

数学软件 Matlab

—— Matlab 基础

主要内容

- Matlab 介绍
- Matlab 数值计算
- Matlab 符号运算
- Matlab 绘图
- Matlab 数据类型
- Matlab 程序设计
- Matlab 文件操作

本讲内容

■ Matlab 介绍

- Matlab 的安装
- Matlab 特点与功能
- Matlab 工作界面

■ Matlab 数值计算

- 变量，数据类型，矩阵，...
- 向量运算，矩阵运算

科学计算编程与软件

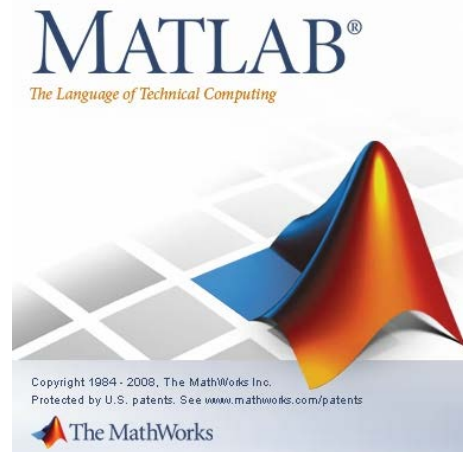
- 科学计算编程与软件

- 编程语言：FORTRAN, C, C++
- 数值计算：Matlab, Scilab, Python
- 符号计算：Mathematica, Maple
- 统计软件：SAS, SPSS, R语言
- 数学规划：Lingo, Lindo
- 工程计算：Phoenics, Fluent, Ansys, FreeFem++
- 线性代数程序库：BLAS, LAPACK, ARPACK

Matlab 介绍

Matlab → Matrix Laboratory 矩阵实验室

Matlab是一种广泛应用于**工程计算及数值分析**领域的新型高级语言，自1984年推向市场以来，已成为国际公认的最优秀的工程应用开发环境。



● Matlab 的发行

1984 年，Matlab 1.0（DOS版，182K，20多个函数）

1993 年推出 Windows 版，加入 Simulink

自 2006 年起，Matlab 每年更新两次（3 月份和9 月份）

● 安装过程见课程主页 (以R2009a为例)，建议安装

Matlab, Curve Fitting, Optimization, PDE, Symbolic Math

Matlab 的功能与特点

- Matlab 是一个交互式软件系统
 - 输入一条命令，立即就可以得到该命令的运行结果
- Matlab 具有很强的数值计算功能
 - 以矩阵为基本操作单位，并实行动态定维
 - 提供十分丰富的数值计算函数，方便计算，提高效率
 - Matlab 命令与数学公式非常接近，可读性强，容易掌握
- Matlab 的符号计算功能
- Matlab 的绘图功能
- Matlab 的编程功能
 - 具有面向对象程序语言特征，简单易学、编程效率高

