A KRYLOV SUBSPACE METHOD FOR LARGE ESTIMATION PROBLEMS

Michael K. Schneider and Alan S. Willsky

Laboratory for Information and Decision Systems
Massachusetts Institute of Technology
Cambridge, MA

ABSTRACT
Computing the linear least-squares estimate of a high-dimensional random quantity given noisy data requires solving a large system of linear equations. In many situations, one can solve this system efficiently using the conjugate gradient (CG) algorithm. Computing the estimation error variances is a more intricate task. It is difficult because the error variances are the diagonal elements of a complicated matrix. This paper presents a method for using the conjugate search directions generated by the CG algorithm to obtain a converging approximation to the estimation error variances. The algorithm for computing the error variances falls out naturally from a novel estimation-theoretic interpretation of the CG algorithm. The paper discusses this interpretation and convergence issues and presents numerical examples.

1. INTRODUCTION
For certain large linear least-squares estimation problems, especially in medical imaging and remote sensing, one is interested in computing not only estimates but also estimation error variances. The error variances provide important quantitative information concerning the quality of the estimates that can be used in subsequent data analysis and fusion. This paper presents a method for computing both the estimates and the error variances. The method is efficient for a significant number of large estimation problems.

The estimation algorithm presented in this paper has connections to a variety of algorithms for solving linear algebra problems. In particular, one can view the estimation algorithm as a variant of the conjugate gradient (CG) method for solving linear systems of equations. Paige and Saunders have also discussed a variant of the CG algorithm, LSQR, that is capable of computing an approximation to the error variances [4]. Unlike the algorithm proposed here, however, that approximation often does not converge. Other work to which our algorithm is related includes that on Krylov subspace model reduction [3, 8]. These algorithms generate a reduced-order, deterministic model of a dynamic system. Our algorithm also generates a reduced-order model using a Krylov subspace method, but the model is static and stochastic, in nature.

Section 2 derives the algorithm, Section 3 discusses convergence issues, and Section 4 presents numerical examples.

2. ALGORITHM DERIVATION
Consider the problem of forming the linear least-squares estimate (LLSE) of a zero-mean n-dimensional random vector \( x \) given an \( m \)-dimensional linear measurement \( y = Cx + v \) where \( v \) is a zero-mean random variable uncorrelated with \( x \) and \( C \) is a deterministic matrix. Denote the covariance of \( x \) by \( \Lambda_x \) and that of \( v \) by \( R \). The LLSE of \( x \) and associated error covariance are given by \( \hat{x}(y) = \Lambda_x^{-1}C^T \Lambda_y^{-1}y \) and \( \Lambda_x(y) = \Lambda_x - \Lambda_x C \Lambda_y^{-1} C \Lambda_x \), respectively, where \( \Lambda_y = C \Lambda_x C^T + R \) is the covariance of \( y \). If one assumes that multiplication by \( \Lambda_x \) and \( C \) is efficient, then the computation of \( \hat{x} \) and \( \Lambda_x \) is dominated by matrix-vector and matrix-matrix products involving the inverse of \( \Lambda_y \).

The work of performing these multiplies by a matrix inverse could be reduced if one had available a set of linearly independent vectors \( p_i \) that whiten the data. In other words, one desires \( p_i \) such that \( E[p_i^T y v^T y] = p_i^T \Lambda_y p_j \) equals one if \( i = j \) and zero otherwise. Then, the estimate of \( x \) based on \( y \) is the same as that based on \( p_1^T y, \ldots, p_m^T y \). Furthermore, one can use the following recursion to compute, for each \( k \), the LLSE of \( x \) based on \( p_1^T y, \ldots, p_k^T y \), \( \hat{x}(p_1^T y, \ldots, p_k^T y) \), and the associated error variances, which are the diagonal elements of the estimation error covariance, \( \Lambda_x(p_1^T y, \ldots, p_k^T y) \):

\[
\hat{x}(p_1^T y, p_2^T y, \ldots, p_{k+1}^T y) = \\
\hat{x}(p_1^T y, p_2^T y, \ldots, p_k^T y) + \Lambda_x C^T p_{k+1}^T y \\
\Lambda_x(p_1^T y, \ldots, p_k^T y)
\]

(1)
\[
\begin{align*}
(\Lambda_{y}(p_{1}^{T}y, p_{2}^{T}y, \ldots, p_{k+1}^{T}y))_{ii} &= \\
(\Lambda_{x}(p_{1}^{T}y, p_{2}^{T}y, \ldots, p_{k}^{T}y))_{ii} - (\Lambda_{x} C^{T}p_{k+1})_{ii}^{2}
\end{align*}
\] (2)

where \( (\cdot)_{ii} \) denotes the \( i \)th element of a matrix, \( (\cdot)_{i} \) denotes the \( i \)th element of a vector, and \( i \) runs from one to \( n \). The recursion is initialized by setting \( \hat{x}(p_{1}^{T}y) = \Lambda_{x} C^{T}p_{1}^{T}y \) and \( (\Lambda_{x}(p_{1}^{T}y))_{ii} = (\Lambda_{x})_{ii} - (\Lambda_{x} C^{T}p_{1})_{ii}^{2} \) for \( i = 1, \ldots, n \).

