MATLAB CHEAT SHEET

Throughout this document x and y will be either row or column vectors and A will always be a matrix. In all cases the entries are real numbers.

Basics

`clc`	Clear command window
`clear`	Clear all variables
`clf`	Clear all plots
`close all`	Close all plots
`doc function`	Open help page for function
`who`	list variables on workspace
`% This is a comment`	Comments
`ctrl-c`	Abort the current operation. On mac ⌘+ . (the Command key and the period key) will also abort.
`format short`	Display 4 decimal places
`format long`	Display 15 decimal places
`disp('text')`	Print text

Defining and Changing Variables

`a = 3`	Define variable a to be 3
`b = 3e-5`	Define variable b to be $0.00003 = 3\text{e-}5$
`x = [1, 2, 3]`	Set x to be the row vector $[1, 2, 3]$
`x = [1; 2; 3]`	Set x to be the column vector $[1, 2, 3]^T$
`A = [1, 2, 3, 4; 5, 6, 7, 8; 9, 10, 11, 12]`	Set A to be a 3×4 matrix
`x(2) = 7`	Change x from $[1, 2, 3]$ to $[1, 7, 3]$
`A(2,1) = 0`	Change $A_{2,1}$ from 5 to 0

Basic Arithmetic and Functions

`3*4, 7+4, 2-6, 8/3`	multiply, add, subtract and divide
`3^7`	Compute 3^7
`sqrt(5)`	Compute $\sqrt{5}$
`log(3)`	Compute $\ln(3)$
`log10(100)`	Compute $\log_{10}(100)$
`exp(3.5)`	Compute $e^{3.5}$
`abs(-5)`	Compute $\lvert -5 \rvert$
`sin(5*pi/3)`	Compute $\sin(5\pi/3)$
`floor(3.8)`	Compute $\lfloor 3.8 \rfloor$
`ceil(3.8)`	Compute $\lceil 3.8 \rceil$

Constructing Matrices and Vectors

`zeros(12, 5)`	Make a 12×5 matrix of zeros
`ones(12, 5)`	Make a 12×5 matrix of ones
`A = []`	make A an empty matrix
`eye(5)`	Make a 5×5 identity matrix
`eye(12, 5)`	Make a 12×5 identity matrix
`linspace(1.4, 6.3, 1004)`	Make a row vector with 1004 elements evenly spaced between 1.4 and 6.3
`logspace(1.4, 6.3, 1004)`	Make a row vector with 1004 elements that are logarithmically spaced between $10^{1.4}$ and $10^{6.3}$
`7:15`	Row vector of $7, 8, \ldots, 14, 15$
`7:2:15`	Row vector of $7, 9, 11, \ldots, 15$
`4:-2:-2`	Row vector of $4, 2, \ldots, -2$

Operations on Matrices and Vectors

`3 * x`	Multiply every element of x by 3
`x + 2`	Add 2 to every element of x
`x + y`	Element-wise addition of two vectors x and y
`A * y`	Product of a matrix and vector
`A * B`	Product of two matrices
`A .* B`	Element-wise product of two matrices
`A ^ 3`	Square matrix A to the third power
`A .^ 3`	Every element of A to the third power
`cos(A)`	Compute the cosine of every element of A
`abs(A)`	Compute the absolute values of every element of A
`A'`	Transpose of A
`inv(A)`	Compute the inverse of A
`det(A)`	Compute the determinant of A
`[eVecs, eVals]=eig(A)`	Compute the eigenvectors (`eVecs`) and eigenvalues (`eVals`) of A
`[rows, cols] = size(A)`	Get the size of A
`numel(A)`	Get the number of elements of A

Constants

`pi`	$\pi = 3.141592653589793$
`i`	Imaginary unit $\sqrt{-1}$
`j`	Imaginary unit $\sqrt{-1}$
`NaN`	Not a number (i.e. $0/0$)
`Inf`	Infinity
`realmax`	Largest positive floating-point number $1.7977 \cdot 10^{308}$
`realmin`	Smallest positive floating-point number $2.2251 \cdot 10^{-308}$

