# **MATLAB入门**

## 课前学习任务

（1）安装MATLAB

下载MATLAB并完成安装。随着MATLAB的使用普及，正版软件的价格已经不高了，核心程序售价在15000元左右。对高校用户，MATLAB推出了集体授权的方式，即由高校官方购买，高校在通过某种方式授权给校内人员。我校（华中科技大学）通过正版软件平台提供软件下载链接，校内人员安装后通过正版软件正版平台进行激活即可。

**有VPN的同学可从学校网址**[**http://software.hust.edu.cn**](http://software.hust.edu.cn)**下载MATLAB 2017a的版本，网站上有详细的安装和激活说明；没有VPN的同学可从MATLAB中文官网<https://ww2.mathworks.cn/>下载MATLAB 2019b的版本，安装和激活方法已上传“2020电气工程建模与仿真”（874801138）群文件。**如果空间足够，可安装全部模块；若空间不足，可先安装基本模块和simscape相关模块，后续使用过程中若有新的需求在安装相应模块即可。

（2）MATLAB入门之旅

完成“MATLAB入门之旅”的学习，地址见官网

<https://ww2.mathworks.cn/learn/tutorials/matlab-onramp.html?s_cid=learn_ONRAMP_BAN>

完成任务（1）和（2）的学习，将学习结果以报告形式交给助教，报告篇幅不限，以证明自己完成学习为标准，可将学习过程中你认为很有趣或很关键的环节录制成视频，随报告一并上交。

## MATLAB简介

MATLAB从字面理解是矩阵实验室的简称，它将数值计算、数据可视化、程序设计和系统仿真融合到一个开放的交互式工作环境中，是目前最好的科学计算和基于模型设计的软件之一，已成为众多技术公司、政府实验室、金融机构以及3500多所大学进行工程项目和科学研究的基本工具。MATLAB应用领域非常广泛，包括科学计算、嵌入式系统、控制系统、数字信号处理、通信系统、图像处理和计算机视觉、FPGA设计、机电、测试和测量、计算生物学以及计算金融学等。

**MATLAB在中国也有广阔的市场，用户非常多，其中有很多是深度用户。这里给大家推荐一个网址**[**http://www.ilovematlab.cn**](http://www.ilovematlab.cn/)，这个是MATLAB的一个中文论坛，资源非常丰富。大家使用MATLAB过程中遇到解决不了的问题，可以上论坛看看，一般都能找到答案或者得到一定的启发。

MATLAB总部位于美国马萨诸塞州的Natick市，是世界领先的技术计算和基于模型的设计的软件开发商和供应商，拥有3000多名员工，2007年成立北京分公司，中国官网是<https://ww2.mathworks.cn/>。现在MATLAB每年一般发布a和b 2个版本，新的版本一般会在旧版本基础上做一些更新，目前最新的版本是2019b。

**MATLAB软件架构如图1所示，主要分为MATLAB和SIMULINK两大块，其中MATLAB部分主要基于技术计算语言，SIMULINK部分主要基于仿真和模型设计。**对于物理建模班同学，我们两个部分都会做一定介绍，对于数学建模班同学，我们主要介绍MATLAB。MATLAB入门这次课主要介绍MATALB软件中基于技术计算语言MATLAB部分。

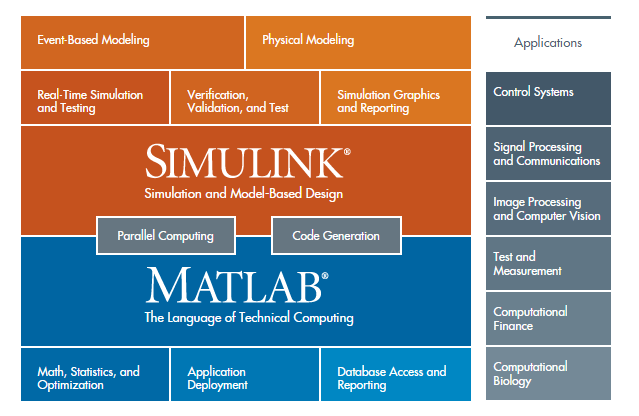


图1 MATLAB软件架构

基于技术计算语言MATLAB的主要优点如下：

1. 解释性语言，简洁紧凑，能符号计算等；
2. 库函数丰富，编程效率高；功能强劲的工具箱。由该领域里的学术专家编写，方便调用；（我们常见的解决问题需要的函数MATLAB里面基本都有，大部分情况下我们通过调用库函数就可以解决问题。同时，由于这些函数都是学术专家编写的，比较严谨，也非常符合大家的使用习惯，使用非常方便）
3. 可视化强，具有较强的编辑图形界面；
4. 开放性源程序，可扩充性强，兼容性强。

