TẠP CHÍ KHOA HỌC & CÔNG NGHỆ ĐẠI HỌC DUY TÂN
DTU Journal of Science and Technology 01(56) (2023) 47-53

TRƯỜNG ĐẠI HỌC DUY TÂN
DUYTAN UNIVERSITY

Kernel regression models developed with Visual C# .NET for data analysis in construction engineering

Các mô hình hồi quy dựa vào hàm nhân được phát triển với ngôn ngữ Visual C# .NET và ứng dụng cho phân tích dữ liệu trong ngành xây dựng

Hoang Nhat Duc[a,b*], Nguyen Quoc Lam[b]
Hoàng Nhật Đức[a,b*], Nguyễn Quốc Lâm[b]

[a]Institute of Research and Development, Duy Tan University, Da Nang, 550000, Vietnam
[a]Viện Nghiên cứu và Phát triển Công nghệ Cao, Đại học Duy Tân, Đà Nẵng, Việt Nam
[b]Faculty of Civil Engineering, Duy Tan University, Da Nang, 550000, Vietnam
[b]Khoa Xây dựng, Trường Đại học Duy Tân, Đà Nẵng, Việt Nam

(Ngày nhận bài: 15/10/2022, ngày phản biện xong: 24/10/2022, ngày chấp nhận đăng: 20/12/2022)

Abstract

This research work relies on kernel regression methods for constructing nonlinear regression models. These models can be used to solve function approximation tasks in construction engineering. The newly developed software program was developed with the Visual C# .NET. The program has been tested with the task of estimating the punching shear strength of steel fibre reinforced concrete slab.

Keywords: Construction engineering; Kernel function; Polynomial kernel; Radial basis function kernel; Regression analysis.

Tóm tắt

Nghiên cứu của chúng tôi sử dụng phương pháp hồi quy dựa trên các hàm nhân để xây dựng các mô hình phân tích dữ liệu. Chương trình phần mềm dựa trên các phương pháp hồi quy này đã được xây dựng với ngôn ngữ Visual C# và nền tảng .NET. Chương trình tính toán đã được kiểm chứng qua việc ước tính khả năng chịu chọc thủng của sàn bê tông được gia cường bằng sợi thép.

Từ khóa: Kỹ thuật xây dựng; Phương pháp hàm nhân; Hàm nhân bậc cao; Hàm nhân Gaussian; Phân tích hồi quy.

1. Introduction

Regression analysis refers to the statistical method used for studying the relationship between a response variable (also called a dependent variable) and one or more independent variables (often called predictor or explanatory variables) [1]. The most common and straight-forward form of regression analysis is linear regression, in which a line or a hyper-plane is used to fit the collected data. However, linear models generally have limiting capability in dealing with data generated from

[*] *Corresponding Author:* Hoang Nhat Duc, Institute of Research and Development, Duy Tan University, Da Nang, 550000, Vietnam; Faculty of Civil Engineering, Duy Tan University, Da Nang, 550000, Vietnam.
Email: hoangnhatduc@duytan.edu.vn

complex engineering processes. In civil engineering, regression analysis often involves the study of complex relationships between a response variable and multiple explanatory variables. Such relationships are often very sophisticated and highly nonlinear. Hence, advanced nonlinear regression methods, such as neural network [2-4], piece-wise linear regression [5, 6], and kernel regression models [7, 8], are often employed to fit the collected data.

In this study, we focus on the application of the kernel regression models for analyzing data in civil engineering. A kernel regression model (KRM) is an approach used for extending a linear regression model into nonlinear regression via the used of nonlinear transformation of the original explanatory variables [9]. This nonlinear transformation is applied to the independent variables before the linear regression is performed. The training process of a kernel regression model involves dealing with a square matrix. This fact enables this model to effectively handle a large number of data instances with low computational cost. Previous works show that the KRM can be faster than the popular support vector machine provided by Scikit-learn package [9]. In addition, one notable advantage of the KRM is that its training phase involves solving a convex optimization problem instead of complex loss landscapes of neural networks. The KRM also offers a certain degree of interpretability of the constructed model because a prediction on a novel data sample is essentially a weighted linear combination of the responses of the training samples [9]. This fact facilitates the model interpretation in the sense that we can inspect which training samples are the most influential with respect to a new inquiry.

