

EXTENDING PARTIAL LEAST SQUARES REGRESSION

ATHANASSIOS KONDYLIS

UNIVERSITY OF NEUCHÂTEL

Outline

Multivariate Calibration in Chemometrics

PLS regression (PLSR) and the PLS1 algorithm

PLS1 from a statistical point of view

PLS1 statistical properties

PLAD regression

PRLS regression

Model Selection using Cross Validation - Goodness-of-fit

Examples

Conclusions

Multivariate Calibration in Chemometrics (1)

Calibration is generally concerned with predicting \mathbf{Y} from the data \mathbf{X} . This is done using the transfer function f which has to be defined.

\mathbf{Y} are usually chemical concentrations and \mathbf{X} are spectral measurements.

\mathbf{X} are often recorded using an instrument as a spectrometer and are transformed to $\hat{\mathbf{Y}}$ via the prediction model $\hat{\mathbf{Y}} = \hat{f}(\mathbf{X})$.

We have to learn how to predict \mathbf{Y} from \mathbf{X} . Here statistics maybe valuable.

Multivariate Calibration in Chemometrics (2)

$$\mathbf{X}_{(n \times p)} = \begin{pmatrix} x_{11} & x_{12} & \dots & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & \dots & x_{np} \end{pmatrix}, \quad \mathbf{Y}_{(n \times m)} = \begin{pmatrix} y_{11} & \dots & y_{1m} \\ y_{21} & \dots & y_{2m} \\ \vdots & \ddots & \vdots \\ y_{n1} & \dots & y_{nm} \end{pmatrix}$$

\mathbf{X} are n spectral measurements at p different wavelengths, and \mathbf{Y} are the absorbances of the m chemical constituents for the n samples.

Multivariate Calibration in Chemometrics (3)

