

## 第 18 讲 MATLAB 的数据处理

司守奎

烟台市，海军航空大学

Email: sishoukui@163.com

MATLAB 数据类型有以下几种：数值类型，字符串，日期和时间，结构数组，细胞数组（元胞数组），函数句柄，Java 对象，逻辑类型等。

数值类型包括双精度浮点型，单精度浮点型，整型类型。下面我们简单介绍细胞数组与结构数组，并主要给出数值类型数据的操作。

### 18.1 细胞数组与结构数组

#### 18.1.1 细胞数组

细胞数组是 MATLAB 中的一类特殊的数组。在 MATLAB 中，由于有细胞数组这个数据类型，才能把不同类型、不同维数的数据组成为一个数组。

##### 1. 细胞数组的创建

细胞数组的创建有两种方法：通过赋值语句或 cell 函数创建。

(1) 使用赋值语句创建细胞数组：细胞数组使用花括号“{}”来创建，使用逗号“,”或空格分隔同一行的单元，使用分号“;”来分行。

(2) 使用 cell 函数创建空细胞数组。

**例18.1** 创建细胞数组。

```
clc, clear %以下构造4个细胞数组
```

```
A={'x', [2;3;6]; 10, 2*pi}
```

```
B=cell(2,2)
```

```
C={'LiMing','LiHong',{'0101','0102'}; {'f','m'}, {[90,80],[88,80]}}
```

```
fprintf('显示C{2, 2}{1}的元素: '), C{2,2}{1}
```

```
fprintf('显示C{2, 2}{2}的元素: '), C{2,2}{2}
```

```
fprintf('显示细胞数组C{2, 2}所有元素(两个元素)的值: '), C{2,2}{:}
```

```
fprintf('以下显示细胞数组C的全部元素:\n')
```

```
celldisp(C)
```

```
fprintf('细胞数组C的全部元素显示完毕!\n')
```

`D={A{:},B{:}}` %把A和B合并成一个细胞数组

## 2. 细胞数组的操作

MATLAB 提供的与细胞数组有关函数如表 18.1 所示.

表 18.1 细胞数组的有关函数

函 数	说 明
<code>cell2struct(cellArray,field,dim)</code>	将细胞数组转换成结构数组
<code>iscell(c)</code>	判断指定数组是否是细胞数组
<code>struct2cell(s)</code>	将 $m \times n$ 的结构数组 $s$ (带有 $p$ 个域) 转换成 $p \times m \times n$ 的细胞数组
<code>mat2cell(A,m,n)</code>	将矩阵拆分成细胞数组矩阵
<code>cell2mat(c)</code>	将细胞数组合并成矩阵
<code>num2cell(A)</code>	将数值数组转换成细胞数组
<code>celldisp(c)</code>	显示细胞数组内容
<code>cellplot(c)</code>	显示细胞数组结构图

**例 18.2** 细胞数组操作示例.

```
clc, clear
```

```
a={1, [2 3 4]; [5;9], [6 7 8; 10 11 12]} %建立细胞数组
```

```
b=cell2mat(a) %将细胞数组合并成矩阵,注意维数必须匹配,否则出错,见下面的转换
```

```
c={1,rand(2,3);rand(3),rand(4)}
```

```
try, d=cell2mat(c), catch, fprintf('维数不匹配, 无法转换! '),end
```

```
e=randi([1,9],4,5) %生成取值为{1, 2, ..., 9}的 4×5 随机整数矩阵
```

```
f1=mat2cell(e,[2,2],[3,2]) %将 e 拆分成 2×2 的细胞数组
```

```
f2={e} %将 e 转换成 1 个元素的细胞数组
```

```
subplot(121), cellplot(f1) %显示 f1 结构图
```

```
subplot(122), cellplot(f2) %显示 f2 结构图
```

**例 18.3** 已知 11 个地点的位置坐标如表 18.2 所示, 画出这 11 个点的位置并进行标注.

表 18.2 地点名称及坐标数据

位置名称	$x$ 坐标	$y$ 坐标	位置名称	$x$ 坐标	$y$ 坐标
基地 R	865	141	无名高地	690	131
基地 S	941	187	山谷 1	254	495
基地 T	711	841	山谷 11	736	443
101 高地	782	726	山谷 01	128	789
12 高地	769	385	山谷 001	349	816
116 高地	453	956			

把表 18.2 中的数据保存在 Excel 文件 data183.xlsx 中，执行如下命令

```
[a,b]=xlsread('data183.xlsx')
```

则返回值 a 读入的是数值矩阵，b 读入的字符串的细胞数组，但 a,b 中有些特殊数据要额外进行处理。

```
clc, clear
```

```
[a,b]=xlsread('data183.xlsx')
```

```
a=[a(:,[1,2]);a([1:end-1],[4,5])]; %提取需要的数据
```

```
b={b{:,1},b{[1:end-1],4}} %提取非空字符串，构造新的字符串细胞数组
```

```
plot(a([1:3],1),a([1:3],2),'P') %画前 3 个点
```

```
hold on, plot(a([4:end],1),a([4:end],2),'*') %画其余点
```

```
text(a(:,1)+5,a(:,2),b) %对所有的点进行标注
```

画出的 11 个地点的示意图见图 18.1。

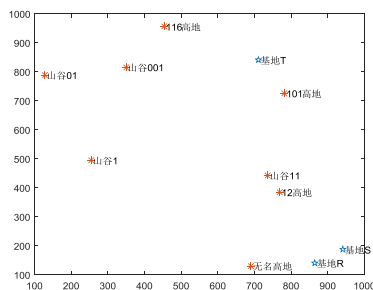


图 18.1 位置示意图

说明：MATLAB 中网络图的顶点字符串等，都要求是细胞数组字符串。

### 18.1.2 结构数组

有时需要将不同的数据类型组合成一个整体，以便于引用。这些组合在一个整体中的数据是相互联系的。例如，一个学生的学号、姓名、性别、年龄、成绩、家庭地址等项都是和该学生有联系的。