With such motivations, this paper constructs and verifies a computer program based on the KRM. This program was developed and compiled with the Visual C# .NET to facilitate its practical applications. The newly developed program was tested with the task of modeling the punching shear strength of steel fibre reinforced concrete slab.

2. The kernel regression approach

As stated earlier, to extend a linear regression model into a nonlinear regression one, we can apply a nonlinear transformation $\psi()$ to the original explanatory variables before performing data fitting process. The loss function used for constructing the regression model can be expressed as follows [9]:

$$L(w) = \frac{1}{2}\sum_{i=1}^{n}(y^{(i)} - \langle w, \psi(x^{(i)})\rangle)^2 \qquad (1)$$

where ψ denotes a nonlinear feature map. $\langle w, \psi(x^{(i)})\rangle$ denotes an inner product. w is the parameter of the linear regression model. $y^{(i)}$ is the actual value of the response variable. n is the number of training samples.

Substituting $w = \sum_{i=1}^{n}\beta_i\psi(x^{(i)})$, the above equation is equivalent to the following one:

$$L(w) = \frac{1}{2}\sum_{i=1}^{n}(y^{(i)} - \langle\sum_{j=1}^{n}\beta_i\psi(x^{(j)}), \psi(x^{(i)})\rangle)^2 \qquad (2)$$

$$L(w) = \frac{1}{2}\sum_{i=1}^{n}(y^{(i)} - [\beta_1, \beta_2, ..., \beta_n]\begin{bmatrix}\langle\psi(x^{(1)}), \psi(x^{(i)})\rangle \\ \langle\psi(x^{(2)}), \psi(x^{(i)})\rangle \\ ... \\ \langle\psi(x^{(n)}), \psi(x^{(i)})\rangle\end{bmatrix})^2$$

(3)

Therefore, the problem of minimizing $L(w)$ over all w is converted into the task of minimizing the loss function L over all β. In addition, it is only required to know the inner product $\langle\psi(x^{(1)}), \psi(x^{(i)})\rangle$ to construct the regression model. This inner product is denoted as a kernel function $K(x^{(i)}, x^{(j)}) = \langle\psi(x^{(i)}), \psi(x^{(j)})\rangle$. Hence, via a suitable selection of the feature transformation, it is able to fit a nonlinear dataset using a linear regression model. The feature transformation is governed by kernel functions [9]. In this study, we utilize the two commonly used kernel functions: the polynomial kernel function (PKF) and the radial basis kernel function (RBKF). The model construction phases of the KRM using the two kernel functions, coded in Visual C# .NET, are demonstrated in **Fig. 2.1** and **Fig. 2.2**.

To summarize, the model construction phase of a KRM is executed in the following steps [7]:

(i) Construct the standard kernel matrix: $K \leftarrow \{K(x_i, x_j)\}_{i,j=1,2,...,n}$

(ii) Construct the augmented kernel matrix: $\tilde{K} \leftarrow K + 1$

(iii) Compute the mixture coefficient (a model's parameters): $\beta \leftarrow (\tilde{K} + \alpha I)^{-1} Y$

For a testing sample *z*, its corresponding response variable is computed as follows [7]:

$$\tilde{K}_z \leftarrow \{1 + K(z, x_j)\}_{i=1,2,...,n} \tag{4}$$

$$y \leftarrow \beta^T K_z \tag{5}$$

For the degree-*d* polynomials, the PKF is defined in the following equation:

