Lecturer Abdullah Mohammed RASHID, PhD
E-mail: abdullah.mohamed@duc.edu.iq
Institute for Mathematical Research
Universiti Putra Malaysia, Malaysia
Department of Business Administration
Dijlah University College, Iraq

Professor Habshah MIDI, PhD (corresponding author)
E-mail: habshah@upm.edu.my
Department of Mathematics and Statistics
Faculty of Science and Institute for Mathematical Research
Universiti Putra Malaysia, Malaysia

IMPROVED *nu*-SUPPORT VECTOR REGRESSION ALGORITHM BASED ON THE PRINCIPAL COMPONENT ANALYSIS

Abstract. *Principal component analysis (PCA) is the most commonly used approach for analysing high-dimensional data in order to achieve dimension reduction. However, outliers have an adverse effect on the PCA, and hence reduce the accuracy of the prediction model. To date, no research has been done to incorporate the PCA into the algorithm of support vector regression (SVR) technique in order to obtain an accurate prediction model with high accuracy. This paper focuses on improving the nu-SVR algorithm to handle the problem of outliers. A new hybrid PCA with the nu-SVR technique (PCA-SVR) has been established. The performance of the proposed PCA-SVR algorithm is extensively assessed by two real data sets and simulation studies. The outcomes indicate that the PCA-SVR algorithm is more efficient and reliable than the nu-SVR.*

Keywords*: Principal component analysis, Support vector regression,*
High-dimensional data, Dimension reduction, Outliers

JEL Classification: C14, C15, C53, C80

1. Introduction

The analysis of high-dimensional data (HDD) in which the number of independent variables is much larger than the sample size ($p >> n$) has become increasingly important in many fields, such as in economics, engineering, and medicine. For instance, there are tens of thousands of gene expression values available in tumour classification utilising genomic data. However, the number of arrays is only on the order of 10. High-dimensional data poses a significant statistical challenge in data classification and other statistical analysis. A matrix associated with some algorithms in high-dimensional data may become singular. The problem

DOI: 10.24818/18423264/57.2.23.03

may become more complicated with non-linear relationships among variables and outliers in the data. In real applications, encountering non-normal data with a non-linear relationship between input and output variables is quite common (Ukil, 2007). According to Hampel et al. (1986), a typical data set generally includes between 1 - 10% outliers, and even the best quality data set cannot be confirmed to be absence of outliers. Outliers may generate apparent non-normality, and the entire classical inferential technique may fall as a result of their presence. In this case, it is imperative to call for new methods and theories, such as non-parametric methods, to take care of these concerns.

Support Vector Machine (SVM) is a non-parametric technique that consists of a new learning algorithm class (Vapnik, 1999) that adheres to statistical learning theory. It has been used to solve classification and regression problems with great success. Additionally, Support Vector Regression (SVR) has the advantage of addressing rank-deficient, non-linear, and high-dimensional issues by applying the kernel method to turn non-linear relationships in the input space into linear relationships in high-dimensional feature space (Lahiri & Ghanta, 2009; Üstün et al. 2006). The main idea behind SVM modelling is its ability to classify and separate positive and negative training data with the greatest margin possible (Balfer & Bajorath, 2015). Although the SVR algorithm has proven to be efficient in dealing with high-dimensional data (see (Mohammed Rashid et al. 2022; Rashid et al. 2021), it is unable to achieve dimensionality reduction simultaneously.

Nowadays, real data sets usually contain several dimensions or high dimensions, due to the huge flow of data resulting from technological progress. To handle the high- dimensional data, this dimensionality needs to be shrunk or reduced. The process of transforming highly dimensional data into a useful representation of decreased dimensionality is known as dimensionality reduction. The dimensionality of the reduced representation should ideally match the intrinsic dimensionality of the data. The bare minimum of parameters required to explain the observable qualities of the data is known as the intrinsic dimension of the data (Fukunaga, 2013). Dimensionality reduction is crucial in many fields because it reduces the negative effects of dimensionality and other high-dimensional space characteristics (Jimenez & Landgrebe, 1998). As a result, dimensionality reduction makes it easier to categorise, visualise, and compress high-dimensional data.

The principal component analysis (PCA) is a very common statistical multivariate technique that uses a limited number of components to try to explain the covariance structure of data (Hubert et al., 2005). Those components are linear collections of the main variables and are frequently used to analyse and better comprehend the many sources of variation. The PCA is commonly used for the analysis of high-dimensional data, because it is concerned with dimension reduction, which can be applied in many fields such as zip code classification, human face recognition, engineering, genetics, etc. see (Alter et al., 2000; Hastie et al., 2009; Zou and Xue, 2018). Most of PCA studies are focused on its applications in feature extraction and dimensional reduction. The majority of PCA applications involve transforming samples into a new space and then denoting the sample using a lower-dimensional representation of the new space. Due to the efficiency of the PCA method in many fields of application, it has been used in the development of many new statistical models, for example, Virmani et al. (2016) improved classification approaches of PCA-SVM and PCA with probabilistic neural network (PCA-PNN) based on computer-aided diagnostic (CAD) systems for the classification of breast tissue density. Moreover, Ye & Peng (2018) have developed an image processing model by combining the PCA method with the convolution neural network (CNN) to solve the problem of low recognition rate. Gan et al. (2012) improved the PCA method with linear discriminant analysis (LDA) that can be used to solve pattern recognition and classification methods. There are many combined methods with PCA in the literature, which have inspired us to propose a hybrid prediction model by combining the PCA with the newly developed *nu*-SVR method of Mohammed Rashid et al. (2022) to achieve dimensional reduction and high prediction accuracy at the same time. This hybrid model is denoted as PCA-SVR.

