SVR. In other words, the working likelihood method can estimate appropriate hyper-parameters to find the most appropriate ϵ-Laplacian distribution to the real noise distribution. Our working likelihood (or D-D) method works as a vehicle for the ϵ loss function parameter estimation. In addition, different from the computational standardization, the target in the proposed model is standardized in a statistical manner using the scale of the noise. Thus, our D-D method is more practicable and intelligent. In our simulations (linear and nonlinear), three types of error distributions were used to test the D-D insensitive parameter estimation, namely, the insensitive Laplacian distribution, normal distribution, and uniform distribution. Furthermore, some case studies were applied to validate that our D-D SVR has novel generalization in real applications.

This rest of this paper is organized as follows. Section 2 reviews the framework of SVR and outlines the working likelihood for insensitive parameter estimation in SVR. Numerical simulations for three different types of noise sources (the insensitive Laplacian distribution, normal distribution, and uniform distribution) were implemented, and Section 3 presents a discussion of the analyses of the simulation results, which proved the efficiency of the working likelihood. Then, in Section 4, we discuss the validation of our D-D SVR on five real data sets: energy efficiency, Boston housing, yacht hydrodynamics, airfoil self-noise, and concrete compressive strength. Finally, in Section 5, we summarize the results that indicate the working likelihood (D-D) method has superior performance on insensitive parameter estimation based on the real noise information in SVR, indicating that our D-D SVR is very promising for predictions.

2 Data-driven support vector regression (SVR)

2.1 The framework of SVR

Assume the training data $(x_1,\ y_1), ..., (x_n,\ y_n) \in \chi \times \mathbb{R}$, where χ denotes the space of the input patterns. In ϵ-SVR, the target is to obtain a function $f(x)$ that has at most ϵ deviation from the actual obtained target y_i for all the training data, and at the same time, is as flat as possible (Smola & Schölkopf, 2004; Drucker et al., 1997). This means that smaller errors ($\leq \epsilon$) are ignored, and larger errors will be accounted for in the loss function. The case of linear function $f(\cdot)$ can be formed as

$$f(x) = \langle \omega, x \rangle + b \quad \omega \in \chi, b \in \mathbb{R}, \tag{1}$$

where $\langle ., . \rangle$ represents the dot product in χ. Flatness in Eq. (1) means finding a small ϵ. Now we are interested in minimizing the Euclidean norm, meaning $\|\omega\|^2$, which can be expressed with a convex optimization problem as (Smola & Schölkopf, 2004),

$$\begin{aligned} &\text{minimize} \quad \frac{1}{2}\|\omega\|^2 \\ &\text{subject to} \quad \begin{cases} y_i - \langle \omega, x_i \rangle - b \leqslant \epsilon, \\ \langle \omega, x_i \rangle + b - y_i \leqslant \epsilon. \end{cases} \end{aligned} \tag{2}$$

Here, the optimization problem is feasible; it means that there exists such a function f that approximates all pairs $(x_i,\ y_i)$ with ϵ precision. Then, the slack variables ξ_i and ξ_i^* are introduced to cope with the otherwise infeasible constraints of the optimization version in Eq. (2). Hence, the formulation is shown as,

$$\begin{aligned} &\text{minimize} \quad \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \\ &\text{subject to} \quad \begin{cases} y_i - \langle \omega, x_i \rangle - b \leqslant \epsilon + \xi_i, \\ \langle \omega, x_i \rangle + b - y_i \leqslant \epsilon + \xi_i^*, \\ \xi_i, \xi_i^* \geqslant 0. \end{cases} \end{aligned} \tag{3}$$

The regularization parameter C (a positive constant) determines the trade-off between the flatness of f and the amount up to which deviations are larger than ϵ. The optimization problem can be transformed to its dual problem as follows (Smola & Schölkopf, 2004):

$$
\begin{aligned}
\text{maximize} \quad & -\frac{1}{2}\sum_{i,j=1}^{n}(\alpha_i-\alpha_i^*)(\alpha_j-\alpha_j^*)\langle x_i,x_j\rangle \\
& -\epsilon\sum_{i=1}^{n}(\alpha_i+\alpha_i^*)+\sum_{i=1}^{n}y_i(\alpha_i-\alpha_i^*) \\
\text{subject to} \quad & \begin{cases} \sum_{i=1}^{n}(\alpha_i-\alpha_i^*)=0, \\ \alpha_i,\alpha_i^*\in[0,C]. \end{cases}
\end{aligned}
\tag{4}
$$

Here, α_i and α_i^* are Lagrange multipliers for $\epsilon+\xi_i-y_i+\langle\omega,x_i\rangle+b$ and $\epsilon+\xi_i^*-\langle\omega,x_i\rangle-b+y_i$, respectively. This dual optimization has a general solution,

$$
f(x)=\sum_{i=1}^{n}(\alpha_i-\alpha_i^*)k(x_i,x)+b, \tag{5}
$$

where the dual optimization is subjected to the constraints $0\leqslant\alpha_i,\alpha_i^*\leqslant C$, and $k(x_i,x)$ is the kernel function including linear function as a special case.

As illustrated by Vapnik (2013), three important parameter settings in SVR significantly impact the model's generalization: the regularization parameter C, the kernel parameters, and the insensitive parameter ϵ. The first one, C, can be estimated by the 0.95 quantile of $|y_i|$ (Cherkassky & Ma, 2004),

$$
C_{CM}=|y_i|_{(0.95)}, i=1,...,n. \tag{6}
$$

Then, the second kernel parameter is applied to adjust the mapping from the original space to the high-dimensional space; this is decided by the type of kernel function and the application domain. The last one is the most important parameter, ϵ, which controls the number of support vectors. We will explore how to estimate the insensitive parameter ϵ based on the loss function mechanism from a statistical perspective in the next section.

2.2 Working likelihood for insensitive parameter estimation

Suppose the training data set consists of n samples $(x_i,y_i),(i=1,2,...,n)$, and the target y_i, is generated from the following model:

$$
y_i=f(x_i)+su_i, \tag{7}
$$

where $f(\cdot)$ represents the expected value, while the second component, su_i (which is denoted by U_i) is the noise (s is the scale, and u_i is the noise after scaling s).

In ϵ-SVR, the loss function is defined as

$$
\begin{aligned}
V(u) &= |u|_\epsilon, \\
&= \begin{cases} u-\epsilon & u>\epsilon, \\ 0 & -\epsilon\leqslant u\leqslant\epsilon, \\ -u-\epsilon & u<-\epsilon, \end{cases}
\end{aligned}
\tag{8}
$$

where $u = y - \langle \omega, x \rangle - b$ is the residual item. The corresponding density function for u_i is,

$$g(u; \epsilon) = \frac{1}{2(1+\epsilon)} \exp(-|u|_\epsilon), \tag{9}$$

which will correspond to the loss function given by Eq. (8) up to a constant.

Thus, suppose that all u_i are identically and independently distributed with a density function $g(\cdot)$. Let θ be a vector collecting all the unknown parameters (ϵ, s). The negative log-likelihood based on the training data is then

$$L(\theta) = -\sum_{i=1}^{n} \log\left(g\left(\frac{y_i - f(x_i)}{s} \right) \right) + l \log(s). \tag{10}$$

Once the SVR approach is adapted, we essentially assume u_i follows a density function that is proportional to $\exp(-V(u))$. Our working likelihood D-D method estimates all the parameters in θ by maximizing $L(\theta)$. In particular, we investigate the choice of the insensitivity parameter ϵ in the SVR approach. Clearly, the ϵ value that results by maximizing L is data dependent and expected to be more effective. Meanwhile, the scale of the noise s can also be estimated.

