

MATLAB 语言及应用

MATLAB Language and Its Applications

第一章 MATLAB 基础入门

Ch.1 MATLAB Fundamentals

刘世东

Shidong Liu

School of Physics and Physical Engineering
Qufu Normal University



March 12, 2020

本章目录

1. MATLAB 桌面

2. 命令窗口的指令输入

3. 数值, 变量与表达式

3.1 数值

3.2 变量

3.3 赋值语句

3.4 预定义变量

4. MATLAB 中的基本数学函数

5. 关系操作和逻辑操作

6. M 文件

6.1 M 文件的类型及创建

6.2 例子: 函数文件

7. MATLAB 控制流

8. MATLAB 中常用的控制指令

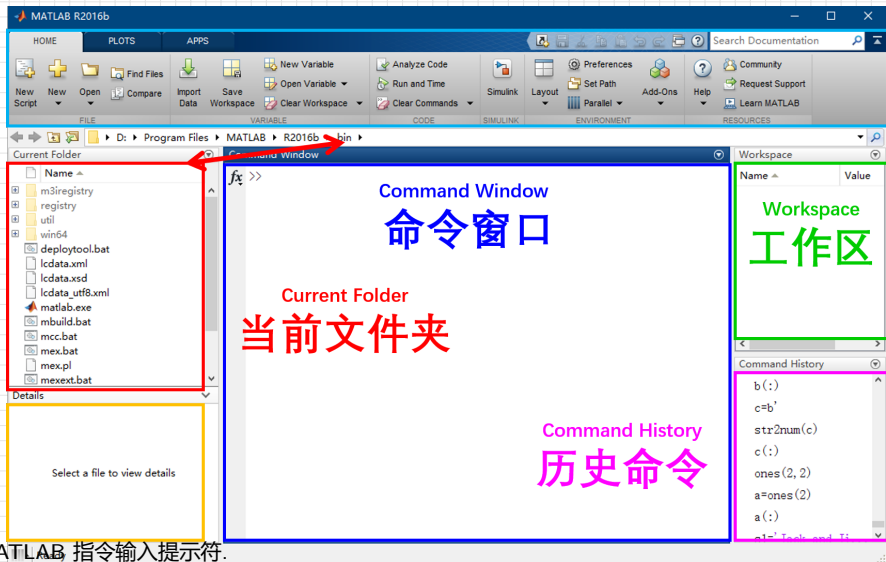
9. 数据或者变量的存取

9.1 保存

10. p 码文件

11. MATLAB 的帮助系统


MATLAB 桌面



命令窗口内的指令输入与运行—简单的数值计算

数学运算由操作数 (Operands, 如数) 和操作符 (Operators, 如加号).

MATLAB 中数学运算方式: 命令窗口内输入数学公式, 回车运行.



A screenshot of the MATLAB Command Window. The title bar is blue with the text 'Command Window' and a close button. The window has a white background with a light gray grid. The text inside shows a command prompt '>>' followed by '1+2'. Below this, the output 'ans = 3' is displayed. At the bottom, there is a prompt 'fx >>' followed by a vertical cursor line.

```
>> 1+2  
  
ans =  
  
3  
  
fx >> |
```

分别输入以下数学表达式

$2*3-4/5$ 输出结果为 $\text{ans} = 5.2000$

$2*(3-4)/5$ 输出结果为 $\text{ans} = -0.4000$

$931.5*3^2$ 输出结果为 $\text{ans} = 8.3835\text{e}+03$

以上计算可简单体现 MATLAB 的数值和运算特点/规律 (小小的一部分).

MATLAB 的数值特点/规律

- ① MATLAB 中数值用十进制, 如 233 2.33 0.233; 小数, 0 可省略, 如 .233.
- ② 可用科学记数法表示数值 (e或E), 如 6.02e23 1.6e-19
- ③ MATLAB 的数值计算, 默认是定义在复数域上的浮点计算, 标准是 IEEE754:1985 < 二进制浮点运算 >.
- ④ MATLAB 的默认计算精度为 double (双精度), 即 64 位, 故 MATLAB 中数值的精度为 2^{-52} , 即 MATLAB 中的预定义变量 eps 的数值.
- ⑤ 显示精度不是计算精度.
format long 显示更多有效数字 (默认为 format short)¹
- ⑥ 运算顺序与实运算顺序一样, 遵循: 括号() - 幂^ - 乘除* / - 加减+ -.
- ⑦ MATLAB 默认返回值存储在变量 ans 中 (缺省变量)²

NOTE: 操作数与操作符之间可以由空格. 适当加空格可以增加代码的可读性 (Readable code is debugable code).