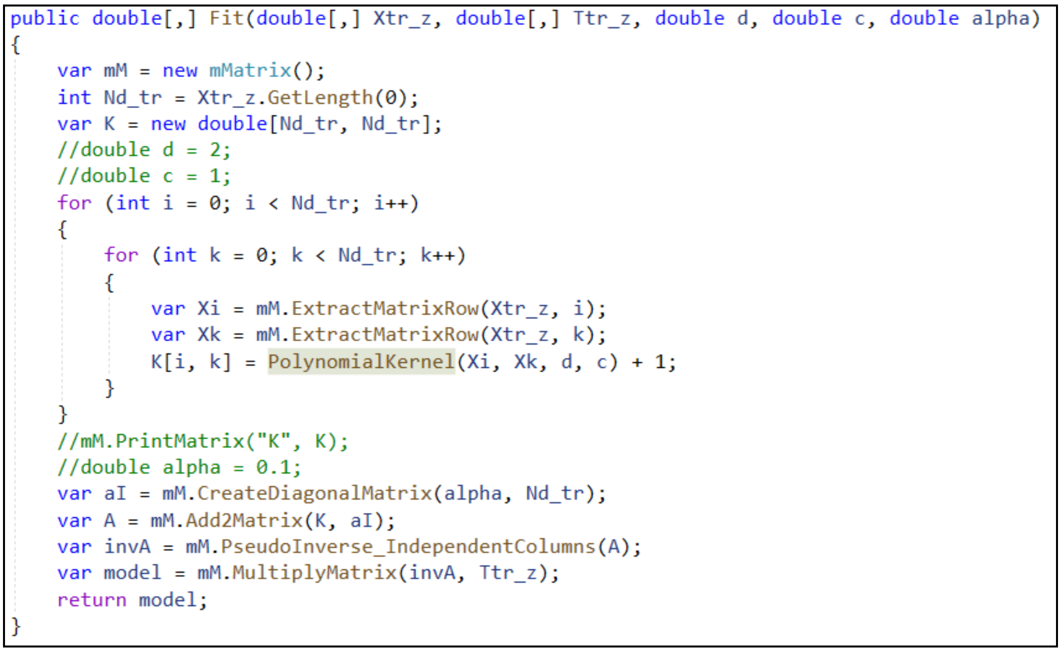
$$K(x, y) = (x^T y + c)^d \tag{6}$$

where *c* and *d* are the two hyper-parameters of this kernel function.

The RBFK is defined as follows:

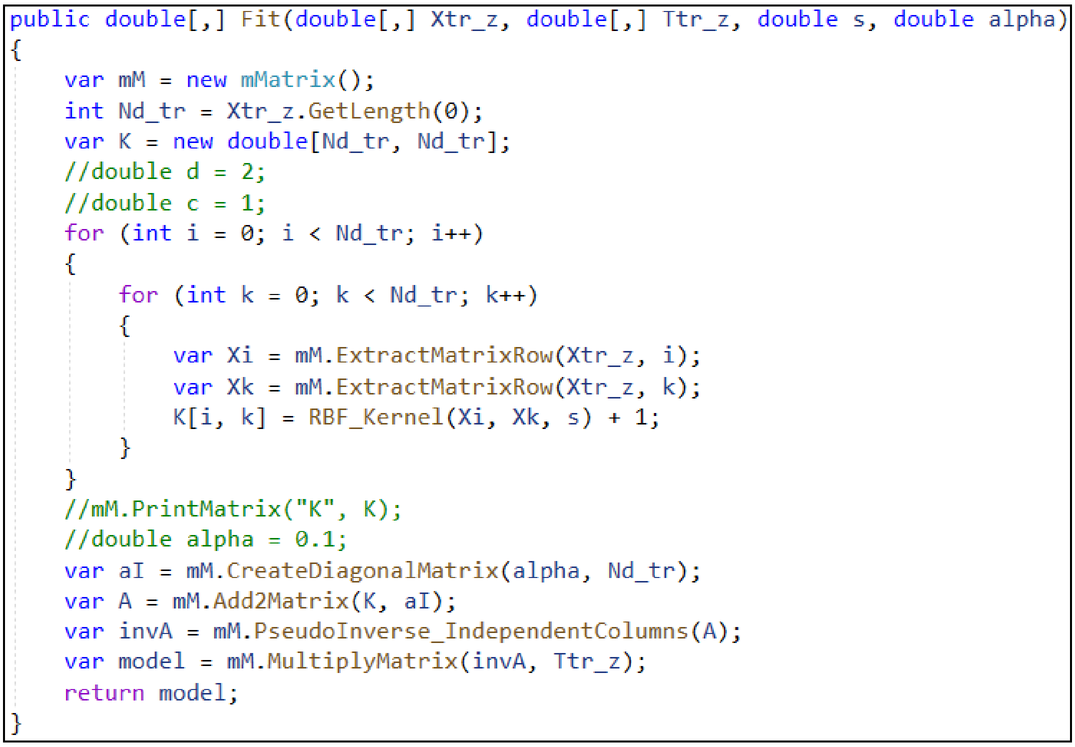
$$K(x, y) = \exp(-\frac{\| x - y \|}{2\sigma^2}) \tag{7}$$

