A New Robust Partial Least Squares Regression Method

Javier González, Daniel Peña and Rosario Romera

8th of September, 2005

UNIVERSIDAD CARLOS III DE MADRID
DEPARTAMENTO DE ESTADISTICA

Why robust methods?

"... just which *robust methods* you use is not important, what is important is that you use *some*." **J. W. Tukey (1979)**

Fundamental Continuity Concept

- Small changes in the data result in only small changes in estimate.
- Change a few, so what? **J.W. Tukey (1977).**

Outliers

Outliers are atypical observations that are "well" separated from the bulk of the data. Covariance matrix and means vector are very sensitive to outliers in data.

- 1-D (relatively *easy* to detect).
- 2-D (*harder* to detect).
- Higher-D (*very hard* to detect).

Literature review

- **Wakeling and Macfie (1992)**. First PLSR robustification. Algorithm with weighted regressions for PLS1 and PLS2.

Literature review

- **Wakeling and Macfie (1992)**. First PLSR robustification. Algorithm with weighted regressions for PLS1 and PLS2.
- **Griep et al. (1995)**. Comparative study of LSM, RM, and IRLS. (Algorithms not resistant to leverage points).

Literature review

- **Wakeling and Macfie (1992)**. First PLSR robustification. Algorithm with weighted regressions for PLS1 and PLS2.
- **Griep et al. (1995)**. Comparative study of LSM, RM, and IRLS. (Algorithms not resistant to leverage points).
- **Gil and Romera (1998)**. Robustification of the cross variance matrix through the SD estimator.

Literature review

- **Wakeling and Macfie (1992)**. First PLSR robustification. Algorithm with weighted regressions for PLS1 and PLS2.
- **Griep et al. (1995)**. Comparative study of LSM, RM, and IRLS. (Algorithms not resistant to leverage points).
- **Gil and Romera (1998)**. Robustification of the cross variance matrix through the SD estimator.
- **Hubert and Vandem Brandem (2003)**. PLS Robustification based on the SIMPLS algorithm.

Data and General Considerations

- We analyze the case q=1.

Data and General Considerations

- We analyze the case q=1.
- Population loading vector computed as in **Helland (1988)**.

Data and General Considerations

- We analyze the case q=1.
- Population loading vector computed as in **Helland (1988)**.
- Optimal number of PLS component through *leave-one-out cross* validation.

Data and General Considerations

- We analyze the case q=1.
- Population loading vector computed as in **Helland (1988)**.
- Optimal number of PLS component through *leave-one-out cross* validation.
- One-step robustification comes from using the robust covariance matrix.

PLS Algorithm

$$\sum_{[y,x]} = \begin{pmatrix} \sigma_y^2 & \delta_{y,x}^T \\ \delta_{y,x} & \sum_x \end{pmatrix}$$

The population loading vectors given by **Helland (1988)**:

- $w_1 \propto \delta_{y,x}$
- $w_{a+1} \propto \delta_{y,x} - \sum_x W_a (W_a^T \sum_x W_a)^{-1} W_a^T \delta_{y,x}$

Where W_a, $1 < a \leq A$ are the loading vectors W_i and A the selected numbers of PLS components.

$$W_a = [w_1, w_2, ..., w_a]$$

PLS Algorithm

- Then the *population regression vector* (non robust) is given by:

$$\beta_a = W_a (W_a^T \sum_x W_a)^{-1} W_a^T \delta_{y,x}$$

PLS Algorithm

- Then the *population regression vector* (non robust) is given by:

$$\beta_a = W_a(W_a^T \sum_x W_a)^{-1} W_a^T \delta_{y,x}$$

- The proposed global Robust algorithm come from using the covariance matrix $\tilde{S}_{[y,x]}$ obtained from the original data being in this case w_1 a normalization of $\tilde{\delta}_{y,x}$ and:

$$w_{a+1} \propto \tilde{\delta}_{y,x} - \tilde{S}_x W_a (W_a^T \tilde{S}_x W_a)^{-1} W_a^T \tilde{\delta}_{y,x}$$

Outliers detection and computation of $\tilde{S}$

The algorithm **Peña and Prieto(2005)** works in three steps after data are *scaled* and *centered*:

$$\breve{x}_i = S^{-1/2}(x_i - \bar{x}), \quad i = 1, \ldots, n.$$

- STEP 1: finding directions maximizing and minimizing the kurtosis, projecting the data and identifying outliers.

