

《量化投资：以 MATLAB 为工具》-基础篇

N 分钟学会 MATLAB(60<N<180)Beta 版本

前言

之所以采用下文这种 Q&A 的形式来作为《量化投资：以 MATLAB 为工具》的基础篇，是想让刚刚接触 MATLAB 的读者能快速有效地了解 MATLAB，毕竟在一个注重时间效率的年代大家更喜欢速成的东西。

本篇形式上参考了刘思喆老师的《153 分钟学会 R》，当然内容方面结合了 MATLAB 本身的特色，本篇的内容来源多样，既有来自于 MATLAB 的官方帮助文档，也有来自我个人的一些总结，还有若干来自 MATLAB 技术论坛（<http://www.matlabsky.com>）的讨论问题。

MATLAB 是一个非常庞大的体系，其官方工具箱就有数十种，内部函数有数百个，可以说学习 MATLAB 是一件没有尽头的事情。

想来自己接触 MATLAB 已经有快 10 年的时间了，在学习 MATLAB 的道路上接触很多志同道合的朋友，希望这篇简单的“N 分钟学会 MATLAB(60<N<180)”能成为你认识、学习 MATLAB 的好助手。

李洋（faruto）

邮箱：faruto@163.com

微博：<http://weibo.com/faruto>

2013 年 7 月 31 日星期三

基础知识

1. MATLAB 是做什么的？为什么 MATLAB 叫做 MATLAB？

MATLAB 是美国 MathWorks 公司出品的商业数学软件，用于算法开发、数据可视化、数据分析以及数值计算的高级技术计算语言和交互式环境，主要包括 MATLAB 和 Simulink 两大部分。

MATLAB 是 Matrix 和 Laboratory 两个词的组合，意为矩阵工厂（矩阵实验室），之所以叫做矩阵工厂（矩阵实验室），是因为 MATLAB 的基本数据单位是矩阵。

2. MathWorks 公司的创始人是谁？

Cleve Moler 和 Jack Little 是 MathWorks 公司的创始人。其中 Cleve Moler 是 MATLAB 首个版本的开发者，Jack Little 是信号处理工具箱和控制系统工具箱早期版本的共同开发者和首席架构师。

3. 我是新手，我如何开始学习 MATLAB？

如果你的英文阅读还算可以，那么 MATLAB 官方的帮助文档将是最好且最全面的学习材料，你可以通过在 MATLAB 的命令窗口(Command Window)中键入“doc”来调出 MATLAB 官方的帮助文档，或者你也可以在 MathWorks 公司的官方网站的文档中心 (<http://www.mathworks.cn/cn/help/documentation-center.html>) 中查看在线的 MATLAB 官方帮助文档。

另外 MathWorks 公司的官方网站的用户中心 (<http://www.mathworks.cn/matlabcentral/>) 中有一些 MATLAB 相关问题的讨论以及一些 MATLAB 大牛的博客，其中包括 Cleve Moler 的博客——Cleve's Corner (<http://blogs.mathworks.com/cleve>)，在 MATLAB 用户中心你可以学习 MATLAB。

MATLAB 技术论坛 (<http://www.matlabsky.com>) 是一个不错的学习 MATLAB 的中文网站，本书的作者之一李洋(faruto)是 MATLAB 技术论坛核心管理团队的成员之一，在这个论坛里你可以找到大量的学习资料（代码、数据、视频等）或直接提出问题同大家讨论。

当然，你也可以购买一些 MATLAB 相关的外文书籍进行学习。

4. 使用 MATLAB 需要很厉害的编程能力吗？

大多数时候并不需要，因为 MATLAB 有很多的函数和工具箱，官方的工具箱还有数十种，第三方的工具箱不计其数而且每天都在增加，你用到的一般方法和函数都可以在 MATLAB 的工具箱中找到。

5. 能否简单举一个 MATLAB 的例子？

生成 100 个高斯（正态）分布随机数，并对这 100 个数进行特征描述。

MATLAB 代码

```
Mean_Value = 0;
STD_Value = 1;
Data_Num = 100;

x = random('Normal', Mean_Value, STD_Value, Data_Num, 1);
x_dataset = dataset(x);

whos
mean_x = mean(x)
std_x = std(x)
summary(x_dataset)
```

运行结果（由于是生成随机数，多次运行结果可能不同。）

Name	Size	Bytes	Class	Attributes
Data_Num	1x1	8	double	
Mean_Value	1x1	8	double	
STD_Value	1x1	8	double	
x	100x1	800	double	
x_dataset	100x1	2300	dataset	

```

mean_x =
    -0.0684
std_x =
    0.9568
x: [100x1 double]

```

min	1st quartile	median	3rd quartile	max
-2.8216	-0.7437	-0.088884	0.63638	2.3738

6. 如何查看使用的 MATLAB 的版本信息以及内存信息？

通过在 MATLAB 的命令窗口（Command Window）中键入“version”和“memory”来查看所使用的 MATLAB 版本信息和内存信息。

运行结果（由于 MATLAB 版本的不同以及电脑配置的不同，运行结果可能不同。）

```

>> version
ans =
7.14.0.739 (R2012a)
>> memory
Maximum possible array:      444 MB (4.659e+08 bytes) *
Memory available for all arrays: 1190 MB (1.248e+09 bytes) **
Memory used by MATLAB:      427 MB (4.476e+08 bytes)
Physical Memory (RAM):      3327 MB (3.489e+09 bytes)

* Limited by contiguous virtual address space available.
** Limited by virtual address space available.

```

7. MATLAB 支持中文吗？

支持！但在 MATLAB 中，有相当一部分工具箱的作者都是以英文为母语的，故建议熟练全英文环境。

8. MATLAB 支持自动补全（Tab Completion）么？

支持！在较新版本中，MATLAB 引入了命令自动补全功能，使用 Tab 键能自动补全 MATLAB 命令或给出所有可能的补全命令列表。

9. MATLAB 有哪些常用的热键？

MATLAB 除了支持常见的 Windows 热键（复制：Ctrl+C，剪切：Ctrl+X，粘贴：Ctrl+P 等等），在 MATLAB 文件编辑器（Editor）中，还有如下一些常见的热键：

F5：运行当前的 M 文件；

F9：运行选中的代码段；

Ctrl+R：注释选中的代码段；

Ctrl+T：反注释选中的代码段；

Ctrl+I：将选中的代码段智能缩进调整。

10. 如何清除变量？

使用 clear 命令可以清除工作空间（Workspace）中的所有变量，清除工作空间（Workspace）名字为 name 的变量，使用 clear name 命令。

11. 如何清空命令窗口（Command Window）？

使用 `clc` 命令可以清空命令窗口（Command Window）。

12. MATLAB 常用的数据类型有哪些？

MATLAB 支持的基本数据类型有基本数值类型、字符串、元胞数组、结构、函数句柄、Java 对象、逻辑类型等等。

其中基本数值类型包括基本数值类型包括双精度类型、单精度类型、整数类型等。

13. 如何查看函数的代码？

MATLAB 文件编辑器（Editor）菜单栏中可以使用 `file->Open` 直接打开相关 M 文件查看源码，或者也可以只用 `edit` 函数打开，比如

```
edit mean;
```

会直接打开求均值函数 `mean` 查看源码：

```
function y = mean(x,dim)
%MEAN    Average or mean value.
%
%   For vectors, MEAN(X) is the mean value of the elements in X. For
%   matrices, MEAN(X) is a row vector containing the mean value of
%   each column.  For N-D arrays, MEAN(X) is the mean value of the
%   elements along the first non-singleton dimension of X.
%
%   MEAN(X,DIM) takes the mean along the dimension DIM of X.
%
%   Example: If X = [1 2 3; 3 3 6; 4 6 8; 4 7 7];
%
%   then mean(X,1) is [3.0000 4.5000 6.0000] and
%   mean(X,2) is [2.0000 4.0000 6.0000 6.0000].
%
%   Class support for input X:
%       float: double, single
%
%   See also MEDIAN, STD, MIN, MAX, VAR, COV, MODE.

%   Copyright 1984-2009 The MathWorks, Inc.
%   $Revision: 5.17.4.5 $   $Date: 2010/09/02 13:35:22 $

if nargin==1,
    % Determine which dimension SUM will use
    dim = find(size(x)~=1, 1 );
    if isempty(dim), dim = 1; end

    y = sum(x)/size(x,dim);
else
    y = sum(x,dim)/size(x,dim);
```

end

14. MATLAB 里面可以使用科学计数法么？

可以。

$1e10 = 10000000000$

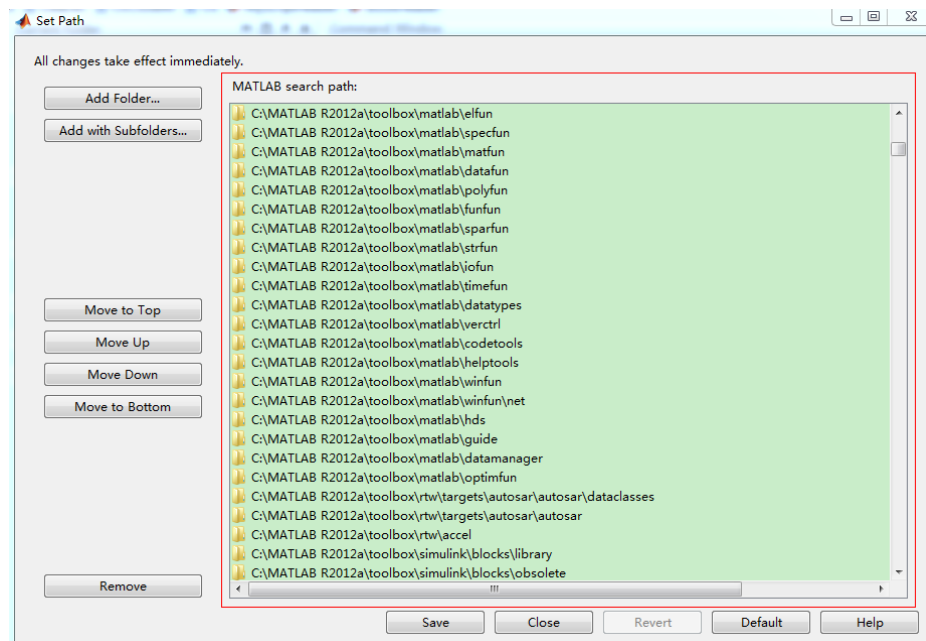
$1.2e-4 = 0.00012$

15. 什么叫做 MATLAB 的当前路径（目录）、工作搜索路径（目录）？

“当前路径（目录）（Current Folder）”是指 MATLAB 当前所在的路径，MATLAB 菜单栏下面有一个 Current Folder 可以在这里进行当前所在目录的更改，如下图所示：



“工作搜索路径（目录）”是指当你使用某一个函数的时候，MATLAB 可以进行搜索该函数的所有的目录集合。可以在 MATLAB 菜单栏中 File->Set Path 中查看所有的工作搜索路径（目录）集合，如下图所示：



当你使用某一个函数的时候，MATLAB 首先会从当前目录搜索调用该函数，如果当前目录没有该函数，MATLAB 就会从工作搜索目录按照从上到下的顺序进行搜索调用该函数，如果工作搜索目录中也没有该函数，会给出如下的报错

??? Undefined function or variable 'XXX'

16. 在 MATLAB 如何安装第三方的工具箱？

如果待安装的第三方的工具箱使用纯 M 语言编写的，这种情况比较简单，工具箱下载后只需将第三方的工具箱所在目录添加到 MATLAB 工作搜索目录即可，具体操作就是在 MATLAB 菜单栏中选择 File->Set Path->Add with Subfolders 然后选择之前存放第三方的工具箱的文件夹，然后点击保存（Save）就可以了。

如果待安装的第三方的工具箱并不是单纯使用 M 语言编写的，比如 LIBSVM 工具箱的 MATLAB 版本核心是使用 C++代码编写的，安装这样的第三方工具箱的步骤就稍微复杂一些，下面以安装 LIBSVM 工具箱为例简要介绍一下。

安装 LIBSVM 工具箱是在 MATLAB 平台下使用 LIBSVM 的前提, 如果没有安装好也就无法使用, 在 MATLAB 平台下安装 LIBSVM 工具箱的本质其实就是将 LIBSVM 工具箱的 MATLAB 版本文件 svmtrain.c 和 svmpredict.c 在 MATLAB 中进行编译生成 mex 文件(依操作系统不同, 32 位操作系统编译后生成 svmtrain.mexw32 和 svmpredict.mexw32, 64 位操作系统编译后生成 svmtrain.mexw64 和 svmpredict.mexw64), 进而能在 MATLAB 中进行使用, 安装 LIBSVM 工具箱主要有以下几个步骤:

(1) 下载 LIBSVM 工具箱并将其所在目录添加到 MATLAB 工作搜索目录

可在 <http://www.csie.ntu.edu.tw/~cjlin/libsvm/> 下载最新版本的 LIBSVM 工具箱, 然后在 MATLAB 菜单栏中选择 File->Set Path->Add with Subfolders, 并选择之前存放 LIBSVM 工具箱的文件夹, 最后点击保存 (Save) 就可以了。

(2) 选择编译器

由于 LIBSVM 的原始版本是用 C++写的, 这里为了能在 MATLAB 平台下使用, 需要用编译器编译一下, 生成一个 *.mexw32 或 *.mexw64 文件, 这样就可以在 MATLAB 平台下使用 LIBSVM 了。

如果编译器没有选择好的话, 下一步进行 make 编译的时候就会出现如下等报错:

```
Unable to complete successfully.
```

这个表示你没有选择好编译器。

这里需要你本机事先安装一个 C++编译器然后才能进行选择, 推荐使用 Microsoft Visual C++ 6.0 编译器或者更高版本的 Visual Studio, 一般 MATLAB 会自带一个编译器 Lcc-win32 C 但这个在这里无法使用, 因为 LIBSVM 源代码是用 C++写的, 而 Lcc-win32 C 是一个 C 编译器, 无法编译 C++源代码。

下面看一下选择编译器的具体操作。

首先在 MATLAB 命令窗 (Command Window) 中输入:

```
mex -setup
```

注意 mex 后面要打一个空格然后是-setup。

然后会出现类似如下内容:

```
Please choose your compiler for building external interface (MEX) files:
```

```
Would you like mex to locate installed compilers [y]/n?
```

这里问你是否选择本机已安装的编译器, 正规的选择 y 然后指定编译器即可, 我们先来看一下选择 y。

根据你本机安装的编译器, 会出现类似如下内容:

```
Please choose your compiler for building external interface (MEX) files:
```

```
Would you like mex to locate installed compilers [y]/n? y
```

```
Select a compiler:
```

```
[1] Lcc-win32 C 2.4.1 in D:\MATLAB~1\sys\lcc
```

```
[2] Microsoft Visual C++ 6.0 in D:\Microsoft Visual Studio
```

```
[0] None
```

```
Compiler:
```

然后你选择相应的编译器并确认即可:

```
Compiler: 2
```

Please verify your choices:

Compiler: Microsoft Visual C++ 6.0

Location: D:\Microsoft Visual Studio

Are these correct [y]/n? y

Trying to update options file:
C:\Users\faruto\AppData\Roaming\MathWorks\MATLAB\R2009b\mexopts.bat

From template: D:\MATLAB~1\bin\win32\mexopts\msvc60opts.bat

Done . . .

这样就表示编译器选择成功了（此步骤中可能会出现警告（warning）是正常现象）。

MATLAB 支持的编译器列表可以在这里查看：

http://www.mathworks.com/support/compilers/current_release/。

如果你输入 `mex -setup` 后在

Please choose your compiler for building external interface (MEX) files:

Would you like mex to locate installed compilers [y]/n?