where σ is the hyper-parameter of this kernel function..



```
public double[,] Fit(double[,] Xtr_z, double[,] Ttr_z, double d, double c, double alpha)
{
    var mM = new mMatrix();
    int Nd_tr = Xtr_z.GetLength(0);
    var K = new double[Nd_tr, Nd_tr];
    //double d = 2;
    //double c = 1;
    for (int i = 0; i < Nd_tr; i++)
    {
        for (int k = 0; k < Nd_tr; k++)
        {
            var Xi = mM.ExtractMatrixRow(Xtr_z, i);
            var Xk = mM.ExtractMatrixRow(Xtr_z, k);
            K[i, k] = PolynomialKernel(Xi, Xk, d, c) + 1;
        }
    }
    //mM.PrintMatrix("K", K);
    //double alpha = 0.1;
    var aI = mM.CreateDiagonalMatrix(alpha, Nd_tr);
    var A = mM.Add2Matrix(K, aI);
    var invA = mM.PseudoInverse_IndependentColumns(A);
    var model = mM.MultiplyMatrix(invA, Ttr_z);
    return model;
}
```

Fig. 2.1. The function used to fit the PKRM



```
public double[,] Fit(double[,] Xtr_z, double[,] Ttr_z, double s, double alpha)
{
    var mM = new mMatrix();
    int Nd_tr = Xtr_z.GetLength(0);
    var K = new double[Nd_tr, Nd_tr];
    //double d = 2;
    //double c = 1;
    for (int i = 0; i < Nd_tr; i++)
    {
        for (int k = 0; k < Nd_tr; k++)
        {
            var Xi = mM.ExtractMatrixRow(Xtr_z, i);
            var Xk = mM.ExtractMatrixRow(Xtr_z, k);
            K[i, k] = RBF_Kernel(Xi, Xk, s) + 1;
        }
    }
    //mM.PrintMatrix("K", K);
    //double alpha = 0.1;
    var aI = mM.CreateDiagonalMatrix(alpha, Nd_tr);
    var A = mM.Add2Matrix(K, aI);
    var invA = mM.PseudoInverse_IndependentColumns(A);
    var model = mM.MultiplyMatrix(invA, Ttr_z);
    return model;
}
```

