

Article

Kernel-Free Quadratic Surface Support Vector Regression with Non-Negative Constraints

Dong Wei ^{1,2}, Zhixia Yang ^{1,2,*} , Junyou Ye ^{1,2} and Xue Yang ^{1,2}

¹ College of Mathematics and Systems Science, Xinjiang University, Urumuqi 830046, China; weidongmath@163.com (D.W.); yejymath@163.com (J.Y.); yx11092021@163.com (X.Y.)

² Institute of Mathematics and Physics, Xinjiang University, Urumuqi 830046, China

* Correspondence: yangzhx@xju.edu.cn

Abstract: In this paper, a kernel-free quadratic surface support vector regression with non-negative constraints (NQSSVR) is proposed for the regression problem. The task of the NQSSVR is to find a quadratic function as a regression function. By utilizing the quadratic surface kernel-free technique, the model avoids the difficulty of choosing the kernel function and corresponding parameters, and has interpretability to a certain extent. In fact, data may have a priori information that the value of the response variable will increase as the explanatory variable grows in a non-negative interval. Moreover, in order to ensure that the regression function is monotonically increasing on the non-negative interval, the non-negative constraints with respect to the regression coefficients are introduced to construct the optimization problem of NQSSVR. And the regression function obtained by NQSSVR matches this a priori information, which has been proven in the theoretical analysis. In addition, the existence and uniqueness of the solution to the primal problem and dual problem of NQSSVR, and the relationship between them are addressed. Experimental results on two artificial datasets and seven benchmark datasets validate the feasibility and effectiveness of our approach. Finally, the effectiveness of our method is verified by real examples in air quality.

Keywords: regression problem; quadratic surface; kernel-free; non-negative constraints; air quality composite index dataset



Citation: Wei, D.; Yang, Z.; Ye, J.; Yang, X. Kernel-Free Quadratic Surface Support Vector Regression with Non-Negative Constraints. *Entropy* **2023**, *25*, 1030. <https://doi.org/10.3390/e25071030>

Academic Editors: Marcelo Bourguignon and Diego I. Gallardo

Received: 8 June 2023

Revised: 4 July 2023

Accepted: 5 July 2023

Published: 7 July 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

For regression problems, sometimes there is a priori information, such as the response variable increasing as the explanatory variable increases. It is more natural to expect that the the air quality will decrease when the pollution gas concentration increases. However, the model sometimes obtains regression coefficients that do not match this a priori information, which can reduce the credibility and prediction accuracy of the model. Therefore, to solve this problem, we restrict the range of values of the regression coefficients to ensure the soundness of the model.

At present, several types of constraints have been utilized, including non-negative constraints [1–4], monotonicity constraints [5–8], smoothing constraints [9–11], etc. Powell et al. [12] proposed a Bayesian hierarchical model for estimating constraints conditional random fields to analyze the relationship between air pollution and health. Moreover, non-negative constraints have been applied to various problems. The non-negative least squares problem (NNLS) was introduced by Lawson [13]. Chen et al. [14] presented non-negative distributed regression as an effective method specifically designed for analyzing data in wireless sensor networks. Shekkizhar et al. [15,16] proposed non-negative kernel regression to handle graph construction from data and dictionary learning. Additionally, Chapel et al. [17] proposed non-negative penalized linear regression to address the challenge of unbalanced optimal transport.

Due to its excellent generalization ability, support vector regression (SVR) [18] has been widely used in various fields, such as the financial industry [19,20] and construction

industry [21,22]. However, the selection of appropriate kernel functions and their corresponding parameters can be time-consuming during experiments, prompting researchers to explore kernel-free regression models. Su proposed the non-negative constraints SVR (NNSVR) for analyzing air quality data, which is only suitable for regression problems since it cannot use kernel functions. Based on the idea of quadratic kernel-free support vector machine (QSSVC) [23], Gao et al. [24] proposed a kernel-free fuzzy reduced quadratic surface ν -support vector machine for Alzheimer's disease classification. Zhou et al. [25] proposed a kernel-free QSSVC. For the regression problem, Ye et al. [26,27] proposed two kernel-free nonlinear regression models, quadratic surface kernel-free least squares SVR (QLSSVR) and ϵ -kernel-free soft QSSVR (ϵ -SQSSVR), respectively. Zhai et al. [28] proposed a linear twin quadratic surface support vector regression. Zheng et al. [29] developed a hybrid QSSVR and applied it to stock indices and price forecasting. These advancements have attracted significant attention from the research community seeking more efficient approaches in regression analysis without relying on kernel functions.

In this paper, a kernel-free quadratic surface support vector regression with non-negative constraints (NQSSVR) is proposed, which incorporates the idea of a kernel-free technique and non-negative constraints. The main contributions are summarized as follows.

1. NQSSVR is proposed by utilizing the kernel-free technique, which avoids the complexity of choosing the kernel functions and their parameters, and has interpretability to some extent. In fact, the task of NQSSVR is to find a quadratic regression function to fit the data, so it can achieve better generalization ability than other linear regression methods.
2. The non-negative constraints with respect to the regression coefficients are added to construct the optimization problem of NQSSVR, which can obtain a monotonically increasing regression function with explanatory variables on a non-negative interval. In some cases, the value of the response variable grows as the explanatory variable grows. For example, when exploring the air quality examples, the air quality index will increase as the concentration of gases in the air increases.
3. Both the primal and dual problems can be solved, since our method does not involve kernel functions. In the theoretical analysis, the existence and uniqueness of solutions to the primal and dual problems, as well as their interconnections, are analyzed. In addition, the properties of regression function on the domain of definition are given.
4. Numerical experiments on artificial datasets demonstrate the visualization results of the regression function obtained by our NQSSVR. The results on benchmark datasets show that the comprehensive performance of the method is relatively better than that of linear-SVR and NNSVR. In addition, more importantly, by exploring the practical application of air quality, it can be shown that our method is more applicable than QLSSVR and ϵ -SQSSVR.

The paper is structured as follows. Section 2 introduces a brief introduction to the ϵ -SQSSVR model, and some definitions and notations. In Section 3, we construct the primal and dual problems for NQSSVR and analyze the corresponding properties. Section 4 presents the results of numerical experiments conducted on datasets. Finally, Section 5 provides conclusions from this study.

2. Background

In this section, we give the related definitions and notations, and review the ϵ -SQSSVR model.

2.1. Definition and Notations

The following mathematical notations are utilized in this paper. Lowercase bold and uppercase bold represent vectors and matrices, respectively. I is the identity matrix of any size, \mathbb{S}^m is the set of m -dimensional symmetric matrices, $\mathbb{R}^{n \times m}$ is the set of $n \times m$ dimensional matrices. Next, define the operators as follows.

Definition 1. For any symmetry matrix $\mathbf{U} = (u_{kl})_{m \times m} \in \mathbb{S}^m$, its half-vectorization operator can be defined as follows:

$$\text{hvec}(\mathbf{U}) = (u_{11}, \dots, u_{1m}, u_{22}, \dots, u_{2m}, \dots, u_{mm})^T \in \mathbb{R}^{\frac{m(m+1)}{2}}. \tag{1}$$

Definition 2. For any vector $\mathbf{u} = (u_1, u_2, \dots, u_m)^T \in \mathbb{R}^m$, define the following quadratic operator:

$$\text{lvec}(\mathbf{u}) = \left(\frac{1}{2}u_1u_1, \dots, u_1u_m, \frac{1}{2}u_2u_2, \dots, u_2u_m, \dots, \frac{1}{2}u_mu_m\right)^T \in \mathbb{R}^{\frac{m(m+1)}{2}}. \tag{2}$$

Definition 3. For a vector $\mathbf{u} = (u_1, u_2, \dots, u_m)^T \in \mathbb{R}^m$, define the vector-to-matrix operator as follows:

$$\text{mat}(\mathbf{u}) \triangleq \begin{bmatrix} u_1 & u_2 & \cdots & u_m & 0 & \cdots & 0 & \cdots & 0 \\ 0 & u_1 & \cdots & 0 & u_2 & \cdots & u_m & \cdots & 0 \\ 0 & 0 & \ddots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & u_1 & 0 & \cdots & u_2 & \cdots & u_m \end{bmatrix} \in \mathbb{R}^{m \times \frac{m(m+1)}{2}}. \tag{3}$$

2.2. ϵ -SQSSVR

Given the training set

$$T = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_n, y_n)\}, \tag{4}$$

where $\mathbf{x}_i \in \mathbb{R}^m, y_i \in \mathbb{R}, i = 1, 2, \dots, n$. The task of ϵ -SQSSVR is to seek the quadratic regression function

$$g(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{W}\mathbf{x} + \mathbf{b}^T\mathbf{x} + c, \tag{5}$$

where $\mathbf{W} \in \mathbb{S}^m, \mathbf{b} \in \mathbb{R}^m$, and $c \in \mathbb{R}$. To obtain the regression function (5), the optimization problem is established as follows:

$$\min_{\mathbf{W}, \mathbf{b}, c, \boldsymbol{\xi}^{(*)}} \frac{1}{2} \sum_{i=1}^n \|\mathbf{W}\mathbf{x}_i + \mathbf{b}\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*), \tag{6}$$

$$\text{s.t.} \quad \left(\frac{1}{2}\mathbf{x}_i^T\mathbf{W}\mathbf{x}_i + \mathbf{b}^T\mathbf{x}_i + c\right) - y_i \leq \epsilon + \xi_i, \quad i = 1, \dots, n, \tag{7}$$

$$y_i - \left(\frac{1}{2}\mathbf{x}_i^T\mathbf{W}\mathbf{x}_i + \mathbf{b}^T\mathbf{x}_i + c\right) \leq \epsilon + \xi_i^*, \quad i = 1, \dots, n, \tag{8}$$

$$\xi_i \geq 0, \xi_i^* \geq 0, \quad i = 1, 2, \dots, n, \tag{9}$$

where $C > 0$ is a penalty parameter, and $\boldsymbol{\xi}^{(*)} = (\xi_1, \xi_1^*, \dots, \xi_n, \xi_n^*)^T$ is the slack vector. The optimization problem (6)–(9) is a quadratic programming problem, so it can be solved directly. In addition, this model uses the quadratic surface kernel-free technique, which avoids the difficulty of choosing the kernel function and the corresponding parameters.

