Level choice in truncation methods for linear models

Diana Sima and Sabine Van Huffel

February 10, 2006

Abstract

In the context of ill-conditioned linear systems, regularization techniques where a hyperparameter is present, are often employed. In particular, a truncation level must be chosen in the case of truncation-based methods. Model selection methods that are often applied in the context of *truncated singular value decomposition* are modified in this paper for the context of *truncated total least squares*.

1 Introduction

In this paper, we study truncation methods for the regularization of linear discrete ill-posed problems. Ill-posed problems are problems where the solution does not depend continuously on the input data, where arbitrarily small perturbations in the input data produce arbitrarily large changes in the solution. Discrete ill-posed problems might arise from the discretization of continuous problems, such as integral or differential models.

Extensive characterizations of discrete ill-posed problems of the type $Ax \approx b$ exist in the literature (see the book [11] and the references therein). It is known, for instance, that the properties of such a system can be understood from the singular value decomposition of A. In an ill-posed system, there is no clear gap in the decay of the singular values of A. This makes it difficult to decide the best value that can be used as numerical rank. Moreover, it usually happens that the singular vectors corresponding to smaller and smaller singular values have increasing complexity (meaning that they contain more and more features, oscillations). If the least squares or total least squares solution of $Ax \approx b$ is computed, then these oscillations are emphasized in the solution by the noise in the data or by the effects of working in finite precision arithmetic.

A better understanding of truncation methods (such as truncated singular value decomposition (TSVD) and truncated total least squares (TTLS)) is possible in view of the recent results on *core problems* of linear systems [16]. The core reduction of an incompatible linear system is a tool that is able to avoid nonunique and nongeneric solutions of the total least squares problem (and variations). We propose the use of *truncated core problems* in order to avoid close-to-nongenericity in ill-posed linear approximation problems.

Among the numerous methods of regularization that can be used for approaching discrete ill-posed problems [11], we revisit in this paper one of the simplest: truncation. The aim of this study encompass:

- a better understanding of truncation methods (such as truncated singular value decomposition (TSVD) and truncated total least squares (TTLS)) in view of the recent results on *core* reductions of linear systems [16];

- the analysis of methods for choosing the truncation level, by adapting several classical methods for model selection.

2 Truncation methods for linear ill-posed problems

2.1 Linear ill-posed problems

We illustrate with an example the fact that it is not advisable to use any of the classical estimation methods (least squares - LS, total least squares - TLS [7, 21]) directly on an ill-posed linear problem.

Example 1. *A matrix A_0 and a vector b_0 with real valued elements and dimensions $m = 200$, $n = 30$, are simulated using the example* `ilaplace` *from the Regularization Tools [10]. This example that originally comes from [22] constructs A_0 as a discretization of the inverse Laplace transform, using Gauss-Laguerre quadrature for the discretization of the inverse Laplace integral operator. In [22] and in Hansen's toolbox [10], the sampling points where the integral equation is evaluated were as many as the quadrature points, thus yielding a square discretized system. We slightly modify this setting in order to construct rectangular systems, by allowing more sampling points than quadrature points.*

Figure 1 shows the singular values of $\begin{bmatrix} b_0 & A_0 \end{bmatrix}$, as they decay without noticeable gap towards machine-precision zero.

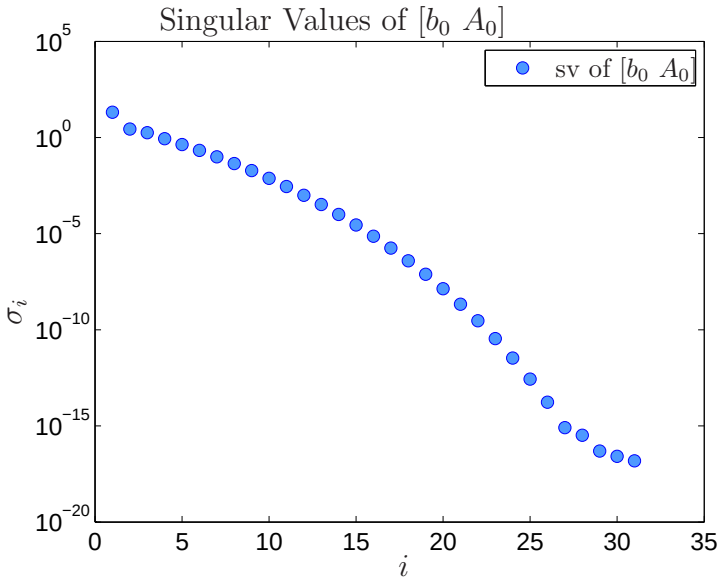


Figure 1: Singular values of $[b_0\ A_0]$ decaying without gap towards zero, where the data $[b_0\ A_0]$ comes from a discrete ill-posed problem.

In this simulation, $A_0x \approx b_0$ has a certain exact solution vector x_0, satisfying the equation $A_0x_0 = b_0$. We construct an incompatible noisy example $Ax \approx b$, by adding Gaussian white noise with a standard deviation of 1e-5 to all elements of A_0 and b_0. For comparison, Figure 2 shows also the decaying singular values of the noisy $\begin{bmatrix} b & A \end{bmatrix}$, and it is noticeable that the smallest singular values stagnate at a certain level, determined by the standard deviation of the added noise.