这一步选择 y 后，可选择的编译器里面有没有你已经安装的编译器，表示 MATLAB 可能没有识别记录你安装的编译器的名字和目录（有时候会发生这种情况），此时你应该重新输入 `mex -setup` 后选择 n 手动进行编译器的设置：

`mex -setup`

Please choose your compiler for building external interface (MEX) files:

Would you like mex to locate installed compilers [y]/n? n

Select a compiler:

- [1] Intel C++ 9.1 (with Microsoft Visual C++ 2005 SP1 linker)
- [2] Intel Visual Fortran 10.1 (with Microsoft Visual C++ 2005 SP1 linker)
- [3] Lcc-win32 C 2.4.1
- [4] Microsoft Visual C++ 6.0
- [5] Microsoft Visual C++ .NET 2003
- [6] Microsoft Visual C++ 2005 SP1
- [7] Microsoft Visual C++ 2008 Express
- [8] Microsoft Visual C++ 2008 SP1
- [9] Open WATCOM C++

[0] None

Compiler: 4 %选择的这个编译器一定是你本机安装了的，否则选择了也没有用

Your machine has a Microsoft Visual C++ compiler located at

```
D:\Microsoft Visual Studio. Do you want to use this compiler [y]/n?
```

这样的话就可以手动选择你想要的编译器了，在

```
Your machine has a Microsoft Visual C++ compiler located at
```

```
D:\Microsoft Visual Studio. Do you want to use this compiler [y]/n?
```

这个确认步骤，如果你的编译器的确是安装在 MATLAB 给出的这个目录（我这里是 D:\Microsoft Visual Studio）那么选择 y 确认即可，如果不是说明 MATLAB 没有识别出安装的地方，选择 n 手动指定目录即可，比如选择 n 后的结果如下：

```
Compiler: 4
```

```
Your machine has a Microsoft Visual C++ compiler located at
```

```
D:\Microsoft Visual Studio. Do you want to use this compiler [y]/n? n
```

```
Please enter the location of your compiler: [C:\Program Files\Microsoft Visual Studio]
```

此时输入你安装的编译器的完整目录即可。

（3）编译文件

这一步的具体操作就是运行 LIBSVM 工具箱中 MATLAB 版本文件夹中的 make.m 文件。

首先需要把 MATLAB 的当前目录（Current Folder）调整到 LIBSVM 工具箱所在的文件夹，然后在 MATLAB 命令窗口（Command Window）输入

```
make
```

如果成功运行没有报错，到此就说明 LIBSVM 工具箱成功安装了。LIBSVM 工具箱中有自带的 heart_scale.mat 测试数据集，可以运行以下代码来检查一下是否安装成功：

```
load heart_scale;
model = svmtrain(heart_scale_label,heart_scale_inst);
[predict_label,accuracy]=svmpredict(heart_scale_label,heart_scale_inst,model);
```

如果出现下面这个结果，则说明肯定安装成功了：

```
Accuracy = 86.6667% (234/270) (classification)
```

这里之所以需要将 MATLAB 的当前目录（Current Folder）调整到 LIBSVM 工具箱所在的文件夹是因为这一步要运行 LIBSVM 工具箱中的 make.m 文件，而当你使用某一个函数的时候，MATLAB 首先会从当前目录搜索调用该函数，为了防止其他位置也有类似名字的 make.m 函数进而运行错误，所以这一步要把 MATLAB 的当前目录（Current Folder）调整到 LIBSVM 所在的文件夹，优先运行 LIBSVM 文件夹下的 make.m 文件。

make.m（LIBSVM 版本 3.14）的文件内容如下：

```
% This make.m is for MATLAB and OCTAVE under Windows, Mac, and Unix
```

```
try
```

```
    Type = ver;
```

```
    % This part is for OCTAVE
```

```
    if(strcmp(Type(1).Name, 'Octave') == 1)
```

```
        mex libsvmread.c
```

```
        mex libsvmwrite.c
```

```
        mex svmtrain.c ../svm.cpp svm_model_matlab.c
```

```
        mex svmpredict.c ../svm.cpp svm_model_matlab.c
```

```
    % This part is for MATLAB
```

```
    % Add -largeArrayDims on 64-bit machines of MATLAB
```

```
    else
```



```

        mex CFLAGS="$CFLAGS -std=c99" -largeArrayDims libsvmread.c
        mex CFLAGS="$CFLAGS -std=c99" -largeArrayDims libsvmwrite.c
        mex CFLAGS="$CFLAGS -std=c99" -largeArrayDims svmtrain.c ../svm.cpp
svm_model_matlab.c
        mex CFLAGS="$CFLAGS -std=c99" -largeArrayDims svmpredict.c ../svm.cpp
svm_model_matlab.c
    end
catch
    fprintf('If make.m fails, please check README about detailed instructions.\n');
end

```

安装完 LIBSVM 工具箱后，可能有人要用 `help svmtain` 和 `help svmpredict` 来查看这两个函数的帮助文件，但结果会是运行 `help svmtain` 在较新版本下得到的是 MATLAB 自带的 `svmtrain` 函数的帮助文件；运行 `help svmpredict` 会有如下报错：

```
svmpredict not found.
```

因为 `svmtrain` 和 `svmpredict` 的源代码是 `svmtrain.c` 和 `svmpredict.c` 即源代码用 C++ 写的，编译后生成的文件是 `svmtrain.mexw32` 和 `svmpredict.mexw32`（或 `svmtrain.mexw64` 和 `svmpredict.mexw64`），而 `*.mexw32`（或 `*.mexw64`）这个编译后的文件是加密过的，打开是乱码，根本就没有帮助文件说明注释，想看 `svmtrain` 和 `svmpredict` 的源代码可以直接查看 `svmtrain.c` 和 `svmpredict.c`。

由于较新版本的 MATLAB 有自带的 SVM 实现，其函数名也为 `svmtrain`，文件位置在 MATLAB 根目录下 `\toolbox\bioinfo\biolearning\svmtrain.m`，为避免 MATLAB 自带的 `svmtrain` 函数与 LIBSVM 工具箱 `svmtrain` 函数调用错误，可以将 MATLAB 自带的 `svmtrain.m` 函数备份后改名，比如改成 `svmtrain_matlab.m` 或者 `svmtrain.m_backup`。

17. 如何恢复 MATLAB 的文件关联？

有时候重装系统或者其他误操作，会导致 MATLAB 文件关联失效，运行如下代码可以恢复 MATLAB 的文件关联。

```

cwd=pwd;
cd([matlabroot '\toolbox\matlab\winfun\private']);
fileassoc('add',{'m','.mat','.fig','.p','.mdl','.' mexext}));%重点
cd(cwd);
disp('Changed Windows file associations. FIG, M, MAT, MDL, MEX, andP files are now associated with
MATLAB.')

```

输入输出

18. 如何输入一个矩阵并求其转置和逆？

有以下几种方式可以输入一个矩阵。

（1）直接输入矩阵

比如在命令窗口（Command Window）中输入

```
>> A = [1 2 3 4; 0 1 0 0; 1 1 2 4]
```

可以定义矩阵 $\begin{pmatrix} 1 & 2 & 3 & 4 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 2 & 4 \end{pmatrix}$

如果矩阵中的元素的间隔是等步长的，可以使用如下方式输入矩阵

```
>> A=1:0.1:5
```

表示从 1 到 5，按照 0.1 步长生成行向量。

(2) 使用函数生成矩阵

linspace 函数可以批量生成向量，其用法有两种：

linspace(a, b)表示将 a 到 b 等分成 100 份形成向量；

linspace(a, b, n)表示将 a 到 b 等分成 n 份形成向量；

ones 函数和 zeros 函数可以生成指定维数的全为 1 和全为 0 的矩阵，eye 函数可以生成单位阵，magic 函数可以生成魔方矩阵，vander 函数可以生成范得蒙(Vandermonde)矩阵，hilb 函数可以生成希尔伯特矩阵，Toeplitz 函数可以生成托普利兹(Toeplitz)矩阵，compan 函数可以生成伴随矩阵，pascal 函数可以生成帕斯卡(Pascal)矩阵。

单引号的作用是求矩阵的转置，inv 函数可以求一个矩阵的逆，见下例

```
>> A=[1 2 3 4; 0 1 0 0; 1 1 2 4; 3 1 2 4]
```

```
A =
```

```
1     2     3     4
0     1     0     0
1     1     2     4
3     1     2     4
```

```
>> A'
```

```
ans =
```

```
1     0     1     3
2     1     1     1
3     0     2     2
4     0     4     4
```

```
>> inv(A)
```

```
ans =
```

```
0    -0.0000   -0.5000    0.5000
0     1.0000     0         0
1.0000  -1.0000  -1.0000     0
-0.5000   0.2500   0.8750  -0.1250
```

19. MATLAB 可以读 Excel 的数据？

可以，MATLAB 不但可以直接读取 Excel 文件，还可以向 Excel 文件中写入数据。使用 xlsread 和 xlswrite 函数可以达到 MATLAB 与 Excel 通讯的目的，见下例：

```
values = {1, 2, 3 ; 4, 5, 'x' ; 7, 8, 9};
headers = {'First', 'Second', 'Third'};
xlswrite('eg_Mat2Excel.xlsx', [headers; values]);
```

```
A = xlsread('eg_Mat2Excel.xlsx')
```

运行的结果是，通过 `xlswrite` 函数建立了 `eg_Mat2Excel.xlsx` 文件，其内容为：

First	Second	Third
1	2	3
4	5	x
7	8	9

通过 `xlsread` 函数可以读取 `eg_Mat2Excel.xlsx` 的内容：

```
A =  
    1     2     3  
    4     5  NaN  
    7     8     9
```

20. 可以将 MATLAB 命令窗口（Command Window）中显示的结果输出到文件吗？

可以，使用 `diary` 函数可以将命令窗口（Command Window）中显示的结果输出到文件。

21. MATLAB 可以从系统剪切板（clipboard）中读取数据吗？

可以，使用 `clipboard` 函数可以从系统剪切板读取数据，使用方法如下：

```
clipboard('copy', data)  
str = clipboard('paste')  
data = clipboard('pastpecial')
```

22. MATLAB 可以直接从数据库读取数据吗？

可以，MATLAB 专门有一个数据库工具箱（Database Toolbox），可以通过 ODBC/JDBC 接口访问具体的数据库，以 Matlab 通过 ODBC 建立到 Oracle 数据库的连接为例进行介绍。

（1）在 Windows 下“开始”→“控制面板”→“性能和维护”→“管理工具”，点击“数据源（ODBC）”

（2）选择“系统 DSN”→“添加”→选择“Oracle in OraClient11g_home1”，点击“完成”。

（3）配置 Oracle ODBC 驱动。

（4）点击“确定”，即可。

（5）在 Matlab 中建立到 Oracle 的 JDBC 连接测试。

```
>> conn = database(instance,username,password);
```

数据处理

23. 如何删掉缺失值？

在 MATLAB 中使用 NaN（Not-a-Number）表示缺失值，

```
A = [1 NaN 3]
```

判断矩阵（向量）某一元素是否为 NaN 时不能使用如下方式：

```
A(2) == NaN
```

需要使用 `isnan` 函数，使用如下方式可以删除缺失值：

```
A(isnan(A)) = []
```

运行结果

```
A =  
    1     3
```

可以看到缺失值已经被删除了。

24. 如何将字符串转变为命令执行？

使用 `eval` 函数可以将字符串转变为命令执行，参考下例：

```
A = 3;  
B = 5;  
  
string = [num2str(A),'+',num2str(B)]  
  
eval(string);
```

运行结果

```
string =  
3+5  
ans =  
8
```

有时候巧妙的使用 `eval` 函数会使某些程序实现起来更加方便灵活。

25. 如何向一个向量追加元素？

这在 **MATLAB** 中实现非常容易，见下例

```
A = [1 2 3 4]  
  
A(end+1) = 5
```

运行结果

```
A =  
    1     2     3     4  
A =  
    1     2     3     4     5
```

26. 如何移除矩阵的某行（列）数据？

涉及到 **MATLAB** 中矩阵的一些操作，见下例

```
A = magic(5)  
B = A;  
C = A;  
  
B(5,:) = []  
C(:,5) = []
```

运行结果

```
A =  
    17    24     1     8    15  
    23     5     7    14    16  
     4     6    13    20    22  
    10    12    19    21     3
```

```

    11    18    25     2     9
B =
    17    24     1     8    15
    23     5     7    14    16
     4     6    13    20    22
    10    12    19    21     3
C =
    17    24     1     8
    23     5     7    14
     4     6    13    20
    10    12    19    21
    11    18    25     2

```

27. 如何比较两个矩阵是否相同呢？

比较每个元素是否相同，如果每个元素都相同，那么这两个矩阵也相同，可以使用循环来达到这一目的，但使用 `all` 函数可以更方便的实现这一目的，`all` 函数会检查一个矩阵的每一列是否都为非零或者逻辑真，见下例

```

A = magic(5)
B = A;

B(1) = 888

whetherAequalsB = all( all( A == B ) )

```

运行结果

```

A =
    17    24     1     8    15
    23     5     7    14    16
     4     6    13    20    22
    10    12    19    21     3
    11    18    25     2     9
B =
  888    24     1     8    15
    23     5     7    14    16
     4     6    13    20    22
    10    12    19    21     3
    11    18    25     2     9
whetherAequalsB =
     0

```

28. 如何去掉数据中的重复元素？

参考 `unique` 函数，`unique` 函数可以去掉矩阵中的重复元素。

```

A = [1 1 2 2 3 4 5 6 6]

A_unique = unique(A)

```

运行结果

```
A =  
    1    1    2    2    3    4    5    6    6  
A_unique =  
    1    2    3    4    5    6
```

29. 如何求数据的极值？

使用 `min` 和 `max` 函数可以求数据的极小值和极大值，见下例：

```
A = 1:10;  
  
A_min = min(A)  
  
A_max = max(A)
```

运行结果

```
A =  
    1    2    3    4    5    6    7    8    9   10  
A_min =  
    1  
A_max =  
   10
```

30. 如何求矩阵最大(小)值并返回其行列号？

主要有几种方式，见下面的实例

```
% 方法 1  
disp('方法 1');  
A=[0 17 50;-12 40 3;5 -10 2;30 4 3]  
[C,I]=max(A(:))  
[m,n]=ind2sub(size(A),I)  
  
% 方法 2  
disp('方法 2');  
A=[0 17 50;-12 40 3;5 -10 2;30 4 3]  
[M,I]=max(A)  
[N,J]=max(M)  
[I(J),J]  
  
% 方法 3  
disp('方法 3');  
A=[0 17 50;-12 40 3;5 -10 2;30 4 3]  
N=max(max(A))    %或者 N=max(A(:))  
[r,c]=find(N==A)  
  
% 方法 4  
disp('方法 4');  
A=[0 17 50;-12 40 3;5 -10 2;30 4 3]  
[Y_col,Ind_row]=max(A)    %每列的最大值及行号  
[Y_row,Ind_col]=max(A')    %每行的最大值及列号
```

运行结果

方法 1

```
A =  
    0    17    50  
   -12   40     3  
     5   -10     2  
    30     4     3
```

```
C =  
    50
```

```
I =  
     9
```

```
m =  
     1
```

```
n =  
     3
```

方法 2

```
A =  
    0    17    50  
   -12   40     3  
     5   -10     2  
    30     4     3
```

```
M =  
    30    40    50
```

```
I =  
     4     2     1
```

```
N =  
    50
```

```
J =  
     3
```

```
ans =  
     1     3
```

方法 3

```
A =  
    0    17    50  
   -12   40     3  
     5   -10     2  
    30     4     3
```

```
N =  
    50
```

```
r =  
     1
```

```
c =  
     3
```

方法 4

```

A =
     0     17     50
    -12     40      3
     5    -10      2
    30      4      3
Y_col =
    30     40     50
Ind_row =
     4      2      1
Y_row =
    50     40      5     30
Ind_col =
     3      2      1      1

