A VARIABLE PROJECTION METHOD FOR SOLVING SEPARABLE NONLINEAR LEAST SQUARES PROBLEMS

by

Linda Kaufman

DAIMI PB-27
April 1974

Institute of Mathematics University of Aarhus
DEPARTMENT OF COMPUTER SCIENCE
Ny Munkegade - 8000 Aarhus C - Denmark
Phone 06-12 83 55

A VARIABLE PROJECTION METHOD FOR SOLVING SEPARABLE NONLINEAR LEAST SQUARES PROBLEMS

Abstract

Consider the separable nonlinear least squares problem of finding $\underset{\sim}{a} \in R^n$ and $\underset{\sim}{\alpha} \in R^k$ which, for given data (y_i, t_i) $i = 1, \ldots, m$ and functions $\varphi_j(\underset{\sim}{\alpha}, t)$ $j = 1, 2, \ldots, n$ $(m > n)$, minimize the functional

$$r(\underset{\sim}{a}, \underset{\sim}{\alpha}) = || \underset{\sim}{y} - \Phi(\underset{\sim}{\alpha}) \underset{\sim}{a} ||_2^2$$

where $\Phi(\underset{\sim}{\alpha})_{i,j} = \varphi_j(\underset{\sim}{\alpha}, t_i)$. This problem can be reduced to a nonlinear least squares problem involving $\underset{\sim}{\alpha}$ only and a linear least squares problem involving $\underset{\sim}{a}$ only. The reduction is based on the results of Golub and Pereyra, SIAM J. Numerical Analysis, April 1973, and on the trapezoidal decomposition of Φ, in which an orthogonal matrix Q and a permutation matrix P are found such that

$$Q \, \Phi R = \left(\begin{array}{c|c} R & S \\ \hline O & O \end{array} \right) \}\, r$$

where R is nonsingular and upper triangular. To develop an algorithm to solve the nonlinear least squares problem a formula is proposed for the Frechet derivation $D(\Phi_2(\underset{\sim}{\alpha}))$ where Q is partitioned into

$$Q = \begin{bmatrix} Q_1 \\ \cdots \\ Q_2 \end{bmatrix} \begin{array}{l} \}\, r \\ \\ \}\, m-r \end{array} .$$

INTRODUCTION

Golub and Pereyra [4] have recently proposed an algorithm for solving the separable nonlinear least squares problem in which, for given data (y_i, t_i) $i = 1, 2, \ldots, m$ and given functions $\varphi_j(\underset{\sim}{\alpha}, t)$, $j = 1, 2, \ldots, n$ $(m > n)$ where $\underset{\sim}{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_k)$, the vectors $\underset{\sim}{\alpha}$ and $\underset{\sim}{a} = (a_1, a_2, \ldots, a_n)$ are determined which minimize the nonlinear functional

$$r(\underset{\sim}{a}, \underset{\sim}{\alpha}) = \| \underset{\sim}{y} - \Phi(\underset{\sim}{\alpha})\underset{\sim}{a} \|_2^2 \qquad (1.1)$$

where $\Phi(\underset{\sim}{\alpha})_{i,j} = \varphi_j(\underset{\sim}{\alpha}, t_i)$.
This problem often arises in the physical and biological sciences when one wants to fit data in a least squares sense to a nonlinear model as in exponential fitting.

The approach taken by Golub and Pereyra uses the explicit coupling between $\underset{\sim}{\alpha}$ and $\underset{\sim}{a}$ to reduce (1.1) to two subproblems. The first subproblem is a nonlinear least squares problem in the variable $\underset{\sim}{\alpha}$ and involves finding that $\underset{\sim}{\alpha}$ which minimizes

$$r_2(\underset{\sim}{\alpha}) = \| P^{\perp}_{\Phi(\underset{\sim}{\alpha})} \underset{\sim}{y} \|_2^2 \qquad (1.2)$$

where $P^{\perp}_{\Phi(\underset{\sim}{\alpha})} = I - P_{\Phi(\underset{\sim}{\alpha})}$ and $P_{\Phi(\underset{\sim}{\alpha})}$ is the orthogonal projection of the linear space spanned by the columns of $\Phi(\underset{\sim}{\alpha})$. The second subproblem is simply a linear least squares problem of finding $\underset{\sim}{a}$ which minimizes