The conditioning of the matrix A (about 2e+5) is much better than that of A_0 (about 3e+17), but this is a misleading improvement. In fact, the added noise has a disastrous effect on the problem of estimating x using, e.g., *the TLS method.*

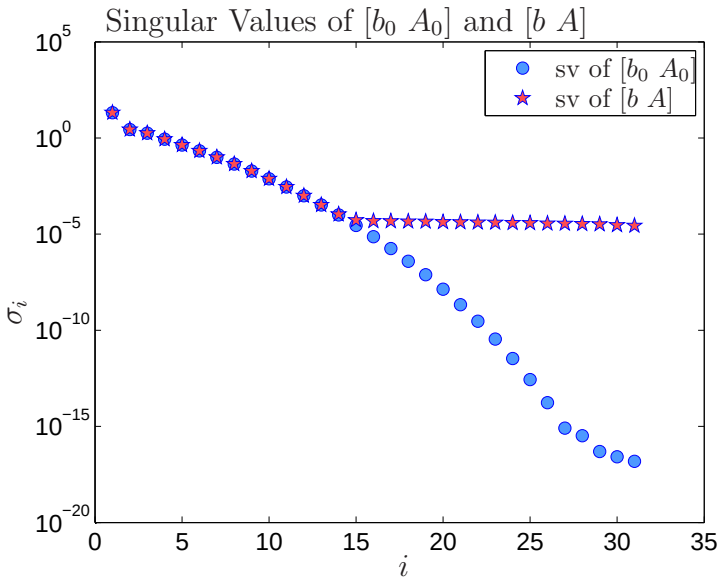


Figure 2: Singular values of $[b_0\ A_0]$ together with the singular values of the noisy data $[b\ A]$.

Figure 3 shows the true solution x_0 together with the TLS solution computed from the data $\begin{bmatrix} b & A \end{bmatrix}$.

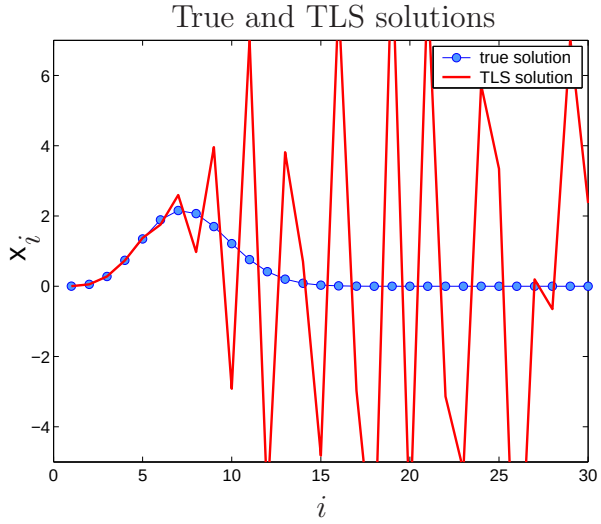


Figure 3: The true solution x_0 is chosen to be the discretization of a smooth function. However, the TLS solution x computed from the noisy data $[b\ A]$ is influenced by the noise and it is far from being a reasonable approximation of the underlying true solution x_0.

2.2 Truncation methods for linear estimation

Truncation methods are an effective way of eliminating unwanted subspaces (such as the noise subspace) from given data sets, using the singular value decomposition (SVD) [8] or other factorizations of the sort. For instance, if for a certain application the noise level is *a priori* known, then it is reasonable to eliminate all information that is below the noise level. It may happen that useful information from the problem is also embedded in that

noise; this is an intuitive explanation why *truncation methods* might be too drastic, and thus they are not always the most appropriate methods for some problem settings. The rich theory and practise of regularization has much more examples of techniques that are better suited for various situations.

Algorithms 1 and 2 outline the simple procedures of truncation in the LS and, respectively, TLS settings.

Algorithm 1 Truncated SVD.

1: Compute the SVD of $A = U'\Sigma'V'^{\top}$, where U' is $m \times n$, V' is $n \times n$ and $U'^{\top}U' = V'^{\top}V' = V'V'^{\top} = I_n$, and Σ' is an $n \times n$ diagonal matrix with the singular values $\sigma'_1 \geq \sigma'_2 \geq \ldots \geq \sigma'_n \geq 0$ on the diagonal;
2: Choose an appropriate truncation level $k \leq n$;
3: Compute the truncated SVD solution

$$x_{\mathrm{TSVD},k} = V'\Sigma'_k U'^{\top} b, \qquad \text{with } \Sigma'_k = \mathrm{diag}\{\sigma'_1, \ldots, \sigma'_k, \underbrace{0, \ldots, 0}_{n-k}\}.$$

Algorithm 2 Truncated TLS.

1: Compute the SVD of $\begin{bmatrix} b & A \end{bmatrix} = U\Sigma V^{\top}$, where U is $m \times (n+1)$, V is $(n+1) \times (n+1)$ and $U^{\top}U = V^{\top}V = VV^{\top} = I_{n+1}$, and Σ is an $(n+1) \times (n+1)$ diagonal matrix with the singular values $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n \geq \sigma_{n+1} \geq 0$ on the diagonal;
2: Choose an appropriate truncation level $k \leq n$.
3: Build $\begin{bmatrix} b_k & A_k \end{bmatrix} := U\Sigma_k V^{\top}$, where Σ_k is Σ with the smallest $n-k$ singular values set to zero, $\Sigma_k = \mathrm{diag}\{\sigma_1, \ldots, \sigma_k, \sigma_{k+1}, 0, \ldots, 0\}$;
4: Solve in the TLS sense the truncated problem $A_k x \approx b_k$ and obtain the truncated TLS solution $x_{\mathrm{TTLS},k}$.

In the next example, we aim to illustrate that truncated TLS is a good regularization method for the type of problem described in Example 1.