Fig. 2.2. The function used to fit the RBFKRM

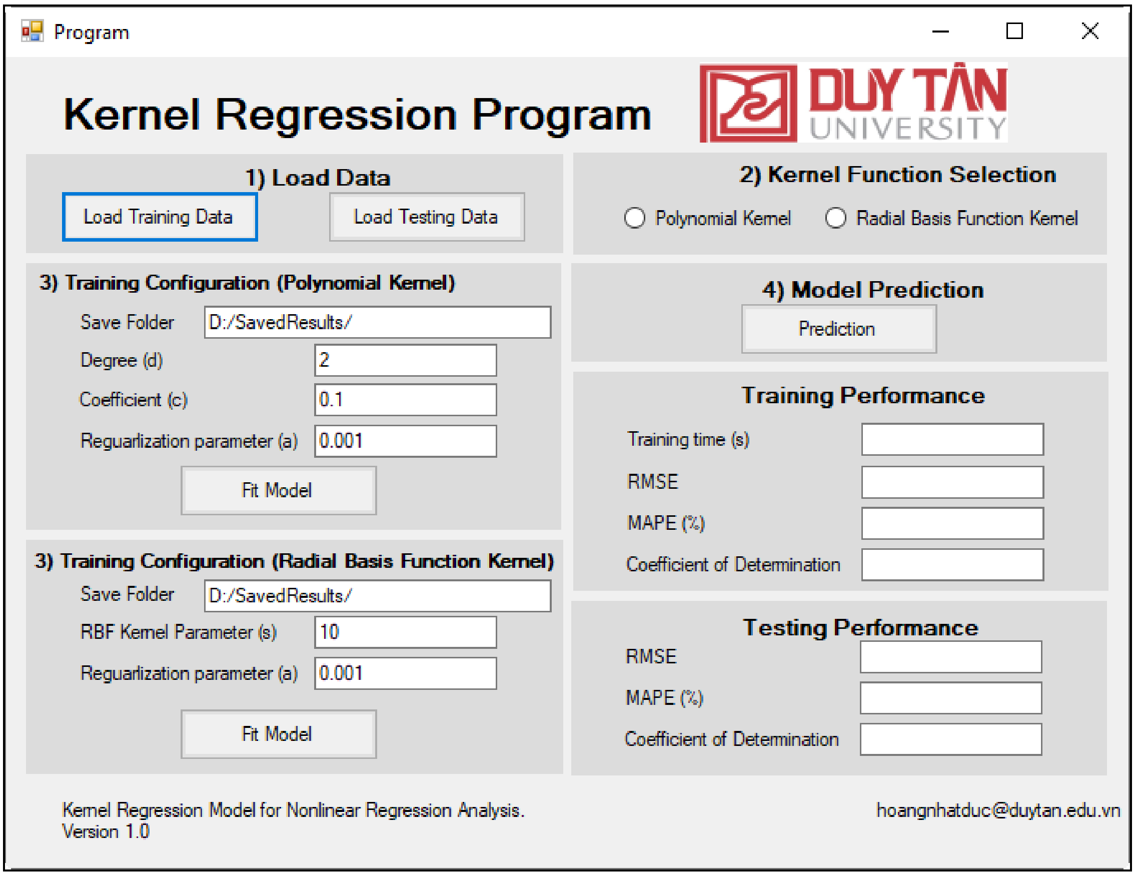
3. Program application

In this section, the performance of the KRM, developed with Visual C# .NET, is verified by a regression analysis task. The graphical user interface of the program is illustrated in Fig. 3.1. The user needs to prepare two .csv files that store the training and testing datasets. It is noted that in each .csv file, the last column of the dataset is the response variable. The other columns store the explanatory variables. The user can also select the kernel functions (either the PKF or the RBFK). Herein, the KRM is used to model punching shear strength of steel fibre reinforced concrete slab. A dataset, collected from [10], is employed to train and test the KRM. This dataset, consisting of 140 samples, is randomly divided into a training set (90%) and a testing set (10%). It is noted that the explanatory variables includes slab depth, effective depth of the slab, length or radius of the loading pad or column, compressive strength of concrete, the reinforcement ratio, and the fibre volume. To negate the effect of randomness in data sampling process, the processes of model training and testing were performed 30 times. In addition, the conventional Multiple Linear Regression (MLR) model was used as a benchmark approach. It is also noted that the KRMs and the MLR use Z-score equation to normalize the original dataset. Additionally, five-fold cross validation processes are used to determine the hyper-parameters of the KRMs.

