SIAM J. Matrix Anal. Appl.
Vol. 35, No. 1, pp. 279–291

STABILITY OF TWO DIRECT METHODS FOR BIDIAGONALIZATION AND PARTIAL LEAST SQUARES[*]

ÅKE BJÖRCK[†]

Dedicated to Michael Saunders on his 70th birthday

Abstract. The partial least squares (PLS) method computes a sequence of approximate solutions $x_k \in \mathcal{K}_k(A^TA, A^Tb)$, $k = 1, 2, \ldots,$ to the least squares problem $\min_x \|Ax - b\|_2$. If carried out to completion, the method always terminates with the pseudoinverse solution $x^\dagger = A^\dagger b$. Two direct PLS algorithms are analyzed. The first uses the Golub–Kahan Householder algorithm for reducing A to upper bidiagonal form. The second is the NIPALS PLS algorithm, due to Wold et al., which is based on rank-reducing orthogonal projections. The Householder algorithm is known to be mixed forward-backward stable. Numerical results are given, that support the conjecture that the NIPALS PLS algorithm shares this stability property. We draw attention to a flaw in some descriptions and implementations of this algorithm, related to a similar problem in Gram–Schmidt orthogonalization, that spoils its otherwise excellent stability. For large-scale sparse or structured problems, the iterative algorithm LSQR is an attractive alternative, provided an implementation with reorthogonalization is used.

Key words. partial least squares, bidiagonalization, core problem, stability, regression, NIPALS, Householder reflector, modified Gram–Schmidt orthogonalization

AMS subject classifications. 65F25, 65G05, 65F05, 65F20

DOI. 10.1137/120895639

1. Introduction.

NIPALS (nonlinear iterative partial least squares), due to H. Wold [27], is a method for prediction and cause-effect inference. It originated in statistical applications, specifically economics. Wold et al. [28] developed the related partial least squares (PLS) algorithm, which is used extensively in chemometrics and other applications.

Let $A \in \mathbf{R}^{m\times n}$ be a given matrix, let $b \in \mathbf{R}^m$ be a given right-hand side, and consider the least squares problem

$$\min_x \|Ax - b\|_2. \tag{1.1}$$

Recall that the pseudoinverse solution $x^\dagger = A^\dagger b$ of (1.1) is the unique least squares solution that satisfies $x \in \mathcal{R}(A^T)$. The PLS approximations to $x^\dagger$ are orthogonal projections onto certain Krylov subspaces.

Definition 1.1. *The PLS approximations x_k, $k = 1, 2, \ldots,$ to problem* (1.1) *are the solutions to the subproblems*

$$\min_{x_k} \|Ax_k - b\|_2 \quad \textit{subject to} \quad x_k \in \mathcal{K}_k(A^TA, A^Tb), \quad k = 1, 2, \ldots, \tag{1.2}$$

where $\mathcal{K}_k(B, y)$ denotes the Krylov subspace $\operatorname{span}\{y, By, \ldots, B^{k-1}y\}$.

If carried out to completion, PLS terminates with the pseudoinverse solution $x^\dagger = A^\dagger b$. This is a linear function of b, but intermediate PLS approximations x_k

[*] Received by the editors October 19, 2012; accepted for publication (in revised form) by D. B. Szyld December 3, 2013; published electronically March 11, 2014.
http://www.siam.org/journals/simax/35-1/89563.html

[†] Department of Mathematics, Linköping University, Linköping S-581 83, Sweden (akbjo@math.liu.se).

depend nonlinearly on b; see Eldén [10]. Therefore, the application of many model-selection criteria is not appropriate for PLS; see Frank and Friedman [11]. Several generalizations of the basic PLS algorithm have been devised; see Wold, Sjöström, and Eriksson [29].

The aim of this paper is to describe and analyze two direct algorithms for PLS. The first is based on an algorithm, due to Golub and Kahan [12], for the reduction of A to upper bidiagonal form by Householder reflections. Using known results it can be shown that for this algorithm the computed results are very close to the exact results corresponding to small perturbations of the data. Following Higham [13, p. 7], we call this mixed forward-backward stability. The second is the NIPALS PLS algorithm by Wold et al. [28], which is based on rank-reducing orthogonal projections, as in the modified Gram–Schmidt (MGS) QR factorization of A.

It was noticed already by Wold et al. [28] that in exact arithmetic, the sequence of PLS approximations is the same as of those computed by the conjugate gradient (CG) method applied to the normal equation (Stiefel [24]) and the LSQR algorithm (Paige and Saunders [17]). LSQR is based on a Lanczos-type process for computing the *lower bidiagonal* decomposition of A called Bidiag1 in [17]. The related *upper bidiagonal* decomposition due to Golub and Kahan [12] is referred to as Bidiag2. The relationship between PLS, the CG method, and LSQR is also discussed by Manne [14], Phatak and de Hoog [21], and Eldén [10].

The outline of the paper is as follows. In section 2, some properties of the Krylov subspaces used in PLS and their relations to the pseudoinverse solution are given. Section 3 presents the Householder upper bidiagonalization algorithm and its use for computing the PLS approximations. The NIPALS PLS algorithm is presented in section 4. Section 5 analyzes an unfortunate and unnecessary "simplification" made in many descriptions of the NIPALS PLS algorithms, which is shown to destroy its stability. Numerical results are given showing that, properly implemented, the NIPALS PLS algorithm gives results that are as accurate as those from the Golub–Kahan Householder algorithm. In section 6 the stability properties of the two algorithms are studied. The mixed forward-backward stability of the Householder PLS algorithm is shown using known results. It is conjectured that the properly implemented NIPALS PLS algorithm shares this stability property. Section 7 contains some comments on stopping criteria and the rate of convergence of the PLS approximations

