

Matlab for Math 240: Week 1

1 Introduction and Set Up

Matlab (short for Matrix Laboratory) was initially designed as a front end for linear algebra, and has evolved into one of the main pieces of software for applied mathematicians and engineers.

Matlab is installed on all of the computers in the Mathematics Computer Lab, so you'll need an account on them to work with Matlab (see your instructor if you need an account or have forgotten your password).

To **find the Matlab icon**, go to the icon in the upper left part of your screen- This should allow you to search your computer. Type in Matlab and you should see the icon (if you don't, ask the student lab assistant or send your instructor a quick email).

1.1 Matlab for your home machine

Unfortunately, our site license doesn't allow for students to use the College license on personal computers. If you're planning on either applied mathematics or engineering in your future, you might look into a student edition of Matlab (at the time of this writing, Matlab alone is about \$49, and with a wide array of numerical toolboxes that will take you into graduate school, the cost is \$99).

Alternatively, there is a software package called "Octave" that was designed to be a free version of Matlab in the sense that most Matlab commands will work on it. In the past, the documentation was hard to follow, so we haven't used this for the class yet. It is currently being updated, so if you are an experienced computer user, you might try it out.

1.2 Opening and Saving Matlab

When you open Matlab for the first time, you will see several windows. The center window is where you type commands (so it is the command window). In that center window, you should see a bar with the line:

New to Matlab? See resources for [Getting Started](#)

If you click on the "Getting Started" link, a browser window will come up. If you look at the bottom of the link, you'll find two videos that you should watch.

The opening interface for Octave is very similar to that of Matlab- A main screen with several smaller screens to the left.

1.3 Defining Arrays

An array (or matrix) is defined in Matlab (or Octave) by creating the array, then storing that array in a variable.

Here's an example: We want to put the following array into Matlab, then store that array in the variable A :

$$\left[\begin{array}{ccc|c} 1 & -2 & 1 & 0 \\ 0 & 2 & -8 & 8 \\ -4 & 5 & 9 & -9 \end{array} \right]$$

The matrix is typically entered row-wise. You denote the end of a row with a semicolon or you can "enter". Here are the two examples that do the same thing. Type the following into the command window of Matlab.

```
%This is a comment- Matlab does not evaluate it.  
%Enter the matrix A (two ways below, just choose one, but either works)
```

```
A=[1 -2 1 0;0 2 -8 8;-4 5 9 -9];
```

```
% The semicolon at the end of the line "suppresses output".  
% You might leave it out to see what happens.
```

```
A=[1 -2 1 0  
0 2 -8 8  
-4 5 9 -9];
```

```
% To row reduce the matrix stored in A, type:  
B=rref(A)
```

You should see that the matrix B is:

$$B = \left[\begin{array}{ccc|c} 1 & 0 & 0 & 29 \\ 0 & 1 & 0 & 16 \\ 0 & 0 & 1 & 3 \end{array} \right]$$

And similarly, if we check Example 3, we want to see if the system they give is consistent. That is, we want to see what the row reduced form of the augmented matrix is. Type the following into the command window:

```
A=[0 1 -4 8;2 -3 2 1;5 -8 7 1];  
rref(A)
```

Matlab gives us the following matrix:

$$\left[\begin{array}{ccc|c} 1 & 0 & -5 & 0 \\ 0 & 1 & -4 & 0 \\ 0 & 0 & 0 & 1 \end{array} \right]$$

Is this matrix row equivalent to the matrix given in the textbook? Explain.

Using a “script file”

A script file is a plain text file that Matlab (or Octave) will treat as if you were typing in the text live.

When we do homework in Matlab, write the command in a script file, then to submit your work, you’ll “publish” the script, then send me the output (in PDF).

Here’s an example all the way through:

Exercise 0

1. Given the matrix A below, find the row reduced echelon form.

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

2. Plot the sine function for $x \in [-\pi, 3\pi]$.

SOLUTION: Open a text editor (you can type `edit` in the command window to use Matlab’s editor), then you can type in the commands that will solve the homework. For this problem:

```
%% Exercise 0

%% Part 1: Row reduce the given matrix.

% Here is the matrix A (this is a "comment")

A=[1 2 3 ; 4 5 6 ; 7 8 9 ]

% Now we row reduce:
B=rref(A)

%% Part 2: Plot the sine function.
x=linspace(-pi,3*pi);
y=sin(x);
plot(x,y)
```

Now, save this file with a `.m` suffix- For example, `Ex00.m`.

The next step is to “publish” the script, then make it a PDF file before submitting it on CLEO.

Publishing a script

- In Matlab:

- When you have the editor open (and with your `Ex00.m` file showing), you will see a "Publish" tab at the top. You can go through that, or you can publish directly from the command line (see below).
- To publish directly from the command line, simply type the following in the command window:

```
publish('Ex00.m','pdf'); % Note the single quotes
```

and the resulting PDF file will be in a subfolder with the title `html`.

- In Octave:

In Octave, you can publish using the command line in exactly the same way:

```
publish('Ex00.m','pdf');
```

Again, the resulting PDF file will be in a subfolder with the title `html`.

In either case, please only upload the PDF file (unless I ask you for both the *m*-file and the PDF file).

Matlab Homework, Week 1

- Get Octave set up on your home machine, or use Matlab in the math computer labs.
- Go through the introduction to Matlab.
- Write up the homework solution to Exercise 0; that is, write the M-file, publish the M-file, and upload the PDF file to CLEO (please create a folder called **Week 1**).

The Matlab homework this week is just to get you set up with the software, and have you practice the homework submission process.