Sains Malaysiana 54(11)(2025): 2773-2783
http://doi.org/10.17576/jsm-2025-5411-16

Determination of the Optimal Number of PLS Components Based on the Combination of Cross-Validation and RMD-MRCD-PCA Weighting Function

(Penentuan Bilangan Komponen PLS yang Optimum berasaskan Gabungan Pengesahan Silang dan Fungsi Pemberat RMD-MRCD-PCA)

Habshah Midi[1], Siti Zahariah Abdul Wahab[2,*] & Azree Shahrel Ahmad Nazri[1]

[1]*Institute for Mathematical Research, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia*
[2]*Malaysian Institute of Information Technology, Universiti Kuala Lumpur, 50250 Kuala Lumpur, Malaysia*

Received: 19 November 2024/Accepted: 24 October 2025

ABSTRACT

Partial least squares (PLS) regression is a very useful tool for the analysis of high dimensional data (HDD). Choosing the ideal number of PLS components is a vital step in developing the best model. The accuracy of the model will be affected if there are too many or too few PLS components being selected. Numerous classical methods, such as the leave-one-out cross-validation (LOOCV) and K-fold cross-validation (K-FoldCV) are developed to determine the optimal number of PLS components. Nonetheless, they are easily affected by high leverage points (HLPs). Thus, robust cross validation techniques, denoted as RMD- MRCD-PCA-LOOCV and RMD-MRCD-PCA-K-FoldCV are proposed to remedy this problem. The results of the simulation study and real data set indicate that the proposed methods successfully select the appropriate number of PLS components.

Keywords: High leverage points; leave-one-out cross validation; minimum regularized covariance determinant; partial least squares; principal component analysis

ABSTRAK

Regresi kuasadua kecil separa (PLS) adalah kaedah yang sangat berguna bagi menganalisis data berdimensi tinggi (HDD). Pemilihan bilangan komponen PLS yang ideal adalah langkah penting bagi membangunkan model terbaik. Ketepatan model akan dipengaruhi sekiranya terlalu banyak atau terlalu sedikit komponen PLS yang dipilih. Pelbagai kaedah klasik seperti pengesahan silang *leave-one-out* (LOOCV) dan pengesahan silang lipatan *K* (*K*-FoldCV) dibangunkan untuk menentukan bilangan komponen PLS yang optimum. Namun begitu, mereka mudah dipengaruhi oleh titik tuasan tinggi (HLPs). Oleh itu teknik pengesahan silang teguh yang ditandakan dengan RMD- MRCD-PCA-LOOCV dan RMD-MRCD-PCA-K-FoldCV dicadangkan bagi menyelesaikan masalah ini. Keputusan kajian simulasi dan set data sebenar menunjukkan kaedah yang dicadangkan berjaya memilih bilangan komponen PLS yang sesuai.

Kata kunci: Analisis komponen utama; kuasadua terkecil separa; penentu kovarian teratur minimum; pengesahan silang *leave-one out;* titik tuasan tinggi

INTRODUCTION

In regression modelling, the problem of collinear explanatory variables can be solved by separating them into orthogonal latent variables. With a vast number of explanatory variables being measured in high dimensional data (HDD), the situation of collinearity has become the norm rather than the exception. High dimensional data refers to the situations where the number of covariates or predictors is much larger than the number of data points (i.e., $p >> n$) (Abdullah & Habshah 2023; Abdullah et al. 2021; Habshah et al. 2025). Partial least squares regression is the most commonly used method for handling collinearity. However, in PLS modelling, it is very crucial to select the ideal number of PLS components or PLS latent variables in order to determine the optimal level of complexity for a PLS regression model. PLS model's complexity is controlled by the number of PLS components. Too few components indicate under fitting, while too many PLS components indicate overfitting of data. Both outcomes may have an adverse effect on prediction performance. The ideal number of PLS components can be estimated statistically using cross validation (CV) procedure. The CV is the widely used method to determine the PLS components. The approach is to divide the data at random into training and test dataset. Then, PLS is used on the training data as well as the test dataset for $a = 1, 2,\ldots, A$, where A is the maximum number of PLS components. The prediction error with respect to the actual test data can be calculated. Repeating this technique multiple times gives the prediction error distribution for 1 to a components, which helps to determine the ideal number of PLS components.

The CV is basically a leave-one-out cross validation (LOOCV) suggested by Mosteller and Wallace (1963). It evaluates the prediction power of the predicted models according to the number of components included in the model. Nonetheless, when dealing with huge datasets, LOOCV can be computationally intensive and time consuming. This is due to the fact that LOOCV fits the model iteratively on the entire training set. The other problem with LOOCV is that it can be prone to high variance or overfitting, which means that it requires almost all of the training data to learn and only a single observation to evaluate. To address the drawbacks of LOOCV, Geisser (1975) introduced the *K*-Fold cross validation (*K*-FoldCV). In *K*-FoldCV, the data are split into k randomly equal folds or groups. Then, in k different iterations, each of these folds is treated as a validation set. However, both CVs are evaluated using the classical mean square error (MSE), which is easily influenced by outliers and high leverage points, i.e., outlying observations in the X space. Moreover, when dealing with HDD, CVs can be time consuming, overfitting and under fitting. To remedy the problems of under fitting and overfitting of CVs, Filzmoser, Liebmann and Varmuza (2009) proposed a repeated double cross validation (RDCV) to determine the optimal number of PLS components to be included in the PLS regression model. The RDCV consists of two nested loops, outer loop and inner loop. The optimal number of components are selected in the inner loop and the performance of optimized model is validated in the outer loop. The RDCV, however, requires greater user expertise for proper use and the method is complicated. Furthermore, through our investigation, the downside of RDCV method is that it always underfitting by selecting fewer number of PLS components. Underfitting models will cause inaccurate results and the model is hard to be generalized to the future data. We also notice that the LOOCV, *K*-FoldCV, and RDCV are easily influenced by HLPs, outlying observations in the X space.

