Perspectives in Drug Discovery and Design, **12/13/14:** 105–113, 1998.
KLUWER/ESCOM

Alternative Partial Least-Squares (PLS) Algorithms

Fredrik Lindgren[a] and Stefan Rännar[b]
[a] *Department of Medicinal Chemistry, Astra Draco AB, P.O. Box 34, S-22100 Lund, Sweden*
[b] *Umetri AB, P.O. Box 7960, S-907 19 Umeå, Sweden*

1. Introduction

Mathematical treatments and modelling of large data structures have always created problems. From the infancy of computers to the late 1980s, the limiting factor when modelling large data structures was often the size of the computer memory. Due to the strong evolution in the field of computer technology, this problem is steadily decreasing. Consequently, when hardware restrictions are becoming less significant, one allows for the development of new, interesting but also calculation-intensive techniques. Typical examples within the area of drug design are techniques like 3D QSAR and molecular library characterization and modelling. However, improved hardware puts the focus on other limiting factors such as speed and efficiency of the mathematical operations performed when processing data. Algorithms and programs must be refined and optimized to meet the demands of today. The desired ‘interactiveness’ in data processing and molecular modelling serves as a good example of the needs of a modern drug design chemist.

A group of data-analytical tools which steadily increase their applicability are the latent variable based ones, such as Principal Components analysis (PCA) [1,2]; Principal Components Regression (PCR) [3]; and Partial Least-squares Regression (PLS) [4–18]. Especially in the disciplines of natural science, their impact has been large during the past few decades, even if statistical methods based on diagonalization of covariance matrices have been used earlier. The usefulness and advantages of projection methods have been discussed by several authors, and for their introduction and applicability we refer to the vast literature [1–22]. However, these methods are frequently studied and their algorithms have been subjects for refinement and optimization.

In this chapter, we will focus on the further developments of the PLS algorithm, using the classical algorithm as a reference for comparison. During the past years, several authors have published modified PLS algorithms with the main aim of increasing the computational speed. Often the code is optimized for a certain type of computational job or a special shape of data matrix. One common step which ties all new developments together is the calculation of some useful variance/covariance and association matrices. Our aim is to point out some commonalities and differences between the individual PLS algorithms in a simple and transparent way. No deep-penetrating computational evaluation was carried out. Instead, the paper will provide a detailed reference list of original articles.

2. Background

Many users of PLS are familiar with its Non-linear Iterative Partial Least-squares (NIPALS) algorithm [5], often referred to as the ‘classical’ algorithm (Fig. 1). The

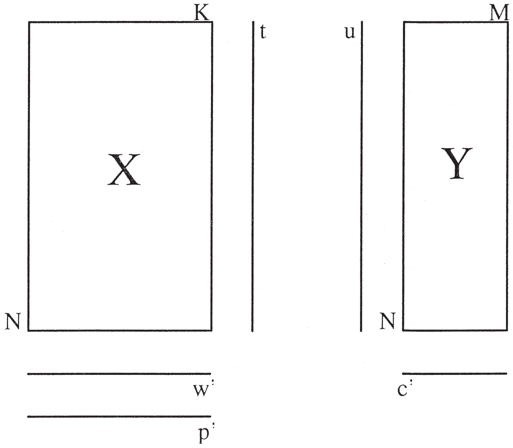

Fig. 1. A graphical illustration of the classical PLS algorithm. The predictor variable matrix **X** *(size* N × K*) and the response variable matrix* **Y** *(size* N × M*), together with the score vectors* **t** *and* **u**, *the weight vectors* **w** *and* **c**, *and the loading vector* **p**.

development was initiated by H. Wold [4–6] and later extended by S. Wold [7, 9]. Several authors have since then shown their interest in the method and many investigations and comparative studies have been performed. The most common topic for comparison is how the predictive properties of PLS relate to other regression methods, but this is not further discussed in this chapter.