Example 2. *We used the SVD of the noisy* $\begin{bmatrix} b & A \end{bmatrix}$ *and we computed truncated TLS solution* ($x_{TTLS,k}$) *for all possible values of* k*, from 1 to* $n = 30$*. We plot in Figure 5 several reconstructed solutions, together with the exact* x_0*. Some facts are to be noted:*

- *the "nontruncated" solution* $x_{TTLS,n}$ *is identical to the TLS solution, and the truncated solutions with large* k *indices are also very much influenced by the noise and resemble the TLS solution;*
- *the truncated solutions with very small* k *level are too simplistic to capture the characteristics of the true solution* x_0*;*
- *there are several values of* k*, such as* $k = 10$ *in Figure 5, for which the truncated TLS solution* $x_{TTLS,k}$ *is a very good reconstruction of the true solution* x_0*.*

The aim of regularization by truncation is thus to appropriately identify a good truncation level, and to construct a truncated solution that can capture the essential features of the unknown true solution, without explicit knowledge about the true solution, and even without *a priori* knowledge about the magnitude of the noise in the data.

3 Core reduction with embedded truncation

3.1 The scaled total least squares formulation

In [15, 16], the notion and properties of core problems in linear algebraic systems are introduced and thoroughly analyzed. The context is solving overdetermined nearly compatible linear systems of equations, written in matrix notation as $Ax \approx b$, where $A \in \mathbb{R}^{m\times n}$ and $b \in \mathbb{R}^m$ are given ($m > n$), and x is an unknown vector. In [15, 16], the classical methods of least squares, total least squares and data least squares are treated in an unified manner, under the more general *scaled total least squares (ScTLS)* formulation. The ScTLS solution is the optimal x that solves the minimization

$$\min_{x,\Delta A,\Delta b} \| \begin{bmatrix} \Delta b & \Delta A \end{bmatrix} \|_F^2, \qquad \text{subject to} \quad (A + \Delta A)x\gamma = \gamma b + \Delta b. \tag{1}$$

Here, γ is a positive scaling parameter that allows the corrections on the matrix A to have a different magnitude compared to the corrections on b. The extreme cases when γ goes to zero or to infinity correspond to the least squares and data least squares problems (*i.e.* corrections *only* on b or *only* on A, respectively), while the particular choice $\gamma = 1$ gives the total least squares problem [21].

The core problem advertised in [16] avoids the nonuniqueness and nongenericity issues, by transforming the original data in A and b to reduced versions A_{11} and b_1, which may be smaller in size, but essential for the approximation problem at hand. All redundant and not useful information (if any) that was present in the data A and b is eliminated by reducing to A_{11} and b_1.

An important computational method for reducing $\begin{bmatrix} b & A \end{bmatrix}$ to $\begin{bmatrix} b_1 & A_{11} \end{bmatrix}$ involves the *bidiagonalization* of $\begin{bmatrix} b & A \end{bmatrix}$. See [16] for a wealth of properties and computational remarks regarding the use of bidiagonalization for computing a core problem.

In well-posed problems, the bidiagonalization process should be stopped when a zero value is encountered, either on the main diagonal (corresponding to a compatible linear system), or on the superdiagonal (incompatible system). This stopping criterion is not suitable in the ill-posed problems case, where an exact zero on the (super)diagonal will never be possible, and thus the bidiagonalization would run up to the end. We analyze several different rules for deciding when to stop the bidiagonalization of A. In all cases, we express the truncation rule as a minimization problem over the truncation level k (where k is the number of columns of the bidiagonally reduced A_{11}), and, moreover, we aim at being able to evaluate the value of these criteria using only the partial bidiagonal reduction obtained after k steps. We summarize the whole procedure of using the core reduction for truncation in Algorithm 3.

Algorithm 3 Algorithm for reduction to a core problem, while monitoring a truncation criterion

1: k = 0
2: **repeat**
3: $k = k + 1$
4: compute the k^{th} bidiagonalization step of $\begin{bmatrix} \gamma b & A \end{bmatrix}$ using either the Householder rotations or the Lanczos bidiagonalization method
5: compute value of truncation criterion at k
6: **until** $(k = n + 1)$ or truncation criterion cost function starts increasing

Remark 1. It is interesting to note that the bidiagonalization of $\begin{bmatrix} b & A \end{bmatrix}$, followed by truncation, was proposed also in [4], a paper that studied the truncated total least squares problem. At that time, the link with the core formulation and the special properties of this reduction in general were not yet put forward.

Remark 2. Another remark concerning bidiagonalization and truncation is related to the partial least squares (PLS) method, which is quite popular in application areas such as chemometrics and chemical engineering. PLS is an alternative to principal component regression (= truncated SVD) that projects observed data onto a lower dimensional subspace; it is designed to handle multicollinearities. One computational algorithm designed for PLS is in fact equivalent to the Lanczos bidiagonalization method, as discussed in [3]. The properties of PLS can thus be explained in view of the fact that the bidiagonalization reduction of the data leads to a core problem.

The motivation of stopping prematurely the bidiagonalization process (as opposed to stopping when a zero diagonal or superdiagonal element is encountered, as it happens in the reduction to a core problem) is explained by the following fact. Bidiagonalization of a matrix can be used in large scale problems in order to obtain a lower rank approximation of the original matrix. If one stops the bidiagonalization process at step k, then a suboptimal lower rank approximation of rank k can be computed from this partial bidiagonalization. As well-known, the best rank k approximation of a matrix in Frobenius norm or 2-norm can be expressed in terms of its SVD (the Eckart-Young-Mirsky theorem [8, 21]); ideally, the *partial SVD* involving the largest k singular values and associated singular vectors are the only required elements that can set up the best rank k approximation.

