Robust Partial Least Squares For Regression and Classification

Except where reference is made to the work of others, the work described in this dissertation is my own or was done in collaboration with my advisory committee. This dissertation does not include proprietary or classified information.

Asuman Seda Turkmen

Certificate of Approval:

Asheber Abebe
Associate Professor
Mathematics and Statistics

Nedret Billor, Chair
Associate Professor
Mathematics and Statistics

Mark Carpenter
Associate Professor
Mathematics and Statistics

Chris Rodger
Professor
Mathematics and Statistics

George T. Flowers
Interim Dean
Graduate School

Robust Partial Least Squares For Regression and Classification

Asuman Seda Turkmen

A Dissertation

Submitted to

the Graduate Faculty of

Auburn University

in Partial Fulfillment of the

Requirements for the

Degree of

Doctor of Philosophy

Auburn, Alabama
August 9, 2008

Robust Partial Least Squares For Regression and Classification

Asuman Seda Turkmen

Signature of Author

Date of Graduation

Vita

Asuman Seda Türkmen, daughter of Ümit Peyda Topçubaşı and Vicdan Topçubaşı, was born July 24, 1977, in Adana, Turkey. She graduated with honors from Çukurova University in 1999 with a Bachelor of Science degree in Mathematics. After graduation, she continued her graduate studies in the same department from which she received her Master of Science degree in Statistics in 2001. She came to Auburn University in August 2003 and she entered the doctoral program in the Department of Mathematics and Statistics where she completed her Ph.D. in Statistics in 2008.

Dissertation Abstract

Robust Partial Least Squares For Regression and Classification

Asuman Seda Turkmen

Doctor of Philosophy, August 9, 2008
(M.A., Çukurova University, 2002)
(B.S., Çukurova University, 1999)

133 Typed Pages

Directed by Nedret Billor

Partial Least Squares (PLS) is a class of methods for modeling relations between sets of observed variables by means of latent variables where the explanatory variables are highly collinear and where they outnumber the observations. In general, PLS methods aim to derive orthogonal components using the cross-covariance matrix between the response variable(s) and the explanatory variables, a quantity that is known to be affected by unusual observations (outliers) in the data set. In this study, robustified versions of PLS methods, for regression and classification, are introduced.

For regression with quantitative response, a robust PLS regression method (RoPLS), based on weights calculated by BACON or PCOUT algorithm, is proposed. A robust criteria is suggested to determine the optimal number of PLS components which is an important issue in building a PLS regression model. In addition, diagnostic plots are constructed to visualize and classify outliers. Robustness of the proposed method, RoPLS, is studied in detail. Influence function for the RoPLS estimator is derived for low dimensional data and empirical robustness properties are provided for high dimensional data.

PLS was originally designed for regression problems with quantitative response, however, it is also used as a classification technique where the response variable is qualitative. Although several robust PLS methods have been proposed for regression problems, to our knowledge, there has been no study on the robustness of the PLS classification methods. In this study, the effect of outliers on existing PLS classification methods is investigated and a new robust PLS algorithm (RoCPLS) for classification is introduced.

The performances of the proposed methods, RoPLS and RoCPLS, are being assessed by employing several benchmark data sets and extensive simulation experiments.

Acknowledgments

I would like to express my sincere appreciation to my advisor and mentor, Dr. Nedret Billor, who has always provided her enthusiastic assistance, persistent encouragement and help throughout my graduate education. Words cannot express how thankful I am for her support in every aspect of my academic development. Without her and her continuous guidance, I never would have seen the end of this study.

Special thanks go to Dr. Asheber Abebe for his valuable suggestions and opinions. I am also grateful to Dr. Mark Carpenter and Dr. Chris Rodger for kindly consenting to serve on the dissertation committee. Much-deserved gratitude is expressed to my parents and family for their endless love and support throughout my life. I am grateful for the loving kindness of my husband, Özgür Türkmen, for his patience, love, and support; he has been my true joy in this journey together. Finally, my sincere appreciation is also extended to all of the people, who have always been there for me whenever I needed them.

I dedicate this dissertation to the memory of my grandparents: Güzide & Mustafa Topçubaşı, Rabia & Lokman Karagöl who truly believed in me.

Style manual or journal used Journal of Approximation Theory (together with the style known as “aums”). Bibliograpy follows van Leunen’s *A Handbook for Scholars.*

Computer software used The document preparation package TEX (specifically LATEX) together with the departmental style-file `aums.sty`.

Table of Contents

List of Figures

Chapter 1

Introduction

Most traditional statistical techniques are especially designed for low dimensional data sets where the number of observations (n) is greater than the number of variables (p). Application of the statistical methods for problems such as, survival time or tumor class prediction of a patient, based on a high dimensional data ($n << p$), is a difficult and challenging task. On the other hand, nowadays data sets in many scientific fields are high dimensional because of the fact that advances in technology have made simultaneous monitoring of thousands of features (variables) possible. Therefore, analyzing such data sets has been a focus for many researchers in a wide range of scientific fields due to the requirement of resolutions for various statistical problems that have emerged.

Recently, partial least squares (PLS) has become an important statistical tool for modeling relations between sets of observed variables by means of latent variables especially for statistical problems dealing with high dimensional data sets. PLS is a member of nonlinear iterative least squares (NILES) procedures developed by Wold ([87], [88]). The use of PLS methods for regression problems began in the early 80's. The main idea in PLS regression (PLSR) is to summarize high dimensional and/or collinear explanatory variables into a smaller set of uncorrelated, so called latent variables, which have the "*best*" predictive power.

Although PLSR was initially developed for social and economic science problems having scarce information, it has received a great amount of attention in the chemometrics literature. The main application of PLSR in chemometrics is the prediction of constituent concentrations of a sample based on its spectrum obtained by spectroscopic techniques, such as near infrared (NIR) spectroscopy, energy-dispersive X-ray fluorescence spectroscopy, and ultraviolet(UV) spectroscopy. Spectral data contain a large amount of information since a spectrum typically ranges over a large number of wavelengths (variables) with limited number of concentrations (observations). The spectroscopic techniques, in combination with PLSR analysis, have proved to be a powerful analytical tool for analyzing on-line industrial processes. Its speed, relative good performance, and ability to handle data sets with more variables than observations resulted in a lot of applications of PLSR in many other scientific areas such as bioinformatics, food research, medicine, and pharmacology. For instance, in the area of drug design, a large amount of chemicals need to be evaluated for their toxicity and effectiveness before they are used by pharmacists. *Quantitative Structure-Activity Relationships* (QSAR) analysis employs theoretical molecular descriptors for reliable estimates of the toxic and therapeutic potential of chemicals. QSAR data sets contain some degree of multicollinearity which can be handled by using PLSR. Beside this, because of the large time scale of the data collection process, QSAR data often contain outliers.

Despite of the fact that PLSR handles the multicollinearity problem, it fails to deal with data containing outliers since it is based on maximizing the sample covariance matrix between the response(s) and a set of explanatory variables, a statistic that is known to be sensitive to outliers. Existence of multicollinearity and outliers is no exception in real data sets, and it leads to a requirement of robust PLSR methods ([13], [35], [38], [48], [76], [83]) in chemometrics as well as other application areas.

The problem of classifying entities into one of several groups has been another important goal in many scientific investigations. It is important that this activity is done in a manner that minimizes the misclassification error rate. High dimensionality and collinearity make the application of most statistical classification methods difficult, or even impossible, for some cases. The procedure for classification of high dimensional data often consists of two steps: the first step is to construct a few components from a large number of explanatory variables by using dimension reduction techniques, and the second step is to employ classical classification methods on the constructed components. Although PLS was originally designed for regression problems, it has started to be used frequently as a dimension reduction tool for classification problems and recent studies have showed that classification via PLS performs quite well ([4], [8], [60], [65], [66]) especially for microarray data analysis. An important application of microarray technology is tumor diagnosis. A reliable and precise classification of tumors is potentially life-saving and hence is essential to physicians. In the presence of outliers, e.g. tissue-specific genes whose expression profile is considerably different (could be "erroneous" or "genuine") in particular tissue(s) than in others, dimension reduction via PLS would yield unreliable results since PLS is known to be sensitive to outliers. Although several robust PLS methods have been proposed when the response variable is quantitative, to our knowledge, there has been no study on the robustness of PLS when the response variable is qualitative.

The main contribution of this work is the construction of robust algorithms for partial least squares methods. The chapters are organized as follows: in Chapter 2, a detailed literature review on PLSR, that includes the existing PLS algorithms; asymptotic variance, consistency, geometric and, peculiar shrinkage properties of PLSR estimator; and the relationship between PLSR and other biased estimation methods such as principal component regression, ridge regression and continuum regression, is given. A robust PLSR method

(RoPLS) is introduced in Chapter 3 and its robustness properties are explored in Chapter 4. The effect of outliers on existing PLS classification methods is investigated and a new robust PLS algorithm for classification (RoCPLS) is proposed in Chapter 5. Finally, conclusions and proposed future work are given in Chapter 6. A recapitulation of the notations that are used throughout the work can be found in Appendix.

Chapter 2

An Overview on Partial Least Squares

2.1 Introduction

The standard multiple regression model defined by the equation

$$y = X\beta + \varepsilon \tag{2.1}$$

where X is a $n \times p$ matrix of explanatory variables (predictors), y is a $n \times 1$ vector of response variable, β is a $p \times 1$ vector of unknown parameters, and ε is a $n \times 1$ vector of error terms whose rows are identically and independently distributed.

In multivariate models, the response vector y in the model (2.1) is replaced by the $n \times g$ response matrix, Y, where $g \geq 2$. Throughout this study, although PLS algorithms are given for multivariate models in general, only multiple univariate regression model is considered and emphasized due to its simplicity. Furthermore, multiple univariate models always give better results than multivariate models in terms of the variance explained ([9], [31], [34], [59]) as long as response variables are unrelated.

The outline of this chapter is as follows. In Section 2.2 most commonly used PLS algorithms, NIPALS and SIMPLS, are described. Popular approaches for determining the optimal number of components are discussed in Section 2.3. The relationship between PLSR and other biased regression techniques is summarized in Section 2.4, followed by statistical properties of PLSR estimator given in Section 2.5.

2.2 Partial Least Squares Regression (PLSR)

The ordinary least squares (OLS) estimator of β, $\widehat{\beta}_{OLS}$, in the model given by (2.1) is the solution of the following optimization problem:

$$\widehat{\beta}_{OLS} = \underset{b}{\operatorname{argmax}}\, corr\{Xb, y\}^2. \tag{2.2}$$

In many applications of multiple regression (e.g., spectral data analysis in chemometrics), multicollinearity is inevitable as a result of large number of variables collected by modern technologies of computers, networks, and sensors. Despite having desirable properties, the OLS estimator can have an extremely large variance and results in imprecise prediction when the data are multicollinear. Moreover, solution of (2.2) is not unique when $n \leq p$.

One solution to deal with multicollinearity and/or dimensionality problem is regressing the response variable y on a subset of the k orthogonal (latent) vectors stored in a score matrix of size $n \times k$ by which important features of X have been retained. Score matrix is formed by taking linear combinations of columns of X. PLS regression (PLSR) constructs the columns of score matrix, $T = [\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_k]$, by solving the following optimization problem for $h = 1, 2, \ldots, k$ $(k \leq p)$:

$$r_h = \underset{\|r\|=1}{\operatorname{argmax}}\, cov(Xr, y)^2 = \underset{\|r\|=1}{\operatorname{argmax}}(r'X'yy'Xr) \tag{2.3}$$

$$\text{subject to } \mathbf{t}_h'\mathbf{t}_j = r_h'X'Xr_j = 0 \text{ for } 1 \leq j < h.$$

So, PLSR balances the maximal correlation criteria for OLS given in (2.2) with the requirement of explaining as much as variability in both X and $y-$space.

Several iterative procedures have been proposed to solve nonlinear optimization problem in (2.3) such as PLS Mode A, PLS-SB, NIPALS and SIMPLS algorithms that differ by the deflation theme required for the orthogonality of derived components. PLS Mode A algorithm ([88]) aims to model existing relationships between variables rather than to model for prediction. PLS-SB computes all eigenvectors at once, and the score vectors obtained by this method are not necessarily orthogonal. The most commonly used methods, NIPALS and SIMPLS, consist of two steps may be called calibration (deriving components) and prediction. These algorithms, for both univariate and multivariate responses, are explained in the following subsections. The extension of two-block PLS model, where X and y (or Y for multivariate model) are block variables, to multi-block PLS model is also given in the literature ([84], [88]) and is not discussed in this study.

2.2.1 NIPALS Algorithm

The NIPALS algorithm ([87]) was developed as an alternative to principal component algorithms. NIPALS employs sequential simple linear regressions instead of singular value decomposition to calculate principal components. PLS algorithm can be considered as carrying out two simultaneous NIPALS principal component analyses, one for X and one for Y, while interchanging the results from X for analysis of Y and vice versa ([50], [34]) to solve the following maximization problem

$$\max_{\|r\|=\|s\|=1} cov(Xr, Ys)^2$$

under the orthogonality constraint of derived components, where $s = 1$ and $Y = y$ for univariate model. Since both X and Y are used in the computation of the components, PLS is presented as a member of the bilinear class of methods and the bilinear model can be written as:

$$X = TP' + E, \tag{2.4}$$
$$Y = UQ' + F. \tag{2.5}$$

The equations given in (2.4) and (2.5) are called *outer relations* where T and U are score matrices derived from X and Y, respectively; P (x-loadings) represents the regression coefficients of X on T; Q (y-loadings) represents the regression coefficients of Y on U; E and F are the matrices of errors. It is assumed that the score matrix T is a good predictor for Y and a linear, *inner relationship* between the score matrices T and U exists, i.e. $U = TB + H$ where B is a $k \times k$ diagonal matrix and H is a matrix of errors. The *mixed relation* then becomes:

$$Y = UQ' + F = (TB + H)Q' + F = TA' + F^* \tag{2.6}$$

where $A' = BQ'$ is a matrix of regression coefficients and $F^* = HQ' + F$ is matrix of errors. For the univariate case ($g = 1$), the matrix B in the model (2.6) becomes identity matrix, so the equation (2.6) represents both *outer* and *mixed relationships* which can be rewritten as

$$y = T\mathbf{a} + f^* \tag{2.7}$$

where $\mathbf{a}$ is a vector of regression coefficients and f^* is a vector of errors. The NIPALS algorithm for univariate response variable (y) based on mixed relationship in (2.7) is called PLS1, whereas NIPALS algorithm for multivariate response variable (Y) is called PLS2. The calibration step of PLS2 algorithm for k component is described as follows:

Algorithm 2.1 *(PLS2)*

Step 1: Let E_0 and F_0 be the copies of X and Y, respectively.

Step 2: For $h = 1, 2, \ldots, k$ do steps 2.1 – 2.4:

Step 2.1: Let $\mathbf{u}_h$ be a column of F_{h-1}, e.g., the one having maximum variance.

Step 2.2: Repeat steps 2.2.1 – 2.2.4 until the convergence of $\mathbf{w}_h$ (or $\mathbf{t}_h$).

Step 2.2.1: Perform the regression $E_{h-1} = u_h w'_h + \varepsilon_1$, yielding the least squares solution

$$w_h = \frac{E'_{h-1} u_h}{u'_h u_h} \tag{2.8}$$

and normalize $w_h := w_h / ||w_h||$.

Step 2.2.2: Perform the regression of $E'_{h-1} = w_h t'_h + \varepsilon_2$ yielding the least squares solution

$$t_h = E_{h-1} w_h.$$

Step 2.2.3: Perform the regression of $F_{h-1} = t_h q'_h + \varepsilon_3$ with the least squares solution

$$q_h = \frac{F'_{h-1} t_h}{t'_h t_h} \tag{2.9}$$

and normalize $q_h := q_h / ||q_h||$.

Step 2.2.4: Perform the regression of $F'_{h-1} = q_h u'_h + \varepsilon_4$ yielding the following solution

$$u_h = F_{h-1} q_h.$$

Step 2.3: Perform the regression of E_{h-1} and F_{h-1} on $\mathbf{t}_h$, separately to compute residuals

$$E_h = E_{h-1} - t_h p'_h$$

$$F_h = F_{h-1} - b_h t_h q'_h$$

where $\mathbf{p}'_h$ is the regression coefficient vector obtained by regressing E_{h-1} on $\mathbf{t}_h$ (outer relation) and b_h is the regression coefficient obtained by regressing $\mathbf{u}_h$ on $\mathbf{t}_h$ (inner relation), i.e.,

$$p_h = E'_{h-1}t_h/t'_ht_h$$
$$b_h = u'_ht_h/t'_ht_h$$

Step 2.4: Store vectors $\mathbf{w}_h$, $\mathbf{t}_h$, and $\mathbf{p}_h$ into matrices $W_h = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_h]$, $T_h = [\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_h]$, and $P_h = [\mathbf{p}_1, \mathbf{p}_1, \ldots, \mathbf{p}_h]$, respectively. Set $h =: h + 1$

For univariate case (PLS1), y and f_0 are used instead of Y and F_0 in the first step of Algorithm 2.1. Steps 2.1 and 2.2 are replaced by 2.2.1 (with $\mathbf{u}_h = f_{h-1}$) and 2.2.2, respectively while the steps 2.3 and 2.4 remain the same where $b_h = 1$. In other words, convergence of $\mathbf{w}_h$ is obtained in the first iteration.

Score matrix, T_k, can be written in terms of linear combinations of the columns of $E_0 = X$, that is $T_k = XR_k$, where the h^{th} column of R_k is called h^{th} PLS-weight vector. The matrix, R_k is related to P_k and W_k stored in NIPALS algorithm, via the formula

$$R_k = W_k(P'_kW_k)^{-1} \tag{2.10}$$

which follows from the fact that R_k and W_k share the same column space and that P'_kR_k is equal to the identity matrix ([41], [45], [55]).

In the prediction step, ordinary least squares estimate for $\mathbf{a}$ in the univariate mixed model given in (2.7) is obtained by regressing the response vector y onto these k components which yields

$$\widehat{a}^{(k)} = (T_k'T_k)^{-1}T_k'y = (R_k'X'XR_k)^{-1}R_k'X'y.$$

Therefore, the PLS estimate of β in the model given in (2.1) is

$$\widehat{\beta}_{PLS}^{(k)} = R_k\widehat{a}^{(k)}.$$

Alternative formulations of the PLS1 are suggested by Helland ([41]) and Garthwaite ([33]). An extensive simulation study by Breiman and Friedman ([9]) on the comparison of univariate and multivariate regression methods including PLS1, PLS2, OLS, and other biased regression methods demonstrated that performing separate PLS1 regressions on each individual response would be a better strategy than employing PLS2 (see also [31], [34], and [59]).

The major drawback of NIPALS algorithm (PLS1 and PLS2) is that the columns of score matrix, T, are obtained as linear combinations of deflated data matrix X. Since one looses sight of what is in the depleted data, the interpretation of the components gets complicated. SIMPLS algorithm, given in the next subsection, resolves this drawback by using a different deflating scheme.

2.2.2 SIMPLS Algorithm

SIMPLS algorithm ([15]) is an alternative to NIPALS algorithm that aims to derive PLS components directly in terms of the original data which results in faster computation with less memory requirements and interpretation easiness. SIMPLS deflates the cross-covariance matrix, $S_{xy} \propto X'Y$, whereas NIPALS deflates the original data matrix X to obtain orthogonal components.

SIMPLS algorithm can be summarized as follows:

Algorithm 2.2 *(SIMPLS)*

Step 1: Compute cross-product matrix: $S_{xy}^{0} = X'Y$ (X and Y are centered),

Step 2: Repeat steps 2.1 − 2.6 for $h = 1, 2, \ldots, k$:

Step 2.1: Compute first left singular vector of S_{xy}^{h-1} as h^{th} PLS weight vector r_h,

Step 2.2: Compute h^{th} score, $t_h = Xr_h$, and normalize $t_h =: t_h/\|t_h\|$,

Step 2.3: Update h^{th} PLS weight, $r_h =: r_h/\sqrt{r_h' X' X r_h}$,

Step 2.4: Compute h^{th} x-loading by regressing X on t_h: $p_h = X' t_h$,

Step 2.5: Store vectors $\mathbf{r}_h$, $\mathbf{t}_h$, and $\mathbf{p}_h$ into matrices $R_h = [\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_h]$, $T_h = [\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_h]$, and $P_h = [\mathbf{p}_1, \mathbf{p}_1, \ldots, \mathbf{p}_h]$, respectively.

Step 2.6: $h =: h + 1$ and $S_{xy}^{h-1} = (I_p - V_{h-1}V_{h-1}')X'y$ where columns of V_{h-1} form an orthonormal basis for P_{h-1}.

The orthogonality constraint of components is fulfilled when the PLS weight vector $\mathbf{r}_h$ is orthogonal to all previous x-loadings $P_{h-1} = [\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_{h-1}]$. As a result of this, the h^{th} pair of SIMPLS weight vector $\mathbf{r}_h$ for $h = 2, \ldots, k$ is obtained as the first left singular vector of ${S_{xy}}^{h-1}$ which is projection of ${S_{xy}}^{h-2}$ on a subspace orthogonal to P_{h-1}. Therefore, if the columns of $V_{h-1} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_{h-1}]$ form an orthonormal basis of P_{h-1} obtained by GramSchmidt method, then

$$ {S_{xy}}^{h-1} = (I_p - v_{h-1}v_{h-1}'){S_{xy}}^{h-2} = (I_p - V_{h-1}V_{h-1}')X'Y. \tag{2.11} $$

Once k components are derived, PLS estimate is obtained by using score matrix as explanatory variables as in the prediction step described in NIPALS algorithm.

After h components are derived, the data matrix is reduced implicitly to $X(I_p - V_h V_h')$ with SIMPLS algorithm which can be seen from (2.11). In PLS1 algorithm, the h^{th} derived component, $\mathbf{t}_h$, is equal to $E_{h-1}w_h$, where $\mathbf{w}_h$ is the normalized form of:

$$E_{h-1}'f_{h-1} = X'(I_n - T_{h-1}(T_{h-1}'T_{h-1})^{-1}T_{h-1}')^2 y = X'(I_n - T_{h-1}(T_{h-1}'T_{h-1})^{-1}T_{h-1}')y. \tag{2.12}$$

Therefore, data matrix is reduced explicitly to $(I_n - T_h(T_h'T_h)^{-1}T_h')X$ with PLS1. Although residual matrices differ, application of both algorithms on data sets demonstrated that SIMPLS and NIPALS algorithms are equivalent for univariate case which is also stated in the next proposition.

Proposition 2.1 *(De Jong, [15]) SIMPLS is equivalent to PLS1.*

Proposition 2.1 can be proven by induction. However, when applied to multivariate set of response variables ($g > 1$), the SIMPLS results are different from the results of PLS2 ([15]). In this study, SIMPLS algorithm is employed because of its speed and efficiency.

2.3 Determining the Optimal Number of Components in PLSR

The decision on the optimal number of components, k, is a very important issue in building the PLSR model. Although, it is possible to calculate as many components as the rank of the X, it does not make sense in practice. Because data are never noise-free and some of the smaller components will only describe noise. Due to uncertain statistical behavior of PLSR, explained in Section 2.5, it is difficult to perform inferential tasks such as assessing the number of components. Consequently, developing as well as comparing PLS component selection rules have been and apparently continue to be subjects of active research in chemometrics. Cross validation, adjusted Wold's criterion and randomization test are leading methods that are proposed to seek out the optimum dimensionality of PLS models.

Among the many approaches proposed in the past, the cross-validation (CV) scheme stands out in particular. In M-fold cross-validation, the original sample is partitioned into M sub-samples. Of the M subsamples, a single sub-sample is retained as the *validation set* for testing the model, and the remaining $M-1$ sub-samples are used as *learning set* for estimating the model. The cross-validation process is then repeated M times (number of folds), with each of M sub-samples used exactly once as the validation set. The M results from the folds then can be combined to produce a single estimate for the optimal number of components. Particularly, the n-fold cross validation ($M = n$), where only one observation is deleted and the process is repeated as many times as samples, is called leave-one-out cross validation. The resulting residual sum of squares, PRESS, is a measure of the predictive power of the components in the model. The PRESS value for h component univariate PLSR using leave-one-out cross validation is:

$$PRESS^{(h)} = \sum_{i=1}^{n} \left(y_i - \widehat{y}_{-i(h)} \right)^2 \tag{2.13}$$

where the predicted values $\widehat{y}_{-i(h)}$ are based on the parameter estimates that are obtained from the data set which does not include observation i using a PLSR model with h components. The optimal number of components is the one that yields the minimum PRESS or root mean square error, RMSE,

$$k = \underset{h}{\operatorname{argmin}}\{PRESS^{(h)}\} = \underset{h}{\operatorname{argmin}}\{RMSE^{(h)}\} \tag{2.14}$$

where

$$RMSE_{(h)} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \widehat{y}_{-i(h)}\right)^2} = \sqrt{\frac{1}{n}PRESS_{(h)}}. \tag{2.15}$$

A simple and classical method is the Wold's R criterion ([89]) which compares two successive values of PRESS via their proportion, that is

$$R = \frac{PRESS^{(h+1)}}{PRESS^{(h)}} \tag{2.16}$$

where $PRESS^{(h)}$ is given in equation (2.13). When R is greater than 1, it is considered that the optimal number of components is h. Instead of comparing this ratio to unity, it was proposed by ([89]) to fix it at 0.90 or 0.95 which is named *Adjusted Wold's Criteria.*

The randomization test ([85]) is a recent method that assesses the statistical significance of each individual component that enters the model. Theoretical approaches to achieve this goal (using a t- or F-test) have been put forth, but they are all based on some assumptions. Randomization test is a data-driven approach and therefore ideally suited for avoiding assumptions.

Denham ([19]) evaluated performances of several mean squared error (MSE) estimation approaches in terms of their accuracy and usefulness in determining the optimal number of components to include PLSR model. It is concluded that all methods perform very compatible for data sets with few variables, while the cross-validation method results in better MSE estimates for the data sets with large number of variables. One area where the method of cross-validation works poorly is design of experiments, where the randomization test should have merit ([85]).

2.4 PLS Regression Among Other Biased Methods

In the multiple linear regression, the OLS estimator of the regression coefficient vector has minimum variance in the class of unbiased estimators. Existence of multicollinearity problem results in large variances of the coefficient estimators. Therefore, several biased estimation methods have been proposed as alternatives to OLS estimator when multicollinearity is present. The main goal of biased methods is to decrease the mean squared error of prediction by introducing a reasonable amount of bias into the model. This is done by shrinking the solution coefficient vector away from the OLS solution toward directions in which the projected data have larger spread. PLSR regression is a biased regression method and it is related to other biased methods such as principal component regression (PCR, [62]) and ridge regression (RR, [44]).

Most of these methods can be unified under a generalized approach called continuum regression. The continuum regression (CR) is a technique that can generate a range of models including OLS, PLSR and PCR. CR weight vectors r_h for $h = 1, 2, \ldots, k$ are defined as proposed by Stone and Brooks ([80]), according to the criterion:

$$r_h = \underset{\|r\|=1}{\operatorname{argmax}}\, cov(Xr, y)^2 [Var(Xr)]^{\frac{\delta}{1-\delta}-1} = \underset{\|r\|=1}{\operatorname{argmax}}\, (r'X'y)^2 (r'X'Xr)^{\frac{\delta}{1-\delta}-1} \tag{2.17}$$

under the constraint that $cov(Xr_h, Xr_j) = 0$ for $h > j$. The parameter δ $(0 \leq \delta \leq 1)$ adjusts the amount of information of the regressors to be considered for predicting the response variable.

The single vector ($h = 1$) that maximizes the squared sample correlation between the response and the corresponding linear combination of the predictor variables is the OLS solution which is obtained by taking $\delta = 0$ in equation (2.17). Similarly, for $h = k$, δ=0.5 and δ=1 yield PLSR and PCR based solutions on k components, respectively.

Alternatively, the OLS estimator can be obtained as the solution of the normal equations

$$X'Xb = X'y. \tag{2.18}$$

Through this section, it is assumed that $n > p$ and X has full-column rank, i.e. $rank(X) = p$, however, with some minor modifications, the results given in this section can be established for other cases, as well. Then, the OLS estimator of β is

$$\widehat{\beta}_{OLS} = (X'X)^{-1}X'y. \tag{2.19}$$

If $rank(X) < p$, then $(X'X)^{-1}$ in (2.19) is replaced by $(X'X)^{+}$ which yields unique minimum length least squares (MLLS) solution.

The idea of PCR is to replace the original regressors by $h \leq p$ principal components (PCs), stored in the score matrix, $Z_h = XV_h$, where the first h eigenvectors of $X'X$ form V_h. These eigenvectors are the solutions of (2.17) for δ=1. Therefore, the PCR estimator of β in (2.1) based on h components is:

$$\hat{\beta}^{(h)}_{PCR} = (Z'_h Z_h)^{-1} Z'_h y = V_h V'_h \hat{\beta}_{OLS}. \tag{2.20}$$

It can be seen from (2.20) that the PCR estimator based on h components is the orthogonal projection of OLS estimator onto the space spanned by the first h eigenvectors of $X'X$, since $V_hV_h' = V_h(V_h'V_h)^{-1}V_h$. On the other hand, the PLSR estimator of β for h components is given by

$$\hat{\beta}_{PLS}^{(h)} = R_hP_h'\hat{\beta}_{OLS} = W_h(P_h'W_h)^{-1}P_h'\hat{\beta}_{OLS}. \tag{2.21}$$

The matrix $W_h(P_h'W_h)^{-1}P_h'$ is an idempotent matrix, hence it is a projection matrix. However, it is not a symmetric matrix, so it is called *oblique projector*. Therefore, the PLSR estimator for h components is the oblique projection of $\hat{\beta}_{OLS}$ onto W_h along to the space $P_h^{\perp}$ ([69]). In PCR, as well as in PLSR, the degree of bias is controlled by the dimension of the space , h, on which orthogonal projection of $\hat{\beta}_{OLS}$ is taken. The smaller the value of h, the larger the bias.

