

# 数学实验

张晓伟

zhangxiaowei@uestc.edu.cn

<http://staff.uestc.edu.cn/zhangxiaowei>

- QQ群： 297754403
- 群昵称（群名片）：
- 班内小号+姓名
- 如： A001张三
- 班内小号查询： 见群公告或群文件或



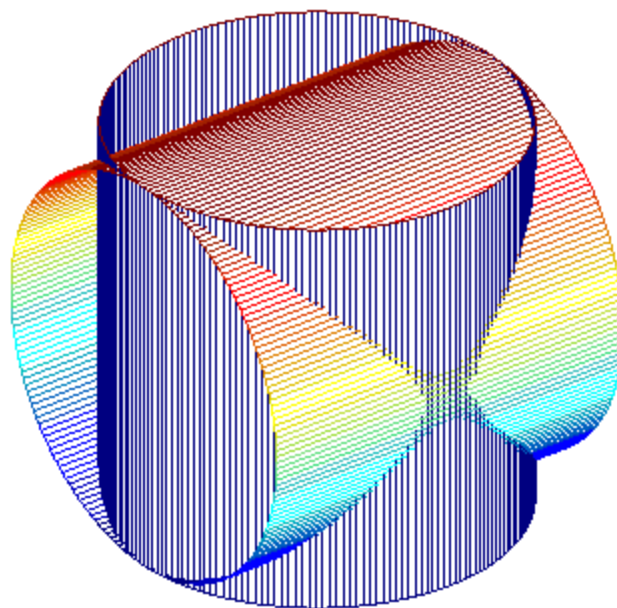
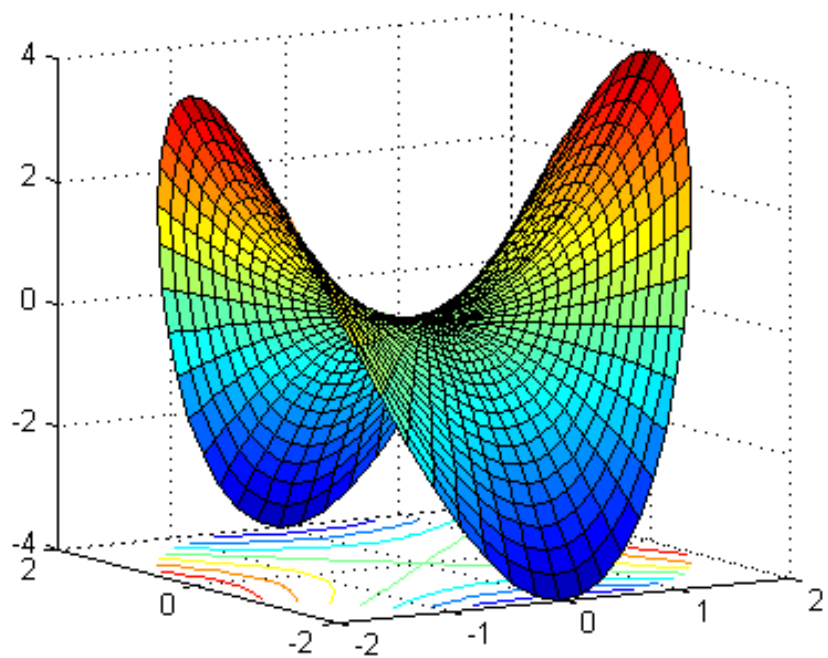
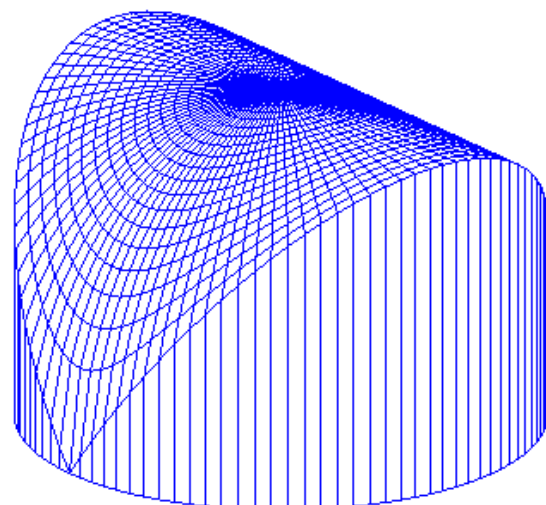
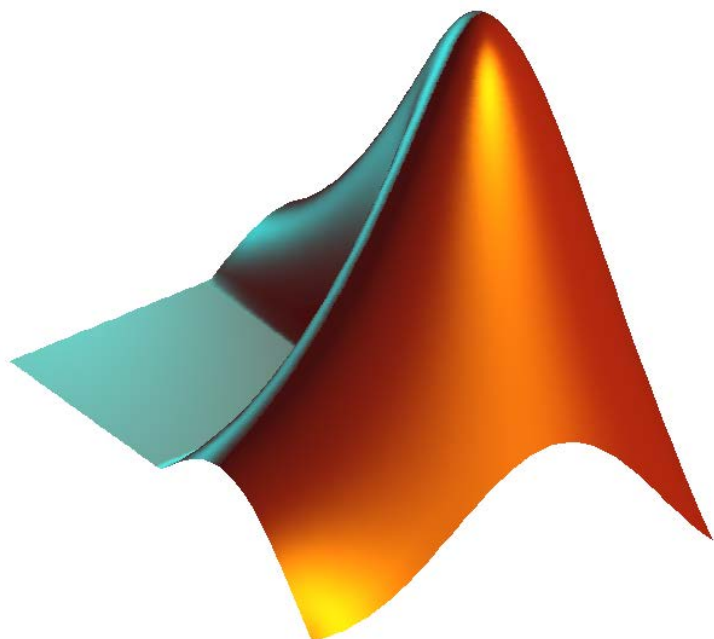
# 第一、二讲 MATLAB入门

