

MATRIX ITERATIVE SOLUTION

PURPOSE

Solve a system of linear equations in the following form:

$$A * X = B$$

where A is the matrix of coefficients, B is a vector of constants, and X are the variables to be solved for. After finding the solution, use the method of iterative improvement to provide greater precision to the solution vector.

DESCRIPTION

In practice, the solution of a system of linear equations, finding the inverse of a matrix, and computing the determinant of a matrix are all closely related. Each of these can be computed from the LU decomposition of a matrix. The method of iterative improvement is described in the LINPACK manual. DATAPLOT uses the algorithm they provide. Iterative improvement is used to make the precision of the solution vector equivalent to the original matrix. In most cases, the MATRIX SOLUTION command, which solves a system of linear equations without iterative refinement, should be adequate.

SYNTAX

LET <var2> = MATRIX ITERATIVE SOLUTION <mat> <var1> <SUBSET/EXCEPT/FOR qualification>

where <mat> is a matrix containing the coefficients of the equation;

<var1> is a vector of constants (i.e., the values for the right hand side of the equation);

<var2> is a vector where the resulting matrix solution is saved;

and where the <SUBSET/EXCEPT/FOR qualification> is optional and rarely used in this context.

EXAMPLES

LET X = MATRIX SOLUTION A B

NOTE 1

DATAPLOT uses LINPACK routines to calculate the LU decomposition.

NOTE 2

Matrices used to solve systems of linear equations must have the same number of rows and columns. An error message is printed if they do not.

DEFAULT

None

SYNONYMS

None

RELATED COMMANDS

MATRIX DETERMINANT	=	Compute a matrix determinant.
MATRIX INVERSE	=	Compute a matrix inverse.
MATRIX SIMPLEX SOLUTION	=	Compute a matrix simplex solution.
MATRIX SOLUTION	=	Solve a system of linear equations.
TRIANGULAR SOLUTION	=	Solve an upper triangular system of linear equations.
TRIDIAGONAL SOLUTION	=	Solve a tridiagonal system of linear equations.

REFERENCE

"LINPACK User's Guide," Dongarra, Bunch, Moler, and Stewart, Siam, 1979.

"Numerical Recipes: The Art of Scientific Programming (FORTRAN Version)," Press, Flannery, Teukolsky, and Vetterling, Cambridge University Press, 1989 (chapter 2).

APPLICATIONS

Linear Algebra

IMPLEMENTATION DATE

93/10

PROGRAM

```
. THIS IS THE DATAPLOT PROGRAM FILE CHEMMIX.DP
. PURPOSE--DETERMINE PROPER FEED RATE OF 4 INPUT COMPOSITIONS
. TO YIELD A PRE-SPECIFIED OUTPUT COMPOSITION AT A
. PRE-SPECIFIED OUTPUT RATE.
. ANALYSIS TECHNIQUE--SOLVING A SYSTEM OF 4 LINEAR EQUATIONS
. APPLICATION--CHEMICAL MIXING
. SOURCE--FOGIEL, THE LINEAR ALGEBRA PROBLEM SOLVER RESEARCH
. RESEARCH AND EDUCATION ASSOCIATION, 1980 (PAGE 792)
. GIVEN--INPUT PROPORTIONS--
. IN1 IN2 IN3 IN4
. SULFURIC ACID 80% 0% 30% 10%
. NITRIC ACID 0% 80% 10% 10%
. WATER 16% 20% 60% 72%
. INERT 4% 0% 0% 8%
. GIVEN--DESIRED OUTPUT (2000 LB/HR) AT THE FOLLOWING
. PROPORTIONS--
. SULFURIC ACID 40% (800 LB PER HOUR)
. NITRIC ACID 27% (540 LB PER HOUR)
. WATER 31% (620 LB PER HOUR)
. INERT 2% (40 LB PER HOUR)
. TO FIND X1 = FLOW RATE OF INPUT STREAM 1
. X2 = FLOW RATE OF INPUT STREAM 2
. X3 = FLOW RATE OF INPUT STREAM 3
. X4 = FLOW RATE OF INPUT STREAM 4
. NOTE--TO SET UP THE 4 EQUATIONS (ONE FOR EACH COMPOUND)
. APPLY CONSERVATION OF MASS (MASS BALANCES)
. NOTE--FOR TESTING PURPOSES, THE SOLUTION IS
. 800 600 500 100 LB PER HOUR
. -----START POINT-----
.
. STEP 1-- DEFINE THE LEFT-HAND SIDE.
.
. READ MATRIX A
.80 .00 .30 .10
.00 .80 .10 .10
.16 .20 .60 .72
.04 .00 .00 .08
. END OF DATA
. STEP 2-- DEFINE THE RIGHT-HAND SIDE.
. LET B = DATA 800 540 620 40
. PRINT A B
. STEP 3-- SOLVE THE LINEAR SYSTEM
. LET X = MATRIX ITERATIVE SOLUTION A B
. PRINT X
```

The solution vector 800, 600, 500, 100 is generated.