Matlab 的特点与功能

- Matlab 丰富的工具箱（toolbox）

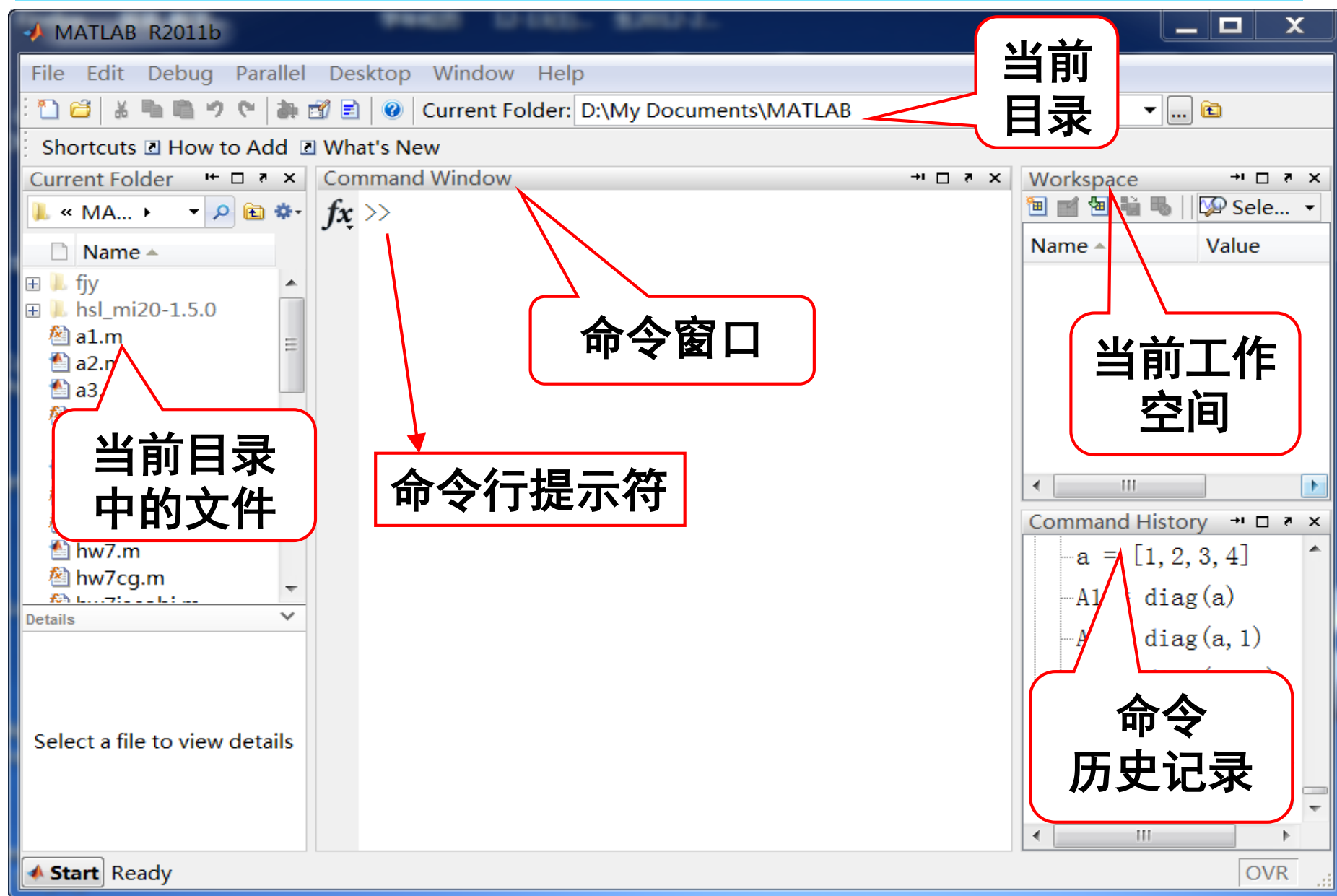
根据专门领域中的特殊需要而设计的各种可选工具箱

Symbolic Math	Signal process	Control System
PDE	Image Process	System Identification
Optimization	Statistics	

- Matlab 的 Simulink 动态仿真集成环境

提供建立系统模型、选择仿真参数和数值算法、启动仿真程序对该系统进行仿真、设置不同的输出方式来观察仿真结果等功能

Matlab 的工作界面



本讲内容

■ Matlab 介绍

■ Matlab 基础

- 变量，数据类型，矩阵， ...
- 向量运算，矩阵运算

Matlab 基础

- 变量与常量
- 数据类型
- 字符串
- 运算符
- 语句与表达式

变量

- 变量是任何程序设计语言的基本元素之一
- Matlab 变量的特点：
 - 不要求事先声明
 - 不需要指定变量类型：根据所赋的值或对变量所进行的操作来确定变量的类型
 - 变量的类型可随时改变：在赋值时，若变量已经存在，则用新值代替旧值，并以新数据的类型作为变量类型

如果事先给大数组分配空间，则可以提高程序的执行效率

- 变量命名要求
 - 必须以字母开头，含字母（大小写）、数字和下划线
 - 变量名长度不超过 65（Matlab6.5.1 版本以上）
 - 区分大小写

常量

- 常量：值不会改变的量，包括常数和符号常量，如 3.14
- Matlab 预定义常量

i, j	虚数单位	realmin	最小正浮点数
pi	圆周率	realmax	最大正浮点数
eps	浮点运算相对精度	intmin	最小整数
Inf	无穷大	intmax	最大整数
NaN/nan	不定值		