3. Kernel-Free QSSVR with Non-Negative Constraints (NQSSVR)

In this section, we establish the primal and dual problems of the kernel-free QSSVR with the non-negative constraints (NQSSVR). The properties of primal and dual problems are discussed, and the properties of the regression function with non-negative constraints are proved.

3.1. Primal Problem

Given the training set T (4), to find the regression function (5), the following optimization problem is formulated

$$\min_{\mathbf{W}, \mathbf{b}, c, \boldsymbol{\zeta}^{(*)}} \quad \frac{1}{2} \sum_{i=1}^n \|\mathbf{W}x_i + \mathbf{b}\|^2 + C \sum_{i=1}^n (\zeta_i + \zeta_i^*), \tag{10}$$

$$\text{s.t.} \quad \left(\frac{1}{2}x_i^T \mathbf{W}x_i + \mathbf{b}^T x_i + c\right) - y_i \leq \epsilon + \zeta_i, \quad i = 1, 2, \dots, n, \tag{11}$$

$$y_i - \left(\frac{1}{2}x_i^T \mathbf{W}x_i + \mathbf{b}^T x_i + c\right) \leq \epsilon + \zeta_i^*, \quad i = 1, 2, \dots, n, \tag{12}$$

$$w_{kl} \geq 0, b_k \geq 0, k, l = 1, \dots, m, \tag{13}$$

$$\zeta_i \geq 0, \zeta_i^* \geq 0, i = 1 \dots, n, \tag{14}$$

where $\mathbf{W} = (w_{kl})_{m \times m} \in \mathbb{S}^m$, $\mathbf{b} = (b_1, \dots, b_m) \in \mathbb{R}^m$, $c \in \mathbb{R}$. $w_{kl} \geq 0, b_k \geq 0, k, l = 1, \dots, m$ mean that each component of \mathbf{W} and \mathbf{b} is greater than or equal to zero. $C > 0$ is the penalty parameter, and $\boldsymbol{\zeta}^{(*)} = (\zeta_1, \zeta_2^*, \dots, \zeta_n, \zeta_n^*)^T$ is a slack vector.

In the above optimization problem (10)–(14), we impose constraints on the regression coefficients, namely $w_{kl} \geq 0, b_k \geq 0, k, l = 1, \dots, m$. Restricting the range of values of regression coefficients can help us to obtain regression functions that are more consistent with a priori information. In addition, the optimization problem does not involve kernel functions, which can avoid the complicated process of kernel functions, and its parameters selection further reduced computation time.

According to Definitions 1–3, the primal optimization problem (10)–(14) is simplified to the following form:

$$\min_{\mathbf{z}, c, \boldsymbol{\zeta}^{(*)}} \quad \frac{1}{2} \mathbf{z}^T \mathbf{G} \mathbf{z} + C \sum_{i=1}^n (\zeta_i + \zeta_i^*), \tag{15}$$

$$\text{s.t.} \quad (\mathbf{z}^T \mathbf{s}_i + c) - y_i \leq \epsilon + \zeta_i, \quad i = 1 \dots, n, \tag{16}$$

$$y_i - (\mathbf{z}^T \mathbf{s}_i + c) \leq \epsilon + \zeta_i^*, \quad i = 1 \dots, n, \tag{17}$$

$$\mathbf{z} \geq 0, \tag{18}$$

$$\zeta_i \geq 0, \zeta_i^* \geq 0, i = 1 \dots, n, \tag{19}$$

where $\mathbf{z} = [\mathbf{w}, \mathbf{b}] \in \mathbb{R}^{\frac{m(m+3)}{2}}$, $\mathbf{w} \triangleq \text{hvec}(\mathbf{W})$, $\mathbf{G} = \sum_{i=1}^n (\mathbf{V}^i)^T \mathbf{V}^i$, $\mathbf{V}_i \triangleq [\text{mat}(x_i), \mathbf{I}] \in \mathbb{R}^{m \times (\frac{m(m+3)}{2})}$, $c \in \mathbb{R}$, $\mathbf{I} \in \mathbb{S}^m$, $\mathbf{s}_i \triangleq \text{lvec}(x_i)$. $\mathbf{z} \geq 0$ means that each component of \mathbf{z} is greater than or equal to zero. The matrix \mathbf{G} is positive semidefinite matrix, since $\mathbf{G} = \sum_{i=1}^n (\mathbf{V}^i)^T \mathbf{V}^i \geq 0$.

3.2. Dual Model

Next, the Lagrange multiplier vectors $\boldsymbol{\alpha}^{(*)} = (\alpha_1, \alpha_1^*, \dots, \alpha_n, \alpha_n^*)^T$, $\boldsymbol{\eta}^{(*)} = (\eta_1, \eta_1^*, \dots, \eta_n, \eta_n^*)^T$, and $\boldsymbol{\beta} \in \mathbb{R}^{\frac{m(m+3)}{2}}$ are introduced to the optimization problem (15)–(19). The Lagrange function can be expressed as follows:

$$\begin{aligned} L(\mathbf{z}, c, \boldsymbol{\zeta}^{(*)}, \boldsymbol{\alpha}^{(*)}, \boldsymbol{\eta}^{(*)}, \boldsymbol{\beta}) &= \frac{1}{2} \mathbf{z}^T \mathbf{G} \mathbf{z} + C \sum_{i=1}^n (\zeta_i + \zeta_i^*) - \boldsymbol{\beta}^T \mathbf{z} \\ &\quad - \sum_{i=1}^n \alpha_i (y_i + \epsilon + \zeta_i - (\mathbf{z}^T \mathbf{s}_i + c)) - \sum_{i=1}^n \eta_i \zeta_i \\ &\quad - \sum_{i=1}^n \alpha_i^* (\epsilon + \zeta_i + (\mathbf{z}^T \mathbf{s}_i + c) - y_i) - \sum_{i=1}^n \eta_i^* \zeta_i^*. \end{aligned} \tag{20}$$

The Karush–Kuhn–Tucker (KKT) conditions for problems (15)–(19) are given as follows:

$$\nabla_z L = \mathbf{G}z - \sum_{i=1}^n (\alpha_i - \alpha_i^*) \mathbf{s}_i - \boldsymbol{\beta} = 0, \tag{21}$$

$$\nabla_c L = \sum_{i=1}^n (\alpha_i - \alpha_i^*) = 0, \tag{22}$$

$$\nabla_{\zeta_i} L = C - \alpha_i - \eta_i = 0, \quad i = 1 \dots, n, \tag{23}$$

$$\nabla_{\zeta_i^*} L = C - \alpha_i^* - \eta_i^* = 0, \quad i = 1 \dots, n, \tag{24}$$

$$\epsilon + \zeta_i - (\mathbf{z}^T \mathbf{s}_i + c) + y_i \geq 0, \zeta_i \geq 0, \quad i = 1, \dots, n, \tag{25}$$

$$\epsilon + \zeta_i^* - y_i + (\mathbf{z}^T \mathbf{s}_i + c) \geq 0, \zeta_i^* \geq 0, \quad i = 1, \dots, n, \tag{26}$$

$$\alpha_i (y_i + \epsilon + \zeta_i - \mathbf{s}_i^T \mathbf{z} - c) = 0, \alpha_i \geq 0, \quad i = 1, \dots, n, \tag{27}$$

$$\alpha_i^* (-y_i + \epsilon + \zeta_i + \mathbf{s}_i^T \mathbf{z} + c) = 0, \alpha_i^* \geq 0, \quad i = 1, \dots, n, \tag{28}$$

$$\eta_i^* \zeta_i^* = 0, \eta_i \zeta_i = 0, \eta_i^* \geq 0, \eta_i \geq 0, \quad i = 1 \dots, n, \tag{29}$$

$$\boldsymbol{\beta} \mathbf{z} = 0, \boldsymbol{\beta} \geq 0. \tag{30}$$

According to the Equation (21),

$$\mathbf{z} = \mathbf{G}^{-1} \left(\sum_{i=1}^n (\alpha_i - \alpha_i^*) \mathbf{s}_i + \boldsymbol{\beta} \right). \tag{31}$$

By substituting (21)–(30) into (20), the dual problem of the optimization problem (15)–(19) is formulated as

$$\begin{aligned} \min_{\boldsymbol{\alpha}^{(*)}, \boldsymbol{\beta}} \quad & \frac{1}{2} \left(\sum_{i=1}^n (\alpha_i - \alpha_i^*) \mathbf{s}_i + \boldsymbol{\beta} \right)^T \mathbf{G}^{-1} \left(\sum_{i=1}^n (\alpha_i - \alpha_i^*) \mathbf{s}_i + \boldsymbol{\beta} \right) \\ & + \sum_{i=1}^n (\alpha_i + \alpha_i^*) \epsilon - \sum_{i=1}^n (\alpha_i - \alpha_i^*) y_i, \end{aligned} \tag{32}$$

$$\text{s.t.} \quad \sum_{i=1}^n (\alpha_i - \alpha_i^*) = 0, \tag{33}$$

$$0 \leq \alpha_i, \alpha_i^* \leq C, \quad i = 1 \dots, n, \tag{34}$$

$$\beta_j \geq 0, \quad j = 1, \dots, \frac{m(m+3)}{2}, \tag{35}$$

where $\boldsymbol{\alpha}^{(*)} = (\alpha_1, \alpha_1^*, \dots, \alpha_n, \alpha_n^*)^T$ and $\boldsymbol{\beta} = (\beta_1, \dots, \beta_{\frac{m(m+3)}{2}})^T$ are vectors, and $C > 0$ is the penalty parameter. Since the proposed model does not involve kernel functions, both the primal and dual problems can be solved directly. Solving the primal problem saves the cost of computing the inverse matrix. In particular, when the input dimension is large, solving the dual problem is more convenient.