Next, recalling that $U_i = su_i$, assume that $U_1, U_2, \dots U_n$ are independent and identically distributed random variables. Denote $(\epsilon, s) = \theta$. Their joint working likelihood function is

$$L(\theta) = \prod_{i=1}^{n} \left(\frac{1}{s} g(\frac{U_i}{s}; \epsilon, s) \right) = \left(\frac{1}{s} \right)^n \cdot \left(\frac{1}{2(1+\epsilon)} \right)^n \cdot \exp\left(-\sum_{i=1}^{n} |\frac{U_i}{s}|_\epsilon \right). \tag{11}$$

Therefore, $L(\theta)$ is a likelihood function with parameters ϵ and s properly regularized. Their estimators can thus be achieved by minimizing the negative log-likelihood function,

$$\begin{aligned} -\log L(\theta) &= n \log s + n \log\left[2(1+\epsilon)\right] + \sum_{i=1}^{n} |\frac{U_i}{s}|_\epsilon \\ &= n \log s + n \log\left(2(1+\epsilon)\right) + \sum_{i=1}^{n} \left((\frac{U_i}{s} - \epsilon) \cdot \mathbb{I}(\frac{U_i}{s} > \epsilon) + (-\frac{U_i}{s} - \epsilon) \cdot \mathbb{I}(\frac{U_i}{s} < -\epsilon) \right). \end{aligned} \tag{12}$$

The derivatives of $(-\log L(\theta))$ with respect to ϵ and s are given as

$$\begin{cases} \dfrac{\partial\left(-\log L(\theta)\right)}{\partial \epsilon} = \dfrac{n}{1+\epsilon} - \displaystyle\sum_{i=1}^{n} \mathbb{I}(|\frac{U_i}{s}| > \epsilon), \\ \dfrac{\partial\left(-\log L(\theta)\right)}{\partial s} = \dfrac{n}{s} - \dfrac{1}{s^2} \displaystyle\sum_{i=1}^{n} |U_i| \cdot \mathbb{I}(|\frac{U_i}{s}| > \epsilon). \end{cases} \tag{13}$$

By equating them to 0, both parameters (ϵ, s) can be expressed as,

$$\begin{cases} \epsilon = \dfrac{\displaystyle\sum_{i=1}^{n} \mathbb{I}(|\frac{U_i}{s}| \leqslant \epsilon)}{\displaystyle\sum_{i=1}^{n} \mathbb{I}(|\frac{U_i}{s}| > \epsilon)}, \\ s = \dfrac{\displaystyle\sum_{i=1}^{n} |U_i| \cdot \mathbb{I}(|\frac{U_i}{s}| > \epsilon)}{n}. \end{cases} \tag{14}$$

Thus, the parameters ϵ and s can be estimated by minimizing Eq. (12) or calculating the root of Eq. (14). In addition, the meaning of (ϵ, s) now becomes clear. This indicates that ϵ is the odds ratio of being inside the box ($\leqslant \epsilon$) versus outside the box ($\geqslant \epsilon$). The parameter s is the average distance of the support vectors, while the distance of non-support vectors is regarded as 0.

As $n \to \infty$, we can obtain the limiting values of ϵ and s for a given distribution of noise u_i. Suppose that $g(\cdot)$ is the density function of the noise term u_i. Asymptotically, Eq. (14) becomes,

$$\begin{cases} \dfrac{1}{\epsilon^* + 1} = Pr(|\dfrac{U}{s^*}| > \epsilon^*), \\ 1 = \displaystyle\int_{\epsilon^*}^{+\infty} u\,(g(u) - g(-u))\,du. \end{cases} \tag{15}$$

Each paired $\theta = (\epsilon, s)$ value corresponds to a potential key to a real data set. We now propose obtaining the"best" key in the tool box. Figure 1 shows some potential keys for inferring the unknown noise. This means the ϵ-Laplacian distribution can approximate the real noise distribution by adapting the scale parameter s and the insensitive parameter ϵ.

Finally, the framework of our D-D SVR with D-D insensitive parameters can be given as follows:

Step 1. Apply the ϵ-SVR ($\epsilon = \mathit{0}$, $C = \mathit{1}$) in training sets, and obtain residuals U_i;

Step 2. Estimate the insensitive parameter ϵ and the scale parameter s by minimizing Eq. (12);

Step 3. Train our D-D SVR using the updated $\hat{\epsilon}$ and $\hat{s}$; and

Step 4. Predict the targets in the test set.

3 Simulation experiments

To illustrate how the working likelihood produces D-D parameter estimation (D-D) and a prediction, we now consider three types of residuals generated from the uniform distribution, the norm distribution, and the ϵ-Laplacian distribution, respectively.

For comparison, we will investigate other three insensitive parameter estimation methods for the ϵ-SVR. The first one is the tuning parameter setting (tuning) ($C = 1.0$ and $\epsilon = 0.1$) (Chang & Lin, 2011). The second method, Cherkassky & Ma (2004)'s empirical parameter approach (CM), is

$$\epsilon_{CM} = 3\sigma_{\text{noise}}\sqrt{\frac{\ln n}{n}}, \tag{16}$$

where the standard deviation of noise σ_{noise} is obtained from the residuals using $\epsilon = 0$. The last one is the k-cross validation (k-CV), where k is fixed at 10, and 5 alternative ϵ settings are set as $0.01, 0.05, 0.1, 0.2$ and 0.3. Both mean absolute error (MAE) and root mean square error (RMSE) are calculated for comparison.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n} |\hat{y}_i - y_i|, \tag{17}$$

