Notes

Notes

lr	ntroduction to Ma	tlab	
	Eugeniy E. Mikhailo	I	_
	The College of William & Ma	ry	
	Lecture 02		
		 + (1) + (2) + (2) 	E •0 ⊂ G
Eugeniy Mikhailov (W&M) Matlab variable ty	Practical Computing	Lecture 02	
Eugeniy Mikhailov (W&M)	Practical Computing	 · · · · · · · · · · · · · · · · · · ·	≣ •ી લ.ઉ 2/27
Matlab variable ty	pes		
 integer 			

Notes

Matlab variable types

• 123, -345, 0

integer

• 123, -345, 0

Eugeniy Mikhailov (W&M)

- real or float
 - 12.2344

 - 5.445454engineering notation
 - 4.2323e-9 = 4.2323 × 10⁻⁹

Notes

10

Lecture 02

2/27

≥ •0 < @

Practical Computing

Matlab variable types

integer 123, -345, 0

- real or float
 - real of noal
 - 12.2344
 - 5.445454
 - engineering notation
 4.2323e-9 = 4.2323 × 10⁻⁹
- imaginary $1i = \sqrt{-1}$
 - 34.23+21.21i
 - (1+1i) * (1-1i) = 2
 - (1+11) * (1-11) =

. . Practical Computing Eugeniy Mikhailov (W&M) Lecture 02 2/27 Matlab variable types Notes integer • 123, -345, 0 real or float • 12.2344 • 5.445454 engineering notation • 4.2323e-9 = 4.2323 × 10⁻⁹ • imaginary $1i = \sqrt{-1}$ • 34.23+21.21i • (1+1i) * (1-1i) = 2 • strings (put your words inside apostrophes) handy for file names and messages • 'programming is fun' • s='Williamsburg' Eugeniy Mikhailov (W&M) Practical Computing Lecture 02 Some built in constants and functions Notes • $\pi = 3.141592653589793238462643383279502 \cdots$ • use pi • trigonometry functions By default angle is in radians But can be done in degrees • sin, cos, tan, cot • sind, cosd, tand, cotd • asin, acos, atan, acot • asind, acosd, atand, acotd sin(pi/2)=1 sind(90)=1 hyperbolic functions • sinh, cosh, tanh, coth \bullet asinh, acosh, atanh, acoth logarithms natural log • base of 10 log10 ower • x^y use x^y or alternatively power (x, y) • e^y use exp (y) Eugeniy Mikhailov (W&M) Practical Computing Lecture 02

Assignment operator

x = 1.2 + 3.4

Eugeniy Mikhailov (W&M)

 Image: Practical Computing
 Image: Lecture 02

4/27

Notes

Assignment operator

. .

x = 1.2 + 3.4
Despite the look = is not the equality operator.
= is the assignment operator.

>> x = 1.2 + 3.4 x = 4.6000

			-0.0.0
Eugeniy Mikhailov (W&M)	Practical Computing	Lecture 02	4/27
Assignment opera	ator		
x = 1.2 + 3.4 Despite the look = is no = is the assignment op	ot the equality operator. erator.		
>> x = 1.2 + 3.4 x = 4.6000			
The expression above • evaluate expression • assign the result on • now variable x hole We are free to use the expressions	should be read as on at the right hand side f the RHS to the variabl ds the value 4.6 value of the variable x i	of equality symbol e on the left hand sig in any further	n
>> x+4.2 ans =			
0.0000		 · · · · · · · · · · · · · · · · · · ·	৵৽৻৽
Eugeniy Mikhailov (W&M)	Practical Computing	Lecture 02	

Efficient editing - Tab-completition

Notes

Notes

Notes

Once you typed some expressions in "Command window"

- type couple of first symbols of variable or function name
- hit tab and you will get
 - either fully typed name (if it is unique)
 - or little chart with choices
 - use <up> or <down> arrows to choose
 - alternatively <Ctrl-p>, <Ctrl-n>
 - then hit <enter> to make your choise

Help related commands

Notes

These are the most important commands

- \bullet docsearch word
 - will search for word in the help files and show up matched help files
 example: docsearch trigonometry

Practical Computing

• help name

Eugeniy Mikhailov (W&M)

- output short help text into "Command window" about
- function/method named name
- example: help sin
- doc name
 - show a reference page about function/method named name in the help browser
 - \bullet usually has more information in comparison to ${\tt help\ name}$
 - example: doc sin

Lecture 02

Operators Precedence

Look at the following Matlab expression

-2^4*5 + tan(pi/8+pi/8)^2

Guess the answer.