3.3. Some Theoretical Analysis

In this subsection, the theoretical properties of the primal and dual problems, as well as the regression function after adding the non-negative constraints, are analyzed into properties.

Theorem 1. *Given the training set T (4) and $C > 0$, if \mathbf{G} is a positive definite matrix and $(\mathbf{z}^*, c^*, \zeta^*, \zeta^{**})$ is the optimal solution to the primal problem (15)–(19), then \mathbf{z}^* is unique.*

Proof. Suppose $(\bar{\mathbf{z}}, \bar{c}, \bar{\zeta}, \bar{\zeta}^*)$ and $(\hat{\mathbf{z}}, \hat{c}, \hat{\zeta}, \hat{\zeta}^*)$ are the two optimal solutions of the primal problem. There exists $\mu \in (0, 1)$ such that the following equation holds:

$$(\mathbf{z}, c, \zeta, \zeta^*) = \mu (\bar{\mathbf{z}}, \bar{c}, \bar{\zeta}, \bar{\zeta}^*) + (1 - \mu) (\hat{\mathbf{z}}, \hat{c}, \hat{\zeta}, \hat{\zeta}^*). \tag{36}$$

Clearly, $(z, c, \bar{\zeta}, \bar{\zeta}^*)$ is also an optimal solution to the optimization problem (15)–(19), so the following inequalities hold:

$$\frac{1}{2}z^T Gz + C \sum_{i=1}^n (\bar{\zeta}_i + \bar{\zeta}_i^*) \geq \frac{1}{2}\bar{z}^T G\bar{z} + C \sum_{i=1}^n (\bar{\zeta}_i + \bar{\zeta}_i^*), \tag{37}$$

$$\frac{1}{2}z^T Gz + C \sum_{i=1}^n (\bar{\zeta}_i + \bar{\zeta}_i^*) \geq \frac{1}{2}\hat{z}^T G\hat{z} + C \sum_{i=1}^n (\hat{\zeta}_i + \hat{\zeta}_i^*). \tag{38}$$

Multiplying inequality (37) by μ and inequality (38) by $(1 - \mu)$, then summing them up yields

$$\frac{1}{2}z^T Gz + C \sum_{i=1}^n (\bar{\zeta}_i + \bar{\zeta}_i^*) \geq \frac{\mu}{2}\hat{z}^T G\hat{z} + \frac{1-\mu}{2}\bar{z}^T G\bar{z} + C \sum_{i=1}^n (\mu\hat{\zeta}_i + (1-\mu)\bar{\zeta}_i + \mu\bar{\zeta}_i^* + (1-\mu)\hat{\zeta}_i^*). \tag{39}$$

By simplifying the Formula (39), we obtain

$$\frac{1}{2}(\mu\bar{z} + (1-\mu)\hat{z})^T G(\mu\bar{z} + (1-\mu)\hat{z}) \geq \frac{\mu}{2}\hat{z}^T G\hat{z} + \frac{1-\mu}{2}\bar{z}^T G\bar{z}. \tag{40}$$

Then, we have

$$\mu(1-\mu)(\bar{z} - \hat{z})^T G(\bar{z} - \hat{z}) \leq 0. \tag{41}$$

Since the matrix G is a positive definite matrix and $\mu \in (0, 1)$, inequality (41) holds if and only if $\bar{z} = \hat{z}$. \square

Theorem 2. For the training set T (4) and $C > 0$, if the matrix G is positive definite, the optimal solution $\alpha^{(*)} = (\alpha_1, \alpha_1^*, \dots, \alpha_n, \alpha_n^*)^T$ of the dual problem (32)–(35) exists and is unique, and the optimal solution of the primal problem (15)–(19) can be expressed as

$$z = G^{-1} \left(\sum_{i=1}^n (\alpha_i - \alpha_i^*) s_i + \beta \right) \tag{42}$$

$$c = y_j + \epsilon - \left(\sum_{i=1}^n (\alpha_i - \alpha_i^*) s_i + \beta \right) G^{-1} s_j, \quad \text{for some } \alpha_j \in (0, C), \tag{43}$$

or

$$c = y_k - \epsilon - \left(\sum_{i=1}^n (\alpha_i - \alpha_i^*) s_i + \beta \right) G^{-1} s_k, \quad \text{for some } \alpha_k^* \in (0, C). \tag{44}$$

Proof. By the (21) equation in the KKT condition, we have

$$z = G^{-1} \left(\sum_{i=1}^n (\alpha_i - \alpha_i^*) s_i + \beta \right). \tag{45}$$

If $\alpha^{(*)}$ there exists components, α_j and α_k^* , such that $\alpha_j \in (0, C)$ or $\alpha_j^* = 0, \alpha_k = 0$ or $\alpha_k^* \in (0, C)$, by the complementary slackness conditions, $\bar{\zeta}_j = \bar{\zeta}_k^* = 0$, we have

$$c = y_j + \epsilon - \left(\sum_{i=1}^n (\alpha_i - \alpha_i^*) s_i + \beta \right) G^{-1} s_j, \quad \alpha_j \in (0, C), \tag{46}$$

or

$$c = y_k - \epsilon - \left(\sum_{i=1}^n (\alpha_i - \alpha_i^*) s_i + \beta \right) G^{-1} s_k, \quad \alpha_k^* \in (0, C). \tag{47}$$

\square

Next, the properties of the regression function (5) after adding non-negative constraints with respect to the regression coefficients are analyzed. The domain D is defined as follows:

$$D = \{x = ([x]_1, \dots, [x]_m)^T \mid [x]_1 \geq 0, \dots, [x]_m \geq 0\}. \tag{48}$$

Theorem 3. *On the domain D (48), the regression function $g(x)$ (5) is monotonically non-decreasing respect to each component of x if and only if the regression coefficients have the following restrictions:*

$$\begin{aligned} w_{kl} &\geq 0, k, l = 1, \dots, m, \\ b_k &\geq 0, k = 1, \dots, m, \end{aligned} \tag{49}$$

where $w_{kl} \geq 0$, and $b_k \geq 0, k, l = 1, \dots, m$ mean that each component of W and b is greater than or equal to zero.

Proof. The function $g(x)$ can be written as

$$g(x) = \frac{1}{2} \sum_{k,l=1}^m [x]_k w_{kl} [x]_l + \sum_{k=1}^m b_k [x]_k + c. \tag{50}$$

It only remains to justification that this holds for the k -th component of x , the quadratic function containing $[x]_k$ can be expressed as

$$g([x]_k) = \frac{1}{2} [x]_k w_{kk} [x]_k + [x]_k \sum_{l \neq k}^m w_{kl} [x]_l + b_k [x]_k + c. \tag{51}$$

Taking the derivative of the above equation yields

$$\frac{\partial g([x]_k)}{\partial [x]_k} = \sum_{l=1}^m w_{kl} [x]_l + b_k. \tag{52}$$

On the domain D , the function $g([x]_k)$ monotonically non-decreasing is equivalent to being non-negative at the right end of the above equation, so it is a necessary and sufficient condition to prove that the latter holds as follows:

$$\begin{aligned} w_{kl} = w_{lk} &\geq 0, l = 1, \dots, m, \\ b_k &\geq 0, k = 1, \dots, m. \end{aligned} \tag{53}$$

Sufficiency is obvious, and we only need to prove necessity. Supposing $b_k < 0$ and taking $x = ([x]_1, \dots, [x]_m)^T = 0$, then we obtain the following:

$$\frac{\partial g([x]_k)}{\partial [x]_k} = \sum_{l=1}^m w_{kl} [x]_l + b_k = b_k < 0. \tag{54}$$

The above relation is contrary to the known conditions. Supposing the existence of $w_{kl} < 0$ with all other components being zero. Obviously, when the $[x]_l$ is sufficiently large, the following formula holds:

$$\frac{\partial g([x]_k)}{\partial [x]_k} = \sum_{l=1}^m w_{kl} [x]_l + b_k = w_{kl} [x]_l + b_k < 0. \tag{55}$$

This is a contradict. Similarly, it can be shown that Theorem 3 holds. \square

4. Numerical Experiments

To verify the validity of our proposed NQSSVR model, we compare it with other methods, including linear SVR (lin-SVR), SVR with Gaussian kernel (rbf-SVR), and polynomial kernel (poly-SVR), and linear SVR with non-negative constraints (NNSVR), as well as QLSSVR and ϵ -SQSSVR. The primal and dual problems of the NQSSVR method are denoted as NQSSVR(p) and NQSSVR(d), respectively. The above experiments are tested on 2 artificial datasets, 7 UCI [30] datasets, and AQCI datasets. All numerical experiments in this section are conducted on a computer equipped with a 2.50GHz (i7-9700) CPU and 8G RAM using MatlabR2016(a).

To validate the fitting performances of various methods, the following four evaluation criteria are introduced as shown in Table 1. Without loss of generality, let \hat{y}_i and \bar{y}_i be the predicted and mean values, respectively. The penalty parameters C and ϵ -insensitive parameter, as well as the Gaussian kernel parameter σ , are selected from $\{2^i \mid i = -6, -5, \dots, 5, 6\}$, while the polynomial kernel parameter p is selected from $\{1, 2\}$. All methods are selected through 5-fold cross-validation to obtain the optimal parameters.