MATLAB 与其它高级语言一样具有结构类型的数据。结构类型是包含一组彼此相关、数据结构相同但类型不同的数据类型。结构类型的变量可以是任意一种 MATLAB 数据类型的变量，也可以是一维的、二维的或者多维的。但是，在访问结构类型数据的元素时，需要使用下标配合域名的形式。

#### 1. 结构数组的建立

MATLAB 提供两种方法建立结构数组，用户可以直接给结构数组成员变量赋值建

立结构数组，也可以利用函数 `struct` 建立结构数组。

**例 18.4** 利用赋值建立结构数组。

```
clc, clear  
  
stu(1).name='LiMing'; stu(1).number='0101';  
stu(1).sex='f'; stu(1).score=[90,80];  
stu(2).name='LiHong'; stu(2).number='0102';  
stu(2).sex='m'; stu(2).score=[88,80];  
  
stu %显示结构数组的结构  
  
stu(1) %显示结构数组第1个元素  
stu(2) %显示结构数组第2个元素
```

**例18.5** 利用函数`struct`建立结构数组，并对两个结构数组进行连接。

```
clc, clear  
  
stu=struct('name',{'LiMing','LiHong'},'number',{'0101','0102'},...  
    'sex',{'f','m'},'score',[90,80],[88,80]))  
  
stu2=[stu,stu] %对两个结构数组进行连接  
  
name1=stu2(4).name %提取结构数组stu2的第4个元素的name域的值  
name2={stu2.name} %提取结构数组stu2的name域的所有取值，构成细胞数组
```

## 2.结构数组的操作

MATLAB 提供的与结构数组有关函数如表 18.3 所示。

表 18.3 结构数组的有关函数

函数名	作用
<code>struct</code>	生成结构数组
<code>fieldnames(s)</code>	获取指定结构数组所有域名
<code>getfield(s,'field')</code>	获取指定域的值
<code>isfield(s,'field')</code>	判断是否是指定结构数组中的域
<code>orderfields(s)</code>	对结构数组域名按首字符重新排序
<code>setfield(s,'field',value)</code>	设置结构数组指定域的值
<code>rmfield(s,'field')</code>	删除指定结构数组中的域
<code>isstruct</code>	检查数组是否为结构类型

**例 18.6** 结构数组的操作。

```
clc, clear  
  
stu=struct('name',{'LiMing','LiHong'},'number',{'0101','0102'},...  
    'sex',{'f','m'},'score',[90,80],[88,80]))
```

```

a=fieldnames(stu) %获取 stu 所有域名
b1=getfield(stu,'number') %获取第 1 个元素的 number 域的值
b2=getfield(stu,{2},'number') %获取第 2 个元素的 number 域的值
c=isfield(stu,'sex') %判断 sex 是否为 stu 中的域
stunew=orderfields(stu) %按结构数组域名首字符重新排序
rmfield(stu,'sex') %删除 sex 域

```

**例 18.7** 结构数组和细胞数组的相互转化.

```

clc, clear
s_cell={'LiMing','0101','M',20;'ZhangSan','0102','F',30} %建立细胞数组
fields={'name','number','sex','age'} %建立细胞数组
s_struct=cell2struct(s_cell,fields,2) %沿第 2 维方向, 将细胞数组转换成结构数组
t_struct=struct('name',{'LiMing','LiHong'},'number',{'0101','0102'},...
    'sex',{'f','m'},'age',{20,19}) %构造结构数组
t_cell=struct2cell(t_struct) %将 4 个域的结构数组转换成 4×1×2 的细胞数组

```

**MATLAB 命令**

```
f=dir('*.m')
```

可以显示当前目录下所有后缀名为 m 的文件信息, 返回值 f 是一个结构数组, 包括 5 个域: name, date, bytes, isdir, datenum; 通过结构数组的元素个数就可以知道当前目录下 m 文件的个数, 通过 name 域可以知道当前目录下所有 m 文件的名称.

dir 命令可以读出所有类型文件的信息.

**例 18.8** 读入当前目录下所有的后缀名为 bmp 的图片文件, 并把数据保存在细胞数组中.

```

clc, clear
f=dir('*.bmp'); %读入当前目录下所有 bmp 图像文件的信息, 保存在结构数组 f 中
n=length(f) %计算 bmp 文件的个数
for i=1:n
    a[i]=imread(f(i).name); %把第 i 个 bmp 图像数据保存在细胞数组中的第 i 个元素中
end

```

一些 MATLAB 函数的返回值也是结构数组. MATLAB 求微分方程数值解的命令 ode45 也可以返回结构数组, ode45 用龙格库塔方法求一阶微分方程或微分方程组的数值解.

对一阶微分方程或方程组的初值问题

$$\begin{cases} y' = f(t, y), \\ y(t_0) = y_0, \end{cases}$$

其中  $y$  和  $f$  可以为向量. 函数ode45有如下两种调用格式:

$$[t,y]=ode45(fun,tspan,y0) \text{ 或 } s=ode45(fun,tspan,y0)$$

其中 fun 是用 M 函数或匿名函数定义的  $f(t, y)$ , tspan=[ $t_0$ ,tfinal](这里  $t_0$  必须是初值条件中自变量的取值, tfinal 可以比  $t_0$  小)是求解区间, y0 是初值. 返回值 t 是 MATLAB 自动离散化的区间[ $t_0$ ,tfinal]上的点, y 的列是对应于 t 的函数值; 如果只有一个返回值 s, 则 s 是一个结构数组.