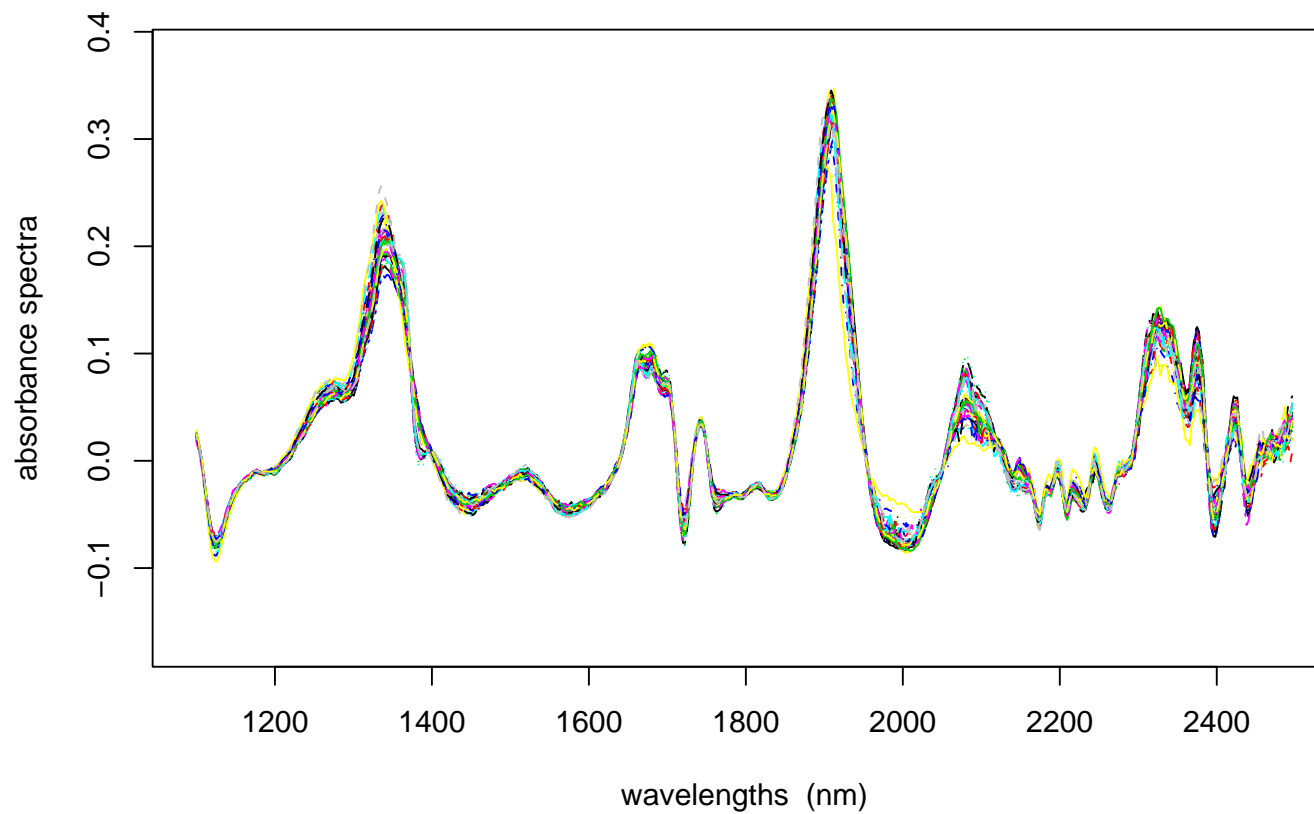


Figure 1: Digitized spectra of fat content scanned from 60 biscuits.

Multivariate Calibration in Chemical Engineering and the bilinear factor model

There is a latent chemical structure expressed in a few latent variables $\mathbf{T}_k = (\mathbf{t}_1, \dots, \mathbf{t}_k)$, which are related to both \mathbf{X} and \mathbf{Y} through the bilinear factor model:

$$\mathbf{Y} = \mathbf{T} \mathbf{q} + \boldsymbol{\epsilon},$$

$$\mathbf{X} = \mathbf{T} \mathbf{p} + \mathbf{f}.$$

The number of the finally retained latent variables are often called **chemical rank**. We note it as k . It is $k \ll p$.

NIPALS for PLS

$\mathbf{X}_0 = \mathbf{X}$, and $\mathbf{Y}_0 = \mathbf{Y}$;

1. **Start** with Y-score $\mathbf{u}_1 = \mathbf{y}$, one of the columns of \mathbf{Y} . For a single response, $\mathbf{u}_1 = \mathbf{y}$.

while a certain convergence criterion is not fulfilled **do**

2a. Compute X-weight vector \mathbf{w} according to $\mathbf{w} = \frac{\mathbf{X}^T \mathbf{u}}{\mathbf{u}^T \mathbf{u}}$, and norm \mathbf{w} to 1;

2b. Compute X-scores \mathbf{t} according to $\mathbf{t} = \mathbf{X}\mathbf{w}$;

2c. Compute Y-weight \mathbf{c} according to $\mathbf{c} = \frac{\mathbf{Y}^T \mathbf{t}}{\mathbf{t}^T \mathbf{t}}$;

2d. Update Y-scores according to $\mathbf{u} = \frac{\mathbf{Y}\mathbf{c}}{\mathbf{c}^T \mathbf{c}}$;

2e. Check for convergence.

end while

3. Compute X-loadings vector \mathbf{p} according to $\mathbf{p} = \frac{\mathbf{t}^T \mathbf{X}}{\mathbf{t}^T \mathbf{t}}$.

4. Remove component \mathbf{t} from both \mathbf{X} and \mathbf{Y} . Take as new data (\mathbf{X}, \mathbf{Y})

$$\mathbf{X}^{new} = \mathbf{X}^{old} - \mathbf{t}\mathbf{p} \text{ and } \mathbf{Y}^{new} = \mathbf{Y}^{old} - \mathbf{t}\mathbf{c}.$$

5. Go to **start** in order to extract the next factor.

Partial Least Squares Regression (PLSR) algorithms

NIPALS was presented in "Wold, H. (1975) Soft modelling by Latent variables. The nonlinear iterative partial least squares (NIPALS) approach *Perspectives in Probability and Statistics, In Honor of M.S.Bartlett*".

It has been the basic algorithm for PLS regression. Yet, it is the most expensive computationally.

Slightly different PLSR algorithms are available in the literature. We notice the PLSR algorithm of Martens (called orthogonal loadings PLSR), the SIMPLS algorithm, as well as the PLS1 and PLS2 algorithms (the orthogonal scores PLSR).

We restrict our attention here on the PLS1 algorithm. That is, the PLSR when the response corresponds to a single vector \mathbf{y} .

