**MATLAB Handbook**

**A. General MATLAB commands**

1. The MATLAB environment

|  |  |
| --- | --- |
| Window | Purpose |
| **Command window** | Main window; enters variables; runs program |
| **Figure window** | Contains outputs from graphics commands |
| **Editor window** | Creates and debugs script and function files |
| **Help window** | Provides help documentation |
| **Command history window** | Logs commands entered in the Command window |
| **Workspace window** | Provides information about the variables that are in use |
| **Current Folder window** | Shows the files in the current directory |

1. General operators and useful functions

|  |  |
| --- | --- |
| Function | Description |
| **clear** | Clears variable |
| **close** | Closes figure (close all closes all open figures) |
| **clc** | Clears screen (clear all clears all variables in workspace) |
| **doc** | Provides documentation in separate window |
| **help** | Provides command-line help information |
| **format** | Sets command window output display |
| **%** | Comment |
| **;** | Suppresses display |
| **…** | Ellipses; line-continuation operator |
| **@** | Function handle |

1. Character types and relevant functions

|  |  |
| --- | --- |
| Function | Description |
| **char** | Character array (defined by apostrophes, ‘’) |
| **string** | String array (defined by quotations, “ “) |
| **double** | Double-precision floating point (Regular #, 64 bits, 2^64) |
| **single** | Single-precision floating point (Regular #, 32 bits, 2^32) |
| **complex** | Complex data |
| **logical** | Boolean true or false |
| **int8, int16, int32, int64** | Signed integer |
| **uint8, uint16, uint32, uint64** | Unsigned integer |
| **cell** | Holds other datatypes of various sizes |
| **struct** | Structure |
| **num2str** | Convert a numeric into string character |

1. Operators and specific characters

|  |  |
| --- | --- |
| Operator | Description |
| **+** | Plus; addition operator |
| **-** | Minus; subtraction operator |
| **\*** | Scalar and matrix multiplication operator |
| **.\*** | Array (point-wise, elemental) multiplication operator |
| **^** | Scalar and matrix exponentiation operator |
| **.^** | Array (point-wise, elemental) exponentiation operator |
| **/** | Right-division operator |
| **./** | Array (point-wise, elemental) right-division operator |
| **:** | Generates regularly spaced elements and denotes entire row or columns |
| **( )** | Parentheses; encloses function arguments and overrides precedence |
| **[ ]** | Brackets; enclosures array elements |
| **.** | Decimal point |
| **,** | Comma; separates statements and elements in a row |
| **;** | Semicolon; separates columns of a matrix and suppresses display |
| **%** | Percent sign; designates a comment |
| **=** | Assignment operator |
| **‘** | Apostrophe; transposes a matrix |
| **pi** | Pre-defined value: 3.14159… |
| **i** | Pre-defined value: sqrt(-1) |
| **inf** | Pre-defined value: Infinity |
| **NaN** | Pre-defined value: Not a number |

1. Common mathematical functions

|  |  |
| --- | --- |
| Function | Description |
| **sqrt(M)** | Square root |
| **sum(M)** | Adding numbers in a matrix |
| **exp(M)** | Exponent (e^M) |
| **abs(M)** | Absolute value |
| **log(M)** | Log base e (natural log) |
| **log10(M)** | Log base 10 |
| **factorial(M)** | Factorial function x! |
| **sin(M) (and other trig)** | Trigonometric operations |
| **round(M)** | Rounds to nearest integer |
| **floor(M) (alt: ceil(M))** | Floor or ceiling rounding |

1. Creating and working with defined matrices

|  |  |
| --- | --- |
| Function | Description |
| **zeros(r,c)** | Generates matrix of all zeros with r rows and c columns |
| **ones(r,c)** | Generates matrix of all ones with r rows and c columns |
| **rand(r,c)** | Generates matrix of random #’s uniformly distributed with r rows & c columns |
| **randi([x y],r,c)** | Generates matrix of random integers uniformly distributed |
| **cat(dim,A,B)** | Concatenates matrices A and B along the specified dimension |
| **horzcat** | Horizontally concatenates N matrices |
| **vertcat** | Vertically concatenates N matrices |
| **size(M)** | R = row #; C = column # |
| **length(M)** | Length |
| **linspace(A,B,n)** | Defines a linearly spaced row vector from A to B with n (default 100) elements |
| **numel(M)** | # of elements |
| **reshape(M,r,c)** | Change the shape of a matrix |
| **flipud(M)** | To flip up/down (use fliplr for left/right) |
| **repmat(M,r,c)** | To replicate a matrix |
| **sort(M)** | Sort the elements of a matrix |
| **sum(M)** | Sums each column |