当然由于MATLAB是解释性语言，执行效率不是太高，这是MATLAB的主要缺点，但和优点比起来，这个缺点也能接受，除特殊场合外一般对用户体验影响不大。

下面以2017a版本为例对MATLAB软件界面做一个简要介绍。MATLAB安装好以后，点击MATLAB图标，会出来图2所示的界面。MATLAB界面比较简单。**关于MATLAB界面的介绍，我们的课件视频\*\*\*~\*\*\*分钟有详细介绍，大家可以在观看视频的同时打开MATLAB界面进行熟悉。**同时下面的链接里面也介绍的较为详细，同学们可以自行阅读。（链接地址：<https://blog.csdn.net/weixin_44566643/article/details/94871461>）。

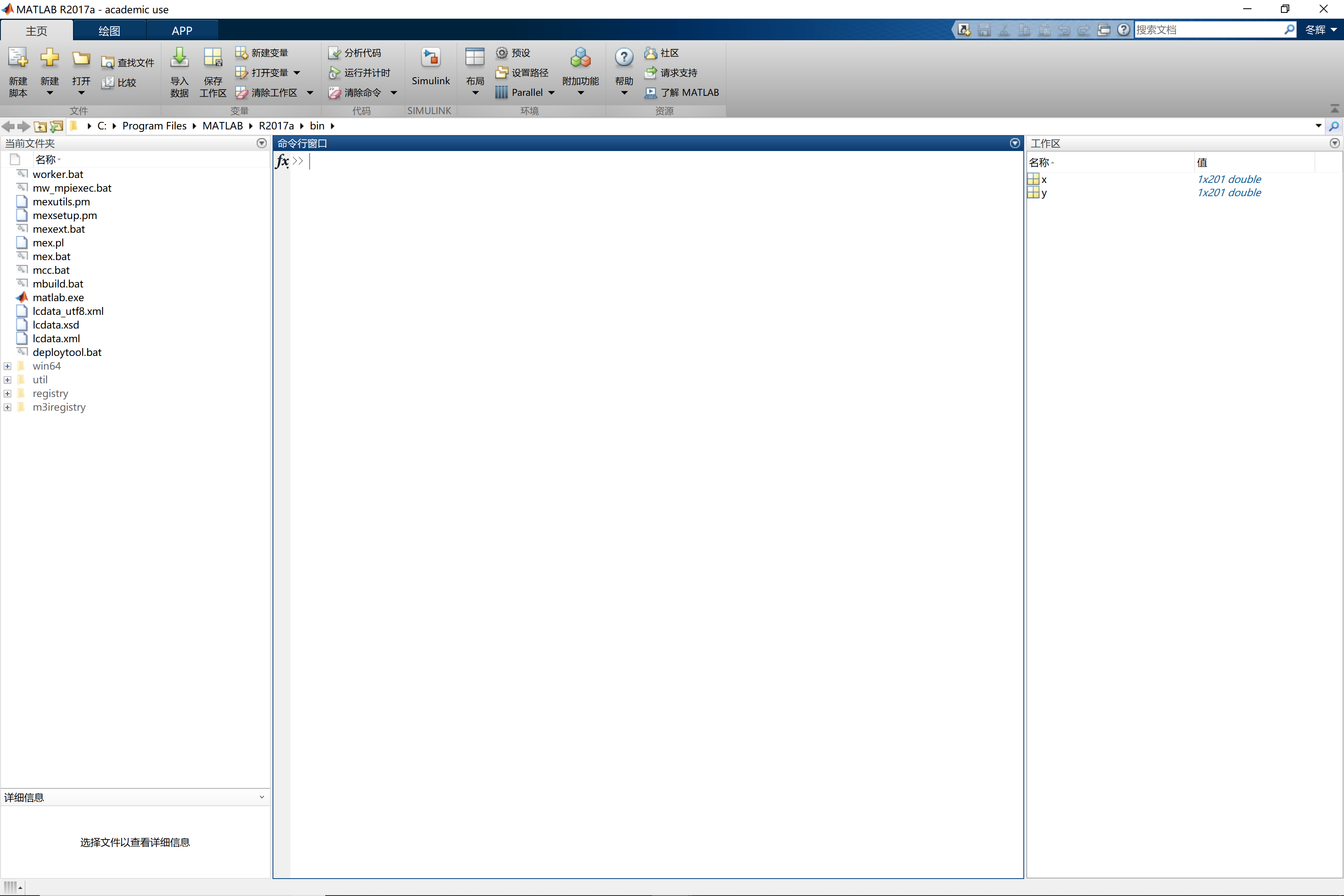


