Nečas Center for Mathematical Modeling

A note on iterative refinement for seminormal equations

M. Rozložník, A. Smoktunowicz and J. Kopal

Preprint no. 2013-023

http://ncmm.karlin.mff.cuni.cz/

A note on iterative refinement for seminormal equations

Miroslav Rozložník[a,*], Alicja Smoktunowicz[b,*], Jiří Kopal[c]

[a] *Institute of Computer Science, Academy of Sciences of the Czech Republic, Pod vodárenskou věží 2, CZ-182 07 Prague 8, Czech Republic*
[b] *Faculty of Mathematics and Information Science, Warsaw University of Technology, Koszykowa 75, 00-662 Warsaw, Poland*
[c] *Technical University of Liberec, Department of Mathematics, Studentská 2, CZ-461 17 Liberec, Czech Republic*

Abstract

We present a roundoff error analysis of the method for solving the linear least squares problem $\min_x \|b - Ax\|_2$ with full column rank matrix A, using only factors Σ and V from the SVD decomposition of $A = U\Sigma V^T$. This method (called SNE_{SVD} here) is an analogue of the method of seminormal equations (SNE_{QR}), where the solution is computed from $R^T Rx = A^T b$ using only the factor R from the QR factorization of A. Such methods have practical applications when A is large and sparse and if one needs to solve least squares problems with the same matrix A and multiple right-hand sides. However, in general both SNE_{QR} and SNE_{SVD} are not forward stable. We analyze one step of fixed precision iterative refinement to improve the accuracy of the SNE_{SVD} method. We show that, under the condition $\mathcal{O}(u)\kappa^2(A) < 1$, this

*Corresponding author

**The work of M. Rozložník was supported by Grant Agency the Czech Republic under the project 108/11/0853. The work of J. Kopal was supported by the Ministry of Education of the Czech Republic under the SGS project no. 7822/115 at the Technical University of Liberec.

Email addresses: miro@cs.cas.cz (Miroslav Rozložník), A.Smoktunowicz@mini.pw.edu.pl (Alicja Smoktunowicz), jiri.kopal@tul.cz (Jiří Kopal)

method (called $CSNE_{SVD}$) produces a forward stable solution, where $\kappa(A)$ denotes the condition number of the matrix A and u is the unit roundoff. However, for problems with only $\mathcal{O}(u)\kappa(A) < 1$ it is generally not forward stable, and has similar numerical properties to the corresponding $CSNE_{QR}$ method. Our forward error bounds for the $CSNE_{SVD}$ are slightly better than for the $CSNE_{QR}$ since the terms $\mathcal{O}(u^2)\kappa^3(A)$ are not present. We illustrate our analysis by numerical experiments.
Keywords: Condition number; numerical stability; normal equations.
2000 MSC: 65F10, 65G50, 15A12

Dedicated to Paul Van Dooren on the occasion of his 60th Birthday

1. Introduction

We study the numerical properties of some correction methods for semi-normal equations for solving linear least squares problem

$$\min_x \|b - Ax\|, \tag{1}$$

where $A \in \mathbb{R}^{m\times n}$ has full column rank, $m \geq n = rank(A)$, $b \in \mathbb{R}^m$ and $\|\cdot\|$ denotes the Euclidean vector or matrix norm (depending on its argument). There exists only one solution $x_* \in \mathbb{R}^n$ to (1) that satisfies the normal equations

$$A^T A x_* = A^T b. \tag{2}$$

Therefore, $x_* = A^\dagger b$, where $A^\dagger = (A^T A)^{-1} A^T$ denotes the pseudoinverse of A. There are many algorithms for solving (1) using various factorizations of the matrix A. For example, if we apply the Householder QR decomposition

$A = QR$, where $Q \in \mathbb{R}^{m\times n}$ has orthonormal columns (i.e. $Q^TQ = I_n$) and $R \in \mathbb{R}^{n\times n}$ is nonsingular upper-triangular, then the normal equations can be written as $R^TRx_* = R^TQ^Tb$, so we can simply solve the system $Rx_* = Q^Tb$. If $m \gg n$ and we do not want to store the factor Q, we can use only the R-factor and solve the *seminormal equations* (SNE) (cf.[3])

$$R^TRx_* = A^Tb. \tag{3}$$

However, the numerical properties of these two methods are very different (cf. [3]–[6]). A similar approach can be applied to other factorizations of A; for example, if $A = U\Sigma V^T$, where $U \in \mathbb{R}^{m\times n}$ has orthonormal columns with $U^TU = I_n$, $V \in \mathbb{R}^{n\times n}$ is orthogonal and Σ has a simple structure (triangular, bidiagonal, diagonal), then we can consider the class of equations

$$\Sigma^T\Sigma(V^Tx_*) = V^T(A^Tb). \tag{4}$$

In this paper, we consider only diagonal Σ, where the system (4) can be solved very accurately. For the case when Σ is bidiagonal we refer to [7].

One of the goals of SNE in (3) is that we do not need Q, while in (4) we do not store the matrix U but we need matrix V. The dimension n is often much smaller than the other dimension m, so the solution of (4) can still be very efficient.