Table 1. Evaluation criteria.

Evaluation Criteria	Formulas
R^2	$R^2 = \frac{SSR}{SST} = \frac{\sum_{i=1}^n (\hat{y}_i - \bar{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y}_i)^2}$
Mean Absolute Error(MAE)	$MAE = \frac{1}{n} \sum_{i=1}^n y_i - \hat{y}_i $
Root Mean Squared Error (RMSE)	$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$
T1	Average test time
T2	Time to select parameters

4.1. Artificial Datasets

The 2 artificial datasets are conducted to validate the performance of the NQSSVR model.

Example 1.

$$y_i = x_i + 0.3 + \zeta, \quad x_i \in [0, 1], \quad \zeta \in N(0, 0.1^2), \quad i = 1, \dots, 100. \tag{56}$$

The data points are indicated by magenta “o”, the red solid line indicates the target function, and the blue dashed line indicates the regression function obtained by the kernel function method. The green dashed line and the black solid line indicate NQSSVR(p), and NQSSVR(d), respectively. The regression functions obtained by NQSSVR and different kernel functions are shown in Figure 1. From Figure 1, it can be seen that NQSSVR yields an approximately linear regression function as well as the other three methods. The regression coefficients obtained by solving the primal and dual problems of our proposed method are $w = 0.0001, b = 1.0050, c = 0.2897$ and $w = 0.0001, b = 0.9946, c = 0.3097$, respectively. So, NQSSVR can handle linear regression problem.

Example 2.

$$z_i = x_i^2 + y_i^2 + 0.1 + \zeta, \quad x_i, y_i \in [0, 1], \quad \zeta \in N(0, 0.2^2), \quad i = 1, \dots, 200. \tag{57}$$

The fitting results of an artificial dataset are shown in Figure 2. The data points are denoted by “.”. Figure 2a,b present the regression surfaces obtained from the primal and dual problems, respectively. On this dataset, the regression coefficients obtained by our method are $W = [0.9982, 0.0000; 0.0000, 0.9956]$, $b = [0.0000, 0.0000]^T$, $c = 0.1005$, and $W = [0.9896, 0.0000; 0.0000, 0.9967]$, $b = [0.0000, 0.0000]^T$, $c = 0.1034$. From Figure 2 and the regression coefficients, the quadratic surface fitted by our method can be matched to the actual distribution of the data.

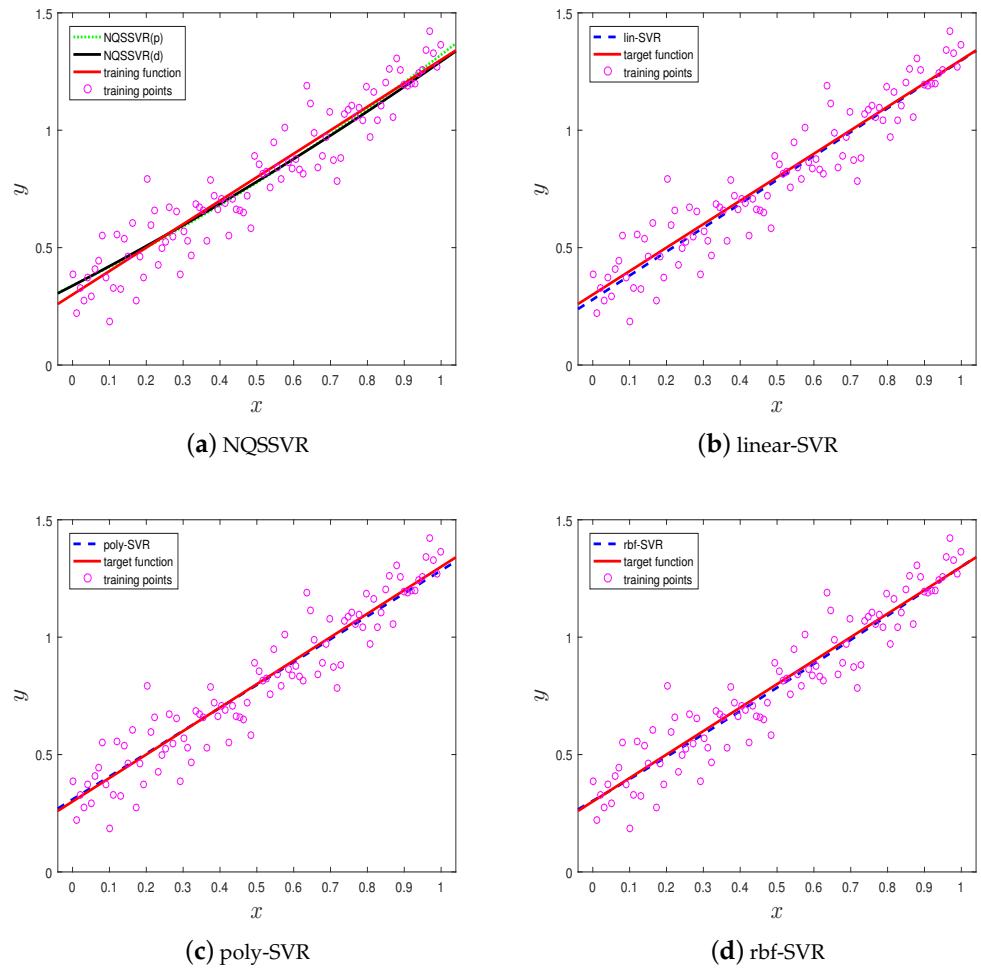


Figure 1. The visualization results on Example 1.

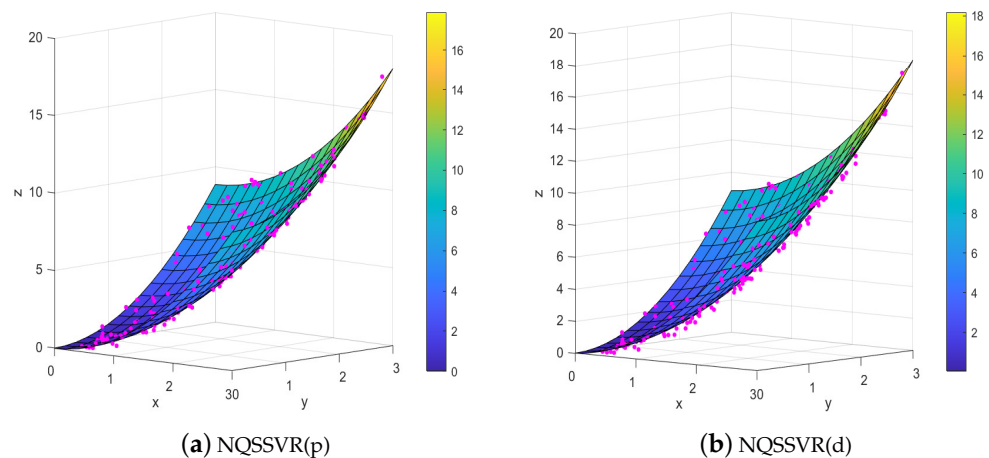


Figure 2. The visualization results on Example 2.

Table 2 shows the results of the our proposed model and SVR with kernel functions on the two datasets mentioned above. When considering the linear regression problems, it is evident that the five models yield n similar outcomes. However, when addressing the non-linear data, our method demonstrates superior performance compared to the other three methods, as evidenced by the smaller average values of RMSE and MAE. Moreover, the difference in R^2 values between our method and the optimal result is minimal. Notably,

the T2 values reveal that our method exhibits faster computation times than SVR with kernel function. This advantage comes from the fact that it does not contain a kernel function, thus eliminating the need for kernel parameter selection.

Table 2. Experimental results on artificial datasets.

Datasets	Algorithms	RMSE	MAE	R ²	T1	T2
Example 1	lin-SVR	0.0901 ± 0.0010	0.0716 ± 0.0011	0.9159 ± 0.0118	0.2085 ± 0.0188	31.2781
	poly-SVR	0.0894 ± 0.0015	0.0716 ± 0.0012	0.9502 ± 0.0134	0.1412 ± 0.0096	36.9959
	rbf-SVR	0.0865 ± 0.0010	0.0658 ± 0.0014	0.9680 ± 0.0153	0.1554 ± 0.0055	62.9346
	NQSSVR(p)	0.0885 ± 0.0012	0.0714 ± 0.0018	0.9066 ± 0.0260	0.0612 ± 0.0094	16.2336
	NQSSVR(d)	0.0889 ± 0.0007	0.0716 ± 0.0004	0.8829 ± 0.0022	0.1278 ± 0.0019	24.2926
Example 2	lin-SVR	1.4762 ± 0.0085	1.2420 ± 0.0074	0.5141 ± 0.0021	0.4143 ± 0.0204	133.6888
	poly-SVR	0.1983 ± 0.0015	0.1567 ± 0.0018	0.9941 ± 0.0017	0.3464 ± 0.0413	130.0461
	rbf-SVR	0.2057 ± 0.0066	0.1615 ± 0.0047	0.9899 ± 0.0014	0.4600 ± 0.0282	573.8611
	NQSSVR(p)	0.1969 ± 0.0016	0.1544 ± 0.0014	0.9930 ± 0.0014	0.1850 ± 0.0139	28.3872
	NQSSVR(d)	0.1982 ± 0.0017	0.1552 ± 0.0017	0.9896 ± 0.0012	0.2056 ± 0.0234	32.7823

For example 2, the influence of the parameter on the accuracy of our proposed method is analyzed. As can be seen by Figure 3, the penalty parameters C and insensitive loss parameter ϵ have a greater impact on the accuracy of the NQSSVR model. So, reasonable parameters can improve the accuracy of the model. In the next experiments, we choose the optimal parameters for the model within the defined parameter range.

