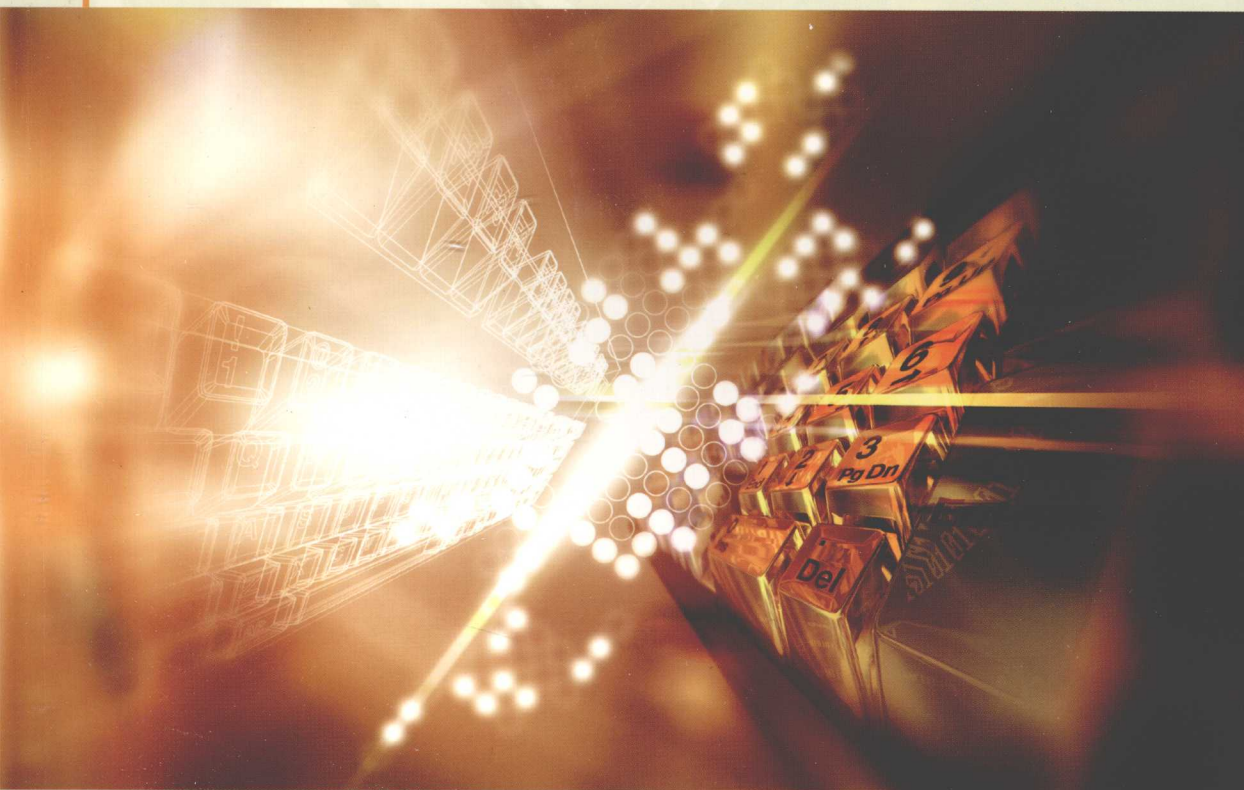


面向21世纪高职高专规划教材

实用数学实验

主 编 罗星海 刘 艳



同济大学出版社
TONGJI UNIVERSITY PRESS

实用数学实验

主 编 罗星海 刘 艳

副主编 丁 勇 金秀云



同济大学出版社
TONGJI UNIVERSITY PRESS

内 容 提 要

本书借鉴基于工作过程的课程开发模式,在“以服务为宗旨,以就业为导向”的办学方针的指导下,结合高职高专学生的特点和“2+1”教学新模式进行编写。全书主要分为6个项目,包括数学应用软件简介、MATLAB在微积分中的应用、数学规划案例、MATLAB在概率统计中的应用、MATLAB在线性代数中的应用及数学建模实践。