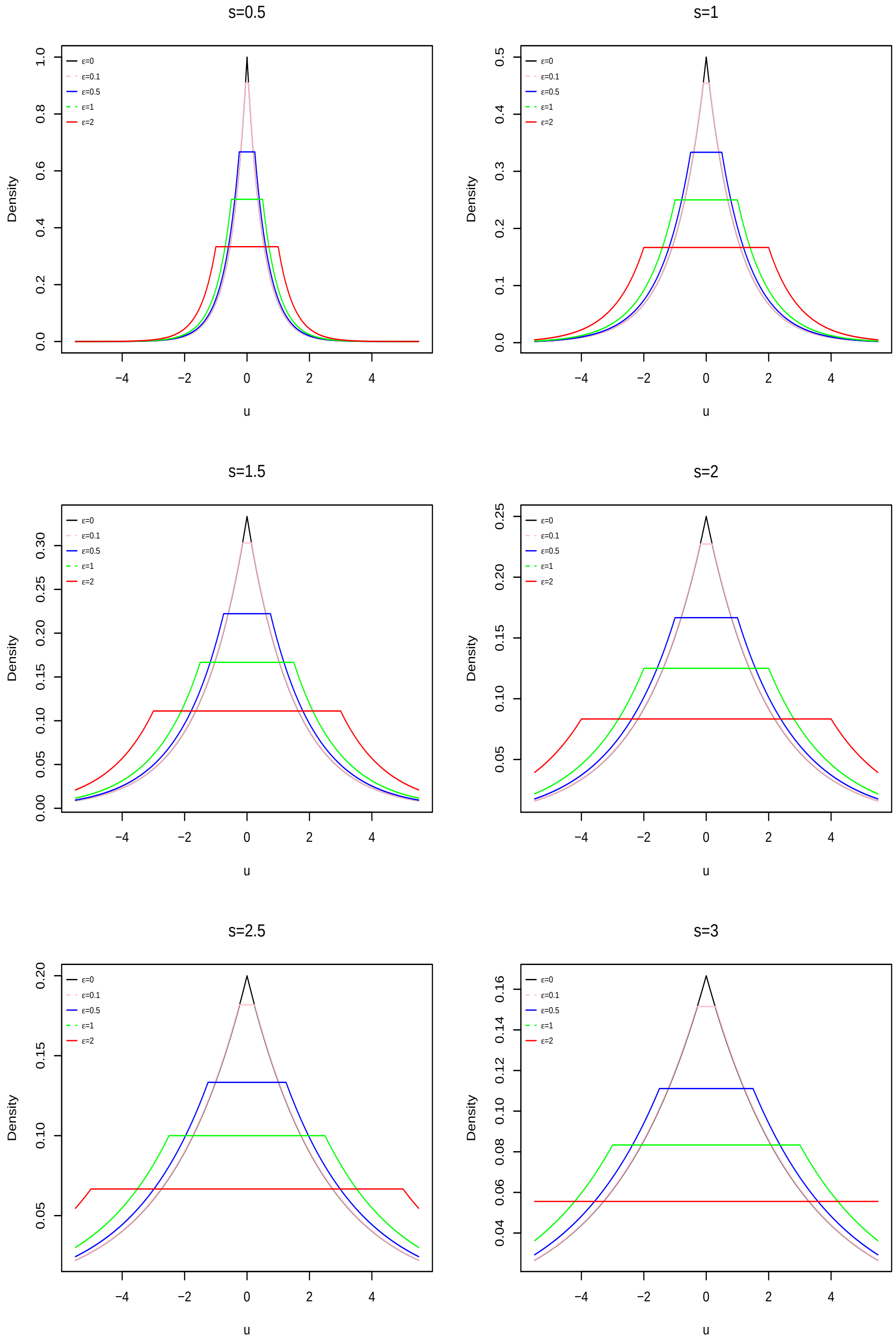

Figure 1: Working likelihood functions with different insensitive parameters at different scales.

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}, \tag{18}$$

where $\hat{y}_i$ is the i-th prediction, and y_i is the i-th observation. For each method X using the tuning method as the benchmark approach, two ratios are defined as

$$\mathrm{ratio}_{\mathrm{RMSE}} = \frac{\mathrm{RMSE}_{\mathrm{tuning}}}{\mathrm{RMSE}_X}, \tag{19}$$

$$\mathrm{ratio}_{\mathrm{MAE}} = \frac{\mathrm{MAE}_{\mathrm{tuning}}}{\mathrm{MAE}_X}. \tag{20}$$

It is obvious that the method X beats the tuning setting only if the ratio is larger than 1, and otherwise, it does not. The nonlinear simulations and linear simulations are applied to show the efficiency of our proposed D-D SVR.

3.1 Nonlinear regression

To demonstrate the performance of our D-D SVR for nonlinear system modeling, the univariate *sinc* target function from the SVR literature (Drucker et al., 1997; Chu et al., 2004; Xu & Wang, 2012; Karal, 2017) is considered as

$$y_i = a \cdot \frac{\sin(x_i)}{x_i} + su_i, \quad i = 1, 2, \ldots, n, \tag{21}$$

where x_i is generated from the uniform distribution $unif[-10, 10]$; s is the scale of the noise level; and the standard noise u_i is generated from a known distribution (ϵ-Laplacian distribution, normal distribution $N(0, \sigma^2)$, and uniform distribution $unif[-b, b]$). In addition, to make our simulations more meaningful, the scale of nonlinear system a is set as 5, 4, and 6 from insensitive-Laplacian noises, normal noises, and uniform noises, respectively. Also, we generate n simulation samples, and then the samples are divided into two groups of the same size. All experiments are repeated 100 times to calculate the average performance of the benchmark SVRs and our proposed D-D SVR. The kernel of the SVR is the default radial basic function. It should be noted that, for our comparison, the ratio is calculated based on the gap between the prediction $\hat{y}_i$ and the μ_i ($\mu_i = a\sin(x_i)/x_i$). This can show the performance of our D-D SVR at eliminating the interruption from noise and model a real system. All of the nonlinear simulation results are displayed in Table 1 (insensitive Laplacian distribution), Table 2 (normal distribution), and Table 3 (uniform distribution).

As illustrated in Table 1, compared with the CM and 10-CV, the ratios of the D-D from both RMSE and MAE are significantly greater than 1, indicating that our proposed SVR allowed for remarkable improvements in the forecasting performance for all 27 simulations. However, the insensitive parameter ϵ tends to be underestimated. The main reason for this is that, as shown in Figure 1, the scale mainly contributes to the working likelihood function when the insensitive parameter is small. Another reason is that the training sample size is not large enough to estimate the insensitive parameter accurately. As the training set size enlarges, the estimated insensitive parameter converges to the true ϵ.

Table 2 shows the second case, where the errors follow normal distributions. Our proposed method works well for approximating the best ϵ-Laplacian distribution, leading to significant improvements in the forecasting accuracy of all the simulation scenarios displayed in the

Table 1: Nonlinear case (ϵ-Laplacian distribution): Relative performance of the CM, 10-CV, and D-D methods in comparison to the tuning approach.

Noise settings			Parameters		CM		10-CV		D-D	
n	s	ϵ	$\hat{s}$	$\hat{\epsilon}$	$\text{ratio}_{\text{MAE}}$	$\text{ratio}_{\text{RMSE}}$	$\text{ratio}_{\text{MAE}}$	$\text{ratio}_{\text{RMSE}}$	$\text{ratio}_{\text{MAE}}$	$\text{ratio}_{\text{RMSE}}$
200	0.8	0.2	0.66	0.00	0.95	0.97	0.99	1.00	1.47	1.32
400	0.8	0.2	0.73	0.02	0.94	0.95	1.11	1.00	1.71	1.55
1000	0.8	0.2	0.77	0.01	0.95	0.95	1.11	1.01	1.86	1.69
200	0.8	0.5	0.70	0.00	0.96	0.97	1.07	1.01	1.44	1.36
400	0.8	0.5	0.77	0.04	0.95	0.96	1.09	1.00	1.66	1.52
1000	0.8	0.5	0.82	0.06	0.96	0.96	1.10	1.01	1.70	1.56
200	0.8	1.0	0.81	0.03	0.96	0.97	1.06	1.00	1.30	1.22
400	0.8	1.0	0.87	0.15	0.97	0.97	1.09	1.00	1.54	1.42
1000	0.8	1.0	0.87	0.53	0.99	0.99	1.10	1.01	1.71	1.56
200	1.0	0.2	0.85	0.00	0.96	0.97	1.10	1.01	1.47	1.34
400	1.0	0.2	0.93	0.01	0.94	0.95	1.10	1.00	1.66	1.51
1000	1.0	0.2	0.98	0.01	0.94	0.94	1.10	1.01	1.78	1.64
200	1.0	0.5	0.89	0.00	0.95	0.96	1.07	0.99	1.41	1.31
400	1.0	0.5	0.97	0.02	0.95	0.96	1.08	1.00	1.55	1.44
1000	1.0	0.5	1.03	0.05	0.96	0.96	1.08	1.01	1.65	1.54
200	1.0	1.0	0.99	0.04	0.98	0.98	1.05	1.00	1.24	1.18
400	1.0	1.0	1.08	0.15	0.97	0.98	1.05	1.00	1.42	1.35
1000	1.0	1.0	1.08	0.57	1.00	1.00	1.09	1.01	1.67	1.54
200	1.2	0.2	1.02	0.00	0.94	0.95	1.08	1.00	1.39	1.28
400	1.2	0.2	1.10	0.00	0.93	0.94	1.09	1.00	1.55	1.41
1000	1.2	0.2	1.17	0.01	0.93	0.93	1.11	1.01	1.73	1.58
200	1.2	0.5	1.08	0.02	0.95	0.95	1.06	1.00	1.26	1.19
400	1.2	0.5	1.16	0.01	0.95	0.95	1.07	0.99	1.45	1.34
1000	1.2	0.5	1.24	0.06	0.96	0.97	1.08	1.01	1.64	1.52
200	1.2	1.0	1.20	0.07	0.99	0.98	1.04	1.00	1.18	1.14
400	1.2	1.0	1.32	0.14	0.99	0.99	1.06	1.01	1.29	1.22
1000	1.2	1.0	1.31	0.49	1.00	1.01	1.09	1.02	1.52	1.42