		$+ \Box \mapsto + \Box \to + \Xi \mapsto + \Xi$
Eugeniy Mikhailov (W&M)	Practical Computing	Lectu
Operators Prec	edence	

Look at the following Matlab expression

-2^4*5 + tan(pi/8+pi/8)^2

Guess the answer.

- (2^4)*5 + (tan((pi/8+pi/8)))^2

Operators Precedence

Look at the following Matlab expression

-2^4*5 + tan(pi/8+pi/8)^2

Practical Computing

Guess the answer.

Eugeniy Mikhailov (W&M)

- (2^4)*5 + (tan((pi/8+pi/8)))^2
 - (16)*5 + (tan((pi/4)))^2

Eugenly Mikhailov (W&M) Practical Computing
Operators Precedence

Look at the following Matlab expression

-2^4*5 + tan(pi/8+pi/8)^2

Guess the answer.

- $(2^4) \times 5 + (\tan((pi/8+pi/8)))^2$
 - (16) $*5 + (tan((pi/4)))^2$
 - $-80 + (1)^{2}$

Notes

Notes

Lecture 02

Lecture 02



Notes

Operators Precedence

Look at the following Matlab expression

-2^4*5 + tan(pi/8+pi/8)^2

Guess the answer.

- $(2^4) * 5 + (tan((pi/8+pi/8)))^2$

$$-(16)*5+(tan((pi/4)))^2$$

 $-80 + (1)^2 = -80 + 1$



Operators Precedence

Look at the following Matlab expression

-2^4*5 + tan(pi/8+pi/8)^2

Practical Computing

Guess the answer.

Eugeniv Mikhailov (W&M)

 $- (2^4) *5 + (tan((pi/8+pi/8)))^2$

 $-(16)*5+(tan((pi/4)))^2$

 $-80 + (1)^2 = -80 + 1 = -79$

Rule of thumb: if not sure use extra parentheses ()

Operators Precedence

Look at the following Matlab expression

 $-2^{4}+5 + \tan(pi/8+pi/8)^{2}$

Practical Computing

Guess the answer.

Eugeniy Mikhailov (W&M)

- $(2^4) * 5 + (tan((pi/8+pi/8)))^2$

$$-(16) \times 5 + (tan((pi/4)))^2$$

 $-80 + (1)^2 = -80 + 1 = -79$

Rule of thumb: if not sure use extra parentheses ()

- Read more by executing doc precedence
- or searching for 'precedence' in the help browser.

Notes

Notes

Notes

Lecture 02

Lecture

Matrices

Recall that Matlab stands for Matrix Laboratory

- So deep inside everything is a matrix
 also referred as array or table
- $\bullet\,$ a number is the case of 1 \times 1 matrix

		+ = > + # > + E > + E > - E	୶୶ଡ଼
Eugeniy Mikhailov (W&M)	Practical Computing	Lecture 02	8 / 27
Matrices			

Recall that Matlab stands for Matrix Laboratory

- So deep inside everything is a matrix
 also referred as array or table
- $\bullet\,$ a number is the case of 1 \times 1 matrix

Let's create a 3×5 matrix (3 rows and 5 columns)

>>	Mz=ze	eros(3,	5)			
Mz	=					
	0	0	0	0	0	
	0	0	0	0	0	
	0	0	0	0	0	

This is not the only way, but it is one which make sure that matrix is filled with zeros

Note: it is possible to have more than 2 dimensional arrays.

						$\langle \Box \rangle \land \Box$	- ・モン・モン・モー	200
	Eugeniy Mikh	nailov (W&M)		Pract	ical Computing		Lecture 02	8/27
M	atrix e	lemen	ts as	signm	ent			
>>	→ Mz(2,	4)=1	% 2r	nd row,	4th co	lumn		
Mr	_							
1.12	. –							
	0	0	0	0	0			
	0	0	0	1	0			
	0	0	0	0	0			

Notes

Notes

Notes

Matrix elements assignment

Eugeniy Mikhailov (W&M)

>>	Mz(2,4)=1 %	2nd r	ow,	4th column		
Mz	=						
	0 0	0 0	0 0	0 1 0	0 0		
	0	0	0	0	0		
>>	Mz(3,5)=4 %	3rd r	ow,	5th column		
Mz	=						
	0	0	0	0	0		
	0	0	0	1	0		
	0	0	0	0	4		
					4	→ (B) (2) (2) (2)	୶ୡୡ
	Eugeniy Mikhailo	v (W&M)		Practi	cal Computing	Lecture 02	9 / 27