利用结构数组 s 和 MATLAB 函数 deval, 我们可以计算任何感兴趣点 x 的函数值, 调用格式为

$$y=deval(s,x)$$

其中 x 为标量或向量, 返回值 y 的行是对应于 x 的数值解.

**例18.9** 求  $x^2y'' + xy' + (x^2 - \frac{1}{4})y = 0, y(\frac{\pi}{2}) = 2, y'(\frac{\pi}{2}) = -\frac{2}{\pi}$  在区间  $[\frac{\pi}{2}, 8]$  上步长间隔为0.01的数值解.

若用MATLAB工具箱求数值解, 微分方程(或方程组)必须是一阶的, 首先做变量替换, 把上述二阶微分方程化成一阶方程组, 令  $y_1 = y, y_2 = y'$ , 则有

$$\begin{cases} y_1' = y_2, & y_1(\frac{\pi}{2}) = 2, \\ y_2' = -\frac{y_2}{x} - (1 - \frac{1}{4x^2})y_1, & y_2(\frac{\pi}{2}) = -\frac{2}{\pi}. \end{cases}$$

求解的MATLAB程序如下:

```
clc, clear
```

```
dy=@(x,y)[y(2);-y(2)/x-(1-1/4/x^2)*y(1)];
```

[x1,y1]=ode45(dy,[pi/2,8],[2;-2/pi]) % 返回值 y1 是两列的矩阵,y1(:,1) 对应 y 的值,y1(:,2)对应y的导数值, 所求的数值解不符合题目的要求

```
s=ode45(dy,[pi/2,8],[2;-2/pi])
```

```
x2=pi/2:0.01:8;
```

```
y2=deval(s,x2) %返回值y2是两行的矩阵, y2(1,:)对应y的值, y2(2,:)对应y的导数值
```

```
subplot(121),plot(x1,y1(:,1)),subplot(122),plot(x2,y2(1,:)) %两种格式比较
```

## 18.2 文件

根据数据的组织形式, MATLAB 中的文件可分为 ASCII 文件和二进制文件. ASCII 文件又称文本文件,它的每一个字节存放一个 ASCII 代码,代表一个字符. 二进制文件是把内存中的数据按其内存中的存储形式原样输出到磁盘上存放.

MATLAB 中的关于文件方面的函数和 C 语言相似, 见表 18.4.

表 18.4 MATLAB 的文件操作命令

函数分类	函数名	作用
打开和关闭文件	fopen	打开文件
	fclose	关闭文件
读写二进制文件	fread	读二进制文件
	fwrite	写二进制文件
格式 I/O	fscanf	从文件中读格式数据
	fprintf	写格式数据
	fgetl	从文件中读行,不返回行结束符
	fgets	从文件中读行,返回行结束符
读写字符串	sprintf	把格式数据写入字符串
	sscanf	格式读入字符串
文件定位	feof	检验是否为文件结尾
	fseek	设置文件定位器
	ftell	获取文件定位器
	frewind	返回到文件的开头

### 18.2.1 文件的打开和关闭

对文件读写之前应该“打开”该文件, 在使用结束之后应“关闭”该文件.

函数 fopen 用于打开文件, 其调用格式为:

```
fid=fopen(filename,permission)
```

fid 是文件标识符 (file identifier), fopen 指令执行成功后就会返回一个正的 fid 值, 如果 fopen 指令执行失败, fid 就返回-1.

filename 是文件名.

permission 是文件允许操作的类型, 可设为以下几个值:

‘r’ 只读

‘w’ 只写

‘a’ 追加 (append)

‘r+’ 可读可写

与 `fopen` 对应的指令为 `fclose`，它用于关闭文件，其指令格式为：

```
status=fclose(fid)
```

如果成功关闭文件，`status` 返回的值就是 0.

说明：一定要养成好的编程习惯，文件操作完之后，要关闭文件，即释放文件句柄，如果不关闭句柄，还占用内存空间，如果打开的文件数量太多，内存会溢出的。

## 18.2.2 文件的读写操作

### 1. 二进制文件的操作

写二进制文件的函数 `fwrite`，其调用格式为

```
fwrite(fileID, A, precision)
```

其中 `fileID` 为文件句柄，`A` 是要写入文件的数组，`precision` 控制所写数据的精度，默认的是无符号整数“`uint8`”格式。

读二进制文件的函数 `fread`，其调用格式为

```
[A,count]=fread(fileID,sizeA,precision)
```

其中，`A` 是用于存放读取数据的矩阵，`count` 是返回所读取的数据元素个数；`fileID` 是文件句柄；`size` 为可选项，若不选用则读取整个文件内容，若选用则它的值可以是下列值：`N`（读取 `N` 个元素到一个列向量）、`inf`（读取整个文件）、`[m,n]`（读取数据到 `m×n` 矩阵中，数据按列存放）；`precision` 控制所读数据的精度。

**例 18.10** 把向量`[1,2,...,9]`写入二进制文件，然后再读出来。

```
clc, clear
```

```
f1=fopen('nine.bin','w') %新建二进制文件 nine.bin，返回句柄 f1
```

```
fwrite(f1,[1:9]) %写入整数 1, 2, ..., 9
```

```
flag=fclose(f1) %关闭文件
```

```
f2=fopen('nine.bin','r') %以只读的格式打开已经建立的二进制文件
```

```
A=fread(f2) %读取二进制文件中的全部数据
```