Table 2: Nonlinear case (normal distribution): Relative performance of the CM, 10-CV, and D-D methods in comparison to the tuning approach.

Noise settings			Parameters		CM		10-CV		D-D	
n	s	σ	$\hat{s}$	$\hat{\epsilon}$	$\text{ratio}_{\text{MAE}}$	$\text{ratio}_{\text{RMSE}}$	$\text{ratio}_{\text{MAE}}$	$\text{ratio}_{\text{RMSE}}$	$\text{ratio}_{\text{MAE}}$	$\text{ratio}_{\text{RMSE}}$
200	0.7	0.5	0.22	0.00	0.96	0.98	1.00	1.00	1.37	1.16
400	0.7	0.5	0.24	0.00	0.97	0.97	1.01	1.01	1.67	1.46
1000	0.7	0.5	0.24	0.47	1.00	1.00	1.03	1.03	1.64	1.48
200	0.7	1.0	0.44	0.00	0.97	0.98	1.00	1.00	1.29	1.20
400	0.7	1.0	0.49	0.06	0.97	0.98	1.00	1.00	1.48	1.36
1000	0.7	1.0	0.45	0.88	0.98	0.99	1.01	1.01	1.50	1.38
200	0.7	1.5	0.66	0.03	0.98	0.98	1.00	1.00	1.17	1.12
400	0.7	1.5	0.70	0.31	0.97	0.98	1.00	1.00	1.38	1.29
1000	0.7	1.5	0.65	1.05	0.98	0.99	1.01	1.01	1.48	1.36
200	0.9	0.5	0.28	0.00	0.97	0.99	1.00	1.01	1.42	1.25
400	0.9	0.5	0.31	0.01	0.96	0.97	1.00	1.01	1.58	1.42
1000	0.9	0.5	0.30	0.68	0.99	0.99	1.02	1.02	1.59	1.45
200	0.9	1.0	0.57	0.00	0.97	0.97	1.00	1.00	1.17	1.10
400	0.9	1.0	0.62	0.17	0.97	0.98	1.00	1.01	1.33	1.23
1000	0.9	1.0	0.57	0.90	0.98	0.99	1.00	1.00	1.48	1.36
200	0.9	1.5	0.86	0.10	0.98	0.98	1.00	1.00	1.09	1.04
400	0.9	1.5	0.89	0.40	0.98	0.99	1.00	1.01	1.25	1.19
1000	0.9	1.5	0.81	1.15	0.99	0.99	1.01	1.01	1.37	1.28
200	1.1	0.5	0.35	0.00	0.96	0.97	1.00	1.00	1.36	1.24
400	1.1	0.5	0.38	0.04	0.97	0.97	1.00	1.00	1.56	1.41
1000	1.1	0.5	0.35	0.85	0.98	0.98	1.01	1.01	1.59	1.45
200	1.1	1.0	0.69	0.03	0.98	0.98	1.01	1.01	1.14	1.08
400	1.1	1.0	0.75	0.29	0.98	0.98	1.00	1.00	1.33	1.24
1000	1.1	1.0	0.68	1.03	0.99	0.99	1.00	1.00	1.42	1.31
200	1.1	1.5	1.03	0.17	1.00	1.00	1.00	1.00	1.01	0.98
400	1.1	1.5	1.06	0.56	0.98	0.98	1.00	1.00	1.17	1.11
1000	1.1	1.5	1.01	1.07	0.99	0.99	1.00	1.00	1.30	1.22

Table. In particular, when the noise level is low (both s and σ are small), the superiority of the D-D approach is more prominent. For the simulation with noise settings (n 1000, s 0.7, and σ 0.5), the D-D's prediction achieves an amazing improvement (MAE, 64%, and RMSE, 48%), while both the CM and 10-CV methods each obtained only a slight increase.

Table 3: Nonlinear case (uniform distribution): Relative performance of the CM, 10-CV, and D-D methods in comparison to the tuning approach.

Noise settings			Parameters		CM		10-CV		D-D	
n	s	b	$\hat{s}$	$\hat{\epsilon}$	ratio$_{\text{MAE}}$	ratio$_{\text{RMSE}}$	ratio$_{\text{MAE}}$	ratio$_{\text{RMSE}}$	ratio$_{\text{MAE}}$	ratio$_{\text{RMSE}}$
200	3.0	0.8	0.86	0.35	0.99	0.99	1.01	1.01	1.26	1.22
400	3.0	0.8	0.69	2.15	1.00	1.01	1.01	1.01	1.74	1.62
1000	3.0	0.8	0.41	4.96	1.00	1.00	1.02	1.02	2.16	1.94
200	3.0	1.0	1.08	0.37	1.02	1.02	1.01	1.01	1.17	1.15
400	3.0	1.0	0.83	2.28	1.01	1.02	1.01	1.01	1.66	1.59
1000	3.0	1.0	0.51	4.91	1.01	1.01	1.02	1.02	2.23	2.06
200	3.0	1.2	1.23	0.80	1.05	1.06	1.02	1.02	1.27	1.26
400	3.0	1.2	0.99	2.36	1.01	1.02	1.01	1.02	1.67	1.59
1000	3.0	1.2	0.62	4.85	1.01	1.01	1.02	1.02	2.10	1.98
200	4.0	0.8	1.10	0.72	1.02	1.02	1.01	1.01	1.24	1.21
400	4.0	0.8	0.87	2.47	1.02	1.03	1.02	1.02	1.67	1.60
1000	4.0	0.8	0.55	4.87	1.01	1.01	1.02	1.02	2.17	2.02
200	4.0	1.0	1.34	0.87	1.07	1.07	1.02	1.02	1.22	1.20
400	4.0	1.0	1.11	2.25	1.03	1.03	1.01	1.02	1.62	1.56
1000	4.0	1.0	0.69	4.85	1.01	1.01	1.01	1.02	2.07	1.97
200	4.0	1.2	1.66	0.76	1.08	1.09	1.02	1.01	1.14	1.13
400	4.0	1.2	1.25	2.67	1.04	1.04	1.02	1.02	1.54	1.51
1000	4.0	1.2	0.84	4.78	1.01	1.02	1.02	1.02	2.01	1.93
200	5.0	0.8	1.38	0.63	1.05	1.05	1.02	1.01	1.17	1.14
400	5.0	0.8	1.05	2.52	1.03	1.04	1.02	1.02	1.59	1.53
1000	5.0	0.8	0.66	5.11	1.01	1.01	1.02	1.02	2.07	1.97
200	5.0	1.0	1.68	0.96	1.07	1.08	1.02	1.03	1.15	1.15
400	5.0	1.0	1.33	2.56	1.03	1.04	1.01	1.01	1.50	1.47
1000	5.0	1.0	0.85	4.97	1.02	1.02	1.02	1.02	2.12	2.03
200	5.0	1.2	1.94	1.11	1.11	1.10	1.03	1.03	1.16	1.14
400	5.0	1.2	1.47	2.86	1.04	1.04	1.01	1.01	1.48	1.45
1000	5.0	1.2	0.96	5.41	1.03	1.03	1.03	1.03	2.00	1.93

