Leave-n-Samples-Out Cross-validation in PCA for Missing Data Recovery and Robustness in front of Measurement Noise

J. Camacho[1] J. Picó[1] A. Ferrer[2]
[1] Departamento de Ingeniería de Sistemas y Automática, Universidad Politécnica de Valencia,
Camino de Vera s/n, 46022, Valencia (Spain), {jcamacho@isa.upv.es,jpico@ai2.upv.es}
[2] Departamento de Estadística e Investigación Operativa Aplicadas y Calidad, Universidad Politécnica de Valencia,
Camino de Vera s/n, 46022, Valencia (Spain), aferrer@eio.upv.es

Keywords: Principal Component Analysis, Cross-validation, Missing Data, Measurement Noise.

1 Introduction

Principal Components Analysis (PCA) is one of the most widely studied multivariate analysis methods. PCA is essentially a dimension reduction technique [1] where the objective is to find the subspace in the space of the variables where data mostly varies. From a general point of view, taking into account the preprocessing -centering and scaling- of the data, PCA can be thought as a model following:

$$(\mathbf{X} - \mathbf{1}_N \cdot \boldsymbol{\mu}^t) \otimes (\mathbf{1}_N \cdot \boldsymbol{\varsigma}^t) = \mathbf{T}_a \cdot \mathbf{P}_a^t + \mathbf{E}_a \tag{1}$$

where $\mathbf{X}$ is a $N \times M$ matrix of data with N observations or objects of M variables, $\mathbf{1}_N$ is a N-vector filled with ones, $\boldsymbol{\mu}$ is the M-vector containing the averages of the variables in $\mathbf{X}$ and $\boldsymbol{\varsigma}$ is the M-vector containing the weights, $\otimes$ stands for the Hadamard (element to element) product, $\mathbf{T}_a$ is the $N \times a$ matrix with the scores of the objects in the a PCs, $\mathbf{P}_a$ is the $M \times a$ matrix with the loads of the variables in the PCs and $\mathbf{E}_a$ is the $N \times M$ matrix of residuals. The left-hand side of (1) performs the preprocessing of the data and the right-hand side performs the PCA modelling.

Much work has been devoted to find an *'optimum'* value for a in (1) - i.e. the number of PCs in the PCA model. A good survey can be found in the book by Jackson [2]. Wold [3] proposed the use of cross-validation for that purpose. In cross-validation, data are divided in G groups. Each time, a model is calibrated from the whole data set but a group. Afterwards, the data from that group are predicted using the model and a criterium of goodness of fit (CGF) is computed. This is repeated for each of the G groups and a total CGF for a model is obtained. In PCA, the CGF is computed for the models with 1 PC, 2 PCs, 3 PCs, a so on. From the shape of the CGF, the optimum number of PCs is estimated. Eastment and Krzanowski [4] and Nomikos and MacGregor [5] suggest the use of cross-validation when the PCA model is going to be used for future observations. This is because cross-validation allows the estimation of the prediction error expected for incoming data, as long as this prediction error is conveniently computed. Although several cross-validatory approaches have been proposed for PCA models, those by Wold [3] and Eastment and Krzanowski [4] are the most cited and influent ones. Wold also suggested a possible alternative for those for which the NIPALS procedure is not available -something very difficult nowadays. Wold did not pay very much attention to this alternative. Nonetheless, it presents an attractive feature: the minimum of the CGF computed, which is the sum of squares of prediction error (PRESS), is supposed to signal the optimum number of PCs. This is, in principle, a logical behavior for the prediction error: decrease as the addition of PCs improves the prediction performance of the model and increase when this addition is noisy. Let us call this alternative the Leave-n-samples-out (LnSO) method. This method is assessed in this paper for the determination of the appropriate number of PCs for PCA models with two different purposes: missing data recovery and robustness in front of measurement noise.