We study the numerical properties of algorithms SNE_{SVD} and $CSNE_{SVD}$, based on the SVD of A and described in Table 1. Our analysis is mostly motivated by Å. Björck's paper [3] and that of Å. Björck and C.C. Paige [6] (see also [1], [2], [4]). Throughout the paper we assume that the computed matrix $\tilde{\Sigma} = diag(\tilde{\sigma}_1, \ldots, \tilde{\sigma}_n)$ in floating point arithmetic is obtained by a backward

Algorithm I ($\mathbf{SNE_{QR}}$) Compute the upper-triangular factor $R \in \mathbb{R}^{n\times n}$ of Householder QR decomposition of A: $A = QR$, where $Q \in \mathbb{R}^{m\times n}$ and $Q^TQ = I_n$. Don't store Q. Solve the seminormal equations $R^TRx = A^Tb$ for x.
Algorithm II ($\mathbf{SNE_{SVD}}$) Find $V \in \mathbb{R}^{n\times n}$ and $\Sigma \in \mathbb{R}^{n\times n}$ of the SVD decomposition of A: $A = U\Sigma V^T$, where $U \in \mathbb{R}^{m\times n}$ and $U^TU = I_n$. Don't store U. Solve the seminormal equations $\Sigma^2(V^Tx) = V^T(A^Tb)$ for x.
Algorithm III ($\mathbf{CSNE_{QR}}$) Let x be computed by Algorithm I. Compute $r = b - Ax$. Solve $R^TR\Delta x = A^Tr$ for Δx. Update $x_{new} = x + \Delta x$.
Algorithm IV ($\mathbf{CSNE_{SVD}}$) Let x be computed by Algorithm II. Compute $r = b - Ax$. Solve $\Sigma^2(V^T\Delta x) = V^T(A^Tr)$ for Δx. Update $x_{new} = x + \Delta x$.

Table 1: Description of SNE and CSNE algorithms

stable algorithm. It means that there exist matrix $\hat{U} \in \mathbb{R}^{m\times n}$ with *exactly orthonormal* columns and *exactly orthogonal* matrix $\hat{V} \in \mathbb{R}^{n\times n}$ such that

$$\hat{A} = A + \Delta A = \hat{U}\tilde{\Sigma}\hat{V}^T, \quad \|\Delta A\| \leq \mathcal{O}(u)\,\|A\|, \tag{5}$$

where u is the unit roundoff. The computed matrix $\tilde{V}$ is close to $\hat{V}$, i.e.

$$\tilde{V} = \hat{V} + \Delta V, \quad \|\Delta V\| \leq \mathcal{O}(u). \tag{6}$$

We will not put any assumptions on the computed matrix $\tilde{U}$ here since its properties will not have any influence upon our analysis. It is interesting to notice that $\tilde{\Sigma}^T\tilde{\Sigma} = \hat{V}^T(\hat{A}^T\hat{A})\hat{V}$, hence our computed solution $\tilde{x}$ of $\Sigma^T\Sigma(V^Tx_*) = V^T(A^Tb)$ will often be related to the solution of the perturbed system of the normal equations $\hat{A}^T\hat{A}\hat{x} = \hat{A}^Tb$.

The paper is organized as follows. Section 2 is devoted to the sensitivity of the least squares problem and Section 3 the numerical stability of algorithms for computing the least squares solution. In Section 4 we study the numerical properties of the SNE_{SVD} method based on the SVD of A. In Section 5 we give a roundoff error analysis of the corrected seminormal equations $CSNE_{SVD}$. We show that the SNE_{SVD} method is not forward stable but that the $CSNE_{SVD}$ method, i.e. one step of iterative refinement, will usually be enough to yield a stable solution to (1). In Section 6 we present some numerical experiments in *MATLAB* to illustrate and compare the behaviour of these algorithms.

Throughout the paper we use only the 2-norm and assume the standard floating point arithmetic with the unit roundoff u (see Chapter 2 in [10]).

2. Perturbation analysis

We recall Wedin's results (cf. [12], [11, pp. 49–51], [5, pp. 26–31], [10, pp. 382–384]) on the sensitivity of the least squares solution to small perturbations in A and b.

Theorem 2.1. *Let $A \in \mathbb{R}^{m\times n}$ have full column rank with $m \geq n$ and $b \in \mathbb{R}^m$. Let $\hat{x}_*$ be the exact solution of the least squares problem $\min_x \|\hat{b} - \hat{A}x\|$, where*

$$\hat{A} = A + \Delta A, \quad \|\Delta A\| \leq \epsilon_A \|A\|,$$

$$\hat{b} = b + \Delta b, \quad \|\Delta b\| \leq \epsilon_b \|b\|$$

and A is the matrix in (1). Let $\hat{r}_* = \hat{b} - \hat{A}\hat{x}_*$ and $r_* = b - Ax_*$ be the residuals for the perturbed and the original least squares problems. Assume that $\epsilon_A \kappa(A) < 1$, where $\kappa(A) = \|A\|\,\|A^\dagger\|$ is the standard condition number of the matrix A. Then $rank(\hat{A}) = rank(A) = n$ and

$$\|\hat{x}_* - x_*\| \leq \frac{\kappa(A)}{1 - \epsilon_A \kappa(A)} \left(\epsilon_A \|x_*\| + \epsilon_b \frac{\|b\|}{\|A\|} + \epsilon_A \|A^\dagger\| \|r_*\| \right), \tag{7}$$

$$\|\hat{r}_* - r_*\| \leq \epsilon_A \|A\| \|x_*\| + \epsilon_b \|b\| + \epsilon_A \kappa(A) \|r_*\|. \tag{8}$$

Remark 2.1. If $x_* \neq 0$, then we can rewrite the bound (7) as follows

$$\frac{\|\hat{x}_* - x_*\|}{\|x_*\|} \leq \frac{\epsilon_A\, \kappa_{LS}(A,b) + \epsilon_b \kappa_b(A,b)}{1 - \epsilon_A \kappa(A)}, \tag{9}$$

where

$$\kappa_{LS}(A,b) = \kappa(A) \left(1 + \kappa(A) \frac{\|r_*\|}{\|A\|\,\|x_*\|} \right) \tag{10}$$

is the *condition number of the LS problem with respect to small perturbations in A* (see p.31 of [5]) and

$$\kappa_b(A,b) = \frac{\|A^\dagger\| \|b\|}{\|x_*\|} \tag{11}$$

is *the condition number of the least squares problem with respect to small perturbations in b.*