Outliers detection and computation of $\tilde{S}$

The algorithm **Peña and Prieto(2005)** works in three steps after data are *scaled* and *centered*:

$$\breve{x}_i = S^{-1/2}(x_i - \bar{x}), \quad i = 1, \ldots, n.$$

- STEP 1: finding directions maximizing and minimizing the kurtosis, projecting the data and identifying outliers.
- STEP 2: generating random directions and stratifying the sample and identifying outliers again.

Outliers detection and computation of $\tilde{S}$

The algorithm **Peña and Prieto(2005)** works in three steps after data are *scaled* and *centered*:

$$\breve{x}_i = S^{-1/2}(x_i - \bar{x}), \quad i = 1, \ldots, n.$$

- STEP 1: finding directions maximizing and minimizing the kurtosis, projecting the data and identifying outliers.
- STEP 2: generating random directions and stratifying the sample and identifying outliers again.
- STEP 3: check the suspicious observations by using the Mahalanobis distance and repit until no more outliers are found.

Step I: Searching kurtosis directions

The direction that maximizes (minimizes) the coefficient of kurtosis is obtained as the solution of the *optimization problem*:

$$\begin{aligned} d_j = \quad & \arg\max(min)_d \quad \frac{1}{n}\sum_{i=1}^{n}\left(d'\check{x}_i^{(j)}\right)^4 \\ \text{s.t.} \quad & \quad d'd = 1. \end{aligned}$$

The sample points are projected onto a lower dimension subspace, orthogonal to the directions d_j

Step I: Searching kurtosis directions

Why kurtosis directions?

Because is possible to study the presence of outliers on the kurtosis values and to use this moment coefficient to identify them.

- Symmetric and and a small proportion of outliers generated with asymmetric contamination increase the coefficient on the observed data.

Step I: Searching kurtosis directions

Why kurtosis directions?

Because is possible to study the presence of outliers on the kurtosis values and to use this moment coefficient to identify them.

- Symmetric and and a small proportion of outliers generated with asymmetric contamination increase the coefficient on the observed data.
- A large proportion of outliers generated by asymmetric contamination can make the kurtosis coefficient very small.

Step II: Searching random directions

- Chosen two observations *randomly* from the sample and compute the direction $\hat{d}_l$ defined by these two observations.

Step II: Searching random directions

- Chosen two observations *randomly* from the sample and compute the direction $\hat{d}_l$ defined by these two observations.
- The observations are then projected onto this direction, to obtain the values $\hat{z}_i^l = \hat{d}_l^T \breve{x}_i$

Step II: Searching random directions

- Chosen two observations *randomly* from the sample and compute the direction $\hat{d}_l$ defined by these two observations.
- The observations are then projected onto this direction, to obtain the values $\hat{z}_i^l = \hat{d}_l^T \check{x}_i$
- Then the sample is partitioned into K groups of size n/K, where K is a prespecified number, based on the ordered values of the projections $\hat{z}_i^l$, so that group k, $1 \leq k \leq K$, contains those observations i satisfying.

$$\hat{z}^l_{(\lfloor (k-1)n/K \rfloor + 1)} \leq \hat{z}_i^l \leq \hat{z}^l_{(\lfloor kn/K \rfloor)}$$

Step II: Searching random directions

- From each group k, $1 \leq k \leq K$, a subsample of p observations is chosen without replacement, the orthogonal direction is computed and the corresponding projections.

Why random directions?