2 Material and methods

For the sake of understanding, the typical nomenclature of missing data literature [6] has been inherited here with a slightly different meaning. A superscript asterisk $(*)$ is used to specify the data used to fit the model in each cross-validation iteration. A superscript pound $(\#)$ is used to specify the data not used in the model fitting and which are currently being predicted in each cross-validation iteration. The LnSO algorithm follows:

For each PC ($a = 1...A$)
    For each group of objects ($g = 1...G$)
        Form $\mathbf{X}^*$ with data from all groups but g
        Form $\mathbf{X}^\#$ with data from g
        Calibrate a PCA model from $\mathbf{X}^*$, obtaining $\mathbf{P}_a^*$ and $\mathbf{T}_a^*$
        For each group of variables ($h = 1...H$)
            Set $\mathbf{X}_h^\# = 0$
            $\mathbf{T}_a^\# = \mathbf{X}^\# \cdot \mathbf{P}_a^*$
            $\hat{\mathbf{X}}_a^\# = \mathbf{T}_a^\# \cdot \mathbf{P}_a^{*t}$
            Restore its actual value to $\mathbf{X}_h^\#$
            $\mathbf{E}_{a,g,h} = \mathbf{X}_h^\# - \hat{\mathbf{X}}_{a,h}^\#$
        end
    end
    $PRESS_a = \sum_{n=1}^{N} \sum_{m=1}^{M} e_{a,n,m}^2$
end

where $e_{a,n,m}$ stands for the single elements of $\mathbf{E}_a$ in (1), containing the prediction errors. One controversial point in a cross-validation algorithm is to decide whether the preprocessing information, i.e. the average and weight of the variables, should be estimated either from $\mathbf{X}$ or else from $\mathbf{X}^*$, and then applied to $\mathbf{X}^\#$. A discussion on this matter can be found in several papers [3, 7]. Here, under the assumption that the model will be applied to future observations, the second option is preferred. Thus, strictly speaking, the models are calibrated following:

$$\mathbf{X}^* = \mathbf{1}_N \cdot \boldsymbol{\mu}^{*T} + (\mathbf{T}_a^* \cdot \mathbf{P}_a^{*T}) \oslash (\mathbf{1}_N \cdot \boldsymbol{\varsigma}^{*T}) + \mathbf{E}_{a,\boldsymbol{\varsigma}}^* \tag{2}$$

where $\boldsymbol{\mu}^*$ is the M-vector containing the averages of the variables and $\boldsymbol{\varsigma}^*$ is the M-vector containing the weights applied to the variables in $\mathbf{X}^*$, and $\oslash$ is the Hadamard (element to element) division.

The LnSO algorithm is grounded on two ideas: a) if the model is going to be used for future observations, in each iteration of the cross-validation the PCA model should not be fitted from the data of the object which is going to be predicted; and b) since the PCA model establishes relationship structures among the variables, its prediction power should be measured by predicting the value of a variable from the rest taking into account these structures -i.e., the PCA model. The core of the LnSO algorithm is composed by the equations where the prediction is performed, i.e.:

$\mathbf{T}_a^\# = \mathbf{X}^\# \cdot \mathbf{P}_a^*$
$\hat{\mathbf{X}}_a^\# = \mathbf{T}_a^\# \cdot \mathbf{P}_a^{*t}$

If one may use the LnSO algorithm for the estimation of the prediction error in missing data, a correction to the core is in due so that the estimation of $\mathbf{T}_a^\#$ is repeated till convergence. Let us call the algorithm that performs this correction the Leave-n-samples-out 2 (LnSO2) algorithm. An alternative to the correction performed in LnSO2 is the use of an imputation method for the estimation of $\mathbf{T}_a^\#$, such as Trimmed Score Regression (TSR) [8] and Projection to Model Plane (PMP) [5, 6].

