**OCTAVE COMMANDS**

BASICS:

**~=** (not)

**&&** (and)

**II** (or)

**disp(a)** (can be used to print formatted version of a)

MATRICES:

A = [1 2; 3 4] (row;row;row…)

V = [1 2 3] (one row)

V = [1;2;3] (one column)

V = 1: .01 :2 (1 to 2 by steps of .1)

**ones(x, y)** (x by y matrix of all 1’s)

**rand(x, y)** (random matrix, size x by y)

**eye(x)** (identity matrix size x)

**size(A)** (returns 1 x 2 matrix with size of A)

**size(A, 1)** (size of first column)

**length(A)** (size of largest dimension, usually on for vectors)

LOADING DATA:

NOTE: must be in correct directory

**who** (variables in workspace)

**whos** (variables + size)

**clear x** (removes from workspace)

**clear** (removes all variables)

**load filename.dat** (loads variables out of file)

**save filex.mat x** (saves x to filex)

**save filex.text x –ascii** (saves human readable)

MANIPULATING DATA:

**A(3, 2)** (element in 3rd row, 2nd column)

**A(2, :)** (second row of A)

**A = [A, [x;y;z]]** (add column x, y, z)

**A(:)** (put all elements of A into single column vector)

**A + B** (concats matrices next to each other)

**[A;B]** (concats matrices on top of each other)

COMPUTATIONAL OPERATIONS:

**A\*C** (multiply matrices)

**A.\*B** (elementwise multiplication)

**func(v)** (elementwise application of function)

**A’** (A transpose)

**[val, ind] = max(a)** (index, value of row vector)

**find(a < 3)** (indices of true for row vector)

(can also use for matrices)

**magic(N)** (matrix where everything adds to x)

**sum(A)** (sums vector)

**prod(A)** (multiplies vector)

**floor(A)** (rounds down vector)

**max** (can be used to find max of row, column, overall…)

**sum** (ditto – 1 = by column, 2 = by row)

**pinv(A)** (pseudo inverse of A)

PLOTTING DATA:

**plot(x, y**) (plots x vs. y)

**hold on** (holds old plot open to plot another on top)

**xlabel(‘x’)** (labels x, same for y)

**legend()** (plot legend)

**title**() (plot title)

**print –dpng ‘x.png’** (save plot – other formats available)

**close** (closes figure)

**figure(1); plot(x,y)** (specifies Figure 1)

**subplot(1, 2, 1)** (divides plot into 1 x 2 grid, access first element)

**axis([x1 x2 y1 y2])** (sets range)

**clf** (clears figure)

**imagesc(A)** (plot matrix as grid of colors)

**colorbar** (show correspondence)

NOTE: to chain function calls, split with commas (or semicolons to not print)

CONTROL STATEMENTS:

**for i = 1:10,** (can also do for i=indices)

**statement**

**end;** (for loop)

**break, continue** (same)

**i = 1**

**while i <= 5,**

**statement**

**end;** (while loop)

**if i ==6,**

**statement**

**end;** (if statement)

**elseif i == 7,**

**statement**

**else**

**statement**

**end;**

**defining functions** (create a file)

**function y = squareThisNumber(x)**

**y = x^2;**

**function [y1, y2] = blah(x)**

**(RETURN MULTIPLE VALUES)**

**addpath(‘file’)** (add a path to your directory w/ function)

VECTORIZATION:

* Use calls to handy linear algebra libraries to make life faster
* Use instead of for loops