2. Preliminaries.

The least squares problem (1.1) has a unique solution $x^\dagger$ of minimal norm $\|x\|_2$. This is called the pseudoinverse solution and can be characterized by the two conditions

$$A^TAx = A^Tb, \qquad x^\dagger \in \mathcal{R}(A^T). \tag{2.1}$$

In the following we assume that $A^Tb \neq 0$, since otherwise the solution trivially equals $x^\dagger = 0$. Let $B \in \mathbf{R}^{n\times n}$ be a given matrix and $y \in \mathbf{R}^n$ a given vector. Then

$$\mathcal{K}_k(B,y) = \operatorname{span}\{y, By, \ldots, B^{k-1}y\} \quad (k \geq 1) \tag{2.2}$$

is a Krylov subspace and $K_k = (y, By, \ldots, B^{k-1}y) \in \mathbf{R}^{n\times k}$ a Krylov matrix. The sequence of Krylov subspaces are nested, i.e.,

$$\mathcal{K}_k(B,y) \subseteq \mathcal{K}_{k+1}(B,y).$$

In any Krylov sequence there is a first vector y_{p+1}, $p \leq n$, that can be expressed as a linear combination of the preceding ones. Then, there is a polynomial

$$\psi(\lambda) = \gamma_1 + \gamma_2\lambda + \cdots + \gamma_p\lambda^{p-1} + \lambda^p$$

of degree p such that $\psi(B)y = 0$. This polynomial is said to annihilate y and to be minimal for y. The grade of y with respect to B is said to be p.

We now give a relation between the Krylov vectors $y_k = (A^TA)^{k-1}A^Tb \in \mathcal{R}(A^T)$, $k = 1, 2, \dots$, and a subset of the right singular vectors v_i of A that is fundamental for the PLS approximations.

LEMMA 2.1. *Let A have s distinct nonzero singular values $\sigma_1 > \sigma_2 > \cdots > \sigma_s$, $s \le \min\{m, n\}$. Let c_i be the norm of the orthogonal projection of b onto the left singular subspace corresponding to σ_i. Then the grade of A^Tb with respect to A^TA is $p \le s$, where p is the number of nonzero coefficients c_i.*

Proof. Using the SVD of A, the Krylov vectors can be written as

$$y_k = (A^TA)^{k-1}A^Tb = \sum_{i=1}^{s} c_i\sigma_i^{2k-1}v_i, \quad k \ge 1. \tag{2.3}$$

The sum is taken over the s *distinct* nonzero singular values. For a simple singular value σ_i with left singular vector u_i, $c_i = u_i^Tb$. For a multiple singular value the orthonormal basis for the left singular subspace can be chosen so that b has a nonzero projection on just one unit left singular vector u_i in the singular subspace. Then, in (2.3) $c_i = u_i^Tb$ and $v_i = A^Tu_i/\sigma_i$. Deleting the terms in (2.3) for which $c_i = 0$ and renumbering the remaining terms accordingly, it follows that the Krylov vectors y_k, $k \ge 1$, are linear combinations of $p \le s$ right singular vectors v_i, $i = 1\!:\!p$. Therefore, the grade can be at most p. Further, $(y_1, y_2, \dots, y_p) = (z_1, z_2, \dots, z_p)W$, where

$$W = \begin{pmatrix} 1 & \sigma_1^2 & \cdots & \sigma_1^{2(p-1)} \\ 1 & \sigma_2^2 & \cdots & \sigma_2^{2(p-1)} \\ \vdots & \vdots & \cdots & \vdots \\ 1 & \sigma_p^2 & \cdots & \sigma_p^{2(p-1)} \end{pmatrix} \in \mathbf{R}^{p\times p}, \tag{2.4}$$

and $z_i = c_i\sigma_iv_i$, $i = 1\!:\!p$, are scaled right singular vectors. Since $\sigma_i \ne \sigma_j$, $i \ne j$, the Vandermonde matrix W is nonsingular. It follows that the Krylov vectors are linearly independent for $k \le p$ and the grade is p. □

The pseudoinverse solution is obtained by setting $k = 0$ in (2.3):

$$x^\dagger = \sum_{i=1}^{s} c_i\sigma_i^{-1}v_i. \tag{2.5}$$

Comparing the expansions (2.5) and (2.3), it follows that $x^\dagger \in \mathcal{K}_p(A^TA, A^Tb)$, where p is the grade of A^Tb with respect to A^TA. Hence, if carried out to completion, the PLS approximations terminate with the pseudoinverse solution of (1.1). This is true without any assumptions about b and the size or rank of A.

3. Householder bidiagonalization and PLS. In a seminal paper, Golub and Kahan [12] gave a direct algorithm for the bidiagonal reduction of an arbitrary rectangular matrix $A \in \mathbf{R}^{m\times n}$:

$$U^TAV = \begin{pmatrix} B & 0 \\ 0 & 0 \end{pmatrix} \in \mathbf{R}^{m\times n}. \tag{3.1}$$

Here B is upper bidiagonal and $U \in \mathbf{R}^{m\times m}$, $V \in \mathbf{R}^{n\times n}$ are chosen as products of Householder matrices

$$U = G_1G_2\cdots G_{m-1}, \qquad V = H_1H_2\cdots H_{n-1}. \tag{3.2}$$

Householder matrices are elementary orthogonal reflectors

$$H = I - 2ww^T, \quad \|w\|_2 = 1, \tag{3.3}$$

and satisfy $H = H^{-1} = H^T$. Premultiplication by a Householder matrix is frequently used to zero out a sequence of entries in a given column vector. The matrix H does not need to be explicitly formed and only the Householder vector w needs to be stored.

In the bidiagonalization algorithm H_1 can be chosen arbitrarily, but as long as no zero element occurs in B, the remaining transformations are uniquely determined. In the PLS algorithm H_1 is taken to be the (essentially) unique Householder matrix for which

$$H_1(A^Tb) = \theta_1 e_1 \in \mathbf{R}^n, \quad \theta_1 \neq 0. \tag{3.4}$$

(Here e_1 denotes the first column of a unit matrix of appropriate dimension.) It follows that $\theta_1(H_1e_1) = A^Tb$. In the algorithm A is multiplied alternately from left and right by Householder transformations. Multiplication of A from the right by H_1 zeros nothing. Next G_1 is chosen to zero the last $m-1$ elements in the first column of AH_1 and H_2 is chosen to zero the last $n-2$ elements in the first row of G_1AH_1. This process can be continued until all rows and columns have been reduced and a bidiagonal matrix remains. We remark that from (3.4) it follows that the process reduces the augmented matrix $\binom{b^TA}{A}$ to *lower* bidiagonal form.