In PCR, the columns of score matrix are derived without the reference to the response variables so that the derived components are optimal in the sense of maximizing the amount of explained variation in X. On the other hand, in PLSR, a set of linear combinations for X and another set of linear combination for y are derived and they are optimally related in yet another sense. This is an advantage of PLSR over PCR especially in the cases, where components obtained by maximizing variation in X, may have no relevance for modeling y which is demonstrated in the next example.

Example 2.1

Hald's data set ([23]), consists of four standardized regressors and one response variable, is used to demonstrate this drawback of PCR. The original response variable is replaced by $y = 2X\mathbf{v}_4 + \varepsilon$, where $\varepsilon \sim N(0,1)$ and $\mathbf{v}_4$ is the eigenvector corresponding to the smallest eigenvalue of $X'X$, as suggested by Hadi et al. ([39]). It can be seen from Figure 2.1 that scatter plots of y versus each of the first three PCs display completely random pattern, while the relationship between y and the last PC is perfectly linear. PCR and PLSR based on $k = 2$ components resulted in mean squared errors 5.3228 and 0.2277, respectively which also can be concluded from Figure 2.2. In general, if the true regression coefficient is in the direction of i^{th} eigenvector of $X'X$, then i^{th} component alone will contribute everything to fit, while the remaining PCs contribute nothing ([39]). In such cases, PLSR is expected to perform better than PCR since optimal directions are determined by considering y.

Another advantage of PLSR over PCR is that the vector of fitted values from PLSR is closer to fitted values from OLS and hence to y than its PCR counterpart. The PLS model always gives a closer fit, in terms of coefficient determination, R^2, than a PCR model with the same number of components ([16], [31], [70]).

Ridge regression (RR) is another well-known biased regression method. The method replaces the covariance matrix, $X'X$, by a better conditioned matrix, $X'X + \xi I_p$ for a value of ξ called *ridge constant* that lies between 0 and 1. The aim is stabilizing the inverse of the possibly ill-conditioned covariance matrix by adding a multiple of I_p. As in OLS, the solution is defined by a single vector

$$\hat{\beta}_{RR} = \underset{\|w\|=1}{\operatorname{argmax}}\, corr(Xw, y)^2 \frac{Var(Xw)}{Var(Xw) + \xi} = (X'X + \xi I_p)^{-1} X'y. \tag{2.22}$$

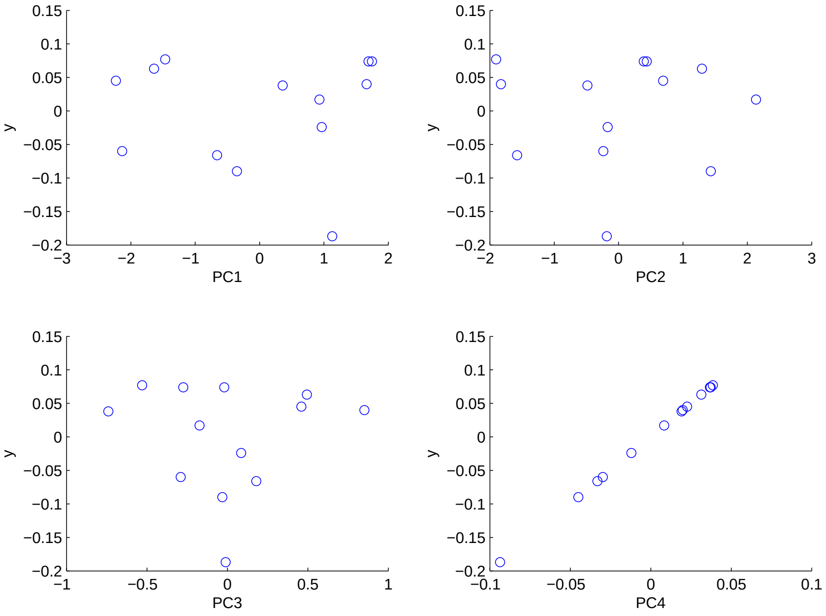


Figure 2.1: Scatter plots of y versus PC1, PC2, PC3, PC4.

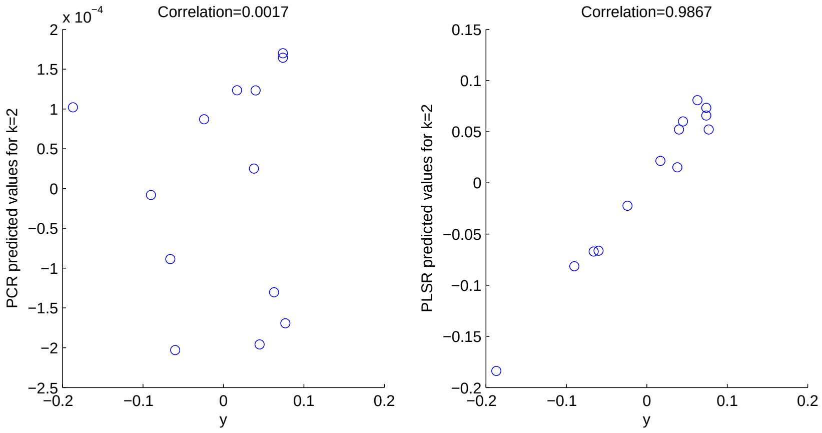


Figure 2.2: Scatter plots of $\hat{y}$ versus y using PCR with k=2 (left) and PLSR with k=2 (right).

Setting the $\xi = 0$ yields the unbiased OLS solution, while the larger values of ξ introduces bias into the model. The relationship between the first factor of CR and RR is described in [81] and it is concluded that there is one-to-one correspondence between the δ (CR parameter) and ξ ([81]).

RR differs in two respects from the PLSR. First of all, it does not derive orthogonal components, it applies explicit shrinkage to the coefficient vector. Secondly, RR is usually applied to univariate regression models, although the generalized RR for multivariate response model is proposed. One of the disadvantages of RR is its heavier computation especially for high dimensional problems.

The comparison of OLS, PLSR, PCR, and RR is given by Frank and Friedman ([31]) and Almåy ([1]). Simulation studies indicated that none of the methods is uniformly better than the others. Frank and Friedman concluded that RR, PLSR and PCR provided substantial improvement over OLS when multicollinearity exists. In all settings, PLSR usually performed very compatible with RR and it was closely followed by PCR. The main result of Almåy's study was that PLSR performs better when the irrelevant eigenvalues are large, whereas PCR performs better when the irrelevant eigenvalues are small.

2.5 Statistical Properties of the PLSR Estimator

Since the PLSR estimator of β is a non-linear function of y, it is very difficult to derive the exact distribution of the estimator which leads difficulties in terms of inference based tasks. Although PLSR is a very popular tool for chemometricians, it is used to be overlooked by statisticians. However, more recently statisticians have attempted to shed some light on the method and its properties. In this section, shrinkage structure of the PLSR estimator

([17], [31], [53], [70]), its asymptotic variance ([18], [71], [75]) and consistency ([42],[64]) properties are conveyed.

2.5.1 Shrinkage Properties of the PLSR Estimator

Characterization of the PLSR estimator in terms of shrinkage properties is useful for providing a link between PLSR and other shrinkage estimators.

In general, if the singular value decomposition of X is given by $X = U\Lambda^{\frac{1}{2}}V'$, where the columns of U, $\mathbf{u}_j$, and the columns of V, $\mathbf{v}_j$, are left and right singular vectors of X, respectively and Λ is diagonal matrix with ordered eigenvalues of $X'X$, $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_p$, on the diagonal; then the shrinkage estimator of β is

$$\widehat{\beta} = \sum_{j=1}^{p} f(\lambda_j) \frac{u_j' y}{\sqrt{\lambda_j}} v_j = \sum_{j=1}^{p} f(\lambda_j) \alpha_j v_j \tag{2.23}$$

where $\alpha_j = \frac{u_j' y}{\sqrt{\lambda_j}}$, $f(\lambda_j)$ are *shrinkage factors* and $u_j' y$ are called *Fourier coefficients* ([53]).

Since the OLS estimator in (2.19) can be rewritten as $\sum_{j=1}^{p} \alpha_j v_j$, all shrinkage factors, $f(\lambda_j)$, are set to be 1 for the OLS estimator. Setting $f(\lambda_j)$ to values other than 1 introduces bias into the estimation process which is beneficial since the increase in bias results in a decrease in mean-squared error. Shrinkage factors less than 1 lead a reduction in the variance of the estimator, whereas factors greater than 1 result in simultaneous increase in both variance and bias.

The shrinkage factors for the RR estimator are

$$f(\lambda_j) = \frac{\lambda_j}{\lambda_j + \xi}$$

for $j = 1, 2, \ldots, p$ while factors of the PCR estimator for $h \leq p$ components are:

$$f(\lambda_j) = \begin{cases} 0 & j > h \\ 1 & j \leq h \end{cases}$$

It can be easily seen that the shrinkage factors for PCR and RR are between 0 and 1. Therefore, they are regarded as shrinkage methods since they shrink $\widehat{\beta}$ by shrinking some of the α_j.

Frank and Friedman ([31]) were the first statisticians who stated the shrinkage property of the PLSR estimator using simulation studies, but they did not provide theoretical proof. The shrinkage properties of the PLSR estimator can be investigated theoretically through its relationship with Krylov space and conjugate gradient method (CG) ([70]). Therefore, initially, a brief information on these concepts would be appropriate. The space spanned by the columns of $\mathbf{z}, A\mathbf{z}, A^2\mathbf{z}, \ldots, A^{m-1}\mathbf{z}$ is called m dimensional *Krylov space* of a square matrix A and a vector $\mathbf{z}$ and denoted by $\mathscr{K}_m(A, z)$. An alternative form of PLSR estimator with h components can be given in terms of h dimensional Krylov space of $\mathscr{K}_h(X'X, X'y)$. W_h, obtained from NIPALS algorithm, is an orthonormal basis for the space $\mathscr{K}_m(X'X, X'y)$ ([41]) and it is central to connection between PLSR and conjugate gradient (CG) method.

CG method aims to solve a system of linear equations $C\mathbf{b} = \mathbf{c}$ arising out of the minimization of the quadratic function $\Psi(b) = \frac{1}{2}b'Cb - c'b$ for a positive semi-definite matrix C. The solution by CG algorithm for h iterations can be obtained from the following algorithm:

Algorithm 2.3 *(CG)*

Step 1: Let $\mathbf{b}_0 = \mathbf{0}$ and $\mathbf{d}_0 = \mathbf{e}_0 = \mathbf{c} - C\mathbf{b}_0 = \mathbf{c}$. Repeat the Step 2− Step 5 for $j = 0, \ldots, h-1$:

Step 2: Calculate z_j

$$z_j = \frac{d_j' e_j}{d_j' C d_j}. \tag{2.24}$$

Step 3: Calculate $\mathbf{b}_{j+1}$

$$b_{j+1} = b_j + z_j d_j. \tag{2.25}$$

Step 4: Calculate residual: $\mathbf{e}_{j+1}$

$$e_{j+1} = c - Cb_{j+1} = c - C(b_j + z_j d_j) = e_j - z_j C d_j. \tag{2.26}$$

Step 5: Calculate $\mathbf{d}_{j+1}$

$$d_{j+1} = e_{j+1} - \left(\frac{e_{j+1}' C d_j}{d_j' C d_j} \right) d_j, \tag{2.27}$$

and set $j := j + 1$.

When $C = X'X$ and $c = X'y$, the normal equations given in (2.18) is obtained and for any arbitrary initial solution, $\mathbf{b}_p$ converges (in exact arithmetic) to $(X'X)^{+}X'y$ which is equal to $\widehat{\beta}_{OLS}$ if X is full rank matrix ([70]).

There are important properties of the vectors generated during the CG algorithm ([36]). For instance, the space spanned by vectors $\{\mathbf{d}_0, \mathbf{d}_1, \ldots, \mathbf{d}_{h-1}\}$ is $\mathscr{K}_h(C, c)$ and the residual vectors $\{\mathbf{e}_0, \mathbf{e}_1, \ldots, \mathbf{e}_{h-1}\}$ span the same Krylov space. Beside this, since the residuals are orthogonal, these vectors are actually the $\mathbf{w}_j$'s from NIPALS algorithm when $C = X'X$ and

$c = X'y$ for $j = 1, 2, \ldots, h$ except for a scale factor. Therefore, the CG estimator, $\mathbf{b}_j$, for $j = 1, 2, \ldots, h$, is identical to the PLSR estimator for j components, $\widehat{\beta}_{PLS}^{(j)}$, and the PLSR estimator for $h = p$ components yields the OLS estimator. Another important property of CG estimator, $\mathbf{b}_j$, is that it minimizes $(\mathbf{b} - \mathbf{b}_j)'C(\mathbf{b} - \mathbf{b}_j)$ over $\mathscr{K}_j(C, c)$ $j = 1, 2, \ldots, h$. Therefore, the PLSR estimator based on h components is given by

$$\widehat{\beta}_{PLS}^{(h)} = \underset{b \in \mathscr{K}_h(X'X, X'y)}{\text{argmin}} (y - Xb)'(y - Xb). \tag{2.28}$$

Furthermore, the relationship between CG and PLSR can be used to prove the following proposition.

Proposition 2.2 *(De Jong, [17]) The norm of $\widehat{\beta}_{PLS}^{(h)}$ is strictly non-decreasing function of the number of components, h, i.e. PLSR estimator of β shrinks.*

This result was proven algebraically by De Jong ([17]). Another proof by Phatak and De Hoog ([70]) given below is more compact which uses relationship between CG method and PLSR estimator.

Proof:

It is necessary to prove that the norms of CG estimators for normal equations, $\mathbf{b}_h = \widehat{\beta}_{PLS}^{(h)}$, increase as h increases, that is $\|b_1\| < \|b_2\| < \ldots < \|b_p\| = \|\widehat{\beta}_{OLS}\|$. In general, (2.25) gives

$$\|b_{j+1}\|^2 = \|b_j\|^2 + {z_j}^2\|d_j\|^2 + 2z_j d_j' b_j.$$

Hence, to prove that $\|b_{j+1}\|^2 > \|b_j\|^2$, it is sufficient to show that $z_j d_j' b_j > 0$ since ${z_j}^2\|d_j\|^2 \geq 0$. By multiplying both sides of (2.27) by $\mathbf{e}_{j+1}'$, the following equality is obtained:

$$e_{j+1}' d_{j+1} = e_{j+1}' e_{j+1} - \left(\frac{e_{j+1}' X'X d_j}{d_j' X'X d_j}\right) e_{j+1}' d_j. \tag{2.29}$$

Since residuals are orthogonal and span$\{\mathbf{e}_0, \mathbf{e}_1, \ldots, \mathbf{e}_{j+1}\}$=span$\{\mathbf{d}_0, \mathbf{d}_1, \ldots, \mathbf{d}_{j+1}\}$, $\mathbf{e}_{j+1}$ is orthogonal to $\mathbf{d}_j$. Therefore $e'_{j+1}d_j = 0$ in (2.29) and $e'_{j+1}d_{j+1} = e'_{j+1}e_{j+1}$ for $j = 1, 2, \ldots, p-1$ from which it follows that

$$z_j = \frac{d'_j e_j}{d'_j X'Xd_j} = \frac{e'_j e_j}{d'_j X'Xd_j} > 0. \tag{2.30}$$

From (2.25), $b_j = \sum_{i=0}^{j-1} z_i d_i$ so $d'_j b_j = \sum_{i=0}^{j-1} z_i d'_j d_i$. Therefore, it is sufficient to prove that $d'_j d_i > 0$ for $i \neq j$ to show that $d'_j b_j > 0$. The statement $d'_j d_i > 0$ can be proven by expressing $\mathbf{d}_j$ vectors in terms of the residuals. If $l_j = -\frac{e'_{j+1}X'Xd_j}{d'_j X'Xd_j}$, then (2.27) gives

$$d_j = e_j + \sum_{i=0}^{j-1} \left(\prod_{k=i}^{j-1} l_k \right) e_i. \tag{2.31}$$

So, $d'_j d_i > 0$ only if $l_j > 0$. Again, utilizing the (2.26), we get

$$e_{j+1} = e_j - z_j X'Xd_j \Longrightarrow X'Xd_j = z_j^{-1}(e_j - e_{j+1}).$$

Therefore, l_j can be rewritten as

$$l_j = -\frac{e'_{j+1}X'Xd_j}{d'_j X'Xd_j} = -\frac{e'_{j+1}(e_j - e_{j+1})}{z_j d'_j X'Xd_j} = \frac{\|e_{j+1}\|^2}{z_j d'_j X'Xd_j} > 0 \tag{2.32}$$

which completes the proof □.

Expressing shrinkage factors of the PLSR estimator, in terms of eigenvalues of the matrices $X'X$ and $W'_h X'XW_h$, is also possible and this expression, given in the next proposition ([53]), provides valuable information on how the shrinkage behavior of the PLSR estimator is.

Proposition 2.3 *(Lindjæerde and Christophersen, [53]) The shrinkage factors, from (2.23), for the PLSR estimator for h components are given by*

$$f(\lambda_j)^{(h)} = 1 - \prod_{i=1}^{h} \left(1 - \frac{\lambda_j}{\theta_i^{(h)}}\right) \tag{2.33}$$

for $j = 1, 2, \ldots, p$, where $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_p$ and $\theta_1^{(h)} \geq \theta_2^{(h)} \geq \ldots \geq \theta_h^{(h)}$ are the eigenvalues of $X'X$ and $W_h'X'XW_h$, respectively.

Shrinkage factors of the PLSR estimator are determined using the relationship between PLSR and Lanchoz method ([41], [55]). Lanchoz method transforms an original symmetric matrix, C, into a symmetric tridiagonal matrix. Therefore, after applying Lanchoz method on $C = X'X$ and obtaining a tridiagonal matrix, eigenvalues and corresponding eigenvectors of $C = X'X$ can be easily calculated due to the nature of being a tridiagonal matrix. Given some starting vector $c = X'y$, the procedure constructs tridiagonal matrices $W_h'X'XW_h$ for $h = 1, 2, \ldots, p$, where the columns of W_h form an orthonormal basis for $\mathscr{K}_h(C, c) = \mathscr{K}_h(X'X, X'y)$. If $\theta_i^{(h)}$ and $\tau_i^{(h)}$, $i = 1, 2 \ldots, h$ are the eigenvalues and unit-norm eigenvectors respectively of $W_h'X'XW_h$, then $(\theta_i^{(h)}, W_h\tau_i^{(h)})$ are called Ritz pairs.

Although it is known that the PLSR estimator shrinks relative to the OLS estimator (Proposition 2.2), it may expand some of the α_j, that is $f(\lambda_j) > 1$. It is determined that eigenvalues of $X'X$ and Fourier coefficients play an important role on the shrinkage properties of the PLSR estimator ([53]). Especially the cases where the corresponding singular value is large and the corresponding Fourier coefficient is small are the ones that anti-shrinkage property is observed. On the other hand, since the contribution of such terms to the PLSR estimator will be small, anti-shrinkage may not be seriously effective.

2.5.2 The Asymptotic Variance of the PLSR Estimator

Unlike the OLS estimator, the PLSR estimator of β in (2.1) is a non-linear function of the response variable. Thus, the covariance matrix of the PLSR estimator cannot be determined easily. This leads to difficulties in inference based tasks such as choosing optimal number of components, testing significance of coefficients and constructing confidence intervals for the regression coefficients.

Methods based on re-sampling techniques, such as bootstrapping and cross-validation, utilize the original data to gain information about the variability of the estimator. However, these methods are computationally intensive. Furthermore, their applications to typical chemometrics problems are not practical since there are often very few observations.

Another approach is linearizing the non-linear estimator to estimate its variance. Denham ([18]) provided a locally linear approximation to the covariance matrix based on the first derivative of the PLSR vector. More compact expression for the asymptotic covariance matrix is given by Phatak et al. ([71]). In their approach, the approximate covariance matrix for the $\widehat{\beta}^{(h)}_{PLS}$ is calculated using the *delta method* stated below.

Theorem 2.1 *Let $\{Y_n\}$ be a sequence of random vectors. Assume that $\sqrt{n}[Y_n - \mu]$ converges in distribution to $N(0, \Psi)$ and $g(.)$ is a vector function whose derivatives exist in a neighborhood of $z = \mu$, then $\sqrt{n}[g(Y_n) - g(\mu)]$ converges in distribution to $N(0, J\Psi J')$, where $J = \frac{\partial g(z)}{\partial z'} |_{z=\mu}$.*

Therefore, assuming that $var(\varepsilon) = \sigma^2 I_n$, the approximate covariance matrix of $\widehat{\beta}^{(h)}_{PLS}$ can be written as

$$var(\widehat{\beta}^{(h)}_{PLS}) = J_o J_o' \sigma^2 \tag{2.34}$$

where J_o is the Jacobian matrix that consists of derivatives of each element of $\widehat{\beta}^{(h)}_{PLS}$ with respect to elements of y evaluated at given data (X_o, y_o). Jacobian matrix, J_o, is derived by using the matrix differential calculus and given as

$$J_o = \frac{\partial \widehat{\beta}^{(h)}_{PLS}}{\partial y'} \mid_{y=y_o} = \{[y'XW_hG_h \otimes L] + [W_hG_h \otimes y'XL]\}(M^{-1'} \otimes I_p)U_h{}' + H_hX' \quad (2.35)$$

where $G_h = (W_h'X'XW_h)^{-1}$, $H_h = W_hG_hW_h'$, $L = (I_p - H_hX'X)$, M is a matrix such that $K_h = W_hM$ with

$$K_h = [X'y, (X'X)X'y, \ldots, (X'X)^{h-1}X'y].$$

and $U_h = [X, X(X'X), \ldots, X(X'X)^{h-1}]$ is $n \times hp$ matrix ([71]). A reasonable estimate of σ^2 in (2.34) can be calculated by dividing residual sum of squares to an appropriate degrees of freedom. Although $n - p$ has been suggested as degrees of freedom, due to non-linear form of PLSR estimator it is suggested the use of the trace of $(I_n - XJ_o)'(I_n - XJ_o)$ ([18]).

The results of simulation studies indicated that the covariance matrix estimate based on Jacobian matrix, J_o, provides a useful approximation to the true covariance matrix ([71]). However, since J_o is evaluated using the data from a single experiment, how well $J_oJ_o'\widehat{\sigma}^2$ approximates the actual covariance matrix is directly related to the optimality of number of components, h.

The covariance matrix for the $\widehat{\beta}^{(h)}_{PLS}$ can also be calculated from the influence function ([75]). This approach has advantages over the methods based on linearization approaches such as independence of model assumption and computational easiness. The relationship between influence function of an estimator and its variance is given in Chapter 4.

2.5.3 Consistency of the PLSR Estimator

A consistent estimator is an estimator that converges in probability to the parameter being estimated as the sample size grows without bound. In this section, it is shown that $\hat{\beta}^{(h)}_{PLS}$ is consistent estimator of β in the model (2.1), when specific assumptions are held.

Linear model given in (2.1) can be rewritten as

$$y_i = x_i'\beta + \varepsilon_i \tag{2.36}$$

where x_i is the i^{th} row of X for $i = 1, 2, \ldots, n$. Assuming that x_i's are independently identically distributed random variables with positive definite covariance matrix Σ_{xx} and are independent of ε_i , then

$$\Sigma_{xy} = cov(x_i, y_i) = cov(x_i, x_i'\beta) = \Sigma_{xx}\beta. \tag{2.37}$$

Therefore,

$$\beta = \Sigma_{xx}^{-1}\Sigma_{xy} = \textstyle\sum_{h=1}^{p} \frac{v_h v_h'}{\lambda_h}\Sigma_{xy}$$

where λ_h and v_h are the h^{th} eigenvalue and the corresponding eigenvector of Σ_{xx} for $h = 1, 2 \ldots, p$. In general, eigenvectors with $v_h'\Sigma_{xy} \neq 0$, one for each λ_h, are called the *relevant eigenvectors* of Σ_{xx} for prediction of y ([42], [59]), i.e., eigenvectors of Σ_{xx} with non-zero components along Σ_{xy}.

Helland ([41]) provided a general form of the $\hat{\beta}^{(h)}_{PLS}$ as

$$\hat{\beta}^{(h)}_{PLS} = \hat{D}_h(\hat{D}_h' S_{xx}\hat{D}_h)^{-1}\hat{D}_h' S_{xy} \tag{2.38}$$

where S_{xx} is the sample covariance estimator of Σ_{xx}, S_{xy} is the sample estimator of Σ_{xy}, and $\hat{D}_h$ is $p \times h$ matrix with columns that span $\mathscr{K}_h(S_{xx}, S_{xy})$. Helland ([42]) also proved that if m is the number of relevant eigenvectors of Σ_{xx} for prediction of y, then m is the least integer h such that $\beta = \Sigma_{xx}^{-1}\Sigma_{xy}$ belongs to $\mathscr{K}_h(\Sigma_{xx}, \Sigma_{xy})$. These facts are used to prove the consistency of the PLSR estimator given in the next lemma.

Lemma 2.1 *Assume that S_{xx} and S_{xy} converge to Σ_{xx} and Σ_{xy} in probability, respectively. If $h = m$ is the number of the relevant eigenvectors of Σ_{xx} for prediction of y, then $\hat{\beta}_{PLS}^{(h)}$ is consistent estimator of β.*

Proof: Since S_{xx} and S_{xy} converge to Σ_{xx} and Σ_{xy} in probability, respectively, h component PLSR estimator in (2.38), $\hat{\beta}_{PLS}^{(h)}$, converges in probability to

$$D_h(D_h'\Sigma_{xx}D_h)^{-1}D_h'\Sigma_{xy} = D_h(D_h'\Sigma_{xx}D_h)^{-1}D_h'\Sigma_{xx}\beta$$

where columns of D_h span $\mathscr{K}_h(\Sigma_{xx}, \Sigma_{xy})$. If $h = m$, then, as proved by Helland ([42]), $\beta \in \mathscr{K}_h(\Sigma_{xx}, \Sigma_{xy})$. Consequently, the space spanned by $D_h^* = \Sigma_{xx}^{1/2}D_h$ contains $\Sigma_{xx}^{1/2}\beta$, which implies

$$D_h^*(D_h^{*'}D_h^*)^{-1}D_h^{*'}\Sigma_{xx}^{1/2}\beta = \Sigma_{xx}^{1/2}\beta.$$

Multiplying each side by $\Sigma_{xx}^{-1/2}$ from left, we obtain

$$D_h(D_h^{*'}D_h^*)^{-1}D_h'\Sigma_{xx}\beta = D_h(D_h'\Sigma_{xx}D_h)^{-1}D_h'\Sigma_{xx}\beta = \beta.$$

So, $\hat{\beta}_{PLS}^{(h)}$ converges in probability to β □.

Chapter 3

RoPLS: Robust Partial Least Squares Regression

3.1 Introduction

Although PLS regression handles the multicollinearity problem, it fails to deal with data containing outliers since PLS algorithms are based on the empirical cross-covariance matrix between the response variable(s) and the regressors. Existence of multicollinearity and outliers in the data sets leads to a requirement of robust PLS methods in many application areas. Consequently, several robust PLS regression methods have been proposed in the literature. In general, these methods can be classified in two groups: those which use iteratively reweighting technique and those which use the idea of robustication of covariance matrix.

The methods utilizing iteratively reweighting idea, assign a weight between 0 and 1 to the each data point in a way that outliers, points which are sufficiently far away from the bulk of the data, gain less weights than inliers, points which are the bulk of the data. The first robust PLS method in this group, proposed by Wakeling and Macfie ([83]), is based on the idea of replacing a set of ordinary regression steps in NIPALS algorithm by robust counterparts. The main drawback of the method is the large amount of the computation time. Therefore, Griep et al. ([38]) suggested a semi-robust method by replacing only the first ordinary regression step by a robust regression method for the sake of computation. However, the method looked at outliers onto planes $[y, X_j]$ where $j = 1, 2, \ldots, p$, while ignoring the multivariate nature of the data. Cummins and Andrews ([13]) gave a slightly different version of iteratively reweighting method by calculating weights after performing an ordinal PLS algorithm and using these weights for the next PLS algorithm. Since the two aforementioned methods employ a robust method within the PLS algorithm they are called "*internal iteratively reweighting*" methods whereas the method by Cummins and Andrews is called "*external iteratively reweighting*" method ([35]). Although the latter method has advantages over the internal reweighting methods, it suffers from the lack of resistance to outliers in X space. Only recently, another external iterative method, called "*Partial Robust M Regression*" (PRM), proposed by Serneels et al. ([76]) which is robust to outliers in X space.

The second group of robust methods, introduced by Gil and Romera ([35]) estimate the regression coefficients with the help of a robust covariance matrix instead of applying robust regression method in PLS algorithms. Gil and Romera estimated covariance matrix using Stahel-Donoho estimator ([22], [78]). However, this method can not be applied to high dimensional data sets since it uses a resampling scheme by drawing subsets of size $p + 2$. Hubert and Branden ([48]) proposed another two step algorithm (RSIMPLS), that can be used for both low and high dimensional data, by estimating covariance matrix using MCD (minimum covariance determinant, [74]) and the idea of projection pursuit. Once the score matrix is obtained, robust regression is used to estimate $\mathbf{a}$ in (2.7), so the β in the equation (2.1).

In this chapter, robustified versions of the SIMPLS algorithm, RoPLS1 and RoPLS2, are introduced. The proposed methods are *external iteratively reweighting* algorithms based on the idea of reweighted least squares method given by Billor et al. ([6]). RoPLS1 uses weights calculated from BACON (blocked adaptive computationally-efficient outlier nominators) algorithm ([5]), whereas RoPLS2 uses weights calculated from PCOUT algorithm

([26]). RoPLS2 is the first method that incorporates PCOUT algorithm as an integral component in a robust estimation procedure rather than just as an outlier detection method. Both RoPLS1 and RoPLS2 have computational advantages over recent robust methods PRM and RSIMPLS and they are resistant to masking problem.