本书模块清晰、实用为主,叙述详细、通俗浅显、例证典型、便于自学,适用于高职“数学实验”课程的教学及参考,也可作为“数学建模竞赛”的培训教材或参考书。

图书在版编目(CIP)数据

实用数学实验/罗星海,刘艳主编. —上海:同济大学出版社,2009.8

面向21世纪高职高专规划教材

ISBN 978-7-5608-4050-5

I. 实… II. ①罗…②刘… III. 高等数学—实验—高等学校;技术学校—教材 IV. 013-33

中国版本图书馆CIP数据核字(2009)第133491号

实用数学实验

主 编 罗星海 刘 艳

责任编辑 张 莉 责任校对 徐春莲 封面设计 潘向葵

出版发行 同济大学出版社 www.tongjipress.com.cn
(地址:上海市四平路1239号 邮编:200092 电话:021-65985622)

经 销 全国各地新华书店
印 刷 江苏句容排印厂
开 本 787 mm×960 mm 1/16
印 张 12
印 数 1—3 500
字 数 240 000
版 次 2009年8月第1版 2009年8月第1次印刷
书 号 ISBN 978-7-5608-4050-5

定 价 26.00 元

本书若有印装质量问题,请向本社发行部调换 版权所有 侵权必究

前 言

目前,“以服务为宗旨,以就业为导向”已成为高等职业院校的办学指导思想,“以人为本、因材施教、按需择教”是高职教学的主导方向,各高职院校都在进行学制改革试点和分层教学探索,在数学课时减少的情况下,加快编写适合各校专业特色的高职教材.在这种现实背景下,2005年,为了适应少学时要求,我们组织湖北交通职业技术学院数学教研室具有中高级职称的教师编写了《实用微积分》,这本书使用效果良好.2007年,我们将《概率论与数理统计》、《线性代数》、《线性规划》三门课程根据我院各专业需要进行整合,删繁就简,编写了《实用应用数学》作为“2+1”新学制下的我院适用高等数学教材,以满足工程、管理和文科类各专业的需要.2008年,根据高职课程与教学改革趋势,我们申报了《数学实验课程开发研究》院级课题,组织编写了《实用数学实验》,加强数学与数学软件的应用培养.

本书直接切入问题,在解决问题中注解方法和技巧,浅显易懂、便于阅读.本书旨在通过例题讲述解题的基本操作方法和技巧,培养学生应用数学和计算机解决实际问题的能力,增加专业应用,突出数学实践,强化动手能力,了解数学建模,适当照顾数学知识体系,适应专业模块化教学要求.

参与本书编写的教师有:罗星海副教授(负责全书策划统稿,项目5、项目6)、刘艳副教授(项目2的部分内容)、丁勇老师(项目3、项目4)、金秀云(项目1、项目2的部分内容),黄本利、唐艳老师参与了审稿讨论,在课题经费的支持下,大家共同研讨、团结协作完成了该书的编写工作.

本书在编写过程中,得到了华中师范大学赵东方教授的指导,得到作者单位学院领导、教务处和各系领导的有力支持.对此,我们表示衷心的感谢.

限于编者水平和经验,书中不完备之处敬请读者批评指正.

编 者

2009年8月

目 录

项目 1 数学应用软件简介	(1)
任务 1.1 MATLAB 简介	(1)
任务 1.2 LINGO 与 MATHEMATICA 简介	(16)
项目 2 MATLAB 在微积分中的应用	(20)
任务 2.1 极限	(20)
任务 2.2 导数与微分	(30)
任务 2.3 中值定理	(33)
任务 2.4 不定积分与定积分	(35)
任务 2.5 微分方程	(37)
任务 2.6 多元函数微积分	(38)
任务 2.7 数值分析(计算方法)	(41)
任务 2.8 级数	(44)
项目 3 数学规划案例	(47)
任务 3.1 线性规划	(47)
任务 3.2 非线性规划	(52)
项目 4 MATLAB 在概率统计中的应用	(55)
任务 4.1 概率	(55)
任务 4.2 随机变量及其分布	(57)
任务 4.3 随机变量的数字特征	(60)
任务 4.4 简单随机样本	(60)
任务 4.5 假设检验和区间估计	(61)
任务 4.6 回归分析和方差分析	(64)
任务 4.7 统计分析	(67)
项目 5 MATLAB 在线性代数中的应用	(69)
任务 5.1 求行列式	(69)