$$\| \underset{\sim}{y} - \Phi(\underset{\sim}{\alpha})\underset{\sim}{a} \|_2^2 \qquad (1.3)$$

where $\underset{\sim}{\alpha}$ minimizes (1.2). In practice the common methods used to solve nonlinear least squares problems, such as Gauss-Newton [6], Marquardt's scheme [5], and the various quasi-Newton methods [2], have required less time to solve (1.2) than to solve (1.1). The variable projection approach of Golub and Pereyra has also proved more efficient than methods which separate $\underset{\sim}{a}$ and $\underset{\sim}{\alpha}$ but do not use the explicit coupling given in (1.2).

In this paper we will describe a modification of the variable projection approach based on the trapezoidal decomposition of a matrix. According to this decomposition (see [3] and [7]), there exists an orthogonal matrix Q and a permutation matrix P such that for a given $m \times n$ matrix Φ of rank r

$$Q \Phi P = \left(\begin{array}{c|c} R & S \\ \hline O & O \end{array}\right) \tag{1.4}$$

where R is an $r \times r$ nonsingular upper triangular matrix. The matrix Q is not unique. If two matrices Q and Q' both satisfy (1.4) and they are partitioned into

$$Q = \left(\begin{array}{c} Q_1 \\ \hline Q_2 \end{array}\right) \quad \text{and} \quad Q' = \left(\begin{array}{c} Q_1 \\ \hline Q'_2 \end{array}\right)\begin{array}{l} r \\ m-r \end{array} ,$$

Then there exists an $(m-r) \times (m-r)$ orthogonal matrix Z such that

$$ZQ_2 = Q'_2 \ . \tag{1.5}$$

The matrix $P^{\perp}_{\Phi(\alpha)}$ used by Golub and Pereyra can be represented as

$$Q^T \left(\begin{array}{c|c} 0 & 0 \\ \hline 0 & I_{m-r} \end{array}\right) Q \ .$$

If Q is partitioned into

$$Q = \left(\begin{array}{c} Q_1 \\ \hline Q_2 \end{array}\right) \begin{array}{l} r \\ m-r \end{array} ,$$

then $QP^{\perp}_{\Phi(\underset{\sim}{\alpha})} = \left(\begin{array}{c} 0 \\ \hline Q_2 \end{array}\right)$ and since $\|QP^{\perp}_{\Phi(\underset{\sim}{\alpha})}\underset{\sim}{y}\|_2 = \|P^{\perp}_{\Phi(\underset{\sim}{\alpha})}\underset{\sim}{y}\|_2$,

$$r_2(\underset{\sim}{\alpha}) = \|Q_2(\underset{\sim}{\alpha})\underset{\sim}{y}\|_2 \equiv r_3(\underset{\sim}{\alpha}) \ . \tag{1.6}$$

Because of (1.5) the non-uniqueness of Q does not affect $r_3(\underset{\sim}{\alpha})$. Thus, using the same proof given for theorem 2.1 of Golub and Pereyra [4], one can prove

Theorem 1.

Let $r(\underset{\sim}{a}, \underset{\sim}{\alpha})$ and $r_3(\underset{\sim}{\alpha})$ be defined as above. If in the open set $\Omega \subset R^k$, $\Phi(\underset{\sim}{\alpha})$ has constant rank $r \leq \min(m, n)$ and

(a) if $\bar{\underset{\sim}{\alpha}}$ is a critical point (or global minimizer in Ω) of $r_3(\underset{\sim}{\alpha})$ and $\bar{\underset{\sim}{a}}$ is the vector of shortest length which minimizes

$$\|\underset{\sim}{y} - \Phi(\bar{\underset{\sim}{\alpha}})\underset{\sim}{a}\|_2 , \tag{1.7}$$

then $(\bar{\underset{\sim}{a}}, \bar{\underset{\sim}{\alpha}})$ is a critical point of $r(\underset{\sim}{a}, \underset{\sim}{\alpha})$ (or a global minimizer for $\underset{\sim}{\alpha}$ in Ω) and $r(\bar{\underset{\sim}{a}}, \bar{\underset{\sim}{\alpha}}) = r_3(\bar{\underset{\sim}{\alpha}})$,

(b) if $(\bar{\underset{\sim}{a}}, \bar{\underset{\sim}{\alpha}})$ is a global minimizer of $r(\underset{\sim}{a}, \underset{\sim}{\alpha})$ in Ω, then $\bar{\underset{\sim}{\alpha}}$ is a global minimizer of $r_3(\underset{\sim}{\alpha})$ in Ω and $r_3(\bar{\underset{\sim}{\alpha}}) = r(\bar{\underset{\sim}{a}}, \bar{\underset{\sim}{\alpha}})$ if $\bar{\underset{\sim}{a}}$ is defined as in (1.7).