- Because is necessary a procedure that detect outliers when the proportion of *contamination is between .2 and .3* and the contamination distribution has the *same variance* as the original distribution. (case when kurtosis fails)

Step III: Deleting outliers and computing $\tilde{S}$

A *Mahalanobis* distance is computed for all observations labeled as outliers in the preceding steps. Being U the set of all observations not labeled as outliers:

$$\tilde{m} = \frac{1}{|U|} \sum_{i \in U} x_i$$
$$\tilde{S} = \frac{1}{|U|-1} \sum_{i \in U} (x_i - \tilde{m})(x_i - \tilde{m})'$$
$$v_i = (x_i - \tilde{m})^T \tilde{S}^{-1} (x_i - \tilde{m})$$

Those observations $i \in U$ such that $v_i < \xi^2_{p-1,0.99}$ are considered not to be outliers and are included in U.

Simulation Study

- Comparative of methods *PLS*(Helland, 1988), *PLS-SD* (Gil and Romera, 1998),*RSIMPLS* (Hubert and Branden 2003) and *PLSKurSD* (González, Peña and Romera).

Simulation Study

- Comparative of methods *PLS*(Helland, 1988), *PLS-SD* (Gil and Romera, 1998),*RSIMPLS* (Hubert and Branden 2003) and *PLSKurSD* (González, Peña and Romera).
- Four types of outliers were generated: bad leverage points, vertical outliers, orthogonal outliers and very concentrated outliers.

Simulation Study

- Comparative of methods *PLS*(Helland, 1988), *PLS-SD* (Gil and Romera, 1998),*RSIMPLS* (Hubert and Branden 2003) and *PLSKurSD* (González, Peña and Romera).
- Four types of outliers were generated: bad leverage points, vertical outliers, orthogonal outliers and very concentrated outliers.
- Three measures for comparing the methods.

Simulation Study

- Comparative of methods *PLS*(Helland, 1988), *PLS-SD* (Gil and Romera, 1998),*RSIMPLS* (Hubert and Branden 2003) and *PLSKurSD* (González, Peña and Romera).
- Four types of outliers were generated: bad leverage points, vertical outliers, orthogonal outliers and very concentrated outliers.
- Three measures for comparing the methods.
- Simulations have been done in a personal computer Pentium III 650 MH with 128 Mb of internal memory. Code implemented in Matlab.

Simulation Study. Bilinear model

$$T \sim N_A(0_A, \Sigma_t) \text{ with } A < p$$
$$X = TI_{A,p} + N_p(0_p, 0.1I_p)$$
$$Y = TQ + N_q(0_q, I_q)$$

$(I_{A,p})_{i,j} = 1$ for $i = j$ and $(I_{A,p})_{i,j} \neq 1$. Q is a matrix of dimensions $A \times p$ with $(A_{i,j}) = 1 \; \forall \; i, j$ The simulation is done with a known values of $A = A_{opt}$ and generating randomly a number n_ϵ of outliers.

Table: Simulation study

q	n	p	A	σ_t	σ_t	Contamination
1	100	5	2	diag(4,2)	1	10% and 30% Out.

Simulation Study. Outliers

- *Bad leverage* regression points:
  $T_\epsilon \sim N_A(10_A, \sigma_t)\ X_\epsilon = T_\epsilon I_{A,p} + N_p(0_p, 0.1 I_p)$

Simulation Study. Outliers

- *Bad leverage* regression points:

$$T_\epsilon \sim N_A(10_A, \sigma_t) \; X_\epsilon = T_\epsilon I_{A,p} + N_p(0_p, 0.1 I_p)$$

- *Vertical* outliers:

$$Y_\epsilon = TQ_{A,q} + N_q(10_q, 0.1 I_q)$$

Simulation Study. Outliers

- *Bad leverage* regression points:

$$T_\epsilon \sim N_A(10_A, \sigma_t) \; X_\epsilon = T_\epsilon I_{A,p} + N_p(0_p, 0.1 I_p)$$