The rest of the chapter is organized as follows: Outlier identification algorithms, BACON and PCOUT, are reviewed in Section 3.2. The detailed algorithm for the proposed algorithm, RoPLS, including a robust method to determine the number of components and two graphical methods to diagnose outliers, are given in Section 3.3. Real and simulated data sets are utilized to demonstrate the performance of the proposed method in Section 3.4.

3.2 Outlier Detection Methods

The outlier challenge is one of the earliest of statistical interests, and since data sets often contain outliers, it continues to be one of the most important. The outlier detection problem has important applications in the fields of fraud detection, astronomy, bioinformatics (e.g., microarray experiments), and many other countless areas. For instance, the great interest of astronomers is to discover unusual, rare or unknown types of astronomical objects or phenomena. The outlier detection approaches in multiple terabyte, and even larger, multi petabyte data sets, correctly meet the need of astronomers.

Outlier detection algorithms fall into two broad classes: those which employ the distance-based methods, such as MULTOUT ([90]), MCD ([74]), BACON ([5]); and those which rely upon projection pursuit ideas such as Kurtosis1 ([68]) and Stahel-Donoho estimator ([56]). Distance-based methods classify outliers as points which are sufficiently far away from the bulk of the data; whereas projection-pursuit methods use lower dimensional projection of data that enables the user to detect outliers. Primary goals in all of these algorithms are explicit outlier detection and/or robust estimation. Most of the distance-based methods are especially designed for low dimensional data sets, that is, the number of observations is greater than the number of variables, and identification of outliers in higher dimensions becomes more complicated as the dimension increases. On the other hand, nowadays data sets in many scientific fields are high dimensional. Although projection pursuit based methods can be applied to such situations, their computational difficulties make them impractical to use. Recent outlier detection method, PCOUT ([26]), which detects outliers very efficiently in high dimensional data sets, combines the advantages of distance-based and projection pursuit methods. In the following subsections, detailed information on BACON and PCOUT methods is provided, since these algorithms are used to build new robust PLS algorithms in Section 3.3.

3.2.1 BACON

BACON ([5]) algorithm is a distance-based method that starts with an outlier-free subset of data (initial basic subset), from which robust distances can be calculated. The initial basic subset can be found algorithmically in one of two ways: Mahalanobis distances based on classical mean and covariance estimates (Version1) or Euclidian distances from medians (Version2). The advantages of Version1 are its affine equivariance and its low computational cost. Then, based on the mean and covariance matrix of the basic subsetof size m, discrepancies are computed and all observations with discrepancy less than correction factor, $C_{npr}\chi_{p,\alpha/n}$, form the new basic subset, where $\chi_{p,\alpha/n}$ is the $1-\alpha$ percentile of the chi-square distribution with p degrees of freedom and,

$$C_{npr} = C_{np} + C_{hr} = max[0, (h-r)/(h+r)] + 1 + \frac{p+1}{n-p} + \frac{1}{n-h-p}$$

where $h = [(n+p+1)/2)]$ and r is the size of current basic subset. This iterative method is repeated until the size of basic subset, r, no longer changes. The observations excluded by the final basic subset are nominated as outliers. BACON estimators of the location vector $(\widetilde{\mu})$ and the covariance matrix $(\widetilde{\Sigma})$, based on this final subset, are employed to calculate robust Mahalanobis distance vector $\mathbf{d}^B$ with i^{th} row for $i = 1, 2, \ldots, n$ given as

$$d_i{}^B = d(x_i, \widetilde{\mu}, \widetilde{\Sigma}) = \sqrt{(x_i - \widetilde{\mu})'\widetilde{\Sigma}^{-1}(x_i - \widetilde{\mu})} \tag{3.1}$$

where $x_i \in \mathbb{R}^p$ is the i^{th} row of data matrix, X, for $i = 1, 2, \ldots, n$. Simulation studies demonstrate that BACON is a robust technique with 40% breakdown point. The method is fast, easy to implement and thus practical for data sets of even million of cases.

Although BACON algorithm is originally designed for low dimensional data, an extension of BACON algorithm to high dimensional data is also possible. A simple solution to this problem is to run the BACON algorithm on the reduced data set, $\widetilde{X}$, of size $n \times p^*$, where $\widetilde{X}$ is the score matrix based on the spectral decomposition of covariance matrix estimate and p^* denotes the number of such score vectors in the reduced data set. Robust distance vector, $\mathbf{d}^B$, is calculated based on $\widetilde{X}$.

3.2.2 PCOUT

PCOUT is a recent outlier identification algorithm that is particularly effective in high dimensions([26]). PCOUT is based on the idea of outlier detection on principal component space which does not require matrix inversion. The method starts by robustly sphering the original data by columnwise median of absolute deviances for $j = 1, 2, \ldots, p$:

$$x_{ij}^* = \frac{x_{ij} - median_i(x_{ij})}{(1.4826)mad_i(x_{ij})} \tag{3.2}$$

where

$$mad_i(x_{ij}) = median_i(|x_{ij} - median_i(x_{ij})|). \tag{3.3}$$

Then, PCA is applied to sphered data matrix, $X^* = [x_{ij}^*]$, and $p^* < p$ semi-robust components, $Z = [\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_{p^*}]$, that contribute to at least 99% of the total variance are retained in the analysis. The semi-robust component matrix, Z, is sphered similar to (3.2) as

$$z_{ij}^* = \frac{z_{ij} - median_i(z_{ij})}{(1.4826)mad_i(z_{ij})}. \tag{3.4}$$

The rest of the algorithm uses the sphered component matrix, $Z^* = [z_{ij}^*]$, to detect location and scatter outliers. The location outlier detection is initiated by calculating the absolute value of a robust kurtosis measure for each component:

$$k_j = |\frac{1}{n}\sum_{i=1}^{n} {z_{ij}^*}^4 - 3| \tag{3.5}$$

where $j = 1, 2, \ldots, p^*$. Kurtosis coefficient measures "*heaviness*" of distribution tails. Small value of kurtosis is an indicator of large amount of asymmetric contamination, whereas large value of kurtosis is an indicator of either symmetric contamination or small amount of asymmetric contamination. Since the presence of location outliers is likely to cause the kurtosis to become different from zero, the relative weight $v_j = k_j / \sum_{j=1}^{p^*} k_j$ for each of the sphered component, $\mathbf{z}_j^*$, is used to reveal outliers. Robust distances, for detecting location outliers, are calculated based on the weighted component matrix:

$$RD_i^L = \sqrt{\sum_{j=1}^{p^*} v_j^2 {z_{ij}^*}^2} \tag{3.6}$$

and then the following equation:

$$d_i^L = RD_i^L \frac{\sqrt{(\chi^2_{p^*}, 0.5)}}{median_i(RD_i^L)}, \tag{3.7}$$

suggested by Maronna and Zamar ([57]), is used to transform distances.

To detect scatter outliers, sphered component matrix, Z^*, is directly used to calculate robust distances, RD_i^S, by plugging $v_j = 1$ for all components in (3.6). Similarly, d_i^S is calculated using the transformation given in equation (3.7) by replacing RD_i^L by RD_i^S.

After calculating robust distances, the translated biweight function ([72]),

$$w(d; c, M) = \begin{cases} 0 & \text{d}\geq\text{c} \\ \left(1 - \left(\frac{d-M}{c-M}\right)^2\right)^2 & \text{M<d<c,} \\ 1 & \text{d}\leq\text{M} \end{cases}$$

is used to assign weights to each observation and take these weights as a measure of outlyingness. Weights for the location and the scatter outliers are calculated using different choices of c and M. Since it is assumed that squared d_i^S is distributed with $\chi^2_{p^*}$, the weights for the scatter outliers are given by

$$w_i^S = w(d_i^S; \sqrt{q_{99}(\chi^2_{p^*})}, \sqrt{q_{25}(\chi^2_{p^*})}) \tag{3.8}$$

where $M = q_{25}(\chi^2_{p^*})$ is the 25^{th} and $c = q_{99}(\chi^2_{p^*})$ is the 99^{th} quantiles of $\chi^2_{p^*}$, respectively. However, the kurtosis weighting scheme destroys any resemblance of squared d_i^L to a χ^2 distribution. Therefore, the weights for the location outliers are given as:

$$w_i^L = w(d_i^L; median_i(d_i^L) + (2.5)mad_i(d_i^L), q_{33.\bar{3}}(d_i^L)) \tag{3.9}$$

where $M = q_{33.\bar{3}}(d_i^L)$ is $33\frac{1}{3}^{rd}$ sample quantile of $\mathbf{d}^L = [d_1^L, d_2^L, \ldots, d_n^L]'$.

Final weights are calculated as a function of scatter and location weights:

$$w_i^{LS} = \frac{(w_i^L + s)(w_i^S + s)}{(1+s)^2} \tag{3.10}$$

where scaling constant s=0.25. Any observation with final weight less than 0.25 is assigned as an outlier. PCOUT is shown to be very fast algorithm which has good performance for high-dimensional data and a comparable performance to standard outlier detection methods in low dimensions.

3.3 Description of the Proposed Algorithm: RoPLS

3.3.1 RoPLS Algorithm

RoPLS is a new iterative robust "*external reweighting algorithm*" which gives low weight to points with high leverage and/or large residuals where the weights change from iteration to iteration. RoPLS consists of two algorithms. In the first algorithm, robust distances and initial weights are calculated using either BACON algorithm (RoPLS1) or modified PCOUT algorithm (RoPLS2). In the second algorithm, iteratively reweighted PLS regression based on SIMPLS algorithm is performed using the initial weights and normalized robust distances. The detailed explanation of these two algorithms is given below:

Algorithm I:

Input: $n \times p$ data matrix, X, and $n \times 1$ vector of response variable, y

Output: Initial weight vector, $\mathbf{w}_0 = [w_1^0, w_2^0, \ldots, w_n^0]'$, and normalized distance vector, $\mathbf{d} = [d_1, d_2, \ldots, d_n]'$

i. RoPLS1: Let $\Gamma = [X : y]$.

Apply BACON algorithm to Γ to obtain robust BACON (Version1) estimators of center, $\widetilde{\mu}$, and scatter matrix, $\widetilde{\Sigma}$, so the robust distance vector, $\mathbf{d}^B$ with i^{th} row $d_i^B = d(\gamma_i, \widetilde{\mu}, \widetilde{\Sigma})$ where $d(:)$ is the distance function defined in (3.1). After calculating the robust distances using BACON algorithm, initial weights can be obtained as $w_i^0 = w^* \left(d_i^B\right)$ where

$$w^*\left(a_i\right) = min\left(1, \frac{1}{max\left(|a_i|, median_i(|a_i|)\right)}\right) = min\left(1, \frac{\psi^H\left(\frac{|a_i|}{median_i(|a_i|)}\right)}{|a_i|}\right) \tag{3.11}$$

and $\psi^H(:)$ in equation (3.11) is Huber's function ([47]) defined as

$$\psi^H(v) = v \min\left(1, \frac{1}{|v|}\right) \tag{3.12}$$

for any non-zero v. The idea is to give low weights to the observations with large robust distances. $\mathbf{w}^0 = [w_1^0, w_2^0, \ldots, w_n^0]'$ will be used as initial weight vector in the second part of the algorithm.

Since normalized distances are between 0 and 1, normalized robust distances in X space, ${\mathbf{d}_x}^B$, can be used as robust measure of leverage. Normalized distances, d_i, for $i = 1, 2 \ldots, n$ are calculated by

$$d_i = \frac{{d_{x(i)}^B}^2}{\sum_{i=1}^n {d_{x(i)}^B}^2} \tag{3.13}$$

where $d_{x(i)}^B = d(x_i, \widetilde{\mu_x}, \widetilde{\Sigma_x})$ with $\widetilde{\mu_x}$ is first p rows of $\widetilde{\mu}$ and $\widetilde{\Sigma_x}$ is the upper pxp matrix in $\widetilde{\Sigma}$. Normalized distance vector, $\mathbf{d}$, will also be used in the second part of the algorithm to update weights.

If the data matrix X is high dimensional ($n << p$) or rank deficient, then BACON algorithm is applied separately to Γ and X to calculate $\mathbf{d}^B$ and $\mathbf{d}_x^B$, respectively.

ii. RoPLS2:

After applying PCA to the robustly sphered matrix, $\Gamma^* = [X^* : y^*]$, p^* semi-robust components, $Z = [\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_{p^*}]$ are obtained. Then, the robust distance vectors d^L and d^S are calculated using $p^* = rank(X)$ components for low dimensional data and $p^* = n - 1$ components for high dimensional data as described in subsection 3.2.2. The initial weights are obtained as:

$$w_i^0 = w^* \left(d_i^L\right) w^* \left(d_i^S\right). \tag{3.14}$$

$\mathbf{d}_x^L$ and $\mathbf{d}_x^S$ are calculated same as above where Γ^* is taken as X^*, that is robustly sphered data matrix, X. Normalized distances, d_i, are then calculated by

$$d_i = \frac{{d_{x(i)}^{PC}}^2}{\sum_{i=1}^n {d_{x(i)}^{PC}}^2} \tag{3.15}$$

where

$$d^{PC}_{x(i)} = max\left(\frac{d^L_{x(i)}}{mad_i(d^L_{x(i)})}, \frac{d^S_{x(i)}}{mad_i(d^S_{x(i)})}\right). \tag{3.16}$$

Algorithm II:

Input: $n \times p$ data matrix, X, $n \times 1$ vector of response variable, y, initial weight vector, $\mathbf{w}^0$, and normalized robust distance vector, $\mathbf{d}$, from Algorithm I:

Output: Coefficient vector, β, score matrix, T, and $p \times k$ PLS weight matrix, R

***Step 0*:** Let $W = W_0 = diag\{\sqrt{w^0{}_1}, \sqrt{w^0{}_2}, \ldots, \sqrt{w^0{}_n}\}$.

***Step 1*:** Weight data matrices X and y by multiplying with W:

$$X_w = WX$$

$$y_w = Wy$$

Perform SIMPLS regression of X_w on y_w to obtain $\widehat{\beta}$, T and R.

***Step 2*:** Calculate the residual vector, $\mathbf{r} = y - X\widehat{\beta}$, and update the weights using the following equation:

$$w_i = (1 - d_i)w^*\left(\frac{r_i}{mad_i(r_i)}\right) \tag{3.17}$$

where d_i is normalized distances obtained in Algorithm I. Redefine $W = diag\{\sqrt{w_1}, \sqrt{w_2}, \ldots, \sqrt{w_n}\}$.

***Step 3*:** Return Step 1 until the convergence of $\widehat{\beta}$.

This algorithm is inspired by the weighted least squares regression method by Billor, et al.([6]). They suggested to use normalized robust distance, d_i, as an alternative to the diagonal element of hat matrix, p_{ii}, since it is known that p_{ii} values can be distorted by the presence of masking problem. We extended this approach to weighted partial least squares regression.

3.3.2 Selecting Number of Components

The decision on the optimal number of components, k, is very important issue in building the PLS regression model. In most of the cases the leave-one-out cross-validated root mean squared error, $RMSE$, is used to choose the appropriate number of components:

$$k = \underset{h}{\operatorname{argmin}} \left(RMSE^{(h)} \right) \tag{3.18}$$

where

$$RMSE^{(h)} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} \left(y_i - \widehat{y}_{-i(h)} \right)^2} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} \left(r_{-i(h)} \right)^2}. \tag{3.19}$$

Here, the predicted values $\widehat{y}_{-i(h)}$ and residuals $\mathbf{r}_{-i(h)}$ are based on the parameter estimates obtained from the data set without the i^{th} observation using a PLS regression model with h components. However if a data set contains outliers, $RMSE^{(h)}$ can attain unreliable values, which results in the wrong decision of k. Therefore, we propose to calculate $RMSE^{(h)}$ robustly using the BACON algorithm. If an $n \times h$ matrix of residuals, $[\mathbf{r}_{-(1)}, \mathbf{r}_{-(2)} \ldots, \mathbf{r}_{-(h)}]$, is constructed, the observations with large residuals can be detected by applying BACON algorithm to the constructed residual matrix. If I is the final subset of clean observations with size n_I, the robust version of $RMSE^{(h)}$ is defined as

$$RRMSE^{(h)} \left(I \right) = \sqrt{\frac{1}{n_I} \sum_{i \epsilon I} \left(y_i - \widehat{y}_{-i(h)} \right)^2} = \sqrt{\frac{1}{n_I} \sum_{i \epsilon I} \left(r_{-i(h)} \right)^2}. \tag{3.20}$$

It is also possible to use robust cross-validated coefficient of determination, R^2 to select optimal number of components:

$$RR^{2^{(h)}} = 1 - \frac{\sum_{i\epsilon I} \left(r_{-i(h)}\right)^2}{\sum_{i\epsilon I} \left(y_i - \overline{y}_I\right)^2} \tag{3.21}$$

with $\overline{y}_I = \frac{1}{n_I} \sum_{i\epsilon I} y_i$. The value of k which makes (3.21) the maximum can be taken as the optimal value.

Once the value of k is determined, the model can be validated by calculating $RRMSE^{(k)}$ based on the final subset obtained by applying BACON algorithm to the residual vector $\mathbf{r}_{-k}$ instead of $n \times h$ residual matrix. Applications of the $RRMSE$ and RR^2 on the real data sets to determine value of k are given in Section 3.4.

3.3.3 Diagnostic Plots

In PLS regression analysis; orthogonal, score and residual distances can be used to measure degree of outlyingness of observations as in PCR ([48]). The orthogonal distance measures the distance between an observation , x_i, and its projection in the $k-$dimensional PLS-subspace, $\widehat{x}_i$, and it is defined as

$$od_i = \parallel x_i - \widehat{x}_i \parallel . \tag{3.22}$$

Score distance measures the outlyingness of a point in the PLS-subspace. The score distance, sd_i, is calculated using the sphered score matrix, T^*, as

$$sd_i = t_i^{*'} t_i^* \tag{3.23}$$

where t_i^* is the i^{th} row of T^*. Residual distance can be regarded as the distance of scaled residuals from 0, that is

$$rd_i = \left| \frac{r_i}{mad_i(r_i)} \right| \tag{3.24}$$

and vertical outliers are known to have large residual distances.

The outcomes of PLS regression analysis on a real data set can be visualized via several graphical displays of these distances. One crucial point here is to determine how large these distances should be to classify an observation as an outlier. Instead of using distribution based threshold points, we use classical nonparametric threshold. For any vector $\mathbf{v}$ with positive entries, $\mathbf{v} \geq 0$, the cutoff point can be given as;

$$tr\,(v) = median_i(v_i) + (2.5)mad_i(v_i). \tag{3.25}$$

Two graphs will be used to distinguish regular observations from outliers:

i. Residual-Score Plot:

This is a scatter plot of the residual distance, rd_i, versus the score distance, sd_i. The vertical ($x = tr(\mathbf{sd})$) and horizontal ($y = tr(\mathbf{rd})$) threshold lines, calculated from (3.25), are also displayed on the scatter plot which help us to flag outliers and determine their types. We can distinguish four types of observations using this plot. Observations in the

1. lower left corner are "*regular*" (i.e., homogeneous),

2. lower right corner, that are far from the PLS-space, but lying in the direction of fitted line or space, are "*good leverage*" points,

3. upper left corner, that are far away from the y space, are "*vertical*" outliers,

4. upper right corner, that have large residual and large score distances, are "*bad leverage*" points.

Especially vertical and bad leverage outliers are known to be very influential for the classical least squares regression fit.

ii. Orthogonal-Score Distance Plot:

This is a scatter plot of the orthogonal distance, **od**, versus the score distance, **sd**. The vertical ($x = tr(\mathbf{sd})$) and horizontal ($y = tr(\mathbf{od})$) threshold lines , calculated from (3.25), are also displayed on the scatter plot which help us to flag outliers in PLS-subspace and determine their type. We can also distinguish four types of observations using this plot. Observations in the

1. lower left corner are "*regular*" (i.e., homogeneous),
2. lower right corner, that have large score but small orthogonal distances, are "*good PLS-leverage*",
3. left corner, that have large orthogonal distances, are "*orthogonal" outliers*,
4. upper right corner, that have large score and orthogonal distances, "*bad PLS-leverage*" observations.

In Figure 3.1 ([49]), four types of observations, described above, can be detected. The regular observations are the homogeneous ones that are close to the PLS-subspace. Observations 1 and 4 in Figure 3.1 are good PLS-leverage points that lie close to the PLS-subspace but far from the regular data. Observation 5 in Figure 3.1 is an orthogonal outlier and it can not be distinguished from the regular observations once it is projected onto the PLS-subspace. Bad leverage points, such as observations 2 and 3 in Figure 3.1, lie far outside the PLS-subspace

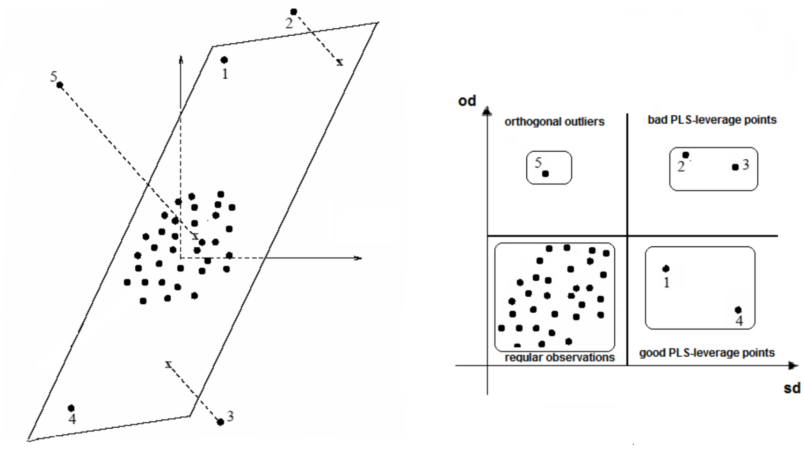


Figure 3.1: Types of outliers when 3 dimensional data set X is projected on 2 dimensional PLS-subspace (left) and corresponding Orthogonal-Score distance plot (right).

and after projection far from the regular data. These graphical displays are constructed for the real data sets to identify outliers in Section 3.4.

3.4 Numerical Examples

In this section, two benchmark data sets and two simulation settings are given to demonstrate the goodness of the proposed algorithms, RoPLS1 and RoPLS2.

3.4.1 Simulation

Two simulation configurations are conducted. The first setting aims to assess the robustness of the proposed algorithms under the different error distributions. The second setting is constructed to assess the robustness of the proposed algorithms to the outliers in $X-$space.

Simulation Setting 1

A similar simulation setting described by Serneels et al. ([76]) is employed in this section. Elements of $n \times k$ score matrix, T, and $p \times k$ x-loadings matrix P are generated from the standard normal distribution. The data matrix X and the response vector y are generated based on the following two models. The first model is

$$X = TP' + E \tag{3.26}$$

where the error matrix E is filled with numbers from normal distribution with mean 0, and standard deviation 0.01. The second model for $i = 1, 2, \ldots, n$ is

$$y_i = x_i'\beta + \varepsilon_i \tag{3.27}$$

where components of β are randomly generated from normal distribution with mean 0 and standard deviation 0.01. Model error term, ε_i, is generated from standard normal, Student t with 2 and 5 degrees of freedom, the Laplace distribution and heavy tailed distributions Cauchy and Slash. The values that have been chosen for the different parameters for this simulation study are the following: the number of iterations, for each fixed value of n, p is $N = 1000$; the size of the data matrices are 30×6 (n/p=5), 25×125 (n/5=0.2), and 20×200 (n/p=0.1). The optimal number of components, k, is fixed as 2 for each setting. The mean square error of $\widehat{\beta}$ given below is utilized as a quantitative measure of the goodness of the estimator:

$$MSE = \frac{1}{N} \sum_{i=1}^{N} \|\widehat{\beta}^i - \beta\|^2 \tag{3.28}$$

Low Dimension: n=30, p=6						
Method \ Error	N(0,1)	t5	Laplace	t2	Cauchy	Slash
RoPLS1	0.0287	0.0381	0.0356	0.0474	0.0799	0.1739
RoPLS2	0.0288	0.0382	0.0354	0.0475	0.0799	0.1735
PRM	0.0276	0.0386	0.0379	0.0504	0.0885	0.1810
RSIMPLS	0.0466	0.0883	0.0644	0.0985	0.1318	0.2369
SIMPLS	0.0246	0.0438	0.0497	0.3110	67.7	153600
High Dimension: n=25, p=125						
RoPLS1	0.0132	0.0135	0.0134	0.0138	0.0153	0.0176
RoPLS2	0.0132	0.0135	0.0134	0.0138	0.0153	0.0176
PRM	0.0132	0.0135	0.0134	0.0139	0.0154	0.0179
RSIMPLS	0.0140	0.0144	0.0142	0.0148	0.0177	0.0204
SIMPLS	0.0131	0.0184	0.0140	0.0261	20	47
High Dimension: n=20, p=200						
RoPLS1	0.0206	0.0207	0.0208	0.0211	0.0223	0.0247
RoPLS2	0.0206	0.0207	0.0208	0.0212	0.0224	0.0248
PRM	0.0206	0.0207	0.0208	0.0212	0.0224	0.0247
RSIMPLS	0.0211	0.0219	0.0219	0.0635	0.0239	0.0275
SIMPLS	0.0205	0.0208	0.0213	0.9080	64.3	3.59

Table 3.1: Simulation results based on MSE for low and high dimensional cases.

where $\widehat{\beta}^i$ is the estimate of β at the i^{th} iteration. The results are summarized in Table 3.1.

As expected, SIMPLS yields the minimum MSE when error terms are normally distributed in both low and high dimensional settings. However, when the error distribution is not normal, SIMPLS clearly breaks down which can be seen from the MSE values in Table 3.1. Especially, at error distributions with heavy tails (Cauchy and Slash), the SIMPLS estimator breaks down drastically and the MSE values go beyond any bound. For low dimensional setting; PRM is slightly more efficient at normal model than RoPLS1 and RoPLS2. RSIMPLS is the least efficient estimator which is not surprising due to the fact that it is based on MCD estimator known to have low efficiency. For all other error distributions that are not normal, RoPLS1 and RoPLS2 perform better than SIMPLS, PRM and RSIMPLS.

For high dimensional settings; PRM, RoPLS1 and RoPLS2 have comparable efficiencies for the normal case and RSIMPLS is the least efficient estimator as in low dimensional setting. RoPLS1 and RoPLS2 perform very well comparing the SIMPLS and RSIMPLS for the models with error terms following non-normal distributions.

As a summary, RoPLS1 and RoPLS2 are very comparable across all possible settings. They beat the robust alternative, RSIMPLS, for all considered error distributions. They yield smaller MSE than SIMPLS does for any non-normal distribution. Overall performances of RoPLS1 and RoPLS2 at non-normal error distributions are slightly better than that of PRM.

Simulation Setting 2

The results of the previous simulation study indicates that proposed methods, RoPLS1 and RoPLS2, are promising robust estimators. However, since the design matrix X is kept fixed across all considered configurations, the simulation setting described above can only be used to show resistance to outliers in y space. In this section, another setting that allows us to demonstrate the robustness of proposed methods to high leverage points is discussed. Tobacco data set ([3]), that consists of $n = 25$ observations on 6 explanatory variables, is used for this setting. Although original data have three response variables, only one of them (the percentage of total nitrogen) is used here. The data set is known to be free of bad leverage points ([48]). It was shown that $k = 1$ component is satisfactory to explain multivariate model. For univariate case, we also observed that adding more components into the model does not provide significant decrease in the $RRMSE^{(h)}$ and decided to take the model with $k = 1$. Figure 3.2 displays the diagnostic plots obtained by RoPLS1 (RoPLS2

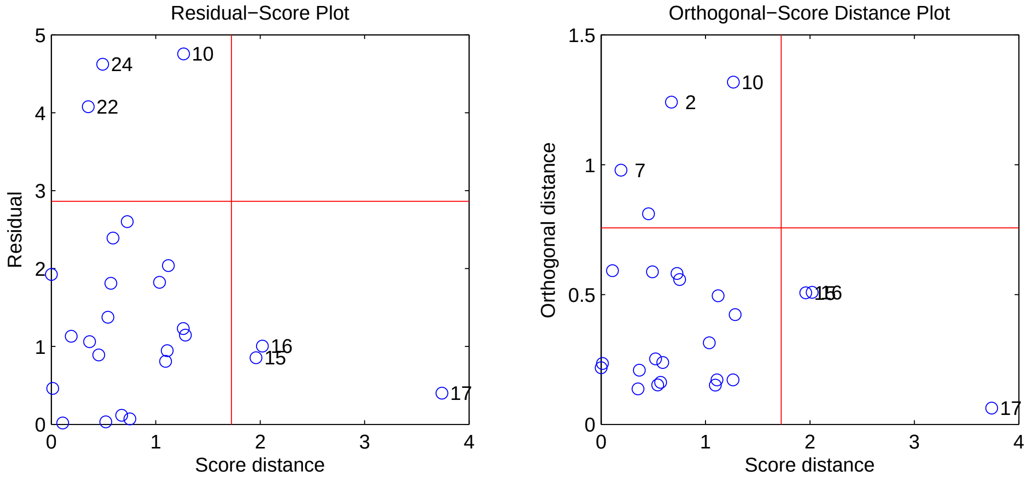


Figure 3.2: Diagnostic plots for original Tobacco data.

yields very similar results). Diagnostic plots indicate that there are no bad leverage points in the data set.

In this section, X will be used to denote 25×6 data matrix and y is the 25×1 response vector. Simulation procedure consists of replacing m randomly chosen observations of the data matrix, X, by m values from multivariate normal distribution with mean $3\mathbf{1}_6 + mean(X)$ and covariance matrix $2cov(X)$ where $\mathbf{1}_6$ is 6×1 vector of ones. In this setting, three different contamination levels are considered by taking m equal to 3, 5 and 13 which correspond to 12%, 20% and 52% of the data, respectively. The setting introduces bad leverage points into the data set. For instance, the diagnostic plots based on the contaminated Tobacco data obtained by replacing observations 6, 13, 17, 20, 21 in X by five observations generated as described earlier can be seen in Figure 3.3. The replaced observations are clearly determined as bad leverage points by RoPLS1.

