MATLAB Lab for Applied Linear Algebra

Tommy Luckner
Department of Mathematics

Basic Layout

When you open MATLAB you should see a few basic windows.

- The command window is the large window where you will type in commands.
- The workspace shows all the currently defined variables.
- The command history shows all the commands executed in this session.
- At the top, there is a line that shows you the current directory, this directory is where all of your files will be saved. Always check the directory.

Basic workings

- To make a **comment**, type %. Everything typed after % on the same line is a comment. MATLAB will not execute a comment. Comments are useful in reminding yourself what a line or command is for. This becomes more important when you start creating your own programs.
- It is sometimes useful to use the diary command which will save everything in one file in the current directory as long as the diary is open.

  1. Using **diary**
     - To begin diary, type

       ```
       >> diary filename
       ```

       The above command first creates the file titled *filename* (if it has not already been created) and then opens the file. Until diary is turned off, everything will be written to *filename*.
     - To end diary, type

       ```
       >> diary off
       ```

- A semicolon at the end of a line will suppress the output.
- Using the up key allows you to scroll through previously used commands. This is useful when you have a long command and simply made a typo, you can scroll up till the command shows up in the current line then edit.
- To define a variable, say $a = 10$, you simply type, "a=10".

Useful Commands and Built-In Functions

- **sqrt(·)** calculates the square root of the input.
- **norm(·)** calculates the magnitude of the input vector.
- **dot(· , ·)** calculates the dot product of the two input vectors.
- { sin(x) , cos(x) , tan(x), csc(x), sec(x), cot(x) } may all be used in MATLAB. Try typing

  ```
  >>sin(0)
  ```

  into the command line. Note that the value x should be in radians.

- { asin(x) , acos(x), ... } are use for { $\arcsin(x), \arccos(x), \ldots$ } and work in the same way.

- log(x) is used for $\ln(x)$.

Built in Constant

• "pi" is used for π.
• "exp(x)" is used for e^x.
*Note that "pi" is a constant that approximates π in MATLAB. Moreover, functions like sine and cosine are only approximations in MATLAB. So typing

```
>>sin(pi)
```

will not give you exactly 0.

format short and format long

• MATLAB's default setting is "format short." In this mode, MATLAB will return 5 significant digits. To obtain more digits, use the "format long" setting. To do this, just type

```
>>format long
```

into the command line.

Basic Math Operations

The basic math operations such as exponentiation, addition, subtraction, multiplication, ... are type in similar to how you type in MAPLE commands. For example to have MAPLE evaluate the expression $\frac{2*3^2+4(5x+7)}{2}$ is

```
>>(2*3^2+4*(5*x+7))/2
```

Vectors and Matrices

MATLAB works in matrices, so working with matrices is fairly easy in general. To define a vector, $\mathbf{u} = (3, 9)$ or $\mathbf{v} = \begin{pmatrix} -5 \\ 2 \end{pmatrix}$, for example, type.

```
>> u = [3,9] or u = [3 9]
>> v = [-5;2]
```

Defining a matrix is similar. A comma separates terms in a row and a semicolon separates rows. So for example in order to define the matrix $\mathbf{A} = \begin{pmatrix} 1 & 4 \\ 3 & 0 \end{pmatrix}$. You would type in

```
>> A=[1,4;3,0]
```

You may omit the commas using only spaces instead.

In order to find the length of a vector you use the command

```
>> length(u)
```

To find the size of a matrix you use the command

```
>> size(A)
```

this returns two numbers, the first number is the number of rows and the second is the number of columns.

Operations with Matrices

```
>> inv(A) %this gives the inverse of matrix A
>>transpose(A)
```

• We can multiply matrices in MATLAB with simple commands like

```
>>(A*x)+2*b;
```

For example let

$$A = \begin{pmatrix} 1 & 4 \\ 3 & 0 \end{pmatrix} \qquad x = \begin{pmatrix} 2 \\ 0 \end{pmatrix} \qquad b = \begin{pmatrix} 3 \\ 5 \end{pmatrix}.$$

Now find $Ax + 2b$.

• Sometimes you will need or want to do operations on vectors or matrices componentwise. The command

```
>>C=A.*B
```

will give us a new matrix C, where the each entry in C is obtained from the product of the entries in A and B. For example, if

$$A = \begin{bmatrix} 1 & 4 \\ 3 & 0 \end{bmatrix}, B = \begin{bmatrix} -3 & 2 \\ 7 & -1 \end{bmatrix},$$

then typing the command above will give

$$C = \begin{bmatrix} -3 & 8 \\ 21 & 0 \end{bmatrix}.$$

Plotting

• Given 2 vectors x and y of the same length

```
>>plot(x,y,'b*')
```

will plot the ordered pairs $(x(1), y(1)), (x(2), y(2)), \ldots, (x(n), y(n))$ in blue, with a * for each point. The last command, 'b*', is optional. Without this option, the plot command will plot the ordered pairs in blue and connect the points with lines. Other options than star include

```
., :, o, x, +, -, *, -.,--.
```

Let $x = (1, 2, 3, 4)$ and $y = (4, 7, 9, 10)$. Now try to use the plot command to plot these ordered pairs. Matlab will plot the points (1,4), (2,7), (3,9), (4,10).

• To plot multiple order pairs together, just extend the plot command, for example

```
>>plot(x,y1, 'bx', z, y2, 'r', x, y3, 'g+', ... )
```

Let $z = (.5, 1.5, 2.5, 3.5)$ and $y2 = (3, 6, 9, 12)$. Plot the order pairs (x,y) and (z,y2) on the same graph.

• The command

```
>>x=linspace(a,b,n)
```

will generate a vector x of dimension n with values $x(i) = a + (b - a) * \frac{i-1}{n-1}$, i.e. n equally spaced points between a and b. This will be convenient for creating the x-coordinates of the order pairs of your function. For example, let $f(x) = 2x + 2$. If we want to plot $f(x)$ in the x-range -10 to 10 with say 500 points we would type in the following commands.

```
>> x=linspace(-10,10,500); %I would suppress the output as it will be a very long vector.
>> f=2.*x+2;
>> plot(x,f,'b-')
```

• To add a legend to your plot use the command

```
>>legend('name1','name2',....)
```

Getting Help with MATLAB

To get info on how to use a function/command type in

```
>> help command
```

For example to get help with the plot command you can type in

```
>> help plot
```

You can also search for info in the "Search Documentation" box in the top right side corner.

In-Class Exercise

1. Turn the diary on (with a filename of your choosing) and begin your file with a comment line including both your name and "Lab 0".
2. Define the vectors $\mathbf{u} = (5, 7, 1)$, $\mathbf{v} = (8, -8, 6)$, $\mathbf{w} = (0, -5, 3)$, and $\mathbf{p} = 4 \cdot \mathbf{u} - 2 \cdot \mathbf{v} - 2 \cdot \mathbf{w}$.
3. Compute the dot products: $\mathbf{u} \cdot \mathbf{p}$, $\mathbf{p} \cdot \mathbf{v}$, $\mathbf{u} \cdot \mathbf{v}$, and $\mathbf{u} \cdot \mathbf{w}$.
4. Compute the length of vector p.
5. 5. Turn the diary off and print your work.