One method for recursively choosing \( p_{i} \) that whiten the data is as follows:

\[
p_{1} = \frac{y}{\sqrt{y^{T} \Lambda_{y} y}} \tag{3}
\]
\[
r_{k} = y - \hat{y}(p_{1}^{T}y, \ldots, p_{k}^{T}y) \tag{4}
\]
\[
q_{k+1} = r_{k} - (r_{k}^{T} \Lambda_{y} p_{k}) p_{k} \tag{5}
\]
\[
p_{k+1} = \frac{q_{k+1}}{\sqrt{q_{k+1}^{T} \Lambda_{y} q_{k+1}}} \tag{6}
\]

where \( \hat{y}(p_{1}^{T}y, \ldots, p_{k}^{T}y) = \Lambda_{y}(p_{1}p_{1}^{T} + \cdots + p_{k}p_{k}^{T})y \) is the best linear estimate of \( y \) based on the linear functionals of \( y, p_{1}^{T}y, \ldots, p_{k}^{T}y \) (for the purposes of forming \( \hat{y} \), the \( p_{1}, \ldots, p_{k} \) are viewed as deterministic vectors). The idea here is to first choose \( p_{1} \propto y \) and such that \( \text{Var}(p_{1}^{T}y) = 1 \). The remaining \( p_{2}, \ldots, p_{k} \) are defined by a recursion. First, the error, \( r_{k} \), in estimating \( y \) based on \( p_{1}^{T}y, \ldots, p_{k}^{T}y \) is computed. Then, \( r_{k}^{T}y \) is made uncorrelated with \( p_{k}^{T}y \) to form \( q_{k+1}y \). Finally, \( q_{k+1}y \) is normalized to have unit variance.

That the \( p_{i} \) chosen according to (3)-(6) whiten the data follows from standard results concerning the CG algorithm. This method for picking the \( p_{i} \) is, in fact, the CG algorithm’s method for picking conjugate search directions when computing \( \Lambda_{y}^{-1}y \). In the context of estimation, \( \Lambda_{y}^{-1} \)-conjugate means white; so, the standard theorems for demonstrating that the \( p_{i} \) are \( \Lambda_{y}^{-1} \)-conjugate imply that the \( p_{i}^{T}y \) are white [1, 2].

Much of the theory regarding the CG algorithm exploits the fact that

\[
\text{span}(p_{1}, \ldots, p_{k}) = \text{span}(y, \Lambda_{y}y, \ldots, \Lambda_{y}^{k-1}y), \tag{7}
\]

which is the Krylov subspace of dimension \( k \) associated with the vector \( y \) and matrix \( \Lambda_{y} \). Thus, the proposed estimation algorithm is computing estimates and error variances for the problem of estimating \( x \) based on the projection of the measurements \( y \) onto a Krylov subspace. Note that the novelty of the Krylov subspace estimation algorithm is its ability to exploit an estimation-theoretic interpretation of CG to compute estimation error variances.

### 3. CONVERGENCE ISSUES

Assuming that one can efficiently multiply vectors by \( \Lambda_{y} \) and \( \Lambda_{x} C^{T} \), the proposed method for computing estimates and error variances is efficient provided that one can stop the recursion in (1) and (2) after a few number of steps \( k \) such that \( \hat{x}(p_{1}^{T}y, \ldots, p_{k}^{T}y) \approx \hat{x}(y) \) and \( (\Lambda_{x}(p_{1}^{T}y, \ldots, p_{k}^{T}y))_{ii} \approx (\Lambda_{x}(y))_{ii} \) for \( i = 1, \ldots, n \). The standard convergence results for CG imply that one can stop after a few number of steps and obtain a good approximation to \( \hat{x}(y) \) [1, 2]. However, the convergence of the computed error variances does not immediately follow from these standard results.

Analyzing the convergence of the computed error variances is difficult, in general, but the analysis can be carried out for certain special cases. In particular, one can analyze the situation in which \( x, v, \) and \( y \) are jointly Gaussian random vectors, \( C = I, R = I, \) and \( \Lambda_{x} \) has eigenvalues that decrease geometrically. A description of the three major pieces of the analysis is provided here. More details can be found in [7].