### 2. 文本文件的操作

把数据写入文本文件的函数为 `fprintf`，其调用格式为

```
fprintf(fileID, format, A1,...,An)
```



其中 fileID 为文件句柄，A1，...，An 是要写入文件的数组，format 控制所写数据的格式。

```
fprintf(format, A1,...,An)
```

把数组 A1，...，An 以 format 指定的格式显示在屏幕上。

从文本文件读数据的函数为 fscanf，其调用格式为

```
A=fscanf(fileID,formatSpec,sizeA)
```

该函数从句柄 fileID 所指向的文本文件，按照指定的格式 formatSpec，读入 SizeA 大小的数据，赋给数组 A。

**例 18.11** 把温度数据字符串'78°F,72°F,64°F,66°F,49°F,50°F'写入文本文件 temperature.dat 中，然后再读取其中的数值数据。

```
clc, clear
str='78°F,72°F,64°F,66°F,49°F,50°F';
fid1=fopen('temperature.dat','w'); %新建 temperature.dat 文件
fprintf(fid1,'%s',str); %向文件 temperature.dat 中写入字符串数据
fclose(fid1);
fid2=fopen('temperature.dat','r'); %以只读的格式打开 temperature.dat 文件
data=fscanf(fid2,'%d°F,') %读取其中的数值数据
```

**例 18.12** 把数据加上表头，以表格的形式写到纯文本文件中，然后再读取其中的数值数据。

```
clc, clear
x=0:0.1:1; A=[x;exp(x)];
fid1= fopen('exp.txt','w'); %新建纯文本文件 exp.txt
fprintf(fid1,'%6s %12s\r\n','x',exp(x)); %\r\n 表示新换一行
fprintf(fid1,'%6.2f %12.8f\r\n',A); %把矩阵 A 写入 exp.txt 中
fclose(fid1); %关闭文件句柄
fid2=fopen('exp.txt','r') %以只读的方式打开文本文件 exp.txt
fgetl(fid2) %读出第一行的字符串
a=fscanf(fid2,'%f%f\n',[2,inf]) %读入文件数据的行，赋给矩阵 A 的对应列
%下面给出另一种读入数据的方式
a=importdata('exp.txt'); %读入文本文件中的全部数据
```

```
b=a.data %提出数值数据
```

## 18.3 各种数据文件的操作

### 18.3.1 MATLAB 的 mat 文件

**例 18.13** 把 MATLAB 工作空间中的数据矩阵 a, b, c 保存到数据文件 data1813.mat 中.

```
clc, clear  
a=rand(3); b=randi([1,10],4,5); c='str';  
save data1813 a b c
```

**例 18.14** 把例 18.13 生成的 data1813.mat 中的所有数据加载到 MATLAB 工作空间中, 并显示当前工作空间中的所有变量.

```
clc, clear  
load data1813  
whos
```

注: MATLAB 中的默认数据文件 mat 文件可以省略后缀名.

**例 18.15** 把矩阵 a, b 保存到数据文件 example.mat 中, 然后重新加载到工作空间中, 并计算矩阵 a 的各列均值.

```
clc, clear  
a=rand(5);  
b=magic(4);  
save example.mat a b %把 a,b 矩阵保存到 mat 文件中  
clear %清理内存空间  
c=matfile('example.mat') %把 mat 文件加载到工作空间  
d=size(c,'a') %读 a 矩阵的维数  
am=mean(c.a) %计算 a 矩阵的每列均值
```

通过上面例子可以看出把 mat 文件加载到工作空间, 既可以使用命令 load, 也可以使用命令 matfile.

### 18.3.2 纯文本文件

## 1.读入纯文本文件

可以把 word 文档中整行整列的数据粘贴到纯文本文件，然后调入到 MATLAB 工作空间中。

**例 18.16** 纯文本文件 data1816 中存放如下格式的数据，把数据读入 MATLAB 中。

```
6   2   6   7   4   2   5
4   9   5   3   8   5   8
5   2   1   9   7   4   3
7   6   7   3   9   2   7
```

纯文本文件 data1816.txt 是用记事本把上述数据保存到 MATLAB 的当前工作路径下。以下所有操作的数据文件必须放在 MATLAB 的当前工作路径下，也就是说程序文件和数据文件要放在同一个目录下。

MATLAB 加载纯文本文件的调用格式如下：

```
a=load('data1816.txt')
```

或者是

```
a=textread('data1816.txt')
```

**例 18.17** 纯文本文件 data1817.txt 中存放如下格式的数据，把其中的数据读入 MATLAB 中。

从产地到销地的单位运价表

	B1	B2	B3	B4	B5	B6	B7	B8	产量
A1	6	2	6	7	4	2	5	9	60
A2	4	9	5	3	8	5	8	2	55
A3	5	2	1	9	7	4	3	3	51
A4	7	6	7	3	9	2	7	1	43
A5	2	3	9	5	7	2	6	5	41
A6	5	5	2	2	8	1	4	3	52
销量	35	37	22	32	41	32	43	38	

MATLAB 程序如下：

```
a=importdata('data1817.txt') %返回值 a 是结构数组
```

```
b=a.data %提出其中的数据
```

```
b(isnan(b))=0 %把其中的不确定值替换成 0
```

**例 18.18** 纯文本文件 data1818.txt 存放如下数据，读入其中的前两行前两列交叉位

置的 4 个元素.

```
6  2  6  7  4  2  5  9
4  9  5  3  8  5  8  2
5  2  1  9  7  4  3  3
7  6  7  3  9  2  7  1
2  3  9  5  7  2  6  5
5  5  2  2  8  1  4  3
```

MATLAB 命令为:

```
a=dlmread('data1818.txt','A1..B2') %这里"表示数据之间的分隔符是空格
```

### 例 18.19 混合数据的读入

纯文本文件 data1819.txt 存放如下数据, 读入其中的数据.