任务 5.2 矩阵分析·····	(71)
任务 5.3 解线性方程组·····	(76)
项目 6 数学建模实践 ·····	(82)
任务 6.1 数学建模简介·····	(82)
任务 6.2 历年学生参加全国数学建模大赛获奖论文·····	(83)
任务 6.3 部分学生竞赛体会·····	(167)
参考文献·····	(185)

项目 1 数学应用软件简介

任务 1.1 MATLAB 简介

MATLAB(Matrix Laboratory,即“矩阵实验室”)是由 MathWorks 公司于 1984 年推出的一套数值计算软件,分为总包和若干个工具箱(如图像处理工具箱),可以实现数值分析(如计算机算法)、优化(如管理运筹学)、统计(如建筑工程管理、各专业教学效果评价、经济与金融分析)、偏微分方程数值解(如道路桥梁与工程力学)、自动控制(如机电工程、汽车检测维修与计算机控制)、通信信息系统与信号处理(如车载音响 DVD 等通信设备产品检测、手机信号传输、音频编解码)、图像处理与模式识别(如神舟飞船测控拍摄,指纹、虹膜、步态、人脸识别,视频编解码、视频会议)、人工智能(如遗传基因、演化计算与可演化系统、神经网络、模拟退火、蚁群、粒子群、聚类与主成分分析、蠕虫等智能算法,智能机器人技术)等多领域的计算和图形显示功能。

这里以常用的 MATLAB 7.1 版本为例(各版本的使用方法基本相同)进行讲解.安装 MATLAB 7.1 成功后,运行它,出现图 1-1 所示的界面。

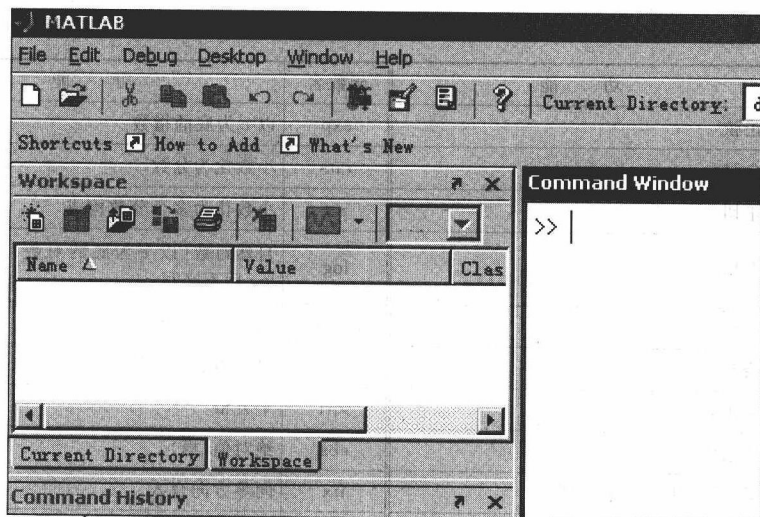


图 1-1

程序输入在上图右端的 Command Window 处。

1. 程序常用起始命令 `clear all`, `clc`;

其中 `clear all` 表示将程序中先前已有的内容(所有的变量和函数)全部清除, 以免在新的操作中出现“重名”情况而产生冲突。

`clc` 表示“清屏”, `clc` 不会真正删除我们先前已经定义的变量和函数, 但可以使屏幕中先前已有的繁多内容不显示, 避免察看时与现写的代码混淆。

逗号“,”和分号“;”的区别是: 逗号“,”表示继续后面的操作, 它和“无标点符号”都在屏幕显示结果, 而分号“;”则做了操作但不在屏幕显示结果。

2. 基本符号说明

符 号	定 义	符 号	定 义
+	加 法	[]	矩阵定义的标志
-	减 法	,	区分列, 函数参数分隔符
*	乘 法	;	区分行, 取消运行显示
/、\	除法(矩阵右除、矩阵左除) ^(注1,2)	'	字符串的标识符号
^	乘方(幂次)	...	续行号(输入后会变蓝色)
=	赋值符号	%	注释(按软件开发规范, 写在其所对应的被注释代码的同行末尾或者上行, 一般不写在其所对应的代码的下行)

注 (1) 矩阵右除 MATLAB 表达形式 $C = A/B$, 结果为 $B * \text{inv}(A)$, $\text{inv}(A)$ 表示 A 的逆矩阵。