Standard SIMPLS algorithm and robust methods (RoPLS1, RoPLS2, RSIMPLS and PRM) are implemented on original and contaminated data. The comparison measures are

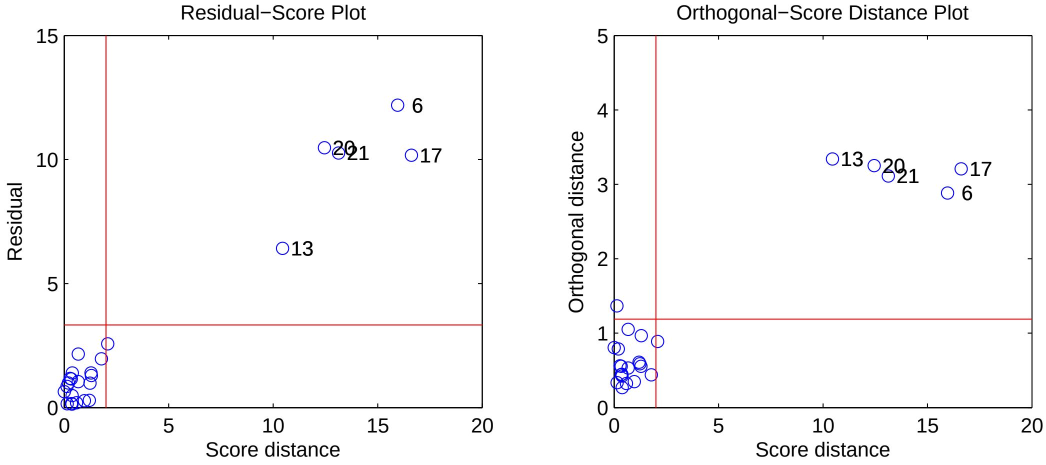


Figure 3.3: Diagnostic plots for contaminated Tobacco data.

the angle (degree) and the norm of the differences between regression estimators for the SIMPLS, RoPLS1, RoPLS2, PRM and RSIMPLS before and after contamination. The results of N=100 iterations for 12%, 20% and 52% contamination levels are given in Figure 3.4.

In the first place, as contamination level increases, the norms and angles corresponding to SIMPLS estimator get larger. Although RSIMPLS provide better results than SIMPLS does, it can not beat any other robust method employed in the setting. RoPLS1, RoPLS2 and PRM are not seriously affected by increasing contamination level as RSIMPLS is.

Several preprocessing methods have been commonly used to eliminate extreme high leverage points before the statistical analysis is applied. However, the vertical outliers are the ones that should be given special attention due to difficulty of detecting distributional assumptions in advance which makes the previous setting more considerable.

Although RoPLS1 and RoPLS2 give very comparable results, one main difference between these two algorithms is that RoPLS1 estimator is orthogonally equivariant (see chapter 4), but RoPLS2 estimator is not since it is calculated based on an algorithm that uses

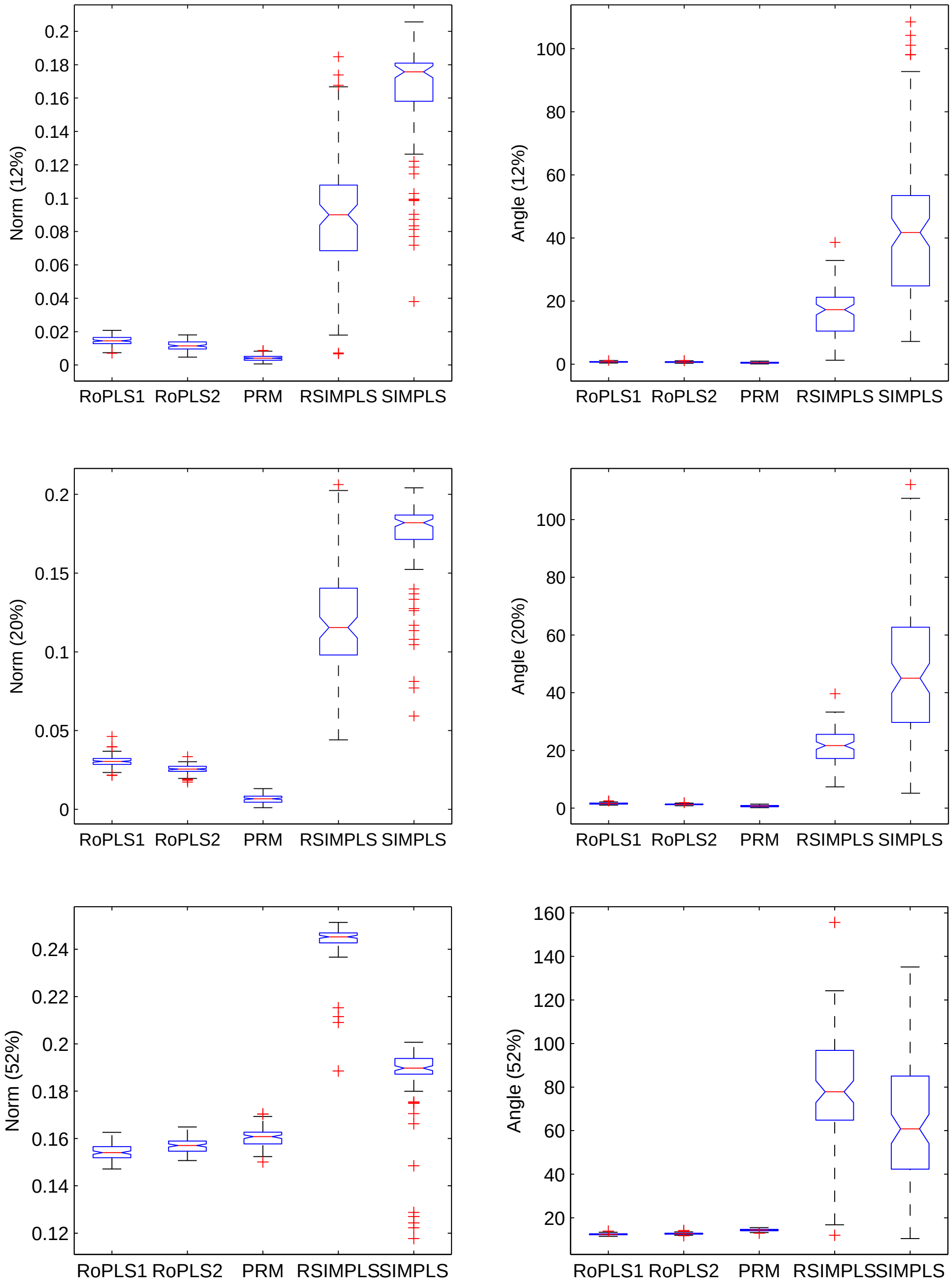


Figure 3.4: Boxplots of norms (left) and angles (right) between slope estimates before and after 12%, 20% and 52% contamination.

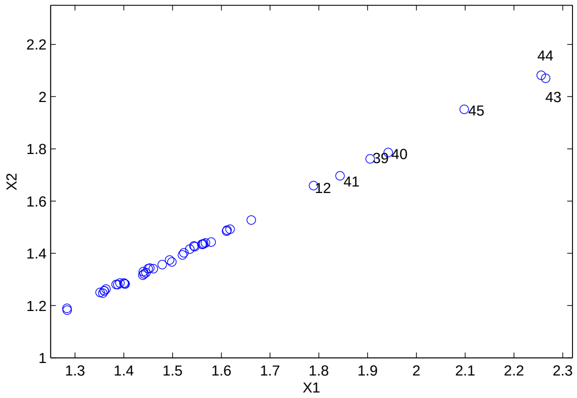


Figure 3.5: The scatter plot of first two columns of Fish data.

column-wise medians. Because of that, only analysis based on RoPLS1 is considered for the real data examples. RoPLS2 yielded very similar results which are not provided here.

3.4.2 Data Sets

Low dimension: Fish Data

In this section, we will illustrate the performance of RoPLS1 algorithm on Fish data that was primarily introduced by Naes ([63]). Data matrix consist of $p = 9$ highly collinear spectra on $n = 45$ measurements of fish (rainbow trout), while the response variable is corresponding to fat concentration. The pairwise correlation coefficients between the columns of X are greater than 0.9. Observations 39 to 45 are reported as outliers ([63]). The scatter plot of the first two columns of the data matrix in Figure 3.5 demonstrates a strong positive relationship between the columns as well as the existence of outlying observations. The aim of the analysis is to determine a model that explains the relationship between these collinear spectra and fat concentration.

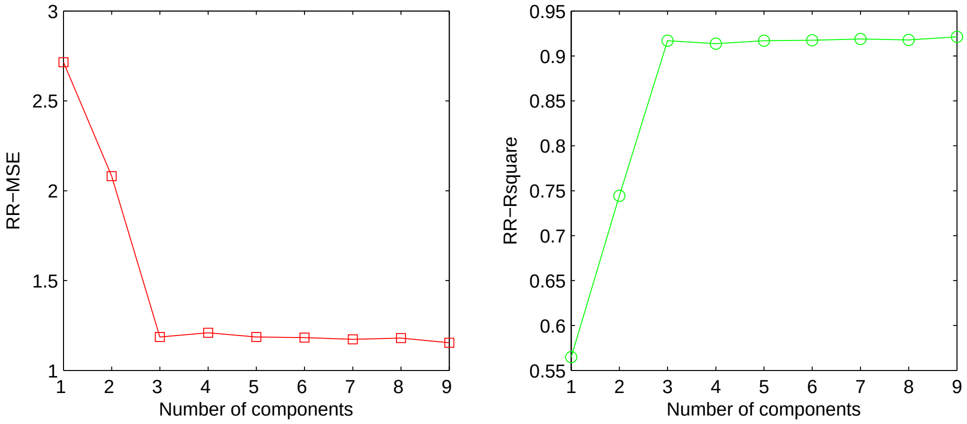


Figure 3.6: $RRMSE$ (left) and RR^2 (right) index plot for Fish data.

In order to select the number of components, $RRMSE$ and RR^2 criteria (Figure 3.6) introduced in Section 3.3 are used and they both indicate that $k = 3$ components are sufficient to perform PLS analysis which coincides the optimal number of components obtained in previous studies. The RoPLS1 analysis for $k = 3$ components gives the diagnostic plots in Figure 3.7. Here, bad leverage points (41, 43, 44), good leverage points (39, 40, 45), vertical outliers (1, 3, 10) and orthogonal outliers (2, 10 and 42) can be identified. Diagnostic plots based on SIMPLS method (Figure 3.8) does not detect all of the outliers effectively because it is not resistant to outlying observations.

High dimension: Biscuit-Dough Data

In this section, SIMPLS and RoPLS1 are applied on the well known chemometrics example from Osborne et al.([67]). Data set consists of four response variables (percentages of fat, sucrose, flour and water) of 40 biscuit dough samples. Although the original spectra had a wider range of 1100nm to 2498nm in steps of 2nm, i.e $p = 700$, only $p = 601$ (1200nm

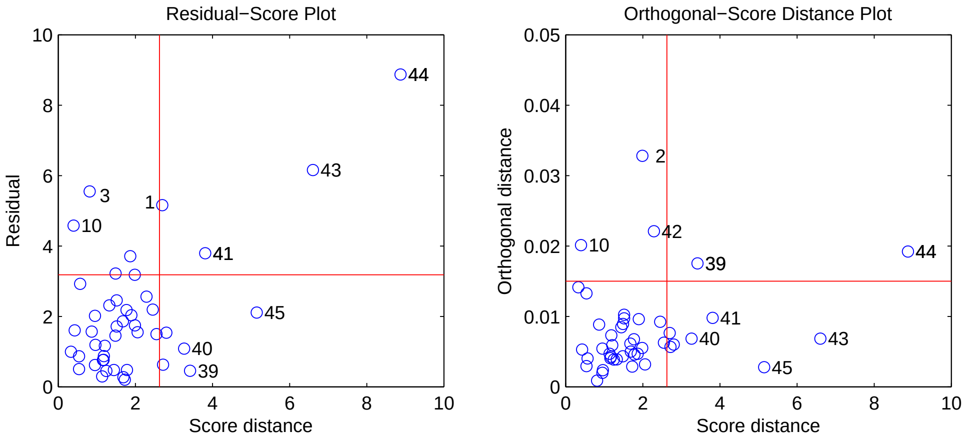


Figure 3.7: Diagnostic plots for Fish data (RoPLS1).

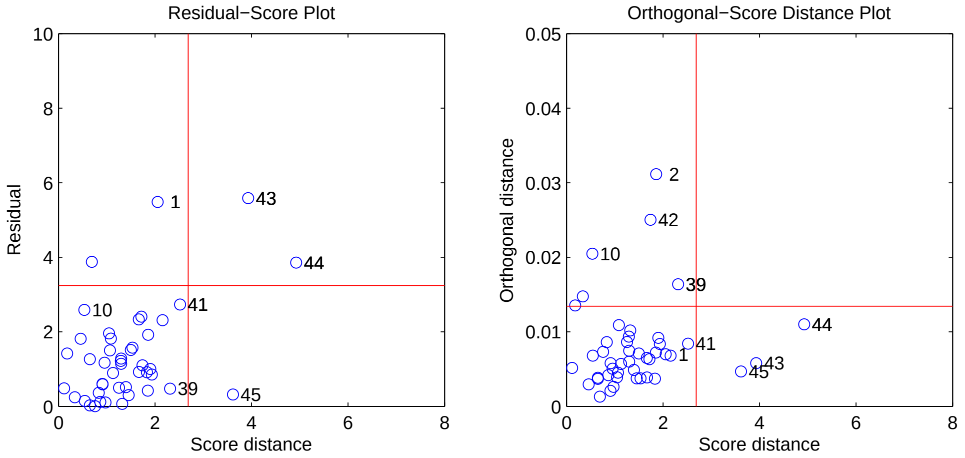


Figure 3.8: Diagnostic plots for Fish data (SIMPLS).

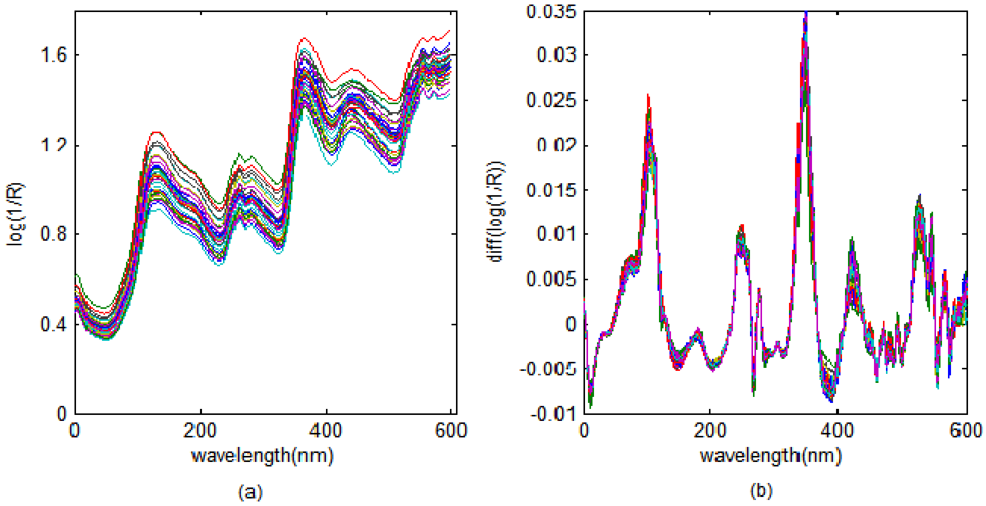


Figure 3.9: (a) Original Biscuit-dough data (b)Preprocessed Biscuit-dough data.

to 2400nm in steps of 2 nm) wavelengths are used since the channels at the ends are known to be less reliable. The purpose of our analysis is to predict percentage of water, based on the 40 NIR spectra with $p = 601$ predictors. In Figure 3.9 (a), there are $n = 40$ curves and each curve represents $p = 601$ predictors and the spectra have clearly shifted due to unequal particle sizes. Therefore, the preprocessing suggested by Marx and Eilers ([61]) is performed by differencing the columns of data matrix to eliminate sudden shifts, Figure 3.9(b), so the dimension of data is reduced to 600. Since observation 23 appears to be an outlier in most analyses, it is suggested to exclude this observation. To show robustness of RoPLS1, we carry out the analysis on the data matrix including observation 23.

Figure 3.10 displays the $RRMSE$ (left) and RR^2 (right) curves based on RoPLS1 and SIMPLS. $k = 3$ components are retained in the analysis and which yields an $RRMSE^{(3)}$

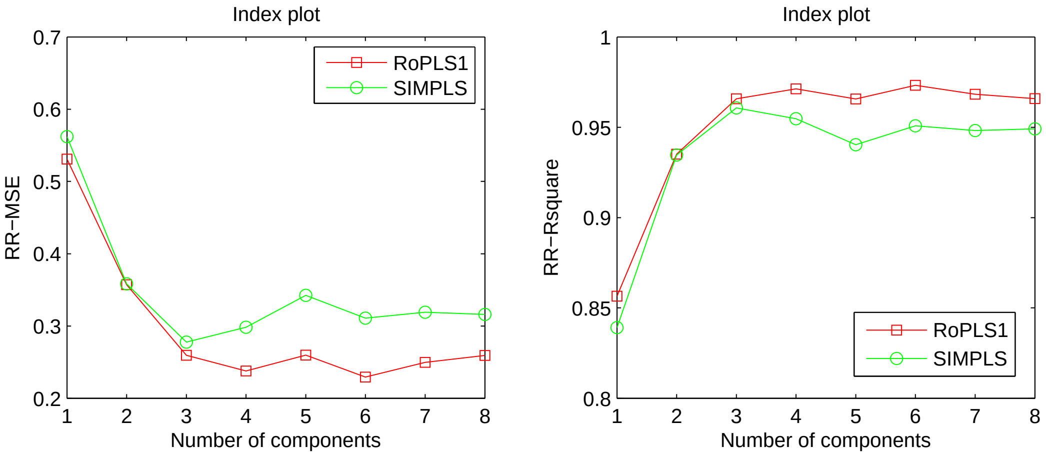


Figure 3.10: $RRMSE$ (left) and RR^2 (right) index plot for Biscuit-dough data.

value of 0.1626, whereas the optimal RMSE value for SIMPLS equals to 0.2322. Diagnostic plots of the data based on RoPLS revealed bad leverage points 7, 21, 23 and 24.

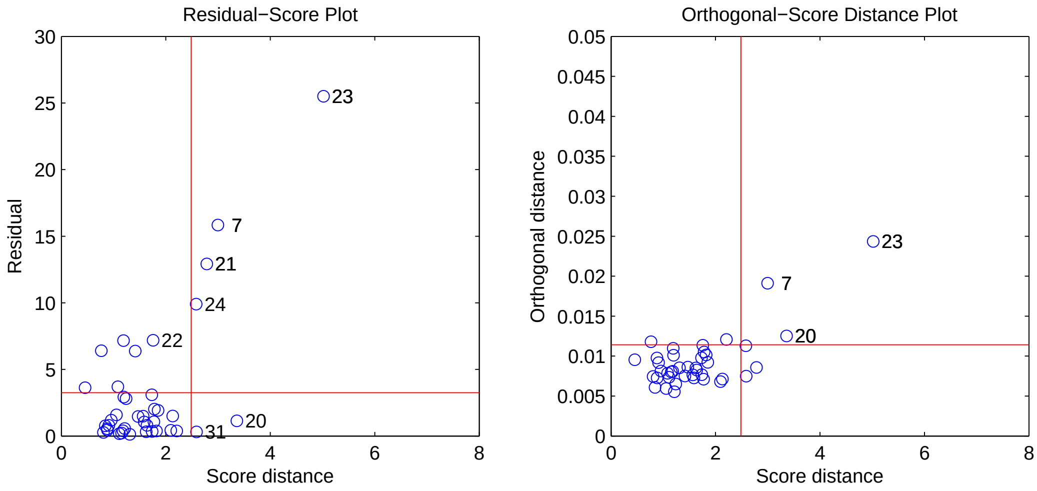


Figure 3.11: Diagnostic plots for Biscuit-dough data.

Chapter 4

Robustness Properties of RoPLS Estimator

4.1 Introduction

Robust statistical methods have emerged as a family of theories and techniques for estimating parameters of a parametric model while dealing with deviations from idealized assumptions. The deviations from strict parametric models include contamination of data by gross errors (outliers), rounding and grouping errors, and departure from an assumed sample distribution. Robust procedures aim to describe the structure best fitting to the bulk of the data and to identify deviating and highly influential data points by producing estimators that are not unduly affected by small departures from model assumptions. To enable the comparison of different robust methods in various situations, measures of performance are required. In general, three basic tools are utilized to determine the robustness of an estimator: qualitative, infinitesimal, and quantitative robustness ([40], [58], [73], [79], [86]).

The intuitive idea of robustness is that "*modifying a small proportion of observations causes only a small amount of change in the estimate*" and it is related to some form of continuity. The first tool, qualitative robustness is associated with continuity of the statistic (estimator) viewed by a functional, T, from $\mathscr{F}$, a set of all cumulative distribution functions for which T is defined, to parameter space Θ. Therefore an estimate, $\widehat{\theta}_n$, of $\theta \in \Theta$ based on a sample, $\mathbf{x}$=$[x_1, x_2, \ldots, x_n]^\top$ can be written as a functional, i.e., $T(F_n) = \widehat{\theta}_n$. In this section, notation $A^\top$ is used for transpose of matrix A, while the notation $(')$ is used as derivative operator. Here F_n is the empirical distribution function given by:

$$F_{\mathbf{X},n}(t) = \frac{1}{n}\sum_{i=1}^{n} I_{(-\infty,t]}(x_i)$$

where $I_A(x) = 1$ for $x \in A$ and 0 otherwise. Thus, if the purpose is to obtain an estimator that is relatively unaffected by small shifts in the cumulative distribution function, this can be achieved by choosing an estimator represented by a continuous functional and an estimator with this property is said to have "*qualitative robustness*".

The second tool for robustness arises when the two other restrictions, existence and boundedness of the functional derivative, are imposed so that small changes in the cumulative distribution do not result in large changes in the value of the functional, T. In order to theoretically assess the influence that an observation $\mathbf{z}^*$ has on the value of a statistical functional, the derivative of the functional called *influence function* is used ([40]).

Definition 4.1 *The influence function (IF) of a functional T at $F \in \mathscr{F}$ is given by*

$$IF(z^*;T,F) = \frac{d}{d\varepsilon}T(F_\varepsilon)\Big|_{\varepsilon=0} = \lim_{\varepsilon\downarrow 0}\frac{T(F_\varepsilon)-T(F)}{\varepsilon} = \lim_{\varepsilon\downarrow 0}\frac{T((1-\varepsilon)F+\varepsilon\delta_{z^*})-T(F)}{\varepsilon} \tag{4.1}$$

in those z^* where this limit exists. Here δ_{z^*} is point mass distribution at $\boldsymbol{z}^*$ and $\downarrow$ denotes the limit from the right.

Influence function given in (4.1) corresponds to the directional (Gateaux) derivative of T at F in the direction of $H = \delta_{z^*} - F$ and can also be written as:

$$IF(z^*;T,F) = T'(H) = \frac{d}{d\varepsilon}T(F+\varepsilon H)\mid_{\varepsilon=0} . \tag{4.2}$$

The functional, so the estimator, is said to have "*infinitesimal robustness*" if $IF(z^*; T, F)$ is a bounded function of $\mathbf{z}^*$. The influence function of T can also be used to determine an explicit formula for the asymptotic variance of T ([47]) since

$$Var(T, F) \approx \frac{\int IF(x; T, F)^\top IF(x; T, F) dF(x)}{n} \tag{4.3}$$

which can be estimated by

$$\widehat{Var}(T, F) = \frac{\sum_{i=1}^{n} IF(x_i; T, F_{\mathbf{X},n})^\top IF(x_i; T, F_{\mathbf{X},n})}{n^2}. \tag{4.4}$$

The third tool, *breakdown point*, addresses the notion of *quantitative robustness* and it can loosely be defined as the smallest fraction of samples (with respect to n) that can render the estimator useless. It describes how greatly a small change in the underlying distribution F changes the distribution of an estimator. The higher the breakdown point of an estimator, the more robust it is. In this chapter, finite-sample version of the breakdown point given in Definition 4.2 is preferred because of the simplicity.

Definition 4.2 *The finite-sample breakdown point, $\varepsilon^*(\boldsymbol{x}, T)$, of an estimator T at a sample $\boldsymbol{x}$=$[x_1, x_2, \ldots, x_n]^\top$ is given by*

$$\varepsilon^*(\boldsymbol{x}, T) = \frac{1}{n} \min_m \{m : sup_{\widetilde{\boldsymbol{x}}} \parallel T(\boldsymbol{x}) - T(\widetilde{\boldsymbol{x}}) \parallel = \infty\} \tag{4.5}$$

where $\widetilde{\boldsymbol{x}}$ is obtained by replacing m $(1 \leq m \leq n)$ observations of $\boldsymbol{x}$ by arbitrary observations.

The outline of this chapter is as follows. In Section 4.2 influence functions for classical PLSR estimator of β is given. Robustness properties of RoPLS estimator of β: influence function for low dimension, empirical influence curve for high dimensional case and finite-sample breakdown properties, are discussed in Section 4.3.

4.2 Influence Function for the SIMPLS Based Regression Estimator

In this section, influence function for the SIMPLS estimator of β is given. Assume that $\mathbf{x} \in \mathbb{R}^p$ and $y \in \mathbb{R}$ are centered random variables. Let F be a cumulative distribution function for a random vector $\gamma = (\mathbf{x}^\top, y)^\top$ and $d = p + 1$. Then the covariance matrix of γ is given by

$$\Sigma = \begin{pmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \sigma_{yy} \end{pmatrix}.$$

A functional, S, can be defined on a suitable class of probability distributions, $\mathscr{F}$, which maps an arbitrary distribution $G \in \mathscr{F}$ into a positive definite symmetric matrix, $S(G) = E_G(\gamma\gamma^\top)$, under the assumption that $S(F) = E_F(\gamma\gamma^\top) = \Sigma$, that is; S is Fisher-consistent for Σ at F. Using functional S, other functionals can also be defined as:

$$S_{xx}(G) = E_G(xx^\top) = [I_p : \mathbf{0}]S(G)[I_p : \mathbf{0}]^\top$$
$$S_{xy}(G) = E_G(xy) = [I_p : \mathbf{0}]S(G)e_d$$

where $\mathbf{0}$ is $p \times 1$ vector of zeros and $\mathbf{e}_d$ is the d^{th} standard basis vector of $\mathbb{R}^d$. Fisher consistency of S at F implies Fisher-consistency of S_{xy} at F for Σ_{xy}, that is $S_{xy}(F) = E_F(xy) = \Sigma_{xy}$. Similarly, $S_{yx}(G) = E_G(yx^\top) = S_{xy}(G)^\top$ and S_{yx} is also Fisher-consistent for Σ_{yx} at F. Since the existence of second moments guarantees the existence of the functional S, it is assumed that $\mathscr{F}$ consists of probability functions, G, at where positive definite matrix $E_G(\gamma\gamma^\top)$ exists which requires the existence of second moments.

Throughout this section, functional for an estimator is denoted by the parameter being estimated unless stated otherwise. $\lambda_h = \rho_h^2$ and τ_h stand for eigenvalues and eigenvectors of $\Sigma_{yx}^{h-1}\Sigma_{xy}^{h-1}$, respectively, satisfying $\Sigma_{xy}^{h-1}\tau_h = \rho_h r_h$ for $h = 1, 2, \ldots, k$. Since $S(F) = \Sigma$, $\lambda_h(F) = \lambda_h$, $\rho_h(F) = \rho_h$ and $r_h(F) = r_h$.

The functional for the SIMPLS estimator of β for h component is given by

$$\hat{\beta}_h(G) = R_h(G)R_h^\top(G)S_{xy} \tag{4.6}$$

where $R_h(G) = [r_1(G), r_2(G), \ldots, r_h(G)]$ and $G \in \mathscr{F}$. So, for the sake of clarity, influence function of $\hat{\beta}_h$ is studied in three steps. In the first step, influence functions of $S_{yx}^0 S_{xy}^0$ and the first PLS-weight vector, r_1, are derived. In the second step, similar approach is followed to find the influence function of h^{th} PLS-weight vector, r_h. Finally, in the third step, influence functions for scaled PLS-weight vector, $\tilde{r}_h$ and PLS slope estimator, $\hat{\beta}_h$, are given.

Step 1: Influence functions for $S_{yx}S_{xy}$ and r_1

Since $S_{xy}^0\tau_1 = \rho_1 r_1$, to determine the influence function of r_1, it is necessary to determine the influence functions of τ_1 and ρ_1 that require influence function of $S_{yx}^0 S_{xy}^0$. The following lemmas are used to obtain influence function for $S_{yx}^0 S_{xy}^0 = S_{yx}S_{xy}$.

Lemma 4.1 *The influence function of S at $F \in \mathscr{F}$ is*

$$IF(\gamma^*; S, F) = \gamma^*\gamma^{*\top} - \Sigma. \tag{4.7}$$

Corollary 4.1 *Let $\gamma^* = (\boldsymbol{u}^\top, v)^\top$ be an arbitrary d-dimensional point. Then,*

i. $IF(\gamma^*; S_{xy}, F) = uv - \Sigma_{xy} = IF(\gamma^*; S_{yx}, F)^\top$,

ii. $IF(\gamma^*; S_{xx}, F) = uu^\top - \Sigma_{xx}$,

iii. $IF(\gamma^*; S_{yx}S_{xy}, F) = 2[vu^\top \Sigma_{xy} - \Sigma_{yx}\Sigma_{xy}]$.