Random Numbers

`rand(m,n)`	returns an m by n matrix of random numbers sampled from a standard uniform distribution $[0, 1]$
`randn(m,n)`	returns an m by n matrix of random numbers sampled from a standard Normal distribution with $\mu = 0$ and $\sigma^2 = 1$

Slicing portions of Matrices or Vectors

`x(2:12)`	The 2nd to the 12th elements of x
`x(2:end)`	The 2nd to the last elements of x
`x(1:3:end)`	Every third element of x from the first to last
`A(5,:)`	Get the 5th row of A
`A(:,5)`	Get the 5th column of A
`A(5, 1:3)`	Get the first to third elements in the 5th row

Array concatenation

`A = [x;y]`	Concatenate arrays x and y vertically. x and y must have the same number of columns
`A = [x y]`	Concatenate arrays x and y horizontally. x and y must have the same number of rows

Summary statistics and others

`mean(x)`	returns the mean value of the entries in vector x
`mean(A,1)`	returns a row vector containing the mean of each column of A
`mean(A,2)`	returns a column vector containing the mean of each row of A
`var(x)`	returns the variance of the entries in vector x
`var(A,[],1)`	returns a row vector containing the variance of each column of A
`var(A,[],2)`	returns a column vector containing the variance of each row of A
`min(x)`	returns the minimum value of the entries in vector x
`[minVal,idx] =min(A,[],1)`	`minVal` returns a row vector with the minimum value of each column of A. `idx` returns a row vector with the indices at which the minimum of each column occurs. For `max` the syntax is the same.
`[minVal,idx] =min(A,[],2)`	`minVal` returns a column vector with the minimum value of each crow of A. `idx` returns a column vector with the indices at which the minimum of each row occurs. For `max` the syntax is the same.
`diff(x)`	if x is a vector with N elements, this command returns a vector with $N-1$ elements as follows: $x_2 - x_1, x_3 - x_2, \ldots, x_N - x_{N-1}$

Plotting

`figure`	Create new empty figure window
`subplot(m,n,p)`	divides the current figure into an m-by-n grid and creates axes in the position specified by p (MATLAB counts row-wise for subplot)
`plot(x,y)`	Plot y versus x (must be the same length)
`scatter(x,y)`	scatter plot of y versus x.
`loglog(x,y)`	Plot y versus x on a log-log scale (both axes have a logarithmic scale)
`semilogx(x, y)`	Plot y versus x with x on a log scale
`semilogy(x, y)`	Plot y versus x with y on a log scale
`bar(x, y)`	Bar chart. Draw bars of height y at locations specified by x
`xlim([xmin xmax])`	Force the x axis to be scaled between $x_{\min}$ and $x_{\max}$
`ylim([ymin ymax])`	Force the y axis to be scaled between $y_{\min}$ and $y_{\max}$
`axis equal`	Force the x and y axes to be scaled equally
`title('A Title')`	Add a title to the plot
`xlabel('x label')`	Add a label to the x axis
`ylabel('y label')`	Add a label to the y axis
`legend('foo', 'bar')`	Label 2 curves for the plot
`grid`	Add a grid to the plot
`hold on`	Multiple plots on single figure
`hold off`	Release hold on current figure

Loading Files

`load('data.mat')`	Load variables from file into workspace

Logicals

```
a = 10;  % Assign a the value of 10
a == 5   % Test if a is equal to 5
    false
a == 10  % Test if a is equal to 10
    true
a >= 5   % Test if a is greater than or equal to 5
    true
a < 11   % Test if a is less than 11
    true
a ~= 4   % Test if a is not equal to 4
    true
a > 1 && a ~= 10 % Test if a is greater than 1 AND
    false        % not equal to 10
a > 1 || a ~= 10 % Test if a is greater than 1 OR
    true         % not equal to 10
```

Conditional Statements

```
if a > 10
    disp('Greater than 10');
elseif a == 5
    disp('a is 5');
else
    disp('Neither condition met');
end
```