(2) 矩阵左除 MATLAB 表达形式 $C = A \backslash B$, 结果为 $\text{inv}(A) * B$ 。

3. 典型函数

函数名	功 能	函数名	功 能
sin	正弦	exp	以 e 为底的指数
cos	余弦	rats	小数化为分数
tan	正切	rat	小数化为多个分数构成
cot	余切	log	自然对数(以 e 为底的对数)(注意程序 log 后不必写 e)
sec	正割	log 10	以 10 为底的对数
csc	余割	log 2	以 2 为底的对数
asin	反正弦	sqrt	平方根
acos	反余弦	abs	绝对值(模)
asec	反正割(其他依此类推)	fix	向零方向舍入
asech	反双曲正割(其他依此类推)	floor	向减小方向舍入
pow2	以 2 为底的幂函数	ceil	向增加方向舍入

续表

函数名	功 能	函数名	功 能
round	四舍五入	complex	建立一个复数 ^(注3)
sign	符号函数 ^(注1)	gcd	最大公因数
rem	求余 ^(注2)	lcm	最小公倍数

注 (1) sign 符号函数表示: 当 $x < 0$ 时 $y = -1$, 当 $x = 0$ 时 $y = 0$, 当 $x > 0$ 时 $y = 1$;

(2) 例 `rem(-14, 3)` 表示用 -14 除以 3, 运行得到余数为 -2;

(3) 例 `complex(1, 2)` 可建立一个复数 $1.0000 + 2.0000i$.

4. 向量和矩阵的基本运算

(1) 向量的生成及其基本运算

向量的生成一般有三种方式: 直接输入法、冒号创建法和等分向量生成法.

直接输入法就是在命令窗口直接输入, 格式的要求是: 向量元素要用 “[]” 括起来, 元素之间用空格、逗号或分号分隔. 其中, 用空格或逗号分隔生成行向量, 用分号分隔生成列向量. 例如

```
>> x1=[1 2 3 4]
x1=
```

```

      1      2      3      4
>> x2=[1, 2, 3, 4]
x2=
```

```

      1      2      3      4
>> x3=[1; 2; 3; 4]
x3=
```

```

      1
      2
      3
      4
```

冒号创建法的格式为: $x = \text{初值} : \text{步长}(\text{即增加量}) : \text{终值}$. 例如

```
>> 2.3:0.3:3.5
ans=
      2.3000      2.6000      2.9000      3.2000      3.5000
```

上述中 `ans` 是 `answer`(结果) 的简写, 在 MATLAB 中表示对应程序运行的结果.

等分向量生成法的格式为: $x = \text{linspace}(a, b, n)$, 即在区间 $[a, b]$ 上产生 n 个等分点, 包括端点. 例如

```
>> linspace(1984, 2008, 4) % 包含端点, 共 4 个点
```

ans=

1984

1992

2000

2008

向量的基本运算如表 1-1 所示.

表 1-1

函数名	功 能	函数名	功 能
max	求向量元素的最大值	sum	求向量所有元素的总和
min	求向量元素的最小值	length	计算向量的长度
mean	求向量元素的平均值	sort	对向量元素按升序排列