Unlike for a square nonsingular linear system $Ax = b$, the sensitivity of the LS problem depends strongly not only on the matrix A, but also on the right-hand side vector b. The incompatibility of the problem is measured by

$$\omega(A, b) = \kappa(A)\frac{\|r_*\|}{\|A\|\,\|x_*\|}. \tag{12}$$

We have $\kappa_{LS}(A, b) = \kappa(A)(1+\omega(A, b))$, and due to $b = Ax_*+r_*$ and $r_*^T Ax_* = 0$ we get $\|b\|^2 = \|Ax_*\|^2 + \|r_*\|^2$, so $\|Ax_*\|,\ \|r_*\| \le \|b\| \le \|A\|\|x_*\| + \|r_*\|$. This leads to the lower and upper bounds for $\kappa_b(A, b)$

$$\omega(A, b) \le \kappa_b(A, b) \le \kappa(A) + \omega(A, b) \le \kappa_{LS}(A, b). \tag{13}$$

If $\omega(A, b) = 0$ (i.e. the system is compatible) then $\|b\| = \|Ax_*\| \le \|A\|\|x_*\|$, so $\kappa_{LS}(A, b) = \kappa(A) \ge \kappa_b(A, b)$. The situation is different in the case where the factor $\omega(A, b)$ is large and the term proportional to $\kappa^2(A)$ dominates in $\kappa_{LS}(A, b)$. However, as indicated in [8, 9], in practice very often the linear systems $Ax \approx b$ are close to being consistent and the corresponding quantities $\omega(A, b)$ are moderate for most vectors b.

3. Numerical stability

In this paper, we study the forward stability of algorithms for computing the least squares solution. More precisely, if the approximate solution $\tilde{x}$ satisfies the bound

$$\|\tilde{x} - x_*\| \le\ \mathcal{O}(u)\left(\kappa(A)\|x_*\| + \kappa(A)\|A^\dagger\|\|r_*\|\right) \tag{14}$$

then we call $\tilde{x}$ a *forward stable solution* to the least squares problem (1). We see that for $x_* \neq 0$ we can rewrite the inequality (14) as

$$\frac{\|\tilde{x} - x_*\|}{\|x_*\|} \leq \mathcal{O}(u)\, \kappa_{LS}(A, b). \tag{15}$$

Å. Björck proved that the seminormal equations method based on the backward stable QR factorization of A, i.e. $R^T R x_* = A^T b$, is not *forward stable* [3], and a more general case was considered by Å. Björck and C.C. Paige in [6]. However, they indicate (without a detailed proof) that if $\mathcal{O}(u)\kappa^2(A) < 1$ then one step of iterative refinement called the *corrected seminormal equations* method ($CSNE_{QR}$) produces a forward stable solution to (1) in the sense of (14). In addition, it was noted that the $CSNE_{QR}$ method is not forward stable for $\mathcal{O}(u)\kappa(A) < 1$. More precisely, Å. Björck proved (see Theorem 3.1 and Theorem 3.2 in [3]) that the vectors $\tilde{x}$ and $\tilde{x}_{new}$ computed by SNE_{QR} and $CSNE_{QR}$ respectively, satisfy

$$\frac{\|\tilde{x} - x_*\|}{\|x_*\|} \leq \mathcal{O}(u)\kappa^2(A) + \mathcal{O}(u)\, \kappa(A)\, \kappa_b(A, b) + \mathcal{O}(u^2)\, \kappa^3(A), \tag{16}$$

$$\frac{\|\tilde{x}_{new} - x_*\|}{\|x_*\|} \leq \mathcal{O}(u)\kappa_{LS}(A, b) + \mathcal{O}(u^2)\, \kappa^2(A)\, \kappa_b(A, b) + \mathcal{O}(u^2)\, \kappa^3(A). \tag{17}$$

Our main results on SNE_{SVD} and $CSNE_{SVD}$ in Sections 4 and 5 will be of a similar type, but we assume the use of the SVD decomposition instead of the QR factorization of A. In our error analysis we obtain error bounds similar to (16) and (17).

4. Error analysis of Algorithm II

In this section we give a normwise analysis of Algorithm II that provides some further insight.

Lemma 4.1. *Suppose that we have computed the SVD decomposition of the matrix A, such that the assumptions (5)–(6) hold with $\hat{A} = A + \Delta A$ and*

$$\|\Delta A\| \leq \mathcal{O}(u)\|A\|, \quad \mathcal{O}(u)\kappa(A) < 1. \tag{18}$$

Then $rank(\hat{A}) = rank(A) = n$ and the computed solution $\tilde{x}$ in floating point arithmetic by Algorithm II satisfies

$$\hat{A}^T\hat{A}\,\hat{x}_* = \hat{A}^T\hat{b}, \quad \hat{b} = b + \Delta b, \quad \|\Delta b\| \leq \mathcal{O}(u)\kappa(A)\|b\|, \tag{19}$$

where $\hat{x}_* = (I + F)\tilde{x}$ with $\|F\| \leq \mathcal{O}(u)$.

