MATLAB NOTES
(MA102)

Lia Vas

Content

1. ALGEBRA

1.1 Basic Arithmetic

MATLAB can perform basic arithmetic operations. You can use +, -, *, \ and ^ to add, subtract, multiply, divide or exponentiate, respectively. For example, if you enter:

>> 2^3 - 2*2

MATLAB gives you the answer:
ans =
4

If you want to perform further calculations with the answer, you can type **ans** rather than retyping the whole answer. For example,

>> sqrt(ans)

gives you
ans =

1.2 Solving Equations

For solving equations, you can use the command **solve**. The command solve is always followed by parenthesis. After that, you type the equation you would like to solve in single quotes. Then type coma, and variable for which you are solving in (single) quotes. Thus, the command solve has the following form

solve('*equation***', '***variable for which you are solving***')**

For example, to solve the equation x^3-2x-4=0, you type:

>> solve('x^3-2*x-4=0', 'x')

and get the following answer:
ans =
[2]
[-1+i]
[-1-i]

Here *i* stands for the imaginary number $\sqrt{-1}$. This answer tells us that there is just one real solution, 2.

Note that we entered **2*x** to represent *2x* in the command above. **Entering * for multiplication is always necessary in MATLAB.**

MATLAB can give you both symbolic and numerical answer. For example, let us solve the equation 3x^2-8x+2=0.

>> solve('3*x^2-8*x+2=0','x')

ans =
[4/3+1/3*10^(1/2)]
[4/3-1/3*10^(1/2)]

If we want to get the answer in the decimal form with, say, three significant digits, we can use the command **vpa**.

>> vpa(ans, 3)

ans =
[2.38]
[0.28]

By changing 3 in the command **vpa(ans, 3)** you can specify the number of digits in the answer. The command **vpa** has the general form

vpa(*expression you want to approximate***,** *number of significant digits***)**

2. FUNCTIONS

2.1 Representing Functions

To represent a function, use the command **inline**. Similarly to solve, this command is followed by parenthesis and has the following form:

inline('*function***', '***independent variable of the function***')**

For example, let us define the function x^2+3x-2. Enter

>> f = inline('x^2+3*x-2', 'x')

MATLAB output will be
f =
Inline function:
f(x) = x^2+3*x-2

After defining a function, we can evaluate it at a point. For example,

>> f(2)

ans =
8

The following table gives an overview of how most commonly used functions or expressions are represented in MATLAB.

function or symbol	representation in MATLAB
e^x	exp(x)
ln x	log(x)
log x	log(x)/log(10)
log. base a of x	log(x)/log(a)
sin x	sin(x)
cos x	cos(x)
arctan(x)	atan(x)
π	pi

If a function is short, it might be **faster to evaluate a function at a point simply by typing the value of *x* directly for *x***. For example, to evaluate sin(*x*) at *x*=2, simply type

>> sin(2)
and obtain the answer
ans = .909297

As when using the calculator, one must be careful when representing a function. For example

$\frac{1}{x(x+6)}$ should be represented as **1/(x*(x+6))** not as **1/x*(x+6)** nor as **1/x(x+6),**

$\frac{3}{x^2+5x+6}$ should be represented as **3/(x^2+5*x+6)** not as **3/x^2+5*x+6,**

e^{5x^2} should be represented as **exp(5*x^2)** not as **e^(5*x^2), exp*(5*x^2), exp(5x^2)** nor as **exp^(5*x^2).**

Ln(x) should be represented as **log(x)**, not **ln(x).**

$\log_3(x^2)$ should be represented as **log(x^2)/log(3)** not as **log(x)/log(3)*x^2.**

2.2 Representing independent variable

Most of the mathematics operations on functions (e.g. graphing, differentiation, integration) are performed easier in MATLAB if you declare independent variable for a MATLAB variable. Use the command **syms to** declare the variables you plan to use. For example, if you want to use variable x, simply type

>> syms x

before any of the commands for graphing, differentiation or integration.

2.3 Graphing Functions

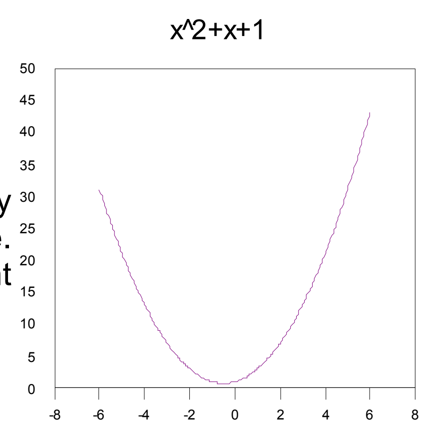
Let us start by declaring that *x* is a variable:
>> syms x

The simplest command in MATLAB for graphing is **ezplot**. The command has the following form

ezplot(*function*)

For example, to graph the function x^2+x+1, you simply type

>> ezplot(x^2+x+1)



A new window will open and graph will be displayed. To copy the figure to a text file, go to **Edit** and choose **Copy Figure**. Then place cursor to the place in the word file where you want the figure to be pasted and choose **Edit** and **Paste**.