¹其他显示方式可以通过 help format 或 doc format 查看

²answer 的简写.

MATLAB 中的变量命名规则

MATLAB 比计算器的强大之一是可以将数值赋值给一个变量. 这样可以重复利用计算结果, 简化代码, 如 `x = 23333` . [More doc Variable Names](#)

- 变量/函数以字母 (A-z) 开头, 之后可以是任意字母 (A-z), 数字 (1-0) 或下划线 (_)
- 变量/函数名区分大小写, so A is different from a.
- 变量/函数名不允许使用空格和标点符号
- 变量/函数名最多 63 个字符 (namelengthmax 命令查看)

以下变量/函数名合法:

Abc; abc; a_bc; a1bc; a123; a_1

Abc 与 abc 是两个不同的变量/函数.

以下变量/函数名不合法

_bc; 1abc; a bc; a,bcq12

NOTE: 以上命名虽然合法, 但是丑陋且晦涩难懂不知所云...

4 种常见的命名方法

变量或函数的命名应该满足: 便于辨认且具有指向性以方便代码维护³.

比较流行的有 4 种: 驼峰命名法、匈牙利命名法、帕斯卡命名法和下划线命名法.

- 驼峰命名法: 混合使用大小写字母来构成变量和函数的名字, 例如 `calCircleAera()`.
每一个逻辑断点处用大写字母区分. 使用常规缩写, 冠词不要 (也可以使用汉语拼音, 如 `jisuanYuanMianji()`, 此时一般词语一组). 正确表达意思, 不必在意语法.
- 帕斯卡命名法: 与驼峰命名法类似, 区别只是首字母大写, 如 `CalCircleAera()`. 从键盘输入上来说, 驼峰更快.
- 下划线命名法: 与以上类似, 只是逻辑点用下划线表示, 如 `cal_circle_aera()`.
- 匈牙利命名法: 匈牙利命名法通过在变量名前面加上相应的小写字母的符号标识作为前缀, 标识出变量的作用域、类型等, 例如 `char_data`, `double_data`.

³长期来看, 代码的维护代价远高于开发...

赋值语句

赋值语句: 变量 = 数学表达式:

```
yuanZhouLv = 3.1415926; %= 是赋值不是等于, 切切  
AvogadrosConstant = 6.02e23;  
goldenRatio = 0.618;  
ultimateAnswer = 42; %ans to life, the unvs, and evrth  
aComplexNum = -3 + 4i; %also -3 + 4j
```

- 分号 ; 使运算结果不显示, 但变量已生成, 存在工作区.⁴
- 百分号 % 后为注释⁵;
- **MATLAB 是面向复数的**. 定义方式还有
 $x = \text{complex}(2,3)$; 或者 $x = 5 * \exp(1i * \pi / 3)$; (可以省略 1, 但建议不要)
 $x = 2 + 3 * i$; 这种方式不建议使用.

⁴产生的变量可以通过clear 或 clear yourVal 清楚

⁵其他标点符号的作用后续的学习中会陆续用到

MATLAB 中的预定义变量/常数

预定义	含义	预定义	含义
ans	计算结果的默认变量	intmax	最大正整数
eps	浮点数相对精度 2^{-52}	intmin	最小负整数
pi	圆周率	realmax	最大正实数
Inf 或 inf	无穷大 (被 0 除导致)	realmin	最小正实数
i 或 j	虚数单位	NaN 或 nan	非数 (Not a Number) $0/0$ 或 ∞/∞
nargin	函数的输入变量数	nargout	函数输出变量数目

- MATLAB 的预定义变量本质是内部的一些函数, 有的没有输入参数 (如 pi), 有的有输入参数 (如 eps).
- 用户自定义变量**尽量**不要与 MATLAB 预定义变量 (如 pi), 内部函数 (如 exp), 关键词, 文件夹重名 (如 R2016b).
 iskeyword usrName 可判断是否是关键词 (如 for, while, 目前共有 20 个关键词, 运行 iskeyword 可以查看!).
 exist usrName 可判断是否预定义变量, 内部函数, 文件夹重名.
 isvarname 判断变量名是否合法.

