MatLab Directions for 18.02

Access MatLab by clicking on MatLab on the Athena screen, or by typing:

```
% add matlab [return]   % matlab [return]
```

Entering matrices and vectors; Basic operations

In MatLab the variables represent matrices and vectors. The symbol = is used to assign
values to the variables. To see how this works, type each of these lines in order; remember:
always hit [return] or [enter] to end a line.

```
A = [1 2 3; 4 5 6; 7 8 9]  (you can use commas instead of spaces: 1,2,3;)
b = [1 0 1]
b′
```

Try a mistake:  
```
C = [1, 2, 3; 4, 5, 6];
```

to correct it, press any arrow key to get the line back.

Sum, difference   
```
A + B, A − B  (matrices must be same size)
```

Product   
```
A * B  (matrices must be compatibly sized)
```

Powers   
```
A^n  (A times itself n times; A must be square)
```

Quotient   
```
left: A\b  (the solution to Ax = b)
right: b/A  (the solution to xA = b)
```

Transpose   
```
A′
```

Inverse   
```
inv(A)
```

Try typing (use the values of A and b above):
```
A + eye(3)  A*b  A*(b′)  A*b′  3*b
```

Special MatLab Operators

Array Operators: Use dots to make component-wise operations. Let  
```
x = [x_1 x_2 ... x_n], 
y = [y_1 ... y_n];
```

```
x. m = [x_1^m ... x_n^m]  (m can be 0)
x. y = [x_1y_1 ... x_ny_n]
f(x) = [f(x_1) ... f(x_n)],  f = sin, cos, log, polynomials, etc.
```

Colon operator   This generates a vector with equally spaced entries; for example,
```
[0 : 2 : 12] = [0 2 4 6 8 10 12];  [2 : -1 : 1.6] = [2.0 1.9 1.8 1.7 1.6]
```

Two-dimensional plots in MatLab

Let  
```
x = [x_1 x_2 ... x_n],  y = [y_1 ... y_n];
```

```
plot (x, y)  (plots the n points (x_i, y_i), joined by solid line segments)
plot (x, y', '−')  (plots the n points, joined by dashed line segments)
plot (x, y', '*')  (plots the n points as individual stars (or dots or circles, etc))
hold  (toggles between on and off (at the start it’s off); when off, the new
      plot replaces the old; when on, the new
      plot is superimposed on the old)
print  (gives a print-out of the current screen plot)
```

Try in order (read L to R; commands are separated by spaces; press [return] after each):
```
x = [0 : .1 : 2]  plot (x, sin(x))  plot(x, cos(x), 's')  hold
plot (x, sin(x), '−')  hold
plot (x, 4 * x. ^ 3)  (plots y = 4x^3; note the need for the array operator)
```
Directions for 3D Graphs in MatLab

To plot the 3D graph of \( z = f(x, y) \), you specify:

**the grid** \((x_i, y_j)\) of lattice points: give the vectors \( x = [x_1 \ldots x_n] \) and \( y = [y_1 \ldots y_n] \).

Example: To make a grid with spacing .1, over the interval \([-2, 2]\) on both axes, type (in what follows, \( \gg \) is the matlab prompt; don’t type it — type the semicolon at the end so Matlab won’t print out all the numbers — remember [return] at the end)

\[
\begin{align*}
\gg & \ x = [-2 : .1 : 2]; \\
\gg & \ y = [-2 : .1 : 2]; \\
\gg & \ [x, y] = \text{meshgrid}(x, y);
\end{align*}
\]

**the function** \( z = f(x, y) \) For example, to graph the function \( f(x, y) = x^2 - y^2 \), type

\[
\gg \ z = x.^2 - y.^2;
\]

**plot the graph** either as a mesh of lines, or as a filled-in surface (the color indicates the value of \( z \), i.e., the height of the graph above the \( xy \)-plane); type first

\[
\begin{align*}
\gg & \ \text{mesh}(x, y, z) \text{ then } \gg \ \text{surf}(x, y, z)
\end{align*}
\]

**change the viewpoint** To change the viewpoint (i.e., rotate the graph left or right, up or down), type

\[
\gg \ \text{rotate3d}
\]

then place the mouse cursor in the graph region, hold down left button, move mouse, release button. The two numbers on the screen are the **azimuth**: angle in degrees from the negative \( y \)-axis to the line of sight, and the **elevation**, the angle in degrees from the \( xy \)-plane to the line of sight. To turn off rotation, type again: \( \gg \ \text{rotate3d} \)

**hidden lines** Try typing: \( \gg \ \text{hidden} \) (type it again to change back)

**changing scale** To change the \( x \)-axis scale to \([-4, 4]\), the \( y \)-axis to \([-5, 5]\), and the \( z \)-axis to \([-20, 20]\), type

\[
\gg \ \text{axis}([-4 4 -5 5 -20 20])
\]

**level curves** To get a 2D plot with 20 level curves, type: \( \gg \ \text{contour}(x, y, z, 20) \)

**contour curves** To get a 3D plot with 20 contour curves, type: \( \gg \ \text{contour3}(x, y, z, 20) \)