We now investigate how the PLS approximations can be obtained from the bidiagonal reduction. After applying G_k, the first k *columns* of A are reduced to upper bidiagonal form:

$$G_k \cdots G_2G_1AV_k = \begin{pmatrix} B_k \\ 0 \end{pmatrix}, \quad V_k = H_1H_2\cdots H_k \begin{pmatrix} I_k \\ 0 \end{pmatrix}, \tag{3.5}$$

where

$$B_k = \begin{pmatrix} \rho_1 & \theta_2 & & & \\ & \rho_2 & \theta_3 & & \\ & & \ddots & \ddots & \\ & & & \rho_{k-1} & \theta_k \\ & & & & \rho_k \end{pmatrix} \in \mathbf{R}^{k\times k} \tag{3.6}$$

is a leading principal submatrix of the final bidiagonal matrix B. After applying H_{k+1}, the first k *rows* in the decomposition of A are reduced to upper bidiagonal form:

$$U_k^T A\, H_1H_2\cdots H_{k+1} = \begin{pmatrix} \widehat{B}_k & 0 \end{pmatrix}, \quad U_k = G_k\cdots G_2G_1 \begin{pmatrix} I_k \\ 0 \end{pmatrix}, \tag{3.7}$$

where $\widehat{B}_k = \begin{pmatrix} B_k & \theta_{k+1}e_k \end{pmatrix} \in \mathbf{R}^{k\times(k+1)}$. From (3.5) and the transpose of (3.7) we get the two fundamental relations

$$AV_k = U_kB_k, \tag{3.8}$$

$$A^TU_k = V_{k+1}\widehat{B}_k^T = V_kB_k^T + \theta_{k+1}v_{k+1}e_k^T. \tag{3.9}$$

Equating the kth columns in (3.8)–(3.9) yields

$$\theta_kv_k = A^Tu_{k-1} - \rho_{k-1}v_{k-1}, \tag{3.10}$$

$$\rho_ku_k = Av_k - \theta_ku_{k-1}. \tag{3.11}$$

Starting with $\theta_1 v_1 = A^T b$ and $\rho_1 u_1 = A v_1$, these recurrence relations can be used to compute v_k and u_k for $k \geq 2$. The bidiagonal elements θ_k and and ρ_k are obtained from the normalization conditions $\|v_k\|_2 = \|u_k\|_2 = 1$. This *Lanczos-type* process for computing the upper bidiagonal decomposition is also due to Golub and Kahan [12]. Using the recursions (3.10)–(3.11) the following properties of V_k and U_k can be proved.

THEOREM 3.1. *Provided that no bidiagonal element in B_k is zero the matrices $U_k = (u_1, \dots, u_k)$ and $V_k = (v_1, \dots, v_k)$ are the unique orthonormal bases for the Krylov subspaces*

$$\mathcal{R}(V_k) = \mathcal{K}_k(A^T A, A^T b), \quad \mathcal{R}(U_k) = \mathcal{K}_k(AA^T, AA^T b), \quad k = 1, 2, \dots. \tag{3.12}$$

Proof. Since $\theta_1 v_1 = A^T b$ and $\rho_1 u_1 = A v_1 = AA^T b/\theta_1$ the theorem is true for $k = 1$. The proof proceeds by induction in k. Using (3.10) and the induction hypothesis it follows that

$$\theta_{k+1} v_{k+1} \in \left(A^T \mathcal{K}_k(AA^T, AA^T b)\right) \cup \mathcal{K}_k(A^T A, A^T b) = \mathcal{K}_{k+1}(A^T A, A^T b).$$

Similarly, using (3.11) and the induction hypothesis

$$\rho_{k+1} u_{k+1} \in \left(A \mathcal{K}_{k+1}(A^T A, A^T b)\right) \cup \mathcal{K}_k(AA^T, AA^T b) = \mathcal{K}_{k+1}(AA^T, AA^T b),$$

which concludes the induction. The matrices U_k and V_k are orthonormal by construction. The bases can also be obtained by computing the QR decomposition of the corresponding Krylov matrices. The uniqueness of the bases is a consequence of the uniqueness (up to a diagonal scaling with elements ±1) of the QR factorization of the full-rank Krylov matrix. □

From (3.12) it follows that the PLS approximation x_k can be expressed as

$$x_k = V_k y_k = H_1 \cdots H_k \begin{pmatrix} y_k \\ 0 \end{pmatrix} \in \mathcal{R}(V_k). \tag{3.13}$$

We now seek y_k so that $\|Ax_k - b\|_2$ is minimized. From the orthogonal invariance of the 2-norm

$$\|Ax_k - b\|_2^2 = \|G_k \cdots G_1 \left(AV_k y_k - b\right)\|_2^2 = \|B_k y_k - c_k\|_2^2 + \|d_k\|_2^2, \tag{3.14}$$

where c_k is obtained from

$$G_k \cdots G_1 b = \begin{pmatrix} c_k \\ d_k \end{pmatrix} = G_k \begin{pmatrix} c_{k-1} \\ d_{k-1} \end{pmatrix}. \tag{3.15}$$

Note that G_k only acts on the vector $d_{k-1} \in \mathbf{R}^{(n-k+1)}$. Hence, the minimum residual norm $\gamma_k = \|d_k\|_2$ is obtained when y_k solves the upper bidiagonal system $B_k y_k = c_k$. Assuming that B_k is nonsingular, y_k can be computed by back substitution:

$$y_k = c_k/\rho_k, \quad y_i = (c_i - \theta_{i+1} y_{i+1})/\rho_i, \quad i = k-1, \dots, 2, 1. \tag{3.16}$$

The maximum dimension p of the Krylov subspace $\mathcal{K}_k(A^T A, A^T b)$ is determined by the bidiagonalization process.

THEOREM 3.2. *In the PLS algorithm the bidiagonalization process terminates with $\rho_p \neq 0$ and $\theta_{p+1} = 0$, where p is the smallest integer for which*

$$\mathcal{K}_{p+1}(A^T A, A^T b) = \mathcal{K}_p(A^T A, A^T b),$$

i.e., when the grade of A^Tb with respect to A^TA is p. Then $x_p = V_py_p$ is the pseudoinverse solution to the least squares problem $\min_x \|Ax - b\|_2$.