```

31. 如何判断数据是否为数字？

参考 `isnumeric` 函数。

32. 如何从一组数据中随机抽取数据？

使用 `randsrc` 函数可以从一组数据中按照指定概率随机抽取数据，其用法之一如下：

```
out = randsrc(m,n,[alphabet; prob])
```

意义是从 `alphabet` 数据中按照概率 `prob` 抽取数据生成一个 `m*n` 的矩阵，见下例

```
out = randsrc(5,5,[-3 -1 1 3; .2 .3 .3 .2])
```

即从`[-3 -1 1 3]`中抽取数据生成 `5*5` 的矩阵，并且使得`-3` 和 `3` 出现的概率为 `0.2`，`-1` 和 `1` 出现的概率为 `0.3`，运行结果（由于是随机抽取数据，多次运行结果可能不同。）

```

out =
     3     -3      3     -1     -1
    -1      1     -1     -1     -1
     1      1      1      3      1
     1     -3     -3     -3      3
     3      3      1     -1     -3

```

33. 如何将数据标准化？

使用 `zscore` 函数可以将数据标准化，这里标准化的定义是将数据减去其均值再除以其标准差，见下例

```
x = [1 1 2 3 5 8 11]
```

```
Z = zscore(x)
```

```
Ztemp = ( x-mean(x) )./std(x)
```

运行结果

```

x =
     1      1      2      3      5      8     11

Z =
    -0.8967    -0.8967    -0.6352    -0.3736     0.1495     0.9341     1.7187

```



```
Ztemp =  
    -0.8967    -0.8967    -0.6352    -0.3736     0.1495     0.9341     1.7187
```

数学运算

34. 如何计算积分？

符号积分可以使用 `int` 函数来实现，比如计算 $\sin(x)$ 的积分，见下例

```
syms x  
int(sin(x))
```

运行结果

```
ans =  
-cos(x)
```

数值积分可以使用 `integral` 函数实现，比如计算 $\sin(x)$ 从 0 到 π 的积分，见下例

```
integral(@sin, 0, pi)
```

运行结果

```
ans =  
    2.0000
```

35. MATLAB 如何进行复数计算？

参看 `complex` 函数，见下例

```
Z = complex(3,4)  
  
% 实部  
Z_real = real( Z )  
% 虚部  
Z_imag = imag( Z )  
% 模  
Z_abs = abs( Z )  
% 辐角  
Z_angle = angle( Z )  
% 共轭  
ZC = conj( Z )
```

运行结果

```
Z =  
    3.0000 + 4.0000i  
Z_real =  
    3  
Z_imag =  
    4  
Z_abs =
```

```

5
Z_angle =
0.9273
ZC =
3.0000 - 4.0000i

```

36. 如何生成对角矩阵？

使用 `diag` 函数可以构造对角矩阵或者取得一个矩阵的对角元素，见下例

```

X = diag([1 1 2 3 5 8])

x = magic(5);

x_diag = diag(x)

```

运行结果

```

X =
    1     0     0     0     0     0
    0     1     0     0     0     0
    0     0     2     0     0     0
    0     0     0     3     0     0
    0     0     0     0     5     0
    0     0     0     0     0     8

x =
    17    24     1     8    15
    23     5     7    14    16
     4     6    13    20    22
    10    12    19    21     3
    11    18    25     2     9

x_diag =
    17
     5
    13
    21
     9

```

37. 求矩阵的特征值和特征向量的函数是什么？

参考 `eig` 函数。

38. 如何构造上（下）三角矩阵？

参考 `tril` 和 `triu` 函数，见下例

```

tl = tril(ones(4,4),-1)

tu = triu(ones(4,4),-1)

```

运行结果

```

tl =

```

```

0    0    0    0
1    0    0    0
1    1    0    0
1    1    1    0
tu =
1    1    1    1
1    1    1    1
0    1    1    1
0    0    1    1

```

39. MATLAB 常用的运算符号有哪些？

除了常见的加 (“+”)、减 (“-”)、乘 (“*”)、除 (分为左除 “\” 和右除 “/”), 幂次方 (“^”) 运算, MATLAB 还有点乘 (“.*”)、点除 (分为点左除 “.\” 和点右除 “./”)、点幂次方 (“.^”) 运算。

这里要特别说明的是当 A、B 为矩阵时, n 为标量数值时, A+B, A-B, A*B, A\B, A/B, A^n 表示的是相关的矩阵运算 (A, B 的维数需要满足相关矩阵运算的要求), 其中 A\B, A/B, A^n 的定义分别为:

- (1) 当 A 是一个方阵时, 则 $A \setminus B = \text{inv}(A) * B$, 其中 inv 为 MATLAB 的函数, 表示求矩阵的逆;
- (2) 当 A 是一个 $m \times n$ 的矩阵 ($m \neq n$), B 是一个 $m \times 1$ 的列向量, 则 $X = A \setminus B$ 是线性系统 $AX = B$ 的最小二乘解;
- (3) 当 B 是一个方阵时, 则 $A/B = A * \text{inv}(B)$, 其中 inv 为 MATLAB 的函数, 表示求矩阵的逆;
- (4) 当 A 是一个 $m \times n$ 的矩阵 ($m \neq n$), B 是一个 $m \times 1$ 的列向量, 则 $X = A/B$ 是线性系统 $XB = A$ 的最小二乘解;
- (5) A\B 与 A/B 的转换关系为 $A/B = (B \setminus A)'$;
- (6) A^n 表示 n 个矩阵 A 相乘。

当 A、B 为矩阵时, 点乘 (“.*”)、点除 (分为点左除 “.\” 和点右除 “./”)、点幂次方 (“.^”) 运算表示的是相关矩阵的元素之间的运算。

看下例, 加深一下对于 MATLAB 常用的运算符号的理解。

```

a=magic(3)
b=pascal(3)

disp('a/b')
a/b
disp('a*inv(b)')
a*inv(b)

disp('a\b')
a\b
disp('inv(a)*b')
inv(a)*b

a/b-(b\A)

disp('a^3')

```

```

a^3
disp('a*a*a')
a*a*a

disp('a.*b')
a.*b
disp('a.\b')
a.\b
disp('a./b')
a./b
disp('a.^b')
a.^b

```

运行结果

```

a =
     8     1     6
     3     5     7
     4     9     2

b =
     1     1     1
     1     2     3
     1     3     6

a/b
ans =
    27    -31     12
     1     2      0
    -13     29    -12

a*inv(b)
ans =
    27.0000   -31.0000    12.0000
     1.0000     2.0000     0.0000
   -13.0000    29.0000   -12.0000

a\b
ans =
    0.0667    0.0500    0.0972
    0.0667    0.3000    0.6389
    0.0667    0.0500   -0.0694

inv(a)*b
ans =
    0.0667    0.0500    0.0972
    0.0667    0.3000    0.6389
    0.0667    0.0500   -0.0694

ans =
     0     0     0
     0     0     0

```

```

      0      0      0
a^3
ans =
      1197      1029      1149
      1077      1125      1173
      1101      1221      1053
a*a*a
ans =
      1197      1029      1149
      1077      1125      1173
      1101      1221      1053
a.*b
ans =
      8      1      6
      3     10     21
      4     27     12
a.\b
ans =
      0.1250      1.0000      0.1667
      0.3333      0.4000      0.4286
      0.2500      0.3333      3.0000
a./b
ans =
      8.0000      1.0000      6.0000
      3.0000      2.5000      2.3333
      4.0000      3.0000      0.3333
a.^b
ans =
      8      1      6
      3     25     343
      4    729     64

```

40. 如何求矩阵各行（列）的均值？

使用 `mean` 函数可以进行均值的计算，见下例

```

a = magic(4)
% 各行均值
a_rowmean = mean(a, 2)
% 各列均值
a_colmean = mean(a, 1)

```

运行结果

```

a =
      16      2      3     13
       5     11     10      8
       9      7      6     12

```

```

    4    14    15    1
a_rowmean =
    8.5000
    8.5000
    8.5000
    8.5000
a_colmean =
    8.5000    8.5000    8.5000    8.5000

```

41. 如何计算组合数或得到所有组合？

nchoosek 函数可以用来计算组合数，并给出所有元素的组合，factorial 函数可以用来计算阶乘，见下例

```

combnum = nchoosek(4, 2)

comb = nchoosek([1 2 3 4], 2)

fac = factorial(4)

```

运行结果

```

combnum =
    6
comb =
    1     2
    1     3
    1     4
    2     3
    2     4
    3     4
fac =
    24

```

42. 如何在 MATLAB 里面求（偏）导数？

使用 diff 函数，见下例

```

syms x
dy_dx = diff( sin(x)/x )

```

运行结果

```

dy_dx =
cos(x)/x - sin(x)/x^2

```

43. 如何求一元方程的根？

使用 solve 函数可以求解方程的根，见下例

```

syms x

s1 = solve( x^2-1 )

```

```
s2 = solve(x^2 + 4*x + 1 == 0)
```

```
s3 = solve(x^4 + 1 == 2*x^2 - 1)
```

运行结果

```
s1 =
```

```
1
```

```
-1
```

```
s2 =
```

```
3^(1/2) - 2
```

```
- 3^(1/2) - 2
```

```
s3 =
```

```
(1 + i)^(1/2)
```

```
(1 - i)^(1/2)
```

```
-(1 + i)^(1/2)
```

```
-(1 - i)^(1/2)
```

44. 如何模拟高斯（正态）分布数据？

使用 `random('Normal', Mean_Value, STD_Value, N, M)` 可以产生 $N \times M$ 个来自于均值为 `Mean_Value`，标准差为 `STD_Value` 的高斯（正态）分布数据。

其中 'Normal' 是个参数选项，改变这个参数也可以生成其他的分布数据（相应地后面的分布的参数也需要修改），主要的分布有

参数	分布名称
'beta' or 'Beta'	Beta Distribution
'bino' or 'Binomial'	Binomial Distribution
'chi2' or 'Chisquare'	Chi-Square Distribution
'exp' or 'Exponential'	Exponential Distribution
'ev' or 'Extreme Value'	Extreme Value Distribution
'f' or 'F'	F Distribution
'gam' or 'Gamma'	Gamma Distribution
'gev' or 'Generalized Extreme Value'	Generalized Extreme Value Distribution
'gp' or 'Generalized Pareto'	Generalized Pareto Distribution
'geo' or 'Geometric'	Geometric Distribution
'hyge' or 'Hypergeometric'	Hypergeometric Distribution
'logn' or 'Lognormal'	Lognormal Distribution
'nbin' or 'Negative Binomial'	Negative Binomial Distribution
'ncf' or 'Noncentral F'	Noncentral F Distribution
'nct' or 'Noncentral t'	Noncentral t Distribution
'ncx2' or 'Noncentral Chi-square'	Noncentral Chi-Square Distribution
'norm' or 'Normal'	Normal Distribution
'poiss' or 'Poisson'	Poisson Distribution
'rayl' or 'Rayleigh'	Rayleigh Distribution
't' or 'T'	Student's t Distribution
'unif' or 'Uniform'	Uniform Distribution (Continuous)
'unid' or 'Discrete Uniform'	Uniform Distribution (Discrete)

字符操作

45. MATLAB 对大小写敏感吗？

MATLAB 对大小写是敏感的，可以使用 `lower`、`upper` 等函数对字符进行转化。

46. 如何在 MATLAB 中定义带引号的字符串？

在 MATLAB 编程中，如果想得到的带有引号的字符串（字符串中本身带有引号），怎么实现？输入 `str = 'string'` 得到的是

```
>> str = 'string'
str =
string
```

其中 `str` 里没有单引号，如果像下面这样

```
>> str = ' 'string' '
??? str = ' 'string' '
Error: Unexpected MATLAB expression.
```

又会出现报错，原因是 MATLAB 在进行匹配是一个单引号' 对一个单引号 '。如果字符串中有单引号，还按照一般的方式进行输入就会造成匹配的错误，就上面那个错误提示了。正确的解决方式是：

```
>> str = "'string'"
str =
'string'
```

输入三个单引号就可以实现了。如果你想要双引号，比如 `str = "string"` 此时就直接输入双引号就行了。

```
>> str = '"string"'
str =
"string"
```

注意：此时不是三个单引号，是一对单引号和一对双引号，放到 MATLAB 中能很容易的看清楚。

47. 在 MATLAB 中如何使用正则表达式（Regular Expressions）？

正则表达式就是一个表达式（也是一串字符），它定义了某种字符串模式，利用正则表达式，可以对大段的文字进行复杂的查找、替换等。MATLAB 提供的正则表达式主要有三个：

`regexp`: 用于对字符串进行查找，大小写敏感；

`regexpi`: 用于对字符串进行查找，大小写不敏感；

`regexprep`: 用于对字符串进行查找并替换。

详细用法可以参考 MATLAB 官方帮助文件以及下面这个帖子：

Matlab 正则表达式零基础起步教程

<http://www.matlabsky.com/thread-22038-1-1.html>

48. 如何在字符串中选取特定位置的字符？

只需根据相应的下标就可以获取特定位置的字符，见下例

```
str = 'abcdef'

substr1 = str( [2 4] )

substr2 = str( 1:3 )
```

运行结果

```
str =
abcdef
substr1 =
bd
substr2 =
abc
```

49. 如何返回字符个数？

使用 `length` 函数可以计算字符的个数。

日期时间

50. 日期可以做算术运算么？

可以，但需要使用 `datenum` 函数将日期（字符串）转化为 MATLAB 可识别的数值类数据，进而可以进行算术运算，见下例

```
n1 = datenum( '01/08/2013', 'dd/mm/yyyy' )

n2 = datenum( '08/08/2013', 'dd/mm/yyyy' )

n3 = n1 + 2

d1 = datestr( n1, 'dd/mm/yyyy' )

d2 = datestr( n2, 'dd/mm/yyyy' )

d3 = datestr( n3, 'dd/mm/yyyy' )
```

运行结果

```
n1 =
    735447
n2 =
    735454
n3 =
    735449
d1 =
```

```
01/08/2013
d2 =
08/08/2013
d3 =
03/08/2013
```

51. 如何将日期表示为“07-Aug- 2013”这种格式？

结合使用 `datenum` 和 `datestr` 函数可以实现日期格式的转换，见下例

```
d = datestr( datenum( '07/08/2013', 'dd/mm/yyyy' ), 'dd-mmm-yyyy' )
```

运行结果

```
d =
07-Aug-2013
```

绘图相关

52. 如何在同一画面画出多张图？

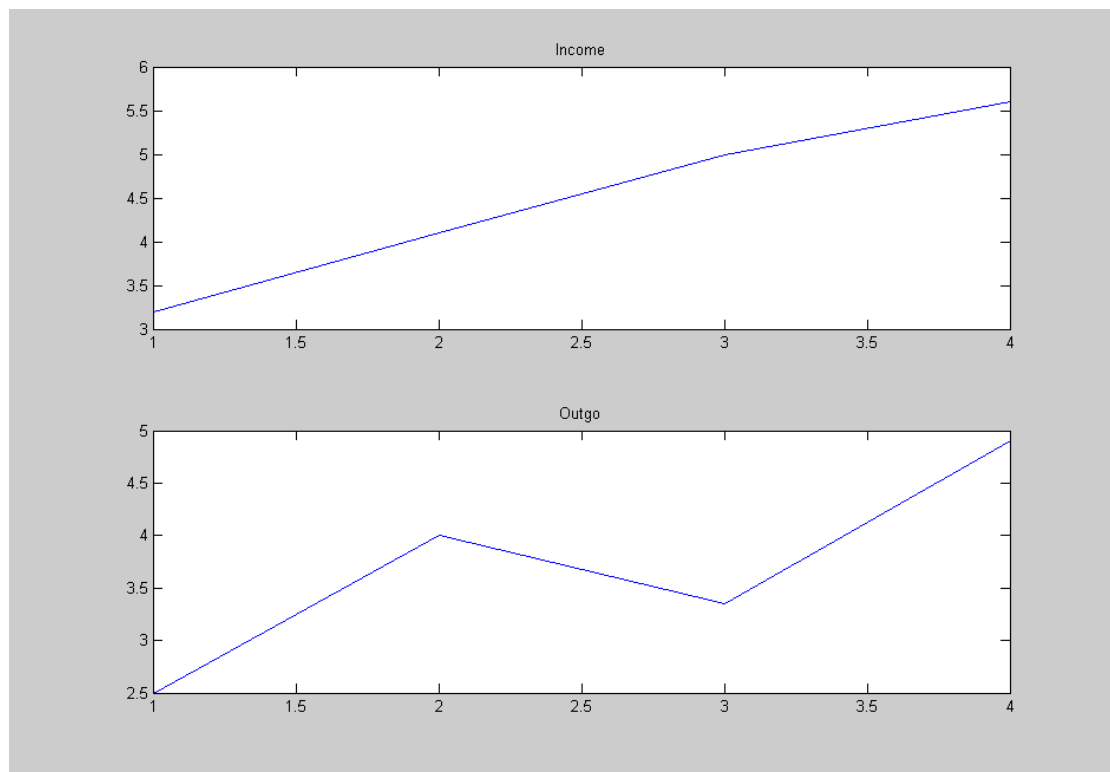
使用 `subplot` 函数可以在同一个画面画出多张图，见下例

```
figure;
income = [3.2,4.1,5.0,5.6];
outgo = [2.5,4.0,3.35,4.9];

subplot(2,1,1);
plot(income);
title('Income');

subplot(2,1,2);
plot(outgo);
title('Outgo');
```