In the family of direct methods, an algorithm for partial SVD [20] was developed in order to compute efficiently a group of extreme singular values/vectors. However, this method needs to compute first the entire matrix bidiagonalization and it spares computations only at the stage of reducing the bidiagonal matrix to diagonal.

In the family of iterative methods, the Arnoldi/Lanczos methods based on Krylov subspaces are used in the last decades for computing a few (dominant) singular values of large (sparse or structured) matrices. The Lanczos bidiagonalization is a central scheme in this field. The singular values of the partial bidiagonally reduced matrix 'converge' to the dominant singular values of the original matrix. However, the computation of singular vectors is generally not accurate in finite precision arithmetic, and needs to be stabilized using reorthogonalization methods.

In *exact* arithmetic, the direct bidiagonalization method based on Householder transformations and the Lanczos bidiagonalization algorithm are equivalent, as they both give identical (partial) decompositions [6]. More precisely, the reduction of $\begin{bmatrix} b & A \end{bmatrix}$ using Householder reflections corresponds to the Lanczos bidiagonalization of A with starting vector b.

In [19], *a priori* and *a posteriori* approximation error bounds are presented for the lower rank approximation computed using the Lanczos bidiagonalization in finite precision, assuming that a reasonable reorthogonalization procedure is also applied.

4 Choice of the truncation level

In this section, we discuss several methods for choosing the truncation level. For the truncated SVD formulation, many classical model selection methods are easily adapted. Our contribution is to specialize some of the classical methods for truncation parameter selection to the other classes of ScTLS problems; in particular, we focus on the TLS formulation.

In the following, we denote by $\widehat{x}_{\mathrm{TSVD},k}$ the truncated SVD solution of the problem $Ax \approx b$ with truncation level k, and by $\widehat{x}_{\mathrm{TTLS},k}$ the truncated TLS solution with truncation level k. These n-dimensional vectors satisfy exactly the following truncated systems:

$$A\widehat{x}_{\mathrm{TSVD},k} = \widehat{b}'_k, \qquad \widehat{A}_k \widehat{x}_{\mathrm{TTLS},k} = \widehat{b}_k,$$

where $\widehat{b}'_k := U'_k {U'_k}^\top b$ (the $m \times k$ matrix U'_k contains the first k left singular vectors of A), and $\begin{bmatrix} \widehat{b}_k & \widehat{A}_k \end{bmatrix}$ is a certain rank k approximation of $\begin{bmatrix} b & A \end{bmatrix}$.

Let r_k be the residual error obtained for the truncation level k. The expression of r_k depends on the formulation that we use:

$$\begin{aligned} r_k^{\mathrm{LS}} &= \|b - \widehat{b}'_k\|_2, \quad \text{for the LS case,} \\ r_k^{\mathrm{TLS}} &= \left\| \begin{bmatrix} b & A \end{bmatrix} - \begin{bmatrix} \widehat{b}_k & \widehat{A}_k \end{bmatrix} \right\|_F, \quad \text{for the TLS case.} \end{aligned}$$

Another variable of interest is N – the total number of elements in $\begin{bmatrix} b & A \end{bmatrix}$ that are assumed noisy,

$$\begin{aligned} N^{\mathrm{LS}} &= m, \quad \text{for the LS case,} \\ N^{\mathrm{TLS}} &= m(n+1), \quad \text{for the TLS case.} \end{aligned}$$

Computing the residual error r_k is needed in all model selection methods. For the LS case, this is straightforward. We focus now on the TLS case, since we did not yet specify what $\widehat{A}_k$ and $\widehat{b}_k$ are.

We assume that the truncated solution $\widehat{x}_{\mathrm{TTLS},k}$ is computed using the partial reduction to a core problem, as described in Algorithm 3. Thus, $\begin{bmatrix} \widehat{b}_k & \widehat{A}_k \end{bmatrix}$ is not necessarily the best rank k approximation of $\begin{bmatrix} b & A \end{bmatrix}$. This happens in the case when some of the largest k singular values of A are multiple or when b is orthogonal onto some of the left singular subspaces of A. Consider the truncated core reduction

$$\begin{bmatrix} b_1^k & A_{11}^k \end{bmatrix} = {P_1^k}^\top \begin{bmatrix} b & AQ_1^k \end{bmatrix} \tag{2}$$

with $\begin{bmatrix} b_1^k & A_{11}^k \end{bmatrix} \in \mathbb{R}^{(k+1)\times(k+1)}$ and $P_1^k \in \mathbb{R}^{m\times(k+1)}$, $Q_1^k \in \mathbb{R}^{n\times k}$ having orthonormal columns; let (u'', σ'', v'') be the smallest (that is, the $k+1^{\mathrm{st}}$) singular triplet of $\begin{bmatrix} b_1^k & A_{11}^k \end{bmatrix}$. Then we have [16]

$$\begin{bmatrix} \widehat{b}_k & \widehat{A}_k \end{bmatrix} = \begin{bmatrix} b & A \end{bmatrix} - P_1^k u'' \sigma'' v_1'' \begin{bmatrix} -1 & \widehat{x}_{\mathrm{TTLS},k}^\top \end{bmatrix},$$

where v_1'' is the first element of the vector v''. It is not difficult to reformulate this expression and to obtain the corrections in their typical formula [21, Thm. 3.7] as:

$$\begin{bmatrix} \widehat{b}_k & \widehat{A}_k \end{bmatrix} = \begin{bmatrix} b & A \end{bmatrix} - \frac{(A\widehat{x}_{\mathrm{TTLS},k} - b)\begin{bmatrix} -1 & \widehat{x}_{\mathrm{TTLS},k}^\top \end{bmatrix}}{\|\widehat{x}_{\mathrm{TTLS},k}\|^2 + 1}.$$