Höskuldsson [14] was the first in reformulating PLS as an eigenvalue/eigenvector problem. He showed that the PLS score and weight vectors (**t, u, w, c**) can be determined as eigenvectors to a set of square variance/covariance matrices;

$$\mathbf{w}\, a_1 = (\mathbf{X'YY'X})\mathbf{w} \tag{1}$$

$$\mathbf{c}\, a_2 = (\mathbf{Y'XX'Y})\mathbf{c} \tag{2}$$

$$\mathbf{t}\, a_3 = (\mathbf{XX'YY'})\mathbf{t} \tag{3}$$

$$\mathbf{u}\, a_4 = (\mathbf{YY'XX'})\mathbf{u} \tag{4}$$

where a_1, a_2, a_3 and a_4 are all eigenvalues and the vectors **w, c, t** and **u**, all considered to have their norm equal to one. This evidence is the platform for all new developments.

The advantage of these matrices (Equations 1–4) is their sizes. The two matrices in Equations 1 and 2, (**X′YY′X**) and (**Y′XX′Y**), have the size of K × K (K is the number of **X**-variables) and M × M (M is the number of **Y**-variables), respectively. Hence, no matter how many observations (objects) there are in the original **X** and **Y** matrices, the size of the these matrices will only be dependent upon the number of **X** and **Y** variables (Fig. 2). The contrary situation holds for the matrices (**XX′YY′**) and (**YY′XX′**) (Equations 3 and 4). Their size is N × N (N is the number of observations), so therefore, the number of **X** and **Y** variables will be of no influence. Consequently, matrices with

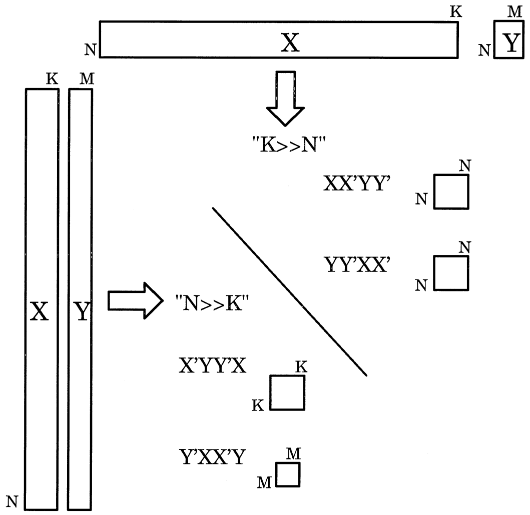

Fig. 2. The 'kernel' matrices of large data structures. For original matrices **X** *and* **Y** *with a large number of objects (case* 'N >> K' *data), the kernel matrices* **X′YY′X** *and* **Y′XX′Y** *are of size* K × K *and* M × M, *respectively. For matrices with a large number of* **X** *and/or* **Y** *variables (case* 'K >> N' *data), the kernel matrices* **XX′YY′** *and* **YY′XX′** *are both of size* N × N.

either a large number of objects or a large number of variables can be condensed into small matrices, containing all information necessary for developing a PLS model.

PLS builds up its model from sequentially calculated dimensions. Before estimating a new dimension, the variance explained by the last component must be removed in a so-called updating procedure. Normally, both **X** and **Y** are updated ($\mathbf{E}_0$ becomes $\mathbf{E}_1$, $\mathbf{E}_1$ becomes $\mathbf{E}_2$, etc., up to $\mathbf{E}_A$), but it has been shown that as long as either of the two is updated, the PLS vectors maintain their orthogonality [14, 23]. The updating procedure is one computation-intensive step and the new algorithms solve this in some alternative ways, either by using small updating matrices or through an orthogonalization procedure.

3. The Algorithms

The choice of algorithm depends strongly on the shape of the data matrices to be studied. In Multivariate Image Analysis [21,22], the number of observations is much larger than the number of variables. This leads to algorithms which utilize the variance/covariance matrices in Equations 1 and 2, since they are independent of the number of observations. An opposite situation occurs in 3D QSAR studies [24,25], where the number of variables usually widely exceeds the number of samples. In this case, one chooses an algorithm based on the association matrices in Equations 3 and 4, since their sizes are independent of the number of variables. In the following sections, we will present some alternative PLS algorithms which all have the advantage of being faster than the classical one for special cases of datasets. For a more thorough comparison of some of the algorithms, we refer to de Jong [26].