- ➔ 数学实验与MATLAB
- ➔ 向量、矩阵的创建和访问
- ➔ 字符串的创建与访问
- ➔ 函数绘图

# ***MATLAB*** (***MATrix LABoratory*** )

一种解释式语言. 易学易用、代码短效率高、  
具有强大的数值计算和绘图功能、扩展性强.

- ✓ 矩阵的数值运算、数值分析、数值模拟
- ✓ 数据可视化、绘制 2维/3维 图形
- ✓ 可以与FORTRAN、C/C++做数据链接
- ✓ 几百个核心内部函数
- ✓ 几十个工具箱(信号处理、自动控制、… )



## 向量的创建和访问

**例：** 给定  $\alpha=15^\circ, 30^\circ, 45^\circ, 60^\circ$ ， 计算 $\sin \alpha$  的值.

```
alpha=[15, 30, 45, 60]*pi/180;
```

```
sin(alpha)
```

```
ans = 0.2588    0.5000    0.7071    0.8660
```

**方括号直接输入法**是创建向量和矩阵的常用方法，输入时将向量元素用方括号括起来，元素之间用逗号或空格隔开.

## 冒号表达式创建向量使用格式

**$x = x0 : step : xn$**

当步长**step=1** 时可省略为,  **$x = x0 : xn$**

当步长**step** 为负数时,  **$x0$**  应大于 **$xn$**

**$alpha = [15 : 15 : 60] * pi / 180;$**   
 **$\sin(alpha)$**

# linspace

- ❑ `linspace(x1,x2,n)`在`x1`和`x2`之间产生`n`个等步长的数据点。
  - ❑ `linspace(x1,x2)`是`linspace(x1,x2,100)`的简写。
  - ❑ `linspace(x1,x2,n)`等价于`x1:d:x2`  
这里 $d=(x2-x1)/(n-1)$ 。
  - ❑ 注意`x2`能取得到。
-



# 矩阵的创建和访问

1.直接输入法; 2.特殊矩阵函数法; 3. 数据文件输入  
注意事项

- ① 矩阵元素必须在方括号[ ]之内;
- ② 同一行相邻元素间用逗号或空格分隔;
- ③ 矩阵的行与行之间用分号分隔.

<b>A=hilb(3)</b>	<b>%用函数创建希尔伯特矩阵</b>
<b>B=invhilb(3)</b>	<b>%创建希尔伯特矩阵的逆阵</b>
<b>A*B</b>	<b>%验证B为A的逆</b>

<b>A =</b>		<b>B =</b>		<b>ans =</b>
<b>1     1/2     1/3</b>		<b>9   -36   30</b>		<b>1     0     0</b>
<b>1/2   1/3     1/4</b>		<b>-36   192   -180</b>		<b>0     1     0</b>
<b>1/3   1/4     1/5</b>		<b>30   -180   180</b>		<b>0     0     1</b>

<b>zeros(m,n)</b>	<b><math>m \times n</math>阶零矩阵</b>
<b>eye(m,n)</b>	<b><math>m \times n</math>阶单位矩阵</b>
<b>ones(m,n)</b>	<b><math>m \times n</math>阶全1矩阵</b>
<b>rand(m,n)</b>	<b><math>m \times n</math>阶随机矩阵</b>
<b>randn(m,n)</b>	<b>正态随机数矩阵</b>
<b>magic(n)</b>	<b><math>n</math>阶魔方矩阵</b>
<b>hilb(n)</b>	<b><math>n</math>阶Hilbert矩阵</b>
<b>invhilb(n)</b>	<b>逆Hilbert矩阵</b>
<b>pascal(n)</b>	<b><math>n</math>阶Pascal矩阵</b>
<b>vander(C)</b>	<b>由向量C生成范德蒙矩阵</b>

## 向量访问

**A=pascal(3)**

**a=A(2,3)**

**A =**

1	1	1
1	2	3
1	3	6

**a =**

3

```
b=A( : , 2 )
```

```
c=A( 3 , [ 1 3 ] )
```

```
d=A( : , 3 :-1 : 1 )
```

```
b =
```

1

2

3

```
c =
```

1

6

```
d =
```

1

1

1

3

2

1

6

3

1

将几个矩阵合成一个:

