

# Matlab Workshop I

Niloufer Mackey and Lixin Shen

Department of Mathematics

Western Michigan University

Email: `{nil.mackey,lixin.shen}@wmich.edu`

# What is Matlab?

- **Matlab** is a commercial “**Matrix Laboratory**” package which operates as an interactive programming environment

# What is Matlab?

- **Matlab** is a commercial “**Matrix Laboratory**” package which operates as an interactive programming environment
- **Matlab** is well adapted to numerical experiments since the underlying algorithms for *Matlab’s built-in functions* and supplied *m-files* are based on the standard libraries LINPACK and EISPACK

# What is Matlab?

- **Matlab** is a commercial “**Matrix Laboratory**” package which operates as an interactive programming environment
- **Matlab** is well adapted to numerical experiments since the underlying algorithms for *Matlab’s built-in functions* and supplied *m-files* are based on the standard libraries LINPACK and EISPACK
- **Matlab** is an interactive program for numerical computation and data visualization

# Most Important Matlab Command: `help`

● `help`

a long list of help `topics` for which on-line help is available

# Most Important Matlab Command: `help`

- `help`

a long list of help **topics** for which on-line help is available

- `help topic`

a list of those **functions** under the topic for which help is available. For example, `help elmat` gives a list of elementary matrices and matrix manipulation

# Most Important Matlab Command: `help`

- `help`

a long list of help `topics` for which on-line help is available

- `help topic`

a list of those `functions` under the topic for which help is available. For example, `help elmat` gives a list of elementary matrices and matrix manipulation

- `help function-name`

information about a specific function. For example, `help eye` gives information about generating an identity matrix.

# Scalar, Vector, and Matrices

*Matlab works with a rectangular numerical matrix  
All variables represent matrices!!!*

# Scalar, Vector, and Matrices

*Matlab works with a rectangular numerical matrix  
All variables represent matrices!!!*

- Scalars: 1-by-1 matrices

# Scalar, Vector, and Matrices

*Matlab works with a rectangular numerical matrix  
All variables represent matrices!!!*

- **Scalars:** 1-by-1 matrices
- **Vectors:** matrices with only one row or one column

# Scalar, Vector, and Matrices

*Matlab works with a rectangular numerical matrix  
All variables represent matrices!!!*

- **Scalars:** 1-by-1 matrices
- **Vectors:** matrices with only one row or one column
- **Matrices**
  - **Entered by an explicit list of elements.** For example,  
 $A = [1 \ 2 \ 3; \ 4 \ 5 \ 6; \ 7 \ 8 \ 9]$
  - **Generated by built-in functions.** For example,  
`eye(m,n)`, `zeros(m,n)`, `ones(m,n)`,  
`linspace(a, b, n)`, `randn(m,n)`,  
`rand(m,n)`, `diag(v,k)`, `diag(A)`, `blkdiag`

# Values of Useful Constants

<code>pi</code>	3.14159265...
<code>i</code>	Imaginary unit, $\sqrt{-1}$
<code>j</code>	Same as <code>i</code>
<code>eps</code>	Floating-point relative precision, $\epsilon = 2^{-52}$
<code>realmin</code>	Smallest floating-point number, $2^{-1022}$
<code>realmax</code>	Largest floating-point number, $(2 - \epsilon)^{1023}$
<code>Inf</code>	Infinity
<code>NaN</code>	Not-a-number

# who(s), clear, save

● who

This shows all active variables. Try the command `whos`

# who(s), clear, save

- `who`

This shows all active variables. Try the command `whos`

- `clear a b`

This removes the variables `a` and `b`

# who(s), clear, save

- `who`

This shows all active variables. Try the command `whos`

- `clear a b`

This removes the variables `a` and `b`

- `clear all`

This remove all variables

# who(s), clear, save

- `who`

This shows all active variables. Try the command `whos`

- `clear a b`

This removes the variables `a` and `b`

- `clear all`

This remove all variables

- `save filename`

This creates a file `filename.mat` which contains all variables

# who(s), clear, save

- `who`

This shows all active variables. Try the command `whos`

- `clear a b`

This removes the variables `a` and `b`

- `clear all`

This remove all variables

- `save filename`

This creates a file `filename.mat` which contains all variables

- `save filename x y z`

This creates a file `filename.mat` which contains variables `x y z`

# Format Functions

<code>format</code>	Default. Same as <code>short</code>
<code>format short</code>	Fixed point with 4 decimal places
<code>format long</code>	Fixed point with 14 decimal places
<code>format short e</code>	Floating point with 4 decimal places
<code>format long e</code>	Floating point with 15 decimal places
<code>format short g</code>	Best of fixed or floating point with 4 DPs
<code>format long g</code>	Best of fixed or floating point with 14 DPs
<code>format hex</code>	Hexadecimal format
<code>format compact</code>	Suppress extra line-feeds
<code>format loose</code>	Put the extra line-feeds back in