The paper is organised as follows: The structure of the proposed method PCA-SVR is described in Section 2. In Sections (3 and 4) the performance of our proposed method is tested by using simulation studies and two real datasets. This paper is concluded with a discussion in Section 5.

2. The Improved *nu*-Support Vector Regression Algorithm Based on Principal Component Analysis (PCA-SVR)

This section summarises the proposed hybrid prediction model. The hybrid prediction model includes two main steps: first, we applied the PCA method to the high-dimensional data to reduce the dimensionality by extracting the principal components. Second, the *nu*-SVR model is used to produce a prediction model. The next sections provide an overview of the fundamentals of the PCA and *nu*-SVR methods.

Hotelling (1933) was the first person to establish Principal Component Analysis (PCA), which is an unsupervised dimension reduction procedure. The PCA analysis aims to find a small number of linear combinations of the predictors that can be used to summarise data without losing too much information. This statistical method transforms a broad set of correlated variables into a smaller number of uncorrelated factors, called principal components. These orthogonal principal components solve the multicollinearity problem.

From a dimension reduction standpoint, the PCA approach can be characterised as a series of orthogonal linear transformations of the original variables that preserve as much information as possible in the transformed variables (Zou and Xue, 2018). For example, let X be a matrix of data $n \times p$, where n represents the sample size and p is the predictor, respectively. For ease of explanation, let us assume that the column means of X are all 0. The first component is described as follows:

$$Z_1 = \sum_{j=1}^{p} a_{1j} X_j, \tag{1}$$

in which $a_1 = (a_{11}, \dots, a_{1p})^T$ is selected in order to maximise Z_1variance.

$$a_1 = \underset{a}{argmax}\ a^T\ \hat{\Sigma} a$$

$$Subject\ to\ \|a_1\| = 1.$$

Where $\hat{\Sigma} = (X^T X)/n$ and the remaining main components can be defined in the following order:

$$a_{k+1} = \underset{a}{argmax}\ a^T\ \hat{\Sigma} a \tag{2}$$

$$Subject\ to\ \|a_1\| = 1\ \ and\ \ a^T a_1 = 0, \forall 1 \leq l \leq k.$$

According to this definition, the first K loading vectors are the first K eigenvectors of $\hat{\Sigma}$. Components in the PCA approach can be obtained from the singular value decomposition (SVD) of the X matrix as follows:

$$X = UDV^T, \tag{3}$$

where $U_{n\times p}$ and $V_{p\times p}$ are the orthonormal matrices, and D is a diagonal matrix with diagonal elements $d_1, \dots, d_p$ in a descending order. The eigenvectors of $\hat{\Sigma}$ are the columns of V, and V is the loading matrix of the principal components. We can deduce from $XV = UD$ that $Z_k = U_k\ d_k$, where U_k is the k^{th} column of U. The best low-rank approximation to the data matrix can be read as SVD.

In this paper, the *nu*-SVR approach of Mohammed Rashid et al. (2022) is used to establish the prediction model, by adding the most components that interpret the variability of the dataset, which can be obtained from $X = UDV^T = X^*$, where $X^*$is represented the selected components. Hence, the regression function is expressed as follows:

$$f(X^*) = w, \Phi(X^*) + b, \tag{4}$$

in which $\Phi(X^*)$ is a transform function from non-linear to linear dimensional space, w is the weight vector, and b is the bias term. In order to improve the ability to generalise (prediction), the parameters w and b has to be estimated by using the *nu*-tube loss function as follows:

$$L_{nu} = \begin{cases} 0 & ; \quad |y_i - f(X^*)| \leq nu \\ |y_i - f(X^*)| - nu & ; \quad otherwise. \end{cases} \tag{5}$$

The *nu* parameter adjusts (minimises ε) the flexible tube automatically to keep track of the support vector's numbers and training errors in the tube. The parameter *nu* is an upper bound on the proportion of margin errors and a lower bound on the fraction of support vectors, as mentioned by Schölkopf et al. (1999). Thus, the problem in Equation (4) could be presented as a convex optimisation problem as follows:

$$minimize\ \frac{1}{2}\|w\|^2 + C\left(nu + \frac{1}{l}\sum_{i=1}^{l}(\xi_i + \xi_i^*)\right)$$

$$subject\ to \begin{cases} f(X^*) - y_i & ; \leq\ nu + \xi_i \\ y_i - f(X^*) & ;\ \leq nu + \xi_i^* \\ \xi_i, \xi_i^*\ \geq\ 0\ ,\ nu \geq\ 0. \end{cases} \tag{6}$$

where C is the penalisation parameter and to understand that *i* = 1, ..., *l*, in which *l* refers to *l*-dimensional vectors of the corresponding variables. Hence, the final regression function for the proposed hybrid prediction model PCA-SVR can be obtained as follows:

$$f(X^*) = \sum_{i=1}^{l}(\alpha_i^* - \alpha_i)k\left(x_j^*, x^*\right) + b. \tag{7}$$

The kernel function $k\left(x_j^*, x^*\right)$ is used to overcome non-linear relationships among variables in the input space. Moreover, it is very important to choose the suitable kernel function in the prediction model. Therefore, the Bessel kernel function is used in the PCA-SVR method because it has already been proven to have high efficiency and good performance; see (Mohammed Rashid et al., 2022).