MATLAB 中的基本数学函数

MATLAB 拥有所有的数学基本函数, 通过 `help elfun` 查看.

三角函数类:

Trigonometric.

[sin](#) - Sine.
[sind](#) - Sine of argument in degrees.
[sinh](#) - Hyperbolic sine.
[asin](#) - Inverse sine.
[asind](#) - Inverse sine, result in degrees.
[asinh](#) - Inverse hyperbolic sine.
[cos](#) - Cosine.
[cosd](#) - Cosine of argument in degrees.
[cosh](#) - Hyperbolic cosine.
[acos](#) - Inverse cosine.
[acosd](#) - Inverse cosine, result in degrees.
[acosh](#) - Inverse hyperbolic cosine.
[tan](#) - Tangent.
[tand](#) - Tangent of argument in degrees.
[tanh](#) - Hyperbolic tangent.
[atan](#) - Inverse tangent.
[atand](#) - Inverse tangent, result in degrees.
[atan2](#) - Four quadrant inverse tangent.
[atan2d](#) - Four quadrant inverse tangent, result in degrees.
[atanh](#) - Inverse hyperbolic tangent.

[sec](#) - Secant.
[secd](#) - Secant of argument in degrees.
[sech](#) - Hyperbolic secant.
[asec](#) - Inverse secant.
[asecd](#) - Inverse secant, result in degrees.
[asech](#) - Inverse hyperbolic secant.
[csc](#) - Cosecant.
[cscd](#) - Cosecant of argument in degrees.
[csch](#) - Hyperbolic cosecant.
[acsc](#) - Inverse cosecant.
[acscd](#) - Inverse cosecant, result in degrees.
[acsch](#) - Inverse hyperbolic cosecant.
[cot](#) - Cotangent.
[cotd](#) - Cotangent of argument in degrees.
[coth](#) - Hyperbolic cotangent.
[acot](#) - Inverse cotangent.
[acotd](#) - Inverse cotangent, result in degrees.
[hypot](#) - Inverse hyperbolic cotangent.
[hypot](#) - Square root of sum of squares.
[deg2rad](#) - Convert angles from degrees to radians.
[rad2deg](#) - Convert angles from radians to degrees.

常用的有

- 弧度制三角函数 `sin`, `cos`, `tan`, `asin`, `acos`, `atan`
- 角度三角函数 `sind`, `cosd`, `tand`, `asind`, `acosd`, `atand`

MATLAB 中的基本函数

指数类, 复数类, 四舍五入类

Exponential.

- [exp](#) - Exponential.
- [expm1](#) - Compute $\exp(x)-1$ accurately.
- [log](#) - Natural logarithm.
- [log1p](#) - Compute $\log(1+x)$ accurately.
- [log10](#) - Common (base 10) logarithm.
- [log2](#) - Base 2 logarithm and dissect floating point number.
- [pow2](#) - Base 2 power and scale floating point number.
- [realpow](#) - Power that will error out on complex result.
- [reallog](#) - Natural logarithm of real number.
- [realsqrt](#) - Square root of number greater than or equal to zero.
- [sqrt](#) - Square root.
- [nthroot](#) - Real n -th root of real numbers.
- [nextpow2](#) - Next higher power of 2.

Complex.

- [abs](#) - Absolute value.
- [angle](#) - Phase angle.
- [complex](#) - Construct complex data from real and imaginary parts.
- [conj](#) - Complex conjugate.
- [imag](#) - Complex imaginary part.
- [real](#) - Complex real part.
- [unwrap](#) - Unwrap phase angle.
- [isreal](#) - True for real array.
- [cplxpair](#) - Sort numbers into complex conjugate pairs.

Rounding and remainder.

- [fix](#) - Round towards zero.
- [floor](#) - Round towards minus infinity.
- [ceil](#) - Round towards plus infinity.
- [round](#) - Round towards nearest integer.
- [mod](#) - Modulus (signed remainder after division).
- [rem](#) - Remainder after division.
- [sign](#) - Signum.