日期	开盘	最高	最低	收盘	交易量	交易额
2007/06/04	33.76	33.99	31.00	32.44	282444.00	921965312.00
2007/06/05	31.90	33.00	29.20	32.79	329276.00	1032631552.00
2007/06/06	31.90	32.86	31.00	32.27	236677.00	756290880.00
2007/06/07	32.41	34.00	32.16	32.73	255289.00	845447232.00
2007/06/08	32.70	32.70	31.18	31.60	272817.00	862057728.00

MATLAB 程序如下:

```
clc, clear
```

```
fid = fopen('data1819.txt');
```

```
fgetl(fid) %读第1行的表头
```

```
A=textscan(fid, '%s %f %f %f %f %f %f', 'CollectOutput', true) %读数据, A为1×2的细胞
```

数组

```
fclose(fid);
```

```
B=A{1,2} %提取需要的数据矩阵
```

### 例 18.20 混合数据的读入

纯文本文件 data1820.txt 存放如下数据, 读入其中的数据.

```
Sally 09/12/2005 12.34 45 Yes
```

```
Larry 10/12/2005 34.56 54 Yes
```

```
Tommy 11/12/2005 67.89 23 No
```

MATLAB 程序如下:

```

clc, clear
fid=fopen('data1820.txt');
A1=textscan(fid, '%s %s %f %d %s') %不合并细胞数组相邻同类型数据
frewind(fid) %移动文件位置指示器到文件头，重新读入数据
A2=textscan(fid, '%s %s %f %d %s', 'CollectOutput', true) %合并细胞数组相邻同类型数据
B1=A2{1,2}, B2=A2{1,3} %提取需要的数值数据
fclose(fid);

```

**例 18.21** 使用 `textscan` 处理大的纯文本文件，假设用户自定义的数据处理函数为 `process_data`.

MATLAB 程序如下：

```

clc, clear
block_size=10000;
format='%s %n %s %8.2f %8.2f %8.2f %8.2f %u8';
fid=fopen('largefile.txt');
while ~feof(fid)
    segarray=textscan(fid, format, block_size);
    process_data(segarray);
end
fclose(fid);

```

注 18.1 我们没有构造大数据文件 `largefile.txt` 和函数 `process_data`，当然程序是无法运行的，我们这里只是说明编程的思路。

**例 18.22** 输入非长方形的纯文本数据。

纯文本文件 `data1822.txt` 中存放如下数据：

```

begin
v1=12.67
v2=3.14
v3=6.778
end
begin
v1=21.78

```

```
v2=5.24
v3=9.838
end
```

MATLAB 读入数值数据的程序如下：

```
clc, clear
fid = fopen('data1822.txt');
c=textscan(fid, '%*s v1=%f v2=%f v3=%f %*s', 'Delimiter', '\n', 'CollectOutput', true)
c=cell2mat(c) %把细胞数组转换成数值矩阵
fclose(fid);
```

注：这里的“\*”表示跳过一个字符串的数据域。

## 2.数据写入纯文本文件

**例 18.23** 使用 save 命令把矩阵 a 保存到纯文本文件 data1823.txt 中。

```
a=randi([1,10],50);
save data1823.txt a -ascii
```

**例 18.24** 使用 dlmwrite 命令把矩阵 b 保存到纯文本文件 data1824.txt。

```
b=rand(5)
dlmwrite('data1824.txt',b)
```

**例 18.25** 生成服从标准正态分布随机数的  $100 \times 200$  矩阵，然后用 fprintf 命令保存到纯文本文件 data1825.txt。

```
clc, clear
fid=fopen('data1825.txt','w');
a=normrnd(0,1,100,200);
fprintf(fid,'%f\n',a);
fclose(fid);
```

注：对于高维矩阵，用 dlmwrite 构造的纯文本文件，Lingo 软件不识别；为了 Lingo 软件识别，纯文本文件必须用 fprintf 构造，而且数据之间的分割符为“\n”。另外，MATLAB 数据是逐列存储的，Lingo 数据是逐行存储的，这里把 a 矩阵的转置矩阵 a' 写入纯文本文件供 Lingo 调用。

### 18.3.3 Excel 文件

#### 1. 读入数据

MATLAB 读入 Excel 文件的命令是 `xlsread`，使用格式为

```
num=xlsread(filename,sheet,Range)
```

```
[num,txt]=xlsread(filename,sheet,Range)
```

其中第 1 个返回值是数值矩阵，第 2 个返回值是字符串的细胞数组，`sheet` 是表单序号，`Range` 是数据域的范围。

**例 18.26** 把 Excel 文件 `data1826.xls` 的表单 `Sheet1` 的域“A2:D5”中的数据赋给 `a`，表单 `Sheet2` 中的全部数据赋给 `b`。

```
a=xlsread('data1826.xls',1,'A2:D5')
```

```
b=xlsread('data1826.xls',2)
```

注 18.2 要运行该程序，MATLAB 的工作路径下必须存在文件 `data1826.xls`，下面我们就一一说明了。

把 Excel 文件的所有表单数据全部读入的命令是 `importdata`。

**例 18.27** 把 Excel 文件 `data1827.xls` 中的所有表单数据全部读入。

```
mydata=importdata('data1827.xls')
```

返回值 `mydata` 是一结构数组。

#### 2. 数据写入 Excel 文件

MATLAB 把数据写入 Excel 文件的命令是 `xlswrite`，使用格式为

```
xlswrite(filename,A,sheet, Range)
```

