Journal of Computational Mathematics, Vol.18, No.4, 2000, 439–448.

DIRECT ITERATIVE METHODS FOR RANK DEFICIENT GENERALIZED LEAST SQUARES PROBLEMS[*1)2)]

Jin-yun Yuan
(*Departamento de Matemática, Universidade Federal do Paraná, Curitiba, Brazil*)
Xiao-qing Jin
(*Faculty of Science and Technology, University of Macau, Macau, China*)

Abstract

The generalized least squares (LS) problem

$$\min_{x\in\mathbf{R}^n}(Ax-b)^TW^{-1}(Ax-b)$$

appears in many application areas. Here W is an $m\times m$ symmetric positive definite matrix and A is an $m\times n$ matrix with $m\ge n$. Since the problem has many solutions in rank deficient case, some special preconditioned techniques are adapted to obtain the minimum 2-norm solution. A block SOR method and the preconditioned conjugate gradient (PCG) method are proposed here. Convergence and optimal relaxation parameter for the block SOR method are studied. An error bound for the PCG method is given. The comparison of these methods is investigated. Some remarks on the implementation of the methods and the operation cost are given as well.

Key words: Rank deficient generalized LS problem, block SOR method, PCG method, convergence, optimal parameter

1. Introduction

The generalized LS problem

$$\min_{x\in\mathbf{R}^n}(Ax-b)^TW^{-1}(Ax-b) \tag{1.1}$$

is frequently found in solving problems from statistics, engineering, economics, image and signal processing. Here $A\in\mathbf{R}^{m\times n}$ with $m\ge n$, $b\in\mathbf{R}^m$ and $W\in\mathbf{R}^{m\times m}$ is

[*] Received September 1, 1997.

[1)] The work of the first author was supported by grant 301035/93-8 from CNPq, Brazil.

[1)] The work of the second author was supported by grant RG010/99-00S/JXQ/FST from University of Macau.

symmetric positive definite. The large sparse rank deficient generalized LS problems appear in computational genetics when we consider mixed linear model for tree or animal genetics [2], [3], [5].

Recently, Yuan [9] and [10], Yuan and Iusem [11] considered direct iterative methods for the problem (1.1) by preconditioned techniques when A has full column rank. They proposed the block SOR-type method and the PCG method for solving the problem (1.1). They also showed that the PCG method is better than the block SOR-type method. However there are few papers to deal with the iterative methods for solving the rank deficient generalized LS problems. It motivates us to propose two iterative methods, the block SOR method and the PCG method, for solving the rank deficient generalized LS problems.

In order to speed up the convergence rate of the block SOR method and the PCG method, the key factor is to find a good preconditioner. Several algorithms were proposed recently in [1], [6] and [7] to find the preconditioners for the LS problems. By using those algorithms, we can develop some good preconditioners of our methods for solving the rank deficient generalized LS problems.

The outline of the paper is as follows. In Section 2, an augmented system for the rank deficient generalized LS problem is given with special transformation. The new system is the base of all work done in this paper. A block SOR method for the problem (1.1) is studied in Section 3. The PCG method is established in Section 4. Comparison of these methods and some remarks on the methods are given in the last section. We always assume that $\text{rank}(A) = k < n$ and W is symmetric positive definite in this paper.

2. Preconditioned Systems

We suppose that A has the following partition

$$A = \begin{pmatrix} A_1 \\ A_2 \end{pmatrix} \tag{2.1}$$

where A_1 is a $k \times n$ full row rank matrix, i.e., $\text{rank}(A_1) = k = \text{rank}(A)$, and A_2 is an $(m-k) \times n$ matrix.

Lemma 2.1[4]. *We have*

$$\mathbf{R}(A^T) = \mathbf{R}(A_1^T) \quad \text{and} \quad \mathbf{N}(A) = \mathbf{N}(A_1)$$

where $\mathbf{R}(B)$ is the range of the matrix B and $\mathbf{N}(B)$ is the null space of B.