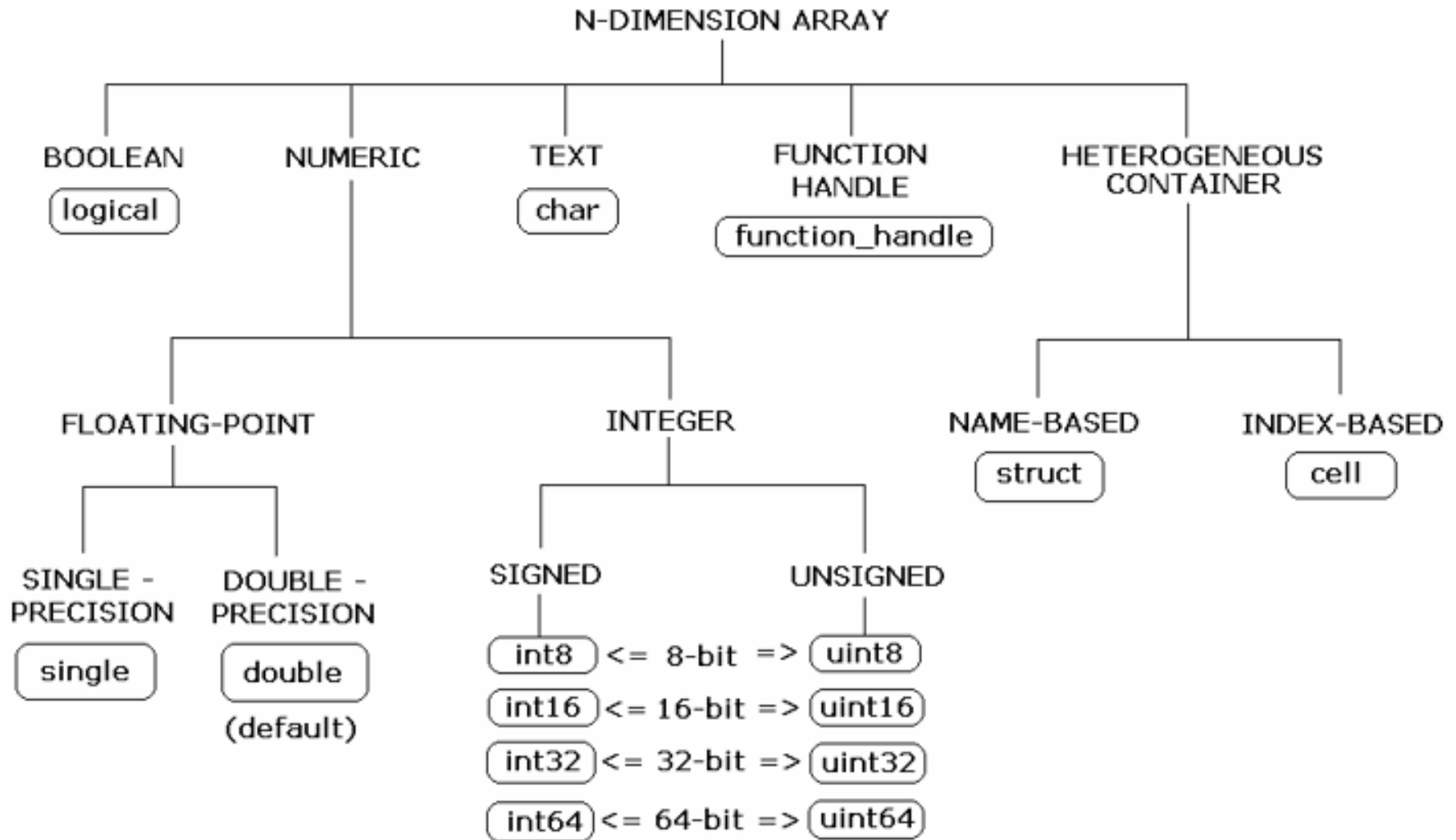
注：应尽量避免给系统预定义变量重新赋值！

- 特殊变量 ans

变量的查看

- 查看已定义的变量： **who**、**whos**
 - **who** 显示工作空间中的所有变量
 - **whos** 显示变量的详细属性
- 清除变量： **clear**
 - **clear** 清除工作空间中的所有变量
 - **clear** 变量名 清除指定的变量