Proof:

i. $IF(\gamma^*; S_{xy}, F) = [I_p : \mathbf{0}]IF(\gamma^*; S, F)e_d = uv - \Sigma_{xy}$,

$IF(\gamma^*; S_{yx}, F) = e_d^\top S(F)[I_p : \mathbf{0}]^\top = vu^\top - \Sigma_{yx} = IF(\gamma^*; S_{xy}, F)^\top$,

ii. $IF(\gamma^*; S_{xx}, F) = [I_p : \mathbf{0}]IF(\gamma^*; S, F)[I_p : \mathbf{0}]^\top = uu^\top - \Sigma_{xx}$,

iii. $IF(\gamma^*; S_{yx}S_{xy}, F) = IF(\gamma^*; S_{yx}, F)\Sigma_{xy} + \Sigma_{yx}IF(\gamma^*; S_{xy}, F)$,

$$= \{vu^\top - \Sigma_{yx}\}\Sigma_{xy} + \Sigma_{yx}\{uv - \Sigma_{xy}\} = 2[vu^\top \Sigma_{xy} - \Sigma_{yx}\Sigma_{xy}] \qquad \square.$$

For the influence functions of the eigenvalue and the eigenvector of $S_{yx}S_{xy}$ (ρ_1 and τ_1), the following lemmas are used.

Lemma 4.2 *(Sibson, [77]) Let A and B be two symmetric matrices, λ_i be the i^{th} simple eigenvalue of A and v_i be the associated eigenvector of unit length. Let A be perturbed to*

$$A(\varepsilon) = A + \varepsilon B + o(\varepsilon^2)$$

and assume that the corresponding perturbations of λ_i and v_i are

$$\lambda_i(\varepsilon) = \lambda_i + \varepsilon\widetilde{\lambda_i} + o(\varepsilon^2)$$
$$v_i(\varepsilon) = v_i + \varepsilon\widetilde{v_i} + o(\varepsilon^2).$$

Then, $\widetilde{\lambda_i} = v_i^\top B v_i$ and $\widetilde{v_i} = -(A - \lambda_i I)^+ B v_i = -[\sum_{j\neq i}(\frac{v_j v_j^\top}{\lambda_j - \lambda_i})]Bv_i$.

The following lemma mimics Lemma 4.2 and provides the influence functions of eigenvalues and corresponding eigenvectors of a positive definite symmetric (PDS) matrix, Λ.

Lemma 4.3 *(Croux and Haesbroeck, [12]) Let $C : \mathscr{F} \rightarrow PDS(m)$ be a statistical functional, F a m-dimensional distribution and $\gamma \sim F$. Suppose that $IF(\gamma^*; C, F)$ exists and $C(F)$=Λ. Denote the i^{th} simple eigenvalue and eigenvector of Λ by λ_i and τ_i, respectively. Then influence functions of functionals λ_i and τ_i at $F \in \mathscr{F}$ are given by*

$$IF(\gamma^*; \lambda_i, F) = \tau_i^\top IF(\gamma^*; C, F)\tau_i \tag{4.8}$$

$$IF(\gamma^*; \tau_i, F) = \left[\sum_{j\neq i}(\frac{\tau_j\tau_j^\top}{\lambda_i - \lambda_j})\right] IF(\gamma^*; C, F)\tau_i \tag{4.9}$$

Proof: The perturbation of Λ, $\Lambda(\varepsilon) = C(F_\varepsilon)$, can be approximated as:

$$C(F_\varepsilon) \approx C(F) + \textstyle\int IF(\gamma^*; C, F)d(F_\varepsilon - F) + o(\varepsilon^2) = \Lambda + \varepsilon IF(\gamma^*; C, F) + o(\varepsilon^2)$$

([47] and [54]). Similarly λ_i and τ_i can be written as

$$\begin{aligned}\lambda_i(F_\varepsilon) &= \lambda_i + \varepsilon\widetilde{\lambda_i} + o(\varepsilon^2) = \lambda_i + \varepsilon IF(\gamma^*; \lambda_i, F) + o(\varepsilon^2)\\ \tau_i(F_\varepsilon) &= \tau_i + \varepsilon\widetilde{\tau_i} + o(\varepsilon^2) = \tau_i + \varepsilon IF(\gamma^*; \tau_i, F) + o(\varepsilon^2),\end{aligned}$$

where $\widetilde{\lambda_i}$ and $\widetilde{\tau_i}$ are obtained from Lemma 4.2 as,

$$\begin{aligned}\widetilde{\lambda_i} &= IF(\gamma^*; \lambda_i, F) = \tau_i^\top IF(\gamma^*; C, F)\tau_i\\ \widetilde{\tau_i} = IF(\gamma^*; \tau_i, F) &= -\left[\textstyle\sum_{j\neq i}(\frac{\tau_j\tau_j^\top}{\lambda_j - \lambda_i})\right] IF(\gamma^*; C, F)\tau_i = \left[\textstyle\sum_{j\neq i}(\frac{\tau_j\tau_j^\top}{\lambda_i - \lambda_j})\right] IF(\gamma^*; C, F)\tau_i\end{aligned}$$

$\square$.

Lemma 4.3 demonstrates that once $IF(\gamma^*; S_{yx}S_{xy}, F)$ is known, influence functions for all PLS-weight vectors, and as a result of this, influence function of the PLS slope estimator can be determined. Using Lemma 4.3, it can be easily seen that influence function for τ_1, the eigenvector of $S_{yx}S_{xy}$, is 0. Since $\tau_1 = 1$, influence function for λ_1 can be given as

$$IF(\gamma^*; \lambda_1, F) = IF(\gamma^*; S_{yx}S_{xy}, F). \tag{4.10}$$

Using $IF(\gamma^*; S_{yx}S_{xy}, F)$ given in Corollary 4.1 and using $\Sigma_{xy} = \sqrt{\lambda_1} r_1$, (4.10) can be rewritten as

$$IF(\gamma^*; \lambda_1, F) = 2[vu^\top \Sigma_{xy} - \Sigma_{yx}\Sigma_{xy}] = 2\sqrt{\lambda_1}[vu^\top r_1 - \sqrt{\lambda_1}]. \tag{4.11}$$

Furthermore, $\rho_1 = \sqrt{\lambda_1}$ so (4.11) can be written in terms of ρ_1 as

$$IF(\gamma^*; \lambda_1, F) = IF(\gamma^*; \rho_1^2, F) = [\frac{d}{d\rho_1}\rho_1^2] IF(\gamma^*; \rho_1, F) = 2\rho_1[vu^\top r_1 - \rho_1] \tag{4.12}$$

which implies

$$IF(\gamma^*; \rho_1, F) = vu^\top r_1 - \rho_1 = vr_1^\top u - \rho_1. \tag{4.13}$$

The functional for the first PLS weight vector, r_1, satisfies $S_{xy}(F)\tau_1(F) = \rho_1(F)r_1(F)$, because it is the left singular vector of S_{xy}. Hence, influence function for r_1, given next, follows immediately from $IF(\gamma^*; S_{xy}, F) = IF(\gamma^*; \rho_1, F)r_1 + \rho_1 IF(\gamma^*; r_1, F)$.

Corollary 4.2 *The influence function for the first weight vector, r_1 at $F \in \mathscr{F}$ is*

$$IF(\gamma^*; r_1, F) = \frac{v}{\rho_1}[I_p - r_1 {r_1}^\top]u. \tag{4.14}$$

It can be seen from (4.14) that $IF(\gamma^*; r_1, F)$ is unbounded since it is a function of arbitrary values of $\mathbf{u}$ and v.

Step 2: Influence functions for $S_{yx}^{h-1} S_{xy}^{h-1}$ and r_h

In general, for $h > 1$ components, the h^{th} PLS weight vector is obtained as the left singular vector of ${\Sigma_{xy}}^{h-1}$. Thus, the functional for the h^{th} PLS weight vector, r_h, satisfies $S_{xy}^{h-1}(F)\tau_h(F) = \rho_h(F) r_h(F)$ where $S_{xy}^{h-1} = [I_p - V_{h-1}V_{h-1}^{\top}]S_{xy}$ with $S_{xy}^{0} = S_{xy}$. Here, the columns of $V_{h-1} = [v_1, v_2, \ldots, v_{h-1}]$ form an orthonormal basis for the space spanned by $[\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_{h-1}]$ where $\mathbf{p}_i \propto \Sigma_{xx} r_i$ for $i = 1, 2 \ldots, h-1$.

Influence function for $S_{yx}^{h-1} S_{xy}^{h-1}$, which is same as the influence function for λ_h, is:

$$IF(\gamma^*; S_{yx}^{h-1} S_{xy}^{h-1}, F) = IF(\gamma^*; \lambda_h, F) = IF(\gamma^*; S_{yx}[I_p - \Upsilon_{h-1}]S_{xy}, F) \tag{4.15}$$

where $\Upsilon_{h-1} = V_{h-1}V_{h-1}^{\top} = \sum_{i=1}^{h-1} v_i v_i^{\top}$. Using multiplication rule, (4.15) can be rewritten as

$$IF(\gamma^*; \lambda_h, F) = 2IF(\gamma^*; S_{yx}, F)[I_p - \Upsilon_{h-1}]\Sigma_{xy} - \Sigma_{yx} IF(\gamma^*; \Upsilon_{h-1}, F)\Sigma_{xy}. \tag{4.16}$$

Since $2\rho_h IF(\gamma^*; \rho_h, F) = IF(\gamma^*; \lambda_h, F)$, after plugging $IF(\gamma^*; S_{yx}, F)$ in (4.16), influence function for ρ_h can be derived as

$$IF(\gamma^*; \rho_h, F) = v r_h^{\top} u - \rho_h - \tfrac{1}{2\rho_h}\Sigma_{yx} IF(\gamma^*; \Upsilon_{h-1}, F)\Sigma_{xy}.$$

The next corollary entails the influence function for r_h that requires the use of $IF(\gamma^*; S_{xy}^{h-1}, F)$ and $IF(\gamma^*; \rho_h, F)$.

Corollary 4.3 *The influence function for the h^{th} $(h > 1)$ weight vector, r_h, at $F \in \mathscr{F}$ is*

$$IF(\gamma^*; r_h, F) = \frac{1}{\rho_h}\left\{[I_p - r_h {r_h}^{\top}]vu - [I_p - \frac{r_h}{2\rho_h}\Sigma_{yx}]IF(\gamma^*; \Upsilon_{h-1}, F)\Sigma_{xy} - \Upsilon_{h-1} uv\right\}. \tag{4.17}$$

Here, $IF(\gamma^*; \Upsilon_{h-1}, F)$ can be calculated recursively.

This function is obtained directly by using the equality

$$IF(\gamma^*; S_{xy}^{h-1}, F) = IF(\gamma^*; \rho_h, F) r_h + IF(\gamma^*; r_h, F) \rho_h.$$

Therefore, influence function for r_h given in (4.17) depends on explicitly to arbitrary point $(\mathbf{u}^\top, v)^\top$ as well as implicitly to the influence functions for all previous PLS-weight vectors. So, $IF(\gamma^*; r_h, F)$ in (4.17) is unbounded.

Step 3: Influence functions for $\tilde{r}_h$ and $\hat{\beta}_h$

The PLS weight vectors, r_h, for $h \geq 1$ should be scaled by functional $\sqrt{r_h^\top S_{xx} r_h}$ to be able make the h^{th} component unit norm. The scaled version of the PLS-weight vector, denoted by $\tilde{r_h}$, has the influence function in the form of

$$IF(\gamma^*; \tilde{r_h}, F) = \frac{[\Psi(h) I_p - r_h {r_h}^\top \Sigma_{xx}] IF(\gamma^*; r_h, F)}{\Psi(h)^{3/2}} - \frac{r_h {r_h}^\top IF(\gamma^*; S_{xx}, F) r_h}{2\Psi(h)^{3/2}} \tag{4.18}$$

where $\Psi(h) = {r_h}^\top \Sigma_{xx} r_h$ for $h \geq 1$.

Corollary 4.4 *Influence function for the PLS slope estimator, represented by functional $\hat{\beta}_h = \widetilde{R_h} \widetilde{R_h}^\top S_{xy}$ with $\widetilde{R_h} = [\tilde{r_1}, \tilde{r_2}, \ldots, \tilde{r_h}]$ for h components at $F \in \mathscr{F}$ is*

$$IF(\gamma^*; \hat{\beta}_h, F) = IF(\gamma^*; \widetilde{R_h}, F) \widetilde{R_h}^\top \Sigma_{xy} + \widetilde{R_h} IF(\gamma^*; \widetilde{R_h}, F)^\top \Sigma_{xy} + \widetilde{R_h} \widetilde{R_h}^\top IF(\gamma^*; S_{xy}, F) \tag{4.19}$$

where i^{th} column of $IF(\gamma^*; \widetilde{R_h}, F)$ is $IF(\gamma^*; \tilde{r}_i, F)$ for $h \geq 1$.

Similarly, $\tilde{r_h}$ in (4.18) is unbounded because it depends on the influence function for r_h which is unbounded. Finally, $\hat{\beta}_h$ has unbounded influence function. The following example ([82]) demonstrates the unboundedness of the influence functions of SIMPLS estimators.

Example 4.1

Assume that $\gamma \sim N_3(\mu, \Sigma)$ where $\mu = (0\ 0\ 0)^\top$ and

$$\Sigma = \begin{pmatrix} 5 & 0.5 & 3 \\ 0.5 & 2 & 0.3\bar{3} \\ 3 & 0.3\bar{3} & 2 \end{pmatrix}.$$

The norms of the theoretical influence functions for r_1 given in (4.14) and $\hat{\beta}_1 = [\hat{\beta}_{11}, \hat{\beta}_{12}]^\top$ given in (4.19) are calculated for $\gamma^* = (\mathbf{u}^\top, v)^\top$ where $\mathbf{u}^\top = (i, 0)$ and $v = j$ with i and j take values between -10 to 10. The unbounded shapes of the influence functions, can be seen in Figure (4.1). Hence, it can be concluded that the SIMPLS estimator of slope is non-robust towards outlying observations.

4.3 Robustness of the RoPLS Estimator of β

SIMPLS regression is scale and orthogonal equivariant method ([43]). The first version of BACON algorithm is scale and affine equivariant, therefore orthogonal equivariant ([5]). Equivariance properties of BACON and SIMPLS guarantee the scale and orthogonal equivariance of the entire RoPLS1 , BACON based RoPLS, estimator of β. Therefore, RoPLS1 estimator, $\widetilde{\beta}$, that is computed from a transformed response vector $\widetilde{y} = \alpha y$ ($\alpha \in \mathbb{R} - \{0\}$) and data matrix $\widetilde{X} = XA$ (A: $p \times p$ orthogonal matrix), is given by

$$\widetilde{\beta} = \alpha A^\top \widehat{\beta} \tag{4.20}$$

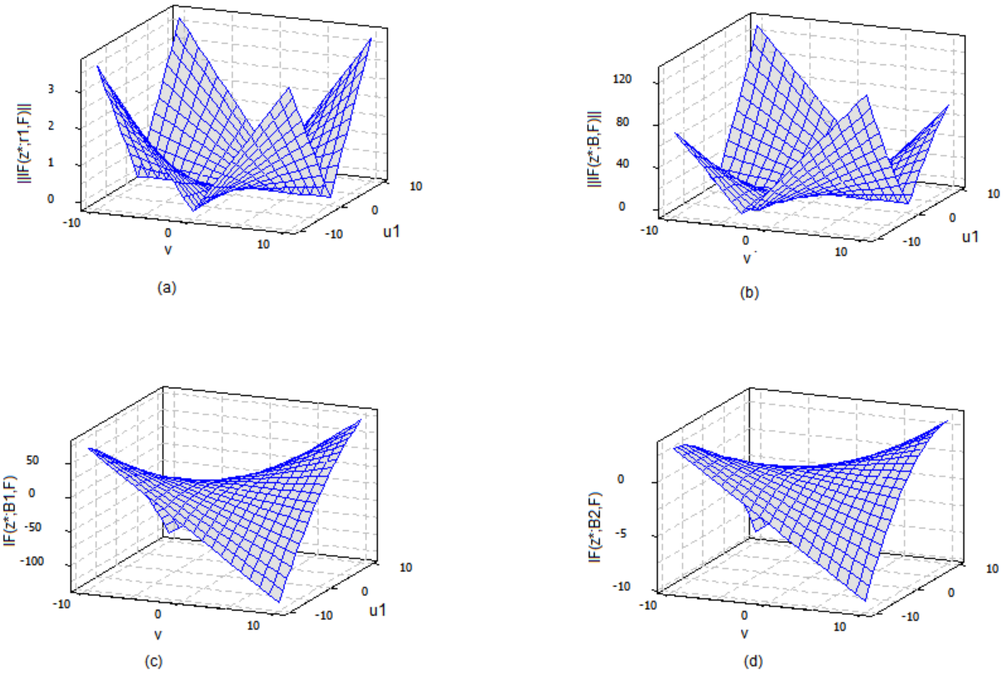


Figure 4.1: (a) The norm of IF of the r_1 (b) The norm of IF of the $\hat{\beta}_1$ (c) The IF of the $\hat{\beta}_{11}$ (d) The IF of the $\hat{\beta}_{12}$.

where $\widehat{\beta}$ is RoPLS1 estimator that is computed from original data matrix, X, and response vector, y. However, (4.20) does not hold for RoPLS2, PCOUT based RoPLS, estimator since the coordinatewise median is not orthogonal equivariant. Because of its equivariance advantage over RoPLS2, only robustness properties of RoPLS1 estimator are explored in this section.

4.3.1 Influence functions for Low Dimension

RoPLS1 algorithm, introduced in Chapter 3, is an iterative algorithm that starts with an initial estimator. This section is presented into two parts. In the first part, influence functions for the initial estimators of PLS-weight vector and slope are derived for $h \geq 1$ components. Then, in the second part, after deriving influence functions for the estimators of PLS-weight vector and slope obtained in i^th iteration, it is demonstrated that influence functions in the following iterations are directly related to influence function of the initial estimators where $0 \leq i \leq a$ and a is the number at which iteration converges.

In general, functional for the h component RoPLS1 estimator of β obtained in the i^{th} iteration is denoted by $\hat{\beta}_h^{(i)}$, while the estimators of r_h, λ_h and ρ_h are represented by functionals $r_h^{(i)}$, $\lambda_h^{(i)}$, and $\rho_h^{(i)}$, respectively.

Part I: Influence Function for the Initial Estimator, $\hat{\beta}_h^{(0)}$

Influence function of the initial estimator $\hat{\beta}_h^{(0)}$ is derived in three steps similar as in Section 4.2.

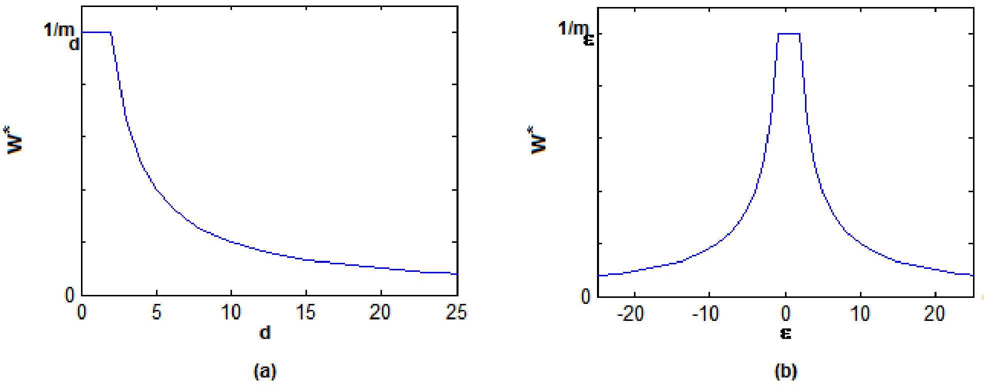


Figure 4.2: (a) Graph of w^* for robust distances (b) Graph of w^* for scaled errors, ε.

Step 1: Influence functions for $S_{wyx}S_{wxy}$ and $r_1^{(0)}$

A functional, $S_{w(0)}$, can be defined as $S_{w(0)}(G) = E_G(w_G^{(0)}(\gamma)\gamma\gamma^\top)$ for a random vector $\gamma{=}(\mathbf{x}^\top, y)^\top$ and $G \in \mathscr{F}$, where $\mathscr{F}$ is the same class of d-variate probability distributions defined in Section 4.2. The weight functional, $w_G^{(0)}(\gamma)$, is

$$w_G^{(0)}(\gamma) = w^*\left(d_\gamma{}^B(G)\right) = w^*\left(\sqrt{\gamma^\top S_B(G)^{-1}\gamma}\right) \tag{4.21}$$

where S_B is the functional representative of the BACON estimator of covariance matrix of γ (Σ) with range of positive definite matrices and $w^*(:)$ is the function given in (3.11).

The weight function, $w_G^{(0)}$, is decreasing function of the distance which can be seen from Figure 4.2 (a) with $m_d(G) = max\left\{1, median(d_\gamma{}^B(G))\right\}$. Furthermore, $0 \le w_G^{(0)}(\gamma) \le 1$ for any $G \in \mathscr{F}$. Existence of $S(G) = E_G(\gamma\gamma^\top)$ implies existence of $S_{w(0)}(G)$, since $w_G^{(0)}(\gamma)$ is bounded for $G \in \mathscr{F}$. Therefore, functionals $S_{wxy(0)}(G) = E_G\left(w_G^{(0)}(\gamma)xy\right)$ and $S_{wxx(0)}(G) =$

$E_G\left(w_G^{(0)}(\gamma)xx^\top\right)$ exist, and for $h > 0$, $S_{wxy(0)}^{h-1}(G) = \rho_h^{(0)}(G)r_h^{(0)}(G)$. Moreover, $S_{w(0)}(G)$ is a non-negative matrix because of the fact that $0 \leq w_G^{(0)}(\gamma)$.

Throughout this section, the following are assumed to be held for $i \geq 0$:

1. $S_{w(i)}(F) = \Sigma$,

2. $S_B(F) = \Sigma$ and $IF(\gamma^*; {S_B}^{-1}, F)$ exists with a boundary,

3. $P_F\{\gamma : q^\top\gamma \neq 0 \text{ and } w_F^{(i)}(\gamma) > 0\} > 0$,

4. $P_F\{\gamma : \sqrt{\gamma^\top\Sigma^{-1}\gamma} = m_d(F)\} = 0$.

Assumption (*1*) implies $S_{wxy(i)}(F) = \Sigma_{xy}$, $S_{wxx(i)}(F) = \Sigma_{xx}$, $r_h^{(i)}(F) = r_h$, $\lambda_h^{(i)}(F) = \lambda_h$, $\rho_h^{(i)}(F) = \rho_h$. When assumption (*2*) holds, ${d_\gamma}^{B'}(H)$ exists with boundary. If for every $q \in \mathbb{R}^d$, the probability of $\{\gamma : q^\top\gamma \neq 0 \text{ and } w_G^{(i)}(\gamma) > 0\}$ is positive under G, then $q^\top S_{w(i)}(G)q > 0$ implying that $S_{w(i)}(G)$ is positive definite. Therefore, assumption (*3*) is needed for positive definiteness of $S_{w(i)}(G)$. The last assumption is required for differentiability of $w_G^{(0)}$ over the support of the random vector γ.

Similar to Section 4.2, the influence functions of $S_{wyx}S_{wxy} = S_{wyx(0)}^0 S_{wxy(0)}^0$ and the influence function for ${\rho_1}^{(0)}$ are used to obtain influence function of $r_1^{(0)}$. The notation $M'(H)$ is used to denote $IF(\gamma^*; M, F)$ where $H = \delta_{\gamma^*} - F$.

Lemma 4.4 *Influence function of* $S_{wyx}S_{wxy}$ *at* $F \in \mathscr{F}$ *is*

$$IF(\gamma^*; S_{wyx}S_{wxy}, F) = 2w_F^{(0)}(\gamma^*)u^\top v\Sigma_{xy} - 2\Sigma_{yx}\Sigma_{xy} + 2C^\top\Sigma_{xy} \tag{4.22}$$

where $C = E_F\left(\frac{\partial w^*}{\partial {d_\gamma}^B(F)}{d_\gamma}^{B'}(H)xy\right)$.

Proof:

The influence function for S_{wxy} is:

$$S_{wxy}{}'(H) = \frac{\partial}{\partial t} E_{F+tH}\left(w^{(0)}_{F+tH}(\gamma)xy\right)|_{t=0}= E_F\left(w^{(0)'}(H)xy\right) + \left\{w^{(0)}_F(\gamma^*)uv - \Sigma_{xy}\right\}$$

where

$$w^{(0)'}(H) = \frac{\partial}{\partial t} w^{(0)}_{F+tH}(\gamma)\,|_{t=0}= \frac{\partial}{\partial t} w^*\left(d_\gamma{}^B(F+tH)\right)|_{t=0}= \frac{\partial w^*}{\partial d_\gamma{}^B(F+tH)}\frac{\partial d_\gamma{}^B(F+tH)}{\partial t}\,|_{t=0}.$$

Therefore,

$$S_{wxy}{}'(H) = \left\{w^{(0)}_F(\gamma^*)uv - \Sigma_{xy}\right\} + E_F\left(\frac{\partial w^*}{\partial d_\gamma{}^B(F)} d_\gamma{}^{B'}(H)xy\right)$$

with the derivative

$$\frac{\partial w^*(d)}{\partial d_\gamma{}^B(F)} = \begin{cases} 0; & d < m_d(F) \\ -(d^{-2}); & d > m_d(F)\}. \end{cases}$$

The derivative exists everywhere except $d = m_d(F)$ and it is bounded. Assumption (*4*) guarantees the existence of the derivative over the support of γ, whereas the assumption (*2*) guarantees the boundedness of $d_\gamma{}^{B'}(H)$. Therefore $E_F\left(\frac{\partial w^*}{\partial d_\gamma{}^B(F)} d_\gamma{}^{B'}(H)xy\right)$ exists. Let $C = E_F\left(\frac{\partial w^*}{\partial d_\gamma{}^B(F)} d_\gamma{}^{B'}(H)xy\right)$ then;

$$S_{wyx}S_{wxy}{}'(H) = S_{wyx}{}'(H)S_{wxy}(F) + S_{wyx}(F)S_{wxy}{}'(H)$$

which is equal to

$$\left\{w^{(0)}_F(\gamma^*)u^\top v - \Sigma_{yx}\right\}\Sigma_{xy} + C^\top\Sigma_{xy} + \Sigma_{yx}\left\{w_F(\gamma^*)uv - \Sigma_{xy}\right\} + \Sigma_{yx}C$$

and this can be simplified as

$$S_{wyx}S_{wxy}{}'(H) = 2w^{(0)}_F(\gamma^*)u^\top v\Sigma_{xy} - 2\Sigma_{yx}\Sigma_{xy} + 2C^\top\Sigma_{xy}\ \square.$$

By Lemma 4.3, the influence function of $\lambda_1{}^{(0)} = \rho_1^{(0)^2}$ is the same as the influence function of $S_{wyx}S_{wxy}$ and $\Sigma_{xy} = \rho_1 r_1$, thus the influence function of $\lambda^{(0)}_1$ is given as

$$\lambda^{(0)'}_1(H) = 2\rho_1\rho^{(0)'}_1(H) = 2\rho_1\left\{w^{(0)}_F(\gamma^*)u^\top v r_1 - \rho_1 + C^\top r_1\right\}$$

and this yields the following

$$\rho_1^{(0)'}(H) = w_F^{(0)}(\gamma^*)u^\top v r_1 - \rho_1 + C^\top r_1 = w_F^{(0)}(\gamma^*)v {r_1}^\top u - \rho_1 + {r_1}^\top C.$$

The following lemma gives the influence function for $r_1^{(0)}$.

Lemma 4.5 *The influence function of* $r_1^{(0)}$ *at* $F \in \mathscr{F}$ *is*

$$IF(\gamma^*; r_1^{(0)}, F) = \frac{1}{\rho_1}[1 - r_1 {r_1}^\top][w_F^{(0)}(\gamma^*)vu + C] \tag{4.23}$$

Proof:

Using the equality that $S_{wxy}(F+tH) = \rho_1^{(0)}(F+tH)r_1^{(0)}(F+tH)$ and taking the derivatives with respect to t for both sides, the following is obtained

$$S'_{wxy}(H) = \rho_1^{(0)'}(H)r_1 + \rho_1 r_1^{(0)'}(H). \tag{4.24}$$

After plugging $S'_{wxy}(H)$ and $\rho_1^{(0)'}(H)$ in (4.24), $r_1^{(0)'}(H)$ can be written as

$$r_1^{(0)'}(H) = \frac{1}{\rho_1}\left\{w_F^{(0)}(\gamma^*)uv - \Sigma_{xy} + C - \left\{w_F^{(0)}(\gamma^*)v{r_1}^\top u - \rho_1 + {r_1}^\top C\right\} r_1\right\}$$

and after simplifications, the influence function for $r_1^{(0)}$ is obtained as

$$r_1^{(0)'}(H) = \frac{1}{\rho_1}[I_p - r_1 {r_1}^\top][w_F^{(0)}(\gamma^*)vu + C]$$

□.

The influence function of $r_1^{(0)}$, given in (4.23), consists of two parts. The first part, $\frac{1}{\rho_1}[I_p - r_1{r_1}^\top]w_F^{(0)}(\gamma^*)vu$, is directly related to the influence function of classical PLS weight vector given in (4.14). The main difference between (4.14) and (4.23) is the presence of additional weight vector, $w_F^{(0)}(\gamma^*)$, in (4.23) which allows the first term of (4.23) to be bounded. The second term, $\frac{1}{\rho_1}[I_p - r_1{r_1}^\top]C$, is obviously independent of γ^*. Since C exists, the second term is a finite valued vector for given F. The following example demonstrates how $w_F(\gamma^*)$ gets smaller for extreme observations, so that it makes the first part of the ${r_1^{(0)}}'(H)$ bounded.