We can specify the different scale on *x* and *y* axis.
To do this, the command **axis** is used.

It has the following form

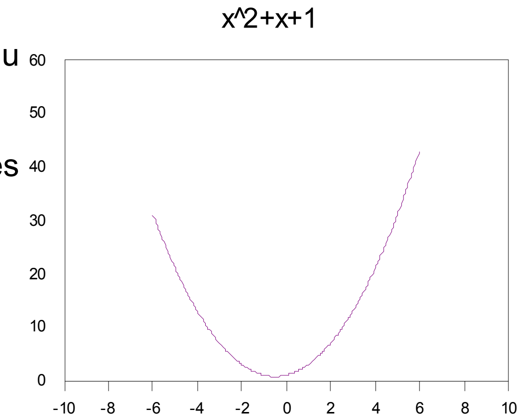
axis([x_{min}, x_{max}, y_{min}, y_{max}])

This command parallels the commands in menu WINDOW on the TI83 calculators.

For example, to see the above graph between x-values -10 and 10 and y-values 0 and 60, you can enter

>> axis([-10 10 0 60])

The result will be:



Note that the domain of function did not change by command axis. To see the graph on the entire domain (in this case [-10, 10]), add that domain after the function in the command ezplot:

ezplot(*function*, [x_{min}, x_{max}])

In this case,

>> ezplot(x^2+x+1, [-10, 10])

will give you the desired graph.

3. CALCULUS

3.2 Differentiation

Start by declaring x for a variable. The command for differentiation is **diff**. It has the following form

diff(*function*)

For example,
>> syms x
>> diff(x^3-2*x+5)

gives us the answer
ans = 3*x^2-2

To get n-th derivative use

diff(*function, n***)**

For example, to get the second derivative of x^3-2x+5, use:

>> diff(x^3-2*x+5, 2)

ans = 6*x

Similarly, the 23rd derivative of sin(x) is obtained as follows.

>> diff(sin(x), 23)

ans =-cos(x)

To evaluate derivative at a point, we need to represent the derivative as a new function. For example, to find the slope of a tangent line to x^2+3x-2 at point 2, we need to find the derivative and to evaluate it at x=2.

>> diff(x^2+3*x-2) (first we find the derivative)

ans = 2*x+3

>> f = inline('2*x+3', 'x') (then we representative the derivative as a function)

```
f =
    Inline function:
    f(x) = 2*x+3
```

>> f(2) (and, finally, we evaluate the derivative at 2)

ans =   7

Recall the steps needed in order to find minimum or maximum values of a given function (using second derivative test)

1. Find first derivative
2. Solve it for zeros. The x-values you obtain are called critical
3. Find second derivative
4. Plug critical points in second derivative. If your answer is negative, the function has a maximum value at a critical point used. If your answer is positive, the function has a minimum value at a critical point used.
5. Plug critical points in your function. The y-values you obtain are your maximum or minimum values.

In MATLAB, start with syms x.

1. Finding derivative: **diff(***function***)**
2. Finding critical points: **solve('***copy-paste the answer from step 1***=0', 'x')**
3. Finding second derivative: **diff(***function, 2***)**
4. Evaluating second derivative at critical points: **g=inline(***second derivative***, 'x')** followed by **g(***critical value***)**
5. Evaluating function at critical points: **f=inline(***function***, 'x')** followed by **f(***critical value***)**

For example, let us find maximum or minimum value of x^3-2x+5. Start by finding first derivative:

>> diff(x^3-2*x+5)
ans = 3*x^2-2

Then find critical point(s):
>> solve('3*x^2-2=0', 'x')
ans =
[6^(1/2)/3]
[-6^(1/2)/3]
or, using **vpa(ans, 3)**
[.816]
[-.816]

Find second derivative
>> diff(x^3-2*x+5, 2)
ans = 6*x
Evaluate this at critical points.
>> g=inline('6*x', 'x')
g(x)= 6*x
>>g(.816)
ans = 4.896 Positive answer means that the function has minimum at x=.816
>> g(-.816)
ans = -4.896 Negative answer means that the function has maximum at x=.816

Finding y-values of maximum and minimum:
>> f=inline('x^3-2*x+5', 'x')
f(x)= x^3-2*x+5
>>f(.816)
ans = 3.911 This is the local minimum value.
>>f(.816)
ans = 6.088 This is the local maximum value.