Proof. We make a standard assumption that the matrix-vector multiplication is backward stable, i.e. there exists a matrix E_1 such that $fl(A^Tb) = (A + E_1)^Tb$ with $\|E_1\| \leq \mathcal{O}(u)\|A\|$. Notice that the computed vector $\tilde{c} = fl(\tilde{V}^T fl(A^Tb))$ satisfies

$$\tilde{c} = \hat{V}^T(I + F_1)(A + E_1)^Tb = \hat{V}^T(A + E_2)^Tb, \tag{20}$$

where $\|F_1\| \leq \mathcal{O}(u)$ and $\|E_2\| \leq \mathcal{O}(u)\|A\|$. There exists a diagonal matrix F_2, such that the computed solution $\tilde{y}$ satisfies $\tilde{\Sigma}^2\tilde{y} = (I + F_2)\tilde{c}, \quad \|F_2\| \leq \mathcal{O}(u)$. This together with (20) gives the identity

$$\tilde{\Sigma}^2\tilde{y} = (I + F_2)\hat{V}^T(A + E_2)^Tb = \hat{V}^T(I + F_3)(A + E_2)^Tb,$$

where $\|F_3\| \leq \mathcal{O}(u)$. Thus, we have

$$\tilde{\Sigma}^2\tilde{y} = \hat{V}^T(A + E)^Tb, \quad \|E\| \leq \mathcal{O}(u)\|A\|. \tag{21}$$

We see that $\tilde{x} = fl(\tilde{V}\tilde{y}) = (I + F_4)\hat{V}\tilde{y}, \quad \|F_4\| \leq \mathcal{O}(u)$. It follows that

$$\tilde{y} = \hat{V}^T(I + F_4)^{-1}\tilde{x} = \hat{V}^T(I + F)\tilde{x}, \quad \|F\| \leq \mathcal{O}(u).$$

This, together with (21), leads to $\tilde{\Sigma}^2\hat{V}^T(I+F)\tilde{x} = \hat{V}^T(A+E)^Tb$, which we rewrite as $\hat{V}\tilde{\Sigma}^2\hat{V}^T(I+F)\tilde{x} = (A+E)^Tb$. Due to $\hat{A}^T\hat{A} = \hat{V}\tilde{\Sigma}^2\hat{V}^T$, we have $\hat{A}^T\hat{A}\,(I+F)\tilde{x} = (A+E)^Tb = (\hat{A}+E-\Delta A)^Tb$. Since $rank(\hat{A}) = n$, we have the identity $\hat{A}^\dagger\hat{A} = I$. This gives $(\hat{A}^\dagger\hat{A})^T = \hat{A}^T\,(\hat{A}^\dagger)^T = I$ and we can write $(A+E)^Tb = \hat{A}^T(b+\Delta b), \quad \Delta b = (\hat{A}^\dagger)^T(E-\Delta A)^Tb$. Then $\|\Delta b\| \le \|\hat{A}^\dagger\|\,(\|E\| + \|\Delta A\|)\|b\| \le \mathcal{O}(u)\|A\|\|\hat{A}^\dagger\|\|b\|$. From (18), we obtain

$$\|\hat{A}^\dagger\| = \|(A+\Delta A)^\dagger\| \le \frac{\|A^\dagger\|}{1-\mathcal{O}(u)\kappa(A)} \le (1+\mathcal{O}(u)\kappa(A))\|A^\dagger\|.$$

The proof is now complete. □

From (19), we see that the computed vector $\tilde{x}$ is a slightly perturbed solution of the LS problem with perturbed data $\hat{A}$ and $\hat{b}$. However, the perturbation of the right-hand side $\epsilon_b = \mathcal{O}(u)\kappa(A)$ is proportional to $\kappa(A)$.

Theorem 4.1. *Assume the hypothesis of Lemma 4.1. Let x_* be the exact solution to (1), $r_* = b - Ax_*$ and $\tilde{x}$ be computed in floating point arithmetic by Algorithm II and $\tilde{s} = b - A\tilde{x}$. Then we have*

$$\|\tilde{x} - x_*\| \le \mathcal{O}(u)\kappa(A)\,\|A^\dagger\|\|b\| \tag{22}$$

and

$$\|\tilde{s} - r_*\| = \|A(\tilde{x} - x_*)\| \le \mathcal{O}(u)\kappa(A)\,\|b\|. \tag{23}$$

Proof. Let $\hat{x}_* = (I+F)\tilde{x}$ be defined by (19). Applying Theorem 2.1 to $\epsilon_A = \mathcal{O}(u)$ and $\epsilon_b = \mathcal{O}(u)\kappa(A)$ gives the following bound:

$$\|\hat{x}_* - x_*\| \;\le\; \mathcal{O}(u)\kappa(A)\,\left(\|x_*\| + \kappa(A)\frac{\|b\|}{\|A\|} + \|A^\dagger\|\|r_*\|\right). \tag{24}$$

Since $\|r_*\| \le \|b\|$ and $\|x_*\| = \|A^\dagger b\| \le \|A^\dagger\|\|b\|$, we rewrite (24) as $\|\hat{x}_* - x_*\| \le \mathcal{O}(u)\kappa(A)\,\|A^\dagger\|\|b\|$. Substituting now for $\hat{x}_* = (I+F)\tilde{x}$, we get $\|\tilde{x} - x_*\| \le$

$\mathcal{O}(u)\|\tilde{x}\| + \mathcal{O}(u)\kappa(A)\,\|A^{\dagger}\|\|b\|$. The bound (22) follows immediately from $\|\tilde{x}\| \leq \|\tilde{x} - x_*\| + \|x_*\|$ and $\|x_*\| \leq \|A^{\dagger}\|\|b\|$.