The algorithm of hybrid prediction model PCA-SVR can be summarised as the following steps:

Step 1: The X and Y variables need to be standardised at the outset to ensure that they are given the same weights. According to Narasimhan and Shah (2008), the data set must be scaled before applying the PCA approach. Hence, the standardisation can be done by the following equations:

$$x_{ij}^* = \frac{x_{ij} - \bar{x}_j}{s_x} \quad and \quad y_i^* = \frac{y_i - \bar{y}}{s_y}$$

$$i = 1, \dots, n \quad and \quad j = 1, \dots, p,$$

Where $\bar{x}_j$, $\bar{y}$ and S_x , S_y are the sample mean and sample standard deviations of X and Y matrices, respectively.

Step 2: Compute the variance covariance matrix.

Step 3: Compute the eigenvectors and eigenvalues using the covariance matrix as in step 2.

Step 4: Determine and select the principal components (PCs) from the Equation (3).

Step 5: Applying the prediction model SVR as in the Equation (7) by using the selected PCs from step 4.

It should be noted that we used different sets of parameters (C, ε, and h) as in (Dhhan et al., 2017). In the next section, the performance of the proposed hybrid prediction model PCA-SVR is assessed and compared with that of *nu*-SVR. This is assessed using a simulation study and real data sets.

3. Simulations Studies

This section includes two simulation studies to compare the proposed PCA-SVR method with the *nu*-SVR method in the case of low-dimensional space ($p < n$) and high dimensional space ($p >> n$). The aim of these simulation studies is to demonstrate the effectiveness of the proposed PCA-SVR method in terms of dimension reduction, prediction accuracy, and time-consuming over the *nu*-SVR technique. It is worth noting that these simulation studies are designed by following Dhhan et al. (2018) simulation design. The proposed PCA-SVR method and *nu*-SVR are evaluated based on the prediction risk /or the mean squared error (MSE). The prediction risk (MSE) is simulated 1000 times in replications. Moreover, the datasets have been partitioned into 70% training data, which is used to build the model, and 30% testing data, which is used to test the model. The kernel function that is used in this paper is the Bessel kernel function.

3.1 Simulation 1

In the first simulation, we consider the following multivariate linear regression model:

$$y_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ij} + \epsilon_i, \tag{8}$$

where β is a set of regression coefficients $\left(\beta_1, \dots, \beta_p\right) = 1,$ and x_{ij} $(i = 1, \dots, n \; and \; j = 1, \dots, p)$ and the errors ϵ_i are generated from standard normal distribution $N(0,1)$ for $p = 20$ and $n = (50,100, and\ 200)$ which indicates that

$p < n$ (low-dimensional space). In order to see the impact of outliers, three good observations (1,2, and 3) are replaced with high values equal to 20. Moreover, this simulation study has been replicated 1000 times.

Table 1 summarises the prediction error (MSE) of the proposed method PCA-SVR and the nu-SVR method based on different sets of free parameters (C, nu, and h). Additionally, in order to achieve dimensional reduction, the PCA-SVR method selected the first two principal components, which are the most components that can explain the variability, as shown in Figure 1. Hence, all of the results show that the proposed PCA-SVR method is superior to nu-SVR for all combinations of parameter. It is interesting to observe that the *nu*-SVR method has the highest MSE when parameter $C = 1$. In contrast, the proposed PCA-SVR method provides good performance in all cases of sets of parameters and sample size, as shown in Table 1 and Figure 2. Another important factor to consider is the time consumed for each method. Table 2 shows the results for the computational running time for both methods, nu-SVR and PCA-SVR. We can observe that the proposed PCA-SVR method has less computation time than the *nu*-SVR, as shown in Figure 3.

3.2 Simulation 2

The second simulation study is similar to the previous simulation (Simulation 1), except that there is a large number of explanatory variables, which is equal to 3000, with different sample sizes *n* = $(50, 100, \text{and } 200)$ which makes it very high dimensional data $(p >> n)$ as well as different sets of free parameters $(C, nu$, and $h)$. In this simulation, we consider the same multiple linear regression model as in Equation (8). We let the regression coefficients $\beta = (\beta_1, \dots, \beta_{3000}) = 1$, x_{ij} and ϵ_i are sampled from the standard normal distribution $N(0,1)$. To see the effect of outliers in both the nu-SVR and PCA-SVR methods, we replaced three clean data points $(1, 2, and$ 3) with large values equal to 20. Moreover, the proposed PCA-SVR method chooses the first and second principal components, as seen in the Figure 4 to achieve dimensional reduction.

The results of the MSE values are shown in Table 3 and Figure 5. We can clearly observe from Figure 5 that the nu-SVR method still has very large values of prediction error (MSE) when the cost parameter $C = 1$. However, the proposed PCA-SVR method shows a very good performance in terms of having the smallest values of MSE. Moreover, Table 4 and Figure 6 summarised the time consumed in both the nu-SVR and PCA-SVR methods. Once again, the proposed PCA-SVR method has proven to be effective because it requires significantly less computation time than nu-SVR. This feature in the PCA-SVR method should be highlighted, especially in the case of high-dimensional data because most of the methods dealing with high-dimensional data are time-consuming, see Figure 6.