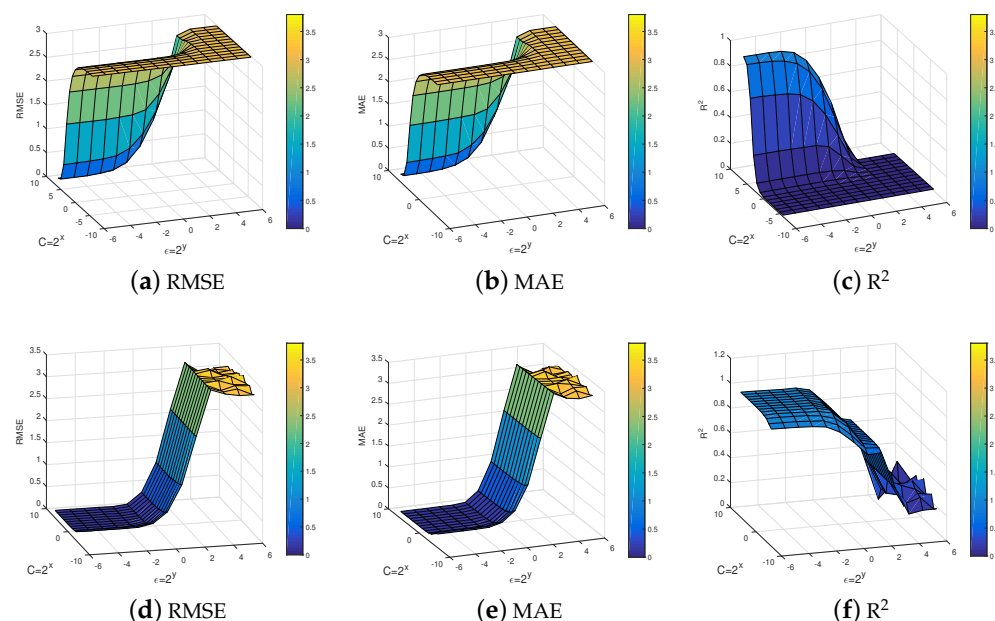


Figure 3. Effect of parameters on method performance NQSSVR.

Next, the average test time is compared for NQSSVR(p), NQSSVR(d), rbf-SVR, and poly-SVR. Since lin-SVR is only applicable to linear regression, no comparison is made here. The CPU running time of the above four methods in different dimensions and data points is shown in Table 3. Where the input dimensions m of data point are 2, 4, 8, and 16 and the number n of data points is 200, 400, 600, 800, and 1000, respectively. It is noteworthy that the time variation of NQSSVR(p) remains small as the number of data points increases for the same input dimension, outperforming both the rbf-SVR and poly-SVR methods. Moreover, when the number of data points is consistent, NQSSVR(p) exhibits shorter average test time costs compared to rbf-SVR and poly-SVR. Furthermore, as the input dimension of

data points increases, the average test time for the dual problem is found to be shorter than that for the primal problem.

Table 3. Experimental results on the benchmark datasets.

Data Points\Dimensions	Methods	m = 2	m = 4	m = 8	m = 16
n = 200	poly-SVR	3.9670 ± 0.1183	3.3426 ± 0.0813	3.6770 ± 0.0596	3.8512 ± 0.0430
	rbf-SVR	1.1682 ± 0.0318	1.0756 ± 0.0575	1.0694 ± 0.0437	1.1634 ± 0.0173
	NQSSVR(p)	0.1592 ± 0.0146	0.2542 ± 0.0066	0.4346 ± 0.0193	1.0058 ± 0.0608
	NQSSVR(d)	0.7322 ± 0.0557	0.8390 ± 0.0219	0.9502 ± 0.0449	1.1458 ± 0.0404
n = 400	poly-SVR	22.1766 ± 0.6565	21.0828 ± 0.6835	20.7110 ± 0.4407	23.2322 ± 0.7712
	rbf-SVR	4.1252 ± 0.1725	4.2152 ± 0.1546	4.2804 ± 0.1735	4.4358 ± 0.1441
	NQSSVR	0.3942 ± 0.0199	0.5334 ± 0.0219	0.9442 ± 0.0281	3.2328 ± 0.1464
	NQSSVR(d)	3.2784 ± 0.1339	3.8628 ± 0.0954	4.2040 ± 0.3310	4.1320 ± 0.1463
n = 600	poly-SVR	64.8220 ± 1.0185	64.9902 ± 1.6431	63.5246 ± 1.3031	69.9104 ± 2.0957
	rbf-SVR	9.6822 ± 0.4247	9.5922 ± 0.4015	10.0280 ± 0.6436	10.7400 ± 0.5887
	NQSSVR	0.6522 ± 0.0114	0.8818 ± 0.0064	1.4464 ± 0.0016	4.4658 ± 0.0995
	NQSSVR(d)	8.8504 ± 0.4567	9.4208 ± 0.3310	10.9936 ± 0.4588	12.3712 ± 1.3552
n = 800	poly-SVR	157.6060 ± 5.8607	139.2794 ± 7.3092	161.7490 ± 2.2436	163.0944 ± 4.0222
	rbf-SVR	19.5872 ± 1.2377	17.9632 ± 1.2597	18.5810 ± 0.3514	19.8404 ± 1.4977
	NQSSVR	0.9408 ± 0.0407	1.2254 ± 0.0537	1.9072 ± 0.0832	6.2810 ± 0.2554
	NQSSVR(d)	16.5370 ± 0.5350	19.1804 ± 0.8508	23.1990 ± 1.2128	26.5350 ± 0.9734
n = 1000	poly-SVR	284.1451 ± 14.6313	288.7770 ± 11.1508	272.6956 ± 10.7412	141.2644 ± 8.5463
	rbf-SVR	30.3622 ± 2.0704	32.5590 ± 2.4960	29.5694 ± 2.4395	30.9340 ± 9.9135
	NQSSVR(p)	1.4494 ± 0.1954	1.7290 ± 0.0310	2.2676 ± 0.0452	5.3260 ± 0.1937
	NQSSVR(d)	24.7908 ± 1.2614	28.3166 ± 1.4892	33.8146 ± 2.2722	44.1248 ± 1.5144

4.2. Benchmark Datasets

In this section, to further validate the reliability of the proposed method, the NQSSVR model is compared with the lin-SVR, poly-SVR, rbf-SVR, NNSVR, QLSSVR, and ϵ -SQSSVR models on seven benchmark datasets. Details of all datasets are listed in Table 4. All datasets are normalized before conducting data experiments, and are divided into training datasets, test datasets and a validation datasets in a ratio of 3:1:1. All methods are compared on evaluation criteria: MAE, RMSE, T1, T2. The top two relatively better results are highlighted in bold. All results were repeated 5 times and their mean values are calculated.

Table 4. Details of the benchmark datasets.

Datasets	Abbreviations	Sample Points	Attributes
Concrete Slump Test	Concrete	103	7
Computer Hardware	Computer	209	9
Yacht Hydrodynamics	Yacht	308	7
Forest Fires	Forest	517	13
Energy efficiency (Heating)	Energy(H)	768	8
Energy efficiency (Cooling)	Energy(C)	768	8
Air quality	Air	1067	6

Table 5 lists the regression results of the eight methods on the seven datasets. In terms of the evaluation criteria RMSE and MAE, it can be seen that NQSSVR is significantly better than lin-SVR and NNSVR. For most of the datasets, the NQSSVR model outperforms QLSSVR and ϵ -SQSSVR, and is not significantly different from rbf-SVR and poly-SVR. In terms of time, our method is second only to QLSSVR and outperforms other methods.

Table 5. Experimental results on the datasets.

