Gram-Schmidt versus Bauer-Rutishauser in alternating least-squares algorithms for three-way data

Kroonenberg, P.M.; Berge, Jos M.F. ten; Brouwer, P.; Kiers, H.

Computational Statistics Quarterly 2, 81-87
© Physica-Verlag 1989

Gram-Schmidt Versus Bauer-Rutishauser in Alternating Least-Squares Algorithms for Three-Mode Principal Component Analysis

P. M. Kroonenberg,[1] J. M. F. ten Berge,[2] P. Brouwer,[1] and H. Kiers[2]

[1]Department of Education, Leiden University, P. O. Box 9555, 2300 RB Leiden, The Netherlands
[2]University of Groningen

Summary

The effect of replacing a Bauer-Rutishauser step using an eigendecomposition by a Gram-Schmidt orthogonalization step in an algorithm for three-mode principal component analysis was explored both theoretically and empirically. The results showed that the latter procedure has a slight to moderate advantage over the former one.

Introduction

In nonlinear data analysis, alternating least squares algorithms are frequently employed. The basic principle underlying these algorithms is that for a specific loss function the parameters can be divided into blocks, which can be estimated separately with least squares conditionally upon the other blocks. By estimating each block in turn, the overall loss function can be shown to converge in a monotone fashion to at least a local optimum. A large number of applications of alternating least squares algorithms have been reviewed by Young (1981).

Many alternating least squares algorithms initially used the singular value decomposition (SVD) in one of their steps. However, as shown by Gifi (1981, p. 180), the SVD can sometimes be replaced by (modified) Gram-Schmidt orthogonalization. The reason for using Gram-Schmidt (GS) orthogonalization rather than a singular value decomposition is that one GS-step is cheaper than one SVD step (see Schwarz, Rutishauser, & Stiefel, 1968, p. 186).

0723-712X/89/2/81-87 $2.50

Three-mode principal component analysis

In this paper, we will explore the possibility of replacing an eigendecomposition by a Gram-Schmidt orthogonalization for TUCKALS3, an alternating least squares algorithm for three-mode principal component analysis. The model for such an analysis is called the Tucker3 model, first developed under the name of three-mode factor analysis by Tucker (1966). The model is generally used to describe patterns present in (or underlying) three-mode data, i.e. data which can be arranged in a block of I-by-J-by-K elements. Three-mode data arise, for instance, if I subjects are measured on a battery of J tests on K occasions, but many other designs exist as well. Three-mode principal component analysis is used to derive both components for each of the modes and the links between components of different modes. For a short introduction see Kroonenberg (1988), a longer treatise is Kroonenberg (1983), while the TUCKALS3 algorithm was first described in Kroonenberg and De Leeuw (1980). These authors also described another algorithm, TUCKALS2, for a different three-mode model, and everything discussed in this paper is valid for that algorithm as well.

In matrix notation the Tucker3 model is

$$X = AG(C' \otimes B') + E, \qquad (1)$$

where the (Ixp) matrix A, the (Jxq) matrix B, and the (Kxr) matrix C are the component matrices for the first, second, and third mode, respectively, G is the (pxqxr) core matrix (here written as a p-by-qr matrix) with the weights for the combinations of components of the three modes, X the (IxJxK) data matrix (written as an I-by-JK matrix), E is the I-by-JK matrix of errors of approximation, and $\otimes$ is the (right) Kronecker product. Without loss of generality, the matrices A, B, and C are constrained to be orthonormal columnwise. In general, not all components are needed, but only the first p, q, and r of them for the first, second and third modes, respectively. To find the estimates for the parameters in the model we have to mimimize the loss function

$$f(A,B,C,G) = ||X - AG(C' \otimes B')||^2 \qquad (2)$$

where $AG(C' \otimes B')$ contains the reconstructed or estimated data on the basis of the model with fixed numbers of components, p, q, and r.