图2 MATLAB 2017a界面

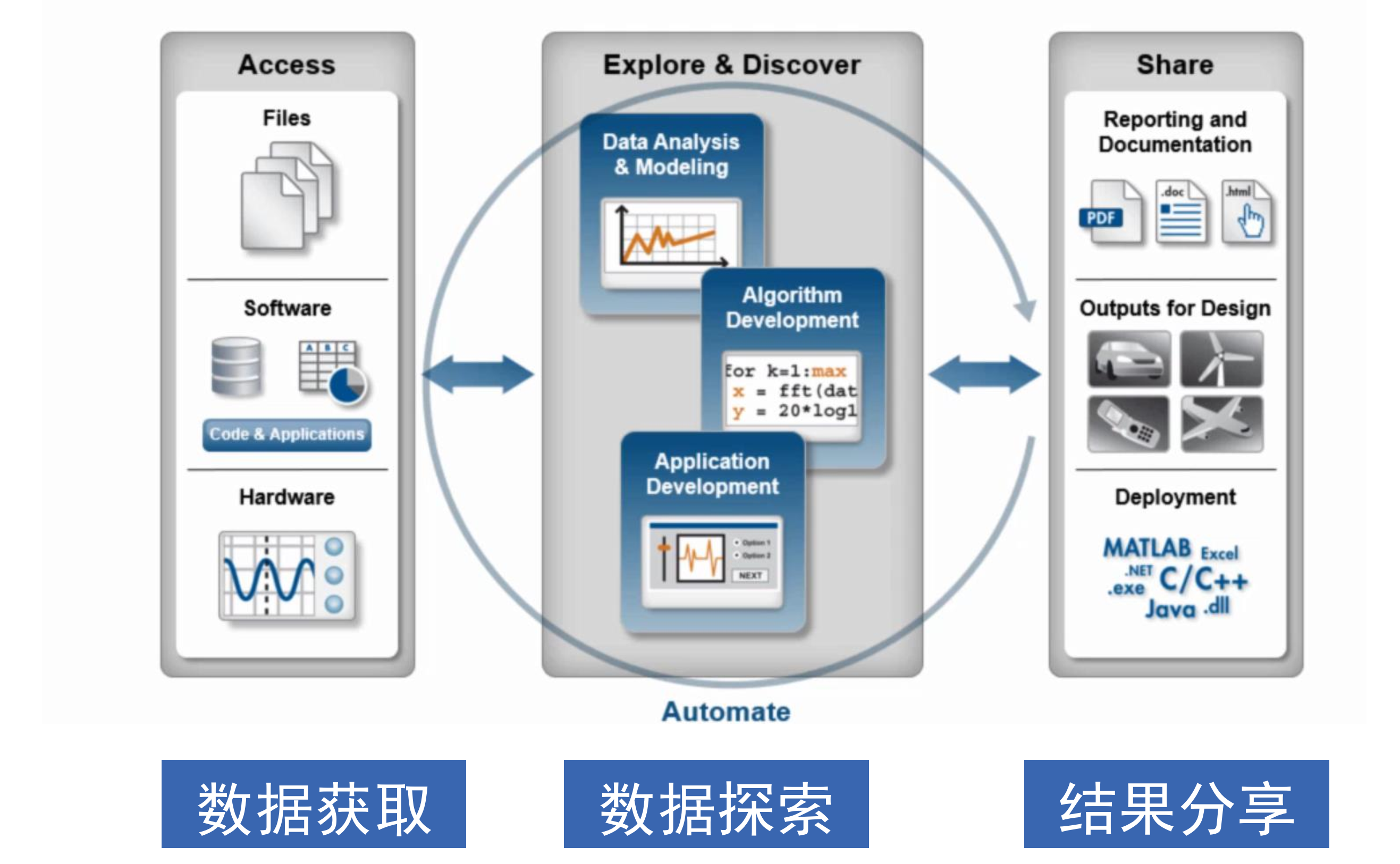


图3 典型的科学计算流程

至此我们对MATLAB有了一个初步的认识。在正式使用MATLAB之前，我们还需要对典型的科学计算流程有个大致了解。典型的计算流程一般分为数据获取、数据探索和结果分享三个部分，其中数据一般从文档、软件或硬件等方式获取，得到数据以后通过对数据的处理、开发和分析等完成数据探索，最后通过报告或文档等形式呈现，或者指导设计，亦或部署到其他平台等多种方式实现结果的分享。使用MATLAB进行科学计算基本流程和图3一样，需要记住的是MATLAB只是一个工具，要用好工具的前提是结合工具特点把思路整理清楚。有了清晰的思路，问题的解决将是水到渠成。