其中 `filename` 是要写入数据的文件名，`A` 是要写入的矩阵，`sheet` 是表单名，`Range` 是数据域的地址或数据域的左上角地址。

**例 18.28** 把一个  $5 \times 10$  矩阵 `a` 写到 Excel 文件 `data1828.xls` 表单 `Sheet1` 中，把一个  $3 \times 3$  矩阵 `b` 写入表单 `Sheet2` 的 B2 开始的域中。

```
a=rand(5,10);
```

```
xlswrite('data1828.xls',a) %默认写入第 1 个表单，A1 开始的数据域中
```

```
b=rand(3);
```

```
xlswrite('data1828.xls',b,'Sheet2','B2') %这里'Sheet2'可以替换成 2
```

### 18.3.4 字符串数据

**例 18.29** 统计下列五行字符串中字符 a、c、g、t 出现的频数.

1.aggcacggaaaaacgggaataacggaggaggacttggcacggcattacacggagg

2.cggaggacaaacgggatggcggtattggaggtggcgggactgttcgggga

3.gggacgggatacggattctggccacggacggaaaggaggacacggcggacataca

4.atggataacggaaacaaaccagacaaacttcggtagaaatacagaagctta

5.cggctggcggacaacggactggcggattccaaaaacggaggaggcggacggaggc

把上述五行复制到一个纯文本数据文件 data1829\_1.txt 中，编写如下程序：

```
clc, clear
```

```
fid=fopen('data1829_1.txt','r');
```

```
i=1;
```

```
while (~feof(fid))
```

```
data=fgetl(fid);
```

```
a=length(find(data==97));
```

```
b=length(find(data==99));
```

```
c=length(find(data==103));
```

```
d=length(find(data==116));
```

```
e=length(find(data>=97&data<=122));
```

```
f(i,:)= [a b c d e a+b+c+d];
```

```
i=i+1;
```

```
end
```

```
f, he=sum(f)
```

```
dlimwrite('data1829_2.txt',f); dlimwrite('data1829_2.txt',he,'-append'); %统计结果写入
```

纯文本文件中

```
fclose(fid);
```

**例 18.30** 某计算机机房的一台计算机经常出故障，研究者每隔 15 分钟观察一次计算机的运行状态，收集了 24 小时的数据（共作 97 次观察）。用 1 表示正常状态，用 0 表示不正常状态，所得的数据序列如下：

111001001111111001111011111100111111110001101101

11101101101011110111011111011111001101111100111

求在 96 次状态转移中，“ $0 \rightarrow 0$ ， $0 \rightarrow 1$ ， $1 \rightarrow 0$ ， $1 \rightarrow 1$ ”各有几次？

把上述数据序列保存到纯文本文件 data1830.txt 中，存放在 MATLAB 的当前工作



目录下. 编写程序如下:

```
clc,clear
fid=fopen('data1830.txt','r');
a=[];
while (~feof(fid))
    a=[a fgetl(fid)]; %把所有读入的字符组成一个大字符串
end
for i=0:1
    for j=0:1
        s=[int2str(i),int2str(j)]; %构造查找的子字符串
        f(i+1,j+1)=length(findstr(s,a));
    end
end
f %显示统计矩阵
```

求得 96 次状态转移的情况是:

0 → 0, 8 次;      0 → 1, 18 次;  
1 → 0, 18 次;      1 → 1, 52 次.

**例18.31** 有纯文本文件data1831.txt, 保存一些英文字符, 统计其中“an”出现的次数, 其中data1831.txt中的数据如下

Oranges and lemons,  
Pineapples and tea.  
Orangutans and monkeys,  
Dragonflies or fleas.

MATLAB 程序如下

```
clc, clear
fid=fopen('data1831.txt');
str0='an'; str=[];
while (~feof(fid))
    str=[str fgetl(fid)]; %把所有读入的字符组成一个大字符串
end
L=length(findstr(str,str0))
```

```
fclose(fid);
```

### 18.3.5 图像文件

**例 18.32** 把一个比较大的 bmp 图像文件 data1832.bmp, 转化成比较小的 jpg 文件, 命名成 data1832.jpg, 并显示.

```
a=imread('data1832.bmp'); %非工具箱图像文件, 我们自己的一个图像文件
subplot(121), imshow(a), title('原 bmp 图像')%显示原 bmp 图像
imwrite(a,'data1832.jpg');
subplot(122), imshow('data1832.jpg'), title('新 jpg 图像') %对比显示新的 jpg 图像
```

**例 18.33** 生成 10 幅彩色 jpg 文件, 依次命名成 jpg1.jpg, ..., jpg10.jpg.

```
clc, clear
for i=1:10
    str=['jpg',int2str(i),'.jpg'];
    a(:, :, 1)=rand(500); a(:, :, 2)=rand(500)+100; a(:, :, 3)=rand(500)+200;
    imwrite(a,str);
end
```

### 4.3.6 时间序列数据

**例 18.34** 时间序列数据的处理.

```
randn('seed',sum(100*clock)); %初始化随机数发生器
a=randn(6,1);
b=[today:today+5]' %从今天到后面 5 天
fts=fints(b,a) %生长 fints 格式数据
fts(3)=NaN; %将第 3 个数据变为缺失值 NaN
newdata=fillts(fts,'linear') %用线性插值填补时间序列中的缺失数据
data=fts2mat(newdata) %把时间序列数据转换成矩阵
```