It remains to prove (23). Let $\hat{r}_* = \hat{b} - \hat{A}\hat{x}_*$. Considering now the result (8) with the perturbations $\epsilon_A = \mathcal{O}(u)$ and $\epsilon_b = \mathcal{O}(u)\kappa(A)$, we get $\|\hat{r}_* - r_*\| \leq \mathcal{O}(u)\|A\|\|x_*\| + \mathcal{O}(u)\kappa(A)\,(\|b\| + \|r_*\|)$. Since $\|r_*\| \leq \|b\|$ and $\|A\|\|x_*\| \leq \kappa(A)\,\|b\|$, we obtain

$$\|\hat{r}_* - r_*\| \leq \mathcal{O}(u)\kappa(A)\,\|b\|. \tag{25}$$

From (19), it follows that $\tilde{s} - \hat{r}_* = b - A\tilde{x} - \hat{b} + \hat{A}(I+F)\tilde{x} = (\hat{A}F + \Delta A)\tilde{x} - \Delta b$, and so $\|\tilde{s} - \hat{r}_*\| \leq \mathcal{O}(u)\|A\|\|\tilde{x}\| + \mathcal{O}(u)\kappa(A)\|b\|$. By (22), we get $\|\tilde{x}\| \leq \|x_*\| + \mathcal{O}(u)\kappa(A)\,\|A^{\dagger}\|\|b\|$, hence $\|\tilde{s} - \hat{r}_*\| \leq \mathcal{O}(u)(\|A\|\|x_*\| + \kappa(A)\|b\|) + \mathcal{O}(u^2)\kappa^2(A)\|b\|$. By the assumption (18) on the numerical "nonsingularity" of A, and noting that $\|A\|\|x_*\| \leq \kappa(A)\,\|b\|$, we have

$$\|\tilde{s} - \hat{r}_*\| \leq \mathcal{O}(u)\kappa(A)\,\|b\|. \tag{26}$$

Now it is easily seen that the bound (23) follows on from (25)-(26) and the inequality $\|\tilde{s} - r_*\| \leq \|\tilde{s} - \hat{r}_*\| + \|\hat{r}_* - r_*\|$. □

5. Error analysis of Algorithm IV

Theorem 5.1. *Let $\tilde{x}$, $\tilde{r}$, $\Delta\tilde{x}$ and $\tilde{x}_{new}$ be computed by Algorithm IV in floating point arithmetic. Under the hypothesis of Lemma 4.1 and for $x_* \neq 0$, we have*

$$\frac{\|\tilde{x}_{new} - x_*\|}{\|x_*\|} \leq \mathcal{O}(u)\kappa_{LS}(A,b) + \mathcal{O}(u^2)\,\kappa^2(A)\,\kappa_b(A,b). \tag{27}$$

Proof. We recall that first we compute the vector $\tilde{x}$ by Algorithm II and then we compute $\tilde{r} = fl(b - A\tilde{x})$. The next step is to find an approximate solution $\Delta\tilde{x}$ to the system $A^TAx = A^T\tilde{r}$ by Algorithm II, and then compute $\tilde{x}_{new}$ as

an update of $\tilde{x}$ with the correction $\Delta\tilde{x}$. From the proof of Theorem 4.1, we have the following identities for the computed vectors $\tilde{x}$ and $\Delta\tilde{x}$

$$\tilde{x} = x_* + e_1, \quad \|e_1\| \leq \mathcal{O}(u)\kappa(A)\,\|A^\dagger\|\|b\|, \tag{28}$$

$$\Delta\tilde{x} = A^\dagger\,\tilde{r} + e_2, \quad \|e_2\| \leq \mathcal{O}(u)\kappa(A)\,\|A^\dagger\|\|\tilde{r}\|. \tag{29}$$

The computed vector $\tilde{x}_{new}$ is given as

$$\tilde{x}_{new} = fl(\tilde{x} + \Delta\tilde{x}) = \tilde{x} + \Delta\tilde{x} + e_3, \quad \|e_3\| \leq u\|\tilde{x} + \Delta\tilde{x}\|. \tag{30}$$

The computed residual $\tilde{r}$ is related to the true residual $\tilde{s} = b - A\tilde{x}$ as follows

$$\tilde{r} = fl(b - A\tilde{x}) = \tilde{s} + f, \quad \|f\| \leq \mathcal{O}(u)(\|b\| + \|A\|\|\tilde{x}\|). \tag{31}$$

By (28), we get the bound $\|\tilde{x}\| \leq \|x_*\| + \|e_1\|$, hence it follows that

$$\|f\| \leq \mathcal{O}(u)(\|b\| + \|A\|\|x_*\|) + \mathcal{O}(u^2)\kappa^2(A)\,\|b\|. \tag{32}$$

Due to (31), we have $A^\dagger\tilde{r} = A^\dagger\tilde{s} + A^\dagger f$, and using $A^\dagger\tilde{s} = x_* - \tilde{x}$, we get $A^\dagger\tilde{r} = x_* - \tilde{x} + A^\dagger f$. This identity, together with (29) and (30), gives

$$\tilde{x} + \Delta\tilde{x} = x_* + A^\dagger f + e_2, \quad \tilde{x}_{new} - x_* = A^\dagger f + e_2 + e_3. \tag{33}$$

Thus, from (30) and (33), we have $\|e_3\| \leq u(\|x_*\| + \|A^\dagger\|\|f\| + \|e_2\|)$ and

$$\|\tilde{x}_{new} - x_*\| \leq (1+u)(\|A^\dagger\|\|f\| + \|e_2\|) + u\|x_*\|. \tag{34}$$

It remains to estimate $\|A^\dagger\|\|f\|$ and $\|e_2\|$. From (32) we obtain

$$\|A^\dagger\|\|f\| \leq \mathcal{O}(u)(\|A^\dagger\|\|b\| + \kappa(A)\|x_*\|) + \mathcal{O}(u^2)\kappa^2(A)\,\|A^\dagger\|\|b\|. \tag{35}$$