Datasets	Algorithms	RMSE	MAE	T1	T2
Concrete	lin-SVR	0.0703 ± 0.0032	0.0499 ± 0.0027	0.1618 ± 0.0122	24.2926
	poly-SVR	0.0476 ± 0.0016	0.0301 ± 0.0011	0.1708 ± 0.0130	49.2486
	rbf-SVR	0.0406 ± 0.0023	0.0275 ± 0.0015	0.1544 ± 0.0116	170.5862
	NNSVR	0.0752 ± 0.0021	0.0597 ± 0.0020	0.0572 ± 0.0057	7.6194
	QLSSVR	0.0571 ± 0.0022	0.0453 ± 0.0017	0.0300 ± 0.0020	5.0718
	ϵ -SQSSVR	0.0378 ± 0.0011	0.0242 ± 0.0014	0.3366 ± 0.0184	44.0096
	NQSSVR(p)	0.0381 ± 0.0016	0.0296 ± 0.0011	0.1608 ± 0.0051	24.0741
	NQSSVR(d)	0.0317 ± 0.0300	0.0267 ± 0.0016	0.1432 ± 0.0020	18.5869
	lin-SVR	0.0393 ± 0.0014	0.0199 ± 0.0003	0.5438 ± 0.0303	64.1097
Computer	poly-SVR	0.0216 ± 0.0006	0.0084 ± 0.0002	0.5014 ± 0.0272	161.7245
	rbf-SVR	0.0179 ± 0.0011	0.0093 ± 0.0005	0.4554 ± 0.0193	507.4994
	NNSVR	0.0376 ± 0.0026	0.0217 ± 0.0009	0.0874 ± 0.0063	11.1911
	QLSSVR	0.0194 ± 0.0015	0.0099 ± 0.0004	0.0324 ± 0.0006	5.2495
	ϵ -SQSSVR	0.0139 ± 0.0007	0.0118 ± 0.0003	0.6674 ± 0.0230	74.2415
	NQSSVR(p)	0.0119 ± 0.0017	0.0077 ± 0.0007	0.2132 ± 0.0115	41.8217
	NQSSVR(d)	0.0098 ± 0.0012	0.0063 ± 0.0005	0.3018 ± 0.0131	26.7490
	lin-SVR	0.1428 ± 0.0010	0.1129 ± 0.0007	3.2864 ± 0.0893	360.9378
	poly-SVR	0.0676 ± 0.0204	0.0559 ± 0.0105	3.4052 ± 0.0965	798.5638
Yacht	rbf-SVR	0.0287 ± 0.0036	0.0204 ± 0.0011	1.6488 ± 0.0835	307.3457
	NNSVR	0.1547 ± 0.0008	0.1012 ± 0.0003	0.1876 ± 0.0182	27.4442
	QLSSVR	0.1056 ± 0.0005	0.0817 ± 0.0006	0.0620 ± 0.0110	8.7560
	ϵ -SQSSVR	0.0711 ± 0.0009	0.0551 ± 0.0007	1.8485 ± 0.1059	56.9315
	NQSSVR(p)	0.0965 ± 0.0006	0.0741 ± 0.0002	0.5432 ± 0.0105	67.1763
	NQSSVR(d)	0.0708 ± 0.0013	0.0533 ± 0.0010	1.8038 ± 0.0704	167.1142
	lin-SVR	0.0528 ± 0.0010	0.0196 ± 0.0001	17.3862 ± 0.7161	1561.1308
	poly-SVR	0.0486 ± 0.0014	0.0179 ± 0.0001	6.8746 ± 0.0634	1603.4728
	rbf-SVR	0.0465 ± 0.0013	0.0170 ± 0.0000	3.2826 ± 0.1706	812.7531
Fores	NNSVR	0.0498 ± 0.0010	0.0198 ± 0.0001	0.2688 ± 0.0151	45.1113
	QLSSVR	0.0500 ± 0.0009	0.0186 ± 0.0001	0.1756 ± 0.0170	29.1384
	ϵ -SQSSVR	0.0484 ± 0.0011	0.0196 ± 0.0001	0.7936 ± 0.1830	696.7686
	NQSSVR(p)	0.0475 ± 0.0031	0.0183 ± 0.0001	0.7360 ± 0.0168	120.5714
	NQSSVR(d)	0.0470 ± 0.0024	0.0175 ± 0.0001	9.1456 ± 0.4374	540.1738
	lin-SVR	0.0801 ± 0.0001	0.0556 ± 0.0001	15.0628 ± 0.6460	1559.3228
	poly-SVR	0.0298 ± 0.0001	0.0218 ± 0.0001	12.9536 ± 0.3113	3408.8491
	rbf-SVR	0.0237 ± 0.0004	0.0208 ± 0.0003	5.7626 ± 0.3335	10,338.7745
	NNSVR	0.0826 ± 0.0006	0.0572 ± 0.0001	0.3374 ± 0.0115	52.2095
Energy(H)	QLSSVR	0.0704 ± 0.0003	0.0499 ± 0.0002	0.1482 ± 0.0081	24.4898
	ϵ -SQSSVR	0.0685 ± 0.0010	0.0464 ± 0.0004	9.8748 ± 0.1596	1065.6020
	NQSSVR(p)	0.0516 ± 0.0002	0.0405 ± 0.0002	0.8624 ± 0.0274	105.5643
	NQSSVR(d)	0.0587 ± 0.0003	0.0423 ± 0.0001	16.8058 ± 0.6380	679.5034
	lin-SVR	0.0855 ± 0.0009	0.0592 ± 0.0014	21.4692 ± 0.5119	2708.5711
	poly-SVR	0.0623 ± 0.0001	0.0408 ± 0.0002	28.3624 ± 0.8986	6579.0492
	rbf-SVR	0.0398 ± 0.0018	0.0276 ± 0.0008	9.2944 ± 0.0669	13,515.7641
	NNSVR	0.0922 ± 0.0006	0.0583 ± 0.0004	0.5539 ± 0.0259	88.6845
	QLSSVR	0.0820 ± 0.0004	0.0572 ± 0.0003	0.2338 ± 0.0409	39.7269
Energy(C)	ϵ -SQSSVR	0.0795 ± 0.0008	0.0512 ± 0.0005	14.7644 ± 0.1254	1782.9128
	NQSSVR(p)	0.0681 ± 0.0002	0.0476 ± 0.0001	1.4326 ± 0.0303	175.3763
	NQSSVR(d)	0.0702 ± 0.0005	0.0463 ± 0.0002	10.3206 ± 0.2744	1161.6354
	lin-SVR	0.1965 ± 0.0004	0.1637 ± 0.0002	18.5452 ± 0.9660	3563.6484
	poly-SVR	0.1197 ± 0.0018	0.0884 ± 0.0001	32.7722 ± 0.7051	7896.3575
	rbf-SVR	0.1246 ± 0.0003	0.0727 ± 0.0002	12.2458 ± 0.3911	16,768.0504
	NNSVR	0.1963 ± 0.0002	0.1038 ± 0.0002	0.4592 ± 0.0041	77.6678
	QLSSVR	0.1346 ± 0.0002	0.0958 ± 0.0001	0.1078 ± 0.0077	17.8609
	ϵ -SQSSVR	0.1265 ± 0.0003	0.1033 ± 0.0003	20.6978 ± 0.1596	2066.3358
Air	NQSSVR(p)	0.1385 ± 0.0001	0.0936 ± 0.0002	0.6869 ± 0.0241	122.4225
	NQSSVR(d)	0.1458 ± 0.0007	0.0965 ± 0.0006	16.4676 ± 0.7046	677.5034

To compare the performances of our proposed method and other six methods, the Friedman test and post hoc test are employed. Initially, the Friedman test is conducted with the null hypothesis states that all methods have the same performances. Furthermore, we can calculate the Friedman statistics for each evaluation criterion using the following formula.

$$\tau_{\chi^2} = \frac{12N}{K(K+1)} \left(\sum_{i=1}^K R_i^2 - \frac{K(K+1)^2}{4} \right), \tag{58}$$

$$\tau_F = \frac{(N-1)\tau_{\chi^2}}{N(K-1) - \tau_{\chi^2}}, \tag{59}$$

where N and K are, respectively, the numbers of datasets and methods, R_i is the average rank of the i -th method.

According to the Formula (59), the Friedman statistics corresponding to the three criteria are 12.2124, 13.8361 and 35.1600, respectively. Next, for $\alpha = 0.05$, the critical value of Friedman statistic is calculated to be $F_\alpha = 2.2371$. Since the Friedman statistic on each regression criteria is greater than F_α , so we reject the null hypothesis. That is, these 8 methods have significantly different performances on the 3 evaluation criteria. To further compare the difference of each method, we proceed with a post hoc test. Specifically, if the difference of average ranks for two methods is larger than the critical difference (CD), then their performances are considered to be significantly different. Where the CD value can be calculated by the Formula (60)

$$CD = q_\alpha \sqrt{\frac{K(K+1)}{6N}}. \tag{60}$$

For $\alpha = 0.05$, we know $q_\alpha = 3.0308$. Thus, we obtain $CD = 3.9685$ by the Formula (60).

Figure 4 visually displays the results of Friedman test and Nemenyi post hoc test on three regression evaluation criteria, respectively. Where the average ranks of each method for three criteria are marked along an axis. The axis is turned so that the lowest (best) ranks is to the right of each criterion. Groups of methods that are not significantly different are linked by a red line. Statistically, the performance of NQSSVR (p) is not significantly different from rbf-SVR and poly-SVR in terms of RMSE, MAE. And our method ranks better than both the kernel-free quadratic surface and lin-SVR models on RMSE, MAE. In terms of time, our model ranks third and fourth, outperforming the SVR with kernel functions and ϵ -SQSSVR. In general, the comprehensive performance of our method is similar to rbf-SVR and poly-SVR, and completely superior to lin-SVR and NNSVR.

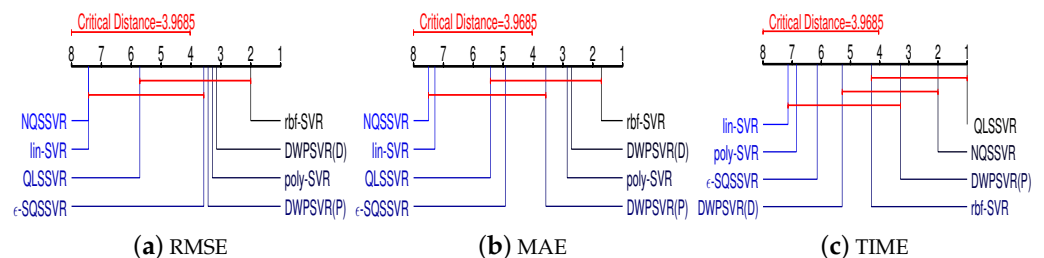


Figure 4. The results of Friedman test and Nemenyi post hoc test.

4.3. Air Quality Composite Index Dataset (AQCI)

This section uses two AQCI datasets, the monthly AQCI dataset and the daily AQCI dataset. These two datasets containing 18 data points and 841 data points, respectively. Each data point has six input features including nitrogen dioxide (NO_2), sulfur dioxide (SO_2), $\text{PM}_{2.5}$, ozone (O_3), carbon monoxide (CO), PM_{10} , respectively. And the output response is AQCI. Our method is compared with QLSSVR, ϵ -SQSSVR and NNSVR.

In Figure 5, the value of AQCI has a tendency to increase as the values of the remaining five input features increase, except for the O₃ feature. Therefore, the regression function we obtain should be monotonically increasing on the monthly AQCI dataset.