Practical Computing

Notes

Lecture 02

Alternative way to assign a matrix

• comma separates column elements

• semicolon separates row elements

>>	Mz	= [
Ο,	Ο,	Ο,	Ο,	0;				
Ο,	Ο,	Ο,	1,	0;				
Ο,	Ο,	Ο,	Ο,	4]				
Mz	=							
0		0		0	0	0		
0		0		0	1	0		
0		0		0	0	4		

Notice . . . mark, which means that input continues on the next line

		er har har i a	
Eugeniy Mikhailov (W&M)	Practical Computing	Lecture 02	10/2
Strength of Matlab			

Native matrix operations

Mz =	=							
0 0	0	0	0	>> Mz-	+5			
0 0	0	1	0	ans =				
0 0	0	0	4	5	5	5	5	5
				5	5	5	6	5
				5	5	5	5	9

5

ø

5

Lecture 02

Eugeniy Mikhailov (W&M) Strength of Matlab

Native matrix operations

Mz	z =	=			
0	0	0	0	0	
0	0	0	1	0	
0	0	0	0	4	

Eugeniy Mikhailov (W&M)

Eugeniy Mikhailov (W&M)

5	5	5	6	5
5	5	5	5	9
>> Mz*	2			
ans =				
0	0	0	0	0
0	0	0	2	0
0	0	0	0	8

5

More example on matrices operations

>> Mz+5 ans = 5

5

Mz =	>> Mz+	+Mz			
0 0 0 0 0	ans =				
0 0 0 1 0	0	0	0	0	0
0 0 0 0 4	0	0	0	2	0
	0	0	0	0	8

Practical Computing

Notes

Notes

Notes

More example on matrices operations

M:	Z =	=			>> Mz	+Mz			
0	0	0	0	0	ans =				
0	0	0	1	0	0	0	0	0	0
0	0	0	0	4	0	0	0	2	0
					0	0	0	0	8
algebra rules									
					>> Mz	*Mz'			
					ans =				
					0	0	0		
					0	1	0		
					0	0	16		
					Here <i>I</i> I Mz, i.e	<i>Iz'</i> corr . <i>Mz'(i</i> ,	esponds $j) = Mz$	to transj (j, i)	posed matrix

(🗇)

Lecture 02

Lecture 02

Lecture 02

13/27

10/07

Eugeniy Mikhailov (W&M) Practical Computing Matrix as a function argument

A function can take a matrix as the function argument, it will evaluate the value of the function for each matrix element

M:	Mz =				>> s	>> sin(Mz)					
0	0	0	0	0	ans =	=					
0	0	0	1	0	0	0	0	0	0		
0	0	0	0	4	0	0	0	0.8415	0		
					0	0	0	0	-0.7568		

Practical Computing

Vectors and column vector

A special case of the matrix is it has only one dimension. Such matrices generally called vectors

• *m* × 1 column vector

Eugeniv Mikhailov (W&M)

• $1 \times m$ just a vector

Notes

Notes

Vectors and column vector

A special case of the matrix is it has only one dimension. Such matrices generally called vectors

- *m* × 1 column vector
- $1 \times m$ just a vector
- To create a vector

Eugeniy Mikhailov (W&M)

>> % use comma to separate column elements >> v=[1, 2, 3, 4, 5, 6, 7, 8] v = 1 2 3 4 5 6 7 8 >> % alternatively you can use spaces >> v=[1 2 3 4 5 6 7 8]; >> % or mix of these two notations (NOT RECOMMENDED) >> v=[1 2 3, 4, 5, 6 7 8] v 3 4 5 6 7 1 2 8 Eugeniy Mikhailov (W&M) Lecture 02 14 / 27

Practical Computing

Notes

Notes

Notes

Construction of column vector

>> vc= % use	[1; : semi	2; 3 colo] n to	separate	row	elements	
vc =							
1							
2							

					$(\Box \rightarrow (\Box \rightarrow$	E - 14	200
E	Eugeniy Mikhailov	(W&M)	Pra	ctical Computing	Lecti	ure 02	15 / 27
⁄et	one mo	ore way	y to crea	ate matrix			
f yo	ou have p	rearrange	ed vectors	or column vec	tors you can us	e ther	n
>> >> >>	vc=[1; % note Mc=[vc,	2; 3]; that; vc, vo	after a c]	a statement	suppresses	outp	out
4c 1	= 1	1					
2	2	2					