PLS1 Algorithm

$$\mathbf{X}_0 = \mathbf{X}, \quad \text{and} \quad \mathbf{y}_0 = \mathbf{y};$$

$$k \leftarrow 1;$$

while a certain model selection criterion is not fulfilled **do**

$$\mathbf{w}_k = \frac{\mathbf{X}_{k-1}^T \mathbf{y}_{k-1}}{\mathbf{y}_{k-1}^T \mathbf{y}_{k-1}}, \quad \text{and normalise } \mathbf{w}_k \text{ to } 1;$$

$$\mathbf{t}_k = \mathbf{X}_{k-1} \mathbf{w}_k;$$

$$\mathbf{X}_k = \mathbf{X}_{k-1} - \mathbf{t}_k \mathbf{p}_k^T, \quad \text{where } \mathbf{p}_k = \frac{\mathbf{X}_{k-1}^T \mathbf{t}_k}{\mathbf{t}_k^T \mathbf{t}_k};$$

$$k \leftarrow k + 1;$$

end while

Give the final PLS1 regression model $\hat{\mathbf{y}} = \mathbf{T}_k \hat{\mathbf{q}}$, where $\mathbf{T}_k = (\mathbf{t}_1, \dots, \mathbf{t}_k)$.

PLS1 in a statistical framework (1)

The columns of the matrix $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_j, \dots, \mathbf{x}_p)$ and the response \mathbf{y} are realizations of the random variables X_j and Y .

They come from a population with distribution $F_{X_j}(\boldsymbol{\theta}_j)$ and $F_Y(\boldsymbol{\phi})$ respectively, where $\boldsymbol{\theta}_j$ and $\boldsymbol{\phi}$ are parameter vectors.

For these random variables $E(X_j)$, $E(Y)$, $E(X_j^2)$ and $E(Y^2)$ exist.

PLS1 in a statistical framework (2)

In PLS data are commonly centered or scaled. The statistical interpretation of the PLS1 algorithm is much more interesting, since:

1. The loading vector \mathbf{w}_k in PLS1 algorithm corresponds to

$$\mathbf{w}_k = (w_{1,k}, \dots, w_{p,k}) \text{ where } w_{j,k} = \text{cov}(\mathbf{x}_{j,k-1}, \mathbf{y}_{k-1}),$$

with $j = 1, \dots, p$ and p denoting the number of the predictors.

2. The data \mathbf{X}_k are orthogonalized with respect to the derived component \mathbf{t}_{k+1} at each iteration k .

We denote

$$\mathbf{x}_{j,k} = \mathbf{x}_{j,k-1} - E(\mathbf{x}_{j,k-1} | \mathbf{t}_k).$$

This results to mutually orthogonal components.

3. Ordinary least squares regression is used to regress the response \mathbf{y} on the derived components in order to construct the final PLS1 model. We denote

$$\hat{\mathbf{y}} = \mathbf{T}_k \hat{\mathbf{q}}_k,$$

where $\mathbf{T}_k = (\mathbf{t}_1, \dots, \mathbf{t}_k)$ is the sequence of the derived components and $\hat{\mathbf{q}}_k$ is the estimated regression coefficient vector.

4. The chemical rank k is the number of finally retained components. The determination of k is based upon several model selection methods. Cross validation is commonly used.

Univariate Partial Least Squares Regression (PLS1)

For $i = 1, \dots, n$ and $j = 1, \dots, p$, $\mathbf{X}_{(n,p)}$ and $\mathbf{y}_{(n,1)}$

1. Center or standardise both \mathbf{X} and \mathbf{y} .

2. For $k = 1, \dots, p$

Compute \mathbf{w}_k according to: $\arg_{\mathbf{w}} \max \{ \text{cov}(\mathbf{X}_{k-1} \mathbf{w}_k, \mathbf{y}) \}$ subject to $\mathbf{w}_k^T \mathbf{w}_k = 1$.

Derive component $\mathbf{t}_k = \sum_{j=1}^p w_{j,k} \mathbf{x}_{j,k-1}$ where $\mathbf{w}_k = (w_{1,k}, \dots, w_{p,k})$.

Orthogonalise each $\mathbf{x}_{j,k-1}$ with respect to \mathbf{t}_k : $\mathbf{x}_{j,k} = \mathbf{x}_{j,k-1} - E(\mathbf{x}_{j,k-1} | \mathbf{t}_k)$.