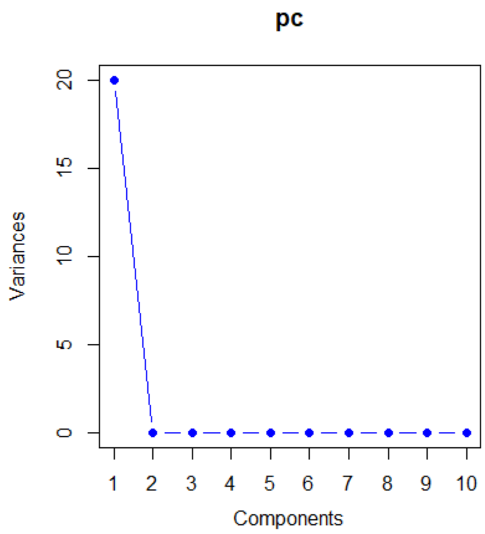

Figure 1. Principal components for PCA-SVR method when $p = 20$

Table 1. The prediction error (MSE) of nu-SVR and PCA-SVR methods when $p = 20$ and n =(50,100 and 200)

n	**Parameters**	nu**-SVR**						**PCA-SVR**					
		nu **=0.2**			nu **=0.3**			nu **=0.2**			nu **=0.3**		
		***C*=1**	***C*=50**	***C*=100**	***C*=1**	***C*=50**	***C*=100**	***C*=1**	***C*=50**	***C*=100**	***C*=1**	***C*=50**	***C*=100**
50	***h*=0.5**	21.186	1.137	1.120	21.214	1.193	1.154	**0.313**	**0.352**	**0.379**	**0.315**	**0.301**	**0.398**
	***h*=1**	21.660	1.131	1.235	21.961	1.110	1.320	**0.411**	**0.328**	**0.408**	**0.316**	**0.417**	**0.418**
100	***h*=0.5**	21.106	1.328	1.417	11.677	1.419	1.459	**0.497**	**0.451**	**0.568**	**0.499**	**0.483**	**0.575**
	***h*=1**	11.953	1.511	1.477	12.016	1.542	1.563	**0.513**	**0.512**	**0.512**	**0.413**	**0.512**	**0.513**
200	***h*=0.5**	6.854	1.905	1.907	6.879	1.915	1.944	**0.739**	**0.607**	**0.619**	**0.645**	**0.616**	**0.608**
	***h*=1**	6.990	1.908	1.928	6.996	1.903	1.908	**0.769**	**0.737**	**0.804**	**0.710**	**0.894**	**0.870**

Table 2. Comparison of the computational times of *nu*-SVR and PCA-SVR methods when $p = 20$ and $n = (50, 100\ and\ 200)$

n	Parameters	*nu*-SVR						PCA-SVR					
		nu =0.2			*nu* =0.3			*nu* =0.2			*nu* =0.3		
		C=1	C=50	C=100	C=1	C=50	C=100	C=1	C=50	C=100	C=1	C=50	C=100
50	**h=0.5**	74.4	115.2	162.6	91.8	147	234	**73.8**	**108**	**141**	**87**	**138**	**208.8**
	h=1	96.6	177.6	265.8	97.2	213	303	**93**	**151.8**	**212.4**	**92.4**	**178.2**	**258.6**
100	**h=0.5**	88.2	171.6	247.2	89.4	196.2	461.4	**84**	**164.4**	**234.6**	**83.4**	**189.6**	**444.6**
	h=1	114	364.8	540.6	121.2	372	537	**109.2**	**316.2**	**427.8**	**112.8**	**256.2**	**529.8**
200	**h=0.5**	139.8	338.4	514.2	125.4	335.4	553.2	**114**	**267**	**535.8**	**112.2**	**380.4**	**547.8**
	h=1	113.4	415.2	690	139.8	525	906	**108**	**376.2**	**610.2**	**99.6**	**457.8**	**805.2**

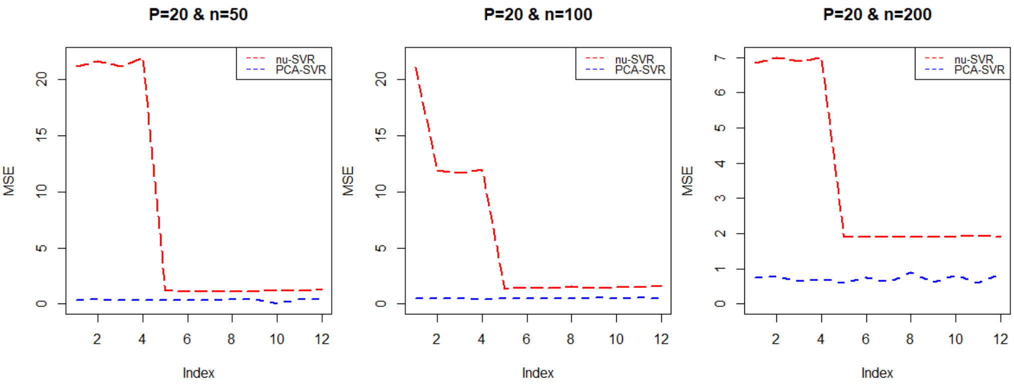

Figure 2. The MSE of *nu*-SVR and PCA-SVR for 20 predictors

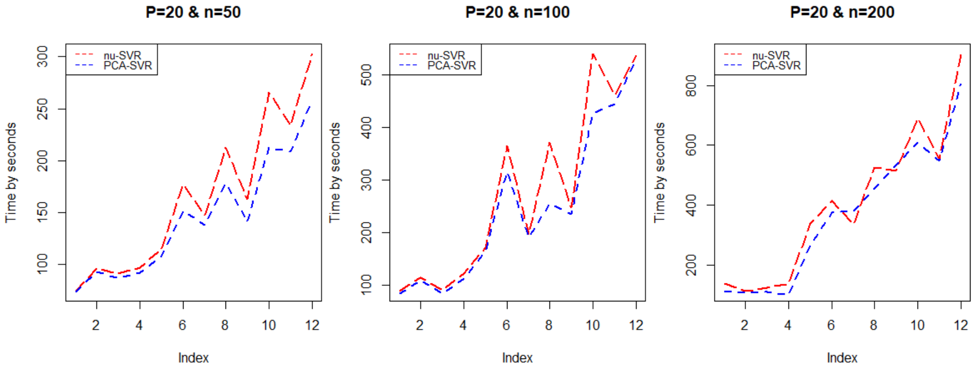

Figure 3. Computational running time for nu-SVR and PCA-SVR when $p =\ 20$ and n =(50,100 and 200)