3.1. The UNIPALS algorithm

In 1989, Glen et al. [27,28] presented one of the first algorithms to utilize the smaller variance–covariance matrices for PLS computations. This algorithm is called UNIPALS (*UN*iversal *PA*rtial *L*east Squares) and is based on the matrix **Y′XX′Y** of size M × M. the eigenvector of **Y′XX′Y** with the largest eigenvalue is the first weight vector **c** for the **Y** block. From this weight vector and the original **X** and **Y** matrices, all other PLS vectors can be calculated without iteration. However, updating between dimensions is performed on the original **X** and **Y** matrices, equivalent to classical PLS. This implies that the **Y′XX′Y** matrix must be regenerated from the deflated **X** and **Y** for every new dimension. Since the original data matrices are deflated in the same way as in the classical algorithm, the results are identical.

The UNIPALS algorithm has been used in several QSAR studies [29–33] and is, according to the authors, implemented in at least two commercial softwares: the QSAR package from Molecular Simulations Inc. and in Molecular Analysis Pro. (For more detailed information please contact the authors directly.)

3.2. The kernel algorithms

The first kernel algorithm [34,35] developed by Lindgren et al. was an alternative to the classical algorithm for handling datasets where N >> K. Instead of working with **Y′XX′Y** (as in UNIPALS), one calculates the weight vector **w** (the eigenvector with the largest eigenvalue) for the **X** block from the K × K matrix **X′YY′X**. From the weight vector (**w**) and the sub-matrices **X′Y** and **X′X**, all other PLS vectors can be calculated in a straightforward manner. The novelty introduced by the first kernel algorithm was how to update the variance/covariance matrices directly, without interfering with the original **X** and **Y** matrices. By multiplication of an updating matrix (**I–wp′**) of size K × K, explained variance is removed from the variance/covariance matrices:

$$\mathbf{E'YY'E} = (\mathbf{I} - \mathbf{wp'})'\,\mathbf{X'YY'X}\,(\mathbf{I} - \mathbf{wp'}) \tag{5}$$

This simplification of the algorithm leads to major improvements in computational speed since the time-consuming step of creating the variance/covariance matrices has to be performed only once. One should note that only the **X** matrix is deflated. This will, however, not influence the results since deflation of **Y** is optional [14,23].

The second kernel algorithm [36,37] presented by Rännar et al. in 1994 is very much like the first kernel algorithm, but with the important difference that is optimized for datasets which K >> N. These types of matrices often occur in 3D QSAR and also in data from industrial processes. The association matrix **XX′YY′** is independent on the number of predictor variables and services, therefore, as a good start for this version of the kernel algorithm. The algorithm starts with the eigenvector analysis of **XX′YY′**, which gives the score vector **t** for the **X** matrix. From this vector and the small association matrix **YY′**, the score vector **u** for the **Y** block is calculated before proceeding to the next PLS dimension. Also in this kernel algorithm, the deflating is directly performed on the small variance/covariance matrices, now using the updating matrix (**I – tt′**). The last step is the calculation of all of the PLS weights (**w** and **c**) and loading (**p**) vectors using the original **X** and **Y** matrices. These vectors are needed to generate the regression coefficient matrix **B**:

$$\mathbf{B} = \mathbf{W}(\mathbf{P}'\mathbf{W})^{-1}\mathbf{C}' \tag{6}$$

One important point is that both kernel algorithms work well with multiple responses and give identical results as those from the classical PLS algorithm.

The kernel algorithms have lately been modified by de Jong et al. [26,38], resulting in faster and simplified kernel algorithms. Further modifications have been purposed by Dayal et al. [23,39]. They utilize the fact that only one of the matrices **X** or **Y** needs to be deflated. Since the **Y** variables often are few, deflating **Y** instead of **X** saves time.