3. Give the resulting sequence of the fitted vectors $\hat{\mathbf{y}}_{\mathbf{k}} = \mathbf{E}(\mathbf{y} | \mathbf{t}_1, \dots, \mathbf{t}_k) = \mathbf{T}_{\mathbf{k}} \hat{\mathbf{q}}_{\mathbf{k}}$, and finally

recover the implied regression coefficients (regression coefficients on \mathbf{x}_j) according to

$\hat{\boldsymbol{\beta}}_{\mathbf{k}} = \mathbf{W}_{\mathbf{k}} \hat{\mathbf{q}}_{\mathbf{k}}$, where $\mathbf{T}_{\mathbf{k}} = (\mathbf{t}_1, \dots, \mathbf{t}_k)$, $\hat{\mathbf{q}}_{\mathbf{k}} = (\hat{q}_1, \dots, \hat{q}_k)$, and $\mathbf{W}_{\mathbf{k}} = (\mathbf{w}_1, \dots, \mathbf{w}_k)$.

Good Statistical properties of PLS1 (1)

Dimension Reduction. PLS1 uses a small k of orthogonal derived components instead of the large p collinear predictors. It derives its components according to

$$\max\{\text{corr}^2(\mathbf{X}\mathbf{w}, \mathbf{y}) \text{ var}(\mathbf{X}\mathbf{w})\}$$

$$\text{subject to } \mathbf{w}^T \mathbf{w} = 1 \text{ and } \text{corr}(\mathbf{X}\mathbf{w}, \mathbf{X}\mathbf{w}') = 0 \text{ for } \mathbf{w} \neq \mathbf{w}'$$

Higher dimension reduction ability compared to similar methods as for example the Principal Components Regression.

Shrinkage properties. PLS1 implied coefficients respect

$$\hat{\beta}_k^{PLS} \leq \hat{\beta}^{OLS}$$

with the equality for $k = p$ and orthogonal designs.

Good Statistical properties of PLS1 (2)

Lower MSE of the coefficients estimates compared to ordinary regression techniques.

$$MSE(\hat{\beta}_k^{PLS}) \leq MSE(\hat{\beta}^{OLS}).$$

Higher precision in prediction compared to ordinary least squares.

PLSR methods have been extensively used to analyse **fat data**, that is when $n \ll p$.

Extensions of PLS1

In a statistical perspective, and accepting the statistical interpretation of PLS1 algorithm, several extensions may be attempted in order to test whether new algorithms can handle open questions concerning

1. Non linearities,
2. Heteroskedasticity,
3. Outliers and heavy tailed error distributions.

Covariance estimates and regression in PLS1 can be modified in order to extend the PLS1 algorithm.

We distinguish between $n > p$ and $n \ll p$ cases. We focus on the latter.

Partial Least Absolute Deviations Regression (1)

Why should we extend PLS1 to PLAD ?

-PLS1 uses ordinary least squares in order to regress the response on the derived components.

Therefore, it is sensitive to outliers and to non-normal or heteroscedastic error term.

-PLAD method uses the LAD regression which models the conditional median of the response instead of the conditional mean of least squares. We denote $\hat{\mathbf{y}}_k^{lad}$ for the LAD fitted values.

-PLS1 derives its components towards directions where the $\text{cov}(\mathbf{X}, \mathbf{y})$ is maximised. Covariance estimates based on crossproducts may mislead these directions.

-PLAD algorithm extracts components from the direction where

$$w_j^M = \frac{1}{4} (\text{mad}_n^2(\mathbf{x}_j + \mathbf{y}) - \text{mad}_n^2(\mathbf{x}_j - \mathbf{y})),$$

is maximised. The use of the $\text{mad}_n^2(\cdot)^a$ instead of common $\text{var}(\cdot)$ modifies the covariance estimates. It protects the derived components from abnormal observations.

^a mad_2 corresponds to the median absolute deviations given by: $\text{mad}_n^2(X) = \text{median}(|\mathbf{X} - \text{median}(\mathbf{X})|)$.

Partial Least Absolute Deviations Regression (2)

For $i = 1, \dots, n$ and $j = 1, \dots, p$, $\mathbf{X}_{(n,p)}$ and $\mathbf{y}_{(n,1)}$

1. Center or standardise both \mathbf{X} and \mathbf{y} .

2. For $k = 1, \dots, p$

Compute \mathbf{w}_k^M according to: $w_{j,k}^M = \frac{1}{4}(\text{mad}_n^2(\mathbf{x}_j + \mathbf{y}) - \text{mad}_n^2(\mathbf{x}_j - \mathbf{y}))$.