On the other hand, the LnSO procedure has a principal limitation: What about the independent variables? Many times, mostly when the number of variables collected is reduced, data may present some variables which behave completely independently from the rest. This is not a problem for PCA modelling since it is able to capture that sort of variables. Nonetheless, it is a problem for the LnSO method, since the value of that independent variables cannot be predicted from the rest. In order to correct this limitation, the approach of this paper is to augment X with redundant information. Let us call the algorithm performing this correction the Cross-validation Corrected LnSO (CCLnSO):

For each PC ($a = 1...A$)
  For each group of objects ($g = 1...G$)
    Form $\mathbf{X}^*$ with data from all groups but g
    Form $\mathbf{X}^\#$ with data from g
    Calibrate a PCA model from $\mathbf{X}^*$, obtaining $\mathbf{P}_a^*$ and $\mathbf{T}_a^*$
    $\mathbf{T}_a^\# = \mathbf{X}^\# \cdot \mathbf{P}_a^*$
    $\mathbf{X}_{aug,a}^* = [\mathbf{X}^*, \mathbf{T}_a^*]$, remember not to scale $\mathbf{T}_a^*$
    Calibrate a PCA model from $\mathbf{X}_{aug,a}^*$, obtaining $\mathbf{P}_{aug,a}^*$ and $\mathbf{T}_{aug,a}^*$
    For each group of variables ($h = 1...H$)
      Set $\mathbf{X}_h^\# = 0$
      $\mathbf{X}_{aug,a}^\# = [\mathbf{X}^\#, \mathbf{T}_a^\#]$
      $\mathbf{T}_{aug,a}^\# = \mathbf{X}_{aug}^\# \cdot \mathbf{P}_{aug,a}^*$
      $\hat{\mathbf{X}}_{aug,a}^\# = \mathbf{T}_{aug,a}^\# \cdot \mathbf{P}_{aug,a}^{*t}$
      Restore its actual value to $\mathbf{X}_h^\#$
      $\mathbf{E}_{a,g,h} = \mathbf{X}_h^\# - \hat{\mathbf{X}}_{a,h}^\#$
    end
  end
  $PRESS_a = \sum_{n=1}^{N} \sum_{m=1}^{M} e_{a,n,m}^2$
end

3 Results and discussion

Three simulated data matrices are used for assessing the corrections proposed. The three have very different nature, with different number of latent and observable variables: 4, 12 and 15 latent variables were generated and from them, a total of 10, 27 and 50 observable variables were obtained, respectively, for the three data sets. In all the cases, the latent variables are generated independently at random following a normal distribution of zero mean and unit variance. Each observable variable is obtained from one latent variable alone or as a linear combination of two or more latent variables. For simplicity, all observable variables are computed so that they have zero mean and unit variance. Notice that the final data sets used in the comparison are composed of observable variables alone. Measurement noise is generated independently for each observable variable and at random, following a normal distribution of zero mean. The data sets are corrupted with noise from 5% to 25%, where this percentage is computed so that the lowest standard deviation of a latent variable is the 100%.