·			2.40
	Lecture	e 02	16/27

Yet one more way to create matrix

If you have prearranged vectors or column vectors you can use them

Practical Computing

	Eugeniy Mikhailov	(W&M)		Practical Computing			Lectu	re 02 16 / 27
						$\langle \Box \rangle \prec D$	> + 2 > + 3	।) E १९५
3	6	9	12	15	18	21	24	
2	4	6	8	10	12	14	16	
1	2	3	4	5	6	7	8	
Mv	=							
>>	Mv=[v;	2*v;	3*v]					
1	2	3	4	5	6	7	8	
v	=							
3	3	3						
2	2	2						
1	1	1						
Mc	=							
>>	Mc=[vc,	, vc,	vc]					
>>	% note	that	; after	a st	atement	suppr	esses	output
>>	vc=[1;	2; 3]	;					

Colon (:) operator

The : operator is extremely useful to create vectors or matrix indexes It usually take form start:increment:stop and creates a vector with following values

[start, start+1*increment, ... , start+m*increment]

where

3

3

Eugeniy Mikhailov (W&M)

3

m=1, 2, 3, 4, ... and

 $\min(\texttt{start},\texttt{stop}) \leq \texttt{start} + \texttt{m} \star \texttt{increment} \leq \max(\texttt{start},\texttt{stop})$

17 / 27

Notes

Colon (:) operator

The : operator is extremely useful to create vectors or matrix indexes It usually take form start:increment:stop and creates a vector with following values

[start, start+1*increment, ... , start+m*increment]

Notes

where

m=1, 2, 3, 4, ... and min(start,stop)≤start + m*increment≤max(start,stop) >> v=5:2:11 v = 5 7 9 11

(**1**1) Practical Computing Eugeniy Mikhailov (W&M) Lecture 02 17/07 Colon (:) operator The : operator is extremely useful to create vectors or matrix indexes It usually take form start:increment:stop and creates a vector with following values [start, start+1*increment, ... , start+m*increment] where $m=1, 2, 3, 4, \ldots$ and min(start, stop) ≤ start + m*increment≤max(start, stop) >> v=5:2:11 v = 5 7 9 11 It is also possible to have negative increment >> v2=12:-3:1 v2 = 12 9 6 3 Eugeniv Mikhailov (W&M) Practical Com Lecture 02 Colon (:) operator continued

One can use form start: stop with the default increment = 1

>> v1=1:5

v1 =

1 2 3 4 5

Notes

Notes

Colon (:) operator continued

Eugeniv Mikhailov (W&M)

One can use form start:stop with the default increment = 1
>> v1=1:5

Practical Computing

1	=					
		1	2	2	л	

But there are some peculiarities:

>> v3=5:1 v3 = Empty matrix: 1-by-0

produces somewhat unexpected result, naively you would expect v3=5. But there are some built extra conditions, see them by executing >> help :

5

Eugeniy Mikhailov (W&M)

110165

Notes

Lecture 02

Lecture 02

18/27

Slicing matrices

It is handy to choose a subset (block) from the matrix We have a matrix Mv with size 3×8 and we want to choose all elements from columns 2,5,6

>>	Mv							
Mv	=							
1	2	3	4	5	6	7	8	
2	4	6	8	10	12	14	16	
3	6	9	12	15	18	21	24	
>>	Mv(:,[2,5,6])					
an	s =							
2	5	6						
4	10	12						
6	15	18						

The meaning of the : now is choose all. Notice also that we use vector to specify desired columns ・ロト・(部)・(三)・(三)・(三)・(の)()

Eugeniy Mikhailov (W&M)	Practical Computing	Lecture 02	19/27
Plotting			

Suppose you have a vector with values of x coordinates and we want to plot sin(x).

```
>> x=linspace(0,2*pi,10)
```

x = 1.3963 0 0.6981 2.0944 2.7925 3.4907 4.1888 4.8869 5.5851 6.2832 >> y=sin(x) у = 0 0.6428 0.9848 0.8660 0.3420 -0.3420 -0.8660 -0.9848 -0.6428 -0.0000 >> plot(x,y,'o') % alternatively plot(x,sin(x),'o') >> % every plot MUST have title, x and y labels >> xlabel('x (radians)') >> ylabel('sin(x)') >> title('Plot of sin(x)')

For 3D plots, please see help files for plot3, mesh, surf Eugeniy Mikhailov (W&M) Practical Computing

Increasing font size for plots

Default font size

Increased font size

Lecture 02

') sin(x)')

Size',24);

>>	plot(x,y,'o')	>>	plot(x,y,'o')
>>	% default font size	>>	<pre>set(gca, 'FontSize',24</pre>
>>	<pre>xlabel('x (radians)')</pre>	>>	<pre>xlabel('x (radians)')</pre>
>>	<pre>ylabel('sin(x)')</pre>	>>	<pre>ylabel('sin(x)')</pre>
>>	<pre>title('Plot of sin(x)')</pre>	>>	<pre>title('Plot of sin(x)</pre>



Saving plots



To save the figure use print.