Thus, these weaknesses have inspired us to establish a robust weighted RMD-MRCD-PCA-LOOCV and robust weighted RMD-MRCD-PCA-*K*-FoldCV to determine the optimal number of PLS components. The new modification methods are the integration of the LOOCV with the weighting function of RMD-MRCD-PCA and the integration of *K*-FoldCV with the weighting function of RMD-MRCD-PCA. The RMD-MRCD-PCA is used to identify and reduce the effect of HLPs. At the outset, HLPs in a dataset are identified. Afterwards, the effect of HLPs is reduced by applying the proposed weighting function and then the CV procedures are executed to determine the number of PLS components.

The rest of the paper is structured as follows. The Cross Validation technique will be introduced in the next section and followed by the proposed robust weighted RMD-MRCD-PCA cross-validation method. The subsequent section will present the simulation study and real data examples. Finally, the conclusion of the study is presented in the last section.

CROSS VALIDATION (CV)

The ideal number of PLS components can be estimated statistically using cross validation procedure. The approach is to divide the data at random into training and test dataset. Then, PLS is used on the training data as well as the test dataset for $a = 1, 2,\ldots, A,$ where A is the maximum number of PLS components. The prediction error with respect to the actual test data can be calculated. Repeating this technique multiple times gives the prediction error distribution for 1 to a components, which helps to determine the ideal number of PLS components. The following existing methods (LOOCV and *K*-Fold CV) are discussed and the computation of their algorithms are illustrated to get better understanding of our proposed methods which incorporates cross validation procedures.

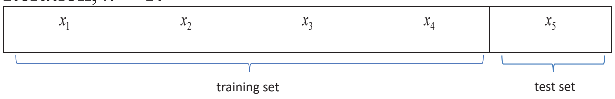
LEAVE-ONE-OUT CROSS VALIDATION (LOOCV)

Mosteller and Wallace (1963) suggested cross-validation method and it is referred as leave-one-out cross validation (Xu & Liang 2000). Let say a model is trained n times, where n is the sample size. Each time only one observation is used as a test set while the rest are used to train the model. The number of iterations, $k = 1, 2,\ldots, K$. For instance, $n = 5$, then it will be 5 test subsets, $S_i = 1, 2,\ldots, n$ and the iterations, $k = 1, 2, 3, 4, 5$, here the maximum number of iterations, $K = 5$.

$S_1 = x_1$, $S_2 = x_2$, $S_3 = x_3$, $S_4 = x_4$, $S_5 = x_5$

Step 1: Split a dataset into a training set and a test set.

Iteration, $k = 1$:



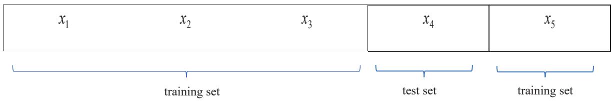
Step 2: Build the model on the training set, y_{train}.

Step 3: Used the trained model in Step 2 to predict the response value, $\hat{y}_{test_i}$ in test set.

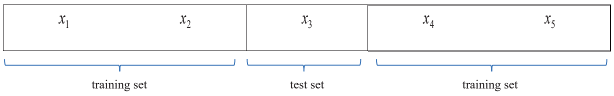
Step 4: Calculate the squared error prediction, $SEP_i = \sum_{i=1}^{n_{test}} (y_{test_i} - \hat{y}_{test_i})^2$ for i equals to the observation in test set.

Step 5: The iteration continues to $k = 2, 3, 4, 5$. Here the value of $K = n$ since the iteration for leave-one-out (LOO) is for each observation.

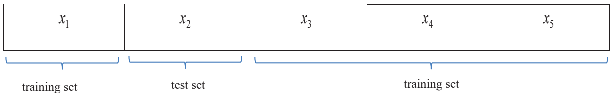
Iteration 2:



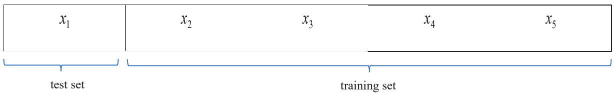
Iteration 3:



Iteration 4:



Iteration 5:



Steps 2 – 4 are repeated for each iteration, *k*.

Step 6: Find the final score of mean squared error prediction,

$$MSEP=\frac{1}{n_{test}K}\sum_{k=1}^{K}\sum_{i=1}^{n_{test}}(y_{test_i}-\hat{y}_{test_i})^2$$

K-FOLD CROSS VALIDATION (*K*-FOLD CV)

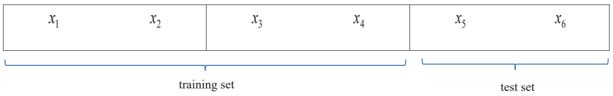
K-Fold cross validation was designed by Geisser (1975). In *K*-Fold cross validation, the dataset is divided into *K* equally sized subsets. Then, repeat the train-test method *k* times such that each time one of the *k* subsets is used as a test set and the rest of *k*-1 subsets are used together as a training set. For instance, the dataset consists of six observations, i.e., $S=\{x_1, x_2, x_3, x_4, x_5, x_6\}$. The following steps are used to illustrate the *K*-Fold cross validation.

Step 1: Divide the data into *K* equal samples.
$S_1=\{x_1, x_2\}$; $S_2=\{x_3, x_4\}$; $S_3=\{x_5, x_6\}$

There are 3 samples, thus, the number of iterations, $K = 3$.

Step 2: Split the data set into training and test set.

Iteration 1:



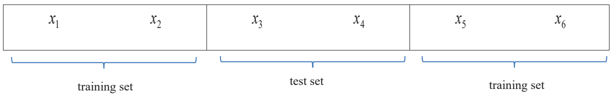
Step 3: Build the model on the training set, y_{train}.

Step 4: Use the model in Step 3 to make predictions on the test set, $\hat{y}_{test_i}$.

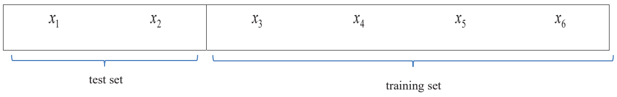
Step 5: Calculate the squared error prediction, $SEP_i=\sum_{i=1}^{n_{test}}(y_{test_i}-\hat{y}_{test_i})^2$ for *i* equals to the observation in test set.

Step 6: The iteration continues to $k = 2, 3$.

Iteration 2:



Iteration 3:



Repeat Steps 3 – 5 for each iteration.