The proof involves the relation (2) and the point (c) of the following lemma:

Lemma 1. *The core problem $A_{11}^k y \approx b_1^k$ with optimal TLS solution x_1^k, and the smallest singular triplet of $\begin{bmatrix} b_1^k & A_{11}^k \end{bmatrix}$, denoted by (u'', σ'', v''), satisfy:*

(a) $\dfrac{\|A_{11}^k x_1^k - b_1^k\|_2^2}{\|x_1^k\|_2^2 + 1} = (\sigma'')^2$	*(b)* $\|x_1^k\|_2^2 + 1 = (v_1'')^{-2}$
(c) $\dfrac{(A_{11}^k x_1^k - b_1^k)}{\|x_1^k\|_2^2 + 1} = u''\sigma''v_1''$	*(d)* $\dfrac{{A_{11}^k}^\top (A_{11}^k x_1^k - b_1^k)}{\|x_1^k\|_2^2 + 1} = -(\sigma'')^2 (v_1'')^2 x_1^k$

Moreover, the n-dimensional vector $\widehat{x}_{TTLS,k}$ is linked to the k-dimensional x_1^k by the relation: $\widehat{x}_{TTLS,k} = Q_1^k x_1^k$.

If the reduction in (2) is a partial bidiagonalization, then we have a way of computing the residual norm efficiently at each bidiagonalization step. The following identity can be used:

$$r_k^{\text{TLS}} = \left\| \begin{bmatrix} b & A \end{bmatrix} - \begin{bmatrix} \widehat{b}_k & \widehat{A}_k \end{bmatrix} \right\|_F^2 = \| \begin{bmatrix} b & A \end{bmatrix} \|_F^2 - \| \begin{bmatrix} \beta_1 e_1 & A_{11}^k \end{bmatrix} \|_F^2 + (\sigma'')^2$$

where $\begin{bmatrix} b_1^k & A_{11}^k \end{bmatrix} = \begin{bmatrix} \beta_1 e_1 & A_{11}^k \end{bmatrix}$ is the bidiagonal matrix obtained after k steps. This identity is proven in [4].

We discuss next how several classical model selection methods can be used in the context of choosing a truncation level in TSVD or TTLS.

4.1 The discrepancy principle

This method requires knowledge about the statistical properties of the noise that perturbs the data [14]. Specifically, let's assume that the noise is i.i.d. with zero mean and variance σ^2. The norm of the residual error is computed for the candidate truncation level k and the following criterion is monitored:

$$\min_k \left| \|r_k\|^2 - \sigma^2 N \right|,$$

thus, a k is sought that provides a residual error of the same magnitude as the noise in the data.

4.2 The L-curve

The norm of the truncated solution $\|\widehat{x}_k\|_2$ is plotted against the residual error norm r_k in log-log scale for various k's, and the *corner* in log-log scale is chosen [9]. A version of the *rotated L-curve* [17] can also be designed.

4.3 Generalized cross validation

Generalized cross validation (GCV) [2, 5, 23] can be written as

$$\min_k \frac{\|r_k\|_F^2}{\left(N - p_k^{\text{eff}}\right)^2}, \tag{3}$$

where the *effective number of parameters* p_k^{eff} is the trace of the generalized information matrix, which is the derivative of the reconstructed data model with respect to the noisy data.

In the LS case, only b is considered noisy; therefore, the generalized influence matrix shows how perturbations in the measured output b are reflected as perturbations in the recomputed model $\widehat{b}'_k$. It becomes

$$S^{\mathrm{LS}}(k) = \frac{\partial \widehat{b}'_k}{\partial b} = \frac{\partial U'_k {U'_k}^\top b}{\partial b} = U'_k {U'_k}^\top,$$

giving the obvious statement that

$$p_k^{\mathrm{eff}} = \mathrm{Tr}(U'_k {U'_k}^\top) = k.$$

Indeed, in a linear regression model, the number of effective parameters is exactly the number of *free* model parameters. This is easily explained in terms of the truncated core problem as follows: in truncated SVD, the core problem has size $(k+1) \times k$; we need to solve it in least squares sense and we obtain a k-dimensional solution vector x_1^k. The final solution $\widehat{x}_{\mathrm{TSVD},k}$, though of length n, is obtained directly from x_1^k by multiplication with an orthogonal transformation matrix (which does not depend on b). Thus, the number of effective parameters is k.

We recover the GCV criterion as the minimization with respect to k of the function:

$$GCV^{\mathrm{TSVD}}(k) = \frac{\|Ax_{\mathrm{TSVD},k} - b\|^2}{m-k} = \frac{\|(I - U'_k {U'_k}^\top)b\|^2}{m-k} = \frac{(r_k^{\mathrm{LS}})^2}{m-k}.$$

This criterion is well-known, and implemented, for instance, in the Regularization Tools [10].

In the TLS case, A and b are both considered noisy. The residual error norm in this case is given by

$$r_k^{\mathrm{TLS}} = \left\| \begin{bmatrix} b & A \end{bmatrix} - \begin{bmatrix} \widehat{b}_k & \widehat{A}_k \end{bmatrix} \right\|_F^2 = \frac{\|A\widehat{x}_{\mathrm{TTLS},k} - b\|_2^2}{\|\widehat{x}_{\mathrm{TTLS},k}\|_2^2 + 1}.$$

Due to the nonlinearity (in $\widehat{x}_{\mathrm{TTLS},k}$) of this expression, we need to use the nonlinear generalized influence matrix definition [13], which in our case becomes the Jacobian of the estimated model $\mathrm{vec}\begin{bmatrix} \widehat{b}_k & \widehat{A}_k \end{bmatrix}$ with respect to the noisy data $\mathrm{vec}\begin{bmatrix} b & A \end{bmatrix}$. It is, thus, an $m(n+1) \times m(n+1)$ matrix.