Table 1. Experimental results

Metrics	PKRM		RBFKRM		MLR	
	Mean	Std	Mean	Std	Mean	Std
RMSE	36.47	14.78	29.72	7.06	43.05	8.76
MAPE	11.40	2.59	11.27	4.06	18.14	4.73
R^2	0.83	0.15	0.89	0.05	0.79	0.10

The experimental results are reported in Table 3. Herein, PKRM, RBFKRM, and MLR stand for Polynomial Kernel Regression Model, Radial Basis Function Kernel Regression Model, and Multiple Linear Regression Model, respectively. Std denotes the standard deviation of the results obtained from 30 independent runs. The Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and coefficient of determination (R^2) are used to measure the models' performance. As can be observed from this table, the RBFKRM has attained the best performance with RMSE = 29.72, MAPE = 11.27%, and R^2 = 0.89. The result of the RBFKRM is better than that of the PKRM, which obtained RMSE = 36.47, MAPE = 11.40%, and R^2 = 0.83. The two KRMs are also better than the MLR (RMSE = 43.05, MAPE = 18.14%, and R^2 = 0.79). Hence, the RBFKRM obtained a roughly 31% improvement in comparison with the conventional linear regression model. Moreover, using RBFKRM, the percentage of variance of the response variable explained by the regression model is enhanced by 10%. The prediction performances of the two KRMs and the MLR are further depicted in Fig. 3.2, 3.3, and 3.4.


Program
Kernel Regression Program
DUY TÂN UNIVERSITY
1) Load Data
Load Training Data
Load Testing Data
2) Kernel Function Selection
○ Polynomial Kernel
○ Radial Basis Function Kernel
3) Training Configuration (Polynomial Kernel)
Fit Model
3) Training Configuration (Radial Basis Function Kernel)
Fit Model
4) Model Prediction
Prediction
Training Performance
Training time (s)
RMSE
MAPE (%)
Coefficient of Determination
Testing Performance
RMSE
MAPE (%)
Coefficient of Determination

Save Folder D:/SavedResults/
Degree (d) 2
Coefficient (c) 0.1
Reguarlization parameter (a) 0.001

Save Folder D:/SavedResults/
RBF Kernel Parameter (s) 10
Reguarlization parameter (a) 0.001