## 1.2 Help的使用

MATLAB的help功能十分强大，包含所有函数命令的用法说明，并附有大量应用实例，非常详细。MATLAB的help调用方法如下：

**在命令行窗口输入 help \*\*（某个函数名）；软件返回该函数的简短说明，后接“doc \*\*”或者“\*\*参考页”链接，点击该链接就可进入函数的详细说明页面。**MATLAB的help还有一个很实用的功能，搜索结果提供了很多关联函数链接，用于提醒使用人员选择更加合适的函数。

下面我们以sum函数做简要介绍。在命名行窗口输入指令help sum，2017a版搜索结果如图4所示（不同版本结果可能略有不同）。从图中可以看出，sum函数的简单说明对函数的使用方法和函数中每个参量的介绍都给出了介绍，同时还有应用实例。从说明可也看出，sum可也对向量（单行矩阵）或矩阵进行求和，通过不同的设置可定义不同的求和方式和不同的结果输出。

如果我们点击“sum参考页”链接，会给出sum函数的详细使用方法和更多的应用实例。刚开始学习一个函数的时候，**建议大家通过函数参考页进行学习，效率较高，可以直接在命令行输入doc \*\*（某个函数名）调用函数参考页。**此外，图4还可以看出，当我们搜索sum函数时，给出了sum的很多关联函数，要了解这些函数的定义，我们可也随便点击一个函数进行学习即可，这里就不在赘述。

当我们知道函数名的时候可也用help搜索函数进行学习，当我们不知道函数名称的时候，只知道大概功能时，可以在lookfor \*\*（某个功能名称，一般只能是英文）进行查找，当然有时候可能搜索出来的参考函数非常多，不利于筛选。建议如果范围比较大的话可以先尝试上网搜索下，适当缩小范围后在MATLAB里面进行搜索。

关于help功能的详细使用方法和一些基本设置我们的课件视频\*\*\*~\*\*\*分钟有详细介绍，大家可以在观看视频的同时打开MATLAB进行help使用方法的练习。