Since this direct differentiation is not easy, it is possible to compute the influence matrix using partial derivatives with respect to some auxiliary model parameter, say a vector y. Then, the influence matrix is written as $S^{\mathrm{TLS}}(k) = JH^{-1}J^\top$, where H is the Hessian of the objective function in y (*i.e.*, the loss function that should be minimized in order to obtain optimal model parameters), and J is the derivative of the final "model" with respect to the model parameter y.

In the truncated core problem with truncation level k, we solve the core system

$$A_{11}^k y \approx b_1^k$$

in TLS sense. This means that we minimize the criterion:

$$\frac{\|A_{11}^k y - b_1^k\|^2}{\|y\|^2 + 1}$$

with respect to y. The optimal solution for y is denoted by x_1^k. Therefore, we choose as auxiliary variable the k-dimensional vector y, and we evaluate the Hessian at its minimizer x_1^k. J is the derivative of the final model $\begin{bmatrix} \widehat{b}_k & \widehat{A}_k \end{bmatrix}$ with respect to y, computed also

at x_1^k. Thus,

$$\begin{aligned} S^{\mathrm{TLS}}(k) &:= JH^{-1}J^\top \\ &:= \frac{\partial}{\partial y}\mathrm{vec}\begin{bmatrix}\widehat{b}_k & \widehat{A}_k\end{bmatrix}\Big|_{y=x_1^k}\left(\frac{\partial^2}{\partial y\partial y^\top}\frac{\|A_{11}^k y - b_1^k\|_2^2}{\|y\|_2^2+1}\Big|_{y=x_1^k}\right)^{-1}\frac{\partial}{\partial y}\mathrm{vec}\begin{bmatrix}\widehat{b}_k & \widehat{A}_k\end{bmatrix}^\top\Big|_{y=x_1^k}. \end{aligned}$$

Let us first compute H:

$$\begin{aligned} H &= \frac{\partial^2}{\partial y\partial y^\top}\frac{\|A_{11}^k y - b_1^k\|_2^2}{\|y\|_2^2+1}\Big|_{y=x_1^k} \\ &= \frac{2}{\|x_1^k\|_2^2+1}\left({A_{11}^k}^\top A_{11}^k - \frac{\|A_{11}^k x_1^k - b_1^k\|_2^2}{\|x_1^k\|_2^2+1}I_k \right. \\ &\quad \left. -4\frac{{A_{11}^k}^\top(A_{11}^k x_1^k - b_1^k){x_1^k}^\top}{\|x_1^k\|_2^2+1} + 4\frac{\|A_{11}^k x_1^k - b_1^k\|_2^2}{(\|x_1^k\|_2^2+1)^2}x_1^k{x_1^k}^\top\right). \end{aligned}$$

The relations given in Lemma 1 help simplifying this expression and make the computation of $S^{\mathrm{TLS}}(k)$ efficient. The matrix H becomes:

$$H = 2(v_1'')^2\left({A_{11}^k}^\top A_{11}^k - (\sigma'')^2 I_k + 8(\sigma'')^2(v_1'')^2 x_1^k{x_1^k}^\top\right).$$

In the case we used a bidiagonalization method, H is a small $(k\times k)$ tridiagonal plus rank-one matrix. Its inversion is easy! Note that the inverse exists: indeed, ${A_{11}^k}^\top A_{11}^k - (\sigma'')^2 I_k$ is positive definite, because $\sigma'' < \sigma_{\min}(A_{11}^k)$ (since the truncated core problem has unique generic solution).

Now, we compute the $m(n+1)\times k$ matrix J:

$$\begin{aligned} J &= \frac{\partial}{\partial y}\mathrm{vec}\begin{bmatrix}\widehat{b}_k & \widehat{A}_k\end{bmatrix}\Big|_{y=x_1^k} = \frac{\partial}{\partial y}\mathrm{vec}\left(\begin{bmatrix}b & A\end{bmatrix} - P_1^k u''\sigma''v_1''\begin{bmatrix}-1 & y^\top Q_1^{k\top}\end{bmatrix}\right)\Big|_{y=x_1^k} \\ &= -\underbrace{\begin{bmatrix}0 & \dots & 0 \\ (P_1^k u''\sigma''v_1'')\otimes Q_1^k\end{bmatrix}}_{k}\begin{matrix}m\\ mn\end{matrix} \end{aligned}$$

Remember that our goal was to evaluate the number of effective parameters, which is the trace of the influence matrix: $p_k^{\mathrm{eff}} = \mathrm{Tr}(JH^{-1}J^\top) = \mathrm{Tr}(J^\top JH^{-1})$. We note that $J^\top J$ simplifies even more:

$$J^\top J = \left((P_1^k u''\sigma''v_1'')^\top(P_1^k u''\sigma''v_1'')\right)\otimes\left({Q_1^k}^\top Q_1^k\right) = (\sigma''v_1'')^2 I_k.$$

Thus, the formula for the effective number of parameters in the truncated core TLS problem becomes:

$$p_k^{\mathrm{eff}} = \frac{1}{2}\mathrm{Tr}\left\{\left(\frac{{A_{11}^k}^\top A_{11}^k}{(\sigma'')^2} - I_k + 8(v_1'')^2 x_1^k{x_1^k}^\top\right)^{-1}\right\}.$$

Finally, we are able to plug in computable formulas for r_k^{TLS} and p_k^{eff} into the GCV function (3).