The third nonlinear case also shows that our D-D method is an effective approach to data modeling with noises from the uniform distribution, and the simulation results are given in Table 3. Obviously, two ratios from the proposed D-D method are notably greater than 1. For instance, compared with the CM and 10-CV methods, both ratios of the simulation from the D-D method with noise setting n 1000, s 5.0 and b 1.2, are nearly 200% (MAE) and 193% (RMSE), respectively, so our D-D method obtained a nearly twofold improvement.

From the above three types of nonlinear simulations, it can be concluded that our proposed D-D method for ϵ-SVR noticeably improves the forecasting performance in nonlinear applications.

3.2 Linear regression

Now we consider the most popular linear model generated by the following:

$$y_i = \beta_0 + \beta_1 \cdot x_i + s u_i, \quad i = 1, 2, \ldots, n, \tag{22}$$

where $\beta_0 = 1$ and x_i is generated from the normal distribution $N(0, 1)$. Considering different noise levels for all simulations, we set β_1 as 2, 2, and 1 for noises generated from the

ϵ-Laplacian distribution, normal distribution, and uniform distribution, respectively. In addition, the kernel of the ϵ-SVR is the linear function. All simulations are implemented 100 times to record the average performance. The linear simulation results for the ϵ-Laplacian distribution, normal distribution $N(0, \sigma^2)$, and uniform distribution $unif[-b, b]$ are listed in Table 4, Table 5 and Table 6, respectively.

Table 4: Linear case (ϵ-Laplacian distribution): Relative performance of the CM, 10-CV, and D-D methods in comparison to the tuning approach.

Noise settings				Parameters		CM		10-CV		D-D	
n	s	ϵ	R^2	$\hat{s}$	$\hat{\epsilon}$	$\text{ratio}_{\text{MAE}}$	$\text{ratio}_{\text{RMSE}}$	$\text{ratio}_{\text{MAE}}$	$\text{ratio}_{\text{RMSE}}$	$\text{ratio}_{\text{MAE}}$	$\text{ratio}_{\text{RMSE}}$
100	0.5	0.8	0.87	0.51	0.71	0.80	0.79	0.99	0.98	1.13	1.13
200	0.5	0.8	0.87	0.51	0.71	0.81	0.83	0.97	0.97	1.21	1.21
300	0.5	0.8	0.87	0.49	0.86	1.39	1.35	1.14	1.12	1.46	1.41
100	0.5	1.0	0.85	0.52	0.74	0.96	0.96	1.07	1.06	1.07	1.09
200	0.5	1.0	0.85	0.51	0.93	1.18	1.16	1.16	1.15	1.26	1.25
300	0.5	1.0	0.86	0.50	0.95	1.15	1.14	1.08	1.08	1.23	1.22
100	0.5	1.2	0.86	0.52	1.12	0.99	1.03	1.08	1.11	1.30	1.32
200	0.5	1.2	0.86	0.50	1.21	1.34	1.35	1.16	1.15	1.38	1.38
300	0.5	1.2	0.84	0.52	1.07	1.20	1.19	1.08	1.07	1.33	1.31
100	1.0	0.8	0.65	0.97	0.76	0.98	0.96	1.08	1.07	1.33	1.28
200	1.0	0.8	0.62	1.01	0.73	0.96	0.92	1.04	0.99	1.52	1.53
300	1.0	0.8	0.62	1.00	0.76	1.19	1.20	1.10	1.11	1.20	1.21
100	1.0	1.0	0.61	1.00	0.98	0.76	0.75	1.02	1.03	1.20	1.18
200	1.0	1.0	0.61	1.02	0.95	1.18	1.16	1.11	1.10	1.31	1.28
300	1.0	1.0	0.60	1.01	0.95	1.30	1.27	1.15	1.15	1.52	1.50
100	1.0	1.2	0.58	0.99	1.27	1.32	1.27	1.12	1.11	1.39	1.37
200	1.0	1.2	0.57	1.00	1.17	1.46	1.42	1.22	1.19	1.55	1.50
300	1.0	1.2	0.58	1.02	1.14	1.37	1.36	1.12	1.10	1.62	1.60
100	1.5	0.8	0.43	1.48	0.78	0.89	0.86	1.09	1.07	1.28	1.27
200	1.5	0.8	0.42	1.47	0.77	1.04	1.05	1.03	1.03	1.27	1.28
300	1.5	0.8	0.42	1.49	0.79	1.19	1.20	1.21	1.22	1.25	1.27
100	1.5	1.0	0.42	1.44	1.09	1.07	1.06	1.09	1.09	1.24	1.23
200	1.5	1.0	0.41	1.49	0.99	1.23	1.23	1.14	1.13	1.37	1.35
300	1.5	1.0	0.40	1.48	1.05	1.25	1.23	1.14	1.13	1.30	1.26
100	1.5	1.2	0.38	1.53	1.15	1.37	1.38	1.30	1.30	1.73	1.74
200	1.5	1.2	0.38	1.60	1.04	1.17	1.20	1.07	1.05	1.44	1.40
300	1.5	1.2	0.38	1.50	1.19	1.32	1.29	1.11	1.11	1.68	1.69

First, in the linear simulation for residuals generated from the ϵ-Laplacian distribution, the estimated insensitive parameter $\hat{\epsilon}$ and the estimated scale parameter $\hat{s}$ all approximate to the real settings with our D-D method in different noise levels, as shown in Table 4. For comparison of the accuracy for the forecasting performance, in the linear regression with $n = 300$ and $R^2 = 0.38$, our proposed D-D SVR performed better than the CM and the 10-CV, with a more than 68% improvement with MAE and a 69% improvement with RMSE. Overall, our D-D method can precisely improve forecasting performance by auto-adapting the insensitive parameter.

The second linear simulation, shown Table 5, is the regression with noises from the normal distribution $N(0, \sigma^2)$. The simulation results show that with R^2 from 0.40 to 0.86, all the $\text{ratio}_{\text{MAE}}$ and $\text{ratio}_{\text{RMSE}}$ for D-D are all significantly greater than 1. In other words, our proposed method can auto-recognize a limited scale and obtain a limiting insensitive parameter to approach real noises; as a result, the forecasting performance is superior. It is interesting that corresponding to the type of noise, the scale is also auto-adapted to match the most approximate ϵ in the insensitive Laplacian distribution. Therefore, our method can make ϵ-SVR more efficient in the linear model with Gaussian noises.

