Appendix C

MATCOM and Selected MATCOM Functions

C.1 MATCOM and Selected MATCOM Functions

C.1.1 What is MATCOM?

MATCOM is a MATLAB-based interactive software package containing the implementation of all the major algorithms of Chapters 4 through 12 of *Numerical Linear Algebra and Applications* by Prof. Biswa Nath Datta.

For each problem considered in this book, there are more than one (in some cases several) algorithms.

By using the programs in MATCOM, students will be able to compare different algorithms for the same problem with respect to accuracy, efficiency, and stability, thereby allowing them to judge for themselves the effectiveness of different algorithms. In other words, they will be able to distinguish between a good and a bad algorithm.

MATCOM is available from www.siam.org/books/ot116.

C.2 How to Use MATCOM

Here is an example.

```
Find a Householder matrix H to zeroing entries in a vector x.
                    houszero.m              7.1
                       /                     \
                      /                       \
              The Program Name        Algorithm or Section Number
```

To use this program in MATLAB, you need to know the input and output variables. To find out these variables, type

```
>> help houszero
input:   vector x
output:  Householder vector u and the number sigma
function [u, sigma] = houszero(x)
```

This produces a vector u defining the Householder matrix H and a scalar sigma such that $Hx = (\text{sigma}, 0, \ldots, 0)^T$.

```
>> x = rand(4,1)

x =

    0.2190
    0.0470
    0.6789
    0.6793

>> [u,sigma] = houszero(x)

u =

    1.7740
    0.0692
    0.9994
    1.0000

sigma =  -0.9862
```

C.3 Chapterwise Listing of MATCOM Programs

CHAPTER 4

Title	Program Name	Algorithm Number
Back substitution	**backsub**	4.3
Forward elimination	**forelim**	4.4

CHAPTER 5

Title	Program Name	Algorithm or Section Number
Triangularization Using Gaussian Elimination without Pivoting	**lugsel**	5.1
Triangularization Using Gaussian Elimination with Partial Pivoting	**parpiv**	5.2
Triangularization Using Gaussian Elimination with Complete Pivoting	**compiv**	5.3

CHAPTER 6

Title	Program Name	Algorithm or Section Number
Solving $Ax = b$ with Partial Pivoting without Explicit Factorization	**linsyswf**	6.2
Cholesky Factorization	**choles**	6.7

Sherman Morrison Formula	**shermor**	6.7.2 (Section)
Inverse by LU Factorization without Pivoting	**inlu**	6.7.3 (Section)
Inverse by Partial Pivoting	**inparpiv**	6.7.3 (Section)
Inverse by Complete Pivoting	**incompiv**	6.7.3 (Section)
Hager's Norm-1 Condition Number Estimator	**hagcond1**	6.5
Iterative Refinement	**iterref**	6.6

CHAPTER 7

Title	**Program Name**	**Algorithm or Section Number**
Creating Zeros in a Vector with a Householder Matrix	**houszero**	7.1
Householder QR Factorization	**housqr**	7.2
Householder QR Factorization of a Nonsquare Matrix	**housqrn**	7.2.2 (Section)
Implicit Multiplication of a Matrix by a Givens Matrix	**pgivmul**	7.3
Creating Zeros in a Vector Using Givens Rotations	**givszero**	7.4
QR Factorization by Givens Rotations	**givqr**	7.5
Classical Gram–Schmidt QR Factorization	**clgrsch**	7.7
Modified Gram–Schmidt QR Factorization	**mdgrsch**	7.8
Orthogonal Projections Using SVD	**orthproj**	7.8.10 (Section)

CHAPTER 8

Title	**Program Name**	**Algorithm or Section Number**
Least-Squares Solution Using Normal Equations	**lsfrnme**	8.1
The Householder–Golub Method for the Full-Rank Least-Squares Problem	**lsfrqrh**	8.2
Least-Squares Solution by MGS	**lsfrmgs**	8.3
Least-Squares Solution for the Rank-Deficient Problem Using SVD	**lssvd**	8.5
Minimum-Norm Least-Squares Solution to Rank-Deficient Least-Squares Problem Using SVD	**minnmsvd**	8.7.3 (Section)
Minimum Norm Solution for the Full-Rank Underdetermined Problem Using Normal Equations	**mnudnme**	8.8 (Section)
Minimum-Norm Solution for the Full-Rank Underdetermined Problem Using QR	**mnudqrh**	8.6
Linear Systems Analogue Least-Squares Iterative Refinement	**lsitrn1**	8.9 (Section)
Iterative Refinement for Least-Squares Solution	**lsitrn2**	8.7

CHAPTER 9

Title	Program Name	Algorithm or Section Number
Power Method	**power**	9.1
Inverse Iteration	**invitr**	9.2
Rayleigh Quotient Iteration	**rayqot**	9.3
Householder Hessenberg reduction	**houshess**	9.5
Givens Hessenberg reduction	**givhess**	9.5.3 (Section)
Sensitivities of Eigenvalues	**senseig**	9.6.2 (Section)
Basic QR Iteration	**qritrb**	9.6
Hessenberg QR Iteration	**qritrh**	9.8.2 (Section)
Explicit Single-Shift QR Iteration	**qritrsse**	9.8.4 (Section)
Explicit Double-Shift QR Iteration	**qritrdse**	9.8.4 (Section)
One Iteration Step of the Implicit Double-Shift QR Iteration	**qritrdsi**	9.7

CHAPTER 10

Title	Program Name	Algorithm or Section Number
Variance-Covariance Matrix Using the SVD	**covsvd**	10.3.3 (Section)
Reduction to Bidiagonal Form	**bidiag**	10.3.4 (Section)

CHAPTER 11

Title	Program Name	Algorithm or Section Number
Hessenberg Triangular Reduction	**hesstri**	11.4.1 (Section)
Inverse Iteration for Eigenvectors of the Pencil (A, B)	**invitrgn**	11.3
Cholesky QR Algorithm for the Symmetric Definite Pencil	**cholqr**	11.4
Simultaneous Diagonalization of a Symmetric Definite Pencil	**simdiag**	11.5
Generalized Rayleigh Quotient Iteration	**genrayqt**	11.6
Quadratic Eigenvalue Problem via Linearization to Standard Eigenvalue Problems	**quadeig1**, **quadeig2**	11.9.3 (Section)

CHAPTER 12

Title	Program Name	Algorithm or Section Number
Jacobi Method	**Jacobi**	12.2 (Section)
Gauss–Seidel Method	**gaused**	12.2 (Section)
Successive Overrelaxation Method	**sucov**	12.2 (Section)

Classical Conjugate Gradient Method	**congrad**	12.5
Incomplete Cholesky Factorization	**icholes**	12.8
No-Fill Incomplete LDL^T Factorization	**nichol**	12.8
Bisection Method for Tridiagonal Symmetric Positive Definite Generalized Eigenvalue Problem	**genstrum**	12.7 (Section)
Symmetric Lanczos Algorithm	**lansym**	12.4

Programs Not Mentioned in This Book

Title	**Program Name**
The Absolute Maximum of a Vector	**absmax**
Interchange Two Vectors	**inter**
Computing the CPU Time	**cputime**