运行结果



53. 如何加图例？

使用 `legend` 函数可以增加图例。

54. 怎么做饼图？

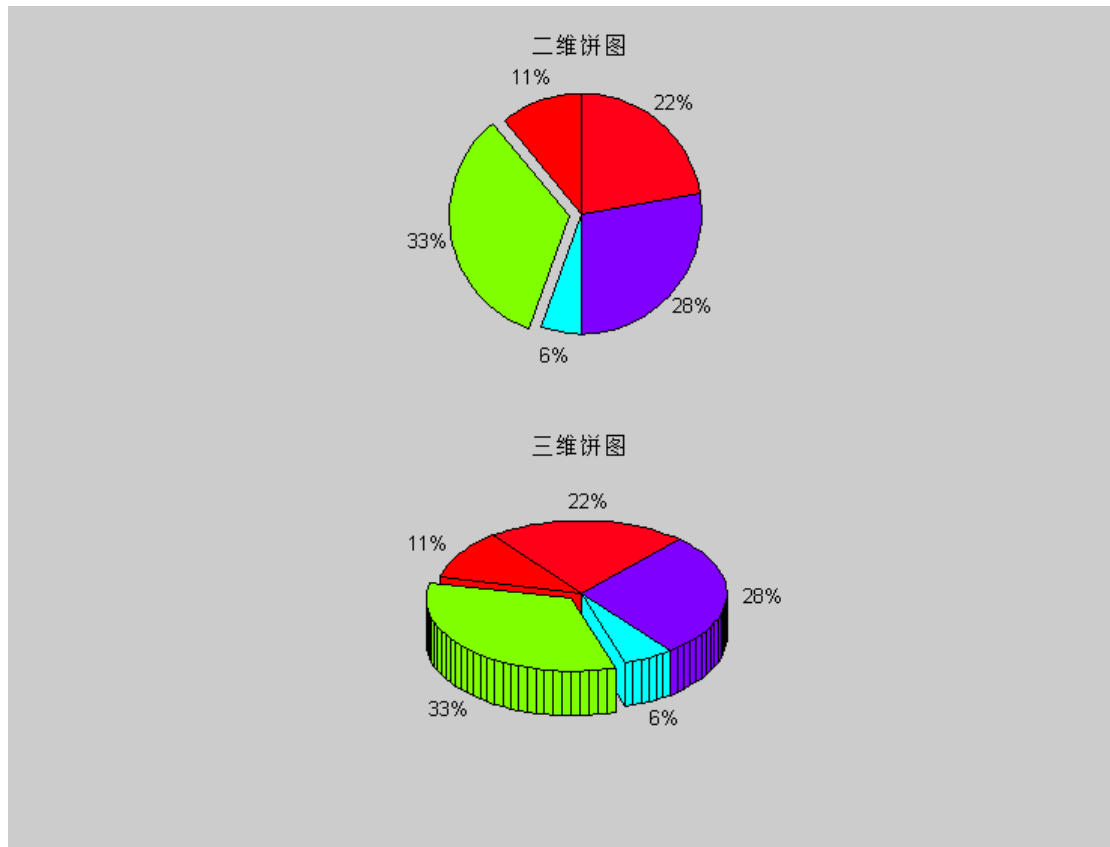
`pie` 函数可以制作二维的饼图，`pie3` 函数可以制作三维的饼图，见下例

```
x = [1 3 0.5 2.5 2];
explode = [0 1 0 0 0];
```

```
figure;
subplot(2,1,1);
pie(x,explode);
title('二维饼图');
colormap jet
```

```
subplot(2,1,2);
pie3(x,explode);
title('三维饼图');
colormap hsv
```

运行结果



55. 如何做茎叶图？

参考 stem 函数。

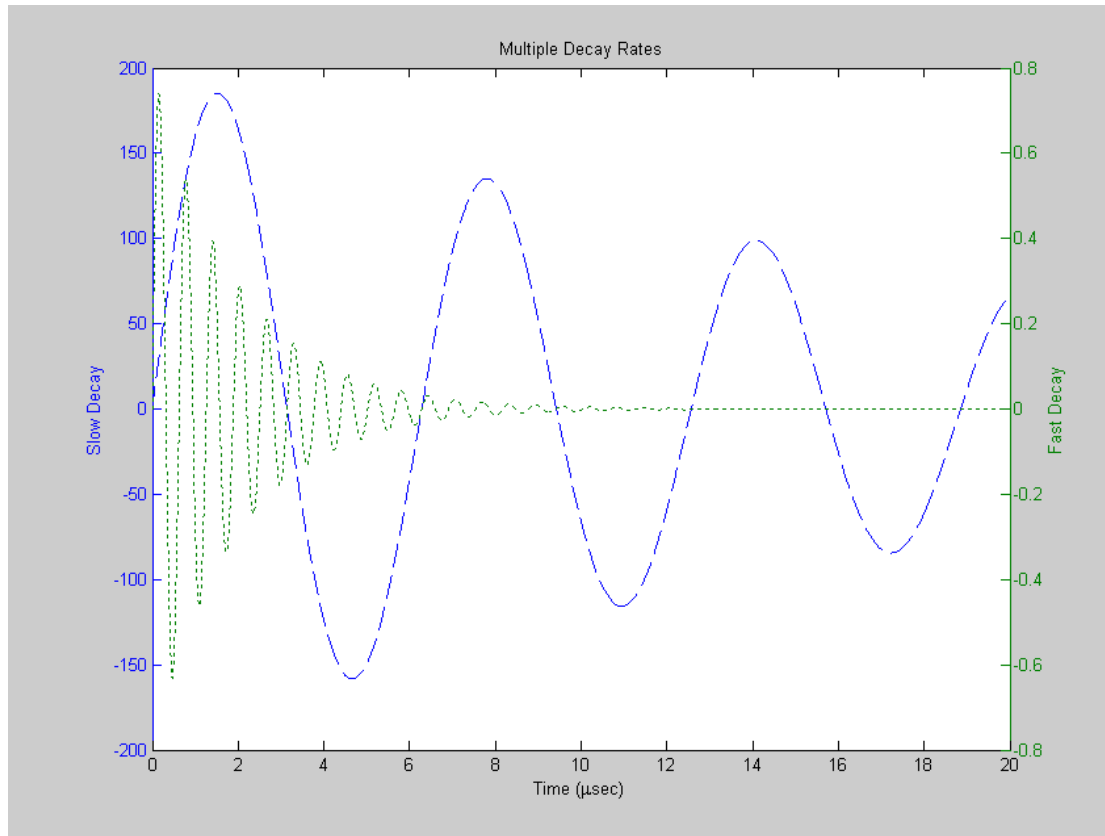
56. MATLAB 如何做双坐标？

使用 plotyy 函数可以绘制双坐标，参考下例

```
figure
x = 0:0.01:20;
y1 = 200*exp(-0.05*x).*sin(x);
y2 = 0.8*exp(-0.5*x).*sin(10*x);
[AX,H1,H2] = plotyy(x,y1,x,y2,'plot');

set(get(AX(1),'Ylabel'),'String','Slow Decay')
set(get(AX(2),'Ylabel'),'String','Fast Decay')
xlabel('Time (\musec)')
title('Multiple Decay Rates')
set(H1,'LineStyle','--')
set(H2,'LineStyle',':')
```

运行结果



57. 如何为绘图加入网格？

参考 `grid` 函数。

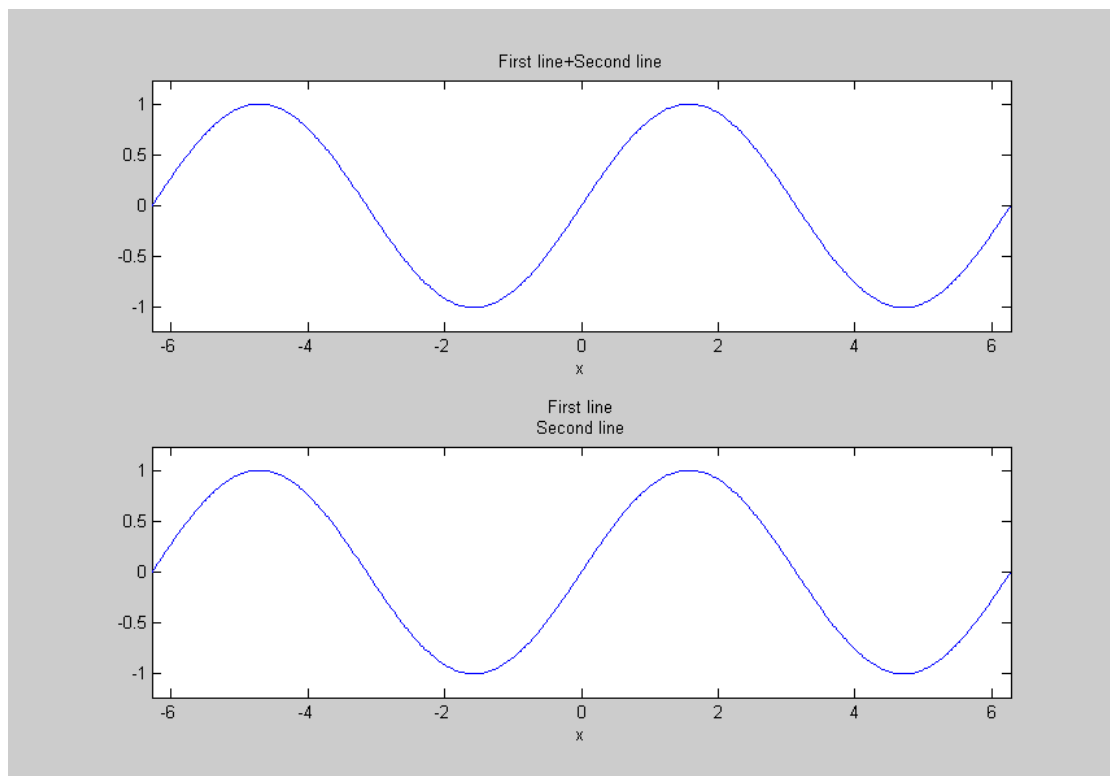
58. 如果绘图时标题太长，如何换行？

使用元胞数组可以输出多行的标题，见下例

```
figure
subplot(2,1,1);
ezplot( @sin );
title('First line+Second line');

subplot(2,1,2);
ezplot( @sin );
str = {'First line';'Second line'};
title( str );
```

运行结果



59. 如何做雷达图？

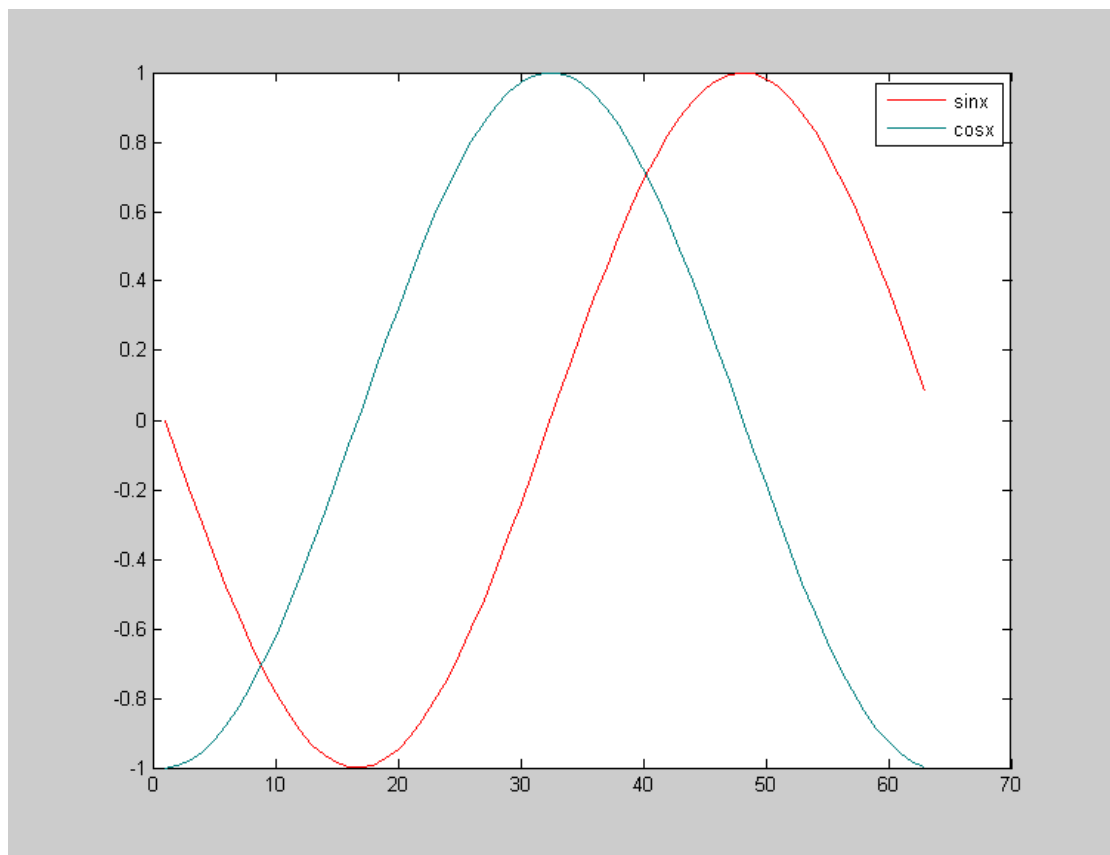
参考 polar 函数。

60. 如何用不同的颜色来代表数据？

可以通过调整 Color 选项参数[R, G, B]，来调整数据绘制的颜色，见下例

```
figure;
x = -pi:1:pi;
y = sin(x);
z = cos(x);
RGB = [1 0 0];
plot( y, 'Color', RGB )
hold on;
RGB = [0 0.5 0.5];
plot( z, 'Color', RGB )
legend('sinx', 'cosx');
```

运行结果



61. 如何调整所绘图形的大小？

初始化图形窗口时，可以通过调整 Position 参数选项[left, bottom, width, height]，来控制图形窗口的位置和大小，见下例

```
left = 100;
bottom = 100;
width = 500;
height = 500;
figure('Position', [left, bottom, width, height]);
ezplot(@sin)
```