常用的有

- e 指数函数 `exp`
- 自然对数函数 `log`
- 平方根函数 `sqrt`
- 绝对值函数 `abs`
- 四舍五入函数 `round`
- 余数函数 `mod`

关于复数的运算/函数, 后面单独讲

利用变量实现程序语言中最重要的操作—迭代

计算黄金分割比 0.618

黄金分割比 $\phi = 0.618$ 与方程 $x^2 - x - 1 = 0$ 的正根密切相关. 解非线性方程的一个重要算法就是不动点 (fixed point) 迭代. 此方程的迭代公式为: $x = \sqrt{x + 1}$.

在 MATLAB 命令窗口中, 输入 $x = 3;$, 然后输入 $x = \text{sqrt}(x+1)$, 然后通过 \uparrow 键不断重复运行上述命令, 观察数值变化.

说明: 起初输入的 $x = 3$ 相当于一个猜想值, 当下情况下, 这个猜想值可以是任意的, 不会改变最后的结果, 只是会改变迭代次数. 不断的通过 \uparrow 键运行命令, 相当于循环操作.

MATLAB 关于关系操作与逻辑操作的约定

关系操作

指令	含义	指令	含义
<	小于	>=	大于等于
<=	小于等于	==	等于
>	大于	~=	不等于

逻辑操作

指令	&		~	xor
含义	与	或	非	与非

关系操作和逻辑操作的返回值都是逻辑值, 即真和假. 1 表示真, 0 表示假.

逻辑数值可以与普通数值一样进行加减乘除等的数学运算.

关系运算或者逻辑运算的事例

判断大小

```
a=1;  
b=1.01;  
aeqb=a==b;
```

赋值运算优先级最低, 但为增加代码的可读性, 最好加上括号, 即变成 `aeqb=(a==b);`

注意 MATLAB 中的数值比较有时会因为浮点数值精度出现错误. 例如 `a=1; b = 1+eps/2;` 严格意义上来说, `a` 不等于 `b`, 但是 MATLAB 给出的结果是 `a` 与 `b` 相等.

MATLAB 中只有 0 是假, 其他的数值都是真, 无论正负, 无论是否为整数. 指令: `-1.5|0` 返回真.

目前, 我们已经学会在命令窗口中计算数学表达式, 并将数值赋给变量及 MATLAB 内置的基本函数.

当我们关闭 MATLAB 之后, 这些变量就会消失, 虽然在历史窗口中可以找到相关命令, 再重新运行, 然而当时间一长, 运行命令多了 (尤其是还伴随错误命令) 之后, 显得极其不方便.

再者, 实际问题往往都是复杂的工程, 需要很多句 M 语言进行解决.

于是, 便有了

M 文件

M 文件

多数实际问题, 很难用一条简单的指令完成; 多数都是用一系列复杂的指令结构完成. MATLAB 的 M 文件就是要解决这种困难—将指令集合起来.

M 文件有两种类型:

- ① 脚本文件 (script): 一般赋值语句或指令的集合.
- ② 函数 M 文件 (function M-files): 有固定格式, 多数情况下有输入和输出参数 ...函数定义名应和文件保存名一致.

启动方式: WIN

- 鼠标点击
- 指令 `edit`
- 快捷键 `Ctrl+N`

运行方式: WIN

- 鼠标点击
- 将 M 文件拖入命令窗口
- 命令窗口中输入脚本名字或函数名字

一. 脚本文件: 一般赋值语句或指令的集合. 脚本文件中产生的变量值会出现在 workspace(工作区) 内.

需要完成以下工作: 数值 9.8 的平方, 平方根, 正弦, 反正切, 指数, 自然对数, 向零取整, 四舍五入操作.

一. 脚本文件: 一般赋值语句或指令的集合. 脚本文件中产生的变量值会出现在 workspace(工作区) 内.

需要完成以下工作: 数值 9.8 的平方, 平方根, 正弦, 反正切, 指数, 自然对数, 向零取整, 四舍五入操作.

```
x = 9.8; % 首先定义变量x, 并赋值2.56
x2 = x^2;
xsqrt = sqrt(x);
xsin = sin(x);
xatan = atan(x);
xexp = exp(x);
xln = log(x); % 注意MATLAB中的自然对数函数不是ln而是log
x20 = fix(x);
x45 = round(x);
```

若后期想对 3.14 进行计算, 直接将 $x=9.8$ 改为 $x=3.14$ 即可. 这体现了使用变量而不是直接对数值 9.8 进行操作的优势.