**A = [1 2; 3 4]**

**B = [A, [5;6] ]**

**C= [A; [5,6]]**

**A =**

1	2
3	4

**B =**

1	2	5
3	4	6

**C =**

1	2
3	4
5	6

# 计算机浮点数表示法

$$6.378137 \times 10^3 \rightarrow 6.378137 \text{ e } +003$$

$$2.99792458 \times 10^5 \rightarrow 2.99792458 \text{ e } +005$$

**例1.24**近似计算地球表面积,并以不同格式输出数据

**R=6378.137; S=4\*pi\*R^2**

**format long,S**

**format bank,S**

<b>format short</b>	<b>短格式</b>	<b>5.1121e+008</b>
<b>format long</b>	<b>长格式</b>	<b>5.112078933958109e+008</b>
<b>format bank</b>	<b>银行格式</b>	<b>511207893.40</b>
<b>Format rat</b>	<b>分式格式</b>	<b>511207893</b>

## ***MATLAB*内部常数**

常 数	返 回 值
<b>ans</b>	默认变量名，保存最近的结果
<b>eps</b>	浮点数相对精度(2.2204e-016)
<b>realmax</b>	最大浮点数(1.7977e+308)
<b>realmin</b>	最小浮点数(2.2251e-308)
<b>pi</b>	圆周率 (3.1416)
<b>i,j</b>	虚数单位
<b>inf</b>	无限大
<b>NaN</b>	不合法的数值，非数值

# 三角函数与双曲函数

$\sin$	正弦函数
$\arcsin$	反正弦函数
$\cos$	余弦函数
$\arccos$	反余弦函数
$\tan$	正切函数
$\arctan$	反正切函数
$\cot$	余切函数
$\operatorname{arccot}$	反余切函数
$\sec$	正割函数
$\operatorname{arcsec}$	反正割函数
$\csc$	余割函数
$\operatorname{arccsc}$	反余割函数

$\sinh$	双曲正弦函数
$\operatorname{arsinh}$	反双曲正弦函数
$\cosh$	双曲余弦函数
$\operatorname{arcosh}$	反双曲余弦函数
$\tanh$	双曲正切函数
$\operatorname{artanh}$	反双曲正切函数
$\operatorname{sech}$	双曲正割函数
$\operatorname{arsech}$	反双曲正割函数
$\operatorname{csch}$	双曲余割函数
$\operatorname{acsch}$	反双曲余割函数
$\coth$	双曲余切函数
$\operatorname{acoth}$	反双曲余切函数



<code>abs(x)</code>	绝对值
<code>sqrt(x)</code>	开平方
<code>conj(z)</code>	共轭复数
<code>round(x)</code>	四舍五入
<code>floor(x)</code> [- $\infty$ ]	舍去正小数
<code>rat(x)</code>	分数表示
<code>gcd(x,y)</code>	最大公因数
<code>exp(x)</code>	自然指数
<code>log(x)</code>	自然对数
<code>log10(x)</code>	10底对数

<code>angle(z)</code>	复数z的相角
<code>real(z)</code>	复数z的实部
<code>imag(z)</code>	复数z的虚部
<code>fix(x)</code> [0]	舍去小数取整
<code>ceil(x)</code> [+ $\infty$ ]	加入正小数取整
<code>sign(x)</code>	符号函数
<code>rem(x,y)</code> [ <code>mod(x,y)</code> ]	求x除以y的余数
<code>lcm(x,y)</code>	最小公倍数
<code>pow2(x)</code>	以2为底的指数
<code>log2(x)</code>	以2为底的对数

# rem mod

□  $\text{rem}(x,y)$ 和 $\text{mod}(x,y)$ 要求 $x,y$ 必须为相同大小的实矩阵或为标量(数字)。

□ 如果 $y \neq 0$ ，则有

$\text{rem}(x,y) = x - n.*y$ ，这里 $n = \text{fix}(x./y)$ ；

$\text{mod}(x,y) = x - n.*y$ ，这里 $n = \text{floor}(x./y)$ 。

□ 如果 $x,y$ 的符号相同，则

$\text{rem}(x,y) = \text{mod}(x,y)$ ;

否则，

$\text{rem}(x,y) \neq \text{mod}(x,y)$ 。

□  $\text{rem}(x,y)$  和 $x$ 的符号相同；  $\text{mod}(x,y)$ 和 $y$ 的符号相同。

---

# 字符串变量

- ❑ 在MATLAB中，字符串是用单撇号括起来的字符序列。MATLAB将字符串当作一个行向量，每个元素对应一个字符，其标识方法和数值向量相同。也可以建立多行字符串矩阵。
  - ❑ MATLAB将字符串当作一个行向量，每个元素对应一个字符，其标识方法和数值向量相同。也可以建立多行字符串矩阵。
  - ❑ 字符串是以ASCII码形式存储的。abs和double函数都可以用来获取字符串矩阵所对应的ASCII码数值矩阵。相反，char函数可以把ASCII码矩阵转换为字符串矩阵。
-