62. 如何绘制三维图？

参考 plot3 函数。

63. 如何绘制椭圆或双曲线？

根据函数式进行基本绘图，直角坐标系下可使用参数方程：

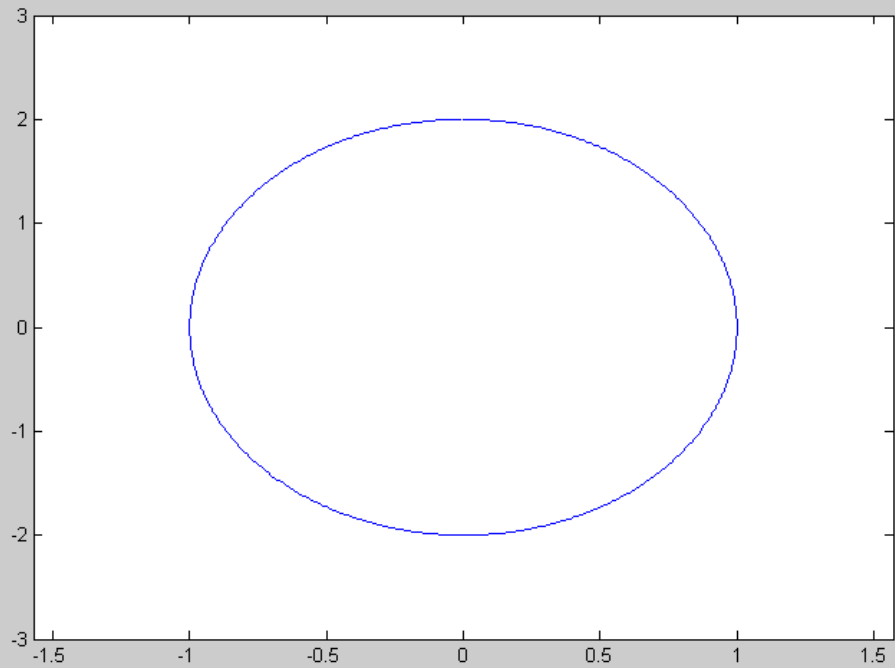
$$\left(\frac{x}{a}\right)^2 + \left(\frac{y}{b}\right)^2 = 1 \Rightarrow x = a \sin \theta, y = b \cos \theta, 0 < \theta < 2\pi$$

实例见下

```
t = 0:0.01:2*pi;
a = 1;
b = 2;
x = a*sin(t);
```

```
y = b*cos(t);  
figure;  
plot(x,y);  
xlim( [-pi/2, pi/2] );  
ylim( [-3, 3] );
```

运行结果

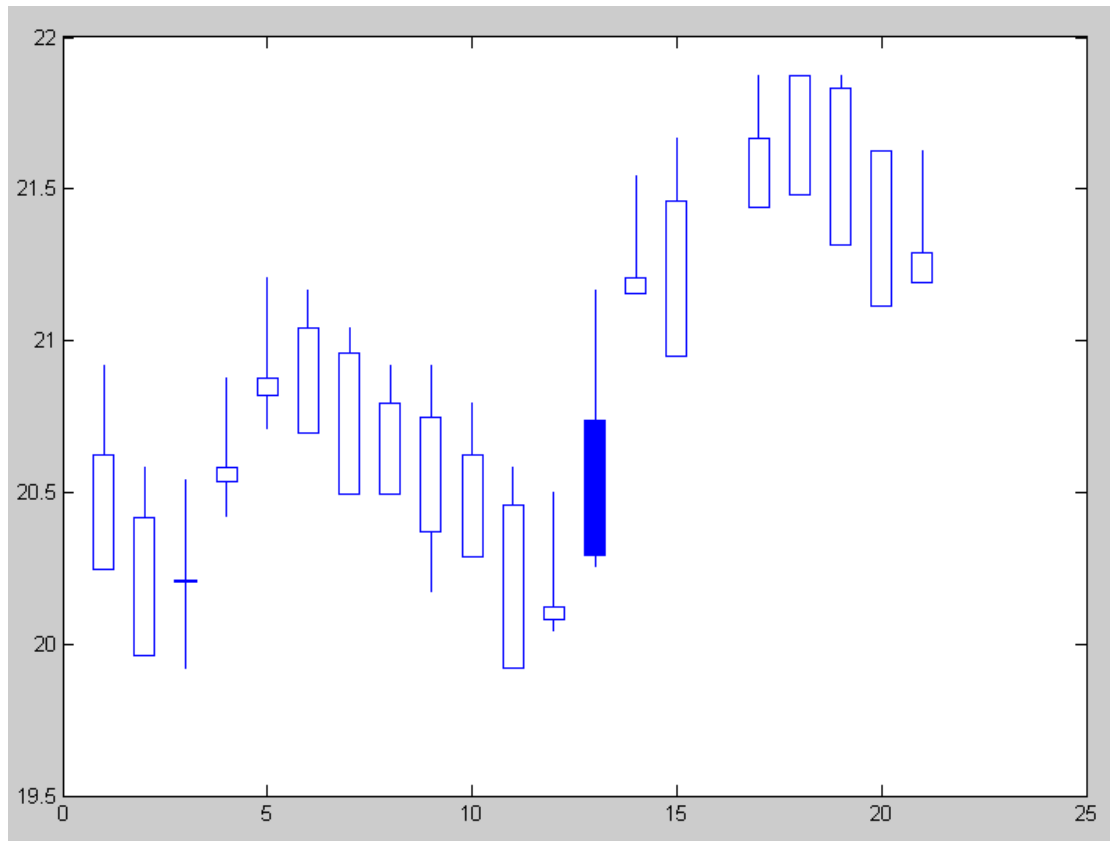


64. 在 MATLAB 中如何绘制 K 线图?

在 MATLAB 中可以使用 `candle` 函数来绘制 K 线，见下例

```
load disney;  
candle(dis_HIGH(end-20:end), dis_LOW(end-20:end), dis_CLOSE(end-20:end),...  
        dis_OPEN(end-20:end), 'b');
```

运行结果



65. 利用 MATLAB 可以制作图形用户界面（GUI）吗？

可以，像很多高级编程语言一样，MATLAB 也有图形用户界面开发环境，利用 MATLAB 建立图形用户界面（GUI）非常方便，其大体过程简要总结起来就是：拖拽放置组件，建立回调函数。想快速掌握 MATLAB GUI 编程可以参看下面这个帖子：

一个实例搞定 MATLAB 界面 GUI 编程

<http://www.matlabsky.com/thread-13055-1-1.html>

数学、金融、统计相关

66. 有没有直接计算峰度和偏度的函数？

使用 `skewness` 和 `kurtosis` 函数可以直接计算峰度和偏度。

67. 如何做线性回归模型？

使用 `lscov` 函数（或左除 “\” 运算）可以直接求解线性回归模型，`lscov` 函数的简要用法如下：

```
x = lscov(A,b)
```

`lscov` 函数的返回值 `x` 是线性系统 $A*x=b$ 的最小二乘解，即 $x=A \backslash b$ ，见下例

```
x1 = [.2 .5 .6 .8 1.0 1.1]';
```

```
x2 = [.1 .3 .4 .9 1.1 1.4]';
```

```
A = [ones(size(x1)) x1 x2]
```

```
b = [.17 .26 .28 .23 .27 .34]'
```

```
a1 = A\b
```

```
a2 = lscov(A, b)
```

运行结果

```
A =  
    1.0000    0.2000    0.1000  
    1.0000    0.5000    0.3000  
    1.0000    0.6000    0.4000  
    1.0000    0.8000    0.9000  
    1.0000    1.0000    1.1000  
    1.0000    1.1000    1.4000
```

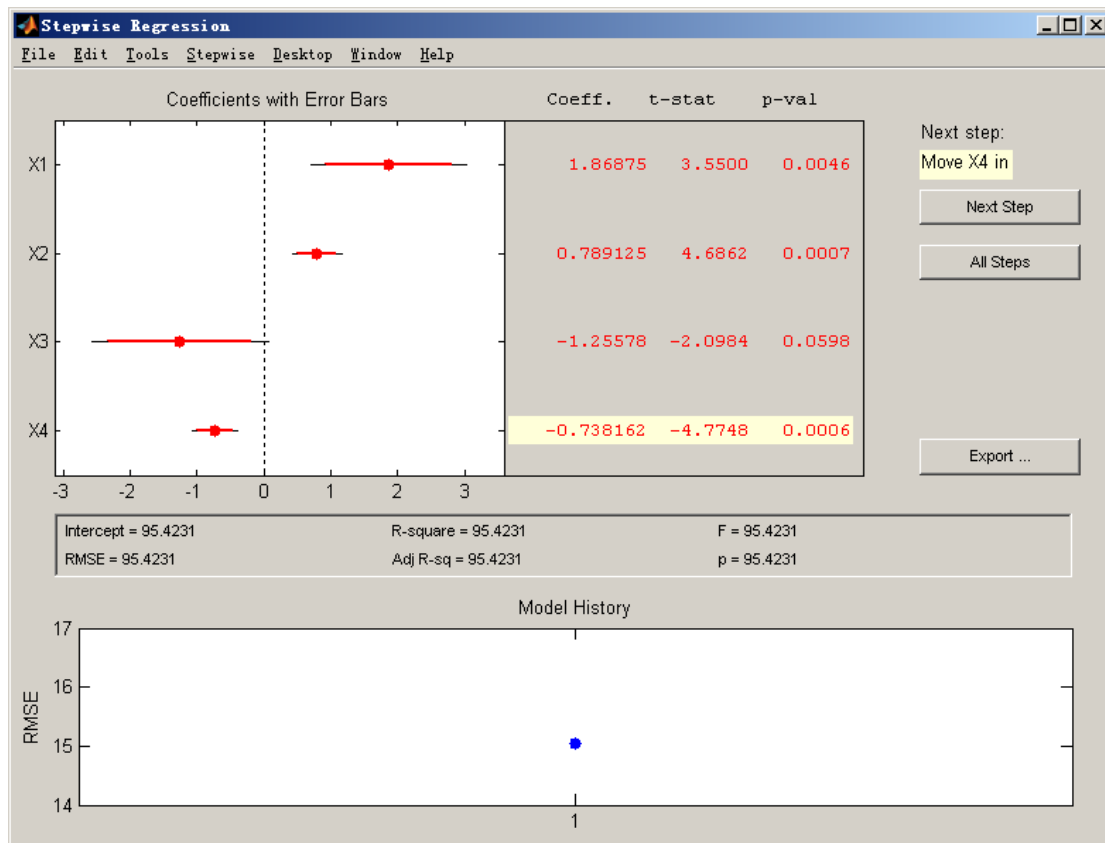
```
b =  
    0.1700  
    0.2600  
    0.2800  
    0.2300  
    0.2700  
    0.3400
```

```
a1 =  
    0.1203  
    0.3284  
   -0.1312
```

```
a2 =  
    0.1203  
    0.3284  
   -0.1312
```

68. 如何使用逐步回归？

参考 `StepwiseRegression` 函数，或者在 MATLAB 命令窗口（Command Window）键入 `stepwise` 开启逐步回归 GUI 界面，如下



69. 如何做聚类分析？

K 均值聚类算法可以用 `kmeans` 函数实现，具体算法过程在这里不详述，可以参考相关书籍，见下例

```
X = [randn(100,2)+ones(100,2);...
      randn(100,2)-ones(100,2)];
opts = statset('Display','final');

[idx,ctrs] = kmeans(X,2,...
                    'Distance','city',...
                    'Replicates',5,...
                    'Options',opts);

figure;
plot(X(idx==1,1),X(idx==1,2),'r.','MarkerSize',12)
hold on
plot(X(idx==2,1),X(idx==2,2),'b.','MarkerSize',12)
plot(ctrs(:,1),ctrs(:,2),'kx',...
      'MarkerSize',12,'LineWidth',2)
plot(ctrs(:,1),ctrs(:,2),'ko',...
      'MarkerSize',12,'LineWidth',2)
legend('Cluster 1','Cluster 2','Centroids',...
      'Location','NW')
```

运行结果

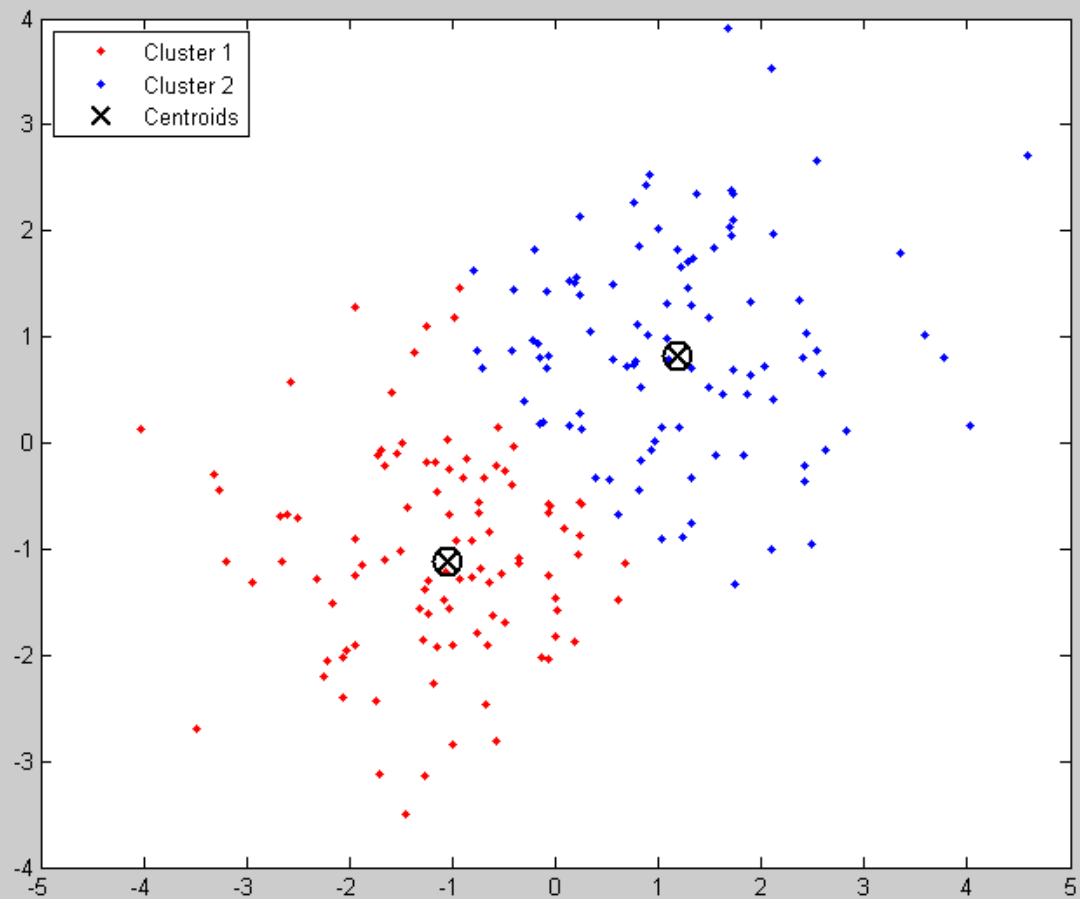
4 iterations, total sum of distances = 310.651.

9 iterations, total sum of distances = 310.651.

5 iterations, total sum of distances = 310.651.

6 iterations, total sum of distances = 310.651.

9 iterations, total sum of distances = 310.651.