The final simulation, shown in Table 6, illustrates that our D-D method can obtain

Table 5: Linear case (normal distribution): Relative performance of the CM, 10-CV, and D-D methods in comparison to the tuning approach.

Noise settings				Parameters		CM		10-CV		D-D	
n	s	σ	R^2	$\hat{s}$	$\hat{\epsilon}$	$\text{ratio}_{\text{MAE}}$	$\text{ratio}_{\text{RMSE}}$	$\text{ratio}_{\text{MAE}}$	$\text{ratio}_{\text{RMSE}}$	$\text{ratio}_{\text{MAE}}$	$\text{ratio}_{\text{RMSE}}$
100	1.0	0.8	0.86	0.48	1.28	1.13	1.13	1.08	1.09	1.27	1.27
200	1.0	0.8	0.88	0.46	1.39	1.09	1.08	1.05	1.04	1.37	1.34
300	1.0	0.8	0.86	0.47	1.41	1.01	1.04	1.03	1.06	1.26	1.27
100	1.0	1.0	0.81	0.59	1.20	0.95	0.95	1.06	1.08	1.17	1.18
200	1.0	1.0	0.79	0.58	1.43	1.17	1.20	1.11	1.13	1.35	1.36
300	1.0	1.0	0.80	0.58	1.39	1.10	1.10	0.99	1.00	1.34	1.35
100	1.0	1.2	0.74	0.72	1.25	0.85	0.81	0.97	0.97	1.35	1.32
200	1.0	1.2	0.74	0.73	1.26	1.03	1.04	1.00	1.00	1.12	1.12
300	1.0	1.2	0.75	0.68	1.51	1.09	1.09	1.01	1.00	1.21	1.21
100	1.5	0.8	0.75	0.71	1.35	1.04	1.03	1.25	1.24	1.22	1.20
200	1.5	0.8	0.73	0.71	1.32	1.03	1.03	1.00	1.00	1.22	1.19
300	1.5	0.8	0.73	0.70	1.38	1.09	1.09	1.03	1.02	1.27	1.28
100	1.5	1.0	0.67	0.85	1.49	0.79	0.79	1.10	1.10	1.65	1.68
200	1.5	1.0	0.64	0.85	1.47	1.22	1.19	1.09	1.08	1.34	1.33
300	1.5	1.0	0.64	0.86	1.48	1.17	1.19	1.17	1.19	1.37	1.39
100	1.5	1.2	0.56	1.03	1.57	1.09	1.10	1.05	1.04	1.20	1.21
200	1.5	1.2	0.55	1.03	1.48	1.04	1.05	0.97	0.96	1.26	1.25
300	1.5	1.2	0.56	1.03	1.41	1.16	1.16	1.09	1.08	1.24	1.25
100	2.0	0.8	0.62	0.90	1.48	1.23	1.20	0.98	0.97	1.25	1.22
200	2.0	0.8	0.61	0.88	1.65	1.15	1.17	1.05	1.04	1.46	1.46
300	2.0	0.8	0.61	0.91	1.54	1.14	1.12	1.14	1.11	1.43	1.39
100	2.0	1.0	0.52	1.13	1.50	1.07	1.09	1.03	1.04	1.21	1.23
200	2.0	1.0	0.51	1.15	1.42	1.12	1.12	1.03	1.03	1.35	1.32
300	2.0	1.0	0.51	1.11	1.54	1.14	1.12	1.08	1.07	1.45	1.41
100	2.0	1.2	0.43	1.37	1.43	1.51	1.62	1.14	1.15	1.82	1.94
200	2.0	1.2	0.42	1.35	1.48	1.20	1.19	1.05	1.06	1.22	1.22
300	2.0	1.2	0.40	1.36	1.52	0.98	1.00	1.02	1.02	1.01	1.03

Table 6: Linear case (uniform distribution): Relative performance of the CM, 10-CV, and D-D methods in comparison to the tuning approach.

Noise settings				Parameters		CM		10-CV		D-D	
n	s	b	R^2	$\hat{s}$	$\hat{\epsilon}$	$\text{ratio}_{\text{MAE}}$	$\text{ratio}_{\text{RMSE}}$	$\text{ratio}_{\text{MAE}}$	$\text{ratio}_{\text{RMSE}}$	$\text{ratio}_{\text{MAE}}$	$\text{ratio}_{\text{RMSE}}$
100	1.0	0.8	0.81	0.19	4.06	1.25	1.23	1.11	1.09	2.72	2.69
200	1.0	0.8	0.82	0.12	6.29	1.21	1.20	1.14	1.14	3.25	3.23
300	1.0	0.8	0.83	0.09	8.25	1.00	1.00	1.13	1.12	3.14	3.09
100	1.0	1.0	0.77	0.22	4.19	1.42	1.43	1.08	1.09	2.42	2.46
200	1.0	1.0	0.76	0.14	7.15	1.20	1.19	1.13	1.13	3.00	2.96
300	1.0	1.0	0.76	0.12	8.15	1.16	1.17	1.16	1.17	3.89	3.89
100	1.0	1.2	0.68	0.25	4.54	1.47	1.47	1.15	1.15	2.56	2.57
200	1.0	1.2	0.67	0.18	6.61	1.14	1.16	1.09	1.09	3.63	3.65
300	1.0	1.2	0.68	0.14	8.37	1.18	1.20	1.19	1.21	3.61	3.58
100	1.5	0.8	0.66	0.28	4.19	1.13	1.13	1.04	1.03	2.34	2.29
200	1.5	0.8	0.68	0.16	7.52	1.07	1.07	1.01	1.02	3.02	3.05
300	1.5	0.8	0.68	0.15	8.06	1.20	1.20	1.16	1.15	3.36	3.32
100	1.5	1.0	0.58	0.33	4.47	1.64	1.63	1.09	1.09	2.45	2.44
200	1.5	1.0	0.57	0.21	7.30	1.21	1.23	1.10	1.10	3.91	3.92
300	1.5	1.0	0.57	0.17	8.69	1.09	1.08	1.10	1.10	4.08	4.16
100	1.5	1.2	0.47	0.35	5.50	1.23	1.20	1.06	1.05	2.08	2.07
200	1.5	1.2	0.50	0.24	7.55	1.17	1.19	1.05	1.05	3.73	3.90
300	1.5	1.2	0.48	0.21	8.53	1.17	1.19	1.13	1.13	4.13	4.14
100	2.0	0.8	0.53	0.30	5.29	1.42	1.40	1.16	1.12	3.06	3.06
200	2.0	0.8	0.54	0.21	7.85	1.18	1.18	1.15	1.16	3.17	3.20
300	2.0	0.8	0.53	0.19	8.45	1.05	1.02	1.06	1.04	4.27	4.13
100	2.0	1.0	0.43	0.34	5.88	1.16	1.20	0.96	0.98	2.46	2.48
200	2.0	1.0	0.44	0.27	7.43	1.05	1.05	1.00	0.99	2.85	2.83
300	2.0	1.0	0.43	0.22	9.25	1.14	1.14	1.17	1.17	4.17	4.18
100	2.0	1.2	0.35	0.46	5.03	1.38	1.40	1.08	1.08	3.22	3.39
200	2.0	1.2	0.35	0.33	7.58	1.12	1.10	1.04	1.02	3.20	3.14
300	2.0	1.2	0.34	0.26	9.20	1.09	1.07	1.10	1.10	4.32	4.20

surprisingly good improvements. This is because the ratios from our D-D method are quite large, indicating that our proposed method can model the linear model with perfect accuracy. The most interesting finding in the parameter estimation analysis is that with an increasing number of samples, our D-D method approaches approximating the ϵ-Laplacian loss function by increasing ϵ and decreasing s; two parameter estimations will converge to limiting values. To sum up, for the noise from uniform distribution, our method is still a powerful tool for improving the linear regression forecasting.