二. 函数文件: 有固定格式, 多数情况下有输入和输出参数 ...函数定义名应和文件保存名一致.

函数文件格式

```
function [output_args] = funName(input_args)
%Detailed explanation goes here
end
```

关键词`function`指明此 M 文件是函数脚本.

关键词`end`表示函数脚本结束, 可以没有, 但是最好有.

输出参数用方括号, 当只有一个输出参数时, 方括号可以省略...

输入参数用小括号, 当不需要输入参数时, 括号内写入 ~ 即可.

若要运行此函数, 只需要在命令行输入 `funName(ShuRu)` 即可, 也可以 `ShuChu = funName(CanShu)`

NOTE: 函数脚本可以没有输入和输出. 但是, 如果那样, 为何还要写成函数文件, 而不是直接写成普通脚本文件.

M 文件的一般结构 (了解)

脚本文件只比函数文件少了一个“函数声明行”(即 `function` 关键词), 二者的语法及构架等均相同。M 文件结构如下:

函数声明行: 位于函数文件首行, 以关键字 `function` 开头, 定义函数名及函数的输入/输出变量。脚本文件无须函数声明行。

H1 行: 紧随函数声明行之后的以 “%” 开头的第一注释行 (可以没有, 建议有)。H1 行包括大写的函数文件名和运用关键词简要描述的函数功能。该行将提供 `lookfor` 命令作为关键词查询和 `help` 帮助文档。

函数主体: 这部分内容包括所有实现该 M 函数文件功能的 MATLAB 指令、接收输入变量、进行程序流控制。

良好的编写习惯:

- 带有 H1 行...
- 含有编写和修改记录: 与 H1 行以一个“空”行相隔; 该行以 “%” 开头, 记录了编写及修改 M 文件的所有的作者、日期及版本号, 以方便后来的使用者查询、修改和使用。
- 函数体与编写修改记录空一行

一个规范函数 M 文件的例子

```
function stuNum = yourClassGrade(a, b)
% YOURCLASSGRADE return the number of students whose grades range from a to b

% a is the lower grade
% b is the high grade
% stuNum is the student number

% finished 2020.03.09, I'm: 98TellU

if nargin>2
error('NOTE only two inputs');
end
bla;
bla;
end
```

函数文件事例

输入一个参数 x , 然后输出它的 2 倍.

```
function y = twoTimes(x)
y = 2*x;
end
```

给定半径 r , 输出此半径对应的球的体积和面积.

```
function [V, A] = calVolumeArea(r)
V = 3/4*pi*r^3;
A = pi*r^2
end
```

给定两个数 a 和 b , 输出二者之和.

```
function c = twoNumPlus(a,b)
c = a+b;
end
```

MATLAB 中的控制流

MATLAB 提供了多种控制流结构, 主要有: 条件和循环以及中断或暂停.

一. if 条件语句

`if-elseif-else-end` 其中 `elseif` 和 `else` 均是可选的, M 语言格式

```
if expression
statements
elseif expression
statements
else
statements
end
```

二. for 循环语句

`for-end` M 语句格式

```
for index = values
statements
end
```

三. while 循环语句

`while-end` M 语句格式

```
while expression
statements
end
```

四. 中断

MATLAB 循环体内可以用 `continue` 和 `break` 进行中断循环体, 常与 `if` 条件连用.

- `continue`: 跳过位于他之后的语句而进行下一个循环.
- `break`: 跳出. 结束整个循环, 进行循环之后的语句.

五. 暂停

- `pause` 使程序暂停执行, 等待用户按任意键继续执行.
- `pause(n)` 可以使程序暂停 `n` 秒后, 再继续执行语句.
- `return` 结束 `return` 所在函数的执行.

MATLAB 常用指令及其作用

- `clc`: 清除命令窗口
- `clf`: 清除图形窗口
- `clear`: 清除工作空间所有变量
- `type`: 查看指定 M 文件内容, 在命令窗口中显示
- `edit`: 查看指定 M 文件内容, 打开 M 文件编辑器查看
- `quit`/`exit`: 退出 MATLAB (同上)
- 上下键: 调用之前使用的历史命令.
- `tab` 键: 自动补全命令

保存

- 鼠标方式
 - 保存全部数据: 点击菜单栏或者工作空间上的图标 (Save Workspace); 或者选中工作空间中的全部变量, 右键点击 Save as
 - 保存个别数据: 选中, 右键...
- 指令保存: save 可以通过 help/doc 查看指令格式为

```
fileName = 'DateFileName.mat';  
save(fileName) % Store all variables  
save(fileName, v1, v2) %Store v1 and v2  
save(fileName, v1, v2, -append) % Add v1 and v2 into existig file
```

NOTE: MATLAB 可以保存很多文件类型, 不限于.mat 类型.