Step 7: Find the final score of mean squared error prediction,

$$MSEP=\frac{1}{n_{test}K}\sum_{k=1}^{K}\sum_{i=1}^{n_{test}}(y_{test_i}-\hat{y}_{test_i})^2$$

THE ROBUST WEIGHTED CROSS-VALIDATION BASED ON RMD-MRCD-PCA

The proposed robust weighted RMD-MRCD-PCA cross-validation is constructed by integrating a proposed robust weighting function based on the RMD-MRCD-PCA. The RMD-MRCD-PCA method is enhanced to reduce the effect of HLPs in a dataset by giving smaller weights to each of HLPs. The proposed method consists of two stages whereby in the first stage, the weighting scheme based on RMD-MRCD-PCA is proposed. Subsequently, in the second stage, the robust weighted RMD-MRCD-PCA is modified to obtain the smallest root mean squared error prediction (RMSEP) of the model.

Stage 1: The Formulation of Robust Weighting Function

The diagnostic robust Mahalanobis distance based on the combination of minimum regularized covariance determinant and principal component analysis (RMD-MRCD-PCA) developed by Siti Zahariah and Habshah (2022) is very successful in the detection of multiple HLPs in high dimensional data. The method consists of three steps whereby, in the first step, the dimension of a high dimensional dataset is reduced by using the PCA method. The idea to exploit PCA is to replace the original *X* matrix with *a* eigenvectors corresponding to the largest eigenvalues, where the number of eigenvectors representing the new dimension size for a reduced dataset. In the second step, generate a fitted $\hat{X}$ matrix in the original dimension *p* based on the chosen number of principal components such that its cumulative variance is at least 80%. In the third step, the fitted $\hat{X}$ matrix will be shrunk, and an invertible covariance matrix for HDD is yielded. The MRCD (Boudt et al. 2018) was then performed on these fitted $\hat{X}$ to determine the robust mean and robust covariance of HDD. By using these robust estimators, the distance of each observation was computed by employing the robust Mahalanobis distance (RMD) based on MRCD-PCA.

The advantage of this method is that data are transformed based on the orthogonal PCA components, and it can reduce noise and solve the multicollinearity issue without losing much information. After obtaining robust multivariate location and scale estimates given by MRCD, the robust Mahalanobis distance (RMD) is computed as $RMD_{mrcd-pca_i} = \sqrt{(\hat{x}_i - \hat{\mu}_{mrcd-pca})\hat{\Sigma}^{-1}_{mrcd-pca}(\hat{x}_i - \hat{\mu}_{mrcd-pca})^T}$, where $\hat{\mu}_{mrcd-pca}$ and $\sum_{mrcd-pca}$ are the estimates of the robust mean and robust covariance matrix of MRCD-PCA, respectively. $\hat{x}$ is the fitted observations of $\hat{X}$ and $i = 1, 2, \ldots, n$. Since the distribution of $RMD_{mrcd-pcai}$ is intractable, as per Abdullah et al. (2021) a confidence bound type cut-off point is used as $(cut-off)_{rmd-mrcd-pca} = median(RMD_{mrcd-pca}) + 3MAD(RMD_{mrcd-pca})$.

Any observation that corresponds to $RMD_{mrcd-pca_i}$ exceeding this cut-off point is considered as HLPs. Following Abdullah et al. (2021), Baba, Habshah and Nur Haizum (2022), Coakley and Hettmansperger (1993), Habshah et al. (2021), Rousseeuw and van Zomeren (1990), and Waleed, Sohel and Habshah (2016), a weight function is formulated based on the diagnostic method of the detection of outliers with the main aim of reducing their effect on the parameter estimates.

Thus, a new weight function is formulated based on RMD-MRCD-PCA as follows:

$$w_{rmd-mrcd-pca_i} = \left[1, \frac{(cut-off)_{rmd-mrcd-pca}}{RMD_{mrcd-pca_i}}\right] \quad (1)$$

where weight equals to $\frac{(cut-off)_{rmd-mrcd-pca}}{RMD_{mrcd-pca_i}}$ is given to leverage points, and a weight equal to 1 is given to regular observations.

Stage 2: The Extension of RMD-MRCD-PCA Cross-Validation

Let consider a multiple linear regression $Y = X\beta + \varepsilon$, where X is an $n \times p$ matrix, Y is an $n \times 1$ vector of response variables with unknown regression coefficients $\beta = p \times 1$ and the error term ε $n \times 1$. The goal of PLS regression is to find a linear relation between the X and Y variables. Instead of finding this relationship directly, in PLS regression both X and Y are modelled by orthogonal linear latent variables as presented in Equations (2) and (3).

$$X = TP^T + \varepsilon_x \quad (2)$$

$$Y = UQ^T + \varepsilon_y \quad (3)$$

With the error matrices ε_x and ε_y, where T and P are the PLS scores and loadings matrices of X. While U and Q are the scores and loadings matrices of Y. The matrices T, P, U and Q have columns, with $a \leq \min(p)$ being the number of PLS components. The X matrix in PLS regression can be decomposed as follows and assume there is no errors in data matrix: $X = t_1 p_1^T + t_2 p_2^T + \ldots + t_a p_a^T$, where $a = 1,2,3, \ldots, A$ and the number of maximum PLS components is denoted as A. The PLS scores, T can be calculated as $T = XW$, where $W = w, w_{2,\ldots,} w_a$ is the eigenvector to the largest eigenvalue of sample covariance matrix. The PLS score, t_i is a linear combination of the x-variables and can be acknowledged as good summaries of the predictor variables. The U scores are linear combinations of the y-variables and can be considered as good interpretations of response variables (for multivariate case). T, P, U and Q can be signified as t_a, p_a, u_a and q_a with a^{th} columns, where $a = 1, 2, 3, \ldots, A$. u_a and t_a are connected by the inner linear relationship, $u_a = d_a t_a + h_a$ with h_a being the residuals and d_a the regression parameters. Thus, if the linear relationship between u_1 and t_1 is strong, then, the x-score of the first PLS component is good for predicting y-scores and finally for predicting y-data. The optimum number of PLS components to model Y by X can be estimated by cross validation. For univariate case, the covariance between $x -$ scores, t_a and y has to be maximized and the formula can be written as $y = d_a t_a + h$. The number of components a is also known as the dimension of the model. The least squares solution of Equation (4) is

$$\hat{d}_a = (T_a^T T_a)^{-1} T_a^{-1} y \quad (4)$$

The PLS estimator $\hat{\beta}_a$ with a component can be defined as in Equation (5)

$$\hat{\beta}_a = W_a (T_a^T T_a)^{-1} W_a^{-1} X^T y \quad (5)$$

To determine the number of PLs component, first, the RMD-MRCD-PCA procedure is performed to identify the HLPs in a dataset. Afterwards, all the HLPs are downweighted by using our proposed weighting function in Stage 1 to reduce the effect of HLPs in the system. The HLPs are downweighted by multiplying the weight function, $w_{rmd-mrcd-pca}$ in Equation (1) and the dataset, X and it can be written as $X^w = w_{rmd-mrcd-pca} X$.