Proof. This result follows from the uniqueness of bidiagonal reduction. □

At termination the PLS method has extracted from $Ax = b$ an equivalent bidiagonal system $B_py_p = c_p$ with nonzero bidiagonal elements. Hence B_p is nonsingular. We now show that its singular values are simple.

Lemma 3.3. *Assume that B_p has nonzero bidiagonal elements. Then all its singular values are simple.*

Proof. The singular values of B_p are the positive square roots of the eigenvalues of the symmetric tridiagonal matrix $T_p = B_p^TB_p$ with off-diagonal elements $\rho_k\theta_{k+1}$. Hence, the matrix T_p is unreduced, i.e., all its off-diagonal elements are nonzero, if and only if B_p has nonzero bidiagonal elements. The lemma now follows from the result that an unreduced symmetric tridiagonal matrix has simple eigenvalues (Parlett [19, Lemma 7.7.1]). □

It follows that $B_py_p = c_p$ is a minimally dimensioned *core problem* of $Ax = b$ in the sense of Paige and Strakoš [18]. They obtained a core problem by considering a *lower* bidiagonal reduction of A, for which there are two possible final states. Hence, our approach of using an *upper* bidiagonal reduction of A is slightly simpler.

In the Householder PLS algorithm the matrices V_k and U_k are kept in product form and never explicitly formed. Since the Householder transformations are orthogonal by construction, there is no loss of orthogonality in floating point arithmetic to worry about. In step k about $8(m-k)(n-k)$ flops are required to apply the Householder transformations to A. The flop counts for the additional scalar products and back substitution are negligible in comparison. When $k \ll \min(m,n)$, about $8mnk$ flops are needed for generating k PLS factors.

4. The NIPALS PLS algorithm.

The NIPALS PLS algorithm is frequently used in chemometrics. It uses elementary orthogonal projections to explicitly generate the orthogonal basis vectors for the Krylov subspaces. We set $A_0 = A$, $b_0 = b$, and for $k = 1, 2, \dots$ we generate (in the terminology used in statistics) *score vectors* u_k and *loading weights* vectors v_k:

$$v_k = A_{k-1}^Tb_{k-1}/\mu_k, \quad \mu_k = \|A_{k-1}^Tb_{k-1}\|_2, \tag{4.1}$$

$$u_k = A_{k-1}v_k/\rho_k, \quad \rho_k = \|A_{k-1}v_k\|_2, \tag{4.2}$$

$$(A_k, b_k) = (I - u_ku_k^T)(A_{k-1}, b_{k-1}). \tag{4.3}$$

In (4.3) A_{k-1} and b_{k-1} are deflated by subtracting their orthogonal projections onto u_k. This operation uses elementary orthogonal projections,

$$P = I - uu^T, \quad \|u\|_2 = 1, \tag{4.4}$$

for which $P = P^T$, $P^2 = P$. The deflation in (4.3) can also be written as

$$A_k = A_{k-1} - u_kp_k^T, \quad p_k = A_{k-1}^Tu_k, \tag{4.5}$$

$$b_k = b_{k-1} - u_k\zeta_k, \quad \zeta_k = b_{k-1}^Tu_k, \tag{4.6}$$

where p_k are *loading vectors*. The process is terminated when either $\|A_{k-1}^Tb_{k-1}\|_2$ or $\|A_{k-1}v_i\|_2$ is zero. We note that if $u_k^TA_{k-1}v_k \neq 0$, then the rank of the matrix A_k is exactly one less than that of A_{k-1}. Thus, (4.5) is a special case of a rank-reduction formula due to Wedderburn [25], further discussed in Chu, Funderlic, and Golub [8].

The mathematical equivalence of the two PLS algorithms follows from the following result.

THEOREM 4.1. *Using exact arithmetic, the vectors* $\{v_1, \dots, v_k\}$ *and* $\{u_1, \dots, u_k\}$ *generated by* (4.1)–(4.3) *are the unique orthogonal bases for the Krylov subspaces* $\mathcal{K}_k(A^TA, A^Tb)$ *and* $\mathcal{K}_k(AA^T, AA^Tb)$*, respectively.*

Proof. See Eldén [10, Proposition 3.1]. □

The Householder and NIPALS PLS algorithms generate orthonormal bases U_k and V_k for the same sequences of Krylov subspaces. The mathematical equivalence of these two PLS algorithms follows from the uniqueness of these bases. In particular, in exact arithmetic, the matrices U_k and V_k generated by NIPALS PLS satisfy relations (3.8) and (3.9). Summing (4.5) and (4.6) gives

$$A = U_kP_k^T + A_k, \qquad b = U_kz_k + b_k, \tag{4.7}$$

where $U_k = (u_1, \dots, u_k)$, $P_k = (p_1, \dots, p_k)$, and $z_k = (\zeta_1, \dots, \zeta_k)^T$. These relations hold to working accuracy and do not rely on orthogonality. The matrix $U_kP_k^T$ is a rank-k approximation to the data matrix A.

From (4.7) we obtain for the residual $r_k = b - AV_ky_k = r_{k,1} + r_{k,2}$ the expression

$$r_{k,1} = U_k(z_k - P_k^TV_ky_k), \qquad r_{k,2} = b_k - A_kV_ky_k. \tag{4.8}$$

Here $r_{k,1} \in \mathcal{R}(U_k)$ vanishes if y_k satisfies

$$(P_k^TV_k)y_k = z_k. \tag{4.9}$$

Assuming the orthogonality of U_k and using (3.8), we get

$$A_kV_k = (I - u_ku_k^T)\cdots(I - u_1u_1^T)AV_k = (I - u_ku_k^T)\cdots(I - u_1u_1^T)U_kB_k = 0.$$

Hence, $r_{k,2} = b_k \perp \mathcal{R}(U_k)$ and independent of y_k. Further, since

$$p_k^T = u_k^TA_{k-1} = u_k^T(I - u_{k-1}u_{k-1}^T)\cdots(I - u_1u_1^T)A = u_k^TA,$$

we have $P_k^T = U_k^TA$. From (3.9) and the orthogonality of V_k, we obtain

$$P_k^TV_k = U_k^TAV_k = \widehat{B}_kV_{k+1}^TV_k = B_k.$$

Thus, in exact arithmetic the matrix $P_k^TV_k$ in (4.9) is upper bidiagonal with elements

$$\theta_k = p_{k-1}^Tv_k, \qquad \rho_k = p_k^Tv_k = \|A_{k-1}v_k\|_2. \tag{4.10}$$

The solution y_k to (4.9) can be computed by back substitution; see (3.16).