Most of the algorithms designed for nonlinear least squares problems of minimizing $\| f(\underset{\sim}{\alpha}) \|_2$ require the Frechet derivative, $D(f(\underset{\sim}{\alpha}))$. In section 2 we show how to compute $D(Q_2(\underset{\sim}{\alpha})\underset{\sim}{y})$ so that it may be used in one such algorithm.

<u>Section 2</u>

Many of the algorithms for solving the nonlinear least squares problem of minimizing

$$\| \underset{\sim}{f}(\underset{\sim}{\alpha}) \|_2^2$$

are variants of the Gauss–Newton–Marquardt algorithm. In this method one begins with an arbitrary $\underset{\sim}{\alpha}^0$ and determines $\underset{\sim}{\alpha}^{(j)}$ iteratively by setting

$$\underset{\sim}{\alpha}^{(j+1)} = \underset{\sim}{\alpha}^{(j)} + K_j^+ \underset{\sim}{f}_j(\underset{\sim}{\alpha}^{(j)})$$

where

$$\underset{\sim}{f}_j(\underset{\sim}{\alpha}^{(j)}) = \begin{pmatrix} \underset{\sim}{f}(\underset{\sim}{\alpha}^{(j)}) \\ \underset{\sim}{0} \end{pmatrix}_k$$

and K_j^+ is the pseudo inverse as described by Penrose [8] of the matrix

$$K_j = \begin{pmatrix} D(\underset{\sim}{f}(\underset{\sim}{\alpha}^{(j)}) \\ \hline \nu_j X_j \end{pmatrix}$$

where $D(\underset{\sim}{f}(\underset{\sim}{\alpha})$ is the Frechet derivative of $\underset{\sim}{f}(\underset{\sim}{\alpha})$ and X_j is the upper triangular Cholesky factor of a $k \times k$ positive definite matrix and ν_j is chosen so that

$$\| \underset{\sim}{f}(\underset{\sim}{\alpha}^{(j+1)}) \|_2^2 \leq \| \underset{\sim}{f}(\underset{\sim}{\alpha}^{(j)}) \|_2^2 \quad .$$

To use such an algorithm on $r_3(\underset{\sim}{\alpha})$ given in equation (1.6) we need an expression for $D(Q_2(\underset{\sim}{\alpha}))$. This is provided in the following theorem.

Theorem 2.

Given an $m \times n$ matrix A of rank r, there exist an orthogonal matrix Q and a permutation matrix P such that

$$QAP = \left(\begin{array}{c|c} R & S \\ \hline O & O \end{array}\right)$$

where R is an $r \times r$ nonsingular upper triangular matrix and if Q and P are partitioned into

$$Q = \left(\begin{array}{c} Q_1 \\ \hline Q_2 \end{array}\right)\begin{array}{l} r \\ m-r \end{array} \qquad \text{and } P = \underset{\;\;r \qquad n-r}{(P_1 \mid P_2)}$$

then

$$D(Q_2) = -Q_2 D(A)\, P_1 R^{-1} Q_1 . \tag{2.1}$$

Proof

The proof depends on the following properties given in the decomposition:

$$Q_1 A P_1 = R \tag{2.2}$$

$$Q_2 A = 0 \tag{2.3}$$

$$Q_2 Q_2^T = I \tag{2.4}$$

$$Q_1 Q_2^T = 0 \tag{2.5}$$

and the rule for differentiating a product

$$D(A\,B) = D(A)\,B + A\,D(B). \qquad (2.6)$$

Note that statements (2.2) through (2.5) do not specify a unique Q_2, and, as we shall see, it is this freedom that is important.