- *Vertical* outliers:

$$Y_\epsilon = TQ_{A,q} + N_q(10_q, 0.1 I_q)$$

- *Orthogonal* outliers:

$$X_\epsilon = T_\epsilon I_{A,p} + N_p((0_A, 10_{p-A}), 0.1 I_p)$$

Simulation Study. Outliers

- *Bad leverage* regression points:
  $$T_\epsilon \sim N_A(10_A, \sigma_t) \; X_\epsilon = T_\epsilon I_{A,p} + N_p(0_p, 0.1 I_p)$$
- *Vertical* outliers:
  $$Y_\epsilon = TQ_{A,q} + N_q(10_q, 0.1 I_q)$$
- *Orthogonal* outliers:
  $$X_\epsilon = T_\epsilon I_{A,p} + N_p((0_A, 10_{p-A}), 0.1 I_p)$$
- Very *concentrated* outliers:
  $$X_\epsilon = T_\epsilon I_{A,p} + N_p(10_p, 0.001 I_p)$$

MC Simulation Study. Measures

- The experimental *slope* of each method $\beta^{(l)}$

$$ang\beta_{1,A} = ang(\beta, \hat{\beta}_{[y^c, X^c], A})$$

MC Simulation Study. Measures

- The experimental *slope* of each method $\beta^{(l)}$

$$ang\beta_{1,A} = ang(\beta, \hat{\beta}_{[y^c, X^c],A})$$

- The *mean squared error* of the norms.

$$MSE_A(\hat{\beta}) = \frac{1}{m}\sum_{l=i}^{m} \|\hat{\beta}_A^{(l)} - \beta\|$$

MC Simulation Study. Measures

- The experimental *slope* of each method $\beta^{(l)}$
$$ang\beta_{1,A} = ang(\beta, \hat{\beta}_{[y^c, X^c], A})$$
- The *mean squared error* of the norms.
$$MSE_A(\hat{\beta}) = \frac{1}{m}\sum_{l=i}^{m} \|\hat{\beta}_A^{(l)} - \beta\|$$
- A *test set* of n_t observations with the original model and we compute:
$$RMSE_A = \sqrt{\frac{1}{n_t}\sum_{i=1}^{n}(y_i - \hat{y}_{i,A})^2}$$
Being $\hat{y}_{i,k}$ the predicted value of y in the observation i.

MC Simulation Study. 10% contamination

Algorithm	PLS	PLS-SD	PLS-KurSD	RSIMPLS
No Contamination				
Mean(Angle)	0.06(0.03)	0.07(0.03)	0.07(0.03)	0.08(0.03)
Norm(β)	0.01(0.01)	0.01(0.01)	0.01(0.01)	0.01(0.01)
MSE(σ_e)	0.16(0.08)	0.17(0.09)	0.17(0.09)	0.17(0.09)
10% Bad leverage points				
Mean(Angle)	1.13(0.22)	0.11(0.06)	0.07(0.03)	0.08(0.03)
Norm(β)	1.23(0.15)	0.07(0.04)	0.01(0.01)	0.02(0.01)
MSE(σ_e)	2.07(0.23)	0.48(0.16)	0.18(0.10)	0.18(0.09)
10% Vertical outliers				
Mean(Angle)	1.14(0.21)	0.11(0.06)	0.07(0.03)	0.08(0.03)
Norm(β)	1.23(0.14)	0.07(0.05)	0.02(0.01)	0.02(0.01)
MSE(σ_e)	2.08(0.24)	0.47(0.17)	0.18(0.10)	0.18(0.10)
10% Orthogonal outliers				
Mean(Angle)	1.13(0.21)	0.11(0.06)	0.07(0.04)	0.08(0.03)
Norm(β)	1.22(0.15)	0.07(0.04)	0.02(0.01)	0.02(0.01)
MSE(σ_e)	2.06(0.22)	0.48(0.16)	0.18(0.10)	0.18(0.10)
10% Concentrated outliers				
Mean(Angle)	1.14(0.21)	0.11(0.06)	0.08(0.04)	0.08(0.04)
Norm(β)	1.23(0.14)	0.08(0.04)	0.02(0.06)	0.02(0.02)
MSE(σ_e)	2.08(0.23)	0.48(0.16)	0.19(0.10)	0.19(0.09)