3.2 Integration

We can use MATLAB for computing both definite and indefinite integrals using the command **int**. For the indefinite integrals, start with **syms x** followed by the command

int(*function***)**

For example, the command
>> int(x^2)
evaluates the integral $\int x^2 dx$ and gives us the answer
ans = 1/3*x^3

For definitive integrals, the command is

int(*function, lower bound, upper bound***)**

For example,
>> int(x^2, 0, 1)
evaluates the integral $\int_0^1 x^2 dx$ The answer is
ans = 1/3

4. PRACTICE PROBLEMS

1. Evaluate the following expressions using MATLAB.
   a) sin(π/6) b) $\frac{\sqrt{(5)}+3}{\sqrt{(3)}-1}$ c) $\log_2(5)$
2. Solve the following equations for x and express your answers as decimal numbers.
   a) x^3-2x+5=0 b) $\log_2(x^2-9)=4$

3. Let f(x)= $\frac{x^3+x+1}{x}$
a) Evaluate f(x) for x=3 and x=-2.
b) Find x-values that corresponds to y-value of 2.
c) Graph f(x) on domain [-4 4].

4. Let f(x)= e^{3x^2+1}.
   a) Find first and second derivative of f(x).
   b) Find the slope of the tangent line to f(x) at x=1.
   c) Find the critical points of f(x).

5. Find the extreme values of
   a) x^3-4x+8 b) xe^{-3x}

6. Evaluate the integral $\int xe^{-3x}dx$.
7. Evaluate the integral $\int_0^1 xe^{-3x}dx$.

Solutions.

1. a) sin(pi/6). ans=.5 b) (sqrt(5)+3)/(sqrt(3)-1). ans=7.152 c) log(5)/log(2). ans=2.3219
2. a) solve('x^3-2*x+5=0', 'x'), only real answer is -2.09. b) solve('log(x^2-9)/log(2)=4','x'). ans= 5, -5.
3. a) f=inline('(x^3+x+1)/x', 'x'), f(3)=10.333, f(-2)=4.5 b) solve('(x^3+x+1)/x=2','x') x=-1.3247 is the only real solution. c) ezplot((x^3+x+1)/x, [-4,4])
4. a) diff(exp(3*x^2+1)), ans=6*x*exp(3*x^2+1); diff(exp(3*x^2+1), 2), ans=6*(1+6*x^2)* exp(3*x^2+1). b) g=inline('6*x*exp(3*x^2+1)','x') g(1)=6*exp(4)=327.59. c) solve('6*x*exp(3*x^2+1)=0','x'), ans=0.
5. a) max (-1.15, 11.079), min (1.15, 4.92). b) max (.333, .1226), no min.
6. int(x*exp(-3*x)), ans=-1/3*x*exp(-3*x)-1/9*exp(-3*x)
7. int(x*exp(-3*x), 0,1), ans=-4/9*exp(-3)+1/9=.08898

5. APPENDIX

5.1 Solving systems of equations

You can solve more than one equation simultaneously. For example suppose that we need to solve the system $x^2 + x + y^2 = 2$ and $2x - y = 2$. We use:

>> [x,y] = solve('x^2+ x+ y^2 = 2', '2*x-y = 2')

And get the solutions
x =
1
2/5
y =
0
-6/5

Note that command **solve** has to be preceded with **[*list of all variables*].**

You can solve an equation in two variables for one of them. For example:

>> solve('y^2-5*x*y-y+6*x^2+x=2', 'y')
ans =
3*x+2
2*x-1

5.2 Graphing multiple functions on the same plot

To graph multiple curves on the same window, you can use the **ezplot** command in combination with hold on and hold off on the following way:

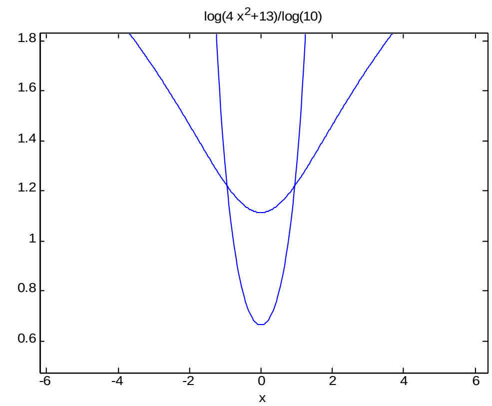
ezplot(*1st function***)**

hold on
ezplot(*2nd function***)**
ezplot(*3rd function)*
...
ezplot(*n-th function***)**
hold off

For example, to graph $e^{x^2/2} - 1/3$ and $\log_{10}(4x^2 + 13)$ on the same plot, use

```
>> ezplot(exp(x^2/2)-1/3)
>> hold on
>> ezplot(log(4*x^2+13)/log(10))
>> hold off
```

The result is displayed on the following graph.



5.3 Limits

You can use **limit** to compute limits, left and right limits as well as infinite limits. For example, to evaluate the limit when $x \to 2$ of the function $\frac{x^2-4}{x-2}$, we have:

```
>>  syms x
>> limit((x^2-4)/(x-2), x, 2)
ans =
 4
```

For left limits, add **'left'**.

```
>> limit(abs(x)/x, x, 0, 'left')
ans =
 -1
```

Similarly for right limits:

```
>> limit(abs(x)/x, x, 0, 'right')
ans =
 1
```

Inf denotes the infinity symbol ∞ in MATLAB. Thus, limits at infinity can be evaluated as follows:

```
>> limit(exp(-x^2-5)+3, x, Inf)
ans =
 3
```