The derivative $D(Q_2(\underset{\sim}{\alpha}))$ is an $m-r \times k \times m$ tensor. If the tensor by matrix product DQ_2Q^T is partitioned into

$$D(Q_2)Q^T = (X : Y)$$

where X is an $(m-r) \times k \times r$ tensor and Y is an $(m-r) \times k \times (m-r)$ tensor, then

$$D(Q_2(\underset{\sim}{\alpha})) = (X : Y)\,Q = XQ_1 + YQ_2 \qquad (2.7)$$

Our problem reduces to imposing conditions on X and Y.

From (2.3) and (2.6) we obtain

$$D(Q_2)A + Q_2D(A) = 0$$

which by (2.7) gives

$$XQ_1A + YQ_2A + Q_2D(A) = 0$$

or

$$XQ_1AP_1 + Q_2D(A)P_1 = 0$$

so that by (2.3)

$$XR = -Q_2 D(A) P_1$$

i.e.

$$X = -Q_2 D(A) P_1 R^{-1} .$$

Differentiating (2.4) yields

$$Q_2 D(Q_2)^T + D(Q_2) Q_2^T = 0$$

which by (2.4), (2.5), and (2.6) gives

$$Y^T + Y = 0 . \qquad (2.8)$$

Without further specifying conditions on Q_2, we can place no other conditions on Y. We note that (2.8) is satisfied when Y is the 0 matrix.

Golub [9] has pointed out another proof to theorem 2. His proof is based on $D(P_{A(\underset{\sim}{\alpha})})$, which as proved in Golub and Pereyra [4] is given by

$$D(P^{\perp}_{A(\underset{\sim}{\alpha})}) = -P^{\perp}_{A(\underset{\sim}{\alpha})} D(A(\underset{\sim}{\alpha})) A(\underset{\sim}{\alpha})^{-} - (P^{\perp}_{A(\underset{\sim}{\alpha})} D(A(\underset{\sim}{\alpha})) A(\underset{\sim}{\alpha})^{-})^T \qquad (2.9)$$

where A^- is any matrix that satisfies

$$A A^- A = A \text{ and } (A A^-)^T = A A^- .$$

Since $Q_2(\underset{\sim}{\alpha})^T Q_2(\underset{\sim}{\alpha}) = P^{\perp}_{A(\underset{\sim}{\alpha})}$,

$$Q_2(\underset{\sim}{\alpha})^T D(Q_2(\underset{\sim}{\alpha})) + D(Q_2(\underset{\sim}{\alpha}))^T Q_2(\underset{\sim}{\alpha}) = D(P^{\perp}_{A(\underset{\sim}{\alpha})})$$

$$= \quad - Q_2^T(\underset{\sim}{\alpha})(Q_2(\underset{\sim}{\alpha})D(A(\underset{\sim}{\alpha}))A(\underset{\sim}{\alpha})^-) - (Q_2(\underset{\sim}{\alpha})D(A(\underset{\sim}{\alpha}))A(\underset{\sim}{\alpha})^-)^T Q_2(\underset{\sim}{\alpha}) .$$

The general solution to this Recatti equation is given by

$$D(Q_2(\underset{\sim}{\alpha})) = - Q_2(\underset{\sim}{\alpha})D(A(\underset{\sim}{\alpha}))A(\underset{\sim}{\alpha})^- + Z(\underset{\sim}{\alpha})$$

where

$$Q_2(\underset{\sim}{\alpha})^T Z(\underset{\sim}{\alpha}) + Z(\underset{\sim}{\alpha})^T Q_2(\underset{\sim}{\alpha}) = 0 \quad . \tag{2.10}$$

In particular if $Z(\underset{\sim}{\alpha})$ is 0, we get

$$D(Q_2(\underset{\sim}{\alpha})) = - Q_2(\underset{\sim}{\alpha})D(A(\underset{\sim}{\alpha}))A(\underset{\sim}{\alpha})^- .$$

Since $P_1R^{-1}Q_1$ is a candidate for A^- and since our condition on Y in (2.7) is equivalent to that given in (2.10), the two proofs give the same result.