举例:

```
>> x=[1 5 2 6 0 4] % 给定向量 x
```

x=

1 5 2 6 0 4

```
>> max(x) % 求最大值
```

ans=

6

```
>> min(x) % 求最小值
```

ans=

0

```
>> mean(x) % 求平均值
```

ans=

3

```
>> (1+5+2+6+0+4)/6 % 验证 mean 是否表示求平均值
```

ans=

3

```
>> sum(x) % 求和
```

ans=

18

```
>> length(x) % 判断向量 x 中有多少个元素,即计算向量的长度
```

ans=

6

```
>> y=sort(x) % 从小到大排序
```

ans=

0 1 2 4 5 6

```
>> fliplr(y) % 将从小到大排序的结果进行左右翻转,即得到从大到小的
```

排序

ans=

6 5 4 2 1 0

(2) 矩阵的生成及其基本运算

矩阵的生成可以直接输入矩阵元素,例如

>> A=[1 1 1; 1 0 -1; 1 -1 1] % 空格,无逗号

A=

```

1      1      1
1      0     -1
1     -1      1

```

>> A=[1, 1, 1; 1, 0, -1; 1, -1, 1] % 有逗号,和上述“空格,无逗号”功能相同

A=

```

1      1      1
1      0     -1
1     -1      1

```

另外,MATLAB 提供了构造特殊矩阵的函数,如表 1-2 所示。

表 1-2

函数名	功 能	函数名	功 能
ones(m, n)	创建 m 行 n 列的全 1 矩阵	rand(m, n)	创建 m 行 n 列的均匀分布矩阵
eye(n)	创建 n 阶单位矩阵	randn(m, n)	创建 m 行 n 列的正态分布矩阵
zeros(m, n)	创建 m 行 n 列的零矩阵	magic(n)	创建 n 阶魔方矩阵(魔方矩阵的性质:每行、每列及两条对角线上的元素之和都相等)

举例(验证魔方矩阵的行、列及对角线上的元素之和都相等):

>> y=magic(8) % 8 阶魔方矩阵 y ,也可用其他字母表示

ans=

```

64      2      3     61     60      6      7     57
 9     55     54     12     13     51     50     16
17     47     46     20     21     43     42     24
40     26     27     37     36     30     31     33
32     34     35     29     28     38     39     25
41     23     22     44     45     19     18     48
49     15     14     52     53     11     10     56
 8     58     59      5      4     62     63      1

```

```

>> x1=[64    2    3    61    60    6    7    57]
x1=
     64     2     3    61    60     6     7    57
>> sum(x1)
ans=
    260
>> y(1:8, 1) % 取矩阵 Y 的从第 1 到第 8 行且第 1 列
ans=
     64
      9
     17
     40
     32
     41
     49
      8
>> sum(y(1:8, 1)) % 矩阵 Y 的第 1 列求和
ans=
    260
>> diag(y) % 提取主对角线元素(从左上角至右下角形成的对角线称为
           主对角线,从右上角至左下角形成的对角线称为副对角线)
ans=
     64
     55
     46
     37
     28
     19
     10
      1
>> sum(diag(y)) % 主对角线元素求和
ans=
    260
>> trace(y) % trace 表示求矩阵的迹(主对角线元素之和),比上述更便捷
ans=

```

260

>> z=fliplr(y) % fliplr 是 flip left/right 的简写,表示将矩阵所有元素按中间对称线进行左右翻转调换

z=

57	7	6	60	61	3	2	64
16	50	51	13	12	54	55	9
24	42	43	21	20	46	47	17
33	31	30	36	37	27	26	40
25	39	38	28	29	35	34	32
48	18	19	45	44	22	23	41
56	10	11	53	52	14	15	49
1	63	62	4	5	59	58	8

>> sum(diag(z)) % 通过前述的翻转后,这里求出的是副对角线元素之和
ans=260

>> sum(y(:, :)) % 一次求出每列之和
ans=

260	260	260	260	260	260	260	260
-----	-----	-----	-----	-----	-----	-----	-----

>> Ty=y' % 程序中 y'表示矩阵 Y' , Y' 表示 Y 的转置,即将 Y 的行列元素互换,而 Ty 是任取的变量名

Ty=

64	9	17	40	32	41	49	8
2	55	47	26	34	23	15	58
3	54	46	27	35	22	14	59
61	12	20	37	29	44	52	5
60	13	21	36	28	45	53	4
6	51	43	30	38	19	11	62
7	50	42	31	39	18	10	63
57	16	24	33	25	48	56	1

>> sum(Ty(:, :)) % 经前述转置后,这里所做的是一次求出 Y 所有列的每行之和

ans=

260	260	260	260	260	260	260	260
-----	-----	-----	-----	-----	-----	-----	-----

按上述可简单验证魔方矩阵的行、列及对角线上的元素之和都相等(此例中都为 260).

矩阵的主要运算小结如表 1-3.