Neither the original nor the modified kernel algorithms have been implemented in any commercial software, but the MATLAB [40] codes are available from the authors of the different versions.

3.3. The SAMPLS algorithm

SAMple-distance Partial Least Squares, or SAMPLS was presented by Bush et al. in 1993 [41,42] and is also focused on the special case of many descriptor variables and few objects (K >> N). However, the algorithm handles only one **Y** response variable, which is a limiting factor compared to other algorithms. Concerning computational time, the SAMPLS algorithm performs superior to both the classical algorithm and the kernel algorithms; however, the magnitude of the improvement will be noted in a later section. In the field of QSAR, and especially in CoMFA analysis where one only has one response variable, this algorithm is very fast and easy to use. The SAMPLS algorithm is available from QCPE [43] and this code, or a code that is supposed to be identical to the SAMPLS algorithm, is used by Tripos in the QSAR module (for further information we suggest contacting the original author).

The SAMPLS algorithm works with the association matrix **XX′** and the response vector **y** to calculate the score vector **t**, using ordinary matrix–vector multiplication without iteration. This algorithm does not give all the weight and loading vectors that come from other algorithms, but it can still be used for predictions. Not having weights and loadings can be a serious disadvantage since the inter-variable correlation information is lost. In the algorithm, Bush et al. also take advantage of the fact that one can choose to deflate either **X** or **Y** [23]; and in this case, where only one response variable exists, it is very fast to deflate only this vector. This construction makes the updating procedure performed essentially in the same way as in the classical PLS algorithm and, therefore, their results will be identical. However, in order to maintain the orthogonal PLS structure, new score vector (**t**'s) must be othogonalized to the previous ones without the algorithm.

3.4. The SIMPLS algorithm

The last algorithm to be mentioned in this chapter is the Straightforward Implementation of a Statistically Inspired Modification of the PLS method, or SIMPLS algorithm by de Jong [44]. This algorithm was first published in 1993 and the main difference between the above-mentioned algorithms and the SIMPLS algorithm is in the way the orthogonalization of the PLS components is performed. The SIMPLS algorithm aims at describing the scores as direct combinations of the original **X** matrix by a constrained optimization instead of using a deflated **X** matrix. This approach does not always give the same model as classical PLS, but the difference is very small and for most cases not significant. The results from SIMPLS are always identical to classical PLS in the first PLS component, but only in the case of one **Y** response are all components identical. The reason for this small difference is that the matrix **X′Y** is not deflated in the same sense as in the classical algorithm or the kernel algorithm. Instead, the eigenvector analysis is performed on the original **X′Y** matrix projected on the loading vectors from earlier components. This version of deflating will cause the small difference between the SIMPLS and the other PLS algorithm. The SIMPLS algorithm is, however, a very fast PLS algorithm for all kinds of shapes of data matrice (the MATLAB code is available from Dr. de Jong upon request).

4. Discussion and Concluding Remarks

The new PLS algorithms are often presented as revolutionary when comparing their speed to the classical algorithm [41]. This holds true in many cases, but sometimes the improvements are poor or even absent. Why is that? In principle the described algorithms contain one initial and rather time-consuming step, namely the computation of the variance/covariance or association matrices. In a comparative study with the classical algorithm, the time spent on calculating these condensed matrices must also be included. This is sometimes forgotten, which inevitably generates misleading results [41].

The classical PLS algorithm is always described as an iterative procedure. However, when only one **Y**-variable is modelled (most common case), the algorithm is non-iterative. This implies that only a fixed number of vector–matrix multiplications must be performed to generate the PLS model of a certain dimensionality.

Adding these two facts together (time-consuming matrix calculation and non-iterative PLS1 modelling), one quickly realizes that the classical PLS algorithm will outperform other algorithms in some cases. A typical situation is the calculation of a low-dimensional (1–3 dimension) PLS1 model without cross-validation [45,46]. In such a case, the calculation of the variance/covariance or association matrices will be more tedious than using the classical algorithm directly.