1. Logical and relational operators and functions

|  |  |
| --- | --- |
| Function | Description |
| **any** | Relational operator: equal to |
| **all** | Relational operator: not equal to |
| **isempty** | Relational operator: less than |
| **isinf** | Relational operator: less than or equal to |
| **isnan** | Relational operator: greater than |
| **find** | Relational operator: greater than or equal to |
| **==** | Relational operator: equal to |
| **~=** | Relational operator: not equal to |
| **<** | Relational operator: less than |
| **<=** | Relational operator: less than or equal to |
| **>** | Relational operator: greater than |
| **>=** | Relational operator: greater than or equal to |
| **& (alt: &&)** | Logical operator: AND |
| **| (alt: ||)** | Logical operator: OR |
| **~** | Logical operator: NOT |
| **xor** | Logical operator: EXCLUSIVE OR |

1. Selection statements

|  |  |
| --- | --- |
| Function | Description |
| **for** | Executes a group of statements in a loop for a specified number of times |
| **if, elseif, else** | Executes a group of statements when the expression is true. |
| **continue** | Continues to next loop |
| **break** | Breaks current loop |

1. Working with strings

|  |  |
| --- | --- |
| Function | Description |
| **strcmp** | Exact string comparison |
| **contains** | Compares patterns within strings |
| **strjoin** | Combine strings |
| **string** | Convert character to string |
| **strip** | Remove white space |
| **strrep** | Replace string pattern with alternate |
| **num2str** | Convert number to string |

**B. Plotting**

1. General

|  |  |
| --- | --- |
| Function | Description |
| **figure()** | Opens new figure handle |
| **close()** | Closes figure handle inside () |
| **close all** | Closes all figures |
| **hold on** | Holds the data on a figure |
| **plot(x,y,LineSpec)** | Creates a 2-D line plot of the data in Y versus the corresponding values in X. |
| **subplot(r,c,n)** | Generates sub plots with r rows, c columns; plots at the n’th location |
| **scatter(x,y)** | Creates a scatter plot with circles specified x and y coordinates |
| **bar(x,y)** | Draws the bars y at the locations specified by x. |
| **histogram(x,nbins)** | Creates a histogram of the data in x with nbins # of specified bins |
| **semilogx(x,y)** | Plots on a log-transformed x-axis |
| **semiology(x,y)** | Plots on a log-transformed y-axis |
| **loglog(x,y)** | Plots on a log-transformed x- and y-axis |

1. Line specs

|  |  |
| --- | --- |
| **‘color’** | **Color** |
| **y** | yellow |
| **m** | magenta |
| **c** | cyan |
| **r** | red |
| **g** | green |
| **b** | blue |
| **w** | white |
| **k** | black |
| **[r g b]** | Any triplicate where 0 <r, g, b < 1 |
| **‘lineStyle’** | **Style** |
| **-** | Solid |
| **--** | Dashed |
| **:** | Dotted |
| **-.** | Dash-dot |
| **‘marker’** | Shape |
| **o** | Circle |
| **+** | Plus sign |
| **\*** | Asterisk |
| **.** | Point |
| **x** | Cross |
| **s** | Square |
| **d** | Diamond |
| **^** | Carrot |
| **v** | V-shape |
| **>** | Right triangle |
| **<** | Left triangle |
| **p** | Pentagram |
| **h** | Hexagram |

Example:

x = linspace(1,3);

y = linspace(0,40);

figure; hold on

plot(x,y,’r’,’linewidth’,4.0)

set(gca,’fontsize’,40,’linewidth’,3.0,’xtick’,[1 2 3],’ytick’,[0:10:40])

xlim([1 3]), ylim([0 40]), title(‘Ex’)