From the second statement (23) of Theorem 4.1 it follows that

$$\|\tilde{s}\| \leq \|r_*\| + \mathcal{O}(u)\kappa(A)\|b\|. \tag{36}$$

The bound (36) and (32), together with the assumption $\mathcal{O}(u)\kappa(A) < 1$ on the numerical "nonsingularity" of A and the inequality $\|x_*\| \leq \|A^\dagger\|\|b\|$, yields the bound $\|\tilde{r}\| \leq \|r_*\| + \mathcal{O}(u)\kappa(A)\|b\|$. The bounds (29), (32), (35) and (36) yield

$$\|A^\dagger\|\|f\| + \|e_2\| \leq \mathcal{O}(u)(\kappa_{LS}(A,b) + \kappa_b(A,b))\|x_*\| + \mathcal{O}(u^2)\kappa^2(A)\,\|A^\dagger\|\|b\|.$$

We see that this together with (34) and the inequality $\kappa_b(A,b) \leq \kappa_{LS}(A,b)$ gives the final bound

$$\|\tilde{x}_{new} - x_*\| \leq \mathcal{O}(u)\left(\kappa_{LS}(A,b) + O(u)\kappa^2(A)\ \ \kappa_b(A,b)\right)\|x_*\|.$$

The proof is now complete. □

Remark 5.1. How do we interpret Theorems 4.1- 5.1? We have derived the bounds (22) and (27). We see that if $x_* \neq 0$ then the vectors $\tilde{x}$ and $\tilde{x}_{new}$ computed by Algorithm II (SNE_{SVD}) and Algorithm IV ($CSNE_{SVD}$) respectively, satisfy

$$\frac{\|\tilde{x} - x_*\|}{\|x_*\|} \leq \mathcal{O}(u)\,\kappa(A)\,\kappa_b(A,b) \tag{37}$$

and

$$\frac{\|\tilde{x}_{new} - x_*\|}{\|x_*\|} \leq \mathcal{O}(u)\left[\kappa_{LS}(A,b) + (\mathcal{O}(u)\kappa(A))\kappa(A)\,\kappa_b(A,b)\right]. \tag{38}$$

These results are similar to the ones obtained by Å. Björck for Algorithm I and Algorithm III respectively, but now the terms $\mathcal{O}(u^2)\,\kappa^3(A)$ are not present (see Theorem 3.1 and Theorem 3.2 in [3]). In this sense, the numerical properties of Algorithm II and Algorithm IV based on the SVD of A are similar to their counterparts based on the QR decomposition of A.

How do we compare the bounds (37) and (38) to (15), that is, when Algorithms II and IV are forward stable in the sense of (15)? We recall that

$$\kappa_{LS}(A,b) = \kappa(A) + \kappa(A)\omega(A,b),$$

where $\omega(A,b)$ is defined in (12).

Consider first the case where $\omega(A,b) \gg \kappa(A)$. From this and (11)-(13), it follows that $\kappa_{LS}(A,b)$ and $\kappa(A)\kappa_b(A,b)$ are of order $\kappa(A)\omega(A,b)$. The bound (38) contains two parts. We see that the second term in the bound (38), i.e., $(\mathcal{O}(u)\kappa(A))\kappa(A)\,\kappa_b(A,b)$, will also be of order $\kappa(A)\omega(A,b)$. The conclusion is that both Algorithms II and IV are forward stable and iterative refinement is not necessary in this case.

Now consider the situation where $\omega(A,b) = \mathcal{O}(1)$. Then $\kappa_{LS}(A,b) \approx \kappa(A)$ and $\kappa(A) \leq \kappa(A)\,\kappa_b(A,b) \lessapprox \kappa^2(A)$, so

$$\kappa(A) + \mathcal{O}(u)\kappa^2(A) \lessapprox \kappa(A) + (\mathcal{O}(u)\kappa(A))\kappa(A)\,\kappa_b(A,b) \leq \kappa(A) + \mathcal{O}(u)\kappa^3(A).$$

This shows why Algorithm IV has better numerical properties than Algorithm II for $\mathcal{O}(u)\kappa(A) < 1$. It also indicates that the condition $\mathcal{O}(u)\kappa^2(A) < 1$ is needed for forward stability of Algorithm IV, and explains why, in general, Algorithms II and IV (likewise Algorithms I and III) are not forward stable. This is also made visible by our numerical experiments in the following section.

6. Numerical experiments

In this section we illustrate our theoretical results. All experiments were performed using *MATLAB* with unit roundoff $u = 1.1 \cdot 10^{-16}$. We assume two

extreme cases, where the least squares solution x_* is equal to the right singular vector corresponding to the smallest or to the largest singular value of the matrix A. As it will be clear from the results, the correction step is important, especially for problems with solutions close (equal) to the right singular vectors corresponding to the largest singular values. The problem with dimensions $m = 20$ and $n = 7$ is defined by the singular value decomposition of the matrix $A = U\Sigma V^T$, where U and V are orthogonal matrices with corresponding dimensions generated by the *orthog* subroutine from the *gallery* in *MATLAB* (we consider only the first n columns for matrix U). The matrix Σ is a diagonal matrix with the singular values given as $\sigma_i(A) = 10^{6-1.5i}$ for $i = 1, \ldots, n$. It is clear then that $\kappa(A) = 10^9$. Thus we consider two problems with the solutions x_1 and x_2 given by two columns in the matrix V corresponding to the largest and smallest singular values, respectively. This leads to the right hand side vectors $b_1 = Ax_1$ with $\kappa_b(A, b_1) = \kappa(A)$ and $b_2 = Ax_2$ with $\kappa_b(A, b_2) = 1$. Finally we take vector h as the scaled $(n + 1)$-st column of the orthogonal matrix that generates the matrix U with $\|h\| = \sigma_n(A)$. It is clear then that $A^T h = 0$ and $\kappa_{LS}(A, b) \approx \kappa(A)$ for all problems with the residual norm smaller than $\|h\|$. Tables 2 and 3 summarize our numerical results. The first rows correspond to the case of the over-determined systems $Ax_1 = b_1$ and $Ax_2 = b_2$ with the solutions x_1 and x_2. In the subsequent rows we have increased the residual norm of the least squares problem via the appropriate scaling of the vector h. The norms of relative errors are scaled by $\kappa_{LS}(A, b)$ and thus correspond to the theoretical bound (27). We see that SNE_{SVD} is satisfactory if the solution x_2 is equal to the right singular vector corresponding to the smallest singular value. This is no longer true for the