Try the example `t = [4/3 1.2345e-6]`

# Sub-matrices and Colon Notation

$1:5$

$[1 \ 2 \ 3 \ 4 \ 5]$

$0.4:0.2:1.2$

$[0.4, 0.6, 0.8, 1.0, 1.2]$

$A(1:4, 3)$

the column vector with the first four entries of the third column

$A(:, 3)$

the third column of  $A$

$A(1:4, :)$

the first four rows of  $A$

$A(:, [2 \ 4])$

columns 2 and 4 of  $A$

$A(:, [2 \ 4]) = B(:, 1:2)$

replacing columns 2, 4

with the first two columns of  $B$

*Effort should be made to become familiar with them!!!*

# Standard Operators

+	addition
−	subtraction
*	multiplication
^	power
'	conjugate transpose
\	left division
/	right division
$x = A \setminus b$	the solution of $A * x = b$
$x = b / A$	the solution of $x * A = b$

# Pointwise Operation

- + addition
- − subtraction
- .\* Element-by-element multiplication
- .^ Element-by-element power
- .\ Element-by-element left division
- ./ Element-by-element right division
- .' Unconjugate array transpose (same as `transpose(A)`)

# Pointwise Operation: Some Examples

Matlab Command

Output

```
[1,2,3].*[1,2,3]
```

```
[1,4,9]
```

```
[1,2,3].^2
```

```
[1,4,9]
```

```
[2,4,6]./2
```

```
[1,2,3]
```

```
2./[2,4,6]
```

```
[1.000 0.5000 0.3333]
```

```
[2,2,2]./[2,4,6]
```

```
[1.000 0.5000 0.3333]
```

```
[2,4,6].\2
```

```
[1.000 0.5000 0.3333]
```

```
[1 1+2*i; i 3].'
```

```
1.0000 0+1.0000i
```

```
1.0000+2.0000i 3.0000
```

# Math Functions

The Matlab command `help elfun` gives a list of elementary math functions

- **Trigonometric:** `sin`, `sinh`, `asin`, `asinh`, `cos`, `cosh`, `acos`, `acosh`, `tan`, `tanh`, `atan`, `atan2`, `atanh`, `sec`, `sech`, `asech`, `csc`, `csch`, `acsc`, `acsch`, `cot`, `coth`, `acot`, `acoth`

*The variables of these functions could be matrices!!!*

# Math Functions

The Matlab command `help elfun` gives a list of elementary math functions

- **Trigonometric:** `sin, sinh, asin, asinh, cos, cosh, acos, acosh, tan, tanh, atan, atan2, atanh, sec, sech, asech, csc, csch, acsc, acsch, cot, coth, acot, acoth`
- **Exponential:** `exp, log, log10, log2, pow2, sqrt, nextpow2`

*The variables of these functions could be matrices!!!*

# Math Functions

The Matlab command `help elfun` gives a list of elementary math functions

- **Trigonometric:** `sin, sinh, asin, asinh, cos, cosh, acos, acosh, tan, tanh, atan, atan2, atanh, sec, sech, asech, csc, csch, acsc, acsch, cot, coth, acot, acoth`
- **Exponential:** `exp, log, log10, log2, pow2, sqrt, nextpow2`
- **Complex:** `abs, angle, complex, conj, imag, real, unwarp, isreal, cplxpair`

*The variables of these functions could be matrices!!!*

# Math Functions

The Matlab command `help elfun` gives a list of elementary math functions

- **Trigonometric:** `sin, sinh, asin, asinh, cos, cosh, acos, acosh, tan, tanh, atan, atan2, atanh, sec, sech, asech, csc, csch, acsc, acsch, cot, coth, acot, acoth`
- **Exponential:** `exp, log, log10, log2, pow2, sqrt, nextpow2`
- **Complex:** `abs, angle, complex, conj, imag, real, unwarp, isreal, cplxpair`
- **Rounding and remainder:** `fix, floor, ceil, round, rem, sign`

*The variables of these functions could be matrices!!!*

# Hardcopy

- The Matlab command

`diary filename`

causes what appears subsequently on the screen (except graphics) to be written to the named diskfile (if no file is specified, the file 'diary' is used). To save space, don't forget the command `format compact`

# Hardcopy

- The Matlab command

`diary filename`

causes what appears subsequently on the screen (except graphics) to be written to the named diskfile (if no file is specified, the file 'diary' is used). To save space, don't forget the command `format compact`

- The Matlab command

`diary off`

suspends `diary`

# Hardcopy

- The Matlab command

`diary filename`

causes what appears subsequently on the screen (except graphics) to be written to the named diskfile (if no file is specified, the file 'diary' is used). To save space, don't forget the command `format compact`

- The Matlab command

`diary off`

suspends `diary`

- The Matlab command

`diary on`

turns `diary` back on