Matlab 数据类型



MatlabR2011b 中的数据类型

Matlab 数据类型

- Matlab 常用的数据类型有：

- `numeric`: 数值
- `char`: 字符
- `logical`: 逻辑
- `cell`: 单元、细胞、元胞
- `struct`: 结构

详细介绍见后面的讲义

- Matlab中的字符串：用单引号括起来的字符序列

Matlab 基本操作

● Matlab 中的数

- Matlab 中的数默认是双精度实数，表示方法同 C 语言

3, -9, 0.4, 1.603e-12, 3.23e+20

- 浮点运算（加减乘除，开方）的相对误差为 `eps`
- 浮点数表示范围： $10^{-308} \sim 10^{308}$
- 复数：虚部单位为 `i` 或 `j`

`z=3+4i` (4 与 i 之间不能有空格)

● 算术运算符

<code>+</code>	<code>-</code>	<code>*</code>	<code>^</code>	加，减，乘，幂
<code>/</code>	<code>\</code>			右除，左除

Matlab 基本操作

- Matlab 语句的通常形式

变量 = 表达式

表达式是用运算符将有关运算量连接起来的式子，其结果被赋给赋值号“=”左边的变量

- 命令或语句的运行：回车

- 命令分隔符：逗号和分号

- 若不想在屏幕上输出结果，可以在语句最后加分号

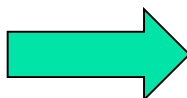
- 续行符：... （三个连续的点）

- 如果语句很长，可用续行符 分成多行

矩阵操作

- 矩阵的输入
- 矩阵元素赋值
- 矩阵元素的应用
- 特殊矩阵生成函数
- 矩阵基本运算
- 矩阵的旋转
- 矩阵的形状与大小
- 将函数作用到矩阵上

矩阵操作

- Matlab 的操作对象  矩阵

- 定义矩阵：直接输入法

例：>> A = [1 2 3; 4 5 6; 7 8 9]

- 适用小矩阵的输入
- 矩阵用方括号 “[]” 括起
- 矩阵同一行中的元素之间用 空格 或 逗号 分隔
- 矩阵行与行之间用 分号 分开
- 直接输入法中，分号可以用 回车 代替

>> 是命令提示符，不用输入

矩阵元素

- 矩阵整体赋值：元素可以是任何数值表达式

例： `x=[-1.3, sqrt(3), (1+2+3)*4/5]`

- 矩阵元素的单独赋值

例： `x(2)=6.8`

例： `x(5)=abs(x(1))`

动态定维功能： Matlab 会自动扩展向量的长度，
并将未赋值部分置零

例： `>> x(5)=abs(x(6)) ??`

矩阵元素

- 空矩阵

例：>> B=[]

- 大矩阵可以把小矩阵作为其元素

例：>> A=[A ; 11 12 13]

→ 在原矩阵的下方加一行

矩阵元素的引用

● 单个元素的引用

$x(i)$	向量 x 中的第 i 个元素
$A(i,j)$	矩阵 A 中的第 i 行, 第 j 列元素

例: $x=A(2,3)+A(1,2)$

● 多个元素的引用: 冒号的特殊用法

$a:b:c$

- 产生一个由等差序列组成的向量
- a 是首项, b 是公差, c 确定最后一项
- 若 $b = 1$, 则 b 和其前面的冒号可以省略

例: $x=1:2:5$
 $y=1:2:6$

例: $x=2:1:5$
 $y=2:5$

例: $x=3:1:0$

矩阵元素的引用

● 多个元素的引用（续）

$x(i:j)$	向量 x 中的第 i 到第 j 个元素
$A(i:j,m:n)$	由第 i 至 j 行和第 m 至 n 列组成的子矩阵

例： $y=x(1:3)$

例： $B=A(2:3,1:3)$
 $y=A(1,1:3)$

$x(i:end)$	向量 x 中的第 i 个到最后一个元素
$A(i:end,m:n)$	第 i 行到最后一行与 m 至 n 列（子矩阵）
$A(i:j,m:end)$	第 m 列到最后一列与 i 至 j 行（子矩阵）

矩阵元素的引用

● 多个元素的引用（续）

$A(:,k)$	矩阵的第 k 列
$A(i,:)$	矩阵的第 i 行
$A(i:j,:)$	矩阵的第 i 行到第 j 行
$A(:,m:n)$	矩阵的第 m 行到第 n 列
$A(:, :)$	整个矩阵
$x(:)$	取向量的所有元素，并按 列向量 方式输出
$A(:)$	将矩阵的所有元素 按列 排成一个 列向量