For the cross-validation process, the weighted data, X^w is split into training and test set. Training set is used for fitting the model and test set is used for validating the model. For each sample split, the model is fitted by the n_{train} samples. $\hat{\beta}_{train,a} = W_{train,a}(T^T_{train,a} T_{train,a})^{-1} W^{-1}_{train,a} X^T_{train,a} y_{train}$, where all the parameters are determined based on the train dataset. Then, the fitted model from the train set predicts the response vectors using the data in test set.$\hat{y}_{test,a} = X_{test}\hat{\beta}_{train,a}$ $\hat{y}_{test,a =} X_{test}$. The square error prediction is determined over all samples in test set. The algorithms for the proposed robust weighted cross-validation based on RMD-MRCD-PCA (RMD-MRCD-PCA-LOOCV and RMD-MRCD-PCA-*K*-FoldCV) method is summarized as follows:

Step 1: Perform the RMD-MRCD-PCA procedure on dataset to determine the HLPs.

Step 2: Reduced the effect of HLPs by multiplying the weight function in Equation (1) to the dataset,
$X^w = w_{rmd-mrcd-pca.} X$

Step 3: Split the weighted dataset, X^w into a training set, x^w_{train} and a test set,x^w_{test}.

For a = 1 and k = 1, where the number of PLS components, a = 1, 2, …, A and the number of iteration of cross validation, k = 1, 2, …, K. The number of K is equal to n for the iteration of leave-one-out (LOO) which is equal to the number of sample size. Whereas, the number of K is equal to k-subsets for the iteration of K-FoldCV.

Step4: Perform the SIMPLS with number of PLS components, a= 1 on the weighted dataset to obtain the parameter estimates, $\hat{\beta}_{train,a} = W_{train,a}(T^T_{train,a}T_{train,a})^{-1}W^{-1}_{train,a}(X^w)^T_{train,a}y_{train}$

Step 5: Used the estimates in Step 4 to predict the response value, $\hat{y}_{test,a} = X^w_{test}\beta_{train,a}$ in test set.

Step 6: Calculate the squared error prediction, $SEP_i = \sum_{i=1}^{n_{test}}(y_{test_i} - \hat{y}_{test_i,a})^2$.

Repeat Steps 4 – 6 for iteration, k = 2, 3, …, K.

Step 7: Determine the root mean squared error prediction $RMSEP = \sqrt{\frac{1}{n_{test}K}\sum_{k=1}^{K}\sum_{i=1}^{n_{test}}(y_{test_i} - \hat{y}_{test_i,a})^2}$ for all split samples K.

Step 8: Repeat

Steps 4 – 8 for the next PLS components, a = 2, 3,…, A.

The iteration will be continued until the value of RMSEP for PLS components a is relatively constant. The optimal number of components is chosen as a for which the RMSEP value decreases significantly and remains constant.

MONTE CARLO SIMULATION

In this study, the performances of cross-validation methods, namely the classical LOOCV, K-FoldCV, and RDCV are compared to the RMD-MRCD-PCA-LOOCV and RMD-MRCD-PCA-K-FoldCV using Monte Carlo simulation. Codes for the simulations are written using R programming. The simulation design in this study is similar to that of Li, Morris and Martin (2002) and Nengsih et al. (2019). Nengsih et al. (2019) updated the data setting for cases with missing data sets and used the same setting for low and high dimension instances. While Li, Morris and Martin (2002) utilised the same simulation design for low dimension scenarios. The data settings are generated by using the function of simul_data_UniYX under plsRglm package in R programming under R = 500 simulation. The function generates a single univariate response value y and a vector of explanatory variables (X_1, X_2,..., X_p) drawn from a model with a given number of PLS components. In this study, we use three different sample sizes, n = 60, 100 and 200 with three different dimensions, p = 100, 500 and 1000. The number of PLS components or latent variables is set at a = 3, 4 and 6. K is set equals to 10 for the K-FoldCV and RMD-MRCD-PCA-K-FoldCV. K is the maximum number of iteration of cross-validation procedure. For K-FoldCV and RMD-MRCD-PCA-K-FoldCV, the iteration is based on the number of groups or folds in dataset, thus, we followed Nengsih et al. (2019) and set the maximum iteration, K equals to 10. Nengsih et al. (2019) evaluate the performance of various component selection methods for Partial Least Squares (PLS) regression under different missing data mechanisms. The primary objective was to assess how accurately these methods could identify the true number of components when data were missing either completely at random (MCAR) or at random (MAR). The simulation scenarios involved generating datasets with controlled missingness patterns and varying proportions of missing data to examine their impact on the accuracy of component selection. The results showed that under the MCAR assumption, the selected number of components was generally more accurate, whereas under MAR conditions, the methods tended to overestimate the number of components required. While Li, Morris and Martin (2002) aimed to compare the effectiveness of several model selection criteria in determining the optimal number of latent variables in PLS models. Their Monte Carlo simulations were designed using a large sample size of 1000 observations and five predictor blocks to mimic realistic multiblock data structures. The study evaluated criteria such as the Akaike Information Criterion (AIC), Osten's F-criterion, Wold's R-criterion, and Adjusted R², across multiple replications. Their analysis showed that most criteria consistently suggested models with four to six latent variables, though this conclusion was specific to low-dimensional settings. Since we are investigating the influence of high leverage points on the selection of optimal number of PLS latent variables in the model, the simulated dataset is randomly contaminated by 5%, 10% and 30% of HLPs. The contaminated observations are generated randomly by using the normal distribution with mean = 100 and standard deviation = 50. The root means square error prediction is calculated for each setting, $RMSEP = \sqrt{\frac{1}{n_{test}K}\sum_{k=1}^{K}\sum_{i=1}^{n_{test}}(y_{test_i} - \hat{y}_{test_i,a})^2}$. The results are exhibited in Tables 1-3. An effective method is one that accurately estimates the optimal number of PLS components, which should be equal to or close to the true number of latent variables. In our simulation settings, the true number of latent variables was set to 3, 4, and 6. In the case of 5% contamination (Table 1), our proposed approaches, RMD-MRCD-PCA-LOOCV and RMD-MRCD-PCA-10-FoldCV, correctly identified the number of latent variables for all settings (a = 3, 4, 6), regardless of sample size, except for datasets with dimensions (100 × 500) and (200 × 1000) when a = 6. In these cases, the methods selected 5 components instead of 6. Conversely, the classical LOOCV, 10-FoldCV, and RDCV methods exhibited underfitting by consistently selecting fewer components than the true number.