读取

- 鼠标方式
 - 双击存取文件
 - 拖入文件至命令窗口
 - 点击菜单栏中的 Import Data 工具.
- 指令读取: load 可以通过 help/doc 查看指令格式为

```
load('gong.mat')           % load all variables  
load('handel.mat', 'y')    % load only variable y
```

NOTE: MATLAB 可以读入很多文件类型 (如: .xlsx), 不仅限于.mat 类型.

p 码文件

M 文件首次被调用（包括在 M 文件编辑器中被打开或者在命令行窗口运行文件名）时，MATLAB 首先对该 M 文件进行语法分析，并把生成的相应内部伪代码（Pseudocode，P 码）文件存放在内存中。

M 文件再次被调用时，将直接调用该 M 文件在内存中的 P 码文件，而不会再对原 M 文件进行重复的语法分析。需要注意的是，MATLAB 的分析器（Parser）总是把 M 文件连同在该 M 文件中被调用的所有函数文件一起转变成 P 码文件。

P 码文件与原码文件具有相同的文件名，但是其扩展名为 “.p”；P 码的运行速度高于原码。在 MATLAB 环境中，假如存在同名的 P 码文件和原码文件，那么当该文件名被调用时，被执行的一定是 P 码文件。

P 码文件并不是仅当 M 文件被调用时才能生成的，用户也可以使用 MATLAB 中的内置指令在 M 文件中生成 P 码文件。

p 码文件的生成与优点

`pcode FunName`: 该指令的功能为在当前文件夹上生成 `FunName.p`。`pcode FunName-inplace`: 该指令的功能为在 `Filename.m` 所在的目录上生成 `FunName.p`。

p 码文件相对于原码文件来说有以下两个优点。

- ① 运行速度快。
- ② p 码文件中的数据为二进制保存，阅读困难，增大了程序的保密性。

MATLAB 帮助文件

MATLAB 有强大/完备/友好的帮助文件/文档.

- 启动 MATLAB 帮助浏览器—电子书 (英文)
 - ① 指令法: doc (以前版本: helpdesk 或 helpbrower, 2006 版后不建议使用)
 - ② 菜单栏启动: help—document (版本不同稍有出入)
- 函数搜索指令
 - ① help fun 直接在命令窗口中显示, 需要知道函数的具体名字, 如 help sin, 注意 see also.
 - ② doc fun 直接打开帮助浏览器 fun 页面, 内容详细, 如 doc sin. 页面最下端也有 see also, 而且有版本信息.
 - ③ lookfor keyword 不知道具体函数时, 可以搜索关键字. 在命令窗口中显示. 是对 M 文件的第一行进行逐词搜索. 如 lookfor exponential
 - ④ docsearch expression 可以进行多词条并且可以使用通配符 * 如
 - docsearch plot 存在 plot 的帮助文档
 - docsearch plot tools 存在 plot 和 tools 的文档
 - docsearch "plot tools" 上一个的子集, 存在 plot tools 的文档
 - docsearch plot OR tools
 - docsearch plot* 存在 plot 前缀词的帮助文档

额外: MATLAB 关于时间

clock: 返回 6 元素变量, 年月日时分秒

date: 返回字符向量: day-month-year.

now: 返回天数

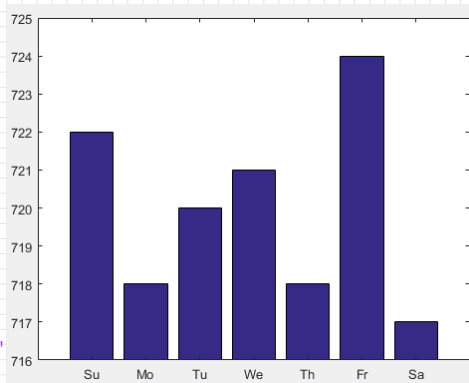
datenum(c): 返回天数 (精确到小数), c 可以是 clock 和 date 的返回值或者其他定义时间的数值, 如 2018,8,8

[DayNumber,DayName] = weekday(D): 返回一周的第几天 (DayNumber) 以及周几 (DayName), D 可以是天数, 例如 datenum 的返回值.