Although the Householder and MGS algorithms compute the same approximate solutions x_k they differ slightly in the way the residual of the data matrix is approximated. In NIPALS PLS the data residuals A_k are given by

$$A_k = A - U_kP_k^T = (I - U_kU_k^T)A. \tag{4.11}$$

For the Householder version the residual is obtained from the rank-k approximations $A \approx U_kB_kV_k^T$. Using $U_kB_k = AV_k$ we obtain the data residual

$$E_k = A - (U_kB_k)V_k^T = A - (AV_k)V_k^T = A(I - V_kV_k^T). \tag{4.12}$$

It was pointed out by Pell, Ramos, and Manne [20] that these two expressions for the data residuals are not the same. Bro and Eldén [6] argue that although both expressions are valid, the residual A_k given by (4.11) is better suited to detect model outliers and should therefore be preferred. However, the residual A_k can be computed also by the Householder algorithm as follows. From (3.9), we have

$$A_k = A - U_k(U_k^T A) = A - U_k \widehat{B}_k V_{k+1}^T, \tag{4.13}$$

where $\widehat{B}_k = \begin{pmatrix} B_k & \theta_{k+1} e_k \end{pmatrix}$. Comparing (4.13) and (4.11) it follows that

$$P_k = V_{k+1} \widehat{B}_k^T = H_1 \cdots H_k H_{k+1} \widehat{B}_k^T. \tag{4.14}$$

If the bidiagonalization is continued so that $\theta_{k+1} e_k$ and v_{k+1} are available, then P_k can be computed from (4.14).

The NIPALS PLS algorithm uses three matrix-vector products and one rank-one deflation, which together require $8mn$ flops per PLS factor. The flop counts for the additional scalar products and final back substitution are negligible in comparison. This is the same number of flops per step as required by the Householder algorithm as long as the number of factors $k \ll \min(m, n)$. The computation of U_k and P_k in the Householder PLS algorithm costs an extra $2(m+n)k^2$ flops.

5. Deflation in NIPALS PLS. In exact arithmetic the orthogonality of the vectors $(u_1, \ldots, u_k)$ implies that

$$u_k^T b = u_k^T \left(b - \sum_{j=1}^{k-1} \zeta_j u_j \right), \quad \zeta_j = u_j^T b_{j-1}. \tag{5.1}$$

Hence, in exact arithmetic the deflation of b can be omitted. However, in floating-point computation the equality in (5.1) does not hold because of a loss of orthogonality. In this section we make an empirical study of the consequences of omitting the deflation of b in NIPALS PLS. This is further discussed in section 6.

In the appendix a MATLAB implementation of NIPALS PLS is given, which incorporates deflation of b and treats the matrix $P_k^T V_k$ as a bidiagonal matrix. To study the loss of orthogonality in V_k and U_k, this was applied to a problem where $A \in \mathbf{R}^{50\times 8}$ is a matrix with singular values $\sigma_i = 10^{-i+1}$, $i = 1:8$. The right-hand side was chosen as $b = Ae$, where $e = (1, \ldots, 1)^T$. Table 5.1 shows the condition number $\kappa_k = \kappa(P_k^T V_k) = \kappa(P_k)$ and the loss of orthogonality in U_k and V_k measured by $\|I_k - U_k^T U_k\|_2$ and $\|I_k - V_k^T V_k\|_2$. Clearly the loss of orthogonality is proportional to κ_k in both U and V. The norm of the error in the computed solution for $k = 8$ is $1.149 \cdot 10^{-10}$. This is of the same magnitude as the loss of orthogonality in V_k and U_k. The corresponding error norm for the Householder algorithm is $2.181 \cdot 10^{-10}$. This strongly suggests that the correctly implemented NIPALS PLS algorithm is forward stable. The columns to the right in Table 5.1 show the effect of omitting the deflation of b. Although the loss of orthogonality in U_k is nearly unchanged, the loss of orthogonality in V_k now is proportional to κ_k^2. The norm of the error in the computed solution, $0.724 \cdot 10^{-1}$, is of the same magnitude. *We conclude that omitting the deflation of b will destroy the very good numerical stability of the NIPALS PLS algorithm with only a minor reduction in work.* It should be noted, however, that if double precision arithmetic is used, this gives a large safety margin and in practice a lack of stability may not be noticed unless the problem is ill-conditioned.

In view of these results, it is unfortunate that in current practice omitting the deflation of b seems to be the norm rather than an exception. It was recommended by

Table 1

Condition number of P_k and loss of orthogonality $\gamma(V_k) = \|I_k - V_k^T V_k\|_2$ and $\gamma(U_k) = \|I_k - U_k^T U_k\|_2$ in NIPALS PLS; left: with deflation of b; right: without deflation.