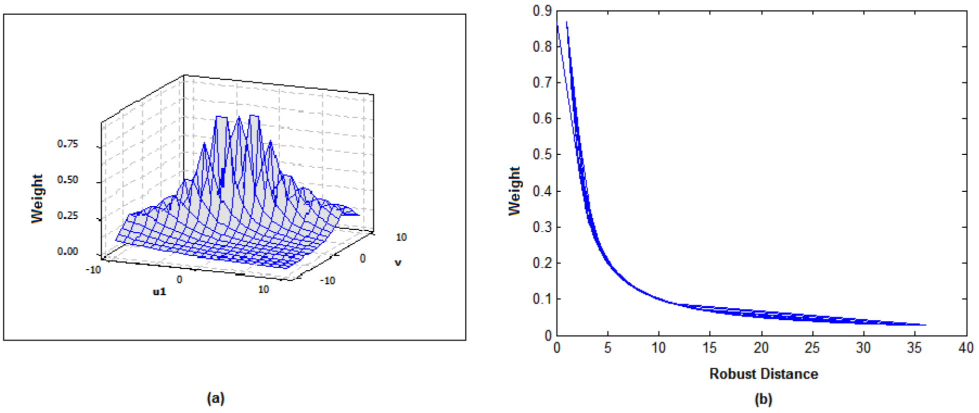
Example 4.2

Consider the same setting given in Example 4.1. $\gamma \sim N_3(\mu, \Sigma)$ where $\mu = (0\ 0\ 0)^\top$. In this example, it is going to be shown that $w_F^{(0)}(\gamma^*)$ is small for extreme $\gamma^* = (\mathbf{u}^\top, v)^\top$. Since, $\gamma^\top\Sigma^{-1}\gamma$ is distributed with χ^2 distribution with degrees of freedom $rank(\Sigma\Sigma^{-1}) = 3$, the $\sqrt{\gamma^\top\Sigma^{-1}\gamma}$ has chi-distribution with degrees of freedom 3 and median is approximately 1.15, i.e., $m_d = median(\sqrt{\gamma'\Sigma^{-1}\gamma}) \approx 1.15$. The weight function is calculated for γ^* where $\mathbf{u}^\top = (i, 0)$ and $v = j$ with i and j take values between -10 to 10. Figure 4.3 shows the behavior of weights for different γ^* values. Obviously, $w_F^{(0)}(\gamma^*)$ takes values close to zero for extreme values of γ^*.

Step 2: Influence functions for $S^{h-1}_{wyx(0)}S^{h-1}_{wxy(0)}$ and $r_h^{(0)}$

The functional for the h^{th} PLS weight vector, $r_h^{(0)}$, satisfies $S^{h-1}_{wxy(0)}(F) = \rho_h^{(0)}(F)r_h^{(0)}(F)$ where $S^{h-1}_{wxy(0)} = [I_p - V^{h-1}_{w(0)}{V^{h-1}_{w(0)}}^\top]S_{wxy}$. Here, the columns of $V^{h-1}_{w(0)}$ form an orthonormal basis for the space spanned by $\{p_1^{(0)}, p_2^{(0)}, \ldots, p_{h-1}^{(0)}\}$. Therefore, the influence function for $S^{h-1}_{wxy(0)}$ is:

$$S^{h-1}_{wxy(0)}{}'(H) = -{\Upsilon^{h-1}_{w(0)}}'(H)\Sigma_{xy} + [I_p - \Upsilon^{h-1}]{S_{wxy}}'(H) \qquad (4.25)$$



(a)

(b)

Figure 4.3: (a) Graph of $w_F^{(0)}(\gamma^*)$ versus (u_1,v) (b) Graph of $w_F^{(0)}(\gamma^*)$ versus robust distances.

with $\Upsilon_{w(0)}^{h-1} = V_{w(0)}^{h-1} V_{w(0)}^{h-1\top}$ and ${\Upsilon_{w(0)}}^{h-1}(F) = \Upsilon^{h-1}$. Similarly, using multiplication rule, the influence function for $S_{wyx(0)}^{h-1} S_{wxy(0)}^{(h-1)} = S_{wyx}[I_p - \Upsilon_{w(0)}^{h-1}]S_{wxy}$ is obtained as

$$2\rho_h \left\{ w_F^{(0)}(\gamma^*) v {r_h}^\top u - \rho_h + C^\top r_h - \tfrac{1}{2\rho_h}\Sigma_{yx}\Upsilon_{w(0)}^{h-1\,\prime}(H)\Sigma_{xy} \right\}$$

which yields the influence function of $\rho_h^{(0)}$ as

$$\rho_h^{(0)\prime}(H) = \left\{ w_F^{(0)}(\gamma^*) v {r_h}^\top u - \rho_h + C^\top r_h - \frac{1}{2\rho_h}\Sigma_{yx}\Upsilon_{w(0)}^{h-1\,\prime}(H)\Sigma_{xy} \right\}. \tag{4.26}$$

So, the general form for the influence functions of $r_h^{(0)}$ is given in the next lemma.

Lemma 4.6 *The influence function of $r_h^{(0)}$ at $F \in \mathscr{F}$ is*

$$\begin{aligned} r_h^{(0)\prime}(H) = \tfrac{1}{\rho_h}\Big\{ &[I_p - r_h {r_h}^\top][v w_F^{(0)}(\gamma^*) u + C] - [I_p - \tfrac{r_h}{2\rho_h}\Sigma_{yx}]\Upsilon_{w(0)}^{h-1\,\prime}(H)\Sigma_{xy} \Big\} \\ &- \tfrac{1}{\rho_h}\left\{ \Upsilon^{h-1}[w_F^{(0)}(\gamma^*) u v + C] \right\} \end{aligned}$$

where $\Upsilon_{w(0)}^{h-1\,\prime}(H)$ is calculated recursively.

Proof:

Using $S_{wxy(0)}^{h-1} = \rho_h^{(0)} r_h^{(0)}$, taking derivatives of both sides and plugging (4.25) and (4.26), we obtain;

$$r_h^{(0)'}(H) = \tfrac{1}{\rho_h}\left\{[I_p - r_h {r_h}^\top][v w_F^{(0)}(\gamma^*)u + C] - [I_p - \tfrac{r_h}{2\rho_h}\Sigma_{yx}]\Upsilon_{w(h-1)}{}'(H)\Sigma_{xy}\right\}$$
$$-\tfrac{1}{\rho_h}\left\{\Upsilon^{h-1}[w_F^{(0)}(\gamma^*)uv + C]\right\}.$$

□.

Step 3: Influence functions for $\tilde{r}_h^{(0)}$ and $\hat{\beta}_h^{(0)}$

$\tilde{r_h}^{(0)}$ is the scaled version of ${r_h}^{(0)}$ that is

$$\tilde{r_h}^{(0)} = \frac{{r_h}^{(0)}}{\sqrt{{r_h}^{(0)\top} S_{wxx} {r_h}^{(0)}}}$$

and using the quotient rule, the influence function of $\tilde{r_h}^{(0)}$ is

$$\frac{[\Psi(h)I_p - r_h {r_h}^\top \Sigma_{xx}] r_h^{(0)'}(H)}{\Psi(h)^{3/2}} - \frac{r_h {r_h}^\top S_{wxx}{}'(H) r_h}{2\Psi(h)^{3/2}}$$

where $\Psi(h) = {r_h}^\top \Sigma_{xx} r_h$ and $S_{wxx}{}'(H) = [w_F^{(0)}(\gamma^*)uu^\top - \Sigma_{xx} + C_x]$ with

$$C_x = E_F\left(\tfrac{\partial w^*}{\partial {d_\gamma}^B(F)} {d_\gamma}^{B'}(H) x x^\top\right).$$

C_x exists under the same conditions that C exists. Similar to $r_1^{(0)}$, the influence function of $r_h^{(0)}$ is comparable to the influence function of the classical PLS weight function, r_h, in (4.17). It consists of two parts: weighted version of equation (4.17) and the part depending on C and C_x which are assumed to be finite valued vector and matrix, respectively. $r_h^{(0)'}(H)$ is bounded, so is influence function of $\tilde{r_h}^{(0)}$, as long as $r_j^{(0)'}(H)$ for $j = 1, 2, \ldots, h-1$

and ${S_{wxx}}'(H)$ are bounded. These conditions also imply the boundedness of the influence function for $\hat{\beta}_h^{(0)}$ given in the next lemma.

Lemma 4.7 *Influence function for the initial RoPLS1 slope estimator for h $(h \geq 1)$ component, represented by functional $\hat{\beta}_h^{(0)} = \widetilde{R_h}^{(0)} \widetilde{R_h}^{(0)\top} S_{wxy}$ with $\widetilde{R_h}^{(0)} = [\tilde{r_1}^{(0)}, \tilde{r_2}^{(0)}, \ldots, \tilde{r_h}^{(0)}]$, at $F \in \mathscr{F}$ is*

$$\hat{\beta}_h^{(0)'}(H) = \widetilde{R_h}^{(0)'}(H) \widetilde{R_h}^{\top} \Sigma_{xy} + \widetilde{R_h} \widetilde{R_h}^{(0)'}(H)^{\top} \Sigma_{xy} + \widetilde{R_h} \widetilde{R_h}^{\top} [w_F^{(0)}(\gamma^*) uv - \Sigma_{xy} + C] \quad (4.27)$$

where j^{th} column of $\widetilde{R_h}^{(0)'}(H)$ is $\tilde{r_j}^{(0)'}(H)$ for $1 \leq \ j \leq h$.

Part II: Influence Function for $\hat{\beta}_h^{(i)}$

Assume that the slope estimator for h component model at iteration $i-1$, $\hat{\beta}_h^{(i-1)}$, is given where $i \geq 1$. In this part, influence function for $\hat{\beta}_h^{(i)}$, is derived as in Part I.

Step 1: Influence functions for $S_{wyx(i)} S_{wxy(i)}$ and $r_1^{(i)}$

A functional, $S_{w(i)}$, can be defined as $S_{w(i)}(G) = E_G(w_G^{(i)}(\gamma)\gamma\gamma^{\top})$ for a random vector $\gamma{=}(\mathbf{x}^{\top}, y)^{\top}$ and $G \in \mathscr{F}$, where $\mathscr{F}$ is the same class of d-variate probability distributions defined in Section 4.2. The weight functional, $w_G^{(i)}(\gamma)$, for $i \geq 1$ is

$$w_G^{(i)}(\gamma) = \Big(1 - d^B(G)\Big)\, w^* \left(\frac{{\epsilon_h}^{(i)}(G)}{mad({\epsilon_h}^{(i)}(G))} \right) = \Big(1 - d^B(G)\Big)\, w^* \left({\varepsilon_h}^{(i)}(G) \right) \quad (4.28)$$

where ${\epsilon_h}^{(i)}(G) = y - X\hat{\beta}_h^{(i-1)}(G)$ and $d^B(G)$ is the normalized version of ${d_x}^B(G) = \sqrt{x^{\top}\left([I_p : \mathbf{0}] S^B(G) [I_p : \mathbf{0}]^{\top}\right)^{-1} x}$ that lies between 0 and 1.

$w^*({\varepsilon_h}^{(i)}(G))$ is decreasing function of the $|{\varepsilon_h}^{(i)}(G)|$ and lie between 0 and 1 which can be seen from Figure 4.2(b) where ${m_{\varepsilon_h}}^{(i)}(G) = max\Big\{1, median(|{\varepsilon_h}^{(i)}(G)|)\Big\}$, under the assumption that the marginal distribution of ${\epsilon_h}^{(i)}(G) = y - X\hat{\beta}_h^{(i-1)}(G)$ is symmetric about 0. Similarly, $0 \leq 1 - d^B(G) \leq 1$ decreases when distances in X space increase. Therefore, the weight function, $w_G^{(i)}$, decreases with extreme residuals and/or large distances. Furthermore, $0 \leq w_G^{(0)}(\gamma) \leq 1$ for any $G \in \mathscr{F}$. Existence and non-negative definiteness of $S_{w(i)}$ can be shown as in Part I. Other assumption made in this section, additional to the *(1)*, *(2)*, *(3)* and *(4)* given in Part I, is that

5. $P_F\{\gamma : |\varepsilon_h^{(i)}| = m_{\varepsilon_h}^{(i)}(F)\} = 0.$

This assumption is required for differentiability of $w^*({\varepsilon_h}^{(i)}(F))$ over the support of the random vector γ.

Similar to previous subsection, the influence function for $S^0_{wyx(i)} S^0_{wxy(i)} = S_{wyx(i)} S_{wxy(i)}$ is needed to determine the influence function of the $r_1^{(i)}$ for $i \geq 1$.

Lemma 4.8 *Influence function of* $S_{wyx(i)} S_{wxy(i)}$ *at* $F \in \mathscr{F}$ *is*

$$IF(\gamma^*; S_{wyx(i)} S_{wxy(i)}, F) = 2w_F^{(i)}(\gamma^*) u^\top v \Sigma_{xy} - 2\Sigma_{yx}\Sigma_{xy} + 2\Sigma_{yx} A_d^i + 2\Sigma_{yx} A_s^i + 2\Sigma_{yx} A_\beta^i \hat{\beta}_h^{(i-1)'}(H) \tag{4.29}$$

with $A_d^i = E_F\left(-d^{B'}(H) w^*({\varepsilon_h}^{(i)}) xy\right)$, $A_s^i = E_F\left(-(1 - d^B) \frac{\partial w^*}{\partial {\varepsilon_h}^{(i)}} \frac{\epsilon_h^{(i)} s'(H)}{s^2} xy\right)$, *and* $A_\beta^i = E_F\left(-(1 - d^B) \frac{\partial w^*}{\partial {\varepsilon_h}^{(i)}} \frac{y}{s} x x^\top\right)$ *where* $s = mad(\epsilon_h^{(i)}(F))$.

Proof:

The influence function for $S_{wxy(i)}$ is:

$$S'_{wxy(i)}(H) = E_F\left(w^{(i)'}(H) xy\right) + E_H\left(w_F^{(i)}(\gamma) xy\right) = E_F\left(w^{(i)'}(H) xy\right) + \left\{w_F^{(i)}(\gamma^*) uv - \Sigma_{xy}\right\}$$

with

$$w^{(i)'}(H) = \frac{\partial}{\partial t} w_{F+tH}^{(i)}(\gamma) \mid_{t=0} = \frac{\partial}{\partial t}\left\{(1 - d^B(F + tH)) w^*\left({\varepsilon_h}^{(i)}(F + tH)\right)\right\} \mid_{t=0}$$

$$= -d^{B'}(H)w^*({\varepsilon_h}^{(i)}) + (1-d^B)\frac{\partial w^*}{\partial {\varepsilon_h}^{(i)}}{\varepsilon_h}^{(i)'}(H)$$

where ${\varepsilon_h}^{(i)}(F) = \frac{\epsilon_h^{(i)}(F)}{s(F)} = \varepsilon_h^{(i)}$ with scaling factor $s(F) = mad(\epsilon_h^{(i)}(F)) = s$, $\epsilon_h^{(i)}(F) = y - x^\top\beta_h^{(i-1)}(F) = \epsilon_h^{(i-1)}$, $d^B(F) = d^B$, and ${\varepsilon_h}^{(i)'}(H)$ is

$${\varepsilon_h}^{(i)'}(H) = \frac{\epsilon_h^{(i)'}(H)s - \epsilon_h^{(i)}s'(H)}{s^2} = \frac{-x^\top\hat{\beta}_h^{(i-1)'}s - \epsilon_h^{(i)}s'(H)}{s^2}.$$

Therefore $S'_{wxy(i)}(H)$ can be written as

$$S'_{wxy(i)}(H) = A_d^i + A_s^i + A_\beta^i\hat{\beta}_h^{(i-1)'} + \left\{w_F^{(i)}(\gamma^*)uv - \Sigma_{xy}\right\}$$

with $A_d^i = E_F\left(-d^{B'}(H)w^*({\varepsilon_h}^{(i)})xy\right)$, $A_s^i = E_F\left(-(1-d^B)\frac{\partial w^*}{\partial {\varepsilon_h}^{(i)}}\frac{\epsilon_h^{(i)}s'(H)}{s^2}xy\right)$, and $A_\beta^i = E_F\left(-(1-d^B)\frac{\partial w^*}{\partial {\varepsilon_h}^{(i)}}\frac{y}{s}xx^\top\right)$.

A_d^i exists since $d^{B'}(H)w^*({\varepsilon_h}^{(i)})$ is bounded. From Figure 4.2(b), it can be seen that $\frac{\partial w^*}{\partial {\varepsilon_h}^{(i)}}$ exists everywhere except $\pm m_{\varepsilon_h}^{(i)}(F)$. Hence, A_β^i exists because of the assumption (*5*). A_s^i exists if A_β^i exists since $s'(H)$ is known to be bounded which is influence function of robust scatter measure *median absolute deviation*. Using multiplication rule and the fact that $S'_{wyx(i)}(H) = S'_{wxy(i)}(H)^\top$, the following influence function is obtained for $S_{wyx(i)}S_{wxy(i)}$:

$$2w_F^{(i)}(\gamma^*)u^\top v\Sigma_{xy} - 2\Sigma_{yx}\Sigma_{xy} + 2\Sigma_{yx}A_d^i + 2\Sigma_{yx}A_s^i + 2\Sigma_{yx}A_\beta^i\hat{\beta}_h^{(i-1)'(H)}$$

□.

Since the influence function of $S_{wyx(i)}S_{wxy(i)}$ is equal to $2\rho_1^{(i)'}(H)$,

$$\rho_1^{(i)'}(H) = w_F^{(i)}(\gamma^*)vr_1^\top u - \rho_1 + r_1^\top A_d^i + r_1^\top A_s^i + r_1^\top A_\beta^i\hat{\beta}_h^{(i-1)'(H)}.$$

The next lemma gives the influence function for $r_1^{(i)}$.

Lemma 4.9 *Influence function of* $r_1^{(i)}$ *at* F *is*

$$IF(\gamma^*; r_1^{(i)}, F) = \frac{1}{\rho_1}\left[I_p - r_1 r_1^\top\right]\left[w_F^{(i)}(\gamma^*)uv + A_d^i + A_s^i + A_\beta^i \hat{\beta}_h^{(i-1)'(H)}\right] \tag{4.30}$$

The proof can be given in a similar way to that of the proof of Lemma 4.5.

Step 2: Influence function for $r_h^{(i)}$

The next lemma gives the influence function for $r_h^{(i)}$ and the proof is similar to the proof of Lemma 4.6.

Lemma 4.10 *The influence function of* $r_h^{(i)}$ *at* $F \in \mathscr{F}$ *is*

$$\begin{aligned} r_h^{(i)'}(H) = \tfrac{1}{\rho_h}&\left\{[I_p - r_h {r_h}^\top][vw_F^{(i)}(\gamma^*)u + A_d^i + A_s^i + A_\beta^i \hat{\beta}_h^{(i-1)'}(H)]\right\} \\ &-\tfrac{1}{\rho_h}\left\{[I_p - \tfrac{r_h}{2\rho_h}\Sigma_{yx}]\Upsilon_{w(i)}^{h-1'}(H)\Sigma_{xy}\right\} \\ &-\tfrac{1}{\rho_h}\left\{\Upsilon^{h-1}[w_F^{(i)}(\gamma^*)uv + A_d^i + A_s^i + A_\beta^i \hat{\beta}_h^{(i-1)'}(H)]\right\} \end{aligned}$$

where $\Upsilon_{w(i)}^{h-1'}(H)$ *is calculated recursively.*

Step 3: Influence functions for $\tilde{r}_h^{(i)}$ and $\hat{\beta}_h^{(i)}$

$\tilde{r_h}^{(i)}$ is the scaled version of ${r_h}^{(i)}$ that is

$$\tilde{r_h}^{(i)} = \frac{{r_h}^{(i)}}{\sqrt{{{r_h}^{(i)}}^\top S_{wxx(i)} {r_h}^{(i)}}}$$

and using the quotient rule, the influence function of $\tilde{r_h}^{(i)}$ is

$$\frac{[\Psi(h)I_p - r_h {r_h}^\top \Sigma_{xx}] r_h^{(i)'}(H)}{\Psi(h)^{3/2}} - \frac{r_h {r_h}^\top {S_{wxx(i)}}'(H) r_h}{2\Psi(h)^{3/2}}$$

where $\Psi(h) = {r_h}^\top \Sigma_{xx} r_h$ and ${S_{wxx}}'(H) = [w_F^{(i)}(\gamma^*)uu^\top - \Sigma_{xx} + A^i_{xd} + A^i_{xs} + A^i_{x\beta}\hat{\beta}_h^{(i-1)'}(H)]$ with $A^i_{xd} = E_F\left(-d^{B'}(H)w^*({\varepsilon_h}^{(i)})xx^\top\right)$, $A^i_{xs} = E_F\left(-(1-d^B)\frac{\partial w^*}{\partial {\varepsilon_h}^{(i)}}\frac{\epsilon_h^{(i)}s'(H)}{s^2}xx^\top\right)$, and $A^i_{x\beta} = E_F\left(-(1-d^B)\frac{\partial w^*}{\partial {\varepsilon_h}^{(i)}}\frac{yx^\top\hat{\beta}_h^{(i')}(H)}{s}xx^\top\right)$.

A^i_{xd}, A^i_{xs}, and $A^i_{x\beta}$ exist under the same conditions that A^i_d, A^i_s, and A^i_β exist. Finally, the influence function for $\hat{\beta}_h^{(i)}$ is given in the next lemma.

Lemma 4.11 *Influence function for the RoPLS1 slope estimator at the i^{th} iteration for h $(h \geq 1)$ component, represented by functional $\hat{\beta}_h^{(i)} = \widetilde{R_h}^{(i)}\widetilde{R_h}^{(i)\top}S_{wxy(i)}$ with $\widetilde{R_h}^{(i)} = [\tilde{r_1}^{(i)}, \tilde{r_2}^{(i)}, \ldots, \tilde{r_h}^{(i)}]$, at $F \in \mathscr{F}$ is*

$$\hat{\beta}_h^{(i)'}(H) = \widetilde{R_h}^{(i)'}(H)\widetilde{R_h}^\top\Sigma_{xy} + \widetilde{R_h}\widetilde{R_h}^{(i)'}(H)^\top\Sigma_{xy} + \widetilde{R_h}\widetilde{R_h}^\top[w_F^{(i)}(\gamma^*)uv - \Sigma_{xy} + A^i_d + A^i_s + A^i_\beta\hat{\beta}_h^{(i-1)'}(H)] \tag{4.31}$$

where j^{th} column of $\widetilde{R_h}^{(i)'}(H)$ is $\tilde{r_j}^{(i)'}(H)$ for $1 \leq\ j \leq h$.

The boundedness of $\hat{\beta}_h^{(i)'}(H)$ for $i \geq 1$ can be proven by induction. For $i = 1$, since $w_F^{(1)}$ gets smaller for the observations lying far from the data (large d^B) and/or the ones which are not fitted well by the model (large residuals), influence function given (4.30) is bounded as long as $\beta_h^{(0)'}(H)$ is bounded. Therefore, $r_1^{(1)'}(H)$ has bounded influence function. Similarly, $S_{wxx(1)}$ has a bounded influence function. These facts imply the boundedness of $\hat{\beta}_h^{(1)'}(H)$.

If we assume that $\hat{\beta}_h^{(i)'}(H)$ is bounded, using the similar argument it can be proven that $\hat{\beta}_h^{(i+1)'}(H)$ is bounded. So, it can be concluded that under conditions described in Part I & II, influence function for the h component RoPLS1 slope estimator, determined implicitly, exists and it has infinitesimal robustness.

4.3.2 Empirical Influence Function for High Dimension

Influence function obtained in Section 4.3 is shown to be bounded, which demonstrates the robustness of RoPLS1 for low dimensional data. For high dimensional case with n observations, the following empirical influence function, defined as

$$\frac{\hat{\beta}(\widetilde{\Gamma}) - \hat{\beta}(\Gamma)}{1/n}, \tag{4.32}$$

is used where $\widetilde{\Gamma}$ is the contaminated data set obtained from varying one observation of Γ, $\hat{\beta}(\widetilde{\Gamma})$ and $\hat{\beta}(\Gamma)$ are the estimated slope vectors for the contaminated and the clean data, respectively. Data are generated by the same simulation setting described in Chapter 3 where the error terms are generated from standard normal distributions, that is, $\varepsilon \sim N(0, 1)$. The contamination added to a randomly chosen observation by varying the explanatory variable and response variable between -50 to 50. For each contamination level, the *norm* of the empirical influence function given in (4.32) is calculated for {n,p,k}={20,200,3} and {25,125,2} . Then three dimensional plots in Figure 4.4 are constructed. Figure 4.4 (a) and (c) clearly illustrate non-robustness of the SIMPLS estimator. However, empirical influence curves for RoPLS1 estimator are clearly bounded which can be seen in Figure 4.4 (b) and (d). The maximal norms of the empirical influence functions for the SIMPLS, RoPLS1, PRM, and RSIMPLS estimators of β are summarized in Table 4.1. RoPLS1 yielded the smallest upper bound for the norm of the empirical influence function.

{n,p,k}\ Method	SIMPLS	RoPLS1	PRM	RSIMPLS
{20,200,3}	218.13	0.16	1.48	0.62
{25,125,2}	88.74	0.17	0.93	1.76

Table 4.1: The maximal norms for SIMPLS, RoPLS1, PRM, and RSIMPLS estimators.

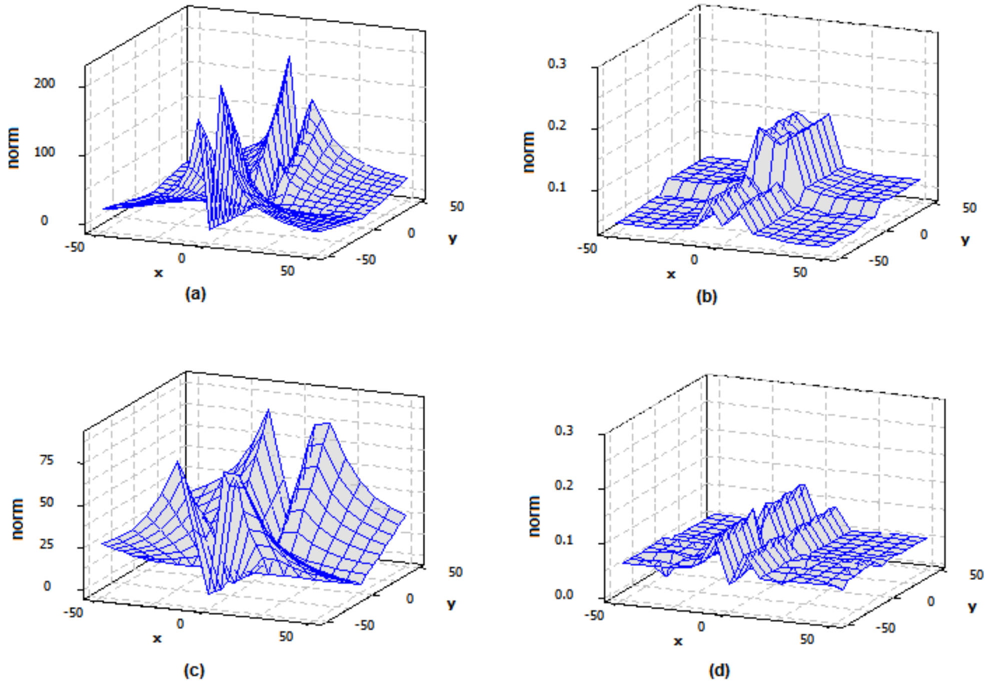


Figure 4.4: Norms of the empirical influence functions for the (a)SIMPLS, n=20, p=200, k=3 (b) RoPLS1 n=20, p=200, k=3 (c) SIMPLS, n=25, p=125, k=2 (d) RoPLS1, n=25, p=125, k=2.

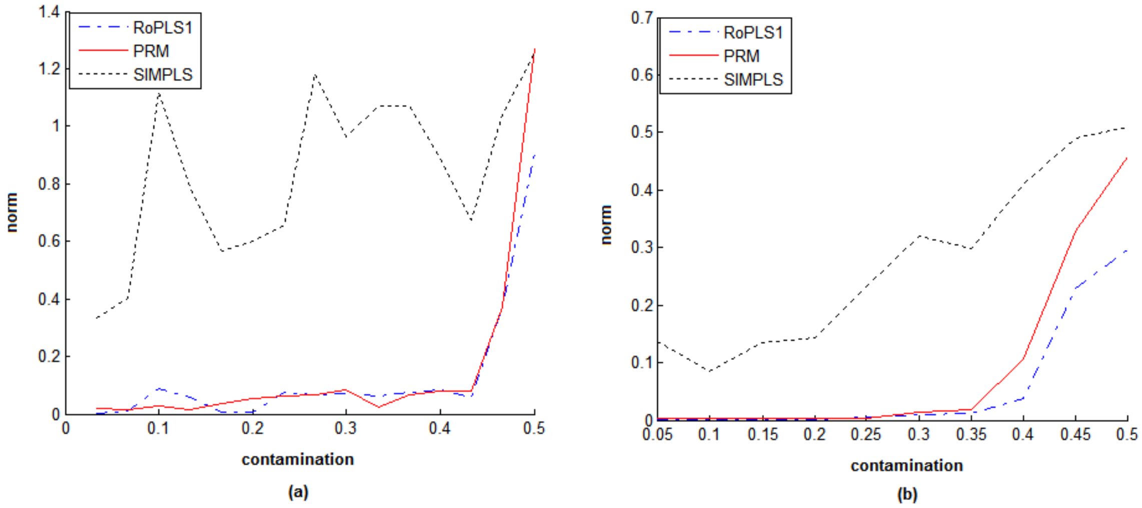


Figure 4.5: The finite-sample breakdown values of the SIMPLS, RoPLS1 and PRM estimators for (a){n,p}={30,6} (b){n,p}={20,200}.