**例 18.35** 对于 MATLAB 当前工作路径下的所有纯文本文件, 进行相关的数据操作.

```
tf=dir('*.txt') %提出纯文本文件的信息, 返回值是结构数据
n=length(tf); %计算纯文本文件的个数
fts=ascii2fts(tf(1).name); %读第一个文件中的时间序列数据
```

```

fts=extfield(fts,{ 'series2','series3'}); %提出第 2 和第 3 字段
for i=2:n
    tp1=ascii2fts(tf(i).name); %读入新的时间序列数据
    tp2=extfield(tp1,{ 'series2','series3'}); %提出第 2 和第 3 字段
    str1=['series',num2str(2*i)]; str2=['series',num2str(2*i+1)];
    tp3=fints(tp2.dates,fts2mat(tp2),{str1,str2}); %把时间序列字段改名,同名字段的
两个时间序列数据是无法合并的
    fts=merge(fts,tp3); %合并两个时间序列的数据
end

```

注 18.3 上面程序是使用我们自己的特定数据，这里只是说明编程思路。

### 18.3.7 日期和时间

MATLAB 日期和时间的函数有 `datenum`, `datevec`, `datestr`, `now`, `clock`, `date`, `calendar`, `eomday`, `weekday`, `addtodate`, `etime` 等，这里就不一一说明各个函数的用法了，下面举一个小例子说明有关函数的使用。

**例 18.36** 统计 1601 年 1 月到 2000 年 12 月，每月的 13 日分别出现在星期日、星期一、星期二，...，星期六的频数，并画出对应的柱状图。

注：MATLAB 中 `weekday` 的 1 对应“星期日”，2 对应“星期一”，...，7 对应“星期六”。

所画出的柱状图见图 18.2，计算及画图的 MATLAB 程序如下：

```

clc, clear
c=zeros(1,7);
for y=1601:2000
    for m=1:12
        d=datenum(y,m,13); %计算日期对应的天数，公元0年1月1日是第1天
        w=weekday(d);
        c(w)=c(w)+1;
    end
end
c, bar(c) %显示频数并画出频数的柱状图
axis([0 8 680 690])

```

```
line([0,8],[4800/7,4800/7],'linewidth',4,'color','k')
set(gca,'xticklabel',{'Su','M','Tu','W','Th','F','Sa'})
```

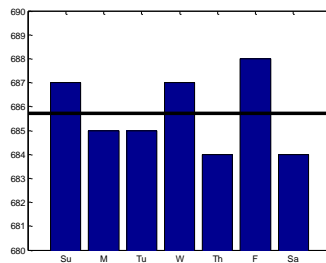


图 18.2 频数分布的柱状图

### 18.3.8 视频文件

MATLAB 除了支持各种图像文件的读写等操作，还支持视频文件的相应处理。实际上，视频文件本质上是由多帧具有一定大小、顺序、格式的图像组成的，只是一般的图像是静止的，而视频是可以将多帧静止的图像进行连续显示，从而达到动态效果。

**例 18.37** 读取一个视频文件 test.avi，并把视频中的每一帧保存成 jpg 文件。

```
clc,clear
ob=VideoReader('test.avi')    %读取视频文件对象
n=ob.NumberOfFrame; %获取视频的总帧数
for i=1:n
a=read(ob,i); %读取视频对象的第i帧
imshow(a) %显示第i帧图像
str=['source\'int2str(i)'.jpg']; %构造文件名的字符串，目录source要提前建好
imwrite(a,str); %把第i帧保存到jpg文件
end
```

下面具体介绍视频文件操作中的几个函数。

#### 1.VideoReader

该函数用于读取视频文件对象，函数调用格式：

```
obj=VideoReader(filename)
obj=VideoReader(filename,Name,Value)
```

其中 obj 为结构体，包括如下成员：

Name-视频文件名

Path-视频文件路径

Duration-视频的总时长（秒）

FrameRate-视频帧速（帧/秒）

NumberOfFrames-视频的总帧数

Height-视频帧的高度

Width-视频帧的宽度

BitsPerPixel-视频帧每个像素的数据长度（比特）

VideoFormat-视频的类型，如 'RGB24'.

Tag-视频对象的标识符，默认为空字符串"

Type-视频对象的类名，默认为'VideoReader'.

UserData-要加到对象中的任何类别的数据域，默认为空矩阵.

## 2. read

该函数读取视频帧，调用格式：

video = read(obj)，获取该视频对象的所有帧

video = read(obj,index)，获取该视频对象的指定帧，如：

video = read(obj, 1);            %获取第一帧

video = read(obj, [1 10]);    %获取前 10 帧

video = read(obj, Inf);        %获取最后一帧

video = read(obj, [50 Inf]); %获取第 50 帧之后的帧

## 3.get

该函数获取视频对象的参数，参数的名字为上述 obj 对象的所有成员。调用格式：

Value = get(obj,Name)

### 18.3.9 声音文件

MATLAB 对声音文件的操作主要有：读声音文件的 audioread 函数，写声音文件的 audiowrite 函数，播放声音文件的 audioplayer 和 play 函数，读声音文件信息的函数 audioinfo 函数.