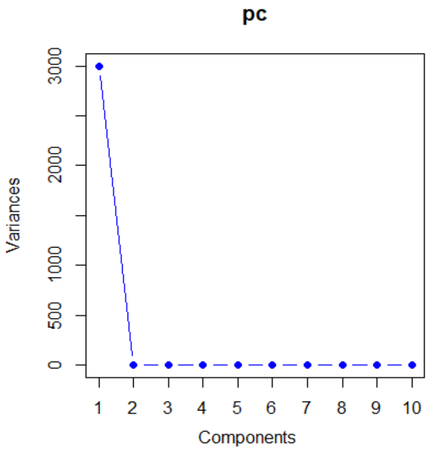

Figure 4. Principal components for PCA-SVR method when p=3000

Table 3. The prediction error (MSE) of *nu*-SVR and PCA-SVR methods when $p = 3000$ and $n = (50, 100\ and\ 200)$

n	Parameters	*nu*-SVR						PCA-SVR					
		nu =0.2			*nu* =0.3			*nu* =0.2			*nu* =0.3		
		C=1	*C*=50	*C*=100	*C*=1	*C*=50	*C*=100	*C*=1	*C*=50	*C*=100	*C*=1	*C*=50	*C*=100
50	*h*=0.5	21.938	1.264	1.311	22.138	1.206	1.229	**0.420**	**0.421**	**0.458**	**0.406**	**0.482**	**0.465**
	h=1	22.025	1.322	1.351	22.186	1.366	1.353	**0.457**	**0.519**	**0.516**	**0.521**	**0.545**	**0.551**
100	*h*=0.5	11.707	1.506	1.544	11.958	1.529	1.619	**0.423**	**0.466**	**0.558**	**0.529**	**0.544**	**0.568**
	h=1	11.688	1.408	1.425	11.805	1.399	1.421	**0.582**	**0.592**	**0.626**	**0.550**	**0.617**	**0.618**
200	*h*=0.5	6.793	1.786	1.750	7.071	1.725	1.856	**0.621**	**0.649**	**0.637**	**0.627**	**0.712**	**0.825**
	h=1	6.577	1.650	1.773	6.653	1.825	1.896	**0.631**	**0.720**	**0.724**	**0.637**	**0.811**	**0.886**

Table 4. Comparison of the computational times of *nu*-SVR and PCA-SVR methods when $p = 3000$ and $n = (50, 100\ and\ 200)$

n	Parameters	*nu*-SVR						PCA-SVR					
		nu =0.2			*nu* =0.3			*nu* =0.2			*nu* =0.3		
		C=1	*C*=50	*C*=100	*C*=1	*C*=50	*C*=100	*C*=1	*C*=50	*C*=100	*C*=1	*C*=50	*C*=100
50	*h*=0.5	2430.6	2490.6	2622.6	1858.8	2526.6	2221.2	**912.6**	**957**	**961.8**	**610.8**	**863.4**	**949.8**
	h=1	2242.8	2320.8	2211.6	2238	2353.8	2731.8	**666.6**	**992.4**	**963**	**607.8**	**882.6**	**990.6**
100	*h*=0.5	1907.4	4500	5832	1974.6	4968	3960	**2298.6**	**2424.6**	**2457.6**	**2272.2**	**2544.6**	**2553.6**
	h=1	2170.2	4140	4932	2412	3780	6120	**2106**	**2452.2**	**2648.4**	**2049.6**	**2567.4**	**2724.6**
200	*h*=0.5	3101.4	8820	10080	4441.6	9000	10296	**2716**	**5400**	**5724**	**2680**	**5436**	**5760**
	h=1	3346.8	9000	10260	4561.6	9216	10440	**2788**	**5436**	**5688**	**2752**	**5508**	**5832**