模糊 C 均值聚类算法可以用 fcm 函数实现，其语法格式如下

```
[center, U, obj_fcn] = fcm(data, cluster_n, options)
```

用法：

1. [center,U,obj_fcn] = fcm(Data,N_cluster,options);
2. [center,U,obj_fcn] = fcm(Data,N_cluster);

输入变量

data ---- $n \times m$ 矩阵,表示 n 个样本,每个样本具有 m 维特征值

cluster_n ---- 标量,表示聚合中心数目,即类别数

options ---- 4×1 列向量, 其中

options(1): 隶属度矩阵 U 的指数, >1 (缺省值: 2.0)

options(2): 最大迭代次数(缺省值: 100)

options(3): 隶属度最小变化量,迭代终止条件(缺省值: $1e-5$)

options(4): 每次迭代是否输出信息标志(缺省值: 0)

输出变量

center ---- 聚类中心

U ---- 隶属度矩阵

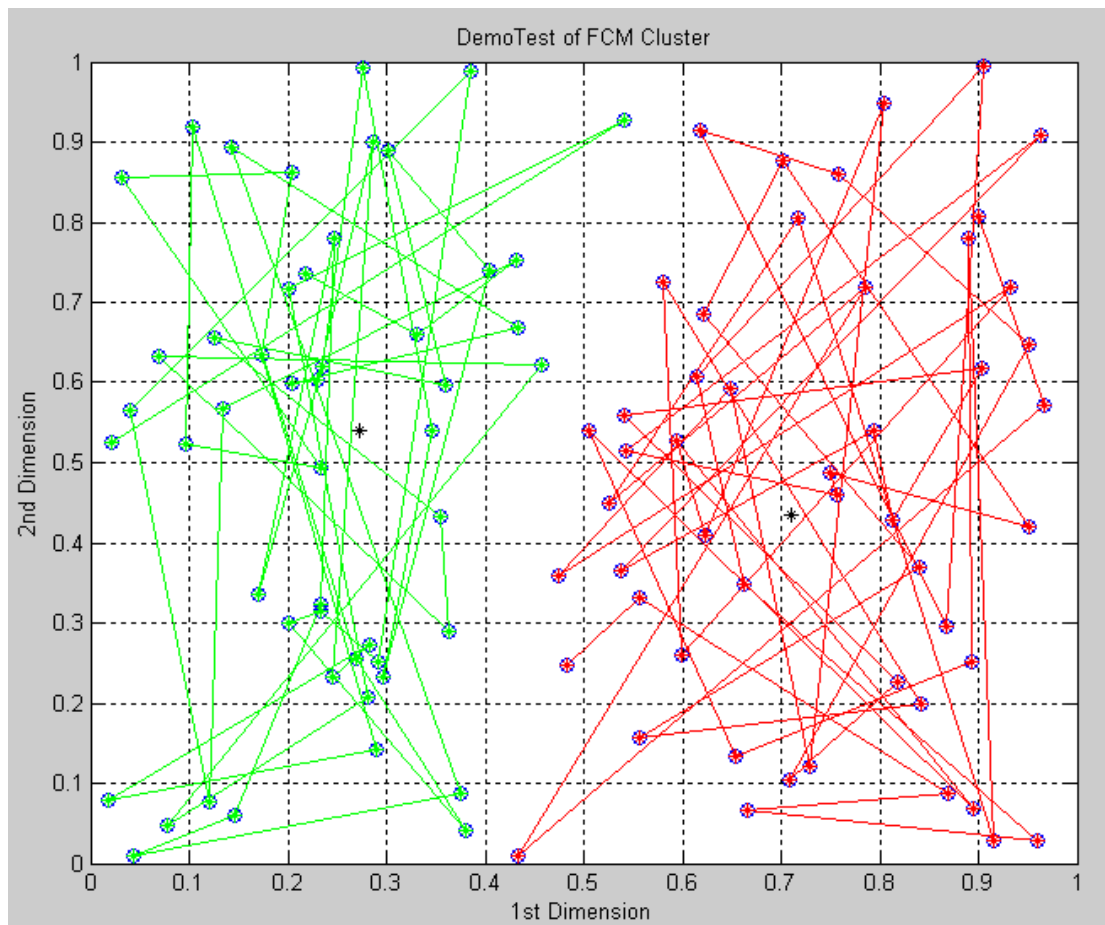
obj_fcn ---- 目标函数值

具体实例

```
figure;
data = rand(100,2);
options = [2;100;1e-5;1];
[center,U,obj_fcn] = fcm(data,2,options);
figure;
plot(data(:,1), data(:,2),'o');
title('DemoTest of FCM Cluster');
xlabel('1st Dimension');
ylabel('2nd Dimension');
grid on;
hold on;
maxU = max(U);
index1 = find(U(1,:) == maxU);
index2 = find(U(2,:) == maxU);
line(data(index1,1),data(index1,2),'marker','*','color','g');
line(data(index2,1),data(index2,2),'marker','*','color','r');
plot([center([1 2],1)],[center([1 2],2)],'*','color','k')
hold off;
```

运行结果

```
Iteration count = 1, obj. fcn = 10.952410
Iteration count = 2, obj. fcn = 8.426638
Iteration count = 3, obj. fcn = 8.275895
Iteration count = 4, obj. fcn = 7.879591
Iteration count = 5, obj. fcn = 7.361839
Iteration count = 6, obj. fcn = 7.095500
Iteration count = 7, obj. fcn = 7.014874
Iteration count = 8, obj. fcn = 6.982787
Iteration count = 9, obj. fcn = 6.965558
Iteration count = 10, obj. fcn = 6.955622
Iteration count = 11, obj. fcn = 6.949833
Iteration count = 12, obj. fcn = 6.946453
Iteration count = 13, obj. fcn = 6.944479
Iteration count = 14, obj. fcn = 6.943325
Iteration count = 15, obj. fcn = 6.942650
Iteration count = 16, obj. fcn = 6.942254
Iteration count = 17, obj. fcn = 6.942023
Iteration count = 18, obj. fcn = 6.941887
Iteration count = 19, obj. fcn = 6.941808
Iteration count = 20, obj. fcn = 6.941761
Iteration count = 21, obj. fcn = 6.941733
Iteration count = 22, obj. fcn = 6.941717
Iteration count = 23, obj. fcn = 6.941708
```



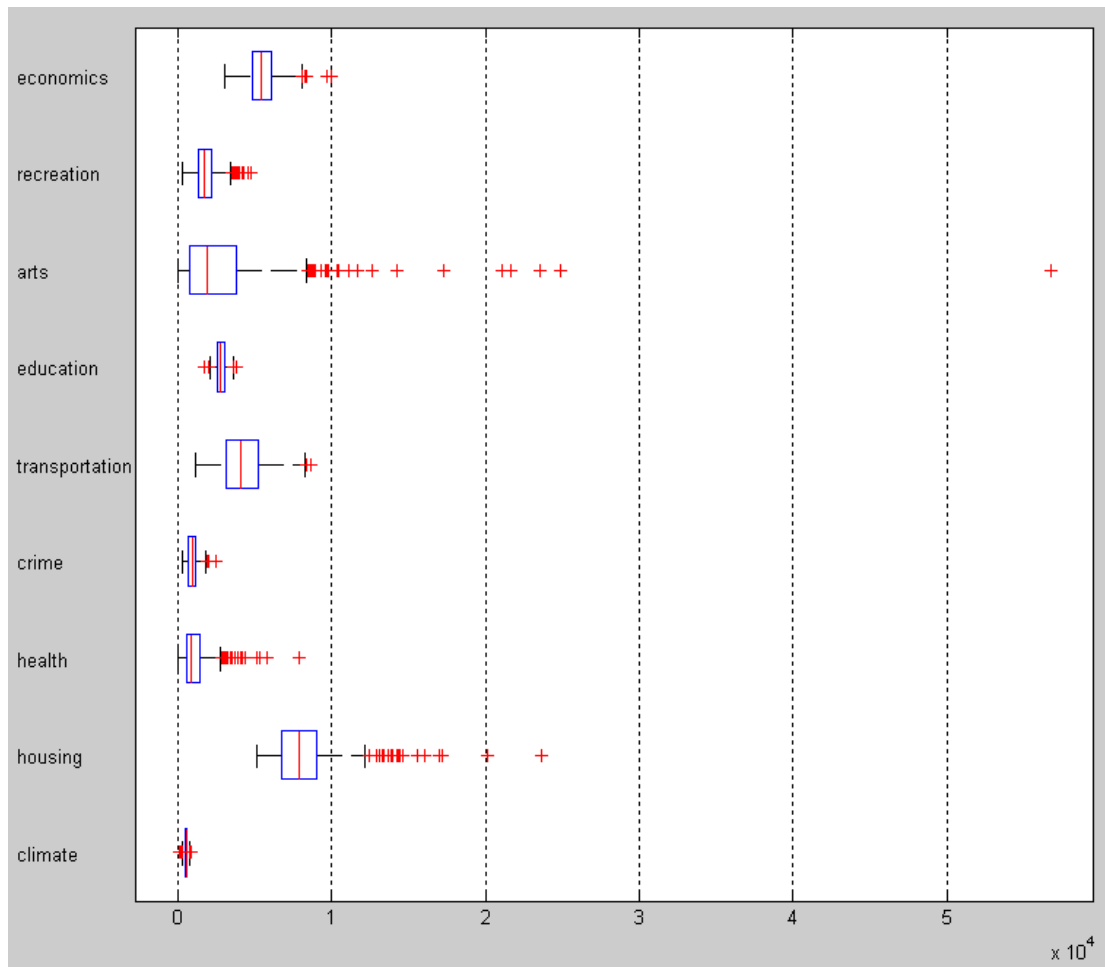
70. 如何做主成分分析?

`princomp` 函数可以进行主成分分析，具体过程可以参看下列

```
% 载入数据
load cities;

% whos
%   Name      Size      Bytes  Class
%   categories  9x14       252    char array
%   names      329x43    28294   char array
%   ratings    329x9       23688   double array

% % ratings 数据的 box 图
figure;
boxplot(ratings,'orientation','horizontal','labels',categories);
grid on;
```



这里使用的是 MATLAB 自带的城市（cities）数据，里面的 ratings 数据是一个 329*9 的矩阵，表示 329 个城市的数据，每个城市有 9 个属性来描述，上面是 ratings 数据的箱式图，下面使用 princomp 函数进行主成分分析。

```
% % 标准化预处理
stdr = std(ratings);
sr = ratings./repmat(stdr,329,1);
% % 使用 princomp 函数进行主成分分析
[coef,score,latent,t2] = princomp(sr);

% % 输出参数讲解

% coef:9*9 矩阵
% 主成分系数:即原始数据线性组合生成主成分数据中每一维数据前面的系数.
% coef 的每一列代表一个新生成的主成分的系数.
% 比如你想取出前三个主成分的系数，如下可实现: pca3 = coef(:,1:3);

% score:329*9 矩阵
% 字面理解:主成分得分
% 即原始数据在新生成的主成分空间里的坐标值.
```

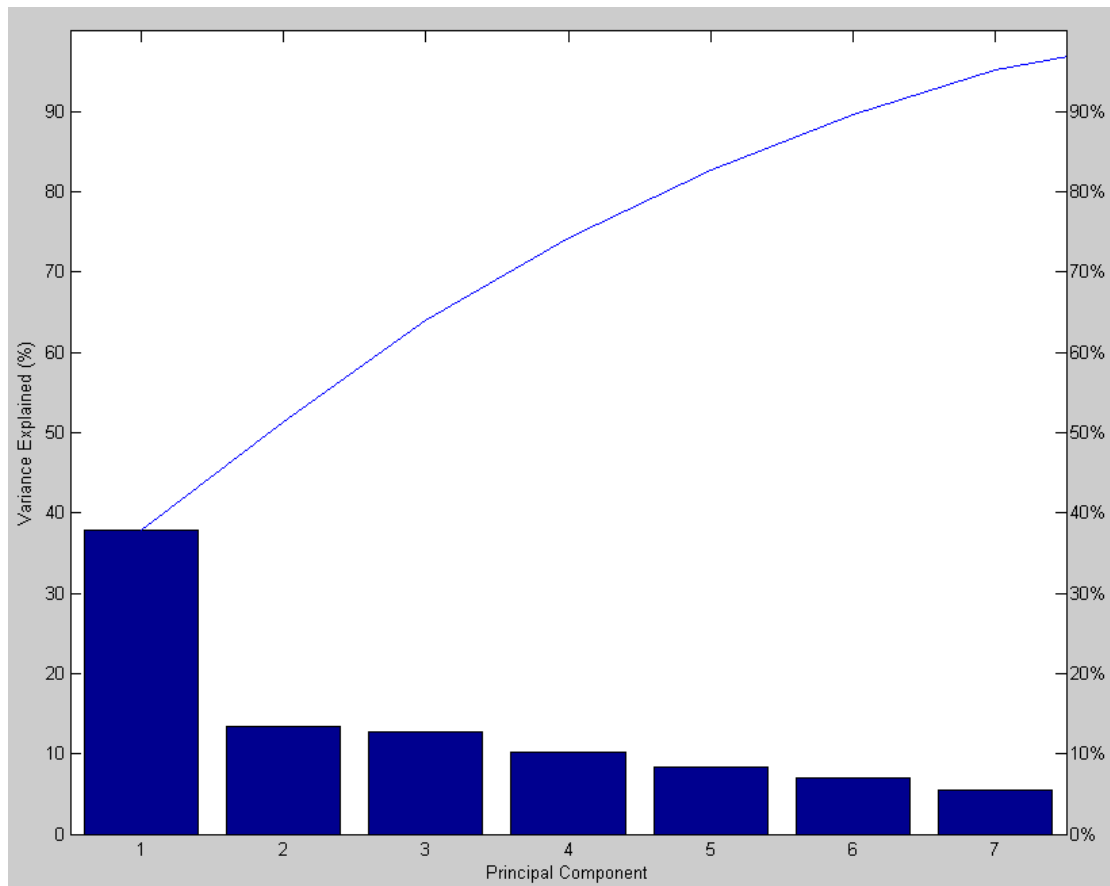
```

% latent:9*1 矩阵
% 一个列向量,由 sr 的协方差矩阵的特征值组成.
% 即 latent = sort(eig(cov(sr)), 'descend');
% 测试如下:
% sort(eig(cov(sr)), 'descend') =
%      3.4083
%      1.2140
%      1.1415
%      0.9209
%      0.7533
%      0.6306
%      0.4930
%      0.3180
%      0.1204
% latent =
%      3.4083
%      1.2140
%      1.1415
%      0.9209
%      0.7533
%      0.6306
%      0.4930
%      0.3180
%      0.1204

% t2:329*1
% 一中多元统计距离,记录的是每一个观察量到中心的距离

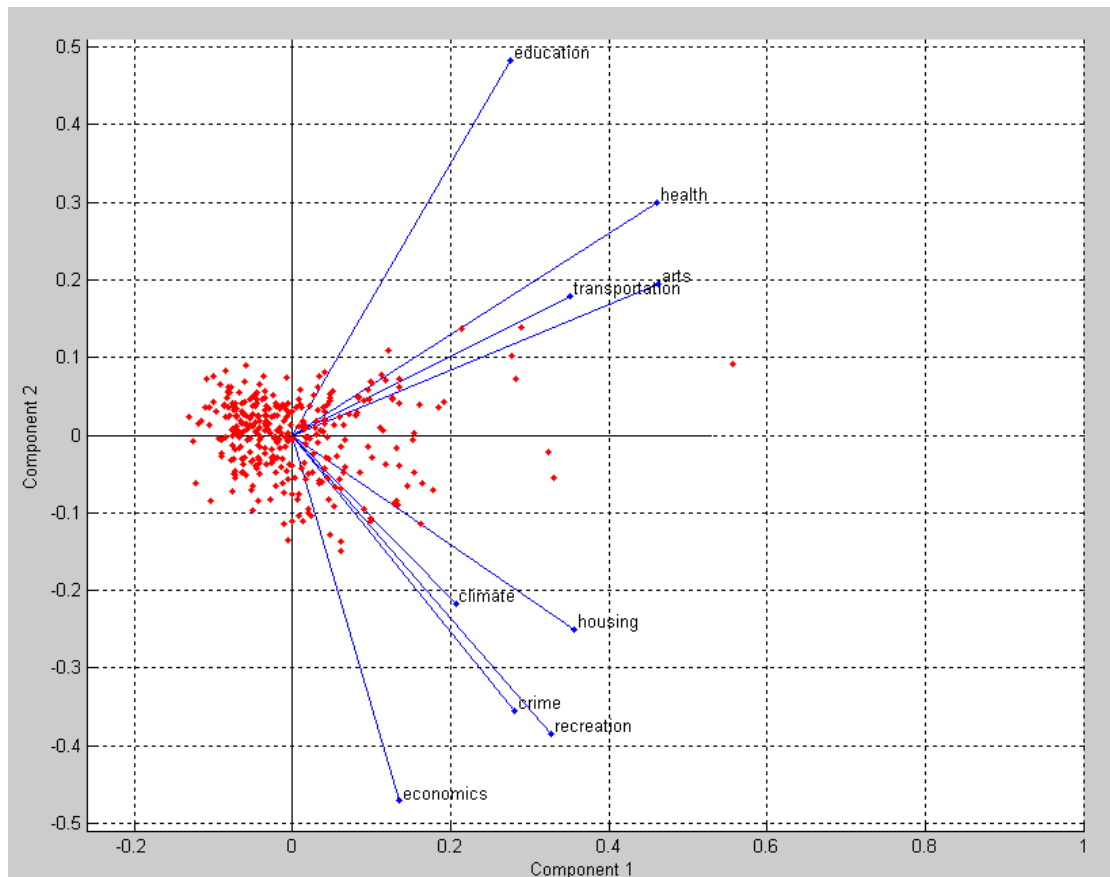
% % 如何提取主成分,达到降维的目的
% 通过 latent,可以知道提取前几个主成分就可以了.
figure;
percent_explained = 100*latent/sum(latent);
pareto(percent_explained);
xlabel('Principal Component');
ylabel('Variance Explained (%)');