k	$\kappa(P_k^T V_k)$	$\gamma(U_k)$	$\gamma(V_k)$	$\gamma(U_k)$	$\gamma(V_k)$
1	$1.000 \cdot 10^{0}$	$6.661 \cdot 10^{-16}$	$2.222 \cdot 10^{-16}$	$6.661 \cdot 10^{-16}$	0
2	$1.000 \cdot 10^{1}$	$1.256 \cdot 10^{-15}$	$2.222 \cdot 10^{-16}$	$1.254 \cdot 10^{-15}$	$7.200 \cdot 10^{-14}$
3	$1.000 \cdot 10^{2}$	$1.258 \cdot 10^{-15}$	$5.562 \cdot 10^{-15}$	$1.255 \cdot 10^{-15}$	$7.187 \cdot 10^{-12}$
4	$1.000 \cdot 10^{3}$	$2.746 \cdot 10^{-14}$	$4.576 \cdot 10^{-14}$	$2.772 \cdot 10^{-14}$	$7.186 \cdot 10^{-10}$
5	$1.000 \cdot 10^{4}$	$2.746 \cdot 10^{-14}$	$2.871 \cdot 10^{-13}$	$2.772 \cdot 10^{-14}$	$7.186 \cdot 10^{-8}$
6	$1.000 \cdot 10^{5}$	$1.667 \cdot 10^{-13}$	$1.024 \cdot 10^{-12}$	$1.674 \cdot 10^{-13}$	$7.186 \cdot 10^{-6}$
7	$1.000 \cdot 10^{6}$	$1.775 \cdot 10^{-13}$	$8.975 \cdot 10^{-12}$	$5.172 \cdot 10^{-13}$	$7.186 \cdot 10^{-4}$
8	$1.000 \cdot 10^{7}$	$6.000 \cdot 10^{-11}$	$6.541 \cdot 10^{-11}$	$5.158 \cdot 10^{-10}$	$7.167 \cdot 10^{-2}$

Manne [14, p. 7] and mentioned by Wold, Sjöström, and Eriksson [29, p. 114]. Dayal and MacGregor [9] proposed faster PLS algorithms which that omit the deflation of b, as do the implementations tested by Andersson [1]. Eldén [10, p. 17] correctly writes that the deflation "is superfluous, at least in exact arithmetic." This practice has spread to several commercial statistical software packages, such as the pls package in R.

Although the matrix $P_k^T V_k$ is upper bidiagonal in exact arithmetic, many implementations of the NIPALS PLS algorithm solve the linear system (4.9) treating $P_k^T V_k$ as a full matrix, which increases the arithmetic cost of solving the subproblem from $2k$ flops to $2k^3/3$ flops. A third possibility is to use only the upper triangular part of the computed matrix $P_k^T V_k$, which gives an operation count of $2k^2$ flops. The table below shows the norm of the error in x_8 for these options, with and without deflation.

Deflation	Full	Triangular	Bidiagonal
yes	$2.28 \cdot 10^{-11}$	$9.43 \cdot 10^{-11}$	$1.15 \cdot 10^{-10}$
no	$5.18 \cdot 10^{-3}$	$5.18 \cdot 10^{-3}$	$7.24 \cdot 10^{-2}$

Without deflation, treating the matrix as full or upper triangular gives almost a factor of 10 better accuracy, compared to using only the bidiagonal part. With deflation of b, the best accuracy is also achieved by using the full matrix, but the differences are much smaller.

6. Stability analysis. It is well known that algorithms based on a sequence of orthogonal transformations with Householder matrices (3.3) have very good stability properties; see Higham [13, section 19.3]. Wilkinson [26] showed that the computation of a Householder vector and the application of a Householder matrix to a given matrix are both normwise backward stable.

Byers and Xu [7] use the Golub–Kahan Householder bidiagonalization algorithm for computing the pseudoinverse $A^\dagger$. They compute $A = UBV^H$ and then solve $BY = U^H$ by back substitution, giving $A^\dagger = VY$. Their proof that this algorithm is mixed forward-backward stable applies also to the Householder PLS algorithm. Similar results are shown in Smoktunowicz and Wróbel [23].

Theorem 6.1 (Byers and Xu [7, Theorem A2]). *Let $\bar{U}\bar{B}\bar{V}^T$ be the computed bidiagonal factorization of A. Then $\bar{U} = U + \Delta U$ and $\bar{V} = V + \Delta V$, where U and V are orthogonal and $\|\Delta U\|_2 \le d_1\mathbf{u}$, $\|\Delta V\|_2 \le d_2\mathbf{u}$. Further,*

$$\bar{B} = U^T(A+E)V, \quad \|E\|_2 \le d_3\mathbf{u}\|A\|_2.$$

Here $d_i(m,n)$ are modestly sized functions and $\mathbf{u}$ the unit roundoff.

The computed vector $G_k \cdots G_1 b$ in (3.15) is the exact result corresponding to a slightly perturbed right-hand side $b + \delta b$, where

$$\|\delta b\|_2 \le d_4\mathbf{u}\|b\|_2.$$

Here and later, for a vector $x = (x_i)$, $|x|$ denotes the vector with elements $|x_i|$. For the bidiagonal back substitution the following componentwise error bound holds.

THEOREM 6.2 (Byers and Xu [7, Lemma A1]). *Consider the system $By = c$, where $B \in \mathbf{R}^{n\times n}$ is nonsingular and upper bidiagonal. Let $\bar{y}$ be the solution computed by back substitution. Then, neglecting terms of order $\mathbf{u}^2$,*

$$\bar{y} = B^{-1}(c + \delta c) + \delta y,$$

where the elementwise inequalities $|\delta c| \le 3n\mathbf{u}|c|$ and $|\delta y| \le 3n\mathbf{u}|y|$ hold.

The solution x_k is computed from (3.13). Hence, the computed result $\bar{x}_k$ is the exact result of

$$H_1 H_2 \cdots H_k \begin{pmatrix} \bar{y}_k \\ 0 \end{pmatrix}$$

plus an error whose norm is bounded by $d_5\mathbf{u}\|\bar{x}_k\|_2$. Together these results prove mixed forward-backward stability for Householder PLS.