Kernel Regression Model for Nonlinear Regression Analysis.
Version 1.0

hoangnhatduc@duytan.edu.vn

Fig. 3.1. Graphical user interface of the developed kernel regression program

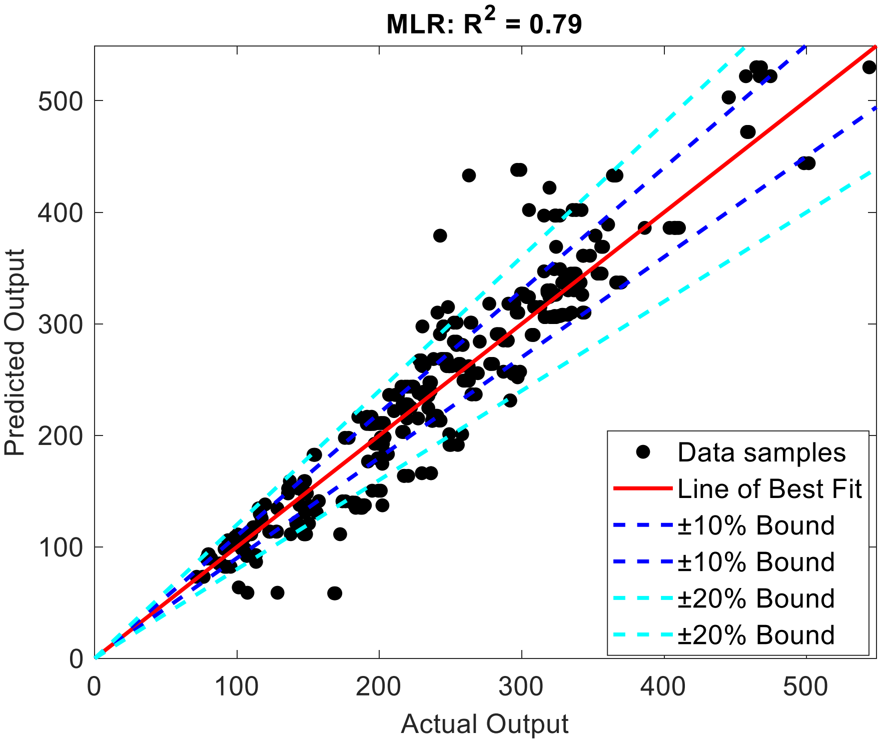


Fig. 3.2. Prediction performance of the M LR

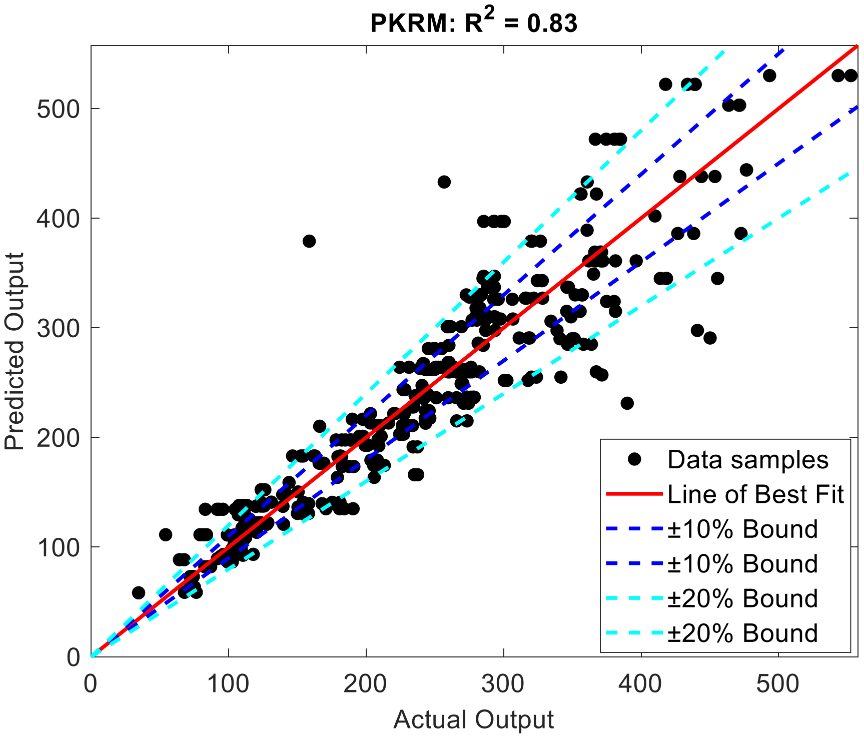


Fig. 3.3. Prediction performance of the PKRM

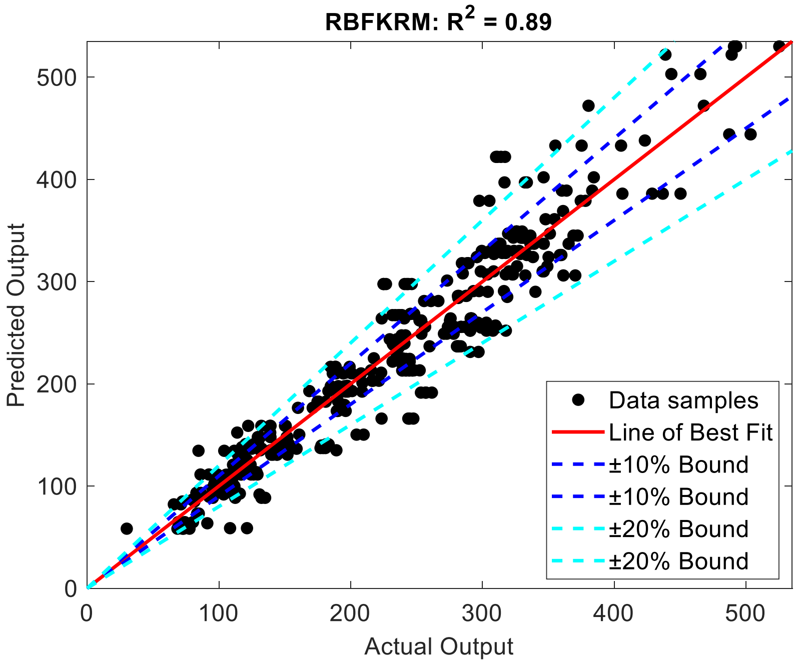


Fig. 3.4. Prediction performance of the RBFKRM

4. Conclusion

This paper has constructed and verified a KRM based on the PKF and RBFK for performing nonlinear regression analysis. The new approach was developed in Visual C# .NET and tested with the task of modeling the punching shear strength of steel fibre reinforced concrete slab. It is expected that this computer program based on the KRM can be a useful tool to assist construction engineers in various data modeling tasks.

Supplementary material

The developed program has been deposited at: https://github.com/NhatDucHoang/Krm_V1.0.

References

[1] W. Mendenhall, T.T. Sincich (2011), A Second Course in Statistics: Regression Analysis (7th Edition), Pearson, ISSN 978-0321691699.

[2] M.-T. Cao, N.-M. Nguyen, W.-C. Wang (2022), Using an evolutionary heterogeneous ensemble of artificial neural network and multivariate adaptive regression splines to predict bearing capacity in axial piles, Engineering Structures, 268 114769, https://doi.org/10.1016/j.engstruct.2022.114769.

[3] Z. Yan, Z. Sun, Y. Zhao, Y. Ji, J. Tian (2022), Prediction of compressive strength of mortar with copper slag addition by using artificial neural network, Structural Concrete, n/a https://doi.org/10.1002/suco.202100204.