On the contrary, the new algorithms will prove advantageous in cases of repetitive modelling, as in cross-validation [45,46], bootstrapping [47] and in some variable selection techniques [48]. The great advantage of both variance/covariance and association matrices is that both objects and variables can be either added or removed, without recalculation of the condensed matrices. Other treatments, like mean-centering and scaling can also be performed directly on the condensed matrix form. These key features lead to considerable speed-up in the computation of repetitive modelling. A typical example is the cross-validation (CV) step, and the use of CV, or some other validation procedure, is strongly recommended in all types of PLS modelling. The only features which alter between consecutive runs in a CV loop are the division between training and test set objects, and some possible rescaling. Hence, CV can easily be performed on these condensed matrices directly. A 'leave-one-out' CV procedure for a typical 3D QSAR dataset would only take a limited number of seconds. The presented SAMPLS algorithm is now commonly used in CoMFA cross-validation runs and gives results identical to those from the classical algorithm, provided that no rescaling is performed within the CV procedure.

Other dataset-related features which favor the new algorithms are PLS2 modelling (more than one Y-variable) and the extraction of a large number of PLS components. Still, one has to remember that the major improvements are gained for datasets with either 'N >> K' or 'N << K'. When N and K are of similar size, no significant improvement is made.

A common problem among all alternative PLS algorithms is how to deal with missing values in the data. One cannot create the appropriate variance/covariance or association matrices without adding some type of an approximate value to fill the data gaps. One approach which deals with this problem was presented by Rännar et al. [37] and involved using the EM algorithm [49]. The classical PLS algorithm has no similar problem since it can deal with missing data in a straightforward way, without addition of approximate values.

Finally, one can conclude that there exist several alternative PLS algorithms, all optimized for different assignments. The choice of algorithm is very much related to questions like, 'What is my application area?' and 'What am I going to do?'. The answers to these questions will define if PLS1 or PLS2 modelling is needed, if K >> N or N >> K, if extensive cross-validation is foreseen, and so forth. These features will outline the computational task and one selects an algorithm which fulfils the defined requirements. The more specific the definition becomes, the more optimized algorithm can be chosen — e.g. the SAMPLS for 3D QSAR. For more general PLS modelling, the two complementary kernel algorithms and the classical algorithm are a sound choice.

References

1. Jackson, J.E., *A user's guide to principal components*, Wiley, New York, 1991.
2. Jolliffe, I.T., *Principle components analysis*, Springer-Verlag, New York, 1986.
3. Martens, H. and Naes, T., *Multivariate calibration*, Wiley, Chichester, U.K., 1989.
4. Wold, H., In David, F. (Ed.) *Research papers in statistics*, Wiley, New York, 1966, pp. 411–444.
5. Wold, H., *Path models with latent variables: The NIPALS approach*, In Blalock, H.M., Aganbegian, A., Borodkin, F.M., Boudon, R. and Capecchi, V. (Eds.) Quantitative sociology, Academic Press, New York, 1975, pp. 307–357.
6. Jöreskog, K.-G. and Wold, H. (Eds.) *System under indirect observation*, Vols 1 and 2, North-Holland, Amsterdam, The Netherlands, 1982.