# 字符串的创建和访问

```
s='abc'  
t='1234'  
s1=[s,t]  
s2=['abc','1234']  
s3=strcat(s,'123','--',t)  
m=['abc';'123']  
s2(3)  
m(2,:)
```

```
s =  
abc  
t =  
1234  
s1 =  
abc1234  
s2 =  
abc1234  
s3 =  
abc123--1234  
m =  
abc  
123  
ans =  
c  
ans =  
123
```

int2str (str2int)	整数 ↔ 字符串	strmatch	查找
num2str (str2num)	数值 ↔ 字符串	strcat	连接
upper (lower)	大小写转换	abs	ASCII码
strcmp	比较	strcmpi	比较(忽略大小写)
findstr	查找	strrep	替换

**例：**三个名人Euler, Elizabeth, Plato职业分别是 mathematician, movie star, philosopher, 编写程序正确联接他们的名字和职业并输出.

```
n1='Euler';  
n2='Elizabeth';  
n3='Plato';  
p1='mathematician';  
p2='movie star';  
p3='philosopher';  
s1=strcat(n1,'--',p1),  
s2=strcat(n2,'--',p2),  
s3=strcat(n3,'--',p3)
```

---

**例** 建立一个字符串向量，然后对该向量做如下处理：

- (1) 取第1~5个字符组成的子字符串。
- (2) 将字符串倒过来重新排列。
- (3) 将字符串中的小写字母变成相应的大写字母，其余字符不变。
- (4) 统计字符串中小写字母的 个数。

**命令如下：**

```
s='FHkjhd6783n&JKHgdt992';
```

```
s1=s(1:5)                                %取子字符串
```

```
s2=s(end:-1:1)                            %将字符串倒排
```

```
Index=find(s>='a' & s<='z');              %找小写字母的位置
```

```
s(Index)=s(Index)-('a'-'A');              %将小写字母变成相应的大写字母
```

```
length(Index)                             %统计小写字母的个数
```

## 例: 生肖问题

```
n=input('input n:=');  
S='鼠牛虎兔龙蛇马羊猴鸡狗猪';  
k=rem(n-4,12)+1;  
s=S(k);  
s=strcat(int2str(n), '年是', s, '年')
```

**input n:=2006**

**s= 2006年是狗年**

**input n:=2016**

**s=2016年是猴年**



```
n=input('input n:=');  
S='猴鸡狗猪鼠牛虎兔龙蛇马羊';  
k=rem(n,12)+1;  
s=S(k);  
s=strcat(int2str(n), '年是', s, '年')
```

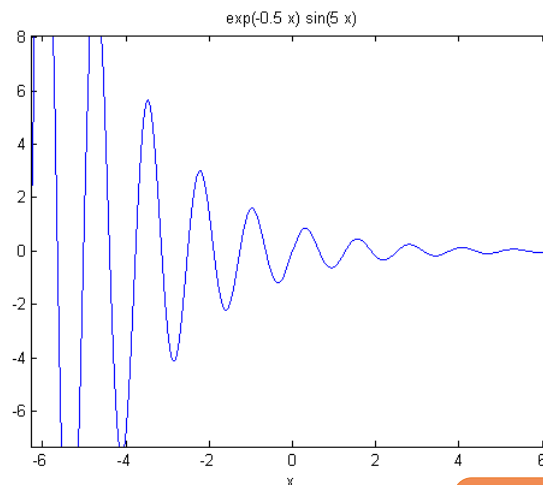
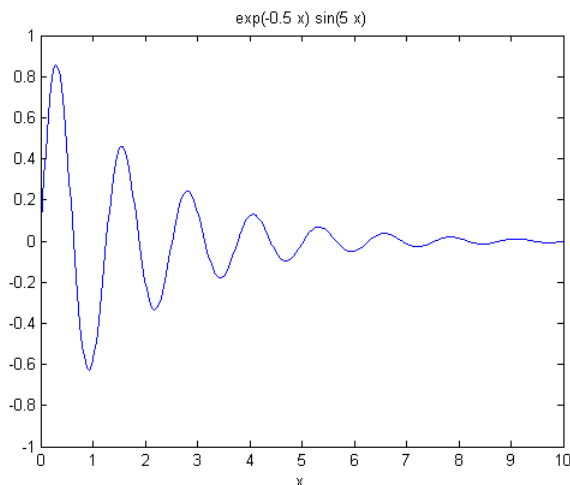
# 函数绘图

例：用ezplot()命令绘衰减振荡曲线函数

$$y = e^{-0.5x} \sin 5x$$

图形.