4.4 Generalized information criteria

A generalized information criterion (GIC) [12] is based on the same elements that we have already discussed for GCV: the residual norm and the number of effective parameters. For some forms of information criteria, the (estimated) noise variance is also required. Note that an estimate for the variance (in the case of Gaussian noise) is $\frac{1}{N-p_k^{\text{eff}}} r_k^2$.

We can write GIC as

$$\min_k \ \log(r_k) + \frac{p_k^{\text{eff}}}{N}. \tag{4}$$

For TSVD, the particular p_k^{eff} is just the number k of free parameters in the model, and GIC gives a form of the classical Akaike information criterion [1],

$$\min_k \ \log\left(\|b - A\widehat{x}_{\text{TSVD},k}\|_2\right) + \frac{k}{N}.$$

5 Example

A Matlab software package for computing core reductions, while monitoring a truncation level, has been implemented and titled "CoRe" [18].

For some simulated ill-posed problems such as the problem described in Example 1, we have some preliminary results that use ill-posed problems from the Regularization Tools [10]. However, not all ill-posed problems provided in the Regularization Tools were solved satisfactory using the truncation methods. We explain this by the fact that not all types of ill-posedness are removable by simple truncation, and more complicated regularization methods should be used in those cases. In general, truncation is good for mildly ill-posed problems, not for very severe ones. Moreover, the studied truncation methods favor solutions of small norms, which might not always be the most appropriate property.

Figure 4 shows the rotated L-curve obtained when applying CoRe with the TLS method to the problem in Example 1.

6 Conclusion

We addressed the approximate linear modeling problems gathered under the roof of the scaled total least squares formulation, in the case when the data is coming from ill-posed problems. We used truncation methods as a form of regularization for the ScTLS problem. We discussed that a good first step in truncation methods is the partial reduction to core problem.

For the model selection techniques for choosing truncation levels, we emphasize the fact that in both the truncated SVD and the truncated TLS problems we obtained closed-forms expressions for each of the classical criteria. For TSVD, these are well-known, but for TTLS these results are new. Their implementation in the CoRe software still waits for more extensive testing.

References

[1] H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, 1974.

[2] P. Craven and G. Wahba. Smoothing noisy data with spline functions: estimating the correct degree of smoothing by the method of generalized cross-validation. *Numerische Mathematik*, 31:377–403, 1979.

[3] L. Elden. Partial least squares vs. Lanczos bidiagonalization I: Analysis of a projection method for multiple regression. *Computational Statistics & Data Analysis*, 46:11–31, 2004.

[4] R. D. Fierro, G. H. Golub, P. C. Hansen, and D. P. O'Leary. Regularization by truncated total least squares. *SIAM Journal on Scientific Computing*, 18(1):1223–1241, 1997.

[5] G. H. Golub, M. Heath, and G. Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21:215–223, 1979.

[6] G. H. Golub and W. Kahan. Calculating the singular values and pseudo-inverse of a matrix. *SIAM Journal on Numerical Analysis*, 2(2):205–224, 1965.

[7] G. H. Golub and C. F. Van Loan. An analysis of the total least squares problem. *SIAM Journal on Numerical Analysis*, 17:883–893, 1980.

[8] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore, Maryland, third edition, 1996.

[9] P. C. Hansen. Analysis of discrete ill-posed problems by means of the L-curve. *SIAM Review*, 32:561–580, 1992.

[10] P. C. Hansen. Regularization Tools, a Matlab package for analysis of discrete regularization problems. *Numerical Algorithms*, 6:1–35, 1994.

[11] P. C. Hansen. *Rank-Deficient and Discrete Ill-Posed Problems. Numerical Aspects of Linear Inversion*. SIAM, Philadelphia, 1997.

[12] S. Konishi and G. Kitagawa. Generalised information criteria in model selection. *Biometrika*, 83(4):875–890, 1996.

[13] J. E. Moody. The *effective* number of parameters: an analysis of generalization and regularization in nonlinear learning systems. In Moody, Hanson, and Lippmann, editors, *Advances in Neural Information Processing Systems*, volume 4, pages 847–854. Morgan Kaufmann, Palo Alto, 1992.

[14] V. A. Morozov. On the solution of functional equations by the method of regularization. *Soviet. Math. Dokl.*, 7:414–417, 1966.

[15] C. C. Paige and Z. Strakoš. Scaled total least squares fundamentals. *Numerische Mathematik*, 91:117–146, 2002.

[16] C. C. Paige and Z. Strakoš. Core problems in linear algebraic systems. *SIAM Journal on Matrix Analysis and Applications*, 27(3):861–875, 2006.

[17] T. Reginska. A regularization parameter in discrete ill-posed problems. *SIAM J. Sci. Comput.*, 17(3):740–749, 1996.

[18] D. M. Sima and S. Van Huffel. CoRe: a compact Matlab tool for regularization of linear systems by core truncations. Manuscript, 2006.

[19] H. D. Simnon and H. Zha. Low-rank matrix approximation using the Lanczos bidiagonalization process with applications. *SIAM Journal on Scientific Computing*, 21(8):2257–2274, 2000.

[20] S. Van Huffel and J. Vandewalle. The partial total least squares algorithm. *Journal of Computational and Applied Mathematics*, 21:333–342, 1988.