7. Wold, S., Martens, M. and Wold, H., *The multivariate calibration problem in chemistry solved by the PLS method*, In Ruhe, A. and Kågström, B. (Eds.) Matrix Pencils, Springer-Verlag, Heidelberg, Germany, 1983, pp. 286–293.
8. Martens, H. and Jensen, S.-A., *Partial least squares regression: A new two-stage NIR calibration method*, In Holas, J. and Kratochvil, J. (Eds.) Progress in cereal chemistry and technology, Elsevier, Amsterdam, The Netherlands, 1983, pp. 607–647.
9. Wold, S., Ruhe, A., Wold, H. and Dunn III, W.J., *The collinearity problem in linear regression: The partial least squares approach to generalized inverses*, Siam J. Sci. Stat. Comput., 5 (1984), 735–743.
10. Geladi, P. and Kowalski, B.R., *Partial least squares regression (PLS): A tutorial*, Analyt. Chim. Acta, 185 (1986), 1–17.
11. Lorber, A., Wangen, L. and Kowalski, B., *The theoretical foundation for the PLS algorithm*, J. Chemometrics, 1 (1987) 19–31.
12. Manne, R., *Analysis of two partial squares algorithms for multivariate calibration*, Chemometrics Intell. Lab. Syst., 2 (1987) 187–197.
13. Helland, I.S., *The structure of partial least squares regression*, Commun. Stat. Simul. Comput., 17 (1988) 581–607.
14. Höskuldsson, A., *PLS regression methods*, J. Chemometrics, 2 (1988) 211–228.
15. Geladi, P., *Notes on the history and nature of partial least squares (PLS) modeling*, J. Chemometrics, 2 (1988) 231–246.
16. Phatak, A., *Evaluation of some multivariate methods and their applications in chemical engineering*, Ph.D. thesis, University of Waterloo, Ontario, Canada, 1993.
17. Garthwaite, P.H., *An interpretation of partial least squares*, J. Am. Stat. Assoc., 89 (1994) 122–127.
18. Wold, S., Albano, C., Dunn III, W.J., Edlund, U., Esbensen, K., Geladi, P., Hellberg, S., Johansson, E., Lindberg, W. and Sjöström, M., *Multivariate data analysis in chemistry*, In Kowalski, B.R. (Ed.) Chemometrics: Mathematics and statistics in chemistry, Reidel, Dordrecht, The Netherlands, 1984, pp. 17–95.
19. McGregor, J.F. and Nomikos, P., *Monitoring batch processes*, NATO Advanced Study Institute for Batch Processing Systems Engineering, Antalya, Turkey, Springer-Verlag, Heidelberg, Germany, 1992.
20. Forina, M., Armanino, C., Castino, M. and Ubigli, M., *Multivariate data analysis as a discriminating method of the origin of wines*, Vities, 25 (1986) 189–201.
21. Esbensen, K. and Geladi, P., *Strategy of multivariate image analysis (MIA)*, Chemometrics Intell. Lab. Syst., 7 (1989) 67–86.
22. Geladi, P. and Esbensen, K., *Regression on multivariate images: Principal component regression for modeling, prediction and visual diagnostic tools*, J. Chemometrics, 5 (1991) 97–111.
23. Dayal, B.S. and MacGregor, J.F., *Improved PLS algorithms*, J. Chemometrics, 11 (1997) 73–85.
24. Cramer III, R.D., Bunce, J.D., Patterson, D.E. and Frank, I.E., *Crossvalidation bootstrapping and partial least squares compared with multiple regression in conventional QSAR studies*, Quant. Struct..-Act. Relat., 7 (1988) 18–25.
25. Kubinyi, H., (Ed.), *3D-QSAR in drug design: Theory, methods and applications*, ESCOM, Leiden, The Netherlands, 1993.
26. De Jong, S., *A comparison algorithms for partial least squares regression*, J. Chemometrics, (1997) (submitted).
27. Glen, W.G., Dunn III, W.J. and Scott, D.R., *Principal components analysis and partial least squares regression*, Tetrahedron Comput. Methodol., 2 (1989) 349–376.
28. Glen, W.G., Dunn III, W.J., Sarker, M. and Scott, D.R., *UNIPALS: Software for principal components analysis and partial least squares regression*, Tetrahedron Comput. Methodol., 2 (1989) 377–396.
29. Hopfinger, A.J., Burke, B.J. and Dunn III, W.J., *A generalized formalism of three-dimensional quantitative structure–activity relationship analysis for flexible molecules using tensor representation*, J. Med. Chem., 37 (1994) 3768–3774.
30. Burke, B.J., Dunn III, W.J. and Hopfinger, A.J., *Construction of a molecular shape analysis: Three-dimensional quantitative structure–analysis relationship for an analog series of pyridobenzodiazepinone inhibitors of muscarinic 2 and 3 receptors*, J. Med. Chem., 37 (1994) 3775–3788.