The loss function may be rewritten in three ways by focussing on either A, B, or C; we will look here only at the A version, the other ways can be found by cyclically permuting A, B, and C. First, minimizing (2) with respect to G leads to a unique $G = A'X(C \otimes B)$, so that (2) can be rewritten (for details, see Kroonenberg & De Leeuw, 1980) as

$$f(A,B,C) = ||X - AA'X(C\otimes B)(C'\otimes B')||^2 =$$
$$= \text{tr } X'X - \text{tr } A'\{X(C\otimes B)(C'\otimes B')X'\}A =$$
$$= \text{tr } X'X - \text{tr } A'PA, \qquad (3)$$

with P implicitly defined. Clearly, P is nonnegative definite, and in practical applications positive definiteness can always be achieved by choosing a small enough number of components. Minimizing f over A is equivalent to maximizing $p(A) = \text{tr } A'PA$ (and analogously for B and C).

Present algorithm

Under the condition that A, B, and C are columnwise orthonormal, the function f is minimized using an alternating least squares algorithm, in which in each main iteration step, A, B, and C are updated in turn, while keeping the other two parameter matrices fixed. In the A-substep tr A'PA is maximized under the columnwise orthonormality constraint on A by computing eigenvectors of P. If one would indeed carry out a full eigendecomposition of the I-by-I matrix P at each substep of the main iteration procedure (and those for the J-by-J and K-by-K matrices, Q and R, respectively, to find B and C as well), huge amounts of computing time would be necessary irrespective of the eigendecomposition procedure used. To circumvent this, Kroonenberg and De Leeuw (1980) used only the first step of the simultaneous iteration procedure of Bauer-Rutishauser (Schwarz, et al., 1968; Rutishauser, 1969; and Nikolai, 1979, for an algorithm), to find a new A. (It is not entirely clear in the literature who actually is the originator of this procedure, neither of the above references are explicit about it; we follow here Kroonenberg and De Leeuw's nomenclature.) The basic updating of A after α steps has the form

$$A_{\alpha+1} = P_\alpha A_\alpha T_\alpha L_\alpha^{-\frac{1}{2}}, \qquad (4)$$

where T_α is an eigenvector matrix of $A_\alpha'P_\alpha^2A_\alpha$, and L_α the diagonal matrix with the corresponding eigenvalues. By choosing a small enough number of components, the eigenvalues in L_α will always be positive, so that their reciprocals exist. The advantage of using the eigendecomposition of $A_\alpha'P_\alpha^2A_\alpha$ is that it requires only the eigendecomposition of a p-by-p matrix, where p, the number of components of A, is in practice very much smaller than I, and usually 2 or 3, and seldom bigger than 5. It turns out that the eigendecomposition can be fruitfully solved by a Jacobi eigendecomposition technique, because in the later stages of the algorithm, $A_\alpha'P_\alpha^2A_\alpha$ becomes more and more a diagonal matrix because A_α converges to the eigenvector matrix of P_α^2, and thus also of P_α. In Kroonenberg and De Leeuw (1980, p. 93, 94) it was shown that (4) decreases the function f, which is all that is required during

the iteration procedure. Note that because $A_{\alpha+1}$ in (4) is not yet the eigenvector matrix of P_α, it does not maximize $p(A) = \text{tr } A'PA$, but only improves its value. Furthermore, equation (4) shows that one step of a Bauer-Rutishauser iteration is in fact an orthonormalization of $P_\alpha A_\alpha$.

The Gram-Schmidt alternative

As mentioned above, the central part of a substep of the main algorithm is to *maximize* $p(A_\alpha) = \text{tr } A_\alpha' P_\alpha A_\alpha$, *or rather improve the objective function* f *by finding* a new $A_{\alpha+1}$ under the condition that $A_\alpha' A_\alpha = I$. Furthermore, the Bauer-Rutishauser step is equivalent to an orthonormalization of $P_\alpha A_\alpha$. By applying the Gram-Schmidt orthonormalization procedure to $P_\alpha A_\alpha$ an update of A_α, $A^*_{\alpha+1} = GS(P_\alpha A_\alpha)$, is produced. As two orthonormal versions of the same matrix are in the same space, this $A^*_{\alpha+1}$ is a rotated version of the Bauer-Rutishauser one in (4). Because p(A) is insensitive to this kind of orthonormal transformations the same value of p(A) results for both procedures. In other words, the Bauer-Rutishauser step may be replaced by a Gram-Schmidt step without changing the value of p(A) and the objective function. Moreover, the objective function should converge in the same number of steps for both procedures.