>> print('-dpdf', 'sin_of_x')

This will generate file $\mathtt{sin_of_x.pdf}$ notice automatic file extension addition.

Notes

Notes

Notes



To save the figure use print.

>> print('-dpdf', 'sin_of_x')

This will generate file $\mathtt{sin_of_x.pdf}$ notice automatic file extension addition.

The -d switch designates the output format:

Eugeniv Mikhailov (W&M)

Eugeniy Mikhailov (W&M)

pdf, ps, eps, png... Eugeniy Mikhailov (W&M) ation Computin ecture 02 Saving plots continued

Matlab still generates pdf with a lot of empty space unsuitable for use as figures. It is better to save into eps format and then convert it to a desired one.

>> print('-deps', 'sin_of_x')

Or generate a png file which can be directly used with pdflatex

>> print('-dpng', '-r100', 'sin_of_x')

By default figure size is 8×6 inches, the -rswitch tells the figure resolution in dpi (dots per inch). In this case it is 100 dpi so resulting image will be 800 \times 600 pixels.



Array element-wise arithmetic operators

There are special arithmetic operators which applied to the elements of matrices (disregard linear algebra rules), they start with . (dot/period).

Practical Computing

• . * element-wise multiplication >> x=1:3

```
2
              3
x = 1
>> % x*x % will generate an error
>> x.*x % equivalent to x.^2 (see below)
ans = 1 4
                9
```

Array element-wise arithmetic operators

There are special arithmetic operators which applied to the elements of matrices (disregard linear algebra rules), they start with . (dot/period). • . * element-wise multiplication

Practical Computing

	>> x=1:3		
	x = 1	2 3	
	>> % x*x	% will generate an error	
	>> x.*x %	equivalent to x.^2 (see below)	
	ans = 1	4 9	
•	. ^ element-v	vise power operator	

>> x.^2 ans = 1 9 4

Notes

Notes

Notes

Notes

Lecture 02

24/2

Array element-wise arithmetic operators

There are special arithmetic operators which applied to the elements of matrices (disregard linear algebra rules), they start with . (dot/period).

- .* element-wise multiplication
 >> x=1:3
 x = 1 2 3
 >> % x*x % will generate an error
 >> x.*x % equivalent to x.^2 (see below)
 ans = 1 4 9
- . ^ element-wise power operator

>> x.^2 ans = 1 4 9

• . / element-wise division

>> x./x ans = 1 1 1

Eugeniy Mikhailov (W&M)

Eugeniy Mikhailov (W&M)

0 0

Eugeniy Mikhailov (W&M) Practical Computing

Array element-wise arithmetic operators continued

>> m=	=[1,2,3	;4,5,6;7	,8,9]				
n =							
L	2	3					
1	5	6					
7	8	9					
Linear algebra rules			Element-wise operation				
>> r	m*m			>> m	.*m		
ans	=			ans	=		
30	36	42		1	4	9	
66	81	96		16	25	36	
102	126	150		49	64	81	

Practical Computing

Array element-wise arithmetic operator . ^

>> m=[1,2,3;4,5,6;7,8,9]								
m =								
1	2	3						
4	5	6						
7	8	9						
Linea	r algebra	a rules	Element-wi	se operation				
>> m	^m % u	ndefined	>> m.^m					
			ans =					
			1	4	27			
			256	3125	46656			

Practical Computing

823543 16777216 387420489

Array element-wise arithmetic operator . /

>> m	>> m=[1,2,3;4,5,6;7,8,9]							
m =								
1	2	3						
4	5	6						
7	8	9						
Line	Linear algebra rules			Ele	ement-w	vise operati	on	
>>	m/m %	unity	matrix	>>	m./m	%matrix	of	ones
ans	; =			an	s =			
1	0	0		1	1	1		
0	1	0		1	1	1		

1

1

1

□ > < @ > < ≥ > < ≥ > ≥ つへで Lecture 02 27/27

Eugeniy Mikhailov (W&M) Practical Computing

1

Notes

Notes

ooturo 00

Lecture 02

Lecture 02

26 / 27

25/27

Notes