MATLAB 时间函数的应用: 十三号星期五

十三号星期五: 每个月的 13 号落在星期五.

```
c = zeros(1,7);% 产生7个0数组, 后面学
for y = 1601:2020 % 1601年至2020年
for m = 1:12 % 月份
d = datenum([y,m,13]);% 相应年月13号对应的天数
w = weekday(d);% 星期数 (1代表星期天, 以此类推)
c(w) = c(w)+1;% 累积, 求和
end
end
bar(c)% 画图
axis([0,8,716,725])% 坐标区间以放大差异
set(gca, 'xticklabel', {'Su', 'Mo', 'Tu', 'We', 'Th', 'Fr', 'Sa'})
```



十三号星期五还是完全由科学依据的哈 (划重点: 这里手动狗头静止滑稽脸)

使用变量的优点

- 避免计算重复使用的数值;
- 随时概念不同变量对应的值;
- 定义中间变量, 使代码简洁可读, 尤其是长公式⁶;
- 变量自身经过一系列运算之后, 可以再赋值给自身—迭代
例如

```
x = 3
```

```
x = sin(x)*log(x)-exp(sqrt(x))
```

⁶MATLAB 中长公式的续行用三个点 ...

错误提示

MATLAB 有很友好的错误提示机制. 在以后的学习中会不经意遇到, 是一个积累过程, 无需在意. 例如, 在命令窗口中输入

```
x = x + abc;
```

MATLAB 返回

Undefined **function** or variable 'abc'. (注意: MATLAB 中红色字体)

注释

注释代码可以节省后期维护成本⁷, 是一个良好的习惯.

注释内容一般包含

- 数值单位
- 公式目的
- 变量含义

不要相信自己今天知道的, 明天还能记住. 想法和灵感都是瞬时的..... 记忆也是如此

⁷人力财力物力, 是一个长期工程

MATLAB 的显示方式

- `format short` 默认显示方式 (显示的最后一位进行了四舍五入)
- `format long`
- `format short e`
- `format long e`
- `format rat` 用最接近结果的分数表示

二进制浮点运算

默认情况下, MATLAB 使用 double (双精度) 来表示一个浮点数 x , 即 64 位来表示 x . 一个非零的浮点数可以表示位

$$x = \pm(1 + f)2^e$$

- f 表示尾数 ($0 \leq f < 1$); 用 52 位表示
- e 是指数, 用 11 位表示, $-1022 \leq e \leq 1023$,
- 最后下一位表示 x 的符号.

f 决定了数的精度, 精度为 $2^{-52} = 2.2204 \times 10^{-16}$, 即预定义的变量 `eps`

e 决定了数值范围, 没有取 1024 和 -1023, 是因为 MATLAB 为这两个数预留了. f 取 0 且 e 取 1024 表示预定义变量 `inf` (无穷大), 非规范化数字 (Denormal numbers) 由 e 取 -1023 来表示 (什么是非规范化数字, 我也不是很懂)

以上资料参考 MATLAB 高效编程技巧与应用

Za7Za8

<https://www.cdslab.org/matlab/index.html> 网站提供了一些 MATLAB 的基础!

doc Function Summary: show those functions for working with numeric classes:

致谢

本课程参考资料包括但不限于 (排名不分先后):

- ① MATLAB 帮助文档
- ② Experiments with MATLAB, C.Moler
- ③ MATLAB 高效编程技巧与应用: 25 个案例分析, 吴鹏 (rocwoods)
- ④ 精通 MATLAB R2011a, 张志勇
- ⑤ MATLAB 2014a 完全自学一本通, 刘浩/韩晶
- ⑥ Elementary Mathematical and Computational Tools for Electrical and Computer Engineers using MATLAB, Jamal T.Manassah
- ⑦ Physical Modeling in MATLAB, Allen B.Downey
- ⑧ 一些网络资源 (知乎, MATLAB 中文论坛等)
- ⑨ ...