For the rank deficient LS problems, since there are many solutions in this case, we are interested in the minimum 2-norm solution which appears in some applications. It is well-known that the minimum 2-norm solution of the problem (1.1) is in $\mathbf{R}(A^T)$, that is in $\mathbf{R}(A_1^T)$ by Lemma 2.1. Hence, we can consider the following linear transformation

$$x = A_1^T y \tag{2.2}$$

with $y \in \mathbf{R}^k$. By using (2.2), the problem (1.1) could be changed into

$$\min_{y\in\mathbf{R}^k} (AA_1^T y - b)^T W^{-1}(AA_1^T y - b) \tag{2.3}$$

which has a unique solution. By the necessary (also sufficient in this case) optimality conditions, the solution of the problem (2.3) is given by the normal equation

$$A_1 A^T W^{-1} A A_1^T y = A_1 A^T W^{-1} b, \tag{2.4}$$

i.e.,

$$A_1 A^T [W^{-1}(b - AA_1^T y)] = 0.$$

Let $r = W^{-1}(b - AA_1^T y)$ be the residual, we then have an augmented system

$$\begin{cases} AA_1^T y + Wr = b \\ A_1 A^T r = 0. \end{cases} \tag{2.5}$$

Since the matrix A has the structure of (2.1), b, r and W also have the corresponding partitions:

$$b = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}, \qquad r = \begin{pmatrix} r_1 \\ r_2 \end{pmatrix}$$

and

$$W = \begin{pmatrix} W_{11} & W_{12} \\ W_{12}^T & W_{22} \end{pmatrix}. \tag{2.6}$$

Here $W_{11} \in \mathbf{R}^{k\times k}$, $W_{22} \in \mathbf{R}^{(m-k)\times(m-k)}$ are symmetric positive definite and $W_{12} \in \mathbf{R}^{k\times(m-k)}$. Thus the system (2.5) can be written as

$$\begin{pmatrix} A_1A_1^T & W_{12} & W_{11} \\ A_2A_1^T & W_{22} & W_{12}^T \\ 0 & A_1A_2^T & A_1A_1^T \end{pmatrix} \begin{pmatrix} y \\ r_2 \\ r_1 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ 0 \end{pmatrix}. \tag{2.7}$$

We therefore have the following theorem.

Theorem 2.1. *The minimum 2-norm solution x of the rank deficient generalized LS problem (1.1) is given by*

$$x = A_1^T y$$

where y is the solution of the system (2.7) *and A_1 is a $k \times n$ submatrix of A with k linearly independent rows. Moreover, the solutions of the problem* (1.1) *are in the form of*

$$x = A_1^T y + z, \qquad \forall z \in \mathbf{N}(A_1).$$

3. The Block SOR Method

We introduce some notations in this section as follows:

$$\bar{A}_1 := A_1 A_1^T, \qquad \bar{A}_2 := A_2 A_1^T,$$

$$\tilde{A}_2 := W_{22}^{-1} \bar{A}_2, \qquad \tilde{W}_{12}^T := W_{22}^{-1} W_{12}^T.$$

We consider the preconditioner D defined as follows,

$$D = \begin{pmatrix} \bar{A}_1 & 0 & W_{11} \\ 0 & W_{22} & 0 \\ 0 & 0 & \bar{A}_1 \end{pmatrix}.$$

Then the preconditioned system of (2.7) with the preconditioner D is given by

$$\begin{pmatrix} I & \bar{A}_1^{-1}(W_{12} - W_{11}\bar{P}^T) & 0 \\ \tilde{A}_2 & I & \tilde{W}_{12}^T \\ 0 & \bar{P}^T & I \end{pmatrix} \begin{pmatrix} y \\ r_2 \\ r_1 \end{pmatrix} = \begin{pmatrix} \bar{A}_1^{-1} b_1 \\ W_{22}^{-1} b_2 \\ 0 \end{pmatrix} \tag{3.1}$$

where $\bar{P} = \bar{A}_2 \bar{A}_1^{-1}$. Thus we can apply the block SOR method to solve the system (3.1) as follows,