4.3.3 Finite-Sample Breakdown Properties of RoPLS1 Estimator

The resistance of a robust statistical method to groups of outliers is another important issue which is measured by breakdown point. In this section, finite-sample breakdown value, given in (4.5), is investigated for RoPLS1 estimator. After X and y are generated as in Section 4.3.2, various amounts of contamination are added to generated data by replacing first i observations ($i = 1, 2, \ldots, n/2$) of the response variable with 50. For each amount of contamination, the norm of the difference between slope estimates for the contaminated and the clean data is calculated and a plot of norm versus contamination percentage is constructed. The sample size and the number of variables are taken as {n,p}={30,6} and {20,200} for $k = 2$ component PLS model. It is clear from Figure 4.5 that the SIMPLS estimator of β is not robust, whereas RoPLS1 estimator copes with up to 43% of irregular observations for low dimensional case and about 40% for high dimensional case. PRM yields comparable results with RoPLS1.

Chapter 5

RoCPLS: Robust Partial Classification

5.1 Introduction

The problem of classifying entities into one of several groups has been one of the main goals of many scientific investigations. For instance, predicting whether someone will have a heart attack on the basis of demographic and clinical measurements or identifying a tumor as one of the many different possibilities on the basis of DNA expression values are potentially life-saving and hence are indispensable to physicians. Numerous other interesting applications of classification can be found in a broad range of scientific areas such as chemistry, economics, marketing research, bio-informatics, image analysis, pattern recognition and data mining.

Classification is a multivariate method of distinguishing among classes of objects by developing a decision rule to assign a new object with unknown class membership to the most likely group. In this study, only two-class problems are considered. (x_i', y_i) denotes the observed data set, with $x_i = [x_{i1}, x_{i2}, \ldots, x_{ip}]' \in \mathbb{R}^p$ consisting of p characteristics that are sampled from two populations and y_i is the class membership for the observation i where $i = 1, 2, \ldots, n$. $I_g = \{i; y_i = g\}$ denotes the set of indices for the n_g observations in the class g where $g = 1, 2$ and $n = n_1 + n_2$.

It is important that classification is done in a manner that the proportion of misclassified observations (misclassification error rate) is minimum. In general, performances of classification methods can be evaluated based on their misclassification error rates which can be obtained by using different approaches. Optimum error rate (OER) and actual error rate (AER) are two quantities that can be used for determining misclassification probabilities. However, they cannot, in general, be calculated, because they depend on the unknown density functions of the populations. Another method, which is employed in this study since it does not depend on the form of populations, is to split the data set randomly into two non-overlapping sets, called learning, $(X_{\mathbb{L}}, y_{\mathbb{L}})$, and test, $(X_{\mathbb{T}}, y_{\mathbb{T}})$ sets. The learning set, $X_{\mathbb{L}}$, allows to construct a decision rule, δ, that associates a new vector $x \in \mathbb{R}^p$ to one of the two classes, that is

$$\delta(x, X_{\mathbb{L}}, y_{\mathbb{L}}) : \mathbb{R}^p \to \{1, 2\}$$

where $y_{\mathbb{L}}$ is the vector containing the class labels of the observations in the learning data set, $X_{\mathbb{L}}$. Based on the determined classification rule, the fraction of the misclassified observations in the test set, $X_{\mathbb{T}}$, is computed. By repeating this process N times, estimated misclassification error rate is obtained as

$$\widehat{MER} = \frac{1}{N n_{\mathbb{T}}} \sum_{r=1}^{N} \sum_{j=1}^{n_{\mathbb{T}}} I_{\{-1,1\}}(y_j - \hat{y}_j) \tag{5.1}$$

where $n_{\mathbb{T}}$ is the number of observations in the test set, y_j is the known class label in $X_{\mathbb{L}}$, $\hat{y}_j$ is the estimated class label for the j^{th} observation in $X_{\mathbb{T}}$, and $I_{\{-1,1\}}(b)$ is the indicator function which takes the value of 1 if $b = -1, 1$ and 0 otherwise. Apparent error rate (APER) is another measure of the performance that can be used for any classification procedure. APER is the fraction of observations in the $X_{\mathbb{L}}$ that are misclassified by the classification rule, $\delta(., X_{\mathbb{L}}, y_{\mathbb{L}})$. Although, it is easy to calculate, it underestimates the error rate. Cross-validation is also a popular approach that consists to split data set into m non-overlapping subsets where $m - 1$ subsets form learning set to construct decision rule and remaining subset is used as test set. If $m = n$, the procedure is called *leave-one-out* cross validation. Throughout this study, leave-one-out classification is employed to estimate the value of meta parameter, k, i.e. optimal number of components.

Since the introduction of the Fisher's discriminant (FD) analysis in 1936 ([28]), several classification rules have been proposed and studied in the literature. FD analysis is based on the idea of finding the directions in multivariate space that yield the best discrimination between the groups of samples. This idea can be written as the optimization problem given by

$$\underset{a\in\mathbb{R}^p}{\operatorname{argmax}} \frac{a'Ba}{a'Wa} = \frac{a'\left\{\sum_{g=1}^{2} n_g(\overline{x}_g - \overline{x})(\overline{x}_g - \overline{x})'\right\} a}{a'\left\{\sum_{g=1}^{2} \sum_{i\in I_g}(x_i - \overline{x}_g)(x_i - \overline{x}_g)'\right\} a} \tag{5.2}$$

where B is the sample *between-group* matrix, W is the sample *within-group* matrix, $\overline{x}_g$ sample mean vector for g^{th} class with $g = 1, 2$ and $\overline{x}$ is the overall sample mean vector. In general, if α_1 is the largest eigenvalue and e_1 is the corresponding eigenvector of $W^{-1}B$, then $a = e_1 = S_p^{-1}(\overline{x}_1 - \overline{x}_2)$ is the solution of the optimization problem in (5.2) where $S_p = W/(n-2)$ is the sample pooled covariance matrix. Therefore FD rule can be given as

$$\delta_{FD}(x, X, y) = \begin{cases} 1; & e_1'x \geq 0.5(\overline{x}_1 - \overline{x}_2)'S_p^{-1}(\overline{x}_1 + \overline{x}_2) \\ 2; & otherwise. \end{cases}$$

FD rule is developed under the assumption that the two populations have a common covariance matrix and it does not explicitly assume any form for the underlying distributions.

Bayes classification is another approach that needs prior probabilities, π_g, and probabilistic structure estimates for each class. The Bayesian discriminant rule assigns an observation $x \in \mathbb{R}^p$ to the population for which posterior probability, $P(y|x)$, is maximal. Under the assumption that each class comes from multivariate normal distribution with equal covariance matrix, the allocation rule is

$$\delta_{LD}(x, X, y) = \underset{g}{\operatorname{argmax}}\, LD_g(x; X, y); \tag{5.3}$$

where

$$LD_g(x; X, y) = \overline{x}_g' S_p^{-1} x - 0.5 \overline{x}_g' S_p^{-1} \overline{x}_g + ln(\pi_g); \tag{5.4}$$

and this is called *linear discriminant* (LD) rule. Provided that the two classes come from the two normal distributions with the same covariance matrix and equal prior probabilities, δ_{FD} is equivalent to linear discriminant rule, δ_{LD}. When the assumption of equality of covariance matrices is not satisfied, an individual covariance matrix for each group can be used in (5.4) and this leads to the so-called quadratic discriminant (QD) analysis as the discriminating boundaries are quadratic curves.

Over the last decade, many sophisticated classification methods, like support vector machine, neural networks, classification and regression trees (CART), have been proposed. In spite of these refined methods, δ_{LD}, that yields optimal discrimination between two classes, is still often used and very popular because of the simplicity, unnecessity of strict assumptions, interpretation easiness and its good performance in many applications. Of course from the point of view of *optimality*, LD analysis should be used for classification when it can be used. However, it becomes a serious challenge to use LD analysis in the settings where the data matrix X is multicollinear or $p >> n$. Because, the sample covariance matrix estimate is near singular if high collinearity exists and high dimensionality makes direct matrix operation difficult. Many solutions have been proposed to deal with these problems such as variable selection, penalized estimation, and dimension reduction.

Variable selection is a very popular method due to its simplicity and interpretability. The most commonly used variable selection methods are based on a score (such as t-statistic, Wilcoxon's rank-sum statistic, false discovery rate) which measures discriminating power of each variable individually and the variables with the best scores are selected (see [20], [24], [25]). These methods are called univariate ranking methods. The major drawback is the selection of variables according to an individual relevance score that ignores the correlations and interactions among variables. Therefore, more complex criteria than the individual scores have been proposed (optimal subset selection methods), which are generally computationally expensive and suffer from over fitting problem ([7], [11], [52]).

Penalization (regularization) methods can be also employed to stabilize the pertinent covariance matrices so that the classical discrimination paradigms might be implemented (see [32]). These methods reduce the variance associated with the sample based estimate at the expense of potentially increased bias.

Dimension reduction (feature extraction) is another alternative to deal with dimensionality problem. It allows the visualization of data in a low dimension, takes into account the correlation structure of the data and the most importantly, utilizes the information on all variables. This topic, particularly PLS as a dimension reduction tool, is examined in Section 5.2. Although, PLS solves dimensionality problem by constructing orthogonal components described in Section 5.2, it fails to deal with data containing outliers. Therefore, in Section 5.3, a new robust method, RoCPLS, is proposed. To our knowledge, there has been no study on the robustness of PLS based classification methods. Performances of the existing PLS based classification methods and RoCPLS are compared using benchmark data sets in Section 5.4.

5.2 Classification Methods Based on Dimension Reduction

A typical DNA microarray data set in tumor tissue classification studies consists of expression measurements on thousands of genes over a relatively small number of tissue samples. Similarly, in food research, classical classification methodologies can not be used for the prediction of presence/absence of a preservative in a particular food product based on spectral data in which number of variables is very large and the correlation among them is substantial.

One approach to classification problems, dealing with high dimensional and/or collinear data sets, is to employ a dimension reduction method and then perform a standard classification method in the reduced space. In this section, we study dimension reduction for classification based on PLS and PCA followed by LD implemented in the reduced subspace. Another classification method such as logistic regression can also be employed instead of LD, however logistic regression does not perform well when the classes are completely or quasi-completely separated which is quite common configuration in microarray data.

Although PLS was originally designed for problems with quantitative response, it has started to be used frequently as a dimension reduction tool for classification problems where response variable is qualitative. There are mainly two approaches when PLS is employed as a dimension reduction method for classification purpose.

One approach is to utilize NIPALS algorithm to determine components. However, since NIPALS algorithm consists of regression steps (see Chapter 2), it seems to be unappealing to use NIPALS algorithm designed to handle continuous response models that do not suffer from heteroscedasticity. So, Marx ([60]) proposed an extension of NIPALS algorithm to handle qualitative response models. He basically incorporated the original NIPALS algorithm into the framework of generalized linear models by employing iteratively reweighted least squares (IRLS). The main drawback of the method is the convergence problem. Therefore, Ding and Gentleman ([21]) modified the Marx's method by applying Firth's procedure ([27]) to resolve complete or quasi-complete separation problem resulted in convergence problem. Recent method by Fort and Lacroix, RPLS, ([30]), combines the NIPALS algorithm and Ridge penalized logistic regression. They also provided an extensive simulation study to compare existing NIPALS based classification methods and concluded that misclassification error rates for IRPLS and Ding and Gentleman's method are lower and less variable.

The other most commonly used approach is to determine PLS components for classification problem is applying original SIMPLS algorithm, described in Chapter 2. Barker and Rayens ([4]), Nguyen and Rocke ([65]) and Boulesteix ([8]) proposed the use of SIMPLS for dimension reduction based on SIMPLS as a preliminary step to classification problems. In this chapter, SIMPLS based classification is considered because not only SIMPLS has computational advantages over NIPALS algorithm (see Chapter 2), but also optimal directions obtained by SIMPLS are related to the Fisher's optimal directions, so there is a relationship between classification based on SIMPLS and Fisher's discrimination. The following lemma gives this relationship.

Lemma 5.1 *(Boluesteix, [8]) If the common covariance matrix, Σ, is assumed to be of the form $\Sigma = \sigma^2 I_p$ for a non-zero constant σ, then $a = e_1$ and the first PLS-weight vector, r_1 are collinear.*

Proof:

Let X and y be centered. $\hat{r}_1$ is the direction that maximizes the square of the covariance between projected explanatory variable and response variable, i.e.

$$\hat{r}_1 = \text{argmax}_a \, cov(Xa, Y) = \frac{X'y}{\sqrt{y'XX'y}}.$$

The centered y is given by

$$y_i = \begin{cases} -n_2/n; & i \in I_1 \\ n_1/n; & i \in I_2. \end{cases}$$

Therefore, the j^{th} row of $p \times 1$ vector r_1 is

$$\frac{-n_2}{n} \sum_{i \in I_1} x_{ij} + \frac{n_1}{n} \sum_{i \in I_2} x_{ij} = \frac{n_1 n_2}{n} (\overline{x}_{2j} - \overline{x}_{1j})$$

and

$$\hat{r}_1 = \frac{(\overline{x}_2 - \overline{x}_1)}{\| \overline{x}_1 - \overline{x}_2 \|}.$$

Therefore r_1 is proportional to the normalized form of $\mu_1 - \mu_2$ which is the dominant eigenvector of between-groups matrix, B. Since $e_1 = \Sigma^{-1}(\mu_1 - \mu_2)$ and $\Sigma = \sigma^2 I_p$, r_1 and e_1 are collinear □.

Lemma 5.1 implies that SIMPLS depends on the between-groups matrix. It is also obvious that the within-group matrix information is not utilized to construct SIMPLS components, that is, since only B not W is involved, classification based on SIMPLS only depends on between-groups matrix. So, LD outperforms in the situations that it can be implemented. However, in the existence of multicollinearity, optimality advantage of LD over SIMPLS based classification would reverse direction.

PCA reduces the dimension of the data set by retaining as much as possible the variation present in the data. So, PCA is only capable of identifying total variability, i.e., $B + W$, and not capable of distinguishing between-groups and within-groups variability which is the main goal of Fisher's discrimination ([4], [8]). Especially, if within-groups variability, W, dominates the total variability, PCA will no longer perform well as a classification tool. Since, SIMPLS depends on the between-groups matrix, when the discrimination is the major goal after dimension reduction, SIMPLS is to be preferred to PCA. The following example also indicates that PLS outperforms the PCA as within-groups variability increases.

Example 5.1

This example is the modified version of the example given by Barker and Rayens, [4]. 50 observations from the two multivariate normal distributions with the means $\mu_1 = (-2, 0, 0)'$ and $\mu_2 = (2, 0, 0)'$ and common covariance matrix

$$\Sigma = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & r\sigma \\ 0 & r\sigma & \sigma^2 \end{pmatrix}$$

are generated N=100 times, where $\sigma^2 = 1, 2, \ldots, 6$ is the variance of the third variable and $r = 0.9$ is the correlation between second and third variable. Misclassification error rates are calculated using leave-one-out cross validation and these rates are averaged over 100 randomly generated data. It can be seen from Figure 5.1 that as σ^2, variance of the third variable, varies from 1 to 6, the misclassification rate based on PCA based classification increases since the PCA loses sight of the discrimination information when within-group matrix dominates the total variability.

PCA and SIMPLS are both linear dimension reduction methods, but SIMPLS uses class information, y, to construct components (supervised), while PCA does not use the class information (unsupervised). There are several other dimension reduction methods that can be applied in the context of classification. For instance, sliced inverse regression

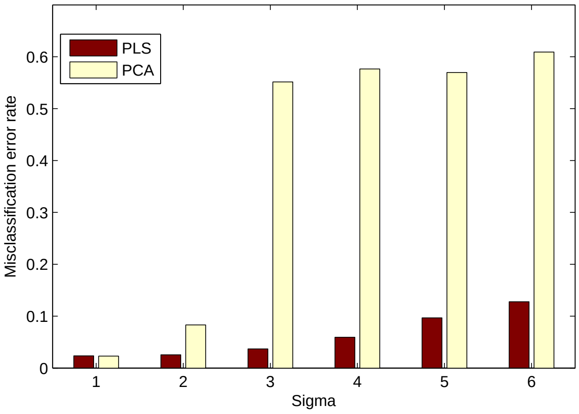


Figure 5.1: Misclassification error rates for PLS and PCA for k=1.

(SIR) is one of the sufficient dimension reduction methods ([51]) which represent a family of dimension reduction procedures. A simulation study by Dai et al. ([14]) demonstrates that SIMPLS and SIR are both effective in dimension reduction for classification and also more effective than PCA which is not surprising since both SIMPLS and SIR are supervised methods. Considering both accuracy and computational efficiency, it is concluded in this study that SIMPLS provides the best performance among PCA and SIR.

5.3 Description of the Proposed Algorithm: RoCPLS

In Chapter 3, it has been shown that RoPLS is successful in regression framework where data contain outliers. Partial least squares is also frequently used as a classification method as described in Section 5.2. In the presence of outliers, dimension reduction via PLS would yield unreliable results since PLS is known to be sensitive to outliers. Although several robust PLS methods have been proposed when the response variable is quantitative, to our knowledge, there has been no study on the robustness of PLS when the response variable is qualitative. In this section, the sensitivity of PLS based classification methods to outliers is demonstrated and an extension of the robust method introduced in Chapter 3 is given.

The proposed algorithm, RoCPLS, is the robustified version of SIMPLS based classification. The main differences between RoPLS and RoCPLS are that weights for the response variable, y, are immaterial and weights for data matrix, X, are computed for each class separately. The detailed algorithm is given below:

Algorithm: *RoCPLS*

Input: $n \times p$ data matrix, X, $n \times 1$ vector of response variable, y, a new observation, $x \in \mathbb{R}^p$

Output: Score matrix, T, and $p \times k$ PLS weight matrix, R and class label for the new observation, $\hat{y}$

***Step 1*:** Let $X_g = \{x_{ij}; i \in I_g, j = 1, 2, \ldots, p\}$ for $g = 1, 2$. Apply PCOUT algorithm, described in Section 3.2.2, to X_1 and X_2 to obtain weight vectors w_1 and w_2, respectively. Take $p^* = n_g - 1$ for high dimensional data and $p^* = rank(X_g)$ for low dimensional data where $g = 1, 2$. Within each group, any observation with final weight less than 0.25 is assigned as an outlier. So, let X_1^* and X_2^* be the clean matrices of observations with corresponding weights greater than 0.25, and merged matrix

$$X^* = \{x_{ij}^*\} = \begin{pmatrix} X_1^* \\ X_2^* \end{pmatrix},$$

with the vector of class labels y^* and n_1^c and n_2^c are the number of observations in X_1^* and X_2^*, respectively with $n^c = n_1^c + n_2^c$.

***Step 2*:** Let $X^0 = \{x_{ij}^0\} = X^*$; $X_{1j}^0 = \{x_{ij}^0; 1 \le i \le n_1^c\}$ and $X_{2j}^0 = \{x_{ij}^0; n_1^c + 1 \le i \le n^c\}$ for $1 \le j \le p$. Repeat steps $2.1 - 2.6$ for $h = 1, 2, \ldots, k$:

Step 2.1: Compute PLS weight vector r_h with j^{th} row equal to:

$$r_h(j) = \frac{m_j^1 - m_j^2}{\parallel m_j^1 - m_j^2 \parallel}$$

where

$$m_j^g = \frac{\sum_{i \in I_g^j} x_{ij}^{h-1}}{n_g^j}$$

with

$$I_1^j = \{i; 1 \leq i \leq n_1^c \text{ and } q_{25}(X_{1j}^{h-1}) - 1.5IQR(X_{1j}^0) \leq x_{ij}^{h-1} \leq q_{75}(X_{1j}^0) + 1.5IQR(X_{1j}^0)\}$$
$$I_2^j = \{i; n_1^c + 1 \leq i \leq n^c \text{ and } q_{25}(X_{2j}^{h-1}) - 1.5IQR(X_{2j}^0) \leq x_{ij}^{h-1} \leq q_{75}(X_{2j}^0) + 1.5IQR(X_{2j}^0)\}$$

where q_{25} and q_{75} are the 25^{th} and 75^{th} sample quantiles; IQR is the interquartile range and n_g^j is the size of I_g^j for $g = 1, 2$ and $j = 1, 2, \ldots, p$.

Step 2.2: Compute h^{th} score, $t_h = X^* r_h$, and normalize $t_h =: t_h / \|t_h\|$,

Step 2.3: Update h^{th} PLS weight, $r_h =: r_h / \sqrt{r_h' X^{*'} X^* r_h}$,

Step 2.4: Compute h^{th} x-loading by regressing X^* on t_h: $p_h = X^{*'} t_h$,

Step 2.5: Store vectors $\mathbf{r}_h$, $\mathbf{t}_h$, and $\mathbf{p}_h$ into matrices $R_h = [\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_h]$, $T_h = [\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_h]$, and $P_h = [\mathbf{p}_1, \mathbf{p}_1, \ldots, \mathbf{p}_h]$, respectively.

Step 2.6: $h =: h + 1$ and $X^{h-1} = X^*(I_p - V_{h-1} V_{h-1}') = \{x_{ij}^{h-1}\}$ where columns of V_{h-1} form an orthonormal basis for P_{h-1}.

***Step 3*:** $\hat{y} = \delta_{LD}(x' R_h, T_h, y^*)$

The constructed component matrix, T_h, is not only utilized to determine classification rule, but also used to plot the first two or three components which helps to display relationships, possible groupings and potential outliers in the data. After projecting the original data matrix, X, on the robustly calculated directions, R, *orthogonal-score distance* plot, described in Section 3.3, can be constructed to distinguish regular, good PLS-leverage, orthogonal and bad PLS-leverage points. It can be also used to visualize outlying observations within each class.

Numerical examples with diagnostic plots in Section 5.4 demonstrated the robustness and efficiency of the proposed method.

5.4 Numerical Examples

In this section, two benchmark data sets are utilized to compare performances of the existing dimension reduction based classification methods and proposed method, RoCPLS.

5.4.1 Low Dimension: Wine Recognition Data

The wine recognition data ([29]) are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents (i.e., the level of alcohol, the level of magnesium, the color intensity) found in each of the types of wines. In this study, only two cultivars with sample sizes 59 and 71 are considered.

To determine the optimal number of components, SIMPLS based cross-validation error rate is calculated for $h = 1, 2, \ldots, 10$ components and the scree plot in Figure 5.2 is obtained. Based on the figure, the optimal number of components is determined as 7, i.e. $k = 7$ which yields the lowest error rate. The orthogonal-score plots for SIMPLS and RoCPLS based on $k = 7$ component model can be seen in Figure 5.3. It is obvious that none of the PLS-bad leverage points can be identified when SIMPLS is employed, while

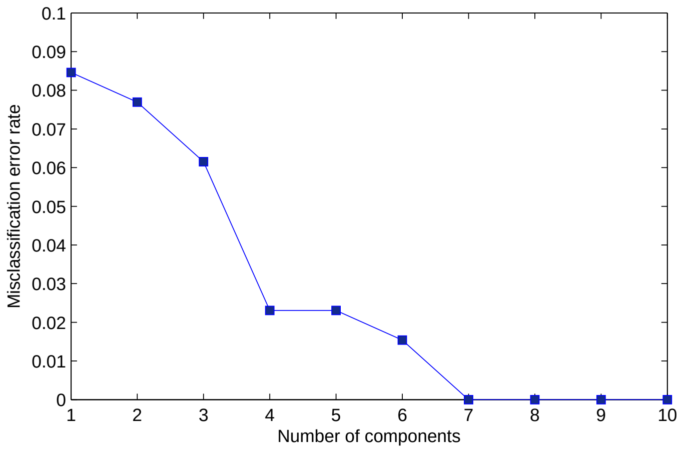


Figure 5.2: Cross validation error rates obtained for Wine data.

observations $74, 79, 96, 111$ and 122 are determined as PLS-bad leverage points when RoCPLS is employed. We deleted all PLS-bad leverage points as well as two good leverage points 70 and 97 that are identified by RoCPLS; and SIMPLS based cross-validation error rate is calculated for the new data which yields Figure 5.4. Excluding these observations clearly yielded smaller error rates, and it also indicates that $k = 4$ is the optimal number of components. This example demonstrates how outlying observations can affect the misclassification error rate based on SIMPLS as well as the decision of the k. The orthogonal-score plots based on $k = 4$ for SIMPLS and RoCPLS yielded very similar patterns observed in Figure 5.3, therefore they are not repeated here.

In order to compare the classification accuracies, 100 random partitions, into learning set containing 70% of the data and a test set containing remaining observations, are generated. We keep observations $70, 74, 79, 96, 97, 111$ and 122 in the learning set for each partition. Then, we calculated the classification rule based on learning set and evaluated estimated class membership of the observations in the test set after projecting them onto

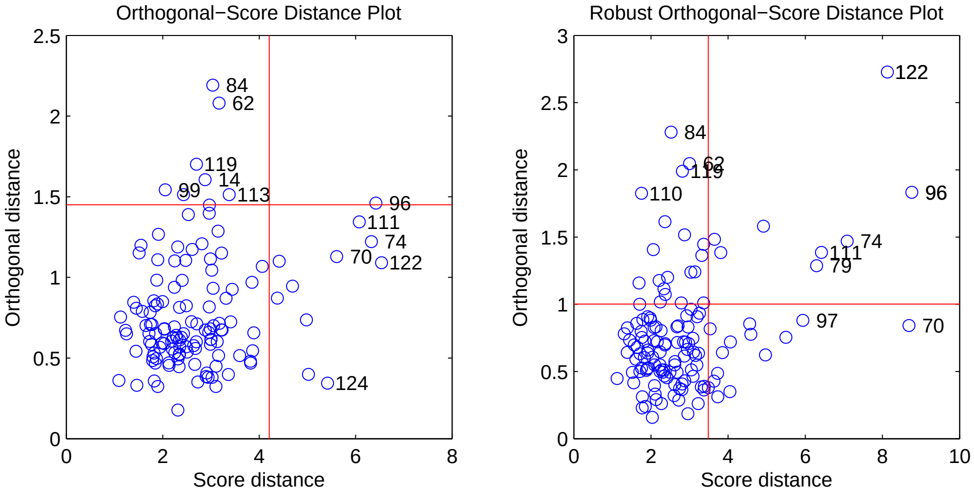


Figure 5.3: Orthogonal-score distance plots based on SIMPLS (left) and RoCPLS (right) for Wine data.

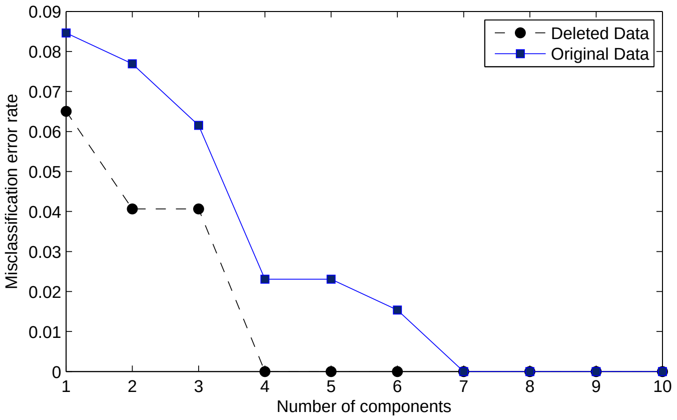


Figure 5.4: Scree plots for original and deleted Wine data.

the directions calculated from learning set and using the rule based on learning set. For $h = 1, 2, 3, 4$, the error rates, given in Table 5.1., are obtained. Both SIMPLS and RoCPLS give lower error rates than PCA does. Beside this, RoCPLS yields the smallest error rates for each h which indicates the robustness of the method.

h	SIMPLS	RoCPLS	PCA
1	0.0703	0.0677	0.0703
2	0.0736	0.0462	0.0744
3	0.0374	0.0292	0.0659
4	0.0187	0.0100	0.0362

Table 5.1: The mean misclassification error rates for Wine data based on SIMPLS, RoCPLS and PCA classification.

5.4.2 High Dimension: Colon Data

Colon data set ([2]) contains the expression levels of $p = 2000$ genes for $n = 62$ patients from two classes. 22 patients are healthy patients and 40 have colon cancer. After the data preprocessed described in Dudoit ([24]), only 1224 variables remain. Cross validation error rates indicated that $k = 4$ components result in the minimum error rate. Therefore, orthogonal-score plots for $k = 4$ components are constructed using SIMPLS and RoCPLS which are given in Figure 5.5. None of the plots indicates the existence of extreme outliers. The scatter plot of the first three components in Figure 5.6 does not display any outlying observations. It can also be seen from Figure 5.6 that classes are not completely separated. This also explains why high misclassification error rates are obtained in Table 5.2 ($k = 4$) where DG stands for Ding and Gentleman's method ([21]). As in previous example, 100 randomly splitted data sets are employed to calculate the error rates. The optimal value of k is estimated in each iteration based on the learning set. Boxplots given in Figure 5.7

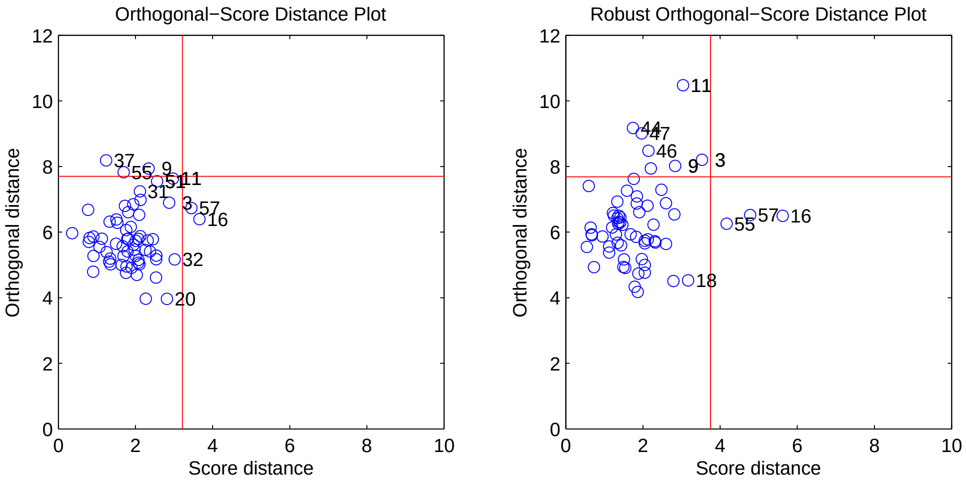


Figure 5.5: Orthogonal-score distance plots based on SIMPLS (left) and RoCPLS (right) for Colon data.

summarize the error rates calculated from each method. Clearly, all methods give better results than PCA. The numerical results and graphics show the comparable performances of SIMPLS and RoCPLS, which demonstrates the efficiency of the proposed method at uncontaminated data sets.