The loss of orthogonality makes the error analysis of the MGS PLS algorithm considerably more difficult than that for the Householder PLS algorithm, where the vectors u_k and v_k are orthogonal by construction. Omitting the deflation of the right-hand side b when using the MGS QR factorization to solve a linear least squares problem causes a loss of stability similar to that in NIPALS PLS; see [4]. Both algorithms use a sequence of elementary orthogonal projections (4.4). MGS is initialized by setting $A = (a_1, \dots, a_n) = A^{(0)} \in \mathbf{R}^{m\times n}$, and $q_1 = a_1/\|a_1\|_2$. At the start of the kth step, we have computed

$$A^{(k)} = \left(q_1, \dots, q_k, a_{k+1}^{(k-1)}, \dots, a_n^{(k-1)}\right),$$

where $(q_1, \dots, q_k)$ are orthonormal vectors. The remaining $n-k$ columns, which are already orthogonal to q_j, $j = 1\!:\!k-1$, are now orthogonalized to q_k,

$$a_j^{(k)} = (I - q_k q_k^T)a_j^{(k-1)} = a_j^{(k-1)} - r_{kj}q_k, \quad r_{kj} = q_k^T a_j^{(k-1)}. \tag{6.1}$$

The right-hand side is deflated similarly setting $b = b_0$, and

$$b^{(k)} = (I - q_k q_k^T)b^{(k-1)} = b^{(k-1)} - \zeta_k q_k, \quad \zeta_k = q_k^T b^{(k-1)}. \tag{6.2}$$

There will be a loss of orthogonality in the computed matrix $\bar{Q}_k = (\bar{q}_1, \dots, \bar{q}_k)$ in MGS due to cancellation occurring when subtracting the orthogonal projections in (6.1). Since cancellation is related to the ill-conditioning of A, the loss of orthogonality can be bounded by

$$\|I - \bar{Q}_n^T\bar{Q}_n\|_2 \le c_1\mathbf{u}\kappa(A), \tag{6.3}$$

where $c_1 = c_1(m,n)$ is of modest size; see Björck [3]. This result was used in [3] to prove forward stability of MGS for computing the QR factorization and solving the linear least squares problem. Omitting the deflation in (6.2) and setting $\zeta_k = q_k^T b$ will introduce an error in the computed solution proportional to $\kappa(A)^2$.

In 1967 Charles Sheffield[1] noticed that the MGS QR factorization of $A \in \mathbf{R}^{m\times n}$ is *numerically* equivalent to Householder QR factorization of the matrix A augmented with a square matrix of zeros on top. If $H_k = I - w_k w_k^T$ is the Householder transformation generated by $w_k = \binom{e_k}{q_k}$, then computing (6.2) is numerically equivalent to

$$\begin{pmatrix} z_k \\ 0_{n-k} \\ b^{(k)} \end{pmatrix} = H_k \cdots H_1 \begin{pmatrix} 0 \\ b \end{pmatrix}, \quad z_k = \begin{pmatrix} \zeta_1 \\ \vdots \\ \zeta_k \end{pmatrix}. \tag{6.4}$$

Björck and Paige [5] used Sheffield's observation to prove backward stability of the MGS QR factorization. Further implications of Sheffield's observation are made in Paige [16]. Using this equivalence, if we let $W_k = H_1 \cdots H_k$, then it follows that the NIPALS PLS algorithm implicitly does the factorization

$$\begin{pmatrix} 0 \\ AV_k \end{pmatrix} = W_k \begin{pmatrix} B_k \\ 0 \end{pmatrix}.$$

This would be a natural starting point for a proof of the stability of the NIPALS PLS algorithm.

The elementary reflections or orthogonal projections used in the Householder and NIPALS algorithms will destroy any sparsity structure in the matrix A. Therefore, the use of the Paige and Saunders LSQR algorithm can be an attractive alternative for large-scale sparse problems. Each step in LSQR requires two matrix-vector multiplications Ax and A^Ty plus some vector operations. If A is sparse or otherwise structured the matrix-vector multiplications may be computed in much less than $2mn$ flops. It is well known (see the seminal paper by Paige [15]) that Lanczos-type algorithms like LSQR may suffer from a severe loss of orthogonality in the computed vectors. This may cause LSQR to require significantly more iterations, unless the computed vectors u_k or v_k are reorthogonalized. In full reorthogonalization, u_k and v_k are reorthogonalized against all previous vectors $u_1, \dots, u_{k-1}$ and $v_1, \dots, v_{k-1}$ as soon as they have been computed. This adds an arithmetic cost of about $4(m+n)k^2$ flops for k factors, which is affordable if $k \ll \min\{m, n\}$. Barlow [2] has shown that one-sided reorthogonalization suffices to prove backward stability. Selective reorthogonalization is less costly but is more complicated to implement. An interesting scheme, which ensures semiorthogonality in u_k and v_k, is developed by Simon and Zha [22].

7. Conclusions.

We have drawn attention to a widespread but unfortunate and unnecessary "simplification" of the original NIPALS PLS algorithm. This consists of omitting the deflation of the right-hand side b, which destroys its otherwise excellent numerical stability. In an extensive experimental test Andersson [1] compares the accuracy and efficiency of no fewer than nine different PLS algorithms. In this the NIPALS PLS algorithm was implemented without deflation of b. Despite this, it was found together with three other algorithms to be the most stable. We conclude that none of the algorithms tested in that paper is numerically stable in our sense.

To ensure computational accuracy and make results reproducible, we recommend that either the Householder or the original NIPALS PLS algorithm be used as the

[1] Personal communication in 1967 relayed to the author by G. H. Golub.

standard algorithm for PLS regression. For the Householder PLS algorithm mixed forward-backward stability can be proved. Our conjecture is that, correctly implemented, the NIPALS PLS algorithm is also mixed forward-backward stable. A strict proof of this conjecture seems difficult and remains an open problem. In our numerical tests these two algorithms always gave results of similar accuracy. For computing a small number of factors, the arithmetic and storage costs are roughly similar for both. If A is large and sparse or otherwise structured, the LSQR algorithm with reorthogonalization is an attractive alternative. The implementation of such an algorithm is left as a future project.

Appendix. The MATLAB function mgspls below was used for the numerical tests of the NIPALS PLS algorithm in section 5. It computes at most $k = q$ PLS factors $u_k p_k^T$ for the problem $\min_x \|Ax - b\|_2$, such that $A^T b \neq 0$, and performs deflation of both A and b. Only the bidiagonal elements in $P_k^T V_k$ are used. Output is x_q, the residual vector $r_q = b - Ax_q$, and the matrices U_q and P_q. Note that this program is written for testing purposes only and practical stopping criteria are not included.

```
function [x,U,V,P] = plsr3(A,b,q)
% MGSPLS  Performs at most q \le rank(A) steps of
% the NIPALS PLS algorithm for min_x||A x - b||_2.
[m,n] = size(A); tol = norm(A,1)*eps;
for k = 1:q
   v = A'*b;  nv = norm(v),
   if nv < tol, q = k-1; break; end
   v = v/nv;  V(:,k) = v;
   if k > 1, theta(k) = p'*v; end
   u = A*v;  rho(k) = norm(u);
   u = u/rho(k);  U(:,k) = u;
   p = A'*u;  P(:,k) = p;
   z(k) = b'*u;
% Deflate A and b.
   A = A - u*p'; b = b - u*z(k);
end
% Solve bidiagonal subsystem.
y(q) = z(q)/rho(q);
for k = q-1:(-1):1
   y(k) = (z(k) - theta(k+1)*y(k+1))/rho(k);
end
x = V*y';
```

Acknowledgments. The author wishes to thank Rasmus Bro for sharing his insight into the practical use of PLS and for drawing attention to the comparison of algorithms in [1]. Stimulating discussions with Lars Eldén and Michael Saunders are also gratefully acknowledged. Finally, the author thanks the referees for their careful reading of the manuscript, which led to several improvements in the presentation.