solution x_1 equal to the right singular vector corresponding to the norm of A. In this case the refinement step in the CSNE$_{SVD}$ method can significantly improve the accuracy of the computed solution. In addition, our problems almost meet the assumption that $\mathcal{O}(u)\kappa^2(A) \leq 1$ and so CSNE$_{SVD}$ computes forward stable approximate solutions.

This is not true for a similar example, where we fix the residual norm as $\|h\| = 10^{-10}$ and generate the system matrices A in the same way as in the first experiment but with $\sigma_1(A) = 1$ and we change their condition numbers in order to have $\mathcal{O}(u^2)\kappa^3(A) \approx 1$. We take again the right hand side vectors as $b_1 = Ax_1$, where x_1 are the left singular vectors corresponding to the largest singular values of A so that $\kappa_b(A, b_1) = \kappa(A)$. It is clear then that for sufficiently small residual norms $\|h\|$ we have $\kappa_{LS}(A, b) \approx \kappa_b(A, b)$. The numerical results are summarized in Table 4. In this case the refinement step in $CSNE_{SVD}$ does not significantly improve the accuracy of the solution computed by SNE_{SVD} for some problems and therefore it does not deliver forward stable approximate solutions.

7. Conclusions

In this paper we have considered two methods for the solution of the least squares problems which are based only on the factors Σ and V from the SVD decomposition of the matrix A. We have observed that while the SNE_{SVD} method (Algorithm II) is not forward stable, for the $CSNE_{SVD}$ method (Algorithm IV), i.e. one step of iterative refinement, it will usually be enough to yield a stable solution to (1). This method could be a method of choice, especially when the matrix A is 'close' to rank deficient.

b	$\kappa_{LS}(A,b)$	$\kappa_b(A,b)$	$\omega(A,b)$	$\frac{\|\tilde{x}-x_1\|}{\|x_1\|*\kappa_{LS}(A,b)}$	$\frac{\|\tilde{x}_{new}-x_1\|}{\|x_1\|*\kappa_{LS}(A,b)}$
b_1	1e+09	1e+09	0	9.9996e-10	3.9902e-15
$b_1 + 10^{-7} \cdot h$	1e+09	1e+09	1.0952e-07	9.9996e-10	5.0226e-15
$b_1 + 10^{-6} \cdot h$	1e+09	1e+09	1.0089e-06	9.9998e-10	8.2096e-15
$b_1 + 10^{-5} \cdot h$	1e+09	1e+09	9.9799e-06	9.9995e-10	5.1573e-15
$b_1 + 10^{-4} \cdot h$	1.0001e+09	1e+09	0.00010001	9.9986e-10	4.8555e-15
$b_1 + 10^{-3} \cdot h$	1.001e+09	1e+09	0.001	9.9897e-10	4.2514e-15
$b_1 + 10^{-2} \cdot h$	1.01e+09	1e+09	0.01	9.9006e-10	4.6066e-15
$b_1 + 10^{-1} \cdot h$	1.1e+09	1e+09	0.1	9.0906e-10	4.3416e-15
$b_1 + 10^{0} \cdot h$	2e+09	1e+09	1	4.9998e-10	1.781e-15
$b_1 + 10^{1} \cdot h$	1.1e+10	1e+09	10	9.0906e-11	2.9992e-16
$b_1 + 10^{2} \cdot h$	1.01e+11	1e+09	100	9.9006e-12	4.9103e-17
$b_1 + 10^{3} \cdot h$	1.001e+12	1e+09	1000	9.9899e-13	3.0604e-19
$b_1 + 10^{4} \cdot h$	1.0001e+13	1e+09	10000	9.9988e-14	8.097e-18
$b_1 + 10^{5} \cdot h$	1e+14	1e+09	1e+05	9.9997e-15	1.6214e-17
$b_1 + 10^{6} \cdot h$	1e+15	1e+09	1e+06	9.9998e-16	1.0055e-18
$b_1 + 10^{7} \cdot h$	1e+16	1e+09	1e+07	9.9998e-17	5.8677e-18

Table 2: Results for the least squares problem where b_1 is such that $\kappa_b(A, b_1) = \kappa(A) = 10^9$.