Scale \mathbf{w}_k^M to 1.

Derive component $\mathbf{t}_k = \sum_{j=i}^p w_{j,k}^M \mathbf{x}_{j,k-1}$ where $\mathbf{w}_k^M = (w_{1,k}^M, \dots, w_{j,k}^M)$.

Orthogonalise each $\mathbf{x}_{j,k-1}$ with respect to \mathbf{t}_k : $\mathbf{x}_{j,k} = \mathbf{x}_{j,k-1} - E(\mathbf{x}_{j,k-1} | \mathbf{t}_k)$.

3. Give the resulting sequence of the fitted vectors $\hat{\mathbf{y}}_k^{lad} = \mathbf{T}_k \hat{\mathbf{q}}_k^{lad}$, and finally recover the implied regression coefficients (regression coefficients on \mathbf{x}_j) according to $\hat{\boldsymbol{\beta}}_k = \mathbf{W}_k^M \hat{\mathbf{q}}_k^{lad}$, where $\mathbf{T}_k = (\mathbf{t}_1, \dots, \mathbf{t}_k)$, $\hat{\mathbf{q}}_k^{lad} = (\hat{q}_1^{lad}, \dots, \hat{q}_k^{lad})$, and $\mathbf{W}_k^M = (\mathbf{w}_1^M, \dots, \mathbf{w}_k^M)$.

Partial Reweighted Least Squares Regression (1)

PRLS regression uses a weight vector \mathbf{h}_k in order to classify at each iteration the observations in *outliers* and *non outliers*. The latter is obtained while regressing the predictors and the response on the derived components using Least Trimmed Squares (LTS). The weight vector \mathbf{h}_k is calculated according to

$$h_{i,k} = \begin{cases} 1 & \text{if } \epsilon_i \leq c, \\ 0 & \text{if } \epsilon_i \text{ otherwise,} \end{cases} \quad (1)$$

where ϵ_i are the scaled residuals, and c is a critical value, usually set to 2.5. For more information about LTS and the choice of the critical value one can see "Leroy, A. and Rousseeuw, P.J. (1987) *Robust Regression & Outlier Detection*, John Wiley & Sons". The weight vectors construct, at each iteration k , the weight matrix $\mathbf{H}_k = \{\mathbf{h}_k^{\mathbf{x}_1}, \dots, \mathbf{h}_k^{\mathbf{x}_p}, \mathbf{h}_k^{\mathbf{y}}\}$. Matrix \mathbf{H}_k is used in order to compute the weighted covariance estimate

$$\text{cov}(\mathbf{x}_j, \mathbf{y}) = \frac{\sum_{i=1}^n h_i^{\mathbf{x}_j} (x_{ij} - \bar{x}_j^c) h_i^{\mathbf{y}} (y_i - \bar{y}^c)}{(\sum_{i=1}^n h_i^{\mathbf{x}_j} h_i^{\mathbf{y}}) - 1}, \quad (2)$$

where

$$\bar{x}_j^c = \frac{\sum_{i=1}^n h_i^{\mathbf{x}_j} x_{ij}}{\sum_{i=1}^n h_i^{\mathbf{x}_j}} \quad \text{and} \quad \bar{y}^c = \frac{\sum_{i=1}^n h_i^{\mathbf{y}} y_i}{\sum_{i=1}^n h_i^{\mathbf{y}}}, \quad (3)$$

and to extract the PRLS components for the following iteration. The subscript k is omitted from (2) and (3) for notational convenience.

At the final step of the PRLS algorithm the response \mathbf{y} is regressed on the whole set of the extracted components \mathbf{T}_k using LTS instead of ordinary least squares which is the case for PLS1. We denote $\hat{\mathbf{y}}_k^{LTS}$ for the LTS fitted values for a model containing k components.

