



David S. Gilliam
Department of Mathematics
Texas Tech University
Lubbock, TX 79409

806 742-2566
gilliam@texas.math.ttu.edu
<http://texas.math.ttu.edu/~gilliam>

Mathematics 4330/5344 – # 1

Matlab and Numerical Approximation

1 Introductory Remarks

The primary purpose of this course is to provide the students with the background and necessary experience (by way of lots of practice) to feel comfortable using the interactive software Matlab. Learning any new software presents certain obstacles by way of needing to memorize new syntax and becoming familiar with the operating environment. Even though Matlab is quite easy to use once you get used to it, we must nevertheless spend some time at the beginning to becoming acquainted with the Matlab interface and syntax. This will be accomplished in several ways. First the text for the course is the Fourth Edition of the “Matlab Primer” by Kermit Sigmon. This book contains a brief but fairly complete description of the most elementary aspects of Matlab. Second there are several files in HTML format, developed by L. Schovanec and D. Gilliam which cover many topics in the use of Matlab.

2 Xterminal and Matlab Basics

1. To use the xterminals in the lab you must first login.
Directions will be given in class.
2. Next you must open a “command-tool” window or “xterm” window. Please ask how to do this if you don’t know.
3. The operating system on our computer is UNIX. There will only be a few cammands for the operating system that you will need in this class.

4. For example, lets make a subdirectory for your m4330 (or m5344) work, e.g., type `mkdir m4330` . Of course you must then type (return) or (enter) which has the effect of telling the operating system to do what was previously typed.
5. Now change to the new subdirectory by typing,
`cd m4330` .
6. To run Matlab, at the prompt, simply type `matlab` and return and your interactive matlab session will start.
7. In matlab every object is a complex matrix in which real entries are displayed as real and integer as integer.
8. There are several ways to enter a matrix into Matlabs workspace:

- (a) You can type in the elements

```
A=[2 4 5;2 6 3;-1 6 2]
```

builds a 3×3 matrix. The entries are typed in rows (elements separated by a space or a comma) with a semicolon used to declare the beginning of a new row.

- (b) You could also generate a matrix using the commands `rand(n)` or `rand(n,m)` to generate a random n by n or n by m matrix whose elements are normally distributed in 0 to 1.

- (c) The commands

```
a=fix(10*rand(5))
b=round(10*rand(5))
```

generate 5×5 matrices with integer entries.

9. If $A = [a_{ij}]$ and $B = [b_{ij}]$ are $n \times m$ matrices and $C = [c_{ij}]$ is an $m \times p$ matrix, then we have the following matrix arithmetic operations:

- (a) $A + B = [a_{ij} + b_{ij}]$ and $A - B = [a_{ij} - b_{ij}]$

- (b) $A * C = D$ where D is a $n \times p$ matrix with entries $d_{ij} = \sum_{k=1}^m a_{ik}c_{kj}$. For matrix multiplication the number of *columns of the first* matrix must be the same as the number of *rows of the second*.

- (c) For a number α , the scalar product $\alpha A = [\alpha a_{ij}]$

For example

```

A=[1 2 5 ; -2 1 4]
B=[4 2 0 ; 4 2 -7]
C=[3 6 ; -2 1 ; -4 2]
A+B
A*C
3*A

```

10. The usual rules of positive integer exponents applies for *square* matrices, $A^2 = A * A$, $A^3 = A * A * A$,

$$A^r = \overbrace{A * A * \cdots * A}^r.$$

```

A=[3 1; 5 2]
A^2, A^3

```

11. In general, division of matrices makes no sense. But for *nonsingular* square matrices it is possible to make an interpretation of matrix division. If A is nonsingular, then it has an inverse, i.e., a matrix A^{-1} satisfying $A * A^{-1} = A^{-1} * A = I$ where I is the identity matrix with ones on the main diagonal and zeros elsewhere. The identity matrix plays the same role as the number 1 does for multiplication $A * I = I * A = A$. In Matlab the identity matrix is given by `eye(n)` where n is an integer. If A is nonsingular, then the inverse in matlab is given by `inv(A)` or A^{-1} . In this case we can think of division as $B * A^{-1}$ just as we do with numbers $b \div a = b * a^{-1}$ for numbers with $a \neq 0$.
12. More generally, in this case, you can compute `C=A^(-r)`.

```

A=[3 1; 5 2]
A^(-1), inv(A)
A*A^(-1)