SIMPLS	RoCPLS	PCA	DG
0.1326	0.1363	0.2289	0.1389

Table 5.2: The mean misclassification error rates for SIMPLS, RoCPLS, PCA and DG based classification.

5.4.3 High Dimension: Leukemia Data

This data set was introduced in Golub et al. ([37]) and it contains the expression levels of 7129 genes for 47 ALL-leukemia patients and 25 AML-leukemia patients. After data preprocessing, only 500 variables remain. Leave-one-out cross validation on the whole data set indicated $k = 2$ components should be retained in the model. For $k = 2$ components,

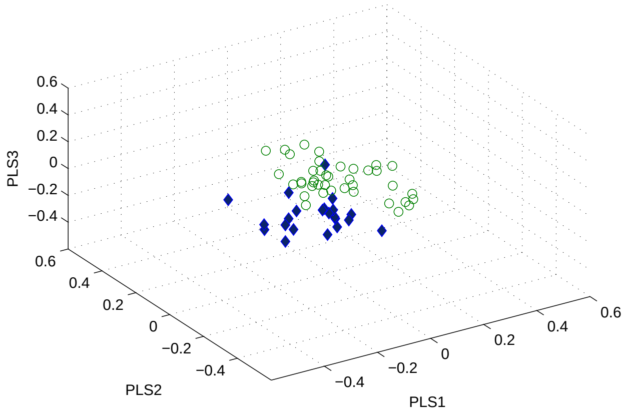


Figure 5.6: 3D scatter plot of the first three components for Colon data obtained from RoCPLS.

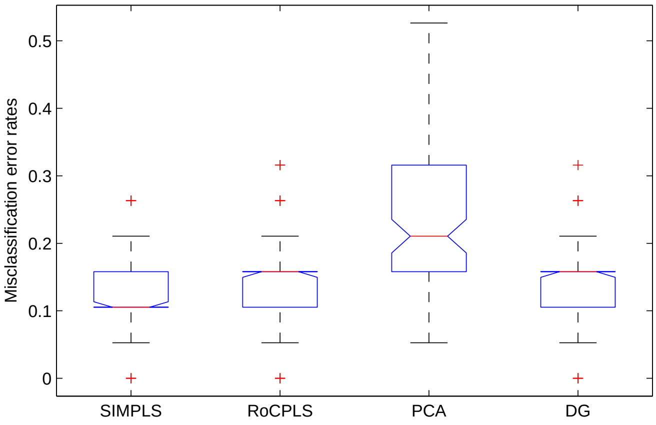


Figure 5.7: Boxplots of the error rates for SIMPLS, RoCPLS, PCA and DG based classification.

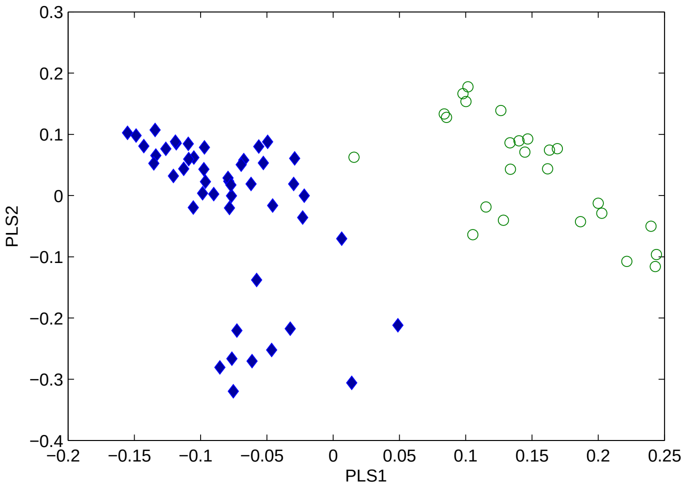


Figure 5.8: Scatter plot of the first two components for Leukemia data.

scatter plot of the first two components is constructed (Figure 5.8), where the separation between two groups can be clearly seen. The orthogonal-score plots obtained from SIMPLS and RoCPLS do not demonstrate any extreme observations in the data as in colon data set.

Leukemia data set is randomly divided into a learning set of size 50 and a test set of size 22, $N = 250$ times. For this simulation study, we introduced 0, 1, 2, and 3 outlying observations to the each class corresponding to 0%, 4%, 7% and 10% contamination, respectively. For class g, contaminated observation is generated from multivariate normal distribution with mean $(10)\mathbf{1}_{500} + \overline{x}_g$ and covariance matrix I_{500} with class label $y = g$, for $g = 1, 2$. For example, when 2 outliers are introduced for each class, the first two component plot and orthogonal-score plot, obtained from RoCPLS in Figure 5.9, indicate that these observations are PLS-bad leverage points.

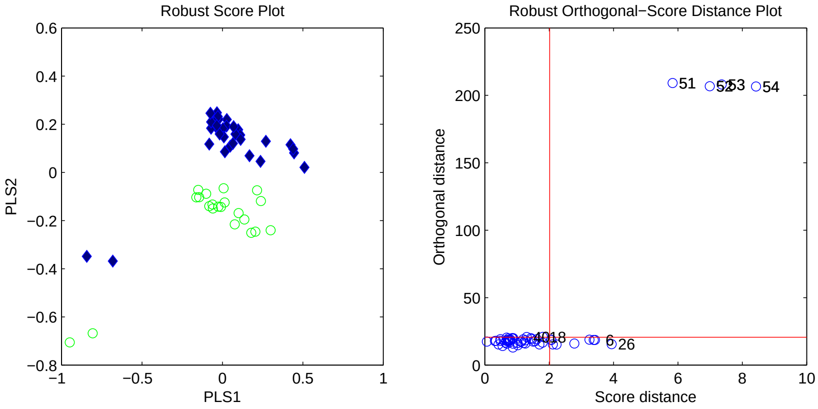


Figure 5.9: The scatter plot of first two components (left) and orthogonal-score plot (right) for Leukemia data.

As in colon data example, leave-one out method applied to clean learning set of size 50 to determine optimal number of components within each division. 60% of the time k is determined as 2, while 40% of the time $k = 1$. For the optimal value, k, the error rates based on the methods SIMPLS, RoCPLS, PCA and DG are calculated for the test set. The boxplots in Figure 5.10 summarize the simulation results. For the clean data (no outliers), all methods yield very comparable results. Once again, it can be seen that RoCPLS is an effective method for uncontaminated data. As the number of outlying observations increases, the error rates for the SIMPLS, PCA and DG increases as well. However, adding outlying observations do not affect the error rates based on RoCPLS. The main reason behind this is that the optimal directions obtained by RoCPLS are robust to outliers. In order to show that, the angle between the first PLS components, r_1, obtained from the clean and the contaminated data is calculated for SIMPLS and RoCPLS. Boxplots, in Figure 5.11, are constructed for each contamination level based on $N = 250$ divisions. The angle for SIMPLS tends to increase as contamination level increases. However, RoCPLS yield smaller

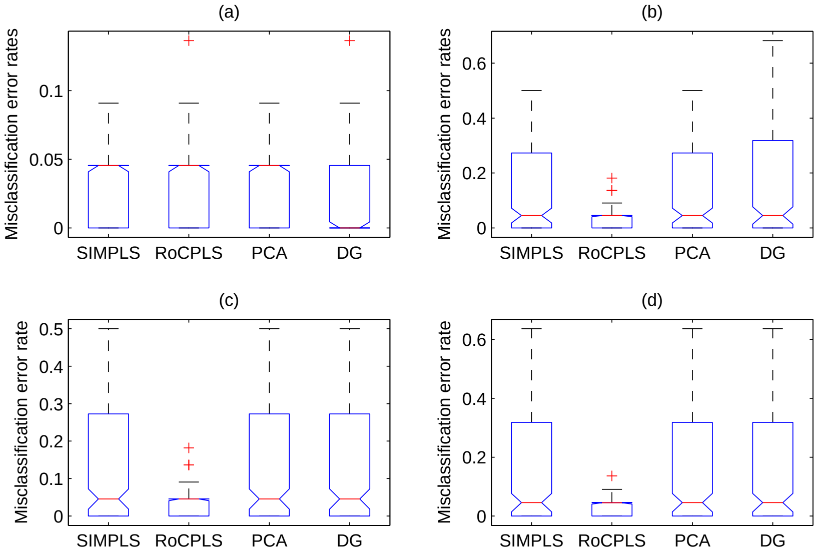


Figure 5.10: Boxplots of the error rates for (a) no outliers (b) 1 outlier (c) 2 outliers (d)3 outliers in each class.

angles than that of SIMPLS, and as contamination level increases, the results remain almost the same.

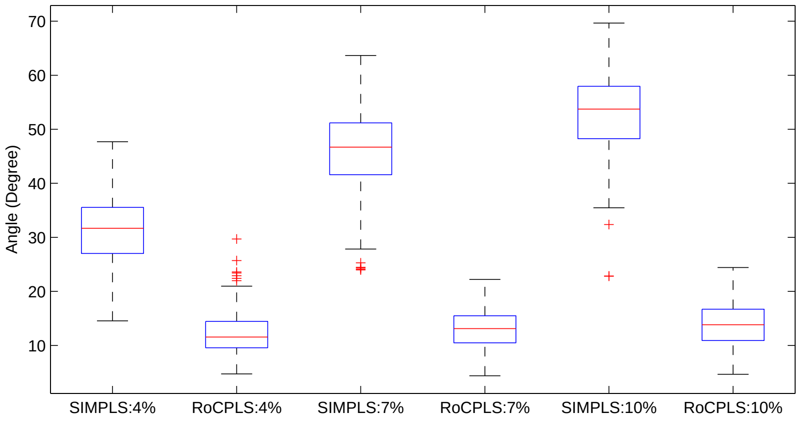


Figure 5.11: The angle between first PLS weight vectors for clean and contaminated data.

Chapter 6

Conclusions and Future Work

In this dissertation, different aspects of partial least squares methods have been studied. In this chapter, final conclusions on all results obtained throughout dissertation are summarized. We also discuss some possibilities for future research.

In Chapter 2, the main concepts of PLS are introduced and a detailed overview of its applications to different data analysis problems is given. Two important algorithms, SIMPLS and NIPALS (PLS1 and PLS2) are described. It is stated that the optimal number of components is an important issue in PLSR model building and several approaches in the literature proposed for determining optimal number of components, k, are reviewed. The connections among the biased estimation methods PLSR, PCR, and RR are examined in detail. Statistical properties of PLSR such as shrinkage, asymptotic variance and consistency are discussed. As a conclusion, computational and implementation simplicity of PLS is a strong aspect of the approach which favours PLS to be used as a first step to understand the existing relations and to analyse real world data.

In Chapter 3, a new iterative robust *external reweighting algorithm* for the regression coefficient vector, which gives low weights to points with high leverage and/or large residuals is proposed. This algorithm is carried out in two main parts: 1. obtain initial weights as robust distances from recent outlier detection methods, BACON (RoPLS1) or PCOUT (RoPLS2), to downweight outlying points in predictor X-space and/or response y-space to get an initial PLS estimate for the regression coefficient vector, 2. perform reweighted PLS regression iteratively by using the initial PLS estimate of the regression coefficient vector obtained in the first part. Both RoPLS1 and RoPLS2 can be applied to low and high dimensional explanatory variables. Simulations have shown that they are resistant towards many types of contamination, whereas their performance is also good at uncontaminated data sets. RoPLS1 is scale and orthogonal equivariant, therefore it can be preferred over RoPLS2 which is not orthogonal equivariant.

In Chapter 4, it is shown that SIMPLS algorithm is highly non-robust towards outlying observations. It is illustrated that a single sample can change the direction of the SIMPLS weight vectors and the regression estimates arbitrarily. This also appears in their unbounded influence functions. Robustness properties of RoPLS estimator of β, including influence function for low dimension, empirical influence curve for high dimensional case and finite-sample breakdown properties, are provided. It is shown that the influence function of all pairs of PLS weight vectors and of the regression estimator are bounded which makes the method resistant towards point contamination. For high-dimensional data, it is illustrated on simulated data sets that the empirical influence function remains bounded and that it can resist large fractions of contamination. The resistance of a robust statistical method to groups of outliers is another important issue which is measured by breakdown point, hence the finite sample breakdown point is determined for RoPLS1 which is approximately 40% for both low and high dimensional settings.

In Chapter 5, the effect of outliers on existing PLS classication methods is investigated and a new robust PLS algorithm for classification (RoCPLS) is proposed. It is shown that the proposed method is very effective for uncontaminated data and it yields better results when data contain outliers.

In this dissertation, we have shown promising results for RoPLS and RoCPLS as a data mining tool and high-dimensional classifier, respectively. There is, of course, more research to be done. We would like to extend RoPLS to multivariate case and RoCPLS to multi-class case. Also, we would like to employ the influence function of the RoPLS1 estimator for the robust estimation of its variance. The relationship between the PLS components and the variable selection is going to be explored to build a robust variable selection method in the future.

Bibliography

[1] Almåy, T., "A Simulation Study on Comparison of Prediction Methods When Only a Few Components are Relevant", *Computational Statistics and Data Analysis*, 21, 87-107, 1996.

[2] Alon, U., Barkai, N., Notterman, D. A., Gish, K., Ybarra, S., Mack, D., and Levine, A., "Broad Patterns of Gene Expression Revealed by Clustering Analysis of Tumor and Normal Colon Tissues Probed by Oligonucleotide Arrays", *Proceedings of the National Academy of Sciences*, 96, 6745-6750, 1999.

[3] Anderson, R.L. and Bancroft, T.A., *Statistical Theory in Research.* McGraw-Hill, New York, 1952.

[4] Barker, M., and Rayens, W., "Partial Least Squares for Discrimination", *Journal of Chemometrics*, 17, 166-173, 2003.

[5] Billor, N., Hadi, A., and Velleman, P., "BACON: Blocked Adaptive Computationally-Efficient Outlier Nominators", *Computational Statistics and Data Analysis*, 34, 279-298, 2000.

[6] Billor, N., Chatterjee, S., and Hadi, A. S., "Iteratively Re-weighted Least Squares Method for Outlier Detection in Linear Regression", *American Journal for Mathematical and Management Science*, 26, 3, 229-252, 2006.

[7] Bo, T. H., and Jonassen, I., "New Feature Subset Selection Procedures for Classification of Expression Profiles", *Genome Biology* , 3, R17, 2002.

[8] Boulesteix, A.L., "PLS Dimension Reduction for Classification with Microarray Data", *Statistical Applications in Genetics and Molecular Biology*, 3, Article: 33, 2004.

[9] Breiman, L. and Friedman, J., "Predicting Multivariate Responses in Multiple Linear Regression", *Journal of the Royal Statistical Society, B*, 59, 3-54, 1997.

[10] Chatterjee, S. and Mächler, M., "Robust Regression: A Weighted Least Squares Approach", *Communications in Statistics: Theory and Methods*, 26, 1381-1394, 1997.

[11] Chilingaryan, A., Gevorgyan, N., Vardanyan, A., Jones, D., and Szabo, A., "Multivariate Approach for Selecting Sets of Differentially Expressed Genes", *Mathematical Biosciences*, 176, 59-72, 2002.

[12] Croux, C. and Haesbroeck, G., "Principal Components Analysis Based on Robust Estimators of the Covariance or Correlation Matrix: Influence Functions and Efficiencies", *Biometrika*, 87, 603-618, 2000.

[13] Cummins, D.J. and Andrews, C.W., "Iteratively Reweighted Partial Least Squares: A Performance Analysis by Monte Carlo Simulation", *Journal of Chemometrics*, 9, 489-507, 1995.

[14] Dai, J. J., Lieu, L., and Rocke, D., "Dimension Reduction for Clasification with Gene Expression Microarray Data", *Statistical Applications in Genetics and Molecular Biology*, 5, Article: 6, 2006.

[15] De Jong, S., "SIMPLS: An Alternative Approach to Partial Least Squares Regression", *Chemometrics and Intelligent Laboratory Systems*, 18, 251-263, 1993.

[16] De Jong, S., "PLS Fits Closer than PCR", *Journal of Chemometrics*, 7, 551-557, 1993.

[17] De Jong, S., "PLS Shrinks", *Journal of Chemometrics*, 9, 323-326, 1995.

[18] Denham, M.C., "Prediction Intervals in Partial Least Squares", *Journal of Chemometrics*, 11, 39-42, 1997.

[19] Denham, M. C., "Choosing the Number of Factors in Partial Least Squares Regression: Estimating and Minimizing the Mean Squared Error of Prediction", *Journal of Chemometrics*, 14, 351-361, 2000.

[20] Dettling, M., Bühlmann, P., "Boosting for Tumor Classification with Gene Expression Data", *Bioinformatics*, 19, 1061-1069, 2003.

[21] Ding, B. and Gentleman, R., "Classification Using Generalized Partial Least Squares", *Bioconductor Project Working Papers*, Technical Report 5, 2004.

[22] Donoho D.L., "*Breakdown Properties of Multivariate Location Estimators*", Ph.D. Qualifying paper, Harvard University, 1982.

[23] Draper, N. and Smith, H., *Regression Analysis by Example*, Wiley, New York, 1981.

[24] Dudoit, S., Fridlyand, J., and Speed, T. P., "Comparison of Discrimination Methods for the Classification of Tumors Using Gene Expression Data", *Journal of the American Statistical Association*, 97, 77-87, 2002.

[25] Dudoit, S., Shaffer, J. P., and Boldrick, J. C., "Multiple Hypothesis Testing in Microarray Experiments", *Statistical Science*, 18, 71-103, 2003.

[26] Filzmoser, P., Maronna, R. and Werner, M., "Outlier Identification in High Dimensions", *Computational Statistics and Data Analysis*, 52, 1694-1711, 2008.

[27] Firth, D., "Bias Reduction of Maximum Likelihood Estimates", *Biometrika*, 80, 27-38, 1993.

[28] Fisher, R. A., "The Use of Multiple Measurements in Taxonomic Problems", *Annals of Eugenics*, 7, 179-188, 1936.

[29] Forina, M., Armanino, C., Castino, M. and Ubigli, M., "Parvus - An Extendible Package for Data Exploration, Classification and Correlation", *Vitis*, 25, 189-201, 1986.

[30] Fort, G. and Lambert-Lacroix, S., "Classification Using Partial Least Squares with Penalized Logistic Regression", *Bioinformatics*, 21(7), 1104-1111, 2005.

[31] Frank, I.E. and Friedman, J.H., "A Statistical View of Some Chemometrics Regression Tools", *Technometrics*, 35, 109-147, 1993.

[32] Friedman, J., "Regularized Discriminant Analysis", *Journal of American Statistical Association*, 84, 165-175, 1989.

[33] Garthwaite P. H., "An interpretation of Partial Least Squares", *Journal of the American Statistical Association*, 89, 122-127, 1994.

[34] Geladi, P. and Kowalski, B. R., "Partial Least Squares Regression: A Tutorial", *Analytica Chimica Acta*, 185, 1-17, 1986.

[35] Gil, J.A. and Romera, R., "On Robust Partial Least Squares Methods". *Journal of Chemometrics*, 12, 365-378, 1998.

[36] Golub, G.H. and Van Loan C.F., *Matrix Computations* , Johns Hopkins University Press, Baltimore, 1996.

[37] Golub, T., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P., Coller, H., Loh, M. L., Downing, J., Caligiuri, M. A., Bloomfield, C. D., and Lander, E. S., "Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring", *Science*, 286, 531-537, 1999.

[38] Griep, M.I., Wakeling, I.N., Vankeerberghen, P., and Massart, D.L., "Comparison of Semirobust and Robust Partial Least Squares Procedures", *Chemometrics and Intelligent Laboratory Systems*, 29, 37–40, 1995.

[39] Hadi, A. S. and Ling, R. F., "Some Cautionary Notes on the Use of Principal Component Regression", The American Statistician, 52, 15-19, 1998.

[40] Hampel, F. R., Ronchetti, E. M., Rousseeuw, P. J. and Stahel, W. A., *Robust Statistics: The Approach Based on Influence Functions*, John Wiley and Sons, New York, 1986.

[41] Helland, I., "On the Structure of Partial Least Squares Regression", *Communications in Statistics: Simulation and Computation*, 17, 581-607, 1988.

[42] Helland, I., "Partial Least Squares Regression and Statistical Models", *Scandivian Journal of Statistics*, 17, 97-114, 1990.

[43] Helland, I., "Rotational Symmetry, Model Reduction and Optimality of Prediction from the PLS Population Model", *Chemometrics and Intelligent Laboratory Systems*, 68, 53-60, 2003.

[44] Hoerl, A.E. and Kennard, R. W., "Ridge Regression: Biased Estimation for Nonorthogonal Problems", *Technometrics,* 8, 27-51, 1970.

[45] Höskuldsson, A., "PLS Regression Methods", *Journal of Chemometrics*, 2, 211-228, 1988.

[46] Höskuldsson, A., "Dimension of Linear Models", *Chemometrics and Intelligent Laboratory Systems*, 32, 37-55, 1996.

[47] Huber, P.J., *Robust Statistics*, John Wiley and Sons, New York, 1981.

[48] Hubert, M. and Vanden Branden, K., "Robust Methods For Partial Least Squares Regression", *Journal of Chemometrics*, 17, 537–549, 2003.

[49] Hubert, M., Rousseeuw, and P. J., Vanden Branden, K., "ROBPCA: A New Approach to Robust Principal Component Analysis", *Technometrics*, 47(1), 64-79, 2005.

[50] Kavšek, B. *Partial Least Squares Regression and Its Robustification*, Diploma Thesis, Technische Universitt Wien, 2002.

[51] Li, K.C., "Sliced Inverse Regression for Dimension Reduction", *Journal of American Statistical Association*, 86, 316-342, 1991.

[52] Li, L., Weinberg, C. R., Darden, T. A., and Pedersen, L. G., "Gene Selection for Sample Classification Based on Gene Expression: Study of Sensitivity to Choice of Parameters of the GA/KNN Method", *Bioinformatics*, 17, 1131-1142, 2001.

[53] Lindjæerde, O.C. and Christophersen, N., "Shrinkage Structure of Partial Least Squares Regression", *Scandinavian Journal of Statistics*, 27, 459-473, 2000.

[54] Mallows, C. P., "On Some Topics in Robustness", *Technical Memorandum* , Murray Hill, New Jersey, 1975.

[55] Manne, R., "Analysis of Two Partial-Least-Squares Algorithms for Multivariate Calibration", *Chemometrics and Intelligent Laboratory Systems*, 2, 187-197, 1987.

[56] Maronna, R. and Yohai, V., "The Behavior of the Stahel-Donoho Robust Multivariate Estimator", *Journal of the American Statistical Association*, 90, 330-341, 1995.

[57] Maronna, R. and Zamar, V., "Robust Estimates of Location and Dispersion for High Dimensional Data Sets", *Technometrics*, 44(4), 307-317, 2002.

[58] Maronna,R., Martin, D. and Yohai,V., *Robust Statistics Theory and Methods*, John Wiley and Sons, New York, 2006.

[59] Martens, H. and Naes, T., *Multivariate Calibration*, Wiley, Chichester, 1989.

[60] Marx, B.D., "Iteratively Reweighted Partial Least Squares Estimation for Generalized Linear Regression", *Technometrics*, 38(4), 374-381, 1996.

[61] Marx, B.D. and Eilers, P.H., "Generalized Linear Regression on Sampled Signals and Curves: A P-spline Approach". *Technometrics*, 41, 1-13, 1999.

[62] Massy, W.F., "Principal Component Regression in Explanatory Statistical Research", *Journal of the American Statistical Association*, 60, 234-246, 1965.

[63] Naes, T., "Multivariate Calibration When the Error Covariance Matrix is Structured", *Technometrics*, 27, 301-311, 1985.

[64] Naik, P. and Tsai, C., "Partial Least Squares Estimator for Single-Index Models", *Journal of the Royal Statistical Society: B* , 62, 763-771, 2000.

[65] Nguyen, D.V. and Rocke, D.M., "Tumor Classification by Partial Least Squares Using Microarray Gene Expression Data", *Bioinformatics*, 18, 39-50, 2002(a).

[66] Nguyen, D.V. and Rocke, D.M., "Multi-Class Cancer Classification Via Partial Least Squares Using Gene Expression Profies".*Bioinformatics*, 18, 1216-1226, 2002(b).

[67] Osborne, B.G., Fearn, T. , Miller, A.R. and Douglas, S., "Application of Near Infrared Reflectance Spectroscopy to the Compositional Analysis of Biscuits and Biscuit Dough", *Journal of Scientific Food Agriculture*, 35, 99-105, 1984.

[68] Peña, D. and Prieto, F.,"Multivariate Outlier Detection and Robust Covariance Matrix Estimation", *Technometrics* , 43, 286-310, 2001.

[69] Phatak, A. and De Jong, S., "The Geometry of Partial Least Squares", *Journal of Chemometrics*, 11, 311-338, 1997.

[70] Phatak, A. and De Hoog, F., "Exploiting the Connection Between PLS, Lanczos Methods and Conjugate Gradients: Alternative Proofs of Some Properties of PLS", *Journal of Chemometrics*, 16, 361-367, 2002.

[71] Phatak, A., Reilly, P. M., and Pendilis, A., "The Asymptotic Variance of the Univariate PLS Estimator", *Linear Algebra and Its Applications*, 354, 245-253, 2002.

[72] Rocke, D., "Robustness Properties of S-estimators of Multivariate Location and Shape in High Dimension", *Annals of Statistics*, 24, 1327-1345, 1996.

[73] Rousseeuw, P. J. and Leroy, A. M., *Robust Regression and Outlier Detection*, John Wiley and Sons, New York, 1987.

[74] Rousseeuw P. J. and Van Driessen K., "A Fast Algorithm for the Minimum Covariance Determinant Estimator", *Technometrics*, 41, 212-223, 1999.

[75] Serneels, S., Croux, C., and Van Espen, "Influence Properties of Partial Least Squares Regression", *Chemometrics and Intelligent Laboratory Systems*, 71, 13-20, 2004.

[76] Serneels, S., Croux, C., Filzmoser, P., and Van Espen, P.J., "Partial Robust M-Regression", *Chemometrics and Intelligent Laboratory Systems*, 79, 55-64, 2005.

[77] Sibson, R., "Studies in the Robustness of Multidimensional Scaling: Perturbation Analysis of Classical Scaling", *Journal of the Royal Statistical Society: B* 41, 217-229, 1979.

[78] Stahel W.A., *Robust Estimation: Infinitesimal Optimality and Covariance Matrix Estimators*, Ph.D. thesis, ETH, Zürich, 1981.

[79] Staudte, R. G. and Sheather, S. J., "*Robust Estimation and Testing*", John Wiley and Sons, New York, 1990.

[80] Stone, M. and Brooks, R.J., "Continuum Regression: Cross-validated Sequentially Constructed Prediction Embracing Ordinary Least Squares, Partial Least Squares and Principal Components Regression", *Journal of the Royal Statistical Society: B*, 52, 237-269, 1990.

[81] Sundberg, R., "Continuum Regression and Ridge Regression", *Journal of the Royal Statistical Society: B*,55, 653-659, 1993.

[82] Vanden Branden, K. and Hubert, M., "Robustness Properties of a Robust PLS Regression Method, *Analytica Chimica Acta*, 515, 229-241, 2004.

[83] Wakeling, I. N. and Macfie, H.J.H., "A Robust PLS Procedure", *Journal of Chemometrics*, 6,189–198, 1992.

[84] Wangen, L.E. and Kowalsky,B.R., "A Multiblock Partial Least Squares Algorithm for Investigating Complex Chemical Systems", *Journal of Chemometrics*, 3, 3-20, 1989.

[85] Wiklund, S.,Nilsson, D., Eriksson, L., Sjöström, M., Wold, S. and Faber, K., " A Randomization Test for PLS Component Selection", *Journal of Chemometrics*, 21, 427-439, 2007.

[86] Wilcox, R. R., *Introduction to Robust Estimation and Hypothesis Testing*, 2nd Edition, Elsevier, 2005.

[87] Wold, H., "Estimation of Principal Components and Related Models by Iterative Least Squares", *Multivariate Analysis*, Academic Press, New York, 391-420, 1966.

[88] Wold H., "Path Models with Latent Variables: The NIPALS Approach", *Quantitative Sociology: International perspectives on mathematical and statistical model building*, Academic Press, 307-357, 1975.

[89] Wold S., "Cross-validatory Estimation of the Number of Components in Factor and Principal Components Models", *Technometrics*, 20, 397-405, 1978.

[90] Woodruff, D. and Rocke, D., "Computable Robust Estimation of Multivariate Location and Shape in High Dimension Using Compound Estimators", *Journal of the American Statistical Association*, 89, 888-896, 1994.

Appendix

X	: $n \times p$ matrix of explanatory variables
Y	: $n \times q$ matrix of response variables
y	: $n \times 1$ vector of response variable
$\mathbf{a}_i$	: jth column of a matrix A (column vector)
a_i	: ith row of a matrix A (column vector)
A'	: Transpose of matrix A
$A^{\perp}$	: Orthogonal complement of A
A^{-1}	: Inverse of matrix A
A^{+}	: Moore-Penrose inverse of matrix A
I_m	: $m \times m$ identity matrix
$\mathbf{1}_m$	: $m \times 1$ vector of ones
$\|\mathrm{a}\|$	: Euclidian norm of a vector a
$\propto$	: Proportional to