[4] T.-H. Tran, N.-D. Hoang (2016), Predicting Colonization Growth of Algae on Mortar Surface with Artificial Neural Network, Journal of Computing in Civil Engineering, 30 04016030, doi:10.1061/(ASCE)CP.1943-5487.0000599.

[5] X.L. Tran, N.D. Hoang (2020), A sequential piecewise linear regression model for data analysis developed with Visual C# .NET, DTU Journal of Science and Technology, 05 1-6.

[6] N.-D. Hoang (2019), Estimating Punching Shear Capacity of Steel Fibre Reinforced Concrete Slabs Using Sequential Piecewise Multiple Linear Regression and Artificial Neural Network, Measurement, 137 58-70, https://doi.org/10.1016/j.measurement.2019.01.035.

[7] A.C. Faul (2019), A Concise Introduction to Machine Learning, Chapman & Hall/CRC Machine Learning & Pattern Recognition, Chapman and Hall/CRC, ISBN-10 : 0815384106.

[8] D.-T. Vu, N.-D. Hoang (2016), Punching shear capacity estimation of FRP-reinforced concrete slabs using a hybrid machine learning approach, Structure and Infrastructure Engineering, 12 1153-1161, 10.1080/15732479.2015.1086386.

[9] A. Radhakrishnan (2022), Lecture 3: Kernel Regression, MLClassLecture3, Edited by: Max Ruiz Luyten, George Stefanakis, Cathy Cai, <https://web.mit.edu › lectures >.

[10] L.F. Maya, M. Fernández Ruiz, A. Muttoni, S.J. Foster (2012), Punching shear strength of steel fibre reinforced concrete slabs, Engineering Structures, 40 83-94, https://doi.org/10.1016/j.engstruct.2012.02.009.