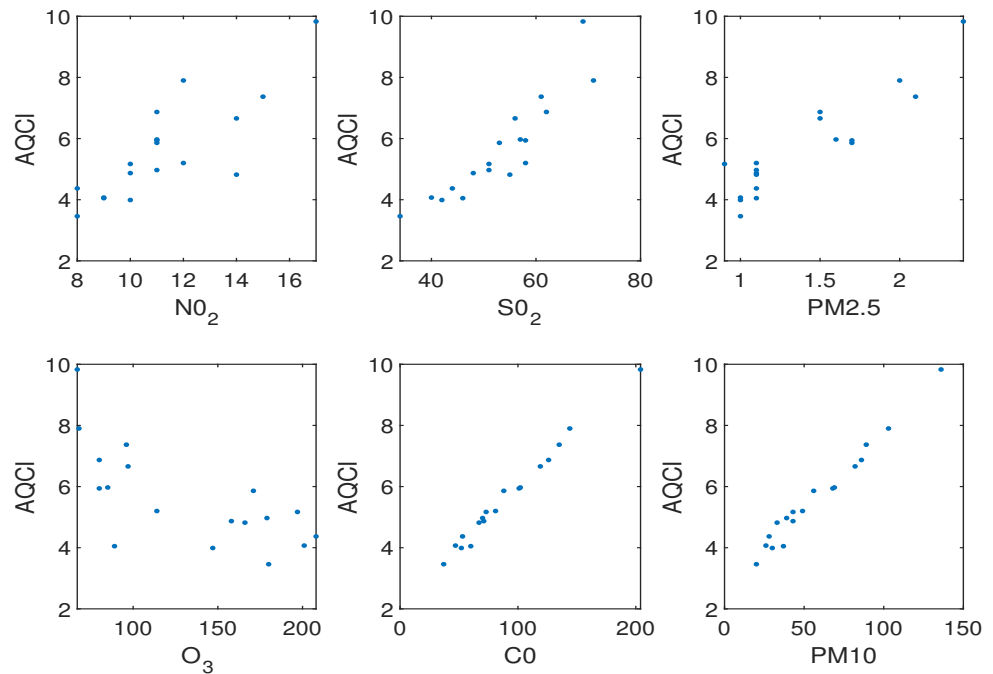


Figure 5. The relationship between six features and AQCI on monthly AQCI dataset.

Table 6 shows the experimental results of our NQSSVR and the other three methods on these two datasets. The accuracy of our model is better than that of QLSSVR and ϵ -SQSSVR on the datasets because our NQSSVR imposes non-negative constraints with respect to the regression coefficients. In addition, our model has greater accuracy than the NNSVR model. Because NNSVR can only obtain a linear regression function, but NQSSVR can obtain a quadratic regression function.

Table 6. Results on the AQCI datasets.

Datasets	Algorithms	RMSE	MAE	R ²	T1	T2
monthly	NNSVR	0.0274 ± 0.0001	0.0244 ± 0.0012	1.0058 ± 0.0635	0.0432 ± 0.0084	7.1174
	QLSSVR	0.0783 ± 0.0011	0.0668 ± 0.0008	0.1027 ± 0.1052	0.0094 ± 0.0015	0.1103
	ϵ -SQSSVR	0.1202 ± 0.0055	0.0898 ± 0.0052	0.9865 ± 0.1071	0.1356 ± 0.0534	1.8532
	NQSSVR(p)	0.0140 ± 0.0012	0.0115 ± 0.0011	1.0130 ± 0.0457	0.1600 ± 0.0156	2.3799
	NQSSVR(d)	0.0185 ± 0.0015	0.0156 ± 0.0011	1.0241 ± 0.0672	0.0482 ± 0.0072	0.4796
	NNSVR	0.0072 ± 0.0003	0.0060 ± 0.0001	1.0000 ± 0.0003	2.7365 ± 0.1292	37.1562
daily	QLSSVR	0.0071 ± 0.0001	0.0056 ± 0.0004	1.0000 ± 0.0002	0.1714 ± 0.0188	2.1832
	ϵ -SQSSVR	0.0071 ± 0.0002	0.0057 ± 0.0001	1.0001 ± 0.0001	32.4848 ± 0.5036	405.5820
	NQSSVR(p)	0.0062 ± 0.0001	0.0054 ± 0.0001	1.0002 ± 0.0001	2.9516 ± 0.0389	43.7415
	NQSSVR(d)	0.0067 ± 0.0001	0.0058 ± 0.0001	1.0000 ± 0.0001	2.3086 ± 0.2897	28.2437

To investigate the effect of adding non-negative constraints on the accuracy of the regression function, we compare the regression coefficients W and b obtained by NQSSVR(p) with those obtained using the other three methods. Since NNSVR is a linear model, it has only linear term coefficients b and does not involve nonlinear term coefficients W . The regression coefficients obtained by the four methods are small, and for comparison purposes, we enlarge the W and b by a factor of 100 and 10, respectively, before drawing

the figure. When the regression coefficient is negative, the color of the color block is closer to blue.

We want to obtain a regression function that is monotonically increasing, by Theorem 3 it is equivalent to the non-negative constraints with respect to regression coefficients. From the Figures 6 and 7, we can see that ϵ -SQSSVR and QLSSVR obtained W and b contain negative numbers, so the regression functions obtained do not match the a priori information. However, our model yields regression coefficients that all match the a priori information. Therefore, adding non-negative constraints can improve the accuracy and reasonableness of the model. Since our method can obtain a quadratic regression function, it is therefore more accurate than the linear regression function obtained by NNSVR.

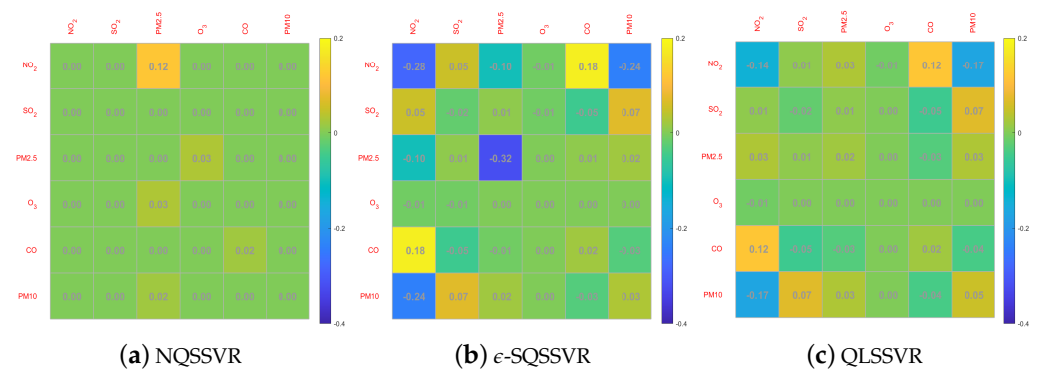


Figure 6. W -matrices.

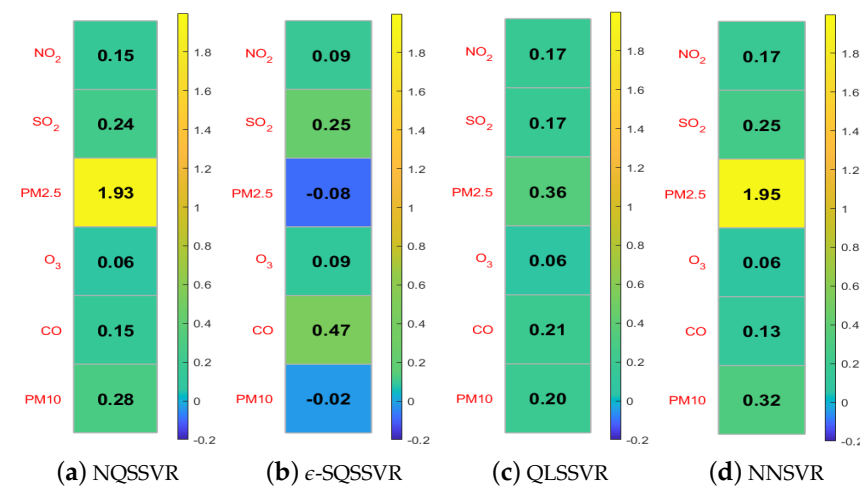


Figure 7. b -vectors.

Figure 8 show the effect of the parameters on the performance of our model. Here we have chosen three evaluation criteria, including RMSE, MAE, and R^2 . Figure 8a–c displays the experimental results obtained for the dual problem, while Figure 8d–f presents the experimental results for the primal problem. It can be observed that the effect of the parameters on our proposed model remains significant. It is worth noting that our model receives ϵ -insensitive parameter influence more and is not very sensitive to the change of penalty parameter C .

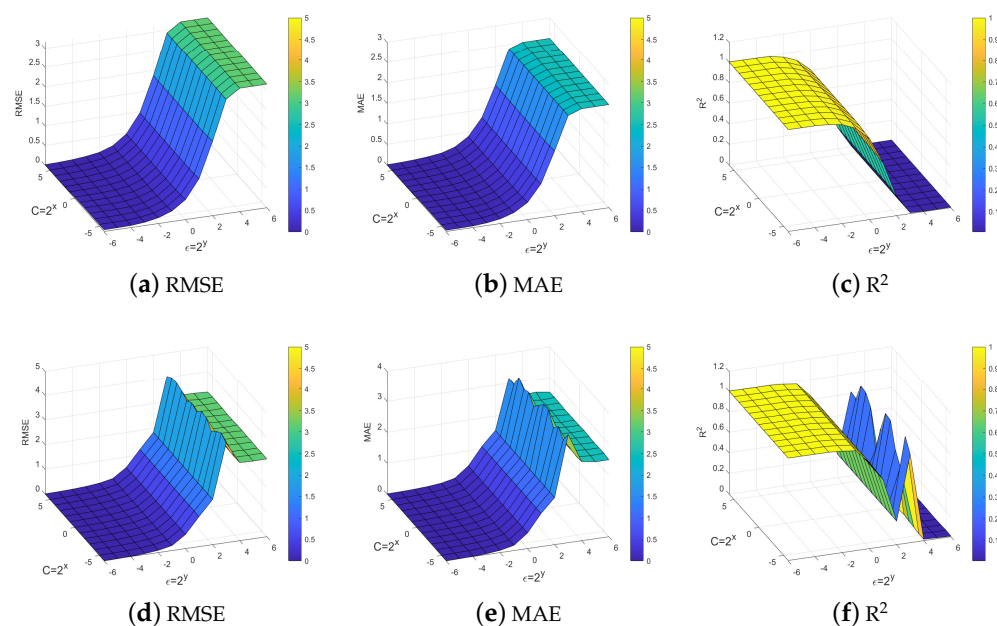


Figure 8. Effect of parameters on method performance NQSSVR.