The performance of the proposed methods remained strong even under 10% contamination (Table 2), where they accurately selected the correct number of components. However, under 30% contamination (Table 3), for the dataset with dimensions (60 × 100) and a = 6, RMD-MRCD-PCA-10-FoldCV selected 5 components instead of 6.

It is evident that the performance of classical LOOCV, 10-FoldCV, and RDCV deteriorates as the contamination level increases, especially when the number of PLS components increases. These classical methods consistently exhibit underfitting, selecting a smaller number of components as contamination rises (Tables 2 & 3). In contrast, our proposed methods consistently select the correct number of PLS components, or values very close to the true number (i.e., 3, 4 or 6), in the majority of settings. Exceptions include datasets with dimensions (60 × 100) and (200 × 1000) when a = 6 under 5% contamination, and (60 × 100) with a = 6 under 30% contamination, where the selected number was 5 instead of 6.

We expect that our proposed methods should be able to correctly identify the true number of latent variables, which, in our simulation settings, are set to 3, 4, and 6. However, as previously mentioned, in certain scenarios, the number of components selected does not exactly match these true values. It is important to note that in simulation studies, even a well-designed method may not always recover the exact number of latent variables in every replicate. This is due to several factors, including sampling variability and finite-sample effects. Each simulated dataset contains random variation, and this noise can obscure the true signal, particularly in smaller samples, leading to slight under- or overestimation of the number of components. While many component selection methods are consistent in large samples, their performance can vary substantially in smaller or high-dimensional settings. For example, in cases where the number of variables p=1000, even a sample size of n=500 may be considered relatively small. Such finite-sample limitations are well-known challenges in simulation studies.

Despite these issues, our proposed methods consistently select values that are equal to or very close to the true number of components (i.e., 3, 4, or 6) more frequently than existing methods, which tend to deviate more substantially. This demonstrates that our proposed methods are more effective and robust in capturing the true underlying latent structure, even under realistic and challenging data conditions.

NUMERICAL EXAMPLE

The Biscuit Dough Dataset is used to assess the performance of our proposed approaches (RMD-MRCD-PCA-LOOCV and RMD-MRCD-PCA-*K*-FoldCV) and to compare the methods to the existing methods such as RDCV, LOOCV, and *K*-FoldCV. This data has been used by Hubert and

TABLE 1. The number of selected optimal number of PLS components for 5% of HLPs contamination

Contamination (%)	5%				
$n \times p$	60 × 100				
Number of latent variables	RMD-MRCD-PCA-LOOCV	RMD-MRCD-PCA-10-FoldCV	RDCV	LOOCV	10-FoldCV
$a = 3$	3	3	2	3	3
$a = 4$	4	4	2	5	3
$a = 6$	6	6	2	3	3
$n \times p$	100 × 500				
Number of latent variables	RMD-MRCD-PCA-LOOCV	RMD-MRCD-PCA-10-FoldCV	RDCV	LOOCV	10-FoldCV
$a = 3$	3	3	2	3	2
$a = 4$	4	4	2	4	4
$a = 6$	5	6	2	2	2
$n \times p$	200 × 1000				
Number of latent variables	RMD-MRCD-PCA-LOOCV	RMD-MRCD-PCA-10-FoldCV	RDCV	LOOCV	10-FoldCV
$a = 3$	3	3	2	3	3
$a = 4$	4	4	4	4	2
$a = 6$	6	5	3	3	2

TABLE 2. The number of selected optimal number of PLS components for 10% of HLPs contamination

Contamination (%)			10%		
n × *p*			60 × 100		
Number of latent variables	RMD-MRCD-PCA-LOOCV	RMD-MRCD-PCA-10-FoldCV	RDCV	LOOCV	10-FoldCV
a = 3	3	3	2	3	3
a = 4	4	4	2	4	3
a = 6	6	6	2	5	5
n × *p*			100 × 500		
Number of latent variables	RMD-MRCD-PCA-LOOCV	RMD-MRCD-PCA-10-FoldCV	RDCV	LOOCV	10-FoldCV
a = 3	3	3	3	2	2
a = 4	4	4	4	3	4
a = 6	6	6	2	4	2
n × *p*			200 × 1000		
Number of latent variables	RMD-MRCD-PCA-LOOCV	RMD-MRCD-PCA-10-FoldCV	RDCV	LOOCV	10-FoldCV
a = 3	3	3	2	2	2
a = 4	4	4	4	3	3
a = 6	6	6	2	3	3

TABLE 3. The number of selected optimal number of PLS components for 30% of HLPs contamination