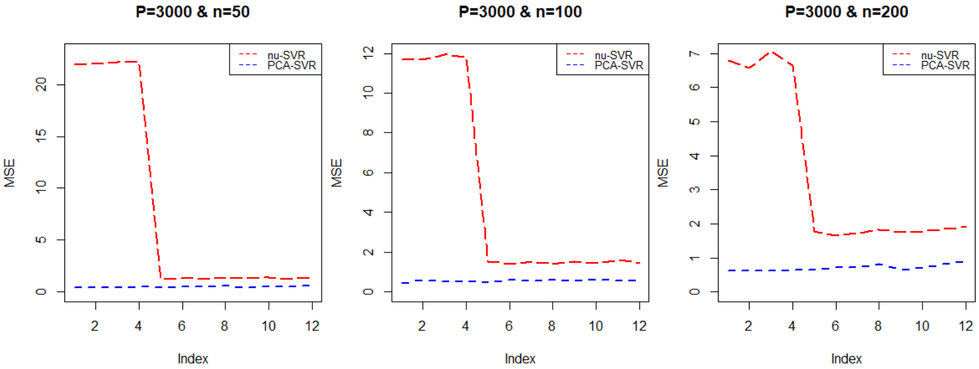

Figure 5. The MSE of *nu*-SVR and PCA-SVR for 3000 predictors

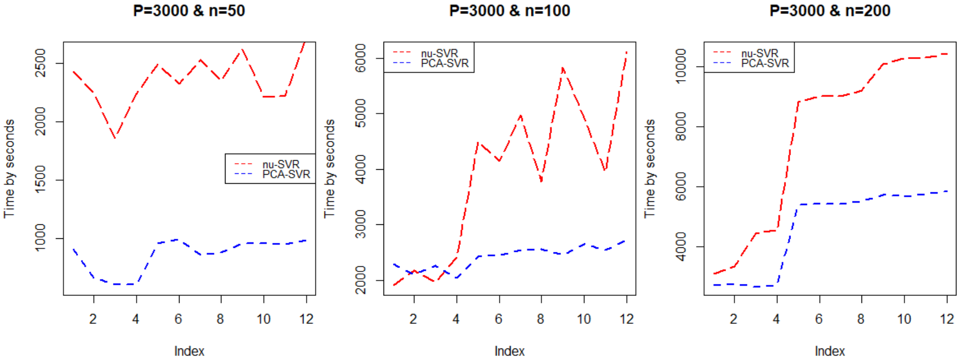

Figure 6. Computational running time for *nu*-SVR and PCA-SVR when $p = 3000$ and $n = (50, 100\ and\ 200)$

4. Real Case Studies

In this section, the comparison methods PCA-SVR and *nu*-SVR are illustrated through the analysis of two real data sets, first the prostate cancer data (low-dimensional data), and second the microarray data (high-dimensional data). These approaches were assessed using the mean square error of the testing data (MSE). The Bessel kernel function is used to transform the input space into high-dimensional feature space. These data sets have been split into 70% as training data and 30% as testing data.

4.1 Prostate Cancer Data

The prostate cancer data is the first real example in this paper. This dataset has been studied for estimations and variable selection by several researchers (Stamey et al., 1989; Friedman, 2001; Dhhan et al., 2018). This data set includes 97 observations as sample size ($n = 97$) and 9 variables ($p = 9$), and that makes it low dimensional data ($p < n$). According to Wahid et al. (2017), the prostate cancer data contains three outliers in the response variable. This data set has been split into 70% as a training sample for constructing the model and 30% as a testing sample for testing the model. The PCA-SVR method has selected three principal components, which are the most components that can explain the variability in this data set, as shown in Figure 7(a). Hence, Table 5 summarises the prediction errors (MSE) for the *nu*-SVR and PCA-SVR methods by selecting each parameter (C, nu, and h) for three different values (small, medium, and large). As we observe in Figure 7 (b), the proposed PCA-SVR method has the lowest MSE than the *nu*-SVR method.

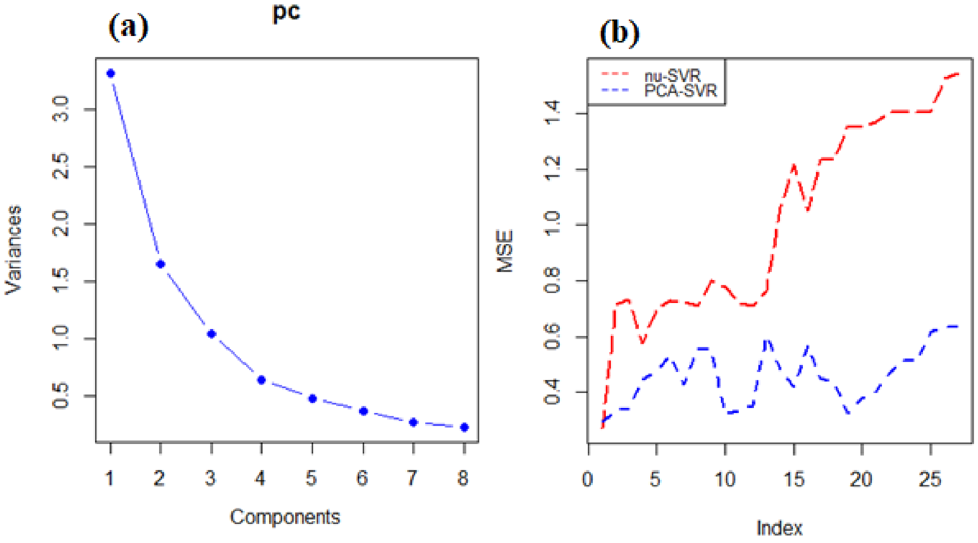

Figure 7. Section (a) provides the principal components for the PCA-SVR method for the Prostate cancer data, and Section (b) provides the MSE of *nu*-SVR and PCA-SVR methods for Prostate cancer data.