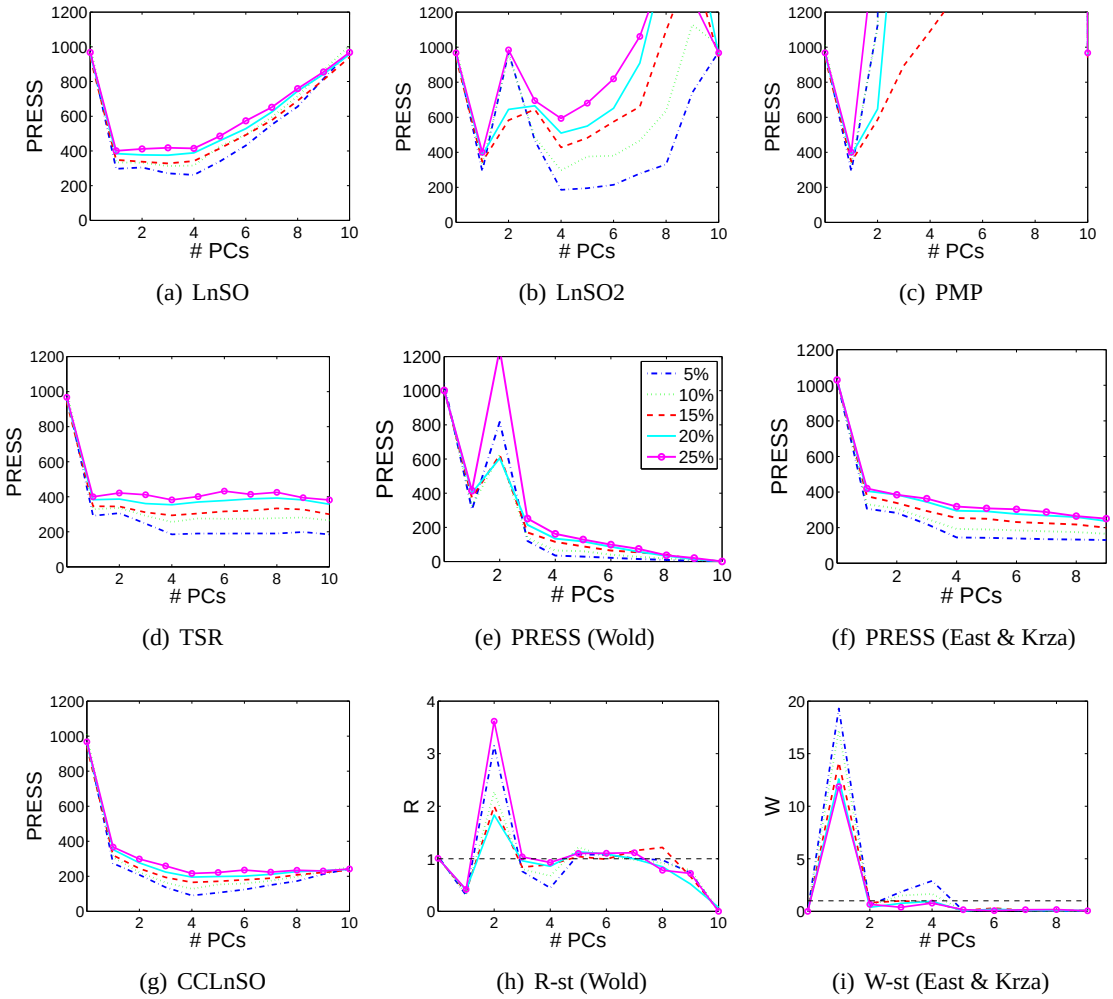
First let us discuss the results obtained for missing data recovery. For this purpose, cross-validation is used to find the number of PCs for which the PCA model yields the minimum value of PRESS. This will be the best choice in terms of missing data recovery. Table 1 contains the minimum value of PRESS for different approaches, namely LnSO, LnSO2, PMP-based LnSO and TSR-based LnSO. Notice the other approaches (CCLnSO, [3], [4]) underestimate the PRESS for missing data recovery in incoming data. That is why they are not used in this first comparison. For the three data sets, LnSO and TSR presented the best outcomes. TSR yields the best performance since it was always between the two best approaches whereas LnSO presents poor

Table 1: Sum of squares of prediction error (PRESS) for missing data recovery.

Noise	First example (4/10)				Second example (12/27)				Third example (15/50)			
	LnSO	LnSO2	PMP	TSR	LnSO	LnSO2	PMP	TSR	LnSO	LnSO2	PMP	TSR
5%	262	186	291	**185**	884	878	878	**873**	814	784	784	**692**
10%	313	297	329	**256**	**1.017**	1.037	1.037	1.030	1.083	1.076	1.076	**981**
15%	328	346	346	**294**	**1.111**	1.147	1.147	1.138	1.351	1.356	1.356	**1.297**
20%	375	383	383	**355**	**1.264**	1.355	1.355	1.310	1.590	1.653	1.653	**1.574**
25%	401	400	400	**382**	**1.314**	1.461	1.461	1.408	**1.756**	1.835	1.835	1.760

Table 2: Number of PCs detected by different approaches. R stands for the R-statistic of [3] and W stands for the W-statistic of [4].