Contamination (%)			30%		
n × *p*			60 × 100		
Number of latent variables	RMD-MRCD-PCA-LOOCV	RMD-MRCD-PCA-10-FoldCV	RDCV	LOOCV	10-FoldCV
a = 3	3	3	2	3	3
a = 4	4	4	2	4	4
a = 6	6	5	3	4	4
n × *p*			100 × 500		
Number of latent variables	RMD-MRCD-PCA-LOOCV	RMD-MRCD-PCA-10-FoldCV	RDCV	LOOCV	10-FoldCV
a = 3	3	3	2	2	2
a = 4	4	4	2	3	2
a = 6	6	6	3	2	2
n × *p*			200 × 1000		
Number of latent variables	RMD-MRCD-PCA-LOOCV	RMD-MRCD-PCA-10-FoldCV	RDCV	LOOCV	10-FoldCV
a = 3	3	3	2	2	2
a = 4	4	4	4	2	2
a = 6	6	6	3	1	1

Branden (2003) to determine the ideal number of PLS components. The objective of PLS is to choose only the PLS components with most variation in the dataset. The optimal number of PLS, *k* is determined based on the Scree plot, i.e., the plot of the number of PLS components against the RMSEP values. The values of RMSEP for different values of *k* are computed. Following Nengsih et al. (2019), ten PLS components are considered. The values of RMSEP will tend to decrease as the number of *k* increases. The optimal value of PLS component is chosen where the value of RMSEP first becomes reasonably stable.

To decide the number of components, *a,* for the biscuit data, first, split the data into training set and test set. The training set is used to construct the estimated model based on *a* components, while the test set is utilised for validation. RMSEP is calculated to validate the estimated model. The RMSEP for ten components is provided in Table 4. The RMD-MRCD-PCA-LOOCV and RMD-MRCD-PCA-10-FoldCV demonstrate a sharp decline from component 2 to component 3 and the values of RMSEP do not show a substantial difference after component 3.

Figures 1 and 2 illustrate the scree plot with a severe decrease in the line at component 3, and after component 3, the graph line exhibits no changes and is practically flat. Hubert and Branden (2003) discovered that using their proposed robust SIMPLS method, three components was the best fit for biscuit dough dataset. While the RDCV shows a significant drop in component 2. As can be seen in Figure 3, the sharp elbow occurs at component 2 and then the line stays horizontal. Similarly, the classical LOOCV and 10-FoldCV methods drop drastically at component 2. The scree plots in Figures 4 and 5 clearly display the sharp point at component 2 for LOOCV and 10-FoldCV. The Classical methods consistently exhibit underfitting by selecting fewer components than our proposed methods. As a result, the findings from simulations that demonstrate the underfitting of the classical methods are consistent with the findings from real data.

TABLE 4. The RMSEP biscuit dough data set

	Cross-validated									
	Comp1	Comp 2	Comp 3	Comp 4	Comp 5	Comp 6	Comp 7	Comp 8	Comp 9	Comp 10
MRCD-PCA-LOOCV	2.370	2.267	1.238	1.211	1.187	1.113	1.121	1.113	1.170	1.181
MRCD-PCA-10-FoldCV	1.616	1.364	0.912	0.624	0.576	0.598	0.525	0.454	0.468	0.483
RDCV	0.5642	0.0116	0.0039	0.0018	0.0015	0.0010	0.000	0.000	0.000	0.000
LOOCV	0.442	0.022	0.004	0.003	0.002	0.000	0.000	0.000	0.000	0.000
10-FoldCV	0.435	0.024	0.004	0.003	0.002	0.000	0.000	0.000	0.000	0.000