31. Collantes, E.R. and Dunn III, W.J., *Amino acid side chain descriptors for quantitative structure–activity relationship studies of peptide analogues*, J. Med. Chem., 38 (1995) 2705–2713.
32. Dunn III, W.J., Hopfinger, A.J., Catana, C. and Duraiswami, C., *Solution of the conformation and alignment tensors for the binding of triethoprim and its analogs to dihydrofolate reductase: 3D-quantitative structure–activity relationship study using molecular shape analysis, 3-way partial least-squares regression, and 3-way factor analysis*, J. Med. Chem. 39 (1996) 4825–4832.
33. Dunn III, W.J. and Rogers, D., *Genetic partial least squares in QSAR*, In Devillers, J. (Ed.) Genetic algorithms in molecular modeling, Academic Press, London, 1996, pp. 109–130.
34. Lindgren, F., Geladi, P. and Wold, S., *The kernel algorithm for PLS.*, J. Chemometrics, 7 (1993) 45–59.
35. Lindgren, F., Geladi, P. and Wold, S., *Kernel-based PLS regression: Cross validation and applications to spectral data*, J. Chemometrics, 8 (1994) 377–389.
36. Rännar, S., Lindgren, F., Geladi, P. and Wold, S., *A PLS kernel algorithm for PLS, for data sets with many variables and less objects: Part 1. Theory and Algorithm.*, J. Chemometrics, 8 (1994) 111–125.
37. Rännar, S., Lindgren, F., Geladi, P. and Wold, S., *A PLS kernel algorithm for data sets with many variables and less objects: part 2. Cross-validation, missing data and examples*, J. Chemometrics, 9 (1995) 459–470.
38. De Jong, S. and Ter Braak, C.J.F., *Comments on the PLS kernel algorithm*, J. Chemometrics, 8 (1994) 169–174.
39. Dayal, B.S. and MacGregor, J.F., *Recursive exponentially weighted PLS and its applications to adaptive control and prediction*, J. Process Contr. (1997) (submitted).
40. MATLB^{TM} Reference Guide, The Math Works Inc., Natick, U.S.A. (1992).
41. Bush, B.L. and Nachbar Jr., R.B., *Sample-distance partial least squares: PLS optimized for many variables, with application to CoMFA*, J. Comput.-Aided Mol. Design , 7 (1993) 587–619.
42. Sheridan, R.P., Nachbar Jr., R.B. and Bush, B.L., *Extending the trend vector: The trend matrix and sample based partial least squares*, J. Comput.-Aided Mol. Design, 8 (1994) 323–340.
43. QCPE 650: Ver. 1.3, 1994, Quantum Chemistry Program Exchange, Indiana University; Bloomington, IN 47404, U.S.A.: qcpe@indiana.edu.
44. De Jong, S., *SIMPLS: An alternative approach to partial least squares regression*, Chemometrics Intell. Lab. Syst., 18 (1993) 25–263.
45. Stone, M., *Cross-validatory choice and assessment of statistical predictions*, J. Royal Stat. Soc., B, 36 (1974) 111–133.
46. Geisser, S., *A Predictive approach to the random effect model*, Biometrika, 61 (1974) 101–107.
47. Leger, C., Politis, D.N. and Romano, J.P., *Bootstrap technology and applications*, Technometrics, 34 (1992) 378–398.
48. Baroni, M., Costantino, G., Cruciani, G., Riganelli, D., Valigi, R. and Clementi, S., *Generating optimal linear PLS estimations (GOLPE): An advanced chemometric tool for handling 3D QSAR problems*, Quant. Struct.-Act. Relat., 12 (1993) 9–20.
49. Little, R.J.A. and Rubin, D.B., *Statistical analysis with missing data*, Wiley, New York, 1987.