Of course, the Q_2 given in Theorem 2 will not necessarily be the one that a particular computational scheme might produce. However, if a different Q_2, call it Q_2', is computed, this is of little consequence, for if

$$K_j = \left(\frac{- Q_2 D(\Phi(\underset{\sim}{\alpha}))P_1 R^{-1} Q_1 \underset{\sim}{y}}{\nu_j X_j} \right)$$

and $$K_j' = \left(\frac{- Q_2' D(\Phi(\underset{\sim}{\alpha})) P_1 R^{-1} Q_1 \underset{\sim}{y}}{\nu_j x_j} \right) ,$$

equation (1.5) guarantees that

$$K_j^+ \begin{pmatrix} Q_2 \underset{\sim}{y} \\ \underset{\sim}{0} \end{pmatrix} = K_j'^+ \begin{pmatrix} Q_2' \underset{\sim}{y} \\ \underset{\sim}{0} \end{pmatrix}$$

so that the same iterates are generated whether Q_2 or Q_2' is used.

To compute $D(Q_2(\underset{\sim}{\alpha})\underset{\sim}{y})$ we proceed as follows:

(1) Determine $\Phi(\underset{\sim}{\alpha})$ and $D(\Phi(\underset{\sim}{\alpha}))$.

In general $D(\Phi)$ will have many zero columns. It is suggested that only its nonzero columns be stored.

(2) Determine the trapezoidal decomposition of Φ by finding an $m \times m$ orthogonal matrix Q and a permutation matrix P such that

$$Q \Phi P = \left(\begin{array}{c|c} R & S \\ \hline O & O \end{array} \right)$$

where R is an $r \times r$ nonsingular upper triangular matrix. Partition P into

$$P = (\underset{r}{P_1} \mid \underset{n-r}{P_2}) \quad .$$

We suggest that Q be the product of r Householder transformations where the i'th transformation Q_i is designed to zero the last $(m-i)$ elements of the i'th column of $Q_{i-1} Q_{i-2} \cdots Q_1 \Phi$. The matrix Q does not have to be explicitly formed. Only the information required to generate the transformations need be stored. For details concerning the generation and application of Householder transformations see Björck and Golub [1].

(3) Set $\underset{\sim}{v} = Q\underset{\sim}{y}$. $\underset{\sim}{v} = \begin{pmatrix} \underset{\sim}{v}_1 \\ \underset{\sim}{v}_2 \end{pmatrix} \begin{matrix} r \\ m-r \end{matrix}$

The vector $\underset{\sim}{f}$ is $\underset{\sim}{v}_2$.

(4) Solve $R\underset{\sim}{b} = \underset{\sim}{v}_1$ and set $\underset{\sim}{c} = P_1\underset{\sim}{b}$.

(5) Set $X = -Q\, D(\Phi)\underset{\sim}{c}$. $X = \begin{pmatrix} X_1 \\ X_2 \end{pmatrix} \begin{matrix} r \\ m-r \end{matrix}$

The Frechet derivative of $\underset{\sim}{f}$ is X_2, an $(m-r) \times k$ matrix. See Golub and Pereyra [4] concerning the multiplication of a vector by a tensor.

In some cases the amount of computation in step 5 can be reduced if the matrix $D = QD(\Phi)$ is formed. If D is partitioned into

$$D = \begin{pmatrix} D_1 \\ D_2 \end{pmatrix} \begin{matrix} r \\ m-r \end{matrix} ,$$

then $D(\underset{\sim}{f})$ is $D_2\underset{\sim}{c}$. This implementation is more efficient only if $(m^2 - n^2)(p-k)/2 - np < 0$ where p is the number of nonzero columns of D. Thus forming D is a good idea if each nonlinear variable appears only once in the model.

For details about the Marquardt algorithm and a method for efficiently computing $K_j^+ \underset{\sim}{f}_j(\underset{\sim}{\alpha}^{(j)})$ we refer the reader to Golub and Pereyra[4].

Computational results

The algorithm just described was implemented in a FORTRAN program and tested on 2 problems on the CDC 6400 at Aarhus University, Denmark. The results were compared with those given by Jennings' and Osborne's version of the Marquardt-Levenberg algorithm (see [6]) for the function $r(\underset{\sim}{a}, \underset{\sim}{\alpha})$ in (1.1) and $r_2(\underset{\sim}{a}, \underset{\sim}{\alpha})$ in (1.2).