Table 5. The MSE of *nu*-SVR and PCA-SVR methods for Prostate cancer data set

Parameters		**nu-SVR**			**PCA-SVR**		
		***C*=1**	***C*=50**	***C*=100**	***C*=1**	***C*=50**	***C*=100**
***h*=0.5**	**nu =0.1**	**0.2700628**	0.5778399	0.7252587	0.292268	**0.4439863**	**0.4261309**
	nu =0.2	0.716582	0.6962048	0.7087298	**0.3315941**	**0.4773234**	**0.5591103**
	nu =0.3	0.7331763	0.7299576	0.7982088	**0.3447063**	**0.5307738**	**0.547459**
***h*=1**	**nu =0.1**	0.7764641	0.765243	1.049053	**0.3192888**	**0.6031799**	**0.5646778**
	nu =0.2	0.716974	1.064234	1.233345	**0.3346457**	**0.4820845**	**0.4509116**
	nu =0.3	0.7106957	1.21813	1.233552	**0.3525686**	**0.4168797**	**0.431277**
***h*=5**	**nu =0.1**	1.354441	1.404198	1.406178	**0.3252778**	**0.46697**	**0.6159731**
	nu =0.2	1.350312	1.404217	1.524214	**0.3763249**	**0.5139255**	**0.6356268**
	nu =0.3	1.369982	1.404226	1.545266	**0.3993474**	**0.5141451**	**0.6357557**

4.2 Microarray Data-Riboflavin Production by Bacillus Subtilis

Real high-dimensional data is our second example in this paper. Riboflavin production by bacillus subtitratus (Microarray data) has been studied by Bühlmann and Van (2011). This dataset consists of the response variable Y, which measures the logarithm of the production rate of riboflavin, and the explanatory variables $p = 4088$, which corresponds to the logarithms of expression levels of genes with a sample size of $n = 71$, taken from the genetically homogeneous sample. In order to achieve the dimensional reduction, the proposed PCA-SVR method selected the first three principal components, which are the most components that can explain the variability in the microarray data set; see Figure 8(a). We used 70% of the data set as training data and 30% as testing data to test the performance of each method.

Table 6 shows the result of applying *nu*-SVR and PCA-SVR with different sets of parameters (C, nu, and h). As shown in Figure 8(b), the proposed PCA-SVR method has lower MSE values compared to the standard method *nu*-SVR, indicating that the PCA-SVR method outperforms the *nu*-SVR method in terms of performance and efficiency.

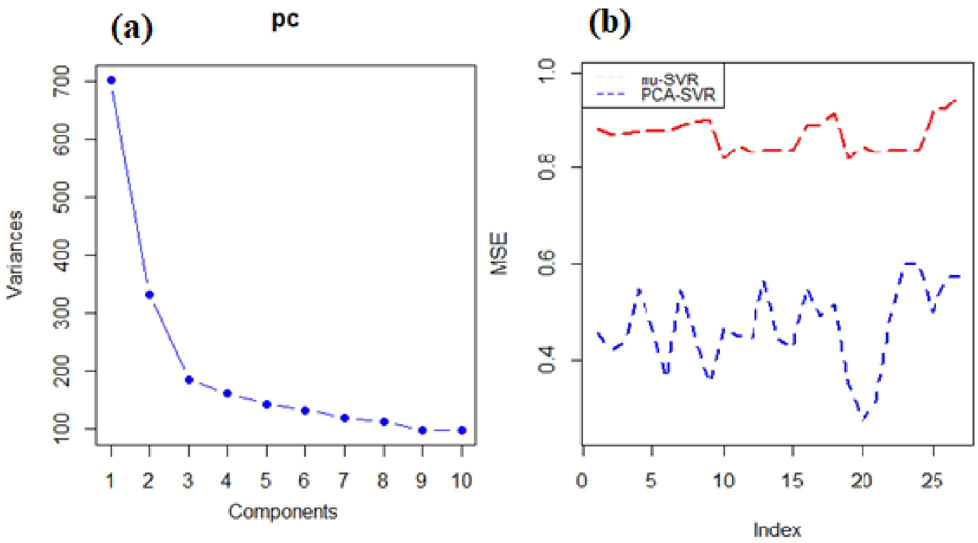

Figure 8. Section (a) provides the principal components for the PCA-SVR method for Microarray data, and Section (b) provides The MSE of *nu*-SVR and PCA-SVR methods for Microarray data

Table 6. The MSE of *nu*-SVR and PCA-SVR methods for Microarray data set

Parameters		***nu*-SVR**			**PCA-SVR**		
		***C*=1**	***C*=50**	***C*=100**	***C*=1**	***C*=50**	***C*=100**
***h*=0.5**	***nu* =0.1**	0.8827066	0.8777829	0.887125	**0.4541419**	**0.5472518**	**0.544104**
	***nu* =0.2**	0.8694467	0.8777833	0.896458	**0.4168072**	**0.4617277**	**0.4462234**
	***nu* =0.3**	0.8715451	0.8777832	0.899845	**0.4346946**	**0.3610354**	**0.3494144**
***h*=1**	***nu* =0.1**	0.8167055	0.8373702	0.886954	**0.4643128**	**0.5628258**	**0.5491446**
	***nu* =0.2**	0.8480435	0.8373725	0.889664	**0.4461962**	**0.4398063**	**0.4896922**
	***nu* =0.3**	0.8330664	0.8373614	0.912354	**0.443401**	**0.4223772**	**0.5141574**
***h*=5**	***nu* =0.1**	0.817066	0.8377712	0.915478	**0.344176**	**0.4876355**	**0.4985552**
	***nu* =0.2**	0.8486222	0.8377717	0.923154	**0.2722404**	**0.6001547**	**0.5708413**
	***nu* =0.3**	0.8332969	0.8377631	0.951472	**0.3113499**	**0.6000107**	**0.5702908**

5. Discussion and Conclusions

In this paper, we developed a hybrid prediction model PCA-SVR to achieve dimensional reduction and high prediction accuracy at the same time. Two simulation studies and real data sets with different scenarios of dimension (low and high) are employed to evaluate the performance of the PCA-SVR method. The numerical evidence shows the superiority of the PCA-SVR method over *nu*-SVR in having the smallest values of MSE and taking less computational time in both situations (low- and high-dimensional data sets) as well as in reducing the curse of dimensionality.