● 注意 $A(:)$ 与 $A(:, :)$ 的区别

矩阵元素的引用

- 多个元素的引用：不连续元素的选取

$A([i_1, i_2, \dots, i_p], [j_1, j_2, \dots, j_q])$

矩阵第 $i_1, i_2, \dots, i_p$ 行和第 $j_1, j_2, \dots, j_q$ 列组成的子矩阵

例： $A([1, 3, 4], [2, 3])$

例： $A([1, 4, 3], [2, 3])$

- 删除指定的行或列

$A(i, :) = []$	删除第 i 行
$A(:, j) = []$	删除第 j 列

特殊矩阵

● 特殊矩阵的生成

- Matlab 提供了一些函数，用于生成一些常见的特殊矩阵

例：

```
A=magic(3)
B=ones(4)
```

```
A=magic(5)
a1=diag(A)
a2=diag(A,1)
a3=diag(A,-2)
```

```
a=[1,2,3,4]
A1=diag(a)
A2=diag(a,1)
A3=diag(a,-2)
```

注意 **diag** 的用法

常见矩阵生成函数

<code>zeros(m,n)</code> <code>zeros(n)</code>	生成一个 m 行 n 列的零矩阵 $m=n$ 时可简写为 <code>zeros(n)</code>
<code>ones(m,n)</code> <code>ones(n)</code>	生成一个 m 行 n 列的元素全为 1 的矩阵 $m=n$ 时可简写为 <code>ones(n)</code>
<code>eye(m,n)</code> <code>eye(n)</code>	生成一个主对角线全为 1 的 m 行 n 列矩阵 $m=n$ 时可简写为 <code>eye(n)</code> ，即为 n 维单位矩阵
<code>diag(X)</code> <code>diag(X,k)</code>	若 X 是矩阵，则 <code>diag(X)</code> 为 X 的主对角线向量 若 X 是向量， <code>diag(X)</code> 产生以 X 为主对角线的对角矩阵
<code>tril(A)</code>	提取一个矩阵的下三角部分
<code>triu(A)</code>	提取一个矩阵的上三角部分
<code>rand(m,n)</code> <code>rand(n)</code>	产生 0~1 间均匀分布的随机矩阵 $m=n$ 时简写为 <code>rand(n)</code>
<code>randn(m,n)</code> <code>randn(n)</code>	产生均值为0，方差为1的标准正态分布随机矩阵 $m=n$ 时简写为 <code>randn(n)</code>
其它特殊矩阵生成函数： <code>magic</code> 、 <code>hilb</code> 、 <code>pascal</code> 等	

矩阵基本运算

- 矩阵的加减：对应分量进行运算

例：

$$A=[1,2; 3,4]; B=[5,6; 7,8]$$

$$C=A+B$$

$$D=B-A$$

参与加减运算的矩阵具有 相同的维数！

- 矩阵的普通乘法

例：

$$A=[1,2,3; 4,5,6]; B=[2,1; 4,3]$$

$$C=B*A$$

参与运算的矩阵须满足线性代数中矩阵相乘的原则！

矩阵基本运算

- 矩阵的除法：若 A 可逆方阵，则

$$B/A \iff A \text{ 的逆右乘 } B \iff B * \text{inv}(A) \longrightarrow \boxed{\text{右除}}$$

$$A \setminus B \iff A \text{ 的逆左乘 } B \iff \text{inv}(A) * B \longrightarrow \boxed{\text{左除}}$$

- 矩阵的幂：若 A 是方阵， p 是正整数，则

$$A^p \implies A \text{ 的 } p \text{ 次幂，即 } p \text{ 个 } A \text{ 相乘}$$

- 矩阵的转置与共轭转置

A'	共轭转置
$A.'$	普通转置，不取共轭，点与单引号之间不能有空格

例：

$$A = [1, 2-3i; 3, 1+4i]$$

$$B = A'$$

$$C = A.'$$

翻转与旋转

● 矩阵的翻转与旋转

<code>fliplr(A)</code>	左右翻转
<code>flipud(A)</code>	上下翻转
<code>rot90(A)</code>	逆时针旋转 90 度
<code>rot90(A,k)</code>	逆时针旋转 $k \times 90$ 度

例:

```
A = [1 2 3;4 5 6]
B = fliplr(A)
C = flipud(A)
D = rot90(A)
E = rot90(A,-1)
```

注意矩阵旋转与转置的区别！

改变矩阵的形状

- 改变矩阵的形状: `reshape(A,m,n)`

将矩阵元素按 **列方向** 进行重新排列成一个 $m \times n$ 的新矩阵

新矩阵的元素个数必须与原矩阵元素个数相等!