At the first iteration the covariance ($\text{cov}(\mathbf{x}_j, \mathbf{y}) = E(\mathbf{x}_j \mathbf{y}) - E(\mathbf{x}_j)E(\mathbf{y})$) is computed by leaving out of the $E(\cdot)$ a proportion δ of the initial values. This trimming constant δ is a number less than 0.5 which gives the proportion trimmed in the calculations for $\text{cov}(\mathbf{x}_j, \mathbf{y})$. It should be a number larger than the suspected fraction of outliers. A value of $\delta = 0.15$ is generally accepted when no prior information is available. When prior information exists it should be taken into account for constructing δ . One can use cross validation to choose the value of δ .

Partial Reweighted Least Squares Regression (2)

For $i = 1, \dots, n$ and $j = 1, \dots, p$, $\mathbf{X}_{(n,p)}$ and $\mathbf{y}_{(n,1)}$

1. Center or standardise both \mathbf{X} and \mathbf{y} .

2. For $k = 1, \dots, p$

Calculate vector \mathbf{w}_k according to (2) and (3) and scale it to 1.

Derive component $\mathbf{t}_k = \sum_{j=i}^p w_{j,k} \mathbf{x}_{j,k-1}$ where $\mathbf{w}_k = (w_{1,k}, \dots, w_{j,k})$.

Regress each $\mathbf{x}_{j,k-1}$ and \mathbf{y}_{k-1} on \mathbf{t}_k using LTS and store $h_{i,kj}^{\mathbf{x}}$ and $h_{i,k}^{\mathbf{y}}$ as in (1).

Orthogonalise each $\mathbf{x}_{j,k-1}$ with respect to \mathbf{t}_k : $\mathbf{x}_{j,k} = \mathbf{x}_{j,k-1} - E(\mathbf{x}_{j,k-1} | \mathbf{t}_k)$.

3. Give the resulting sequence of the fitted vectors $\hat{\mathbf{y}}_k^{LTS} = \mathbf{T}_k \hat{\mathbf{q}}_k^{LTS}$, and finally recover the implied regression coefficients (regression coefficients on \mathbf{x}_j) according to $\hat{\boldsymbol{\beta}}_k = \mathbf{W}_k \hat{\mathbf{q}}_k^{LTS}$, where $\mathbf{T}_k = (\mathbf{t}_1, \dots, \mathbf{t}_k)$, $\hat{\mathbf{q}}_k^{LTS} = (\hat{q}_1^{LTS}, \dots, \hat{q}_k^{LTS})$, and $\mathbf{W}_k = (\mathbf{w}_1, \dots, \mathbf{w}_k)$.

Model Selection

In order to select the final model (number of components to be retained in the final model) we use Cross Validation. The data set (denoted \mathcal{S}) is randomly split into a training set \mathcal{S}_{train} (model construction) and a test set \mathcal{S}_{test} (model validation), where $\mathcal{S}_{train} \cap \mathcal{S}_{test} = \emptyset$ and $\mathcal{S}_{train} \cup \mathcal{S}_{test} = \mathcal{S}$. For each model built on the training set, its prediction error is computed on the test set. The model with the minimum prediction error is finally selected.

The prediction error is computed according to the Root Mean Squared Error (RMSE), which for a model containing k components is given by

$$RMSE_k = \sqrt{\frac{1}{n_{test}} \sum_{i=1}^{n_{test}} (y_i - \hat{y}_{ik})^2}.$$

Goodness-of-fit

In order to assess the goodness-of-fit of the fitted models based on the full data set \mathcal{S} , we measure the variability of the error term as a proportion of the total variability of the response \mathbf{y} .

We use two measures of variability, based on common variance and the mad as follows

$$GOF_{\text{var}} = \frac{\text{var}(\mathbf{e})}{\text{var}(\mathbf{y})}, \quad \text{and} \quad GOF_{\text{mad}} = \frac{\text{mad}^2(\mathbf{e})}{\text{mad}^2(\mathbf{y})}.$$

Experience with real data - 1

Octane Data: A commonly used example arising from NIR experiment. Analysed in:

Tenenhaus, M. (1998) *La régression PLS*

Engelen, S. et al. (2004) *Robust PCR and Robust PLSR: a comparative study.*

Data set really **fat**. 39 observations, 225 predictors. Response of interest: octane concentration

A regression model for the response was built using PLS1, PRLS and PLAD regression methods.

The final regression models were chosen using CV. Initial value δ for PRLS was set to 0.15.

Model Selection Results

Octane The octane data were randomly split using as training sample the rows of the octane data set resulting from the R command `r.s. <- sample(index, nrow(a)*0.666, replace = FALSE)`. Two thirds of the data set are used for training and one third for testing.