Noise	First example (4/10)				Second example (12/27)				Third example (15/50)			
	R	W	LnSO	CCLnSO	R	W	LnSO	CCLnSO	R	W	LnSO	CCLnSO
5%	1	**4**	**4**	**4**	6	6	**12**	**12**	10	12	13	**15**
10%	1	**4**	3	**4**	6	6	**12**	**12**	10	12	13	**15**
15%	1	1	3	**4**	6	6	**12**	**12**	10	12	13	16
20%	1	1	3	**4**	6	6	**12**	**12**	10	12	13	16
25%	1	1	1	**4**	6	6	**12**	**12**	10	12	13	17



(a) LnSO (b) LnSO2 (c) PMP

(d) TSR (e) PRESS (Wold) (f) PRESS (East & Krza)

(g) CCLnSO (h) R-st (Wold) (i) W-st (East & Krza)

Figure 1: Sum of squares of prediction error (PRESS) and associated statistics for different approaches in the first simulated example (4 latent variables), corrupted with 5% (dashdot line), 10% (dotted line), 15% (dashed line), 20% (solid line) and 25% (solid line with circles) of measurement noise.

outcomes in many cases. In Figure 1, the PRESS curves of the different approaches for the first simulated data set are shown. Among the four approaches in Table 1 (Figures 1(a-d)), TSR presents the best suited curve for missing data recovery. In this curve (Figure 1(d)) the PRESS almost remains constant for a number of PCs larger than the true one (4). This means that if the number of PCs is overestimated, the performance in missing data recovery is not worsened in an important degree when TSR is used. This is not true for the other approaches, where the PRESS tends to increase fast in this situation.

In Table 2 the second problem, i.e. the determination of the number of PCs when data is corrupted with measurement noise, is treated. As it can be seen, the CCLnSO method outperforms the other approaches, which tend to underestimate the true number of PCs. Coming back to Figure 1, it can be seen that the traditional approaches (Figures 1(e) and 1(f)) show PRESS curves which cannot be used directly to determine the number of PCs since they are almost monotonous decreasing. This has to be done from new statistics proposed by the authors -the R-statistic (Figure 1(h)) and the W-statistic (Figure 1(i)). These are irregular statistics which are compared against a heuristic and fixed limit. The result is that these approaches tend to underestimate the number of PCs. The PRESS computed by LnSO (Figure 1(a)) can be used directly for determining the number of PCs but, unfortunately, many times the curves presented the minimum for a lower number of PCs than the actual one. This is the effect of the presence of independent variables in the data. CCLnSO (Figure 1(g)) corrects this problem.

4 Conclusion

In this paper, a number of corrections in the Leave-n-samples-out (LnSO) cross-validation method are proposed. These corrections are useful for the determination of the number of PCs for PCA models with two different purposes: missing data recovery and robustness in front of measurement noise. The corrections were assessed using three simulated examples, yielding very promising results. The LnSO based on the imputation method named Trimmed Score Regression (TSR) yielded the best performance for missing data recovery among those approaches under study. The second correction, named Cross-validation Corrected Leave-n-samples-out (CCLnSO), was suggested for determining the number of PCs in a data set corrupted with measurement noise. This correction was also seen to outperform the other approaches.

References

[1] Smilde A.K., Bro R., Geladi P.. *Multi-way Analysis, Application in the Chemical Sciences.* England: John Wiley & Sons 2003.

[2] Jackson J.E.. *A User's Guide to Principal Components.* England: Wiley-Interscience 2003.

[3] Wold S.. Cross-Validatory Estimation of the Number of Components in Factor and Principal Components *Technometrics.* 1978;20:397–405.

[4] Eastment H.T., Krzanowski W.J.. Cross-Validatory Choice of the Number of Components From a Principal Component Analysis *Technometrics.* 1982;24:73–77.

[5] Nomikos P., MacGregor J.F.. Multivariate SPC Charts for Monitoring Batch Processes *Technometrics.* 1995;37:41–59.

[6] Nelson P.R.C., Taylor P.A., MacGregor J.F.. Missing data methods in PCA ans PLS: score calculations with incomplete observations *Chemometrics and Intelligent Laboratory Systems.* 1996;35:45–65.

[7] Louwerse D.J., Smilde A.K., Hiers H.A.L.. Cross-validation of Multiway Component Models *Journal of Chemometrics.* 1999;13:491-510.

[8] Arteaga F., Ferrer A.. Dealing with missing data in MSPC: several methods, different interpretations, some examples *Journal of Chemometrics.* 2002;16:408–418.