MC Simulation Study. 30% contamination

Algorithm	PLS	PLS-SD	PLS-KurSD	RSIMPLS
No Contamination				
Mean(Angle)	0.06(0.03)	0.07(0.03)	0.07(0.03)	0.08(0.03)
Norm(β)	0.01(0.01)	0.01(0.01)	0.01(0.01)	0.02(0.01)
MSE(σ_e)	0.16(0.08)	0.18(0.09)	0.18(0.09)	0.18(0.09)
30% Bad leverage points				
Mean(Angle)	1.36(0.18)	0.61(0.21)	0.10(0.10)	1.29(0.26)
Norm(β)	1.39(0.13)	0.75(0.20)	0.04(0.11)	1.37(0.22)
MSE(σ_e)	2.23(0.24)	1.58(0.22)	0.24(0.22)	2.19(0.25)
30% Vertical outliers				
Mean(Angle)	1.36(0.19)	0.62(0.21)	0.11(0.12)	1.30(0.27)
Norm(β)	1.40(0.14)	0.75(0.19)	0.04(0.13)	1.37(0.19)
MSE(σ_e)	2.25(0.24)	1.58(0.22)	0.26(0.27)	2.20(0.26)
30% Orthogonal outliers				
Mean(Angle)	1.36(0.17)	0.61(0.21)	0.10(0.11)	1.31(0.25)
Norm(β)	1.40(0.16)	0.75(0.19)	0.04(0.13)	1.37(0.17)
MSE(σ_e)	2.26(0.24)	1.59(0.22)	0.25(0.23)	2.22(0.26)
30% Concentrated outliers				
Mean(Angle)	1.36(0.18)	0.61(0.20)	0.10(0.10)	1.29(0.26)
orm(β)	1.39(0.21)	0.74(0.20)	0.04(0.11)	1.37(0.20)
MSE(σ_e)	2.26(0.23)	1.59(0.21)	0.24(0.23)	2.21(0.24)

RENFE Data

- 17 independing variables that present some measures of the RENFE(Public Railroad system in Spain) service.
  - Station security
  - Train cleanness
  - Noise level...

RENFE Data

- 17 independing variables that present some measures of the RENFE(Public Railroad system in Spain) service.
  - Station security
  - Train cleanness
  - Noise level...
- One dependent variable that corresponds with a measure of the global satisfaction of the customers with the service quality.

RENFE Data

- 17 independing variables that present some measures of the RENFE(Public Railroad system in Spain) service.
  - Station security
  - Train cleanness
  - Noise level...
- One dependent variable that corresponds with a measure of the global satisfaction of the customers with the service quality.
- All variables were requested to evaluate on a 0-9 scale.

- The sample include 1499 questionnaires and are available in `http://halweb.uc3m.es"`.

Computational times

Algorithm	PLS	PLS-KurSD	RSIMPLS
Time(seg.)	0.0200	3.5952	777.9286

- The sample include 1499 questionnaires and are available in `http://halweb.uc3m.es"`.
- The first principal component explains the 93.4% of the variability.

Computational times

Algorithm	PLS	PLS-KurSD	RSIMPLS
Time(seg.)	0.0200	3.5952	777.9286

Conclusions

- PLSKurSD behaves well with any type of contamination and is easy to compute.

Conclusions

- PLSKurSD behaves well with any type of contamination and is easy to compute.
- PLSKurSD is resistant to outliers even with a big percent of contamination.

Conclusions

- PLSKurSD behaves well with any type of contamination and is easy to compute.
- PLSKurSD is resistant to outliers even with a big percent of contamination.
- PLSKurSD is very fast an is useful in large data sets.

Future work

- Extend the work to the case of several dependent variables.

Future work

- Extend the work to the case of several dependent variables.
- To develop a Robust PLS with $p \gg n$.

Future work

- Extend the work to the case of several dependent variables.
- To develop a Robust PLS with $p \gg n$.
- To analyze in which cases a robustification of the regression is necessary.

4TH SIMPOSIUM OF PLS AND RELATED METHODS

BARCELONA 7TH-9TH OF SEPTEMBER, 2005