$$z^{(k+1)} = \mathcal{L}_\omega z^{(k)} + q.$$

Here

$$\mathcal{L}_\omega =$$

$$\begin{pmatrix} (1-\omega)I & -\omega \bar{A}_1^{-1}(W_{12} - W_{11}\bar{P}^T) & 0 \\ -\omega(1-\omega)\tilde{A}_2 & (1-\omega)I + \omega^2 W_{22}^{-1}\bar{P}W_{12} & \omega W_{22}^{-1}(\omega \bar{P} W_{11} - W_{12}^T) \\ \omega^2(1-\omega)\bar{P}^T\tilde{A}_2 & -\omega E & (1-\omega)I + \omega^2 F \end{pmatrix},$$

$$q = \omega \begin{pmatrix} \bar{A}_1^{-1} b_1 \\ \\ W_{22}^{-1}(b_2 - \omega \bar{P} b_1) \\ \\ \omega \bar{P}^T W_{22}^{-1}(\omega \bar{P} b_1 - b_2) \end{pmatrix} \quad \text{and} \quad z^{(k)} = \begin{pmatrix} y^{(k)} \\ \\ r_2^{(k)} \\ \\ r_1^{(k)} \end{pmatrix}$$

with

$$E = \bar{P}^T [(1-\omega) I + \omega^2 W_{22}^{-1} \bar{P} W_{12}]$$

and

$$F = \bar{P}^T W_{22}^{-1} [W_{12}^T - \omega \bar{P} W_{11}].$$

Therefore, the block SOR Algorithm is given as follows.

Algorithm 3.1.

1. Factorize A_1 and W_{22}, set $x^{(0)} = 0$, $r^{(0)} = 0$.
2. Select a relaxation parameter ω.
3. Iterate for $k = 0,\ 1,\ \ldots,$ until "convergence",

$$\begin{aligned} y^{(k+1)} &= (1-\omega) y^{(k)} + \omega \bar{A}_1^{-1} [b_1 - (W_{12} - W_{11} \bar{P}^T) r_2^{(k)}], \\ r_2^{(k+1)} &= (1-\omega) r_2^{(k)} + \omega W_{22}^{-1} (b_2 - W_{12}^T r_1^{(k)} - \bar{A}_2 y^{(k+1)}), \\ r_1^{(k+1)} &= (1-\omega) r_1^{(k)} - \omega \bar{P}^T r_2^{(k+1)}. \end{aligned}$$

It follows from (3.1) that the associate Jacobi matrix J for the Algorithm 3.1 is

$$J = \begin{pmatrix} 0 & -\bar{A}_1^{-1}(W_{12} - W_{11}\bar{P}^T) & 0 \\ \\ -\tilde{A}_2 & 0 & -\tilde{W}_{12}^T \\ \\ 0 & -\bar{P}^T & 0 \end{pmatrix}. \tag{3.2}$$

Now we present the convergence interval of ω.

Lemma 3.1 *The eigenvalues μ of the associated Jacobi matrix J in* (3.2) *are either real numbers or pure imaginary numbers, i.e., $\mu^2 \in \mathbf{R}$, such that*

$$\mu^2 < 1.$$

Moreover, the spectral radius $\rho(J)$ of the associated Jacobi matrix J is given by

$$\rho^2(J) \le \frac{(1 + \|\bar{P}\|_2^2)\|W\|_2}{\lambda_{\min}(W_{22})} - 1$$

where $\lambda_{\min}(W_{22})$ is the smallest eigenvalue of W_{22}.

Proof. The proof is similar to the proof of Lemma 3.1 in [10].

Theorem 3.1. *Let β and α be the maximum absolute value of the real and pure imaginary eigenvalues of J respectively and let*

$$\omega_b = \frac{2}{1 + \sqrt{1 + \alpha^2 - \beta^2}}.$$

Then Algorithm 3.1 *converges for* ω *with* $0 < \omega < \frac{2}{1+\rho(J)}$ *and*

$$\rho(\mathcal{L}_\omega) = \begin{cases} \left[\frac{\omega\beta+\sqrt{4(1-\omega)+\omega^2\beta^2}}{2}\right]^2, & \text{if } \; 0 < \omega \le \omega_b \\ \left[\frac{\omega\alpha+\sqrt{4(\omega-1)+\omega^2\alpha^2}}{2}\right]^2, & \text{if } \; \omega_b \le \omega < \frac{2}{1+\alpha} \end{cases}.$$

Moreover ω_b *is the optimal relaxation factor and*

$$\rho(\mathcal{L}_{\omega_b}) = \left(\frac{\alpha+\beta}{1+\sqrt{1+\alpha^2-\beta^2}}\right)^2.$$

Proof. The proof is similar to the proof of Theorem 3.2 in [10].