```

上图中的线表示的累积变量解释程度,通过看图可以看出前七个主成分可以表示出原始数据的 90%,所以在 90%的意义下只需提取前七个主成分即可,进而达到主成分提取的目的。下面进行一些结果的可视化。

```
% % 结果的可视化
figure;
biplot(coef(:,1:2), 'scores',score(:,1:2),...
'varlabels',categories);
axis([-0.26 1 -0.51 0.51]);
% 横坐标和纵坐标分别表示第一主成分和第二主成分
% 红色的点代表 329 个观察量,其坐标就是 score
% 蓝色的向量的方向和长度表示了每个原始变量对新的主成分的贡献,其坐标就是 coef.
```



上图中横坐标和纵坐标分别表示第一主成分和第二主成分,红色的点代表329个观察量,其坐标就是 score, 蓝色的向量的方向和长度表示了每个原始变量对新的主成分的贡献,其坐标就是 coef。

71. 如何对样本数据进行正态检验?

参考 `jbtest` 函数, MATLAB 统计工具箱 (Statistics Toolbox) 中主要的分布检验函数有

`jbtest`: 正态性的 Jarque-Bera 检验

`kstest`: 单样本 Kolmogorov-Smirnov 检验

`kstest2`: 双样本 Kolmogorov-Smirnov 检验

`lillietest`: 正态性的 Lilliefors 检验

72. 如何做配对 t 检验?

参考 `ttest`、`ttest2` 函数, MATLAB 统计工具箱 (Statistics Toolbox) 中主要的假设检验函数有

`ranksum`: 秩和检验

`signrank`: 符号秩检验

`signtest`: 符号检验

`ttest`: 单样本 t 检验

`ttest2`: 双样本 t 检验

`ztest` : z 检验

73. 多项式回归应该用什么函数?

使用 `polyfit` 函数可以进行多项式拟合, 见下例

```

x = (-pi: 0.1: pi)';
y = sin(x);

p = polyfit(x,y,3)

f = polyval(p,x);

figure;
plot(x,y,'-', x,f,'ro');
legend('原始数据','多项式拟合数据');

```

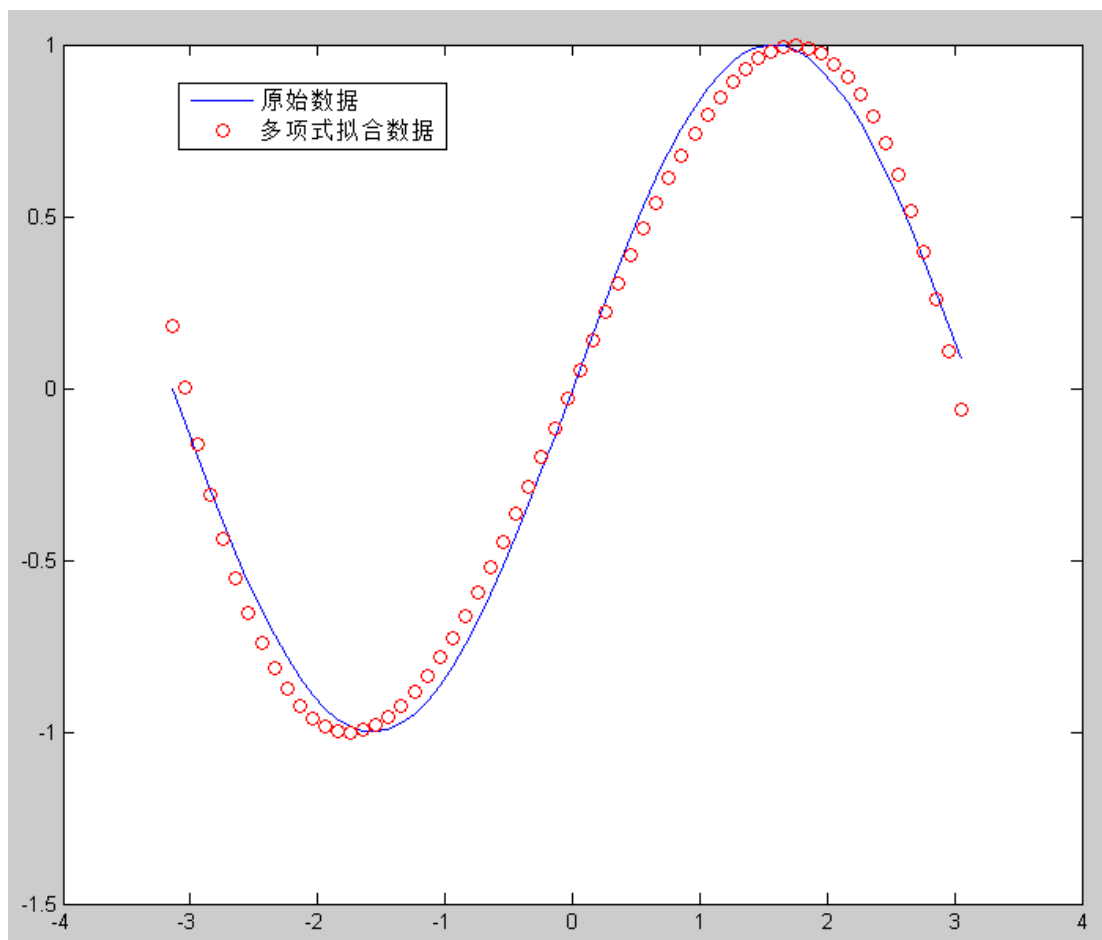
运行结果

```

p =
    -0.0931    -0.0020     0.8557     0.0041

```

即最后求出的拟合多项式为 $-0.0932 * x^3 - 0.002 * x^2 + 0.8557 * x + 0.0041$ ，拟合结果图形展示如下。



74. 如何使用方差分析（ANOVA）？

方差分析的目的是确定因素的不同处理（方法、变量）下，响应变量（类别、结果）的均值是否有显著性差异。MATLAB 提供了适用于单因素方差分析函数 `anova1` 以及双因素方差分析函数 `anova2` 函数。

75. logistic 回归相关函数是？

logistic 回归是关于响应变量为 0-1 定性变量的广义线性回归问题，在 MATLAB 中广义线性回归模型可以用 `glmfit` 函数实现，将广义线性回归模型的分布族设为二项分布（binomial）即可实现 logistic 回归，具体参看 `glmfit` 函数。

76. 如何做 Decision Tree？

决策树（Decision Tree）学习是以实例为基础的归纳学习算法。算法从一组无序的、无规则的事例中推理出决策树表示形式的分类规则，决策树也能表示为多个 If-Then 规则。一般在决策树中采用“自顶向下、分而治之”的递归方式，将搜索空间分为若干个互不相交的子集，在决策树的内部节点（非叶子节点）进行属性值的比较，并根据不同的属性值判断从该节点向下的分支，在树的叶节点得到结论。

MATLAB 统计工具箱（Statistics Toolbox）中的 `ClassificationTree.fit` 函数可实现基于训练集数据创建一个决策树分类器。

77. box-cox 变换？

参考 `boxcox` 函数。

78. 如何使用 MATLAB 做生存分析？

参考 `coxphfit` 函数。

79. 如何利用 MATLAB 获取免费的金融数据？

MATLAB 有一个数据获取工具箱（Datafeed Toolbox），Datafeed Toolbox 提供从主要的金融数据提供商处获取当前的、当天的、历史的和实时市场数据的工具。将金融数据域 MATLAB 集成起来，用户可开发实时模型反应当前金融市场行为。这个工具箱也提供导出 MATLAB 数据到数据服务提供商的函数。用户可建立从 MATLAB 到数据服务提供商提供的历史数据或者预定实时数据流的连接。证券查找对话框可以使工程师在 MATLAB 环境中查找指定证券的代码（仅针对 Bloomberg）。工程师可以使用证券的简要缩写码和代码来表示一个证券。

Database Toolbox 支持如下的金融数据服务提供商：

Bloomberg®

eSignal

Factset®

Federal Reserve Economic Data (FRED)

Haver Analytics

Interactive Data

Kx Systems®

Reuters® Market Data System

Thomson® Datastream

Yahoo!® Finance

其中雅虎财经（Yahoo Finance）的数据接口是完全免费的，你可以免费从雅虎财经下载全球的股票与期货数据进行建模测试，比如下载中国股市的数据，你只需知道相应股票的代码（在 Yahoo Finance 中上海证券交易所股票代码后需要加一个“.SS”，深圳证券交易所股票代码后需要加一个“.SZ”），见下例，获取三友化工(600409.SS 三友化工 Sanyou Chemical)

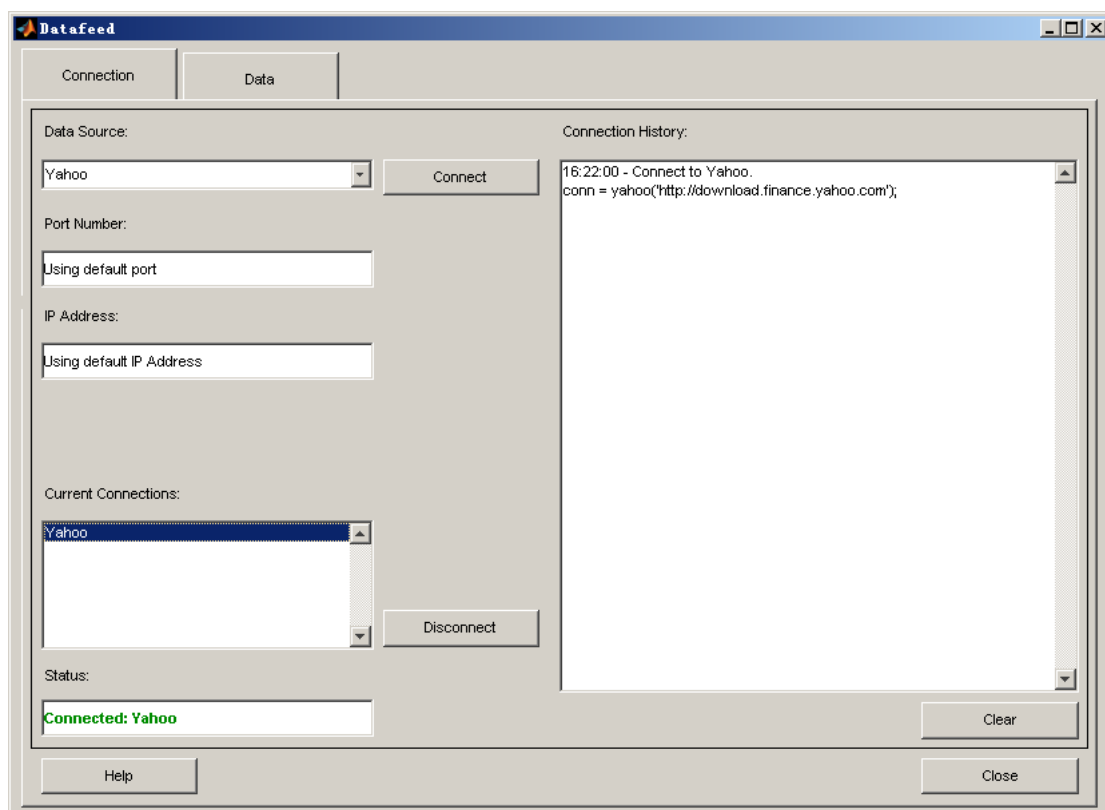
在某一时段的股票日收盘数据数据，

```
conn = yahoo;  
test = fetch(conn,{'600409.SS'},{'Close'},'08/01/13','08/08/13','d');  
  
d = fints(test(:,1), test(:,2), 'Close', 'D', '600409.SS_Close')  
  
close(yahoo)
```

运行结果

```
d =  
  
desc: 600409.SS_Close  
freq: Daily (1)  
  
'dates: (6)'  'Close: (6)'  
'01-Aug-2013' [ 4.2100]  
'02-Aug-2013' [ 4.2100]  
'05-Aug-2013' [ 4.3200]  
'06-Aug-2013' [ 4.3900]  
'07-Aug-2013' [ 4.3500]  
'08-Aug-2013' [ 4.3600]
```

上述操作也可以使用在 Datafeed Toolbox GUI 界面上完成，在 MATLAB 命令窗口（Command Window）键入 dftool 可以调出 Datafeed Toolbox GUI 界面，



80. 利用 MATLAB 可以实现支持向量机（SVM）模型吗？

较新版本的 MATLAB 中有 svmtrain、svmclassfity 等函数可以实现 SVM，但笔者更推荐使用 LIBSVM 工具箱，这个工具箱是个第三方工具箱，由台湾大学林智仁(Lin Chih-Jen)教授等开发设计，该工具箱用于实现 SVM 模型更加方便自由。

更多关于 SVM 的理论和应用方面的知识可以查看下面的免费视频链接进行学习：

支持向量机（SVM）相关免费学习视频集锦

<http://www.matlabsky.com/thread-36823-1-1.html>

81. 如何计算一个资金流的最大回撤？

根据最大回撤的定义：

设 Equity 为资金流，在索引（下表）Index 处，可以找出之前的资金流的最大值，设该最大值为 Max，如果 Index 处的资金流就为这个最大值，则 Index 处的最大回撤为 0，否则 Index 处的最大回撤 $(Equity(Index) - Max) / Max$ 。

下面的 RetraceRatio.m 函数可以实现最大回撤的计算。

```
function [DrawDownPercent,DrawDownAbs] = RetraceRatio(Equity)
% 计算最大回撤比例
% by liyang 2012/5/25
% farutoliyang@gmail.com
%% 初始化
len = numel(Equity);
DrawDownPercent = zeros(len, 1);
DrawDownAbs = zeros(len, 1);
%% 计算最大回撤比例
C = Equity(1);
for i = 2:len
    C = max(Equity(i),C);
    if C == Equity(i)
        DrawDownPercent(i) = 0;
    else
        DrawDownPercent(i) = (Equity(i)-C)/C;
    end
end
%% 计算最大回撤绝对数值
C = Equity(1);
for i = 2:len
    C = max(Equity(i),C);
    if C == Equity(i)
        DrawDownAbs(i) = 0;
    else
        DrawDownAbs(i) = (Equity(i)-C);
    end
end
end
```

下面使用该函数计算一下某一资金流（权益）的最大回撤，见下例

```
load FundMarketCash
fund = TestData(:,1);
```

```
[DrawDownPercent,DrawDownAbs] = RetraceRatio(fund);
```

```
figure;
```

```
subplot(3,1,1:2);
```

```
plot(fund);
```

```
xlim([1 length(fund)]);
```

```
title('Fund', 'FontWeight', 'Bold');
```