```

```

A=[3 1; 5 2]
C=A^(-2)
C*A^2

```

13. *A matrix has an inverse if and only if its determinant is not zero.* Recall the determinant for a square matrix is a number. You can find the definition of the number in most college algebra books. In Matlab it is easy to compute determinants using the command `det`.

```

A=[3 1; 5 2]
d=det(A)

```

14. Recall for a 2×2 matrix

$$\det \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = a_{11}a_{22} - a_{12}a_{21}.$$

15. The problem of determining when a square matrix has an inverse is not easy to answer. The answer is that precisely the nonsingular matrices have inverses. There are several other characterizations of nonsingular given below. We will consider these properties with two examples

A=[3 1;5 2]

B=[3 1;6 2]

A matrix A is nonsingular if and only if any one of the following hold:

- (a) $\det A \neq 0$

`det(A)`

`det(B)`

- (b) The row reduced echelon form of A is the identity

`rref(A)`

`rref(B)`

- (c) A has an inverse

`inv(A)`

`inv(B)`

- (d) The only solution of the equation $Ax = 0$ is $x = 0$, i.e., the null space is the zero vector. The Matlab command “null” computes a basis for the null space. Note It does not list the zero vector.

`null(A)`

`null(B)`

- (e) The matrix A has full rank. If A is $n \times n$ then the rank of A is n .

`rank(A)`

`rank(B)`

16. Since we can do powers, scalar products and sums of matrices we can consider matrix polynomials. Here is an example. Suppose A is an $n \times n$ matrix, c is a $(m+1)$ component row vector, then the matrix polynomial expression $f = c(m+1) * A^m + c(m) * A^{m-1} + \dots + c(2) * A + c(1) * eye(n)$.

17. Here is an example

A=[3 1;5 2]

c=[-3 2 -1 5]

f=c(4)*A^3+c(3)*A^2+c(2)*A+c(1)*eye(2)

g=A*(A*(c(4)*A+c(3)*eye(2))+c(2)*eye(2)) +c(1)*eye(2)

%Horner's method or synthetic division

18. Recall that a system of n linear equations in n unknowns has the form

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots a_{2n}x_n = b_2 \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots a_{nn}x_n = b_n \end{cases}$$

which can also be written in matrix form as

$$Ax = b$$

where

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}, \quad x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}.$$

19. A system of equations in this form can be solved in Matlab several different ways. One way is to use the “backslash” syntax $x = A \backslash b$.

```
A=[3 1;5 2]    % setup A
b=[2;-9]       % setup b
x=A\b          % solve Ax=b
A*x-b          % check the result
```

20. This system can also be solved by writing the augmented matrix $[A \ b]$ and computing the row reduced echelon form. The last column is the solution.

```
A=[3 1;5 2]    % setup A
b=[2;-9]       % setup b
C=[A b]        % setup augmented matrix
rref(C)         % compute row reduced echelon form
```

21. A system of equations in the form $xA = c$ where x and c are row vectors can be solved in Matlab using the “forwardslash” syntax $x = c/A$.

```
A=[3 1;5 2]    % setup A
c=[2 -9]       % setup c
x=c/A          % solve xA=c
x*A-c          % check the result
```

22. Additional information and examples can be found in the Matlab Primer and/or accessing on-line tutorial files on the web using netscape. To use netscape open another command or xterm window and type *netscape* . In the URL (Universal Resource Locator) window type

<http://texas.math.ttu.edu>

and hit return. There a link is provided to my home page and from there to the matlab sessions.

23. For the assignment below it will be very useful to use the “diary” command in Matlab. The command is used to save input and output into a text file. The syntax *diary on* turns copying on and *diary off* turns copying off. You can turn diary on and off as you please and each time it is turned on the new data will be appended to the current diary file in your subdirectory. The diary file can be loaded into a word processor and edited.

ASSIGNMENT 1

1. Enter the matrices

$$A = \begin{bmatrix} 2 & 4 \\ 1 & 3 \end{bmatrix}, \quad B = \begin{bmatrix} -2 & 1 \\ 0 & 4 \end{bmatrix}, \quad C = \begin{bmatrix} 3 & 1 \\ 2 & 1 \end{bmatrix}$$

and carry out the following:

- (a) Verify that $(A + B) + C = A + (B + C)$.
- (b) Verify that $(AB)C = A(BC)$.
- (c) Verify that $A(B + C) = AB + AC$.
- (d) Decide whether AB is equal to BA .
- (e) Find $(A + B)^2$, $(A^2 + 2AB + B^2)$ and $(A^2 + AB + BA + B^2)$.
- (f) Find $A^2 - B^2$, $(A - B)(A + B)$ and $(A^2 + AB - BA + B^2)$.