表 1-3

格 式	功 能	格 式	功 能
$A+B$	矩阵相加	A'	矩阵转置
$A-B$	矩阵相减	$\text{rank}(A)$	矩阵的秩
$A*B$	矩阵相乘(其中乘号*不能掉)	$\det(A)$	方阵的行列式
A/B	矩阵右除 ^(注1)	$\text{inv}(A)$	方阵的逆矩阵
$A\backslash B$	矩阵左除 ^(注2)	$\text{size}(A)$	矩阵的阶数
$A.^n$	矩阵乘方(不是矩阵元素乘方)	$\text{eig}(A)$	特征值及特征向量
$A.*B$	矩阵点乘	$\text{trace}(A)$	矩阵的迹
$A./B$	矩阵右点除 ^(注3)	$\text{rot90}(A)$	矩阵反时针旋转 90°
$A.\backslash B$	矩阵左点除 ^(注4)	$\text{fliplr}(A)$	矩阵左右翻折(lr: left/right)
$A.^B$	矩阵元素乘方 ^(注5)	$\text{flipud}(A)$	矩阵上下翻折(ud: up/down)

注 (1) 矩阵右除 MATLAB 表达形式 $C = A/B$, 结果为 $B * \text{inv}(A)$.

(2) 矩阵左除 MATLAB 表达形式 $C = A\backslash B$, 结果为 $\text{inv}(A) * B$.

(3) 矩阵右点除 MATLAB 表达形式 $C = A./B$, 结果为 A 对应元素除以 B 对应元素.

(4) 矩阵左点除 MATLAB 表达形式 $C = A.\backslash B$, 结果为 B 对应元素除以 A 对应元素, 即为 $C = A.\backslash B = B./A$.

(5) 注意“矩阵乘方”与“矩阵元素乘方”的区别.

举例:

```
>> A=[6 7 8; 3 4 5; 1 2 9]
```

```
A=
```

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

```
>> A^2 % 表示 A*A, 即两矩阵相乘, 而不是简单的对 A 元素取平方
```

```
ans=
```

```
65     86    155
35     47     89
21     33     99
```

```
>> A.^2 % 对 A 元素取平方
```

```
ans=
```

```
36     49     64
9      16     25
1       4     81
```

```
>> A.^A % 对 A 元素取自身为幂,例如下面的 46 656 就是 6 的 6 次方
ans=
```

```
    46656    823543    16777216
      27      256      3125
       1        4    387420489
```

通过如下例子可促进对“逆矩阵”的理解:

```
>> A=[6 7 8; 3 4 5]
```

```
A=
```

```
    6    7    8
    3    4    5
```

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

```
??? Error using ==> inv
```

```
Matrix must be square.
```

由此看到,“非方阵”A 是无法求逆矩阵的,逆矩阵是针对“方阵”的概念. 然后,我们来测试一下矩阵和逆矩阵的关系,设方阵 A1 的逆矩阵为 B1:

```
>> A1=[6 7 8; 3 4 5; 1 2 9]
```

```
A1=
```

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

```
>> B1=inv(A1) % 求方阵 A1 的逆矩阵 B1
```

```
B1=
```

```
    1.4444   -2.6111    0.1667
   -1.2222    2.5556   -0.3333
    0.1111   -0.2778    0.1667
```

```
>> A1*B1
```

```
ans=
```

```
    1.0000    0.0000   -0.0000
   -0.0000    1.0000   -0.0000
   -0.0000    0.0000    1.0000
```

上述说明了矩阵 $A1$ 和其逆矩阵 $B1$ 的乘积是一个对角线元素为 1, 其余元素为 0 的矩阵, 即单位矩阵. 于是验证了式子 $AA^{-1} = E$ (其中 A 为方阵, A^{-1} 表示 A 的逆矩阵, E 表示单位矩阵). 可以看到, 通过数学实验可促进对理论和概念的理解.

5. 快捷键

(1) 用 Shift+Enter 可实现程序换行且不执行程序.

(2) 在 Command Window 中, 不框选任何文字, 直接按 Ctrl+C 可实现停止程序执行、光标直接跳到 >>, 从而节省操作时间 (如果框选了文字后按 Ctrl+C, 则实现的是复制).

6. 简易作图

MATLAB 最基本的绘图命令有二维曲线绘图命令 plot 和三维曲线绘图命令 plot3.

下面举例说明二维曲线绘图命令 plot 的简单应用:

```
>> x=0:0.01:2*pi; % pi 表示圆周率  $\pi$ , “2*pi”的乘号 * 不能掉
```

```
>> y=sin(x);
```

```
>> plot(x, y)
```

结果如图 1-2 所示.

```
>> hold on % 在最近图上继续做图
```

```
>> z=cos(x); plot(x, z)
```

结果如图 1-3 所示.