4. The PCG Method

In this section, we propose two PCG algorithms and give an error bound of the PCG method.

4.1 PCG Algorithm I

We will use the preconditioners D_1 defined as follows,

$$D_1 = \begin{pmatrix} A_1A_1^T & 0 & 0 \\ A_2A_1^T & I & 0 \\ 0 & 0 & A_1A_1^T \end{pmatrix},$$

where $\bar{P} = A_2A_1^T(A_1A_1^T)^{-1}$. It follows from (2.7) and

$$D_1^{-1}\begin{pmatrix} A_1A_1^T & W_{12} & W_{11} \\ A_2A_1^T & W_{22} & W_{12}^T \\ 0 & A_1A_2^T & A_1A_1^T \end{pmatrix}\begin{pmatrix} y \\ r_2 \\ r_1 \end{pmatrix} = D_1^{-1}\begin{pmatrix} b_1 \\ b_2 \\ 0 \end{pmatrix}$$

that

$$\begin{pmatrix} I & (A_1A_1^T)^{-1}W_{12} & (A_1A_1^T)^{-1}W_{11} \\ 0 & W_{22} - \bar{P}W_{12} - (W_{12}^T - \bar{P}W_{11})\bar{P}^T & W_{12}^T - \bar{P}W_{11} \\ 0 & \bar{P}^T & I \end{pmatrix}\begin{pmatrix} y \\ r_2 \\ r_1 \end{pmatrix}$$

$$= \begin{pmatrix} (A_1A_1^T)^{-1}b_1 \\ \\ b_2 - \bar{P}b_1 \\ \\ 0 \end{pmatrix}. \qquad (4.1)$$

Therefore from last equation of (4.1), it holds that

$$A_1A_1^Ty = b_1 - (W_{12} - W_{11}\bar{P}^T)r_2, \qquad (4.2)$$

$$(\,\bar{P}, \quad -I\,)\,W \begin{pmatrix} \bar{P}^T \\ -I \end{pmatrix} r_2 = b_2 - \bar{P}b_1 \qquad (4.3)$$

and

$$r_1 + \bar{P}^Tr_2 = 0.$$

It is evident that the matrix in the left hand side of (4.3) is symmetric positive definite. Hence the CG method can be applied to the reduced system (4.3) and the method converges in at most $m-k$ steps. We present next the PCG algorithm I for the rank deficient generalized LS problem (1.1).

Algorithm 4.1

1. Factorize A_1, and set initial values $r_2^{(0)} = 0$, $v^{(0)} = b_2 - \bar{P}b_1$, $p^{(0)} = v^{(0)}$.
2. Iterate for $k = 0,\ 1,\ \ldots,$ until $v^{(k+1)} = 0$ (or $\|v^{(k+1)}\| \leq$tolerance),

   $q = (\,\bar{P}, \quad -I\,)\,W \begin{pmatrix} \bar{P}^T \\ -I \end{pmatrix} p^{(k)},$

   $\lambda_k = \frac{\|v^{(k)}\|_2^2}{<p^{(k)},q>},$

   $v^{(k+1)} = v^{(k)} - \lambda_k q,$

   $r_2^{k+1} = r_2^k + \lambda_k p^{(k)},$

   $\alpha_{k+1} = \frac{\|v^{(k+1)}\|_2^2}{\|v^{(k)}\|_2^2},$

   $p^{(k+1)} = v^{(k+1)} + \alpha_{k+1}p^{(k)}.$
3. Solve the extra subsystem

   $A_1A_1^Ty = b_1 + (W_{11}\bar{P}^T - W_{12})r_2^{(l)}$

   where $r_2^{(l)}$ is the solution obtained in step 2.
4. Form the desired solution x by

   $x = A_1^Ty.$

For the error bound of the PCG method, we have the following theorem.