1. Axis options

|  |  |
| --- | --- |
| Function | Description |
| **axis tight** | Fill axis box tightly around data |
| **axis equal** | Use the same length for data units along each axis |
| **axis image** | Use the same length for the data units along each axis and fit the axes box tightly around the data. |
| **axis square** | Use axis lines with equal lengths. Adjust the increments between data units accordingly. |
| **axis normal** | Restore default behavior |
| **axis fill** | Enable the “stretch-to-fill” behavior (the default). The lengths of each axis line fill the position rectangle defined in the [Position](https://localhost:31515/static/help/matlab/ref/matlab.graphics.axis.axes-properties.html) property of the axes. |
| **axis tight** | Turn box on or off |
| **box on (or box off)** | Fill axis box tightly around data |

**C. Data input/output**

|  |  |
| --- | --- |
| Function | Description |
| **input** | Load in files to work with |
| **disp** | Display text on command line |
| **num2str** | Converts numbers to string array |
| **load** | Load in files to work with |
| **save** | Save data in stored variables |
| **who** | List variables in your workspace |
| **table(var1,var2,…)** | Writing variables to table structure |
| **readtable(‘filename’)** | Read in a table (.txt, .csv, or .xlsx files) |
| **Writetable(T,‘filename’)** | Write table to filename (.txt, .csv, or .xlsx files) |
| **table2array(T)** | Converts table elements to matrix elements |
| **[num txt raw] = xlsread(‘filename’)** | Read in excel file |
| **addpath(‘foldername’)** | Adds all folders and subfolders to path |

**D. Statistical functions**

1. Descriptive statistics

|  |  |
| --- | --- |
| Function | Description |
| **mean(A,dim)** | M = mean([A](https://localhost:31515/static/help/matlab/ref/mean.html?overload=(matlab)%2Fmean%20false&snc=FVAKZX&container=jshelpbrowser)) returns the [mean](https://localhost:31515/static/help/matlab/ref/mean.html) of the elements of A along the first array dimension whose size does not equal 1. |
| **median(A,dim)** | M = median([A](https://localhost:31515/static/help/matlab/ref/median.html?searchHighlight=median&searchResultIndex=1)) returns the median value of A. |
| **mode(A,dim)** | M = mode([A](https://localhost:31515/static/help/matlab/ref/mode.html?searchHighlight=mode&searchResultIndex=1)) returns the sample mode of A, |
| **std(A,[],dim)** | S = std([A](https://localhost:31515/static/help/matlab/ref/std.html)) returns the [standard deviation](https://localhost:31515/static/help/matlab/ref/std.html) of the elements of A along the first array dimension whose size does not equal 1. |
| **var(A,dim)** | V = var([A](https://localhost:31515/static/help/matlab/ref/var.html)) returns the [variance](https://localhost:31515/static/help/matlab/ref/var.html) of the elements of A along the first array dimension whose size does not equal 1. |
| **min(A,dim)** | [M](https://localhost:31515/static/help/matlab/ref/min.html?searchHighlight=min&searchResultIndex=1) = min([A](https://localhost:31515/static/help/matlab/ref/min.html?searchHighlight=min&searchResultIndex=1)) returns the minimum elements of an array. |
| **max(A,dim)** | [M](https://localhost:31515/static/help/matlab/ref/max.html) = max([A](https://localhost:31515/static/help/matlab/ref/max.html)) returns the maximum elements of an array. |
| **iqr(A)** | Returns the inter-quartile range |

1. Inferential statistics

|  |  |
| --- | --- |
| Function | Description |
| **p=normcdf** | Normal cumulative distribution function |
| **x=norminv** | Normal inverse cumulative distribution function |
| **tcdf** | t cumulative distribution function |
| **tinv** | t inverse cumulative distribution function |
| **ztest** | One-sample z-test |
| **ttest** | One-sample and paired-sample t-test |
| **ttest2** | Two independent sample t-test |
| **anova1** | One-way ANOVA |
| **fcdf** | F cumulative distribution function |
| **chi2cdf** | 2 cumulative distribution function |

**E. Numerical methods for curves**

1. Curve fitting, root finding, and data interpolation

|  |  |
| --- | --- |
| Function | Description |
| **polyval(x,y)** | Linear regression |
| **polyfit(X,Y,1)** | Linear regression |
| **corrcoef(X,Y)** | Correlation coefficients |
| **roots(p)** | Root finding |
| **poly(r)** | Finding polynomial coefficients |
| **interp1([x](https://localhost:31515/static/help/matlab/ref/interp1.html?snc=ZXCNG5&container=jshelpbrowser),**[**v**](https://localhost:31515/static/help/matlab/ref/interp1.html?snc=ZXCNG5&container=jshelpbrowser)**,**[**xq**](https://localhost:31515/static/help/matlab/ref/interp1.html?snc=ZXCNG5&container=jshelpbrowser)**)** | Interpolation/extrapolation |
| **lsqcurvefit(@fun,x0,xdata,ydata)** | Curve fitting for non-linear data |