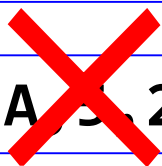
例:

```
A=[1,2,3; 4,5,6; 7,8,9; 10,11,12]
```

```
B=reshape(A,4,3)
```

```
C=reshape(A,2,6)
```

```
C=reshape(A,3,2)
```



查看矩阵的大小

- 查看矩阵的大小： `size`、`length`

<code>size(A)</code>	返回矩阵 A 的行数和列数
<code>size(A,1)</code>	返回矩阵 A 的行数
<code>size(A,2)</code>	返回矩阵 A 的列数
<code>length(x)</code>	若 x 是向量，则返回 x 的长度
<code>length(X)</code>	若 X 是矩阵，则返回行数和列数中大的一个
<code>numel(A)</code>	返回 A 的元素的个数

例：

```
A=[1,2,3; 4,5,6]
size(A)
size(A,1)
size(A,2)
length(A), numel(A)
x=1:5; length(x)
```

矩阵的数组运算

- 数组运算：对应元素进行运算

- 数组运算包括：点乘、点除、点幂
- 相应的四个数组运算符为： `.*` `./` `.\` `.^`

点与算术运算符之间不能有空格！

参与运算的对象必须具有相同的形状！

例：

```
A=[1,2,3; 4,5,6];  
B=[3,2,1; 6,5,4];  
C=A.*B  
D=A./B  
E=A.^B
```

函数取值

● 函数作用在矩阵上的取值

设 x 是变量， f 是一个函数

- 当 $x = a$ 是标量时， $f(x) = f(a)$ 也是一个标量
- 当 $x = [x_1, x_2, \dots, x_n]$ 是向量时，则
 $f(x) = [f(x_1), f(x_2), \dots, f(x_n)]$ 是一个与 x 长度相同的向量

f 作用在 x 的每个分量上！

- 若 A 是矩阵，则 $f(A)$ 是一个与 A 同形状的矩阵

$$f(A) = \begin{pmatrix} f(a_{11}) & f(a_{12}) & \dots & f(a_{1n}) \\ f(a_{21}) & f(a_{22}) & \dots & f(a_{2n}) \\ \vdots & \vdots & \ddots & \vdots \\ f(a_{m1}) & f(a_{m2}) & \dots & f(a_{mn}) \end{pmatrix}$$

例：

```
x=[0:pi/4:pi];  
A=[1,2,3; 4,5,6];  
y1=sin(x)  
y2=exp(A)  
y3=sqrt(A)
```

矩阵的超越函数

- 矩阵函数

怎样计算 e^A ?

- 将矩阵作为一个整体参与运算
- Matlab 提供的矩阵函数: `expm`、`sqrtn`、`logm`
详情参见联机帮助 (`help expm / squarm / logm`)
- 更一般的矩阵函数: `funm`

`funm(A, @fun)`

`fun` 可以是 `exp`, `log`, `cos`, `sin`,

矩阵与数的运算

● 矩阵与数的运算

- **加减**：矩阵的每个元素都与数作加减运算
- **数乘**：矩阵的每个元素都与数作乘法运算
- **矩阵除以一个数**：每个元素都除以这个数
- **数与矩阵的点幂运算**：采用数组运算

例：

```
x=[1 2 3];
```

```
x.^2=[1^2,2^2,3^2]=[1,4,9]
```

```
2.^x=[2^1,2^2,2^3]=[2,4,8]
```

Matlab中所有标点符号必须在英文状态下输入!

常用数学函数

`sin(x)`、`cos(x)`、`tan(x)`、`cot(x)`、
`sec(x)`、`csc(x)`、`sinh(x)`、`cosh(x)`、`tanh(x)` ...

`asin(x)`、`acos(x)`、`atan(x)`、`acot(x)`、
`asec(x)`、`acsc(x)` ...