Theorem 4.1. *The error bound of the PCG method when applied to the rank deficient generalized LS problem* (1.1) *is given by*

$$\frac{\|A(x^* - x^{(k)})\|_{W^{-1}}}{\|A(x^* - x^{(0)})\|_{W^{-1}}} \leq 2\frac{[\frac{(1+\alpha^2)\beta-1}{(1+\sqrt{(1+\alpha^2)\beta})^2}]^k}{1+[\frac{(1+\alpha^2)\beta-1}{(1+\sqrt{(1+\alpha^2)\beta})^2}]^{2k}}. \qquad (4.3)$$

Here $\alpha = \|\bar{P}\|_2$, $\beta = \kappa(W)$ *is the condition number of* W, $x^{(0)}$ *is a vector corresponding to an arbitrary initial vector* $r_2^{(0)}$, $x^{(k)}$ *is the* k*-th iterative solution and* x^* *is the true solution of the problem* (1.1).

Proof. The proof is similar to the proof of Theorem 3.4 in [11].

4.2 PCG Algorithm II

It follows from (2.4) that

$$(I, \bar{P}^T)W^{-1}\begin{pmatrix} I \\ \bar{P} \end{pmatrix}(A_1A_1^T)y = (I, \bar{P}^T)W^{-1}b$$

which is also a symmetric positive definite system. Then we can apply the CG method to solve it and obtain the following PCG algorithm II for (1.1).

Algorithm 4.2.

1. Factorize A_1, and set initial values $z^{(0)} = 0$, $v^{(0)} = (\,I, \quad \bar{P}^T\,)\,W^{-1}b$, $p^{(0)} = v^{(0)}$.
2. Iterate for $k = 0,\ 1,\ \dots,$ until $v^{(k+1)} = 0$ (or $\|v^{(k+1)}\| \leq$tolerance),

   $q = (\,I, \quad \bar{P}^T\,)\,W^{-1}\begin{pmatrix} I \\ \bar{P} \end{pmatrix}p^{(k)}$,

   $\lambda_k = \frac{\|v^{(k)}\|_2^2}{<p^{(k)},q>}$,

   $v^{(k+1)} = v^{(k)} - \lambda_k q$,

   $z^{(k+1)} = z^{(k)} + \lambda_k p^{(k)}$,

   $\alpha_{k+1} = \frac{\|v^{(k+1)}\|_2^2}{\|v^{(k)}\|_2^2}$,

   $p^{(k+1)} = v^{(k+1)} + \alpha_{k+1}p^{(k)}$.
3. Solve the extra subsystem

   $A_1A_1^Ty = z^{(l)}$

   where $z^{(l)}$ is the solution obtained in step 2.
4. Form the desired solution x by

   $x = A_1^Ty$.

In Algorithms 4.1 and 4.2, we do not need to compute the matrix $\bar{P} = A_2A_1^T(A_1A_1^T)^{-1}$ effectively. Instead of computing it, we solve the subsystem $A_1A_1^Ty = c$ by some **direct** methods and perform the matrix-vetor product $z = A_2A_1^Ty$. Observe that for many problems, k is much smaller than m, so that the size of of $A_1A_1^T$ is also much smaller than the size of W. In these cases, it is much easier to factorize $A_1A_1^T$ than W.

5. Comparison Results and Some Remarks

In the previous sections, we presented the block SOR method and the PCG methods for the problem (1.1). In this section, we will give the comparison results of these methods. Some remarks on implementation of the methods and the operation cost are given as well. First of all, analogue the proof of Theorem 4.2 and Corollary 4.3 in [11], we have our comparison theorem.

Theorem 5.1. *If the Algorithm* 3.1 *and the Algorithm* 4.1 *are both started with the same vector* $x^{(0)} \in \mathbf{R}^n$, *then*

$$\|b - Ax_{PCG}^{(k)}\|_{W^{-1}} \leq \|b - Ax_{SOR}^{(k+1)}\|_{W^{-1}}, \quad k = 0, 1, \cdots.$$

Here $x_{PCG}^{(k)}$ *and* $x_{SOR}^{(k)}$ *are the* k*-th iterative solutions generated by the Algorithm* 4.1 *and the Algorithm* 3.1 *respectively.*

Next, we give some remarks on implementation of the methods. For all algorithms proposed in previous section, we shall decompose the submatrix A_1^T as

$$A_1^T = Q \begin{pmatrix} R \\ 0 \end{pmatrix}$$

where Q and R are orthogonal and upper triangular respectively. Now we have

$$\bar{P} = A_2 Q \begin{pmatrix} I \\ 0 \end{pmatrix} R^{-T}.$$

Hence we can solve just one triangular system $R^T z = c$ instead of obtaining $(A_1 A_1^T)^{-1}$ explicitly. By this proccess, we just need two triangular solvers at each iteration.