[21] S. Van Huffel and J. Vandewalle. *The Total Least Squares Problem: Computational Aspects and Analysis*, volume 9 of *Frontiers in Applied Mathematics*. SIAM, Philadelphia, 1991.

[22] J. M. Varah. Pitfalls in the numerical solution of linear ill-posed problems. *SIAM Journal on Scientific and Statistical Computing*, 4(2):164–176, 1983.

[23] G. Wahba. *Spline Models for Observational Data*. CBMS-NSF Regional Conference Series in Applied Mathematics 59. SIAM, 1990.

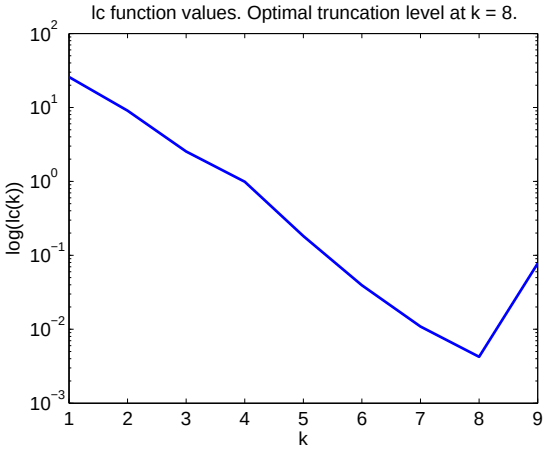


Figure 4: Plot of the rotated L-curve for the problem in Example 1. The minimum value of this criterion at $k = 8$ gives a good reconstruction of the truncated solution compared to the true solution, cf. Figure 5.

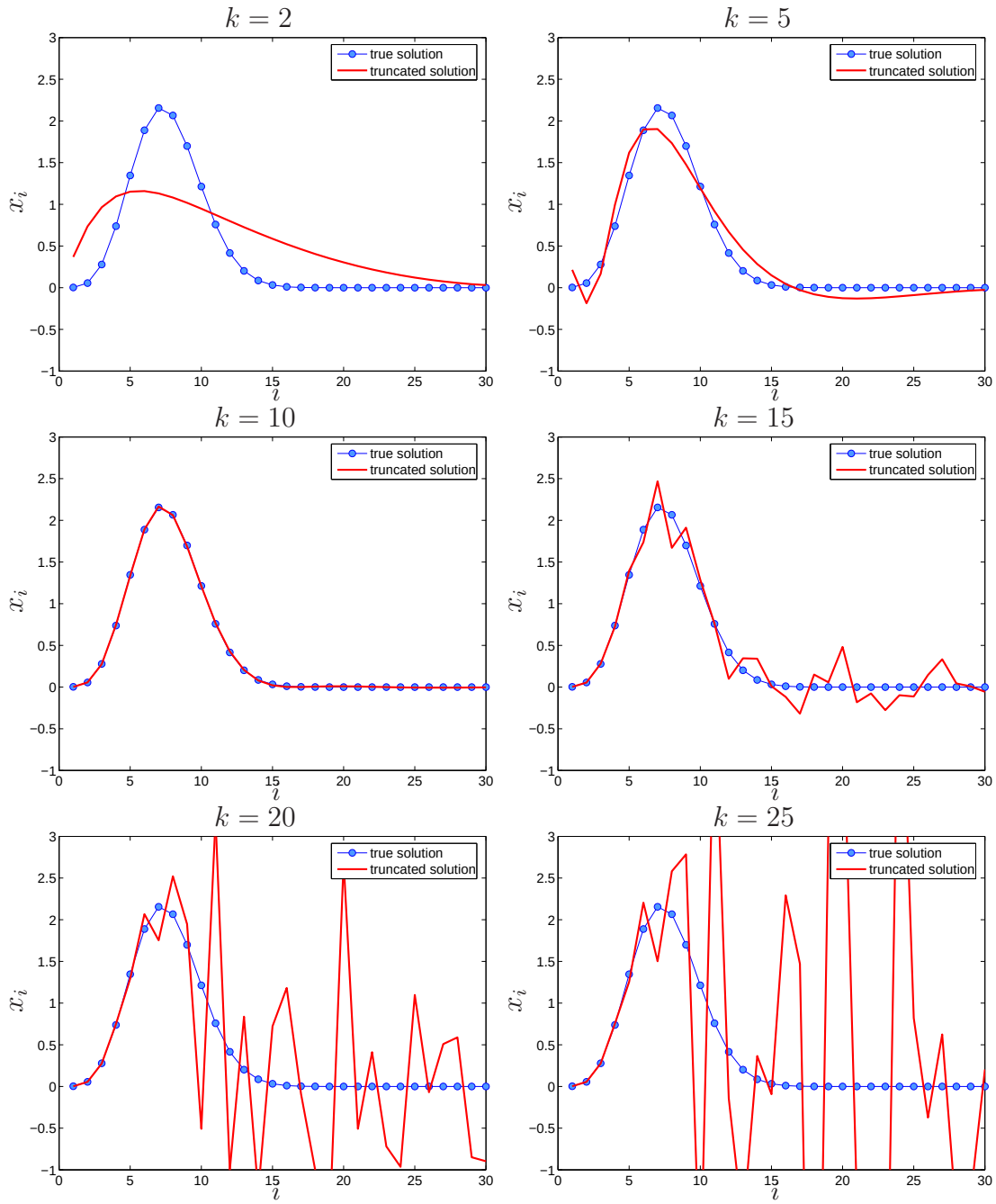


Figure 5: Several truncated TLS solutions x_k, together with the true solution x_0, for the simulation data described in Example 1.