`ezplot('exp(-0.5*x)*sin(5*x)',[0,10,-1,1])`



`ezplot('exp(-0.5*x)*sin(5*x)')`

$(-2\pi, 2\pi)$

在解决实际问题时，如果频繁使用同一个数学表达式，则可定义一个临时函数以方便操作。

**定义方法：函数名 = inline('表达式')**

**例：**定义函数  $f(x) = x \sin \frac{1}{x}$  并分析函数性质.

**myfun = inline('x.\*sin(1./x)')**

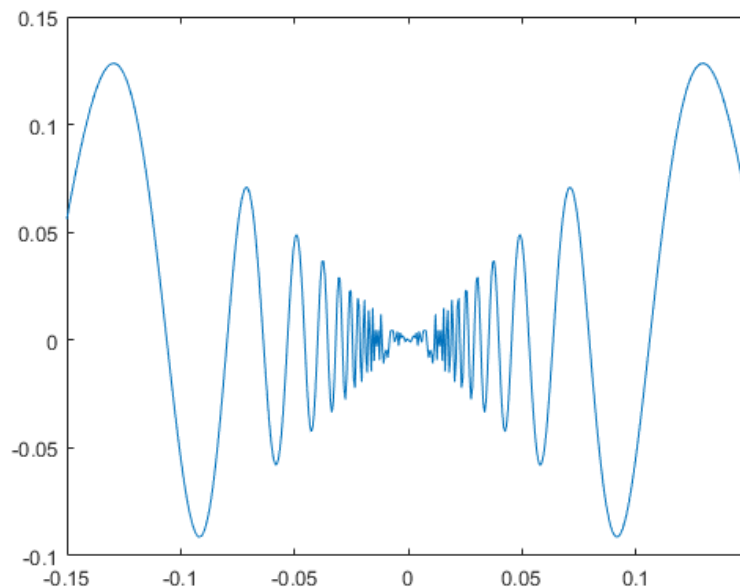
**fplot(myfun, [-0.15, 0.15])**

**x = 1:5;**

**y = myfun(x)**

**y =**

**0.8415    0.9589    0.9816    0.9896    0.9933**



```
ezplot('exp(-0.5*x)*sin(5*x)',[0,10,-1,1])
```

```
ezplot('exp(-0.5*x)*sin(5*x)',[0,10])
```

```
ezplot('exp(-0.5*x)*sin(5*x)')
```

ezplot可以省掉参数

```
fplot('exp(-0.5*x)*sin(5*x)',[0,10,-1,1])
```

```
fplot('exp(-0.5*x)*sin(5*x)',[0,10])
```

或用 `fun=inline('exp(-0.5*x)*sin(5*x)')` 替换  
第一部分

# 一元函数绘图方法

## ezplot()

简易绘图方法，快速方便。

## fplot()

函数绘图方法，与简易绘图相似，但要指定绘图区域。

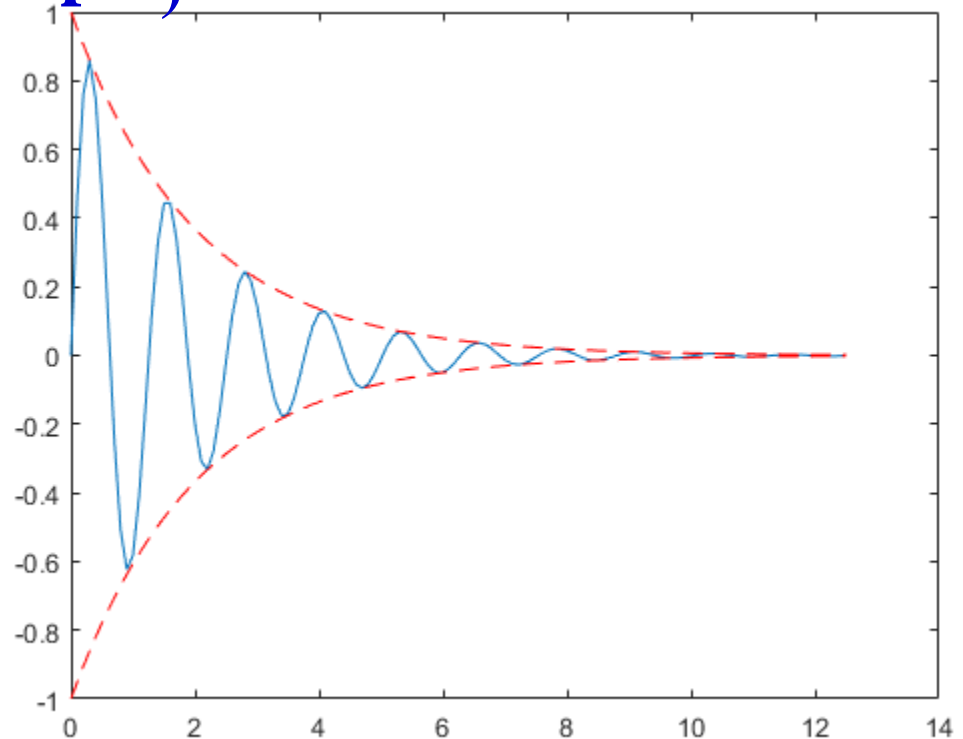
## plot()

基本绘图方法，利用一元函数自变量的一系列数据和对应函数值数据绘图，具有很大灵活性。

**例:**用基本绘图方法绘衰减振荡函数  $y = e^{-0.5x} \sin 5x$  的图形并用虚线表示振幅衰减情况。

```
x = 0:0.1:4*pi;  
y = exp(-0.5*x) ;  
y1 = y .* sin(5*x);  
plot(x, y1, x, y, 'r--', x, -y, 'r--')
```

缺省参数:'b-'[默认值]