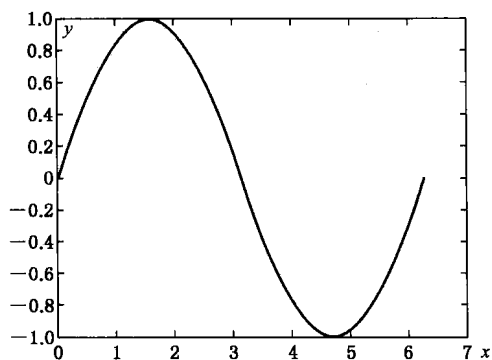


图 1-2

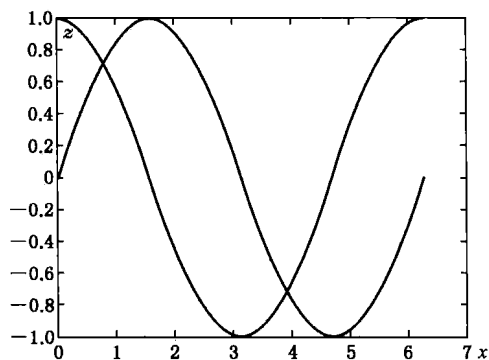


图 1-3

```
>> hold off % 结束 hold on
```

```
>> title('正弦和余弦曲线图')
```

```
>> xlabel('x 的取值')
```

```
>> ylabel('y 和 z 的值')
```

结果如图 1-4 所示.

```
>> text(pi/2, 1, 'y=sinx') % 在横坐标为 pi/2、纵坐标为 1 处标注
                                y = sin x
>> text(pi/2, 0, 'z=cosx') % 在横坐标为 pi/2、纵坐标为 0 处标注
                                z = cos x
```

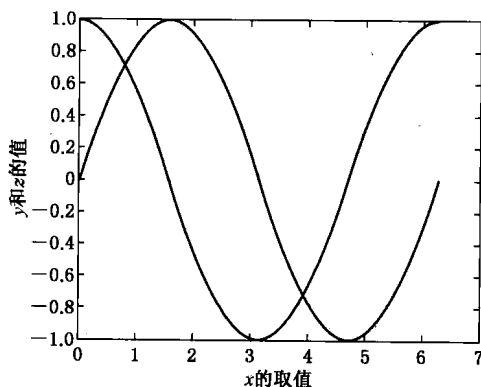


图 1-4 正弦和余弦曲线图

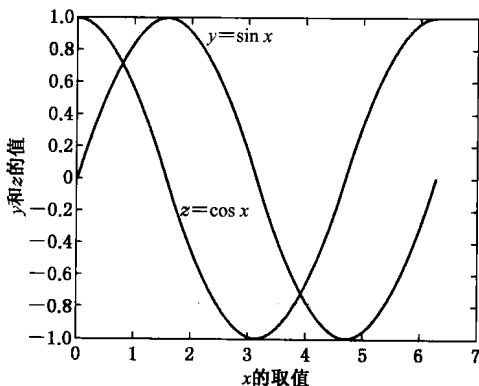


图 1-5 正弦和余弦曲线图

结果如图 1-5 所示。

三维曲线绘图命令 plot3 的用法如下例：

```
>> t=0: pi/50: 10 * pi;
>> plot3(sin(t), cos(t), t) % sin(t), cos(t), t 分别对应 x 轴、y 轴、z 轴
>> title('helix') % helix 表示螺旋
>> xlabel('sin(t)'), ylabel('cos(t)'), zlabel('t')
```

结果如图 1-6 所示。

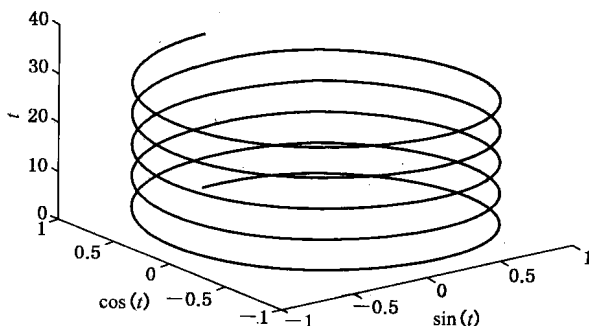


图 1-6 螺旋图

7. 概率统计

(1) 常见命令

factorial(n) 求 n 的阶乘(factorial 表示阶乘)