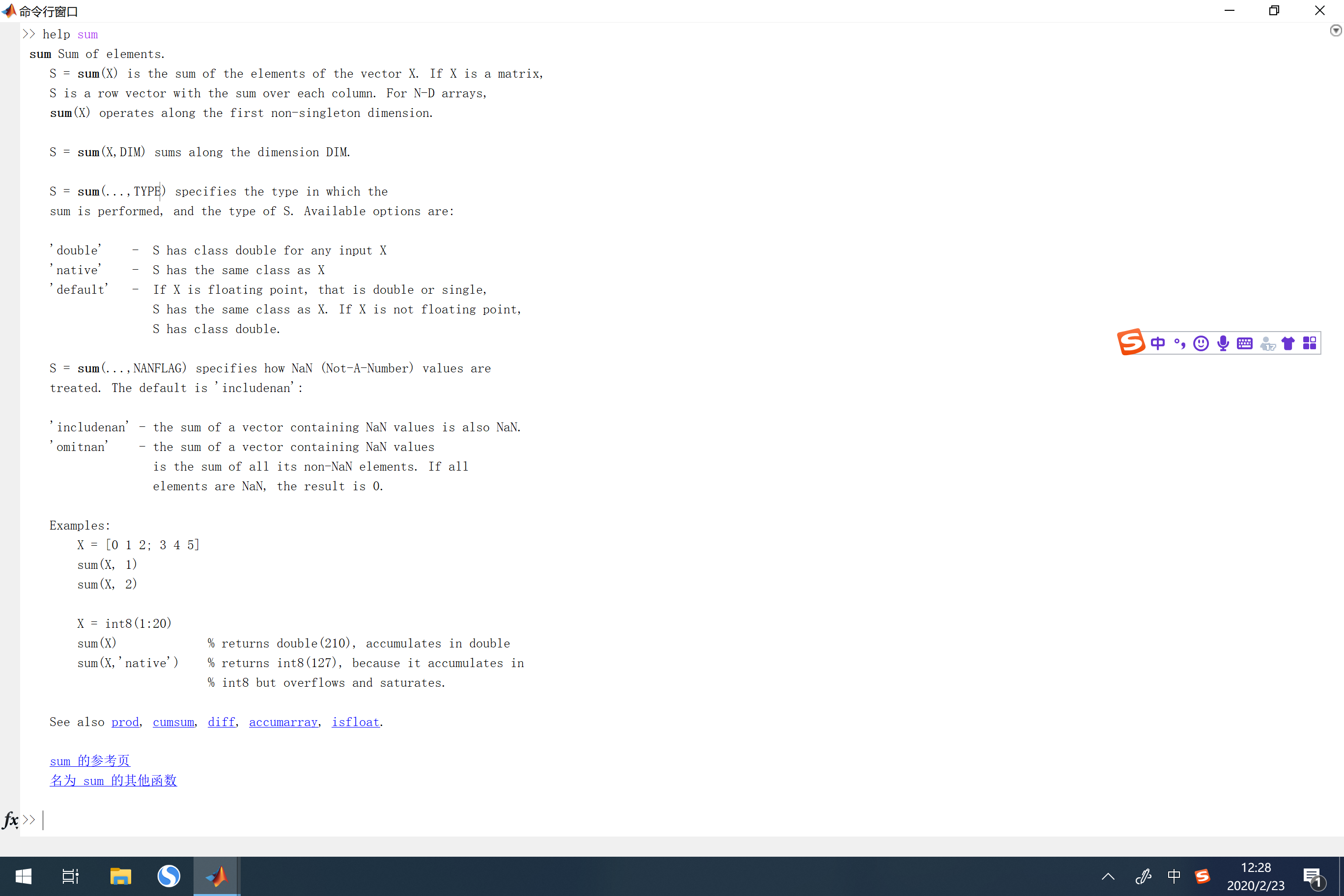


图4 help sum搜索结果

## 1.3数据的导入

MATLAB主持的数据类型非常丰富，各类数据导入可通过主界面：主页-导入数据-选择路径导入，也可通过相关指令导入。

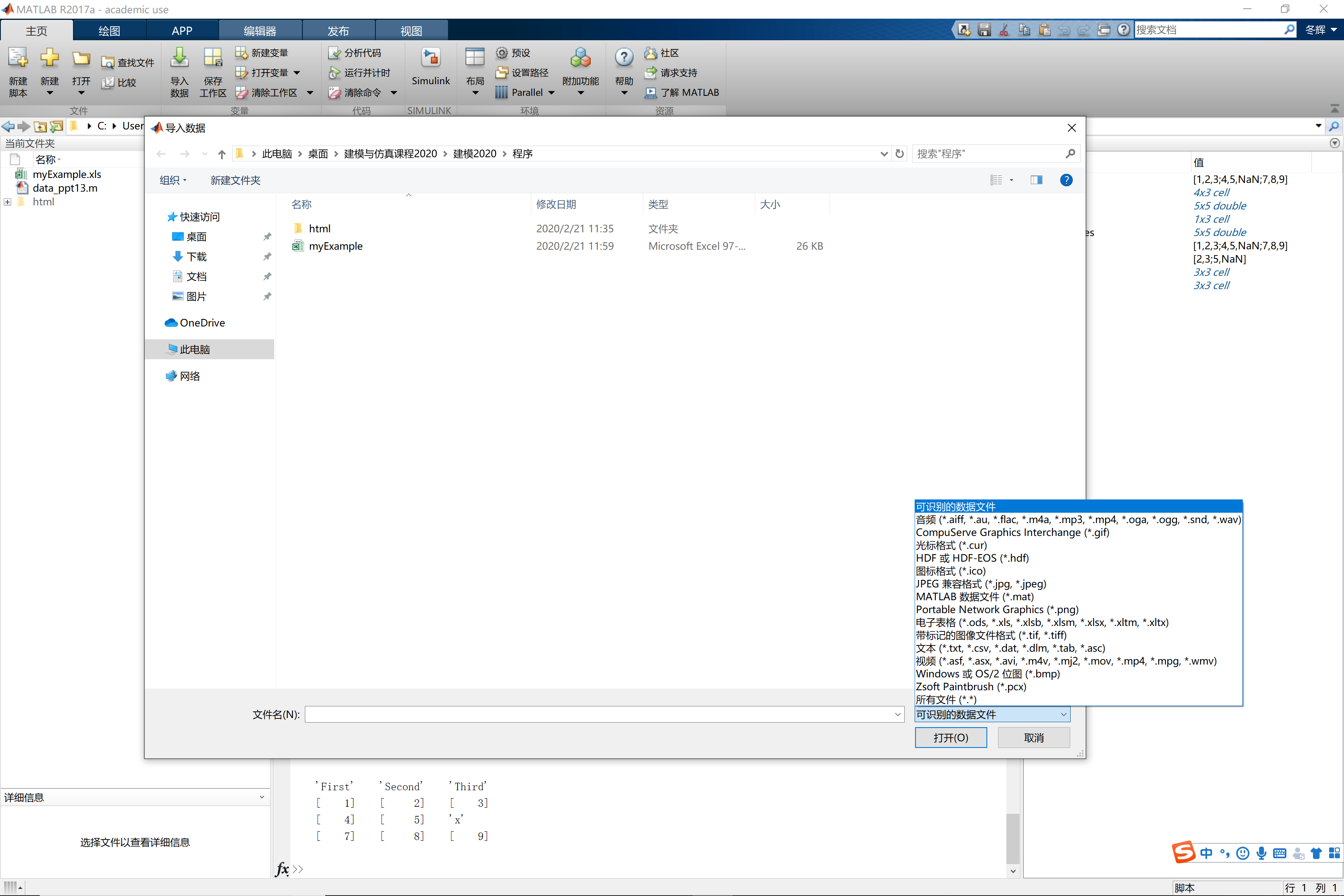


图5 MATLAB 2017a支持的数据格式

### 1.3.1 从Excel文件读取数据

2017a从excel读取数据的函数为**xlsread，对应的写入函数xlswrite。关于xlsread的用法我们依托下面这个例子做简要介绍。**

% Create data for use in the examples that follow:

values = {1, 2, 3 ; 4, 5, 'x' ; 7, 8, 9};

headers = {'First', 'Second', 'Third'};

xlswrite('myExample.xls', [headers; values]);

moreValues = rand(5);

xlswrite('myExample.xls', moreValues, 'MySheet');

% Read data from the first worksheet into a numeric array:

A = xlsread('myExample.xls')

% Read a specific range of data:

subsetA = xlsread('myExample.xls', 1, 'B2:C3')

% Read from a named worksheet:

B = xlsread('myExample.xls', 'MySheet')

% Request the numeric data, text, and a copy of the unprocessed (raw)

% data from the first worksheet:

[ndata, text, alldata] = xlsread('myExample.xls')

上面的例子中，第一段主要定义数组和写入表格，运行的结果将2个数组写入myExample.xls这个表格的第一个sheet和第二个sheet，同时第二个sheet命名为‘MySheet’。第二段主要从myExamples.xls中读取第一个sheet的数值并赋给数组A。第三段主要是读取第一个sheet中指定区域（第2行第2列到第3行第三列）数值并赋给subsetA。第四段主要是读取第二个sheet（MySheet）中数值并赋给B，最后一段读取第一个sheet的数据，数值赋给ndata，文本赋给text，所有的数据赋给alldata。大家可以运行上述代码体学习xlsread函数的用法。

备注：上述代码运行过程过程中可能会提示“调用错误，调度异常: 参数错误”，解决方法可以参考链接<https://www.ilovematlab.cn/thread-563197-1-1.html>

### 1.3.2 从TXT文件读取数据

2017a从txt读取数值的函数为load（ASCII 码文件 ）或textscan（不同类型的文本或数值），**具体用法我们依托下面两个例子做简要介绍。**