REFERENCES

[1] **Alter, O., Brown, P.O., Botstein, D. (2000),** ***Singular value decomposition for genome-wide expression data processing and modeling***. *Proceedings of the National Academy of Sciences*, 97(18), 10101-10106;

[2] **Balfer, J., Bajorath, J. (2015),** ***Systematic artifacts in support vector regression-based compound potency prediction revealed by statistical and activity landscape analysis***. *PloS one*, 10(3), e0119301;

[3] **Bühlmann, P., Van De Geer, S. (2011),** ***Statistics for high-dimensional data: methods, theory and applications***. *Springer Science & Business Media*;

[4] **Dhhan, W., Alshaybawee, T. (2017),** ***Elastic net for single index support vector regression model.*** *Economic Computation & Economic Cybernetics Studies & Research*, 51(2);

[5] **Dhhan, W., Rana, S., Alshaybawee, T., Midi, H. (2018),** ***The single-index support vector regression model to address the problem of high dimensionality***. *Communications in Statistics-Simulation and Computation*, 47(9), 2792-2799;

[6] **Friedman, J., Hastie, T., Tibshirani, R. (2001),** *Vol. 1 of The elements of statistical learning*;

[7] **Fukunaga, K. (2013),** ***Introduction to statistical pattern recognition***. *Elsevier*;

[8] **Gan, L., Lv, W., Zhang, X., Meng, X. (2012),** ***Improved PCA+ LDA applies to gastric cancer image classification process***. *Physics Procedia*, 24, 1689-1695;

[9] **Hampel, F.R., Ronchetti, E.M., Rousseeuw, P., Stahel, W.A. (1986),** ***Robust statistics: the approach based on influence functions***. *Wiley-Interscience*; New York;

[10] **Hastie, T., Tibshirani, R., Friedman, J. (2009),** ***Unsupervised learning.*** *The elements of statistical learning*, 485-585;

[11] **Hotelling, H. (1933),** ***Analysis of a complex of statistical variables into principal components.*** *Journal of educational psychology*, 24(6), 417;

[12] **Hubert, M., Rousseeuw, P.J., Vanden Branden, K. (2005),** ***ROBPCA: a new approach to robust principal component analysis***. *Technometrics*, 47(1), 64-79;

[13] **Jimenez, L.O., Landgrebe, D.A. (1998),** ***Supervised classification in high-dimensional space: geometrical, statistical, and asymptotical properties of multivariate data.*** *IEEE Transactions on Systems, Man, and Cybernetics*, Part C (Applications and Reviews), 28(1), 39-54;

[14] **Lahiri, S.K., Ghanta, K. C. (2009),** ***Support vector regression with parameter tuning assisted by differential evolution technique: Study on pressure drop of slurry flow in pipeline.*** *Korean journal of chemical engineering*, 26(5), 1175-1185;

[15] **Mohammed Rashid, A., Midi, H., Dhhan, W., Arasan, J. (2022),** ***Detection of outliers in high-dimensional data using nu-support vector regression.*** *Journal of Applied Statistics*, 49(10), 2550-2569;

[16] **Narasimhan, S., Shah, S.L. (2008),** ***Model identification and error covariance matrix estimation from noisy data using PCA.*** *Control Engineering Practice*, 16(1), 146-155;

[17] **Rashid, A.M., Midi, H., Slwabi, W.D.., Arasan, J. (2021),** ***An efficient estimation and classification methods for high dimensional data using robust iteratively reweighted SIMPLS algorithm based on nu-support vector regression.*** *IEEE Access*, 9, 45955-45967;

[18] **Schölkopf, B., Bartlett, P.L., Smola, A.J., Williamson, R. (1999),** ***Shrinking the tube: a new support vector regression algorithm***. *Advances in neural information processing systems*, 330-336;

[19] **Stamey, T.A., Kabalin, J.N., McNeal, J.E., Johnstone, I.M., Freiha, F., Redwine, E.A., Yang, N. (1989),** ***Prostate specific antigen in the diagnosis and treatment of adenocarcinoma of the prostate. II. Radical prostatectomy treated patients.*** *The Journal of urology*, 141(5), 1076-1083;

[20] **Ukil, A. (2007),** ***Intelligent systems and signal processing in power engineering***. *Springer Science & Business Media*;

[21] **Üstün, B., Melssen, W.J., Buydens, L.M. (2006),** ***Facilitating the application of support vector regression by using a universal Pearson VII function-based kernel.*** *Chemometrics and Intelligent Laboratory Systems*, 81(1), 29-40;

[22] **Vapnik, V. (1999),** ***The nature of statistical learning theory***. *Springer Science & Business Media*;

[23] **Virmani, J., Dey, N., Kumar, V. (2016),** ***PCA-PNN and PCA-SVM based CAD systems for breast density classification.*** In *Applications of intelligent optimization in biology and medicine*, 159-180, Springer, Cham;

[24] **Wahid, A., Khan, D.M., Hussain, I. (2017),** ***Robust Adaptive Lasso method for parameter's estimation and variable selection in high-dimensional sparse models.*** *PLoS one*, 12(8), e0183518;

[25] **Ye, W., Peng, C. (2018),** ***Recognition algorithm of emitter signals based on PCA+ CNN.*** In *2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference* (IAEAC), 2410-2414, IEEE;

[26] **Zou, H., Xue, L. (2018),** ***A selective overview of sparse principal component analysis.*** *Proceedings of the IEEE*, 106(8), 1311-1320.