5. Conclusions

For the regression problem, a novel kernel-free quadratic surface support vector regression with non-negative constraints (NQSSVR) is proposed by utilizing the kernel-free technique and introducing the non-negative constraints with respect to regression coefficients. Specifically, by using a quadratic surface to fit the data, the regression function is nonlinear and does not involve kernel functions, so the model is unnecessary to select kernel functions and corresponding parameters, and the obtained regression function has better interpretability. Moreover, adding non-negative constraints with respect to regression coefficients to the model ensures that the obtained regression function conforms to the monotonic non-decreasing characteristics on non-negative interval. In fact, when exploring air quality examples, there is a prior information that air quality indicators will increase with the increase in all gas concentrations in the atmosphere. Fortunately, we have proven that the quadratic regression function obtained by NQSSVR is monotonically non-decreasing on the non-negative interval if and only if the non-negative constraints with respect to the regression coefficients hold true. The results of numerical experiments on the two artificial datasets, seven benchmark datasets and air quality datasets demonstrate that our method is feasible and effective.

In this paper, we impose a non-negative restriction on the regression coefficients based on prior information. In the subsequent optimization problems, we can add different restrictions to the values of the regression coefficients based on the prior information. For example, part of the regression coefficients are restricted to non-negative intervals and part of the regression coefficients are unrestricted.

Author Contributions: Methodology, D.W. and Z.Y.; Software, D.W.; Writing—original draft, D.W.; Writing—review & editing, Z.Y. and J.Y.; Supervision, Z.Y., J.Y. and X.Y.; Funding acquisition, Z.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National Natural Science Foundation of China (No. 12061071).

Data Availability Statement: All of the benchmark datasets used in our numerical experiments are from the UCI Machine Learning Repository, which are available at <https://archive.ics.uci.edu/ml/351index.php> (the above datasets accessed on 18 August 2021).

Acknowledgments: The authors would like to thank the editor and referees for their valuable comments and suggestions, which helped us to improve the results of this paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhou, L.; Harrach, B.; Seo, J.K. Monotonicity-based electrical impedance tomography for lung imaging. *Inverse Probl.* **2018**, *34*, 045005. [\[CrossRef\]](#)
2. Chatterjee, S.; Guntuboyina, A.; Sen, B. On matrix estimation under monotonicity constraints. *Bernoulli* **2018**, *24*, 1072–1100. [\[CrossRef\]](#)
3. Wang, J.; Qian, Y.; Li, F.; Liang, J.; Ding, W. Fusing fuzzy monotonic Decision Trees. *IEEE Trans. Fuzzy Syst.* **2019**, *28*, 887–900. [\[CrossRef\]](#)
4. Henderson, N.C.; Varadhan, R. Damped anderson acceleration with restarts and monotonicity control for accelerating em and em-like algorithms. *J. Comput. Graph. Stat.* **2019**, *28*, 834–846. [\[CrossRef\]](#)
5. Bro, R.; Sidiropoulos, N.D. Least squares algorithms under unimodality and non-negativity constraints. *J. Chemom. J. Chemom. Soc.* **1998**, *12*, 223–247. [\[CrossRef\]](#)
6. Luo, X.; Zhou, M.C.; Li, S.; Hu, L.; Shang, M. Non-negativity constrained missing data estimation for high-dimensional and sparse matrices from industrial applications. *IEEE Trans. Cybern.* **2019**, *50*, 1844–1855. [\[CrossRef\]](#)
7. Theodosiadou, O.; Tsaklidis, G. State space modeling with non-negativity constraints using quadratic forms. *Mathematics* **2019**, *9*, 1905. [\[CrossRef\]](#)
8. Haase, V.; Hahn, K.; Schnidube, H.; Stierstorfer, K.; Maier, A.; Noo, F. Impact of the non-negativity constraint in model-based iterative reconstruction from CT data. *Med. Phys.* **2019**, *46*, 835–854. [\[CrossRef\]](#) [\[PubMed\]](#)
9. Yamashita, S.; Yagi, Y.; Okuwaki, R.; Shimizu, K.; Agata, R.; Fukahata, Y. Potency density tensor inversion of complex body waveforms with time-adaptive smoothing constraint. *Geophys. J. Int.* **2022**, *231*, 91–107. [\[CrossRef\]](#)
10. Wang, J.; Deng, Y.; Wang, R.; Ma, P.; Lin, H. A small-baseline InSAR inversion algorithm combining a smoothing constraint and L_1 -norm minimization. *IEEE Geosci. Remote Sens. Lett.* **2019**, *16*, 1061–1065. [\[CrossRef\]](#)
11. Mammen, E.; Marron, J.S.; Turlach, B.A.; Wand, M.P. A general projection framework for constraints smoothing. *Stat. Sci.* **2001**, *16*, 232–248. [\[CrossRef\]](#)
12. Powell, H.; Lee, D.; Bowman, A. Estimating constraints concentration–response functions between air pollution and health. *Environmetrics* **2012**, *23*, 228–237. [\[CrossRef\]](#)
13. Lawson, C.L.; Hanson, R.J. *Solving Least Squares Problems*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 1995.
14. Chen, J.; Richard, C.; Honeine, P.; Bermudez, J.C.M. Non-negative distributed regression for data inference in wireless sensor networks. In Proceedings of the 2010 Conference Record of the Forty Fourth Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, USA, 7–10 November 2010; pp. 451–455.
15. Shekkizhar, S.; Ortega, A. Graph construction from data by non-negative kernel regression. In Proceedings of the ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 3892–3896.
16. Shekkizhar, S.; Ortega, A. NNK-Means: dictionary learning using non-negative kernel regression. *arXiv* **2021**, arXiv:2110.08212.
17. Chapel, L.; Flamary, R.; Wu, H.; F evotte, C.; Gasso, G. Unbalanced optimal transport through non-negative penalized linear regression. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 23270–23282.
18. Drucker, H.; Burges, C.J.; Kaufman, L.; Smola, A.; Vapnik, V. Support vector regression machines. *Adv. Neural Inf. Process. Syst.* **1996**, *9*, 779–784.
19. Fauzi, A. Stock price prediction using support vector machine in the second Wave of COVID-19 pandemic. *Insearch Inf. Syst. Res. J.* **2021**, *1*, 58–62. [\[CrossRef\]](#)
20. Huang, S.; Cai, N.; Pacheco, P.P.; Narrandes, S.; Wang, Y.; Xu, W. Applications of support vector machine (SVM) learning in cancer genomics. *Cancer Genom. Proteom.* **2018**, *15*, 41–51.
21. Zhong, H.; Wang, J.; Jia, H.; Mu, Y.; Lv, S. Vector field-based support vector regression for building energy consumption prediction. *Appl. Energy* **2019**, *242*, 403–414. [\[CrossRef\]](#)
22. Guo, H.; Nguyen, H.; Bui, X.N.; Armaghani, D.J. A new technique to predict fly-rock in bench blasting based on an ensemble of support vector regression and GLMNET. *Eng. Comput.* **2021**, *37*, 421–435. [\[CrossRef\]](#)
23. Dagher, I. Quadratic kernel-free non-linear support vector machine. *J. Glob. Optim.* **2008**, *41*, 15–30. [\[CrossRef\]](#)
24. Gao, Z.; Wang, Y.; Huang, M.; Luo, J.; Tang, S. A kernel-free fuzzy reduced quadratic surface ν -support vector machine with applications. *Appl. Soft Comput.* **2022**, *127*, 109390. [\[CrossRef\]](#)
25. Zhou, J.; Tian, Y.; Luo, J.; Zhai, Q. Novel non-Kernel quadratic surface support vector machines based on optimal margin distribution. *Soft Comput.* **2022**, *26*, 9215–9227. [\[CrossRef\]](#)
26. Ye, J.Y.; Yang, Z.X.; Li, Z.L. Quadratic hyper-surface kernel-free least squares support vector regression. *Intell. Data Anal.* **2021**, *25*, 265–281. [\[CrossRef\]](#)
27. Ye, J.Y.; Yang, Z.X.; Ma, M.P.; Wang, Y.L.; Yang, X.M. ϵ -Kernel-free soft quadratic surface support vector regression. *Inf. Sci.* **2022**, *594*, 177–199. [\[CrossRef\]](#)
28. Zhai, Q.; Tian, Y.; Zhou, J. Linear twin quadratic surface support vector regression. *Math. Probl. Eng.* **2020**, *2020*, 3238129. [\[CrossRef\]](#)

29. Zheng, L.J.; Tian, Y.; Luo, J.; Hong, T. A novel hybrid method based on kernel-free support vector regression for stock indices and price forecasting. *J. Oper. Res. Soc.* **2023**, *74*, 690–702. [CrossRef]
30. Dua, D.; Graff, C. UCI Machine Learning Repository. 2017. Available online: <https://archive.ics.uci.edu/ml/index.php> (accessed on 18 August 2021).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.