如何在 $[-2\pi, 2\pi] \times [-\pi, \pi]$ 绘制 $z=xy$ 的图形？

分析：

$[X,Y] = \text{meshgrid}(-2*\pi:0.9:2*\pi, -\pi:0.5:\pi)$

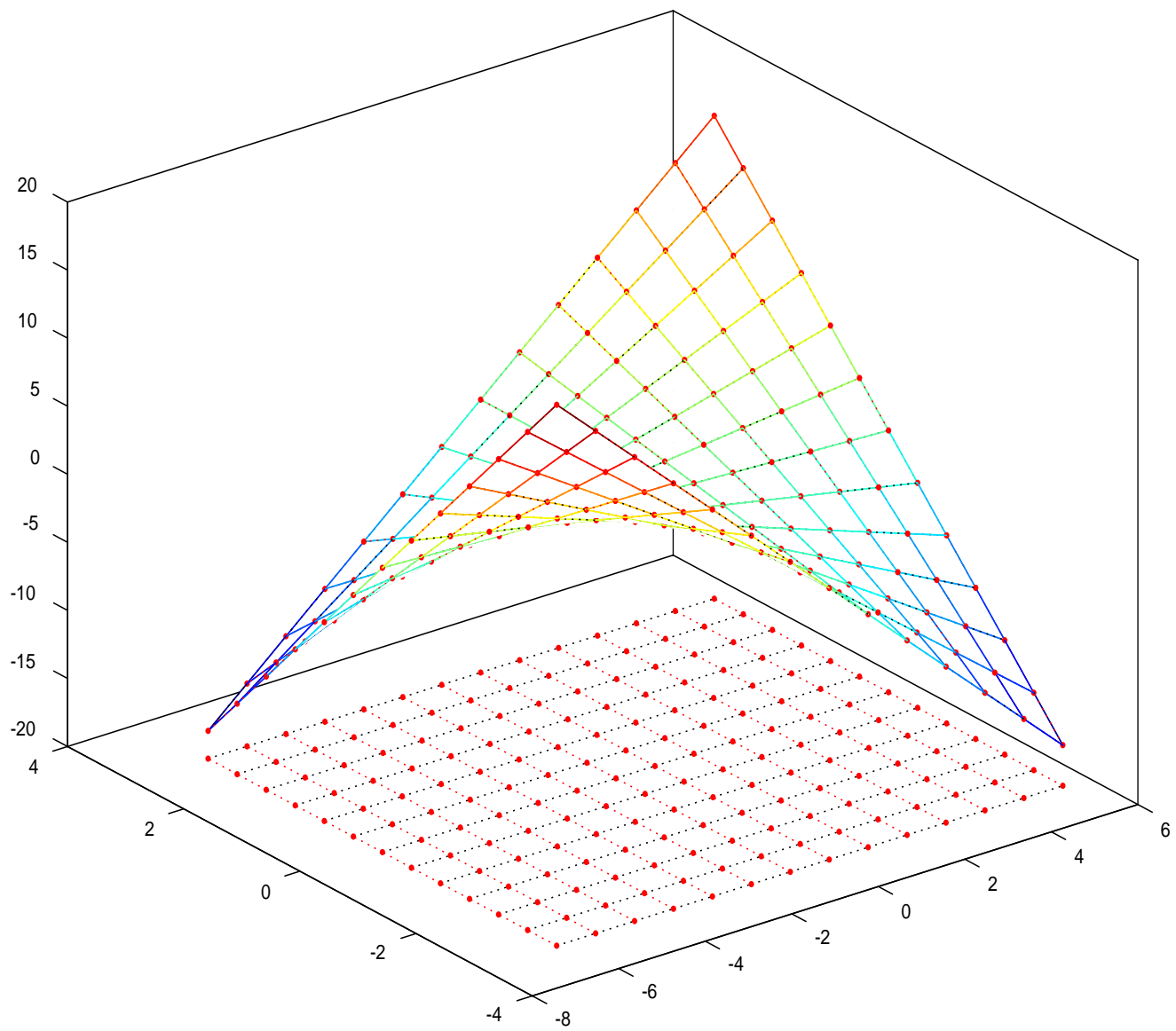
这里 $X,Y$ 为同型矩阵，存储的是绘图区域中抽样点的坐标，即下图 $xoy$ 平面中红色的点。

