



1 SUMMARY

To minimise the sum of squares of m given functions, $f_j, j=1,2,\dots,m$, each of n variables, $\mathbf{x}=(x_1,x_2,\dots,x_n), m \geq n$, without the use of any partial derivatives. i.e. to find \mathbf{x} to minimise

$$F(\mathbf{x}) = \sum_{k=1}^M [f_k(x_1, x_2, \dots, x_N)]^2 \quad m \geq n$$

The user must supply an initial approximation to the required minimum position \mathbf{x} , and a subroutine CALFUN, to calculate values of the functions $f_j, j=1,2,\dots,m$, for any values of the variables \mathbf{x} .

For the method see section § 4.

ATTRIBUTES — **Version:** 1.1.0. **Version:** VA05A, VA05AD **Calls:** MB11, CALFUN (a user subroutine). **Original date:** May 1969. **Origin:** M.J.D.Powell, D.A.M.T.P., University of Cambridge,

2 HOW TO USE THE PACKAGE

2.1 Method of Use

The user is required to calculate the objective function $F(x)$.

2.2 The Argument List

The single precision version

```
CALL VA05A(M,N,F,X,H,DMAX,ACC,MAXFUN,IPRINT,W)
```

The double precision version

```
CALL VA05AD(M,N,F,X,H,DMAX,ACC,MAXFUN,IPRINT,W)
```

M is an INTEGER variable that must be set by the user to m the number of functions f_j occurring in the sum of squares. Restrictions: $M \geq N$.

N is an INTEGER variable that must be set by the user to the number of variables n, x_1, x_2, \dots, x_n Restriction: $N > 0$.

X is a REAL (DOUBLE PRECISION in the D version) array of length at least N , whose elements are set to the users estimate of the required variables $\mathbf{x}=(x_1, x_2, \dots, x_n)$, where

$$X(K) = x_k, K=1, 2, \dots, N$$

These values are changed by subroutine VA05A/AD so that on return $X(K)$ will contain an estimate of the required value of $x_k, K=1, 2, \dots, N$.

H is a REAL (DOUBLE PRECISION in the D version) variable must be set by the user to a scalar parameter which allows VA05A/AD to estimate the partial derivatives of each of the functions from difference approximations. In fact the approximation used is

$$\frac{\partial f_i}{\partial x_j} \approx [f_i(x_1, x_2, \dots, x_j + H, \dots, x_N) - f_i(x_1, x_2, \dots, x_j, \dots, x_N)]/H$$

Note that it is a single, scalar parameter, so that it is important that the variables x_1, x_2, \dots, x_n are scaled to be similar in magnitude. see section § 2.3.

DMAX is a REAL (DOUBLE PRECISION in the D version) variable that must be set by the user to a generous estimate of the distance between the users initial estimate, and the required value of \mathbf{x} (see § 2.3 for definition of distance).

ACC is a REAL (DOUBLE PRECISION in the D version) variable that must be set by the user to the accuracy required to the calculated value of the sum of squares $F(\mathbf{x})$. A normal return is made to the, calling program when it is predicted that the best calculated value of $F(\mathbf{x})$ is not more than ACC greater than the minimum value. see § 2.5.

MAXFUN is an INTEGER variable that must be set by the user to the maximum number of function calls made to the subroutine CALFUN. At least $N+1$ such calls are always made.

IPRINT is an INTEGER variable whose value must be set by the user. It specifies the frequency and amount of printing during the execution of VA05A/AD.

IPRINT = 0 No printing. In this case IPRINT will be used as an output argument if an error return occurs.

IPRINT > 0 Values of the functions and variables are printed every IPRINT iterations.

IPRINT < 0 Values of the variables only are printed every $|IPRINT|$ iterations.

This parameter will be used as an output parameter only if the input value is zero and an error return occurs. For details of the errors that may occur see § 2.4, the output values of IPRINT are

IPRINT = 1 the sum of squares $F(\mathbf{x})$ fails to decrease

IPRINT = 2 MAXFUN calls of CALFUN have been made.

W is a REAL (DOUBLE PRECISION in the D version) workspace array of length at least $2*M*N+2*N*N+2*M+5*N$, used by VA05A/AD as working space. On return from VA05A/AD the approximation to the partial derivatives are held in the first $M*N$ locations of W, so that

$$\frac{\partial f_i}{\partial x_j} \approx W(N(i-1)+j).$$

The output arguments are F, X, W and possibly IPRINT. For X, W, IPRINT see above.

F is a REAL (DOUBLE PRECISION in the D version) array of length M at least, which will be set to the values of the functions at the vector of variables returned by the subroutine.

$$F(K) = f_k(x_1, x_2, \dots, x_n) \quad K=1, 2, \dots, M.$$

2.3 User subroutine CALFUN.

The user must supply a subroutine called CALFUN to calculate values of the function at values of x. It must have the form

```
SUBROUTINE CALFUN(M,N,F,X)
```

The arguments for this subroutine have the same significance as for VA05A/AD, so that F, X, are one dimensional arrays. This subroutine must be provided by the user and must calculate the values of the functions $f_j(x_1, x_2, \dots, x_n)$, $j=1, 2, \dots, m$ for the given variables $\mathbf{x}=(x_1, x_2, \dots, x_n)$. These function values must be planted in the array F so that $F(K) = f_k$.

Note that this subroutine must not change the values of M, N, or X.

2.4 Choice of scaling.

The method used requires some step-length control, for which the distance between two estimates (a_1, a_2, \dots, a_n) and (b_1, b_2, \dots, b_n) of the vector (x_1, x_2, \dots, x_n) is

$$\left\{ \sum_{i=1}^N (a_i - b_i)^2 \right\}^{1/2}$$

Therefore it is important that the user shall scale the variables, perhaps by multiplying them by appropriate constants, so that their magnitudes are similar.

2.5 Error returns.

Two conditions will cause VA05A/AD to produce an error return to the calling program.

More than MAXFUN calls of CALFUN have been made and the required minimum value has not been found.

A number of consecutive iterations have failed to reduce the sum of squares $F(\mathbf{x})$. This may happen if ACC has been chosen too small, or if it has been poorly chosen.

A diagnostic will be printed only if IPRINT \neq 0, otherwise the error return is signalled by changing this parameter (see § 2.2. In either case, control is returned to the calling program.

2.6 Accuracy

Note that a normal return from the subroutine occurs when it is predicted that the required accuracy has been obtained. Sometimes this prediction may be wrong, which has been shown by an example with $M=60$, $N=30$, having the property that the correct final value of the sum of squares, $F(\mathbf{x})$, is large. Here the subroutine finishes too soon, but the final answer may be very close to the required one. Experience with the subroutine on about thirty other examples has shown that the convergence criterion is usually adequate.

3 GENERAL INFORMATION

Use of common: none.

Workspace: see argument W.

Other routines called directly: MB11A/AD.

Input/output: No input; any output on unit 6.

4 METHOD

The method is a compromise between three different algorithms for minimising a sum of squares, namely Newton-Raphson, Steepest Descent and Marquardt. Moreover it automatically obtains and improves an approximation to the first derivative matrix following the ideas of Broyden.