**示例1：**

%generates a row vector of 8 linearly equally spaced points between 1 and 30

a = linspace(1,30,8);

% Save the variable a to the file exper.txt in -ascii format.

save exper.txt a -ascii;

% Load data from exper.txt into workspace.

b = load('exper.txt')

这个例子比较简单，就是生成一个1\*8的数组以ASCII码格式写入exper.txt文件，再通过load函数读取文件。大家可自行运行下代码学习load函数的用法。

示例2：

Suppose the text file 'mydata.txt' contains the following:

Sally Level1 12.34 45 1.23e10 inf Nan Yes 5.1+3i

Joe Level2 23.54 60 9e19 -inf 0.001 No 2.2-.5i

Bill Level3 34.90 12 2e5 10 100 No 3.1+.1i

Read the file:

fid = fopen('mydata.dat');

C = textscan(fid, '%s%s%f32%d8%u%f%f%s%f');

fclose(fid);

textscan returns a 1-by-9 cell array C with the following cells:

C{1} = {'Sally‘;'Joe’;'Bill'} %class cell

C{2} = {'Level1'; 'Level2'; 'Level3'} %class cell

C{3} = [12.34;23.54;34.9] %class single

C{4} = [45;60;12] %class int8

C{5} = [4294967295; 4294967295; 200000] %class uint32

C{6} = [Inf;-Inf;10] %class double

C{7} = [NaN;0.001;100] %class double

C{8} = {'Yes’;'No’;'No'} %class cell

C{9} = [5.1+3.0i; 2.2-0.5i; 3.1+0.1i] %class double

上述例子要求同学们在当前目录下先创建一个名为mydata.txt文件，并将示例中的数据复制到文件中。文件创建完成后，可运行上述代码，其中fopen是打开文件，textscan读取数据，并且定义不同的数据类型。读取完成后通过fclose关闭文件。同学们可自行运行代码学习下使用方法。需要注意的是由于文件中数据类型包含文本和数值，数值C为元胞数组（cell array），调用C中元素需要用大括号。关于MATLAB数据类型和不同类型的含义同学们可以参阅网页<https://ww2.mathworks.cn/help/matlab/data-types.html>进行学习。