Theoretically, *per iteration step Gram-Schmidt should computationally be cheaper* as the procedure uses less floating point operations. How this works in practice, will be taken up in the next section.

Another question is whether after convergence of f over A, B, and C, the orientation of $A_{\alpha+1}$ and $A^*_{\alpha+1}$ (and their B and C counterparts) is the same with both procedures. Schwarz, et al. (1968, p. 182ff) showed that for a fixed matrix the iteration vectors in the Gram-Schmidt orthogonalization converge to eigenvectors. In the present algorithm, P_α changes every main iteration step because B and C are updated as well. However, at the convergence point P_α changes no longer, so that *also here the Gram-Schmidt procedure will converge to the same values for the components*, as the Bauer-Rutishauser algorithm does. It is, however, a matter of numerical convergence speed whether the same number of iterations are necessary to reach the same components, even after the loss function has converged. In theory, in the Gram-Schmidt procedure the convergence to the eigenvectors should take longer, as the Bauer-Rutishauser procedure was especially designed to accelerate such a convergence (Rutishauser, 1969), and this is borne out by practice.

How the different behaviour of the two procedures with respect to the eigenvectors affects the overall convergence given that one wants to have convergence of both the objective function and the eigenvectors, depends a great deal on the *relative accuracies for convergence of the objective function, and that of the* component matrices. In the TUCKALS2 and TUCKALS3 programs (Kroonenberg &

Brouwer, 1985a,b), the default accuracies have been set in such a way that the components nearly always converge before the objective function does.

Numerical results

Below, some numerical results for comparing the two algorithms are summarized. As in both cases the outcomes of the analyses are identical after convergence, only computing time, number of iterations, and the estimated time per iteration are given. Computations were performed on an Amdahl 5860; execution times pertain to the complete run of the TUCKALS3 program.

Table 1

Execution Time Differences between Bauer-Rutishauser (BR) and Gram-Schmidt (GS) Procedures

Components of modes	CPU secs		Niter	Msecs/it	
1 2 3	BR	GS	BR & GS	BR	GS
Tongue Shape Data (10x13x5)					
2 x 2 x 2	.24	.23	14	17	16
4 x 2 x 2	.56	.52	27	21	19
Assimilation Resistance Data (8x6x13)					
2 x 2 x 2	.57	.55	19	30	29
4 x 4 x 4	2.32	2.16	47	49	46
5 x 4 x 10	7.21	6.13	81	89	76
Strange Situation Data (7x5x431)					
2 x 2 x 2	47.29	46.17	31	1525	1489
3 x 2 x 6	224.71	221.02	71	3165	3113

Notes: The "msecs/it" are only estimates, as the CPU secs include the overhead of running the rest of the program.

Data sources: Tongue shape data: Harshman, Ladefoged, & Goldstein, 1977; Assimilation resistance data: Eckblad (unpublished, see Kroonenberg & Snyder, in press, and Eckblad ,1981); Strange situation data: Sagi, Van IJzendoorn, & Koren (1989).

Discussion

The single Bauer-Rutishauser step (4) contains a Jacobi eigenroutine, which is iterative itself, and its number of iterations depends on the diagonality of $A_\alpha' P_\alpha^2 A_\alpha$, on its dimensions and, of course, on the accuracy conditions. The Gram-Schmidt algorithm on the other hand is not iterative, so that it executes faster. The practical results listed above show that Gram-Schmidt, indeed, executes consistently faster per iteration, even though its advantage ranges from minimal to modest. From the numerical information it is not clear which factors are most important in determining when appreciable gains occur for Gram-Schmidt over Bauer-Rutishauser.

As Gram-Schmidt consistently outperforms Bauer-Rutishauser this not really a problem in practice.