2. Enter

$$A = \begin{bmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{bmatrix}, \quad B = \begin{bmatrix} 1/2 & -1/2 \\ -1/2 & 1/2 \end{bmatrix}$$

and do the following:

- (a) Compute A^2 , A^3 , etc. Can you say what A^n will be? Explain why this is true.
- (b) Compute B^2 , Can you explain why this is true. What does this tell you about matrix multiplication that is different from squaring numbers?
- (c) Find AB and BA . What do you learn from this that is not true for multiplication of numbers? (hint: if a is a real number and $a^2 = 0$, then $a = 0$).

3. Find the inverse of the matrices (if they exist) and check that the result is correct by multiplying the matrix times its inverse.

$$\text{a) } A = \begin{bmatrix} -1 & 1 \\ 1 & 0 \end{bmatrix}, \quad \text{b) } B = \begin{bmatrix} 2 & 5 \\ 1 & 3 \end{bmatrix}, \quad \text{c) } C = \begin{bmatrix} 2 & 0 & 5 \\ 0 & 3 & 0 \\ 1 & 0 & 3 \end{bmatrix}, \quad \text{d) } D = \begin{bmatrix} -1 & -3 & -3 \\ 2 & 6 & 1 \\ 3 & 8 & 3 \end{bmatrix}.$$

4. Generate an 8×8 matrix and an 8×1 vector with integer entries by

$$A = \text{round}(10 * \text{rand}(8)), \quad b = \text{round}(10 * \text{rand}(8, 1)),$$

- (a) Use *flops* to count the number of floating point operations needed to solve $Ax = b$ using the `\` notation.
- (b) Reset *flops* to zero and resolve the system using the row reduced echelon form of the augmented matrix $[A \ b]$ (i.e., $U = \text{rref}([A \ b])$). The last column of U (call it y) is the solution to the system $Ax = b$. Count the *flops* needed to obtain this result.
- (c) Which method was more efficient?
- (d) The solutions x and y appear to be the same but if we look at more digits we see that this is not the case. At the command prompt type *format long*. Now look at x and y , e.g., type $[x \ y]$. Another way to see this is to type $x - y$.
- (e) which method is more accurate? To see the answer compute the so-called residuals, $r = b - Ax$ and $s = b - Ay$. Which is smaller?

When you are finished reset format to short – *format short*.

5. Given the matrices

$$A = \begin{bmatrix} 5 & 3 \\ 3 & 2 \end{bmatrix}, \quad B = \begin{bmatrix} 6 & 2 \\ 2 & 4 \end{bmatrix}, \quad C = \begin{bmatrix} 4 & -2 \\ -6 & 3 \end{bmatrix},$$

solve the matrix equations:

- (a) $AX + B = C$,
- (b) $AX + B = X$,
- (c) $XA + B = C$,
- (d) $XA + C = X$.

6. Let $A = \text{round}(10 * \text{rand}(6))$. Change the sixth column as follows. Set

```
B=A'    % (take the transpose of $A$)
```

now type

```
A(:,6)=-sum(B(1:5,:))'
```

Can you explain what this last command does? Compute

```
det(A)
rref(A)
rank(A)
```

Can you explain why A is singular?

7. Let $A = \text{round}(10 * \text{rand}(5))$ and $B = \text{round}(10 * \text{rand}(5))$. Compare the following pairs of numbers.
- (a) $\det(A)$ and $\det(A')$.
 - (b) $\det(A + B)$ and $\det(A) + \det(B)$.
 - (c) $\det(AB)$ and $\det(A)\det(B)$.
 - (d) $\det(A^{-1})$ and $1/\det(A)$.
8. Look at help on `magic` and then compute $\det(\text{magic}(n))$ for $n = 3, 4, 5, \dots, 10$. What seems to be happening? Check $n = 24$ and 25 to see if the patterns still holds. By pattern I mean try to describe in words what seems to be happening to these determinants.