Furthermore, for the mechanism exploration of our D-D method, compared with the CM in linear simulations, which is motivated by the noise following the normal distribution, our D-D's forecasting performance is close, but still is better when addressing the noise from the normal distribution shown in Table 5, while in Table 4 and Table 6, our D-D method's performance can significantly improve the forecasting accuracy. This illustrates that our D-D method can auto-adapt the parameters to approximate any unknown noise distribution and improve the SVR's performance, while the CM method focuses on the normal distribution. Moreover, the computational cost of the 10-CV method with five alternative parameter settings is over 10 times more than our D-D method. In addition, because of the parameter setting for the cross validation, the 10-CV method cannot guarantee its superior performance with high computational costs. Therefore, we can conclude that our D-D method can auto-adapt the ϵ-Laplacian loss function to guarantee the steadiness of a linear model with high levels of accuracy. Furthermore, because it is determined by the type of noise, the scale and the insensitive parameter will converge to true values (the noise is generated from the ϵ-Laplacian distribution) or limiting values (the noise is from any other distribution).

4 Case studies

In the section, our D-D ϵ-SVR is evaluated with five case studies: energy efficiency (768 samples, eight attributes, and two responses (Tsanas & Xifara, 2012), yacht hydrodynamics (308 samples, six attributes, and one response) (Ortigosa et al., 2007), airfoil self-noise (1503 samples, five attributes, and one response) (Lau et al., 2006), concrete compressive strength (1030 samples, eight attributes, and one response) (Yeh, 2006) from the UCI Machine Learning Repository (Dua & Graff, 2017), and Boston housing prices (506 samples, 14 attributes, and one response) from the StatLib collection (Fan, 2019).

Each benchmark data set was randomly divided into two groups: the training set (70% of each data set) and the test set (the remaining data from each set). Then, each experiment was repeated 100 times to obtain the average performance of our proposed SVR. Because the scale of each attribute is different, the standard normalization was applied for attribute pre-processing before the training. The general radial basic function is selected as the kernel. In addition, the 10-CV was applied in the insensitive parameter selection with the same alternative parameter settings as the former simulations.

The ϵ and σ for the five benchmark data sets were estimated using our proposed method, and the work likelihood functions are displayed in Figure 2. It is obvious that the specific ϵ-Laplacian loss function was data-driven by the real data sets. Different from the original ϵ-SVR, our proposed "scale" ϵ-SVR can auto-recognize the scale of noise in real data sets and self-adapt the insensitive parameter accordingly. For example, as illustrated in Figure 2, the working likelihood functions for energy efficiency (heating load) and concrete compressive

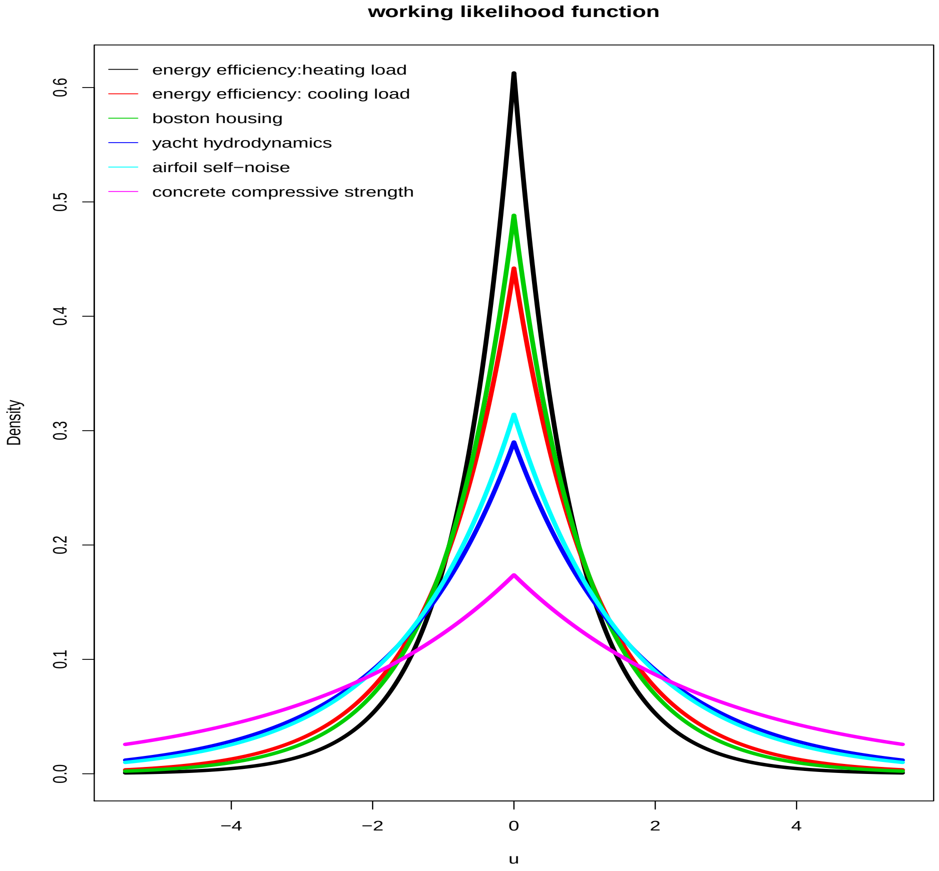

Figure 2: Six working likelihood D-D functions for five case studies.

strength were significantly different; the scale parameter estimation for the former data set was 0.82, while the estimation for the latter was 2.88, as shown in Table 7.

Table 7: Results for four case studies: Relative performance of the tuning, CM, 10-CV, and D-D methods.

	Parameters		MAE				RMSE			
	$\hat{s}$	$\hat{\epsilon}$	tuning	CM	10-CV	D-D	tuning	CM	10-CV	D-D
Energy efficiency										
Heating Load	0.82	0.00	1.49	1.44	1.49	**1.08**	2.27	2.30	2.27	**1.87**
Cooling Load	1.13	0.00	1.84	1.82	1.81	**1.48**	2.68	2.69	2.69	**2.32**
Boston Housing										
	1.02	0.00	2.38	2.39	2.39	**2.22**	3.97	3.96	3.97	**3.53**
Yacht Hydrodynamics										
	1.72	0.00	3.83	4.09	4.17	**2.67**	6.81	6.71	6.70	**4.95**
Airfoil Self-Noise										
	1.59	0.00	2.41	2.42	2.42	**1.96**	3.31	3.31	3.31	**2.79**
Concrete Compressive Strength										
	2.88	0.00	5.00	5.00	4.97	**4.29**	6.84	6.84	6.85	**6.13**

The prediction performance for all five cases are listed in Table 7. Obviously, our proposed method can dramatically improve the accuracy of predictions based on the ratios. The most obvious cases are the MAE (tuning 3.83 vs. CM 4.09 vs. 10-CV 4.17 vs. D-D 2.67) and RMSE (tuning 6.81 vs. CM 6.71 vs. 10-CV 6.70 vs. D-D 4.95) for the yacht hydrodynamics. Compared with the tuning, 10-CV, and CM methods, the MAE and RMSE in the rest of the data sets (energy efficiency, Boston housing, airfoil self-noise, and concrete compressive strength) achieved around 10% improvements.