The disappointingly little gain in execution speed can **a posteriori** easily be explained by performing some calculations on the computational process. Assuming that the Jacobi routine needs 20 iterations for each of the $\frac{1}{2}n(n-1)$ off-diagonal elements of $A_\alpha{}^1P_\alpha{}^2A_\alpha$, the number of floating point operations (flops) for Jacobi can be calculated for each iteration step. For the 7x5x431 data matrix collected by Sagi (Sagi, Van IJzendoorn, & Koren, 1989), extracting 3, 2, and 6 components respectively, the total number of flops in Jacobi for one main iteration of the TUCKALS3 algorithm amount to 21 860 flops. In each iteration, however, the matrices P, Q, and R (the latter defined via permutations of A, B, and C) must be computed and the number of flops to achieve this is over 782 000. If we ignore the matrix multiplications in the Bauer-Rutishauser step, even the (probably) overestimated execution time in Jacobi accounts for only about 3% of the total time for one iteration. Thus Gram-Schmidt, which needs the same P, Q, and R, cannot but provide a small improvement. Especially when the number of components are small, as is true in most cases (also for the Sagi data), the relative gain is small. However, as Gram-Schmidt outperforms Bauer-Rutishauser in all cases, replacing the latter procedure with the former seems called for. In future versions of the programs TUCKALS3 and TUCKALS2 this will be implemented. There is a certain irony in this replacement. To compute the eigenvectors of a fixed matrix, (probably) Rutishauser devised the Bauer-Rutishauser step as an acceleration of the Gram-Schmidt procedure. In the present context, it turns out that it is wiser to fall back on Gram-Schmidt, because **it** is cheaper.

Acknowledgements

We would like to thank Jan de Leeuw for raising the problem and our interest in the research reported here. Requests for reprints and information about the TUCKALS programs may be addressed to the first author.

References

Eckblad, G. (1981). Assimilation resistance and affective response in problem solving. **Scandinavian Journal of Psychology, 22,** 1-16.

Gifi, A. (1981). **Nonlinear multivariate analysis** (preliminary edition). Leiden, The Netherlands: Department of Data Theory, University of Leiden.

Harshman, R.A., Ladefoged, P., & Goldstein, L. (1977). Factor analysis of tongue shapes. **Journal of the Acoustical Society of America, 62,** 693-707.

Kroonenberg, P.M. (1983). **Three-mode principal component analysis: Theory and applications.** Leiden, The Netherlands: DSWO Press.

Kroonenberg, P.M. (1988). Three-mode analysis. In S. Kotz, N. L. Johnson, & C.B. Read (Eds.), **Encyclopedia of Statistical Sciences** (Vol. 9; pp. 231 - 236). New York: Wiley.

Kroonenberg, P.M. & Brouwer, P. (1985a). **User's guide to TUCKALS3, Version 4.0** (WEP-Reeks, WR 85-09-RP). Leiden, The Netherlands: Department of Education, Leiden University.

Kroonenberg, P.M. & Brouwer, P. (1985b). **User's guide to TUCKALS2, Version 4.0**

(WEP-Reeks, WR 85-12-RP). Leiden, The Netherlands: Department of Education, Leiden University.

Kroonenberg, P.M., & De Leeuw, J. (1980). Principal component analysis of three-mode data by means of alternating least squares algorithms. **Psychometrika, 45,** 69-97.

Kroonenberg, P.M., & Snyder Jr, C.W. (in press). Individual differences in assimilation resistance and affective responses in problem solving. **Multivariate Behavioral Research.**

Nikolai, P.J. (1979). Algorithm 538. Eigenvectors and eigenvalues of real generalized symmetric matrices by simultaneous iteration [F2]. **ACM Transactions on Mathematical Software, 5,** 118-125.

Rutishauser, H. (1969). Computational aspects of F.L. Bauer's simultaneous iteration method. **Numerische Mathematik, 13,** 4-13.

Sagi, A., Van IJzendoorn, M.H., & Koren, N. (1989). **Primary appraisal of the Strange Situation: A cross-sectional analysis of preseparation episodes** (Internal report). Haifa, Israel: University of Haifa.

Schwartz, H.R., Rutishauser, H., & Stiefel, E. (1968). **Numerik. Symmetrischer Matrizen** [Numerical analysis of symmetric matrices]. Stuttgart, FRG: Teubner.

Tucker, L.R. (1966). Some mathematical notes on three-mode factor analysis. **Psychometrika, 31,** 279-311.

Young, F.W. (1981). Quantative analysis of qualitative data. **Psychometrika, 46,** 357-388.

Received: August 1987
Revised: May 1989