Table 1: Model selection choice for the octane data set. Summary Table

Loss	Algorithm	k
$RMSE_k$	PLS1	3
	PLAD	2
	PRLS	2

Goodness-of-fit Results

Table 2: Goodness-of-fit measures for the octane data set.

	Algorithm	$k = 2$	$k = 3$
GOF _{var}	PLS1	0.128	0.017
	PLAD	0.028	0.028
	PRLS	0.106	0.037
GOF _{mad}	PLS1	0.039	0.005
	PLAD	0.010	0.009
	PRLS	0.028	0.006

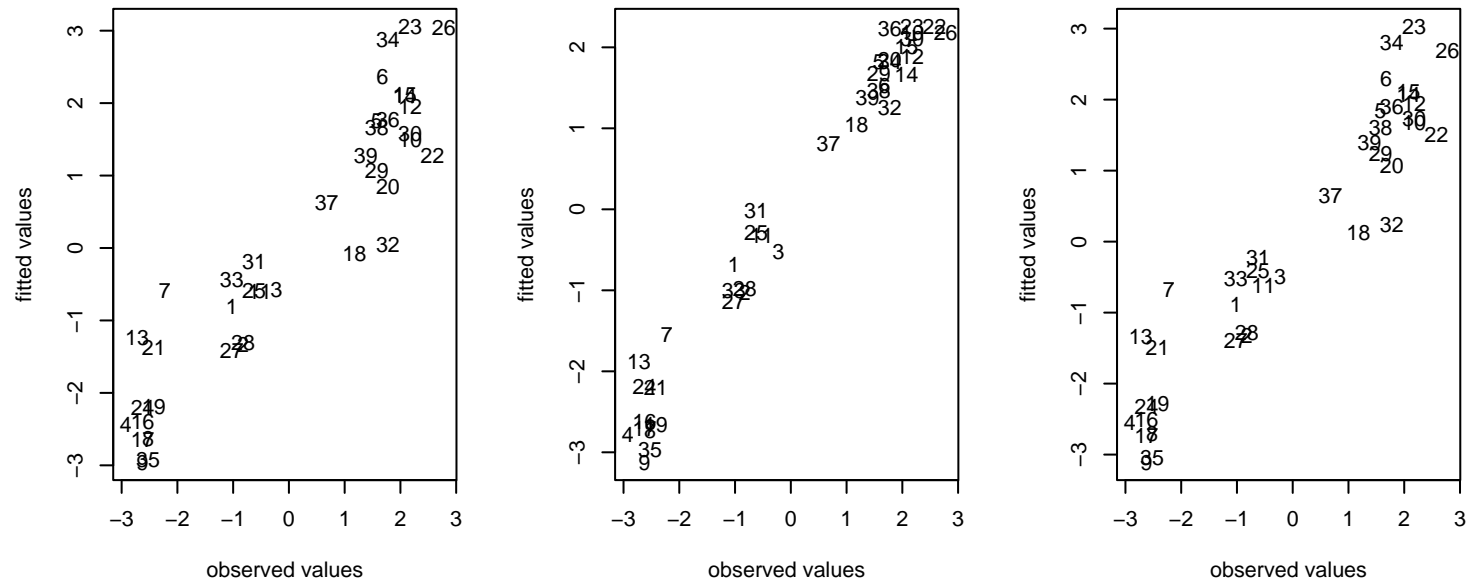


Figure 2: Fitted versus observed values for PLS1, PLAD, and PRLS models on two derived components (order from left to right).

Experience with real data - 2

Biscuits Data: Yet, another example coming from NIR experiments.

Data set contain 40 observations, 600 predictors. The data set is commonly used for PLS2 analysis since four different constituents (fat, sugar, flour, water) are recorded for the digitized spectra. We analyse the concentrations for sugar. Response of interest: sugar concentration

A regression model for the response was built using PLS1, PRLS and PLAD regression methods. The final regression models were chosen using CV. Initial value δ for PRLS was set to 0.15.

Model Selection Results

Biscuit The biscuit data were randomly split using the same procedure as with the octane data set, with training sample size equal to 30.

Table 3: Model selection choice for the biscuits models.