**例 18.38** 声音文件读写、播放示例.

clc, clear

load handel.mat %加载数据 y,Fs

```
audiowrite('myfile.wav',y,Fs) %写声音文件
info=audioinfo('myfile.wav') %读声音文件的信息
[y2,Fs2]=audioread('myfile.wav'); %读入声音文件
player=audioplayer(y2,Fs2); %生成播放对象
play(player); %播放声音文件
```

### 18.3.10 特殊格式的纯文本文件

对于一些特殊格式的纯文本文件，无法使用 `importdata` 和 `textscan` 读入数据，可以使用 `fscanf` 命令读入数据。

**例 18.39** 对于如下格式的保存在 `data1839.txt` 中数据，读入其中的观测值数据。

Measurement Data

N=3

12:00:00

01-Jan-1977

4.21 6.55 6.78

9.15 0.35 7.57

7.92 8.49 7.43

9.59 9.33 3.92

09:10:02

23-Aug-1990

2.76 6.94 4.38

0.46 3.17 NaN

0.97 9.50 7.65

8.23 0.34 7.95

15:03:40

15-Apr-2003

7.09 6.55 9.59

7.54 1.62 3.40

NaN 1.19 5.85

6.79 4.98 2.23

MATLAB 程序如下：

```

clc, clear
fid=fopen('data1839.txt','r');
N=fscanf(fid,'%*s %*s\nN=%d\n\n', 1) %跳过文件头字符串，读入观测值的个数 N
m=4; n=3;%一个观测的行数和列数
for k=1:N %读每个测量值
md(k).mtime = fscanf(fid, '%s', 1);
md(k).mdate = fscanf(fid, '%s', 1);
md(k).meas=fscanf(fid,'%f',[n, m]); %fscanf 以列序把数据填入数组，要做转置
end
fclose(fid); %关闭文件句柄
md.meas %显示读入的数据
data1={md.meas} %把结构数组的数据转化成细胞数组
data2=cell2mat(data1) %把细胞数组转化成矩阵

```

当使用 `textscan`、`fscanf` 或者 `fread` 一次读一部分数据时，可以使用函数 `feof` 测试是否到达文件的尾部。

**例 18.40**（续例 18.39）有纯文本文件 `data1840.txt`，存放如下格式的测量数据，但没有给出测量数据集的个数，读入其中的数据。

```

12:00:00
01-Jan-1977
4.21  6.55  6.78
9.15  0.35  7.57
7.92  8.49  7.43
9.59  9.33  3.92
09:10:02
23-Aug-1990
2.76  6.94  4.38
0.46  3.17  NaN
0.97  9.50  7.65
8.23  0.34  7.95
15:03:40
15-Apr-2003

```

```
7.09  6.55  9.59
7.54  1.62  3.40
NaN   1.19  5.85
6.79  4.98  2.23
```

MATLAB 程序如下:

```
clc, clear
m=4; n=3;
fid=fopen('data1840.txt','r'); %打开文件
block=1;
while ~feof(fid)
    str=fscanf(fid,'%s',1); %读入一行字符串
    if ~isempty(str)
        md(block).mtime=str;
        md(block).mdate=fscanf(fid, '%s', 1); %读入一行字符串
        md(block).meas=fscanf(fid,'%f',[n,m]); %读入矩阵数据
        block=block+1;
    end
end
fclose(fid);
md.metas %显示读入的数据
```

### 18.3.11 数据的批处理

**例 18.41** 现有数据文件 book1.xls, ..., book5.xls, 用命令 importdata 读入数据.

```
clc, clear
n=5 %文件个数
mydata=cell(1, n); %存放各个文件数据的细胞数组
for k=1:n
    filename=sprintf('book%d.xls', k); %构造文件名的格式化字符串
    mydata1{k}=importdata(filename) %从文件读入数据
end
```

**例 18.42** (续例 18.41) 现有数据文件 book1.xls, ..., book5.xls, 读入各 Excel 文件



的第 1 个表单 (Sheet1) 的数据.

```
clc, clear
n=5 %文件个数
mydata=cell(1, n); %存放各个文件数据的细胞数组
for k=1:n
    filename=sprintf('book%d.xls', k); %构造文件名的格式化字符串
    mydata2{k}=xlsread(filename) %从文件读入数据
end
celldisp(mydata2) %显示细胞数组的数据
```

**例 18.43** 随机地生成  $2 \times 8$  的随机矩阵, 其中的元素为区间  $[2, 20]$  上的整数, 矩阵的每一列作为一个点的坐标, 画出这 8 个点, 并在这 8 个点的旁边标注上点 1、点 2、...、点 8, 然后在点 1 和点 5 之间, 点 2 和点 6 之间连两条线段, 以点 3 和点 7 之间的线段为直径画一个圆.

画图的 MATLAB 程序如下:

```
clc, clear
a=randi([2,20],[2,8]);
plot(a(1,:),a(2,:), '*'), hold on
str=strcat('点',int2str([1:8]')); str=cellstr(str);
text(a(1,:)+0.1,a(2,:),str)
plot(a(1,[1,5]),a(2,[1,5]))
plot(a(1,[2,6]),a(2,[2,6]))
x0=(a(1,3)+a(1,7))/2; y0=(a(2,3)+a(2,7))/2; %计算圆心坐标
r=norm(a(:,3)-a(:,7))/2; %计算圆的半径
xy=@(x,y)(x-x0).^2+(y-y0).^2-r^2; %定义圆的隐函数的匿名函数
h=ezplot(xy,[x0-r,x0+r,y0-r,y0+r]);
set(h,'Color','k'), title('')
```

## 习题 18

18.1 表 18.5 给出了一个小组学生的信息, 试用结构数组存储这些信息, 并计算各学生的平均成绩.

表 18.5 一个小组的学生信息

编号	姓名	学习课程	成绩
4515001	张建设	高数, 物理, 英语, 自控	77,78,95,96
4515002	王虹	高数, 物理, 英语, 自控	61,66,76,80
4515003	李密	高数, 物理, 英语, 自控	70,75,86,88
4515004	杨杰	高数, 物理, 英语, 自控	82,86,88,95

18.2 把习题 18.1 中所建立的结构数组转换为细胞数组.