$Z=X.*Y$

这里 $Z$ 为抽样点处的函数值，与 $X,Y$ 同型。注意点运算。

$\text{mesh}(X,Y,Z)$

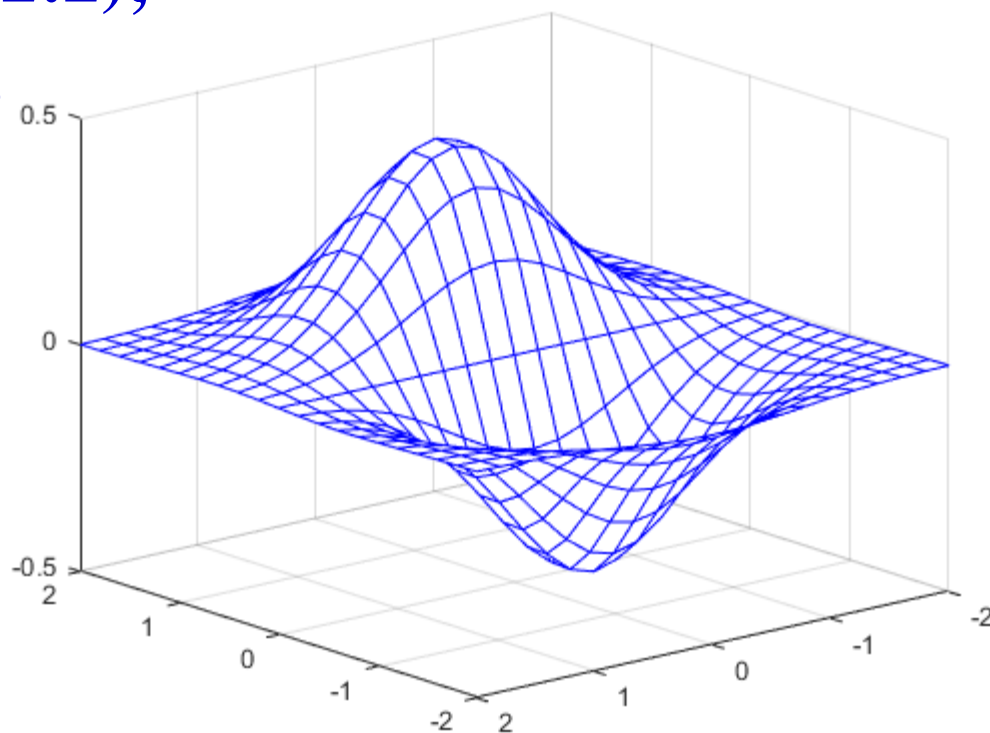
绘制二元函数图形, $X,Y,Z$ 为抽样点的坐标,即下图曲面上红色的点。



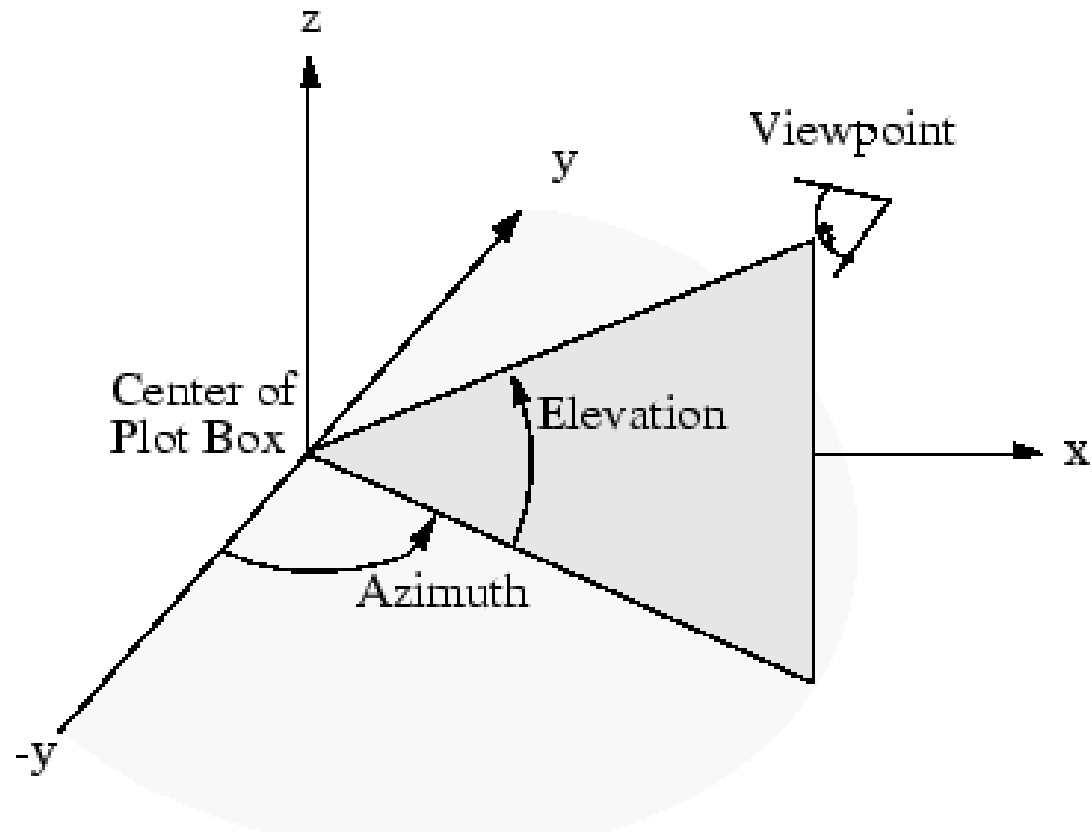


**例：** 绘二元函数  $z = x \exp(-x^2 - y^2)$  的图形。

```
[x, y] = meshgrid(-2:0.2:2);  
z = x.*exp(-x.^2-y.^2);  
mesh(x, y, z)  
colormap([0 0 1])  
view(-130,20)
```