To summarize, our proposed D-D method can auto-adapt the insensitive parameter in the ϵ-Laplacian distribution approach to the real noise distribution; this means our working likelihood method can push the ϵ-Laplacian density function to seek the approximate likelihood function. As a result, our D-D SVR has an excellent performance in real applications.

5 Conclusion

The SVR with ϵ-Laplacian loss distribution is a mainstream algorithm for regression modeling, where the insensitive parameter ϵ determines the support vector. However, to date, after inputs and target scaling, three types of strategies for parameter selection are used: the k-cross validation, which requires huge computational costs, the tuning parameter, which cannot make the SVR work more efficiently, and the empirical statistical estimation, the CM method that is based on normal distribution with some empirical settings. Obviously, the mentioned parameter settings are not the most appropriate hyper-parameters for SVR in most conditions, so, in this paper, we propose optimization of the insensitive parameter based on the working likelihood function developed by Wang et al. (2007), which is a D-D method, to estimate appropriate hyper-parameters for finding the most appropriate ϵ-Laplacian distribution to the real noise distribution in order to guarantee generalization in test sets. In addition, the D-D vector regression is standardized by the scale of the noise in a more meaningful field. In nonlinear and linear simulations conducted with different types of noises (ϵ-Laplacian distribution, normal distribution, and uniform distribution), our proposed method demonstrated that it can automatically estimate the scale and the insensitive parameter. As a result, our D-D SVR showed significantly improved forecasting accuracy in the test sets. Moreover, our D-D algorithm can estimate the approximate likelihood function in five real benchmark applications, and furthermore, the proposed method had dramatically improved performance in unknown sets. Therefore, our proposed D-D SVR is a more intelligent and powerful technique for the regression problem. Furthermore, in machine learning modeling, our D-D method using the framework of working likelihood is a viable general strategy for parameter estimations in different loss functions.

Declaration of interest

The authors declare no conflict interest.

Acknowledgements

Computational (and/or data visualisation) resources and services used in this work were provided by the HPC and Research Support Group, Queensland University of Technology, Brisbane, Australia. This work was supported in part by the ARC Centre of Excellence for Mathematical and Statistical Frontiers. This work was supported by the Australian Research Council project DP160104292.

References

Bartlett, P. L., Boucheron, S., & Lugosi, G. (2002). Model selection and error estimation. *Machine Learning*, *48*(1-3), 85–113.

Chang, C.-C., & Lin, C.-J. (2002). Training v-support vector regression: theory and algorithms. *Neural computation*, *14*(8), 1959–1977.

Chang, C.-C., & Lin, C.-J. (2011). Libsvm: A library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, *2*(3), 27.

Cherkassky, V., & Ma, Y. (2004). Practical selection of svm parameters and noise estimation for svm regression. *Neural networks*, *17*(1), 113–126.

Chu, W., Keerthi, S. S., & Ong, C. J. (2004). Bayesian support vector regression using a unified loss function. *IEEE transactions on neural networks*, *15*(1), 29–44.

Drucker, H., Burges, C. J., Kaufman, L., Smola, A. J., & Vapnik, V. (1997). Support vector regression machines. In *Advances in neural information processing systems* (pp. 155–161).

Dua, D., & Graff, C. (2017, August). *UCI machine learning repository.* Retrieved from http://archive.ics.uci.edu/ml

Fan, R.-E. (2019, August). *Libsvm data: Regression.* Retrieved from https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/regression.html

Hastie, T., Tibshirani, R., Friedman, J., & Franklin, J. (2005). The elements of statistical learning: data mining, inference and prediction. *The Mathematical Intelligencer*, *27*(2), 83–85.

Ito, K., & Nakano, R. (2003). Optimizing support vector regression hyperparameters based on cross-validation. In *Proceedings of the international joint conference on neural networks, 2003.* (Vol. 3, pp. 2077–2082).

Jeng, J.-T., Chuang, C.-C., & Su, S.-F. (2003). Support vector interval regression networks for interval regression analysis. *Fuzzy Sets and Systems*, *138*(2), 283–300.

Karal, O. (2017). Maximum likelihood optimal and robust support vector regression with lncosh loss function. *Neural networks*, *94*, 1–12.

Lau, K., López, R., Oñate, E., Ortega, E., Flores, R., Mier-Torrecilla, M., ... González, E. (2006). A neural networks approach for aerofoil noise prediction. *Master thesis*.

Li, W., Kong, D., & Wu, J. (2017). A new hybrid model fpa-svm considering cointegration for particular matter concentration forecasting: a case study of kunming and yuxi, china. *Computational intelligence and neuroscience*, *2017*.

Mohandes, M. A., Halawani, T. O., Rehman, S., & Hussain, A. A. (2004). Support vector machines for wind speed prediction. *Renewable Energy*, *29*(6), 939–947.

Ortigosa, I., Lopez, R., & Garcia, J. (2007). A neural networks approach to residuary resistance of sailing yachts prediction. In *Proceedings of the international conference on marine engineering marine* (Vol. 2007, p. 250).

Schölkopf, B., Bartlett, P., Smola, A., & Williamson, R. (1998). Support vector regression with automatic accuracy control. In *International conference on artificial neural networks* (pp. 111–116).

Schölkopf, B., Bartlett, P. L., Smola, A. J., & Williamson, R. C. (1999). Shrinking the tube: a new support vector regression algorithm. In *Advances in neural information processing systems* (pp. 330–336).

Schölkopf, B., Smola, A. J., Williamson, R. C., & Bartlett, P. L. (2000). New support vector algorithms. *Neural computation*, *12*(5), 1207–1245.

Smola, A. J., & Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and computing*, *14*(3), 199–222.

Trafalis, T. B., & Ince, H. (2000). Support vector machine for regression and applications to financial forecasting. In *Proceedings of the ieee-inns-enns international joint conference on neural networks. ijcnn 2000. neural computing: New challenges and perspectives for the new millennium* (Vol. 6, pp. 348–353).

Tsanas, A., & Xifara, A. (2012). Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools. *Energy and Buildings*, *49*, 560–567.

Vapnik, V. (2013). *The nature of statistical learning theory.* Springer science & business media.

Vapnik, V., Golowich, S. E., & Smola, A. J. (1997). Support vector method for function approximation, regression estimation and signal processing. In *Advances in neural information processing systems* (pp. 281–287).

Vrablecová, P., Ezzeddine, A. B., Rozinajová, V., Šárik, S., & Sangaiah, A. K. (2018). Smart grid load forecasting using online support vector regression. *Computers & Electrical Engineering*, *65*, 102–117.

Wang, Y.-G., Lin, X., Zhu, M., & Bai, Z. (2007). Robust estimation using the huber function with a data-dependent tuning constant. *Journal of Computational and Graphical Statistics*, *16*(2), 468–481.

Xu, Y., & Wang, L. (2012). A weighted twin support vector regression. *Knowledge-Based Systems*, *33*, 92–101.

Yeh, I.-C. (2006). Analysis of strength of concrete using design of experiments and neural networks. *Journal of Materials in Civil Engineering*, *18*(4), 597–604.