### 1.3.3 从图像文件读取数据

2017a从图像文件读取数值的函数为imread，**具体用法我们依托下面的例子做简要介绍。**

clc, clear, close all

a1=imread('000.bmp');

[m,n]=size(a1);

%% 批量读取图片

dirname = 'ImageChips';

files = dir(fullfile(dirname, '\*.bmp'));

a=zeros(m,n,19);

pic=[];

for ii = 1:length(files)

filename = fullfile(dirname, files(ii).name);

a(:,:,ii)=imread(filename);

pic=[pic,a(:,:,ii)];

end

double(pic);

figure

imshow(pic,[])

上述示例中第一行主要是清空命令行窗口（clc），清空工作区数据（clear）；关闭所有的图形（close all）。第二行是读取000.bmp这个图像并复制给a1，第三行size函数用来读取a1数组的行数和列数参数。第二节是批量读取图片的程序，循环读取文件夹中的图片并赋给三维数组a，最后通过imshow读取的图片。同学们可自行运行代码学习imread使用方法。

两点温馨提示：

（1）运行程序所需的数据文件“000.bmp”和文件夹“ImageChips”已上传群文件（参见文件夹《readimages》），运行上述代码时需要把文件“000.bmp”和文件夹“ImageChips”复制到MATALAB当前目录下，或者把存储这两个文件（夹）的路径添加到MATLAB搜索路径中，可通过主页-设置路径添加。

（2）运行上述程序，会得到一幅按顺序拼接而成的文字，文字顺序显然不正确，出题的意图是通过建模实现拼接的正确率。但实现图像的读入并能将图像拼接是求解问题的基础。

### 1.3.4 从视频文件读取数据

2017a从图像文件读取数值的函数为**vision.VideoFileReader**，**具体用法我们依托下面的例子做简要介绍**

videoFReader = vision.VideoFileReader('vippedtracking.mp4');

% 播放视频文件

videoPlayer = vision.VideoPlayer;

while ~isDone(videoFReader)

videoFrame = step(videoFReader);

step(videoPlayer, videoFrame);

end

release(videoPlayer);

% 读取视频中的图像

videoFrame = step(videoFReader);

n = 0;

while n~=15

videoFrame = step(videoFReader);

n = n+1;

end

figure, imshow(videoFrame)

release(videoPlayer);

release(videoFReader)

上述例子主要是读取视频文件vippedtracking.mp4并播放视频文件，同时提取其中的一帧图像进行显示。同学们可自行运行代码学习视频读取函数的使用方法。运行程序所需的数据文件“vippedtracking.mp4”已上传群文件（文件夹《readvideo》），运行上述代码时需要把文件“vippedtracking.mp4”复制到MATALAB当前目录下，或者把存储这个文件的路径添加到MATLAB搜索路径中，可通过主页-设置路径添加。

## 1.4数据的可视化

MATLAB提供了丰富的绘图函数进行绘制，可绘制各类二维、三维图或图像等，能绘制的图形类别可参考MATLAB软件界面上“绘图”区。MATLAB的example中提供了很多绘图示例可供同学们学习，建议同学们通过示例进行绘图学习，示例的查找方法MATLAB界面上：主页—帮助—示例—Graphics。常见的绘图函数有plot（绘制二维曲线图）、mesh/surf（绘制三维曲线图，surf绘制图形更光滑）及contour和contourf（绘制等高图，其中contouf会在等高线之间填充颜色）等。

此外，和绘图相关的函数有figure（新建一个图形窗口）、hold on/off（保持当前图形或取消保持，保持状态下当前轴及图像保持而不被刷新，准备接受此后将绘制的图形，多图共存）、subplot（将多个子图放到一个Figure中）……由于相关函数非常多，在此不一一列举。同学们可以通过前面提到的help功能、matlab示例等方式进行学习。

在此，需要强调的是，图形可视化的主要目前还是交流。有的时候描述一个问题，可能说了半天也说不清楚，但如果附件一些图形可能立马就一目了然了。虽然如此，但不是简单的画图就完事了，一般需要根据不同的应用场合对图形进行美化，使图形满足不同使用要求如报告撰写、ppt演示等。**一定要记住，漂亮的图形绝对是加分项。**MATLAB图形可通过**属性编辑器或编程调节绘图属性**，通过编辑器的优点是采用交互界面处理，简单易学，非常方便，但当需要处理的图形较多时，这种方式就比较低效了。图形较多时，推荐采用编程调节绘图属性，可以通过matlab的help和example学习如何编程调节属性，这里我们也给了一个链接<https://blog.csdn.net/u013346007/article/details/56675741>，大家可以参考下。