For loops

```
for k = 1:5
    disp(k);
end
```

While loops

```
k = 0;
while k < 7
    k = k + 1;
end
```

Functions

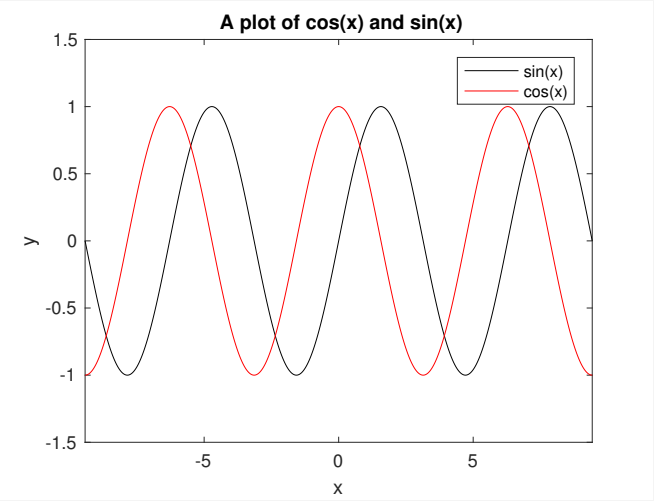
```
function output = addNumbers(x, y)
    output = x + y;
end

addNumbers(10, -5)
    5
```

Function Handles

```
f = @(x) sin(x.^2)./(5*x);

f(pi/2)
    0.0795
f([-pi/2, 0, pi/2])
    -0.0795 NaN 0.0795
```

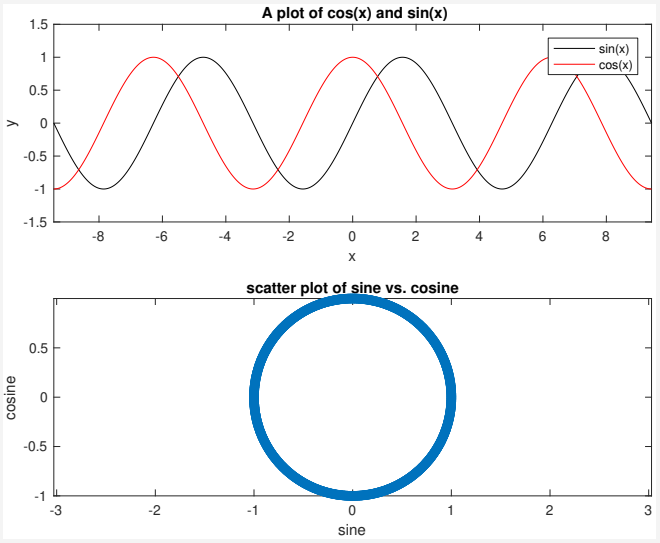
Plotting

```
x = linspace(-3*pi, 3*pi, 1000);
y1 = sin(x);
y2 = cos(x);

figure;
plot(x, y1, 'k-');  % Plot sin(x) as a black line
hold on             % Now we can add another curve
plot(x, y2, 'r-');  % Plot cos(x) as a red line
hold off

% Set the axis limits
xlim([-3*pi, 3*pi]);
ylim([-1.5, 1.5]);

% Add axis labels
xlabel('x');
ylabel('y');

% Add a title
title('A plot of cos(x) and sin(x)');

% Add a legend
legend('sin(x)', 'cos(x)');
```



Plotting with subplots

```
x = linspace(-3*pi, 3*pi, 1000);
y1 = sin(x);
y2 = cos(x);

figure;

% First panel
subplot(2,1,1)
plot(x, y1, 'k-');  % Plot sin(x) as a black line
hold on             % Now we can add another curve
plot(x, y2, 'r-');  % Plot cos(x) as a red line
hold off
% Set the axis limits
xlim([-3*pi, 3*pi]);
ylim([-1.5, 1.5]);

% Add axis labels
xlabel('x');
ylabel('y');

% Add a title
title('A plot of cos(x) and sin(x)');

% Add a legend
legend('sin(x)', 'cos(x)');

subplot(2,1,2)
scatter(y1,y2);
axis equal;
title('scatter plot of sine vs. cosine');
xlabel(' sine ');
ylabel(' cosine ');
```



9.40 Introduction to Neural Computation
Spring 2018