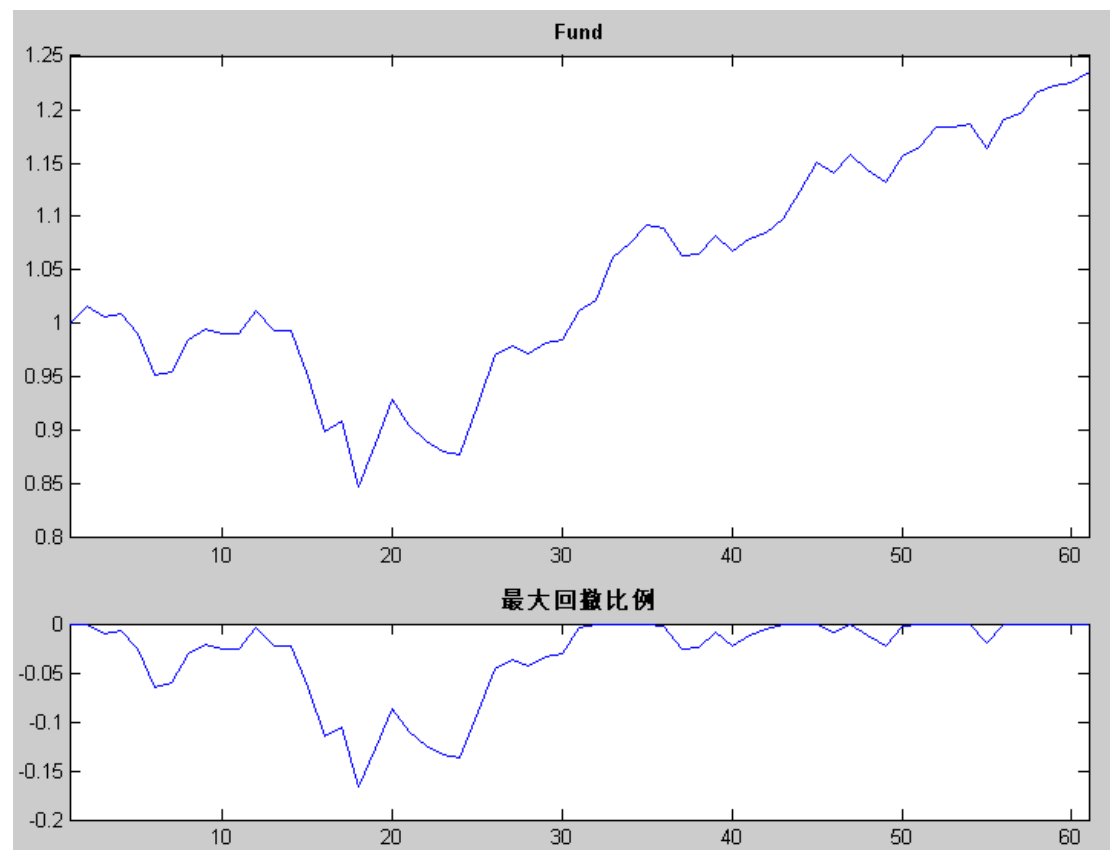
```
subplot(3,1,3);
```

```
plot( DrawDownPercent );
```

```
xlim([1 length(fund)]);
```

```
title('最大回撤比例', 'FontWeight', 'Bold');
```

运行结果



82.

其他

83. Cell 编程模式（Cell Mode）是什么？

在 MATLAB 文件编辑器（Editor）中，开启 Cell 编程模式（Cell Mode）后，可以使用两个分号“%%”将代码分块展示，也可以直接插入 HTML 代码和 Latex 代码等进行代码注释和讲解。

84. MATLAB 可以用网页显示结果么？

可以。在 MATLAB 文件编辑器（Editor）中，选择 File->Publish 可以直接生成代码以及测试结果的网页，结合 Cell 编程模式（Cell Mode）可以直接生成精美的网页方便代码以及测试结果的交流分享，下图为本文档的相应代码和测试结果生成的网页截图。

《量化投资：以MATLAB为工具》-基础篇-N分钟学会MATLAB(60<N<180)

Learn Matlab Using N minutes(60<N<180)

by LiYang

Email:farutoliyang@gmail.com

Last Updated 2013/08

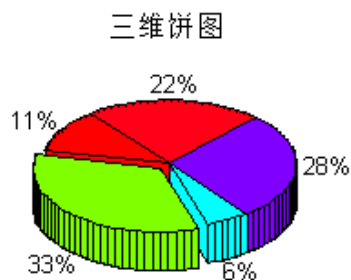
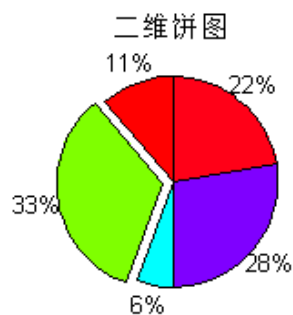
Contents

- [A Little Clean Work](#)
- [能否简单举一个MATLAB的例子？](#)
- [如何查看函数的代码？](#)
- [MATLAB里面可以使用科学计数法么？](#)
- [MATLAB可以读Excel的数据？](#)
- [如何删掉缺失值？](#)
- [如何将字符串转变为命令执行？](#)
- [如何向一个向量追加元素？](#)
- [如何移除矩阵的某行（列）数据？](#)
- [如何比较两个矩阵是否相同呢？](#)
- [如何去掉数据中的重复元素？](#)
- [如何求数据的极值？](#)
- [如何从一组数据中随机抽取数据？](#)
- [如何将数据标准化？](#)
- [如何计算积分？](#)
- [MATLAB如何进行复数计算？](#)
- [如何生成对角矩阵？](#)
- [如何构造上（下）三角矩阵？](#)

可以看到结合 Cell 编程模式（Cell Mode）生成网页时，MATLAB 会自动为你生成内容目录，并添加超链接，点击目录可以直接跳转到你关心的内容：

怎么做饼图？

```
x = [1 3 0.5 2.5 2];  
explode = [0 1 0 0 0];  
  
figure;  
subplot(2,1,1);  
pie(x,explode);  
title('二维饼图');  
colormap jet  
  
subplot(2,1,2);  
pie3(x,explode);  
title('三维饼图');  
colormap hsv
```



85. 怎样来计算函数运行使用时间？

参考 tic, toc 函数。

86. 在 MATLAB 中如何处理地图数据？

MATLAB 有一个地图工具箱（Mapping Toolbox）可以用来处理地图数据和进行图形展示，参加下面的简单例子。

```
% 世界地图  
figure;  
ax = worldmap('World');  
% setm(ax, 'Origin', [0 180 0])  
land = shaperead('landareas', 'UseGeoCoords', true);  
geoshow(ax, land, 'FaceColor', [0.5 0.7 0.5])  
lakes = shaperead('worldlakes', 'UseGeoCoords', true);  
geoshow(lakes, 'FaceColor', 'blue')
```

```

rivers = shaperead('worldrivers', 'UseGeoCoords', true);
geoshow(rivers, 'Color', 'blue')
cities = shaperead('worldcities', 'UseGeoCoords', true);
geoshow(cities, 'Marker', '.', 'Color', 'red')
title('世界地图',FontSize,14,FontWeight,'Bold');

% 中国地图
figure;
worldmap([15 55],[70 140])%纬度经度范围显示
%显示矢量数据
sh1 = shaperead('bou2_4p', 'UseGeoCoords', true);
geoshow(sh1, 'FaceColor', [0.5 1.0 0.5]);

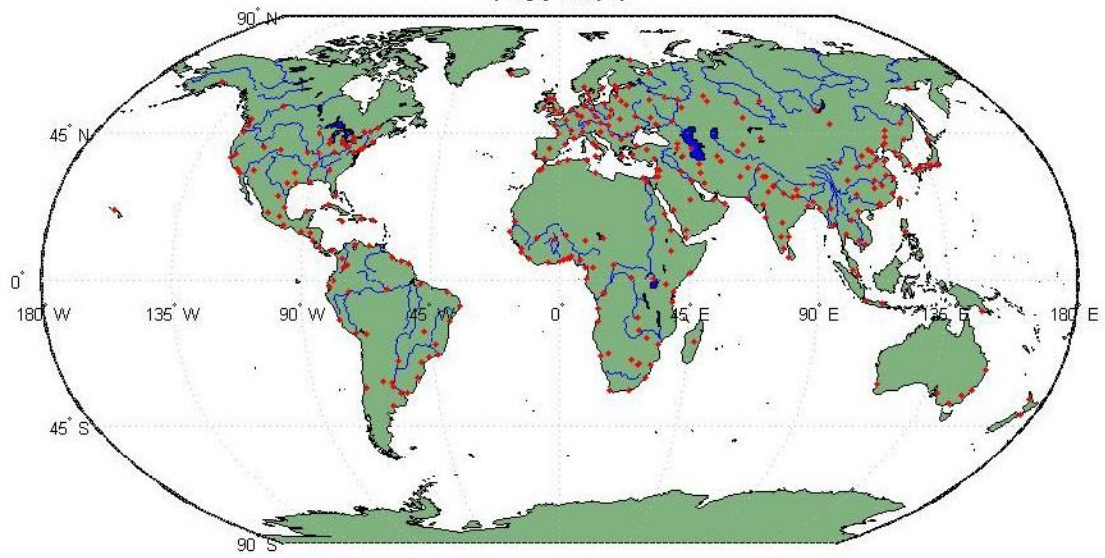
setm(gca,'MLineLocation',5)%设置经度间隔为 5
setm(gca,'PLineLocation',10)%设置经度间隔为 10
setm(gca,'MLabelLocation',5)%设置经度标签为每隔 5 度
setm(gca,'PLabelLocation',10)%设置纬度标签为每隔 10 度
title('中国地图',FontSize,14,FontWeight,'Bold');

% 美国地图
figure;
ax = worldmap('USA');
load coast
geoshow(ax, lat, long,...
'DisplayType', 'polygon', 'FaceColor', [.45 .60 .30])
states = shaperead('usastatelo', 'UseGeoCoords', true);
faceColors = makesymbolspec('Polygon',...
    {'INDEX', [1 numel(states)], 'FaceColor', ...
    polcmap(numel(states))}); % NOTE - colors are random
geoshow(ax, states, 'DisplayType', 'polygon', ...
'SymbolSpec', faceColors)
title('美国地图',FontSize,14,FontWeight,'Bold');

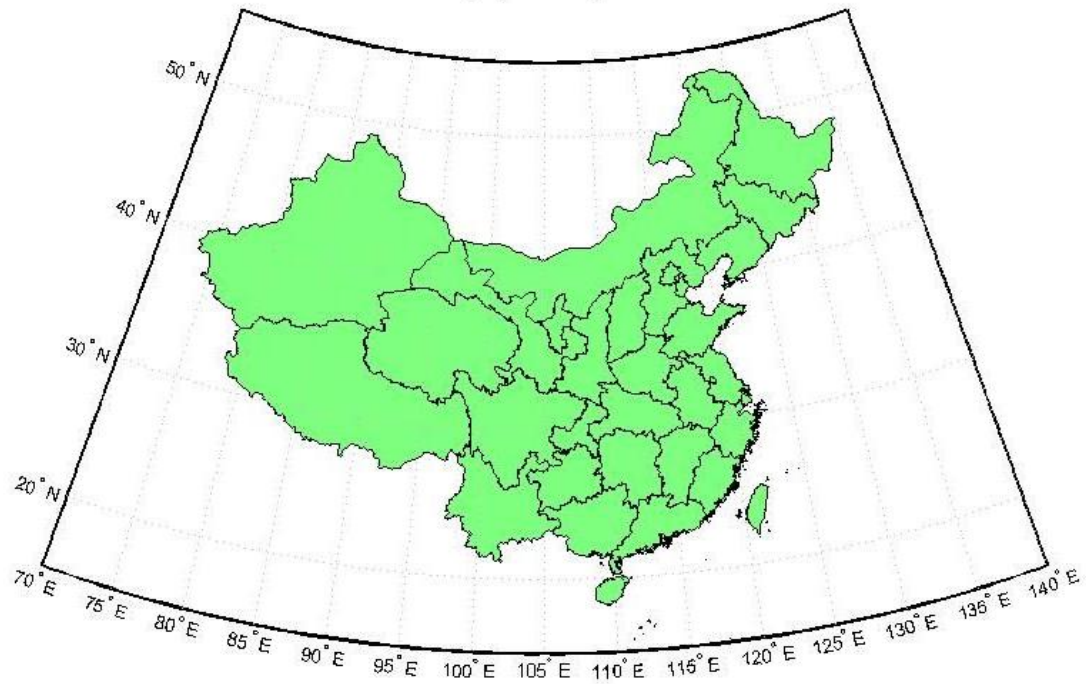
```

运行结果

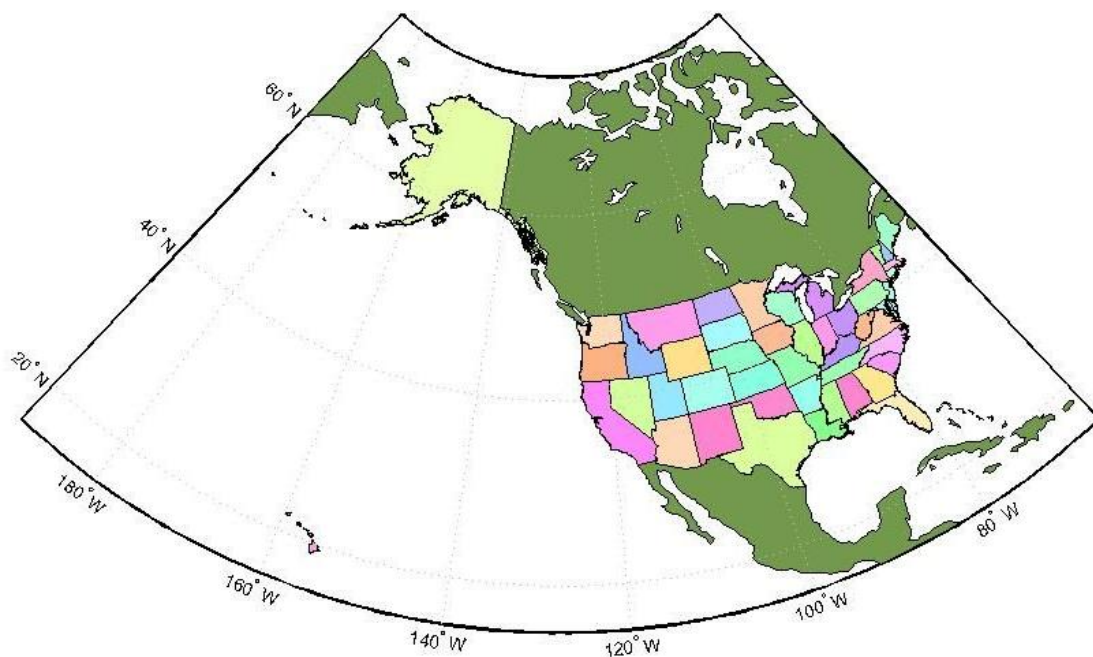
世界地图



中国地图



美国地图



87. MATLAB Notebook 是用来做什么的？

MATLAB Notebook 可以方便用户在 Word 环境中使用 MATLAB 强大的资源，为用户营造融文字处理、科学计算、工程设计于一体的完美工作环境。

MATLAB Notebook 制作的 M-book 文档不仅拥有 Word 的全部文字处理功能，而且具备 MATLAB 无与伦比的数学计算能力和灵活自如的计算结果可视化能力。

在 MATLAB 命令窗口（Command Window）中键入 `notebook` 可以启动 MATLAB Notebook。

88. 如何释放 MATLAB 运行后占用的内存？

参考 `pack` 函数。