Loss	Algorithm	k=1	k=2	k=3	k=4
$RMSE_k$	PLS1	0.8109	0.3832	0.5447	0.5517
	PLAD	0.6851	0.3433	0.5885	0.7508
	PRLS	0.5123	0.2768	1.3066	1.7107

Goodness-of-fit Results

Table 4: Goodness-of-fit measures for the biscuits models.

	Algorithm	$k = 2$	$k = 3$
GOF _{var}	PLS1	0.151	0.135
	PLAD	0.330	0.241
	PRLS	0.210	0.177
GOF _{mad}	PLS1	0.056	0.047
	PLAD	0.211	0.124
	PRLS	0.042	0.030

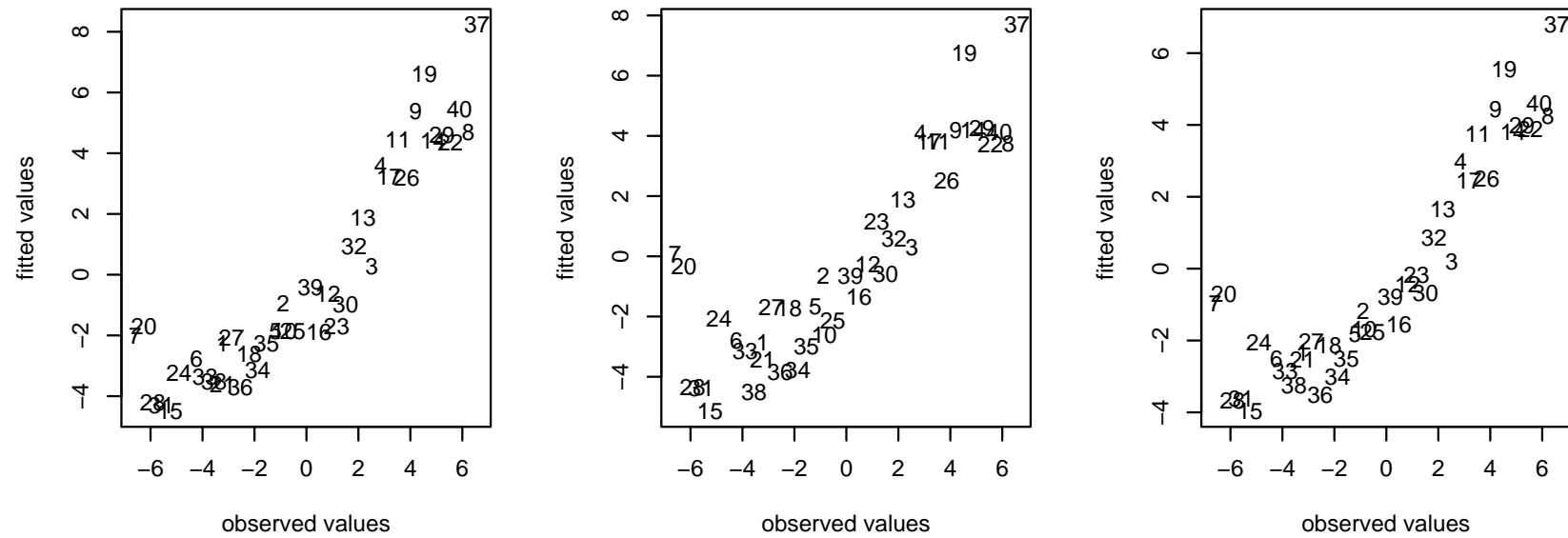


Figure 3: Fitted versus observed values for PLS1, PLAD, and PRLS models on two derived components (order from left to right).

Some conclusions

- Both PLAD and PRLS algorithms retain important features of PLS1.
 1. Orthogonal components
 2. Dimension reduction of the regression problem
- PLAD and PRLS regression methods seek to protect results from outliers.
- PLAD algorithm is simpler.
- PRLS is computationally more expensive.
- PRLS regression method is a generalisation of PLS1 method. PLS1 corresponds to PRLS for $\delta = 0$.
- PLAD may be influenced from outliers on the predictors space. Yet, in NIR experiments this not so clear.
- PRLS depends on the choice of δ .

- PLAD and PRLS regression methods have showed that they may select less derived components in the final regression model compared to PLS1 method. Hence, they may futher reduce the dimension of the regression problem especially when outliers are present.
- Both PLAD and PRLS provide an alternative solution in regression modeling when $n \ll p$.