REFERENCES

[1] M. Andersson, *A comparison of nine PLS1 algorithms*, J. Chemometrics, 23 (2009), pp. 518–529.

[2] J. L. Barlow, *Reorthogonalization for the Golub–Kahan–Lanczos bidiagonal reduction*, Numer. Math., 124 (2013), pp. 237–278.

[3] Å. Björck, *Solving linear least squares problems by Gram–Schmidt orthogonalization*, BIT, 7 (1967), pp. 1–21.

[4] Å. Björck, *Numerics of Gram–Schmidt orthogonalization*, Linear Algebra Appl., 197–198 (1994), pp. 297–316.

[5] Å. Björck and C. C. Paige, *Loss and recapture of orthogonality in the modified Gram–Schmidt algorithm*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 176–190.

[6] R. Bro and L. Eldén, *PLS works*, J. Chemometrics, 23 (2009), pp. 69–71.

[7] R. Byers and H. Xu, *A new scaling for Newton's iteration for the polar decomposition and its backward stability*, SIAM J. Matrix. Anal. Appl., 30 (2008), pp. 822–843.

[8] M. T. Chu, R. E. Funderlic, and G. H. Golub, *A rank-one reduction formula and its application to matrix factorization*, SIAM Rev., 37 (1995), pp. 512–530.

[9] B. S. Dayal and J. F. MacGregor, *Improved PLS algorithms*, J. Chemometrics, 11 (1997), pp. 73–85.

[10] L. Eldén, *Partial least-squares vs. Lanczos bidiagonalization*—I: *Analysis of a projection method for multiple regression*, Comput. Statist. Data Anal., 46 (2004), pp. 11–31.

[11] I. E. Frank and J. H. Friedman, *A statistical view of some chemometrics regression tools*, Technometrics, 35 (1993), pp. 109–135.

[12] G. H. Golub and W. Kahan, *Calculating the singular values and pseudoinverse of a matrix*, SIAM J. Numer. Anal. Ser. B, 2 (1965), pp. 205–224.

[13] N. J. Higham, *Accuracy and Stability of Numerical Algorithms*, 2nd ed., SIAM, Philadelphia, 2002.

[14] R. Manne, *Analysis of two partial-least-squares algorithms for multivariate calibration*, Chemom. Intell. Lab. Syst., 2 (1987), pp. 187–197.

[15] C. C. Paige, *Error analysis of the Lanczos algorithm for tridiagonalizing a symmetric matrix*, J. Inst. Math. Appl., 18 (1976), pp. 341–349.

[16] C. C. Paige, *A useful form of a unitary matrix obtained from any sequence of unit* 2*-norm n-vectors*, SIAM J. Matrix Anal. Appl., 31 (2009), pp. 565–583.

[17] C. C. Paige and M. A. Saunders, *LSQR: An algorithm for sparse linear equations and sparse least squares*, ACM Trans. Math. Softw., 8 (1982), pp. 43–71.

[18] C. C. Paige and Z. Strakoš, *Core problems in linear algebraic systems*, SIAM J. Matrix. Anal. Appl., 27 (2006), pp. 861–875.

[19] B. N. Parlett, *The Symmetric Eigenvalue Problem*, Classics in Appl. Math. 20, SIAM, Philadelphia, 1998.

[20] R. J. Pell, L. S. Ramos, and R. Manne, *The model space in partial-least-squares regression*, J. Chemometrics, 21 (2007), pp. 165–172.

[21] A. Phatak and F. de Hoog, *Exploiting the connection between PLS, Lanczos methods, and conjugate gradients: Alternative proofs of some properties of PLS*, J. Chemometrics, 16 (2002), pp. 361–367.

[22] H. D. Simon and H. Zha, *Low-rank matrix approximation using the Lanczos bidiagonalization process with applications*, SIAM J. Sci. Comput., 21 (2000), pp. 2257–2274.

[23] A. Smoktunowicz and I. Wróbel, *Numerical aspects of computing the Moore-Penrose inverse of full column rank matrices*, BIT, 52 (2012), pp. 503–524.

[24] E. Stiefel, *Ausgleichung ohne Aufstellung der Gausschen Normalgleichungen*, Wiss. Z. Tech. Hochsch. Dresden, 2 (1952/53), pp. 441–442.

[25] J. H. M. Wedderburn, *Lectures on Matrices*, Dover, Mineola, NY, 1964.

[26] J. H. Wilkinson, *Error analysis of transformations based on the use of matrices of the form* $\mathrm{I} - 2ww^{\mathrm{H}}$, in Error in Digital Computation, vol. 2, L. B. Rall, ed., Wiley, New York, 1965, pp. 77–101.

[27] H. Wold, *Estimation of principal components and related models by iterative least squares*, in Multivariate Analysis, P. R. Krishnaiah, ed., Academic Press, New York, 1966, pp. 391–420.

[28] S. Wold, A. Ruhe, H. Wold, and W. J. Dunn, *The collinearity problem in linear regression. The partial least squares (PLS) approach to generalized inverses*, SIAM J. Sci. Statist. Comput., 5 (1984), pp. 735–743.

[29] S. Wold, M. Sjöström, and L. Eriksson, *PLS-regression: A basic tool of chemometrics*, Chemom. Intell. Lab. Syst., 58 (2001), pp. 109–130.