b	$\kappa_{LS}(A,b)$	$\kappa_b(A,b)$	$\omega(A,b)$	$\frac{\|\tilde{x}-x_2\|}{\|x_2\|*\kappa_{LS}(A,b)}$	$\frac{\|\tilde{x}_{new}-x_2\|}{\|x_2\|*\kappa_{LS}(A,b)}$
b_2	1e+09	1	0	4.6511e-17	8.8649e-18
$b_2 + 10^{-7} \cdot h$	1e+09	1	1e-07	4.2913e-17	5.3251e-18
$b_2 + 10^{-6} \cdot h$	1e+09	1	1e-06	3.9881e-17	1.3674e-18
$b_2 + 10^{-5} \cdot h$	1e+09	1	1e-05	4.0147e-17	9.0393e-18
$b_2 + 10^{-4} \cdot h$	1.0001e+09	1	0.0001	4.2962e-17	1.1676e-17
$b_2 + 10^{-3} \cdot h$	1.001e+09	1	0.001	4.1998e-17	2.7233e-18
$b_2 + 10^{-2} \cdot h$	1.01e+09	1	0.01	4.2684e-17	9.828e-18
$b_2 + 10^{-1} \cdot h$	1.1e+09	1.005	0.1	4.1717e-17	1.8121e-18
$b_2 + 10^{0} \cdot h$	2e+09	1.4142	1	1.597e-17	4.6848e-18
$b_2 + 10^{1} \cdot h$	1.1e+10	10.05	10	6.3486e-18	8.9225e-18
$b_2 + 10^{2} \cdot h$	1.01e+11	100	100	9.4034e-18	3.0071e-19
$b_2 + 10^{3} \cdot h$	1.001e+12	1000	1000	1.2916e-18	3.8996e-18
$b_2 + 10^{4} \cdot h$	1.0001e+13	10000	10000	2.4229e-18	7.1273e-18
$b_2 + 10^{5} \cdot h$	1e+14	1e+05	1e+05	2.8613e-18	5.8605e-19
$b_2 + 10^{6} \cdot h$	1e+15	1e+06	1e+06	1.4121e-17	9.603e-19
$b_2 + 10^{7} \cdot h$	1e+16	1e+07	1e+07	8.8245e-18	1.0793e-17

Table 3: Results for the least squares problem where $\kappa(A) = 10^9$ and b_2 is such that $\kappa_b(A, b_2) = 1$.

$\kappa(A)$	$\kappa_{LS}(A,b)$	$\kappa_b(A,b)$	$\omega(A,b)$	$\frac{\|\tilde{x}-x_1\|}{\|x_1\|*\kappa_{LS}(A,b)}$	$\frac{\|\tilde{x}_{new}-x_1\|}{\|x_1\|*\kappa_{LS}(A,b)}$
1e+08	1.01e+08	1e+08	0.01	4.8055e-09	2.1666e-17
1.8e+08	1.8099e+08	1.7783e+08	0.017783	5.4896e-09	5.6397e-16
3.2e+08	3.2623e+08	3.1623e+08	0.031623	1.2488e-11	2.0615e-17
5.6e+08	5.9396e+08	5.6234e+08	0.056234	3.5798e-11	2.3623e-17
1e+09	1.1e+09	1e+09	0.1	9.0894e-10	4.477e-16
1.8e+09	2.0945e+09	1.7783e+09	0.17783	4.7743e-10	4.2348e-15
3.2e+09	4.1623e+09	3.1623e+09	0.31623	2.4025e-10	1.0291e-14
5.6e+09	8.7857e+09	5.6234e+09	0.56234	1.1382e-10	4.3864e-14
1e+10	2e+10	1e+10	1	4.6613e-11	3.6945e-16
1.8e+10	4.9406e+10	1.7783e+10	1.7783	2.0241e-11	3.9526e-13
3.2e+10	1.3162e+11	3.1623e+10	3.1623	7.5975e-12	5.0579e-14
5.6e+10	3.7246e+11	5.6234e+10	5.6234	2.6848e-12	3.359e-13
1e+11	1.1e+12	1e+11	10	9.0909e-13	9.9477e-14
1.8e+11	3.3401e+12	1.7783e+11	17.783	2.9939e-13	2.887e-13
3.2e+11	1.0316e+13	3.1623e+11	31.623	9.6936e-14	9.6935e-14
5.6e+11	3.2185e+13	5.6234e+11	56.234	3.1071e-14	7.9303e-15
1e+12	1.0099e+14	9.9997e+11	99.997	9.9015e-15	9.8987e-15

Table 4: Results for the least squares problem where b_1 is such that $\kappa_b(A,b_1) = \kappa(A)$, $\|A\| = 1$, $b = b_1 + h$ where $\|h\| = 1 \cdot 10^{-10}$.

Acknowledgements. The authors would like to express their gratitude to the referees for their close reading of the manuscript and numerous helpful comments and suggestions.

References

[1] Å. Björck, Iterative refinement of linear least squares solutions I, BIT 7 (1967) 257–278.

[2] Å. Björck, Iterative refinement of linear least squares solutions II, BIT 8 (1968) 8–30.

[3] Å. Björck, Stability analysis of the method of seminormal equations for linear least squares problems, Linear Algebra Appl. 88/89 (1987) 31–48.

[4] Å. Björck, Iterative refinement and reliable computing, in: M. G. Cox, S. J. Hammarling (eds.), Reliable Numerical Computation, Oxford University Press, 1990.

[5] Å. Björck, Numerical methods for least squares problems, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1996.

[6] Å. Björck, C. C. Paige, Solution of augmented linear systems using orthogonal factorizations, BIT 34 (1) (1994) 1–24.

[7] R. Byers, H. Xu, A new scaling for Newton's iteration for the polar decomposition and its backward stability, SIAM Journal on Matrix Analysis and Applications 30 (2008) 822–843.

[8] N. Castro-González, J. Ceballos, F. M. Dopico, J. M. Molera, Multiplicative perturbation theory and accurate solution of least squares problems, a manuscript.

[9] F. M. Dopico, J. M. Molera, Accurate solution on structured linear systems via rank-revealing decompositions, IMA Journal of Numerical Analysis 32(3) (2012) 1096–1116.

[10] N. J. Higham, Accuracy and stability of numerical algorithms, 2nd ed., Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2002.

[11] C. L. Lawson, R. J. Hanson, Solving least squares problems, Prentice-Hall Inc., Englewood Cliffs, N.J., 1974, prentice-Hall Series in Automatic Computation.

[12] P. Å. Wedin, Perturbation theory for pseudo-inverses, BIT 13 (1973) 217–232.