`exp(x)` % e^x 自然指数 (以 e 为底)
`pow2(x)` % 2^x 以 2 为底的指数

`log(x)` % $\ln(x)$ 自然对数 (以 e 为底)
`log2(x)` % 以 2 为底的对数
`log10(x)` % 以 10 为底的对数

`sqrt(x)` % 平方根

`abs(x)` % 绝对值

常用数学函数

`conj(z)` % 复数的共轭
`real(z)` % 复数的实部
`imag(z)` % 复数的虚部
`angle(z)` % 复数的辐角

`sign(x)` % 符号函数

`round(x)` % 取整函数 (四舍五入)
`fix(x)`、`floor(x)`、`ceil(x)` % 另外三个取整函数

`mod(x,y)` % 计算 x 除以 y 的余数 (结果与 y 同号)
`rem(x,y)` % 计算 x 除以 y 的余数 (结果与 x 同号)

`gcd(x,y)` % 最大公约数

`lcm(x,y)` % 最小公倍数

常用数学函数

`linspace(a,b,n)` % 生成 a 到 b , n 个数的等差数列

`logspace(a,b,n)` % 生成 10^a 到 10^b , n 个数的等比数列

`max(x)` % 求 向量 x 中的最大值
`min(x)` % 求最小值
`mean(x)` % 求平均值
`sum(x)` % 求和
`sort(x)` % 排序

若参数 x 是矩阵，
则作用在其各列上

`det(A)` % 矩阵行列式
`inv(A)` % 矩阵的逆
`eig(A)` % 矩阵的特征值
`rank(A)` % 矩阵的秩

更多数学函数，参见课程主页

Matlab 的输出格式

● 输出格式

- Matlab 以双精度执行所有的运算，运算结果可以在**屏幕上输出**，同时**赋给指定变量**；若无指定变量，则系统会自动将结果赋给变量 “**ans**”
- 在屏幕上输出运算结果时，可以通过 **format** 命令指定输出格式

format 只改变输出格式，不会改变变量的值！

各种 format

格式	解释	例
<code>format</code>	短格式（缺省格式），同 <code>short</code>	<code>3.1416</code>
<code>format short</code>	短格式（缺省格式），小数点后 4 位	<code>3.1416</code>
<code>format long</code>	长格式，双精度 15 位，单精度 7 位	<code>3.14159265358979</code>
<code>format shortE</code>	短 E 格式（科学计数格式）	<code>3.1416e+000</code>
<code>format longE</code>	长 E 格式	<code>3.141592653589793e+000</code>
<code>format shortG</code>	短 G 格式	<code>3.1416</code>
<code>format longG</code>	长 G 格式	<code>3.14159265358979</code>
<code>format rat</code>	分数形式	<code>355/113</code>
<code>format compact</code>	紧凑格式	
<code>format loose</code>	宽松格式	
<code>format +</code> , <code>format bank</code> , <code>format hex</code> (详情查看联机帮助)		

变量的存储

- 将变量保存到 mat 文件中

<code>save</code>	将所有变量保存到文件 <code>matlab.mat</code> 中
<code>save fname</code> <code>save fname.mat</code>	将所有变量保存到文件 <code>fname.mat</code> 中
<code>save fname 变量</code>	将指定变量保存到文件 <code>fname.mat</code> 中

- 可同时保存多个变量，各变量之间用空格隔开

例：

```
x=2:5; A=magic(3); a=2.4;  
save mydata A x;
```

- 从 mat 文件中读取变量

<code>load fname</code>	读取 <code>fname.mat</code> 中的所有变量
<code>load fname 变量</code>	从 <code>fname.mat</code> 中读取指定的变量

Matlab帮助系统

- 如何寻找帮助

<code>help</code>	显示指定命令的简短使用说明
<code>doc</code>	以网页形式显示完整的使用帮助

例:

<code>help size</code>
<code>doc size</code>

- 查找命令

<code>lookfor</code>	按指定的关键词查询与之相关的命令
<code>which</code>	显示命令或函数所在的目录

- 其它相关命令

`cd`、`dir`、`more`

更多 Matlab 命令，参见课程主页

几个小技巧

- Matlab 的命令记忆功能：上下箭头键

可以先输入命令的前几个字符，再按上下键缩小搜索范围

- 命令补全功能：Tab 键
- 用 Esc 键 删除命令行
- 命令 home、clc