1. Integration and differentiation

|  |  |
| --- | --- |
| Function | Description |
| **diff(x)** | Returns a vector d containing the differences between adjacent elements in the vector x. |
| **int([expr](https://localhost:31515/static/help/symbolic/int.html),**[**var**](https://localhost:31515/static/help/symbolic/int.html)**)** | computes the indefinite integral of expr with respect to the symbolic scalar variable var. Specifying the variable var is optional. |
| **quad(fun,a,b)** | Numerical method to find the area under the curve |
| **polyint(p,k)** | Integrates polynomial p with an integration constant k |

1. Ordinary differential equation solvers

|  |  |
| --- | --- |
| Function | Description |
| **ode23** | Nonstiff, low order solver |
| **ode45** | Nonstiff, medium-order solder |
| **ode113** | Nonstiff, variable-order solver |
| **ode23s** | Stiff, low-order solver |
| **ode15s** | Stiff, variable-order solver |
| **global** | Declares global variables |

**F. Image processing**

1. Image input/output and image conversions

|  |  |
| --- | --- |
| Function | Description |
| **gray2ind** | Intensity image to index image |
| **im2bw** | Image to binary |
| **im2double** | Image to double precision |
| **im2uint8** | Image to 8-bit unsigned integers |
| **im2uint16** | Image to 16-bit unsigned integers |
| **ind2gray** | Indexed image to intensity image |
| **mat2gray** | Matrix to intensity image |
| **rgb2gray** | RGB image to grayscale |
| **rgb2ind** | RGB image to indexed image |
| **imread** | Read in image |
| **imshow** | Display image in figure |
| **imwrite** | Saving image files |
| **squeeze** | Reduces dimensionality of matrix |

1. Image modification functions

|  |  |
| --- | --- |
| Function | Description |
| **imadd(X,Y)** | Add two images |
| **imsubtract** | Subtract two images |
| **immultiply** | Multiple two images |
| **imrotate** | Rotate an image |
| **imcomplement** | Take the complement of an image |
| **imresize** | Resizing an image |
| **imwrite** | Saving an image |

1. Image filtering, enhancement, and segmentation

|  |  |
| --- | --- |
| Function | Description |
| **bwboundary** | Find boundary |
| **imfill** | Fill mask gaps |
| **graythresh** | Automatically find threshold |
| **imadjust** | Increase brightness |
| **adapthisteq** | Increase brightness |
| **imclose** | Close mask gaps |
| **imopen** | Open mask gaps |
| **strel** | Determining opening and closing pixel size |
| **imdilate** | Expand local mask areas |
| **imerode** | Reduce local mask areas |

1. ROI functions

|  |  |
| --- | --- |
| Function | Description |
| **imshowpair(im,mask)** | Image visualization of im with superimposed mask |
| **false** | n-by-n array of logical falses (0’s) |
| **uint8** | Convert to class 8-bit unsigned integer arrays |
| **im2double** | Convert to class double |
| **drawrectangle** | Create customizable rectangular ROI |
| **regionprops** | Measure properties of image regions |
| **impoly** | Create draggable, resizable polygon |
| **imhist** | Histogram of image data |
| **adapthisteq** | Contrast-limited adaptive histogram equalization (local) |
| **histeq** | Enhance contrast using histogram equalization (global) |

1. Video analysis

|  |  |
| --- | --- |
| Function | Description |
| **implay** | Play .avi files directly from MATLAB |
| **dvread (filename)** | Use this to read in .dv files (dv are delta vision proprietary format) |
| **{ }** | Cell array |
| **ceil** | Rounds to the greater whole integer |
| **floor** | Rounds to the lowest whole integer |

**G. Bioinformatics**

|  |  |
| --- | --- |
| Function | Description |
| **fastaread** | Read in .fasta file |
| **fastawrite** | Write .fasta file |
| **basecount** | Count bases in string |
| **nt2aa** | Convert nucleotide to amino acid sequence |
| **seqrcomplement** | Reverse complement sequence |
| **seqalignviewer** | Visualize and edit multiple aligned sequences |