Finally, we consider the operation cost of our algorithms. The Algorithm 4.1 obtains a vector $r_2 \in \mathbf{R}^{m-k}$ by the CG method and then solves the extra subsystem (4.2). The algorithm requires only $(A_1 A_1^T)^{-1}$ and W but not W^{-1}. The Algorithm 4.2 obtains a vector $z \in \mathbf{R}^k$ by the CG method and then solves another extra subsystem $A_1 A_1^T y = z$. The algorithm requires $(A_1 A_1^T)^{-1}$ and W^{-1} which is very expensive. The Algorithm 3.1 obtains a vector $(y^T, r^T)^T \in \mathbf{R}^{m+k}$ by the SOR method. The algorithm requires $(A_1 A_1^T)^{-1}$ and W_{22}^{-1}. Hence the Algorithm 3.1 is more expensive than the Algorithms 4.1 and 4.2. Therefore the Algorithm 4.1 is more efficient in actual applications than the Algorithms 3.1 and 4.2. If the matrix W is diagonal, that is the problem (1.1) is the weighted LS problem, then the Algorithm 4.2 is preferable than the others.

References

[1] Å. Björck and J.Y. Yuan: *Preconditioner for Least Squares Problems by LU Decomposition*, *Lecture Notes on Scientific Computing* (accepted), Springer-Verlag, 1997.

[2] M.D.V. de Resende, D.F. Prates, A. Jesus, and C.K. Yamada: *Melhor predição linear não viciada (BLUP) de valores genéticos no melhoraamento de Pinus*, Boletim de Pesquisa Florestal, Colombo, 30(1995) 1-5.

[3] M.D.V. de Resende, D.F. Prates, C.K. Yamada, and A. Jesus: *Estimação de componentes de variância e predição de valores genéticos pelo método da máxima verossimilhança restrita (REML) e melhor predição linear não viciada (BLUP) em Pinus*,Colombo, 30(1995) 16-31.

[4] C.H. Santos, B.P.B. Silva, and J.Y. Yuan: *Block SOR methods for rank deficient least squares problems*, Technical Report, Department of Mathematics, UFPR, Brazil, The Brazilian National Congress of Computational and Applied Mathematics, Gramada, RGS, 1997.

[5] S. Searle, G. Casella and C. McCulloch: *Variance Components*, Wiley Interscience, 1992.

[6] B.P.B. Silva: *Topics on Numerical Linear Algebra*, Master Dissertation, UFPR, Curitiba, Brazil, 1997.

[7] B.P.B. Silva and J.Y. Yuan: *Preconditioner for least squares problems by QR decomposition*, Technical Report, Department of Mathematics, UFPR, Brazil, 1997.

[8] D.M. Young:*Iterative Solution of Large Linear Systems, Academic Press, 1971.*

[9] J.Y. Yuan: *Iterative Methods for Generalized Least Squares Problems,Ph.D. Thesis, IMPA, Rio de Janeiro, Brazil, December 1993.*

[10] J.Y. Yuan: *Numerical Methods for Generalized Least Squares Problems*, *J. Computational and Applied Mathematics*, 66(1996)571-584.

[11] J.Y. Yuan and A. N. Iusem: *Preconditioned Conjugate Gradient Method for Generalized Least Squares Problems*, *J. Computational and Applied Mathematics*, 71(1996)287-297.

[12] J.Y. Yuan, R.J.B. de Sampaio and W. Sun: *Algebraic relationships between updating and downdating least squares problems*, *Numerical Mathematics a Journal of Chinese Universities* 3(1996) 203-210 (in Chinese).