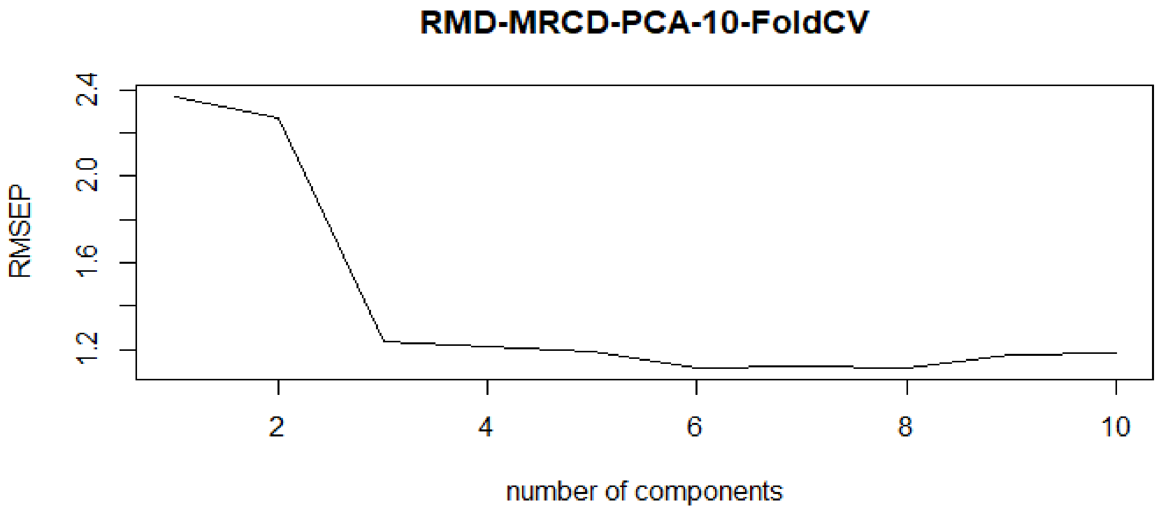


FIGURE 1. Scree plot of RMD-MRCD-PCA-10FoldCV for the biscuit dough data set

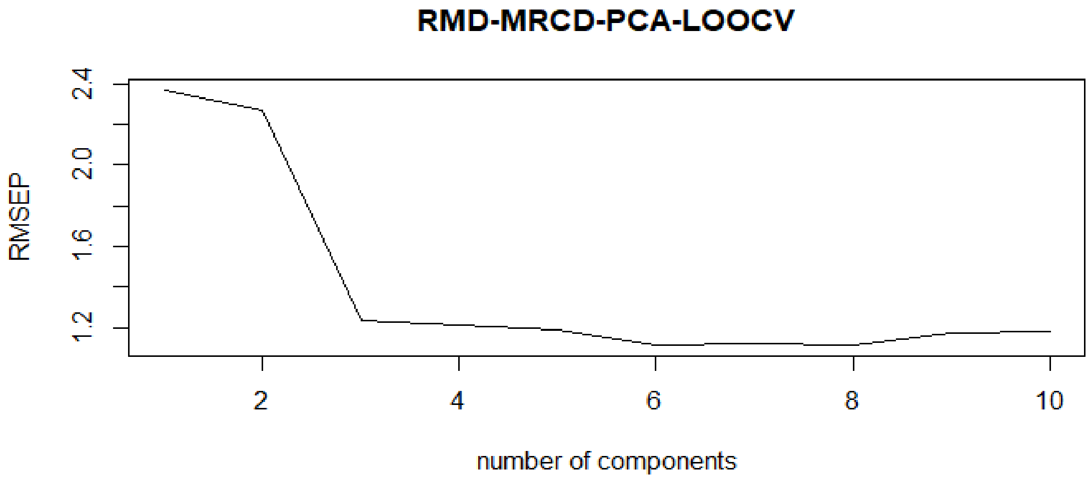


FIGURE 2. Scree plot of RMD-MRCD-PCA-LOOCV for the biscuit dough data set

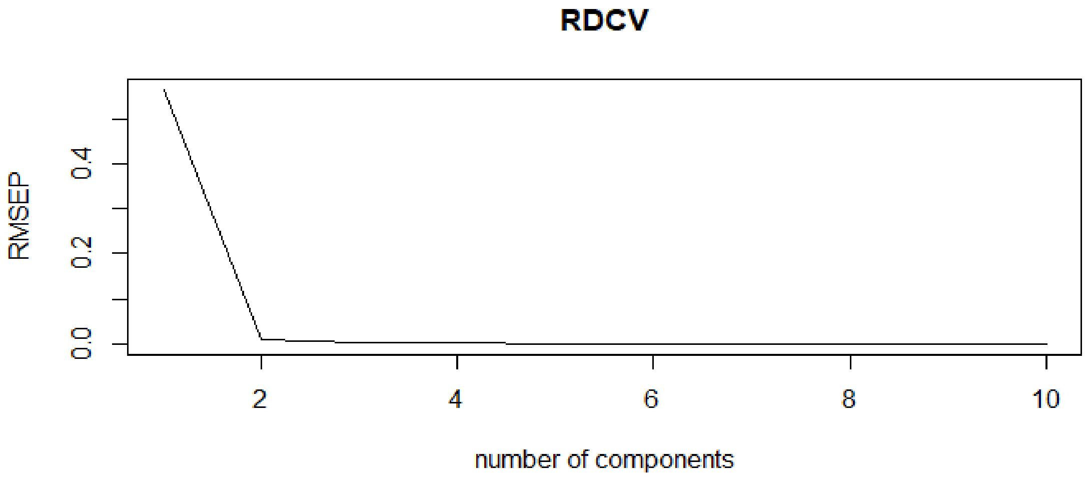


FIGURE 3. Scree plot of RDCV for the biscuit dough data set

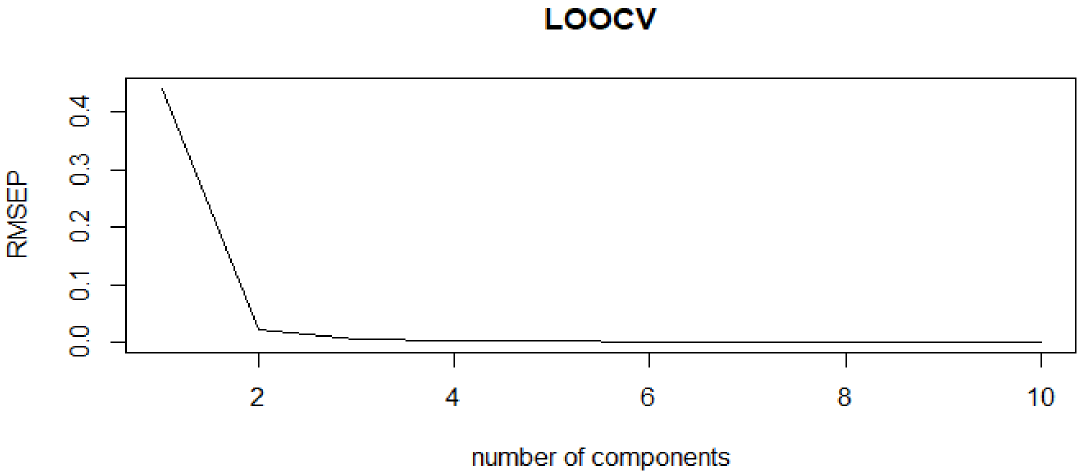


FIGURE 4. Scree plot of LOOCV for the biscuit dough data set

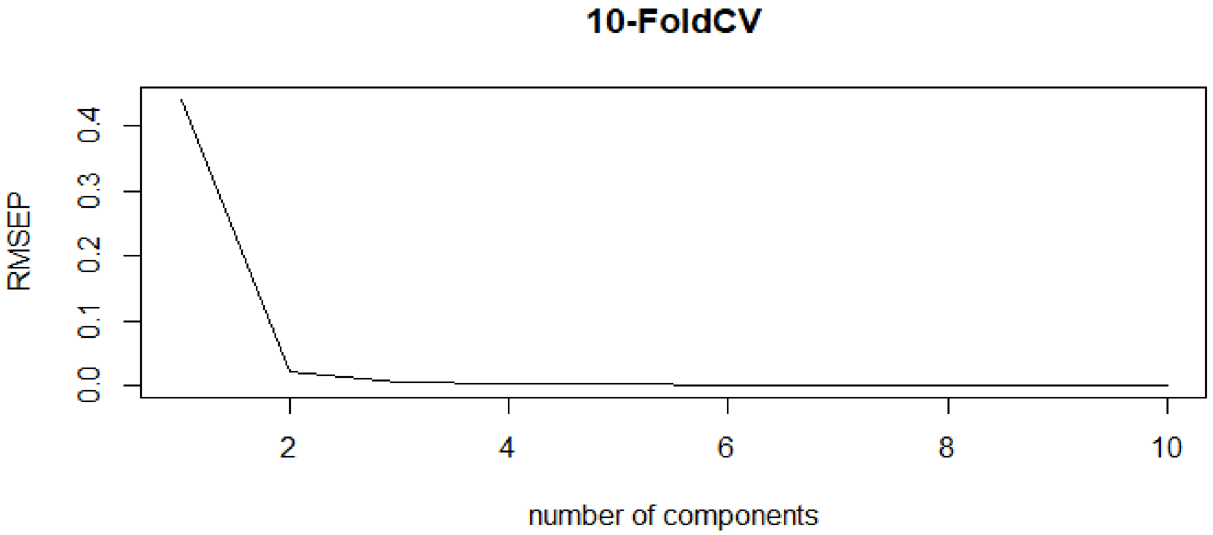


FIGURE 5. Scree plot of 10-FoldCV for the biscuit dough data set

CONCLUSIONS

The primary goal of this research was to develop more reliable cross-validation procedures to determine the optimal number of PLS components for constructing effective PLS regression models. We proposed two robust methods, RMD-MRCD-PCA-LOOCV and RMD-MRCD-PCA-K-FoldCV that incorporate the highly efficient MRCD-PCA covariance estimator and a weighting function to reduce the influence of high-leverage points (HLPs). Unlike conventional approaches, our methods avoid the use of non-robust estimators in calculating the root mean square error of prediction (RMSEP), making them more resilient to outliers. Our simulation studies demonstrate that the proposed methods outperform existing ones (RDCV, LOOCV, and K-FoldCV) by consistently identifying the correct number of components, even in the presence of HLPs. In real data, analysis using the Biscuit Dough dataset, our methods also selected three components, supported by the scree plots and stabilized RMSEP values after component 3. This result is consistent with the findings of Hubert and Branden (2003), who also identified three components as optimal using a robust PLS approach. In contrast, classical methods selected only two components, indicating underfitting and a lack of robustness to influential observations. Overall, the real data results confirm the simulation findings and highlight the practical value of the proposed methods in selecting an appropriate number of PLS components, especially in datasets that may contain outliers or leverage points.

REFERENCES

Abdullah Mohammed Rashid & Habshah Midi. 2023. Improved *nu*-support vector regression algoritm based on the principal component analysis. *Economic Computation and Economic Cybernetics Studies and Research* 57(2): 41-56. https://doi.org/10.24818/18423264/57.2.23.03

Abdullah Mohammed Rashid, Habshah Midi, Waleed Dhhan & Jayanthi Arasan. 2021. Detection of outliers in high-dimensional data using *nu*-support vector regression. *Journal of Applied Statistics* 49(10): 2550-2569. https://doi.org/10.1080/02664763.2021.1911965

Ali Mohammed Baba, Habshah Midi & Nur Haizum Abd Rahman. 2022. Spatial outlier accommodation using a spatial variance shift outlier model. *Mathematics* 10(17): 3182. https://doi.org/10.3390/math10173182

Boudt, K., Rousseeuw, P.J., Vanduffel, S. & Verdonck, T. 2018. The minimum regularized covariance determinant estimator. *Statistics and Computing* 30: 113-128. https://doi.org/10.1007/s11222-019-09869-x