The first piece of the analysis bounds the angle between the span of the first \( k \ conjugate search directions, \[ \text{span}(p_{1}, \ldots, p_{k}), \tag{8} \]

and the dominant eigenvectors of \( \Lambda_{y} \). By Theorem 1 in [6], this angle is rapidly decreasing provided that the data \( y \) has significant components in the directions of all its eigenvectors (see Figure 1). The second piece of the analysis establishes that, with probability one, \( y \) has significant components in the directions of all its eigenvectors. Specifically, the components of \( y \) in the directions of its eigenvectors, divided by the corresponding eigenvalues, are uniformly bounded away from zero. The third piece of the analysis consists of noting that \( \Lambda_{x} \) and \( \Lambda_{y} \) have the same eigenvectors and that the corresponding eigenvalues differ by one. These three facts imply that the \( p_{i} \) are tending to the direction of the small eigenvectors of \( \Lambda_{x} \) so, only the first few \( p_{i} \) are significant in the recursion (1) and (2).
4. NUMERICAL EXAMPLES

Results from running the Krylov subspace estimation algorithm on two synthetic examples are presented here. In both cases, a computer generates a realization of the random vector to be estimated, x, and noisy observations of the vector, y. The estimation algorithm is then run for a certain number of steps to obtain estimates of x and estimation error variances. The number of steps needed is determined by trial and error. Numerical issues surrounding the $\lambda_p$-conjugacy of the $p_i$ are addressed using a non-standard implementation of the CG algorithm that incorporates techniques developed by Parlett and Scott for maintaining the orthogonality of the Lanczos vectors in the Lanczos algorithm [5]. Further implementation details can be found in [7].

Figure 2 depicts results for estimating 1024 samples of a fractional Brownian motion (fBm). The fBm has a Hurst parameter of 0.75 and is scaled to have unit variance at time one. The measurements of the fBm consist of the samples in the intervals $[0, 0.25]$ and $[0.75, 1]$ embedded in independent zero-mean white Gaussian noise with variance 3.2. The solid line in Figure 2(a) is the path of the fBm, and the dashed lines are the computed estimate and the estimate plus and minus the square root of the computed error variances, i.e., the error standard deviations. Figure 2(b) depicts the error standard deviations by themselves, and Figure 2(c) depicts the difference between the error variances computed using the Krylov subspace algorithm and the optimal ones computed using direct methods in MATLAB on a machine with a floating point precision of $2.2 \times 10^{-16}$. The results of Figure 2 were generated using nine steps of the algorithm. That only nine steps were needed indicates that the estimation problem is solved by reducing the 512-dimensional measurement vector to a nine-dimensional one.

Figure 3 depicts results for estimating a stationary Gaussian random field on a $32 \times 32$ toroidal grid. The power spectral density (collection of eigenvalues) of the field is given by

$$
0.3 \sqrt{\rho^2 + j^2} \sum_{-15 \leq k, l \leq 16} 0.3 \sqrt{k^2 + l^2}
$$

where $-15 \leq i, j \leq 16$. The sum in the denominator sets the variance of the field to one. Measurements are made of those random field elements whose coordinates $(i, j)$ are such that $-15 \leq j \leq 16$ and $-15 \leq i \leq -8$ or $9 \leq i \leq 16$. The measurements contain independent zero-mean white Gaussian noise with variance 16. Figure 3(a) depicts the random field to be estimated; 3b, the computed estimates; 3c, the computed error standard deviations; and 3d, the difference between the error variances computed using the Krylov subspace estimation algorithm and the optimal ones computed using direct methods in MATLAB. The results shown in Figure 3 were generated using 50 steps of the algorithm. That 50 steps were used indicates that the estimation problem was solved by reducing the 512-dimensional measurement vector to a 50-dimensional one.

For both of these examples, the Krylov subspace estimation algorithm has efficiently computed estimates and estimation error variances. The error variances are close to the optimal estimation error variances relative to the maximum
prior variance over the domain of the problem. The algorithm has also been tested on estimation problems involving other prior covariances and measurement structures [7].

5. CONCLUSION

This paper presents an estimation-theoretic interpretation of the CG algorithm that has led to a novel method for computing estimation error variances for linear least-squares estimation problems. Analysis and numerical examples establish that the algorithm works and is efficient for certain problems. These promising results encourage further investigation. Potential topics of research include an extensive examination of the range of applicability of the algorithm and the development of an automatic stopping criterion for selecting the number of steps needed to obtain good accuracy.

6. REFERENCES


Figure 3: The surface in (a) depicts a stationary random field on a $32 \times 32$ toroidal grid. The computed LLSE based on noisy sparse measurements of the field is depicted in (b), the computed error standard deviations are depicted in (c), and the difference between the error variances computed using the Krylov subspace estimation algorithm and the optimal ones computed using direct methods in MATLAB are depicted in (d).