类似命令:**contour, contour3,surf, meshc, meshz**等

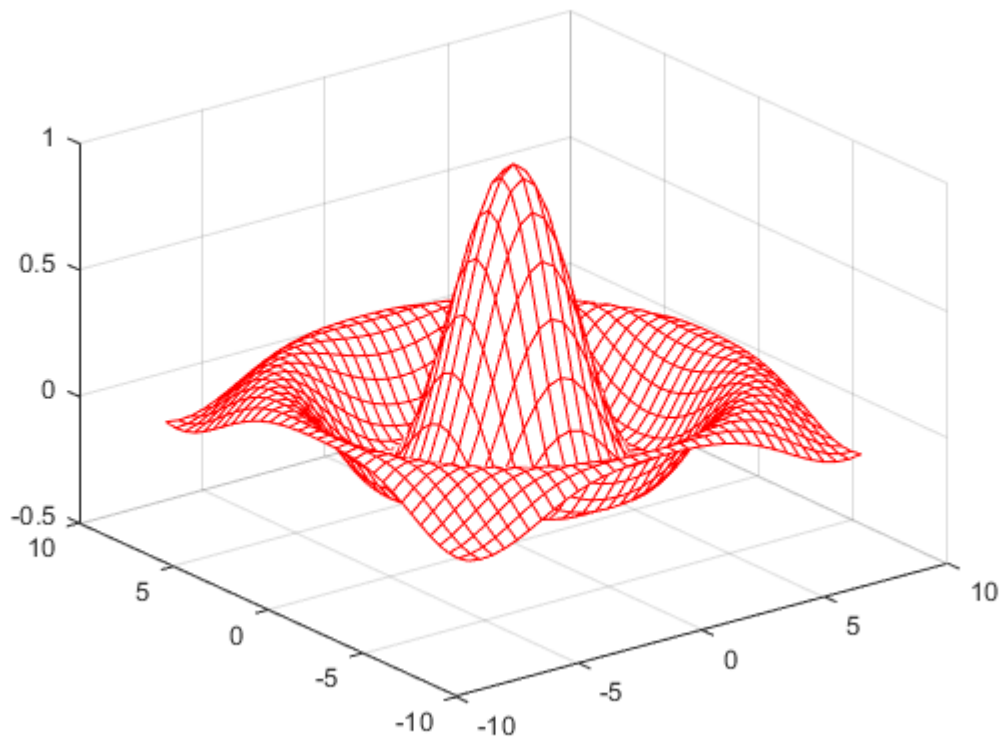


**view(Azimuth,Elevation)**

例: 绘二元函数  $z = \frac{\sin \sqrt{x^2 + y^2}}{\sqrt{x^2 + y^2}}$  图形

```
[x,y] = meshgrid(-8:0.5:8);  
r = sqrt(x.^2+y.^2) + eps;  
z = sin(r)./r;  
mesh(x, y, z)  
colormap([1,0,0])
```

浮点数相对精度  
(2.2204e-016)

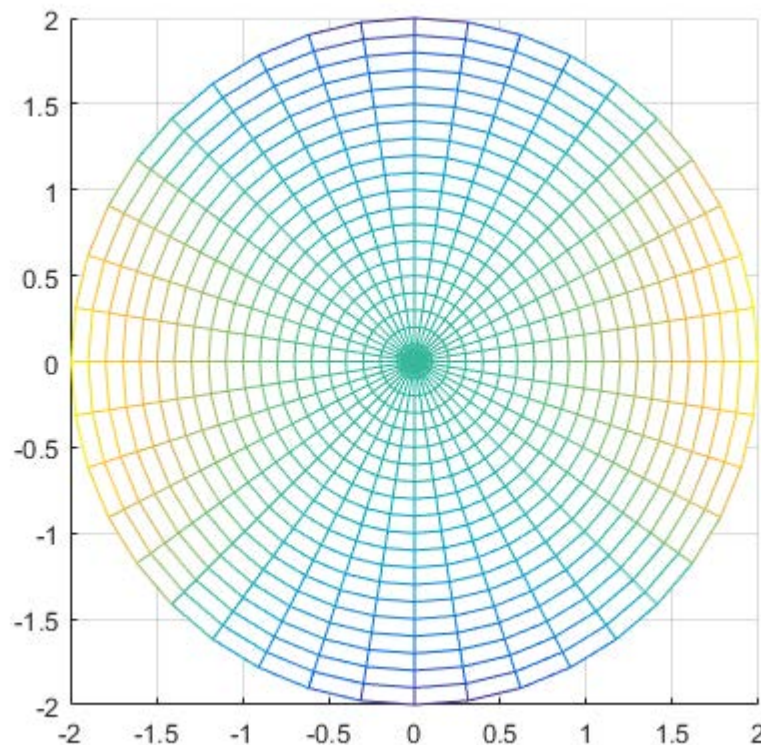


加 **eps** 是避免残缺图形

**例：** 在圆域  $D = \{(x, y) \mid x^2 + y^2 \leq r^2\}$  上绘制马鞍面

$$z = x^2 - y^2$$

```
[t, r] = meshgrid(0:pi/20:2*pi, 0:0.1:2);  
X = r.*cos(t);  
Y = r.*sin(t);  
Z = X.^2-Y.^2;  
mesh(X, Y, Z)
```



## 极坐标绘图 polar()

例：玫瑰线极坐标方程  $\rho = a \cos 3\theta$

```
theta = 0:0.001:2*pi;  
r = cos(3*theta);  
polar(theta, r, 'k')
```