Coakley, C.W. & Hettmansperger, T.P. 1993. A bounded influence, high breakdown, efficient regression estimator. *Journal of the American Statistical Association* 88(423): 872-880. https://doi.org/10.1080/01621459.1993.10476352

Filzmoser, P., Liebmann, B. & Varmuza, K. 2009. Repeated double cross validation. *Journal of Chemometrics* 23(4): 160-171. https://doi.org/10.1002/cem.1225

Geisser, S. 1975. The predictive sample reuse method with applications. *Journal of the American Statistical Association* 70(350): 320-328. https://doi.org/10.1080/01621459.1975.10479865

Habshah Midi, Jaaz Suhaiza, Mohd Aslam, Hani Syahida & Emi Amielda. 2025. Improved robust principal component analysis based on minimum regularized covariance determinant for the detection of high leverage points in high dimensional data. *Sains Malaysiana* 54(8): 2087-2097.

Habshah Midi, Shelan Saied Ismaeel, Jayanthi Arasan & Mohammed A Mohammed. 2021. Simple and fast generalized - M (GM) estimator and its application to real data. *Sains Malaysiana* 50(3): 859-867.

Hubert, M. & Branden, K.V. 2003. Robust methods for partial least square regression. *Journal of Chemometrics* 17(10): 537-549.

Li, B., Morris, J. & Martin, E.B. 2002. Model selection for partial least squares regression. *Chemometrics Intell. Lab. Syst.* 64(1): 79-89. https://doi.org/10.1016/S0169-7439(02)00051-5

Mosteller, F. & Wallace, D.L. 1963. Inference in an authorship problem. *Journal of the American Statistical Association* 58(302): 275-309. https://doi.org/10.4135/9781412961288.n9

Nengsih, T.A., Bertrand, F., Maumy-Bertrand, M. & Meyer, N. 2019. Determining the number of components in PLS regression on incomplete data set. *Statistical Applications in Genetics and Molecular Biology* 18(6):/j/sagmb.2019.18.issue-6/sagmb-2018-0059/sagmb-2018-0059.xml. https://doi.org/10.1515/sagmb-2018-0059

Rousseeuw, P.J. & van Zomeren, B.C. 1990. Unmasking multivariate outliers and leverage points. *Journal of the American Statistical Association* 85(411): 633-639. https://doi.org/10.1080/01621459.1990.10474920

Siti Zahariah & Habshah Midi. 2022. Minimum regularized covariance determinant and principal component analysis-based method for the identification of high leverage points in high dimensional sparse data. *Journal of Applied Statistics* 50(13): 2817-2835. https://doi.org/10.1080/02664763.2022.2093842

Waleed Dhhan, Sohel Rana & Habshah Midi. 2016. A high breakdown, high efficiency and bounded influence modified GM estimator based on support vector regression. *Journal of Applied Statistics* 44(4): 700-714. https://doi.org/10.1080/02664763.2016.1182133

Xu, Qing Song & Yi Zeng Liang. 2000. "Monte Carlo Cross Validation." *Chemometrics and Intelligent Laboratory Systems* 56(1): 1–11. https://doi.org/10.1016/S0169-7439(00)00122-2.

*Corresponding author; email: sitizahariah@unikl.edu.my