关于绘图的属性调节我们的课件视频\*\*\*~\*\*\*分钟有简单介绍，大家可以在观看视频的同时打开MATLAB进行绘图属性调节的练习。 需要说明的每个同学的审美观点是不太一样的，没有绝对的美和不美，如何绘制美观大方的图片没有绝对的标准，但有几个观点可供大家参考：

1. 如果图形有特殊用途，比如学术论文等的插图，一般杂志都会对图形绘制有一定的要求，这个时候我们只需要根据要求绘制图片即可。记住底线是满足绘图要求，在此基础上可适当发挥。
2. 考虑下用图的场景。有些时候你的图片可能是ppt的一部分，这个时候如果图片上的文字比较小，可能比较远的观众就看不到，体验不是太好。需要对图形的使用场景做个设想
3. 图形美不美，大家说的算。前面说了图形没有绝对的美和不美，但一般大部分人觉得漂亮的图片应该是不错的。所以当自己拿捏不准的时候，可以把你画的图给别的同学看看，多听听他人的意见总是没错的。
4. 模仿。教材、论文和一些得奖的竞赛报告等图形一般都是精心处理过，所以当我们不知道如何把图画漂亮时，可以先从模仿做起，假以时日你一定会绘制出一幅漂亮的图片。

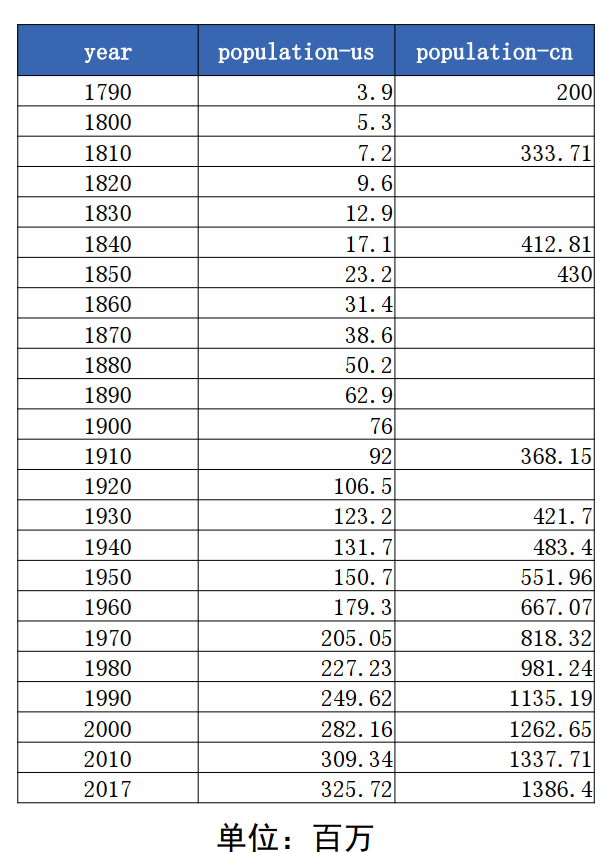
## 1.5 课堂练习

到这个地方，大家应该已经跃跃欲试了吧。这里我们有两个课堂练习题，同学们可以用前面学到的知识来完成下。

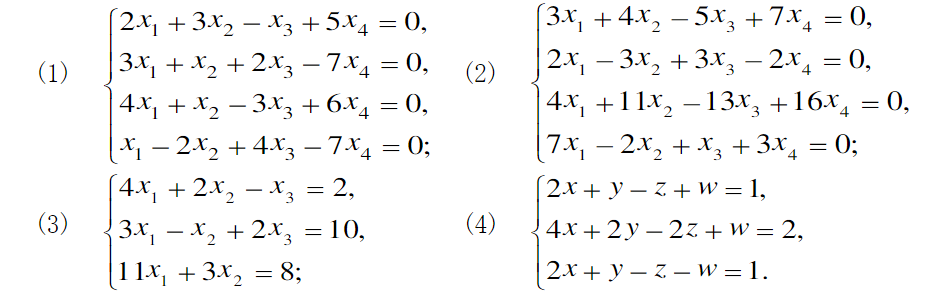
**练习题一：**人口数据分析。表1中数据为美国（us）和中国（cn）人口数据随年份的变化。要求：新建excel表格并将左边的年份和人口数据放置于表中

；新建脚本文件，读取excel表数据并采用多项式拟合对数据进行多项式拟合并绘图，同时并评估拟合效果；绘图后对图进行必要编辑，使图形信息一目了然，清晰美观。

表1 1790~2017年美国和中国人口数据



**练习题二：**利用MATLAB编程求解下列线性方程组



**提示：利用所学的线性代数知识，先利用MATLAB编写一小段程序判断方程解的情况（唯一解/无解/无穷解），然后在求解方程**