The first problem was the exponential fitting problem given in [4] of fitting data to the model

$$a_1 + a_2 e^{-\alpha_1 t} + a_3 e^{-\alpha_2 t} .$$

Here $\varphi_1(\underset{\sim}{\alpha}, t) = 1$, $\varphi_2(\underset{\sim}{\alpha}, t) = e^{-\alpha_2 t}$ and $\varphi_3(\underset{\sim}{\alpha}, t) = e^{-\alpha_3 t}$. In table 1, the results for this problem are given.

	$r(\underset{\sim}{\alpha})$	$r_2(\underset{\sim}{\alpha})$	$r_3(\underset{\sim}{\alpha})$
Number of Derivative Evaluations	26	4	4
Number of Function Evaluations	32	4	4
Time in seconds	2.913	.613	.485

Table 1.

The second problem was that of fitting gaussians with exponential background also given in [4]. Here the model was

$$a_1 e^{-\alpha_1 t} + a_2 e^{-\alpha_2 (t-\alpha_3)^2} + a_3 e^{-\alpha_4 (t-\alpha_5)^2} + a_4 e^{-\alpha_6 (t-\alpha_7)^2}$$

The results are given below.

	$r(\underset{\sim}{\alpha})$	$r_2(\underset{\sim}{\alpha})$	$r_3(\underset{\sim}{\alpha})$
Number of Derivative Evaluations	9	8	8
Number of Function Evaluations	11	10	10
Time in seconds	6.862	8.61	6.587

Table 2.

The results agreed with our expectations that because the expression for $D(Q_2(\underset{\sim}{\alpha}))$ in (2.1) is simpler than that of $D(P^{\perp}_{\Phi(\underset{\sim}{\alpha})})$ in (2.9), the time for $r_3(\underset{\sim}{\alpha})$ would be consistently less than that for $r_3(\underset{\sim}{\alpha})$.

A greater time differential has been observed when $D(\Phi)$ was a complicated expression which one would not care to form explicitly. For example, one pollution problem was proposed in which $D(\Phi)$ was the product of a tensor B times a matrix A. When using $r_3(\underset{\sim}{\alpha})$ one could form the vector $\underset{\sim}{c} = AP_1R^{-1}Q\underset{\sim}{y}$ and then apply B to $\underset{\sim}{c}$, thereby saving a significant amount of time over performing a tensor by matrix product to form $D(\Phi)$ and applying the result to $P_1R^{-1}Q\underset{\sim}{y}$. When the problem was approached using $r_2(\underset{\sim}{\alpha})$, in order to cover the second term in (2.9), one had to either compute $D(\Phi)$ explicitly so $D(\Phi)^T$ would be available or determine the matrix $B^TP^{\perp}_{\Phi}\underset{\sim}{y}$ and then perform a matrix by matrix multiplication to form $A^TB^TP^{\perp}_{\Phi}\underset{\sim}{y}$. Both alternatives required much work.

__Acknowledgements__. The author would like to thank Victor Pereyra, Universidad Central de Venezuela and Gene H. Golub, Stanford University for their suggestions and encouragement.

References

[1] Å. Björck and G.H. Golub, "Iterative Refinement of Linear Least Squares Solution by Householder Transformations", BIT 7, 1967.

[2] P.E. Gill and W. Murray, "Quasi Newton Methods for Unconstrained Optimization", J. Inst. Maths. Applics. 9, 1972.

[3] G.H. Golub, "Matrix Decompositions and Statistical Calculations", Statistical Computation, Roy C. Milton and John A. Nelder, eds., Academic Press, New York, 1969.

[4] G.H. Golub and V. Pereyra, "The Differentiation of Pseudo Inverses and Nonlinear Least Squares Problems whose Variables Separate", SIAM J. Numer. Anal. 10, April 1973.

[5] D. Marquardt, "An Algorithm for Least Squares Estimation of Nonlinear Parameters", J. Soc. Indust. Appl. Math. 11, June 1963.

[6] M.R. Osborne, "A Class of Nonlinear Regression Problems", Data Representation, R.S. Anderssen and M.R. Osborne, eds., 1970.

[7] G. Peters and J.H. Wilkinson, "The least squares problem and pseudo inverses", Comp. J. 13, 1970.

[8] R.C. Rao and S.K. Mitra, Generalized Inverse of Matrices and its Applications, John Wiley, New York, 1971.

[9] G.H. Golub, personal communication, February 1974.