3 信号模型与运算

讲 解 人： 王俊波

E-mail: jbwang@seu.edu.cn

Phone: 13770681926

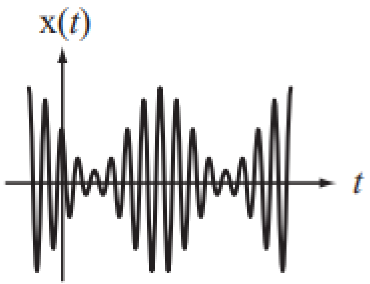
QQ:308322767

主要内容

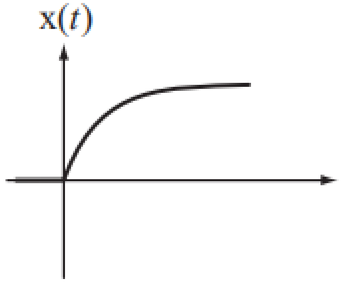
- □ 信号建模概述
- □ 信号在MATLAB中的表示方法
  - ✓ 连续时间信号的表示
  - ✓ 离散时间信号的表示
- □ 信号的时间运算与变换
  - ✓ 连续时间信号的运算与变换
  - ✓ 离散时间信号的运算与变换
- □ 信号的能量与功率
- □ 卷积

信号模型概述

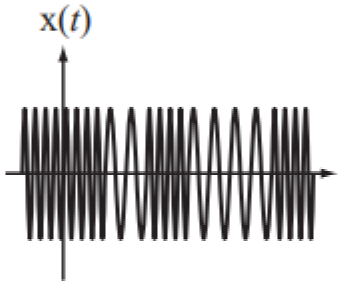
- 模型：对观测到的自然现象的一种简化表示
- 信号：消息的载体，是消息的一种表现形式
- 信号模型：建立信号模型时，将物理知识加入到模型中将提升模型的实用性，通常能够改善基于模型的算法性能
  - ✓ 无线通信中的多径信号，它是发射信号的副本经延迟、衰减后求和的结果，可用横向滤波器来见面，输入为发射信号
- 数学上，信号被定义为根据一个或多个自变量（如时间、空间、频率等）变化的任何自然量
- 常见的信号表现为随时间变化的某些物理量



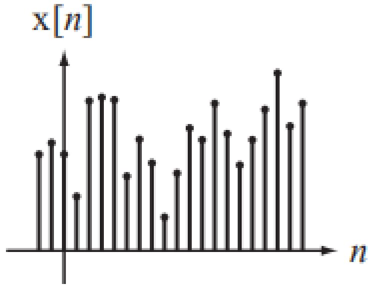
Amplitude-Modulated Carrier in a Communication System



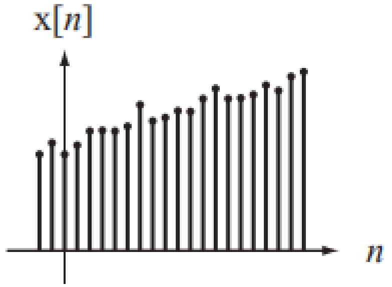
Step Response of an *RC* Lowpass Filter



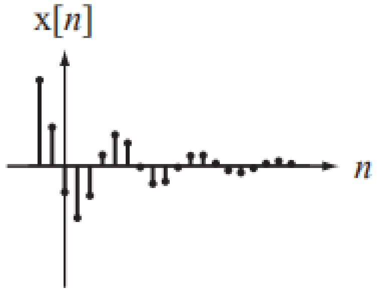
Frequency-Shift-Keyed Binary Bit Stream



Daily Closing NASDAQ Composite Index



Weekly Average Temperature



Samples from an Exponentially Damped Sinusoid

信号的分类

- 信号的数学模型可划分为确定性模型和随机性模型
  - ✓ 确定性信号模型：包括所有信息已知的信号，也包括数学形式已知但部分参数未知的信号（但不是随机的未知参数）
  - ✓ 随机性信号模型：通常为一个随机过程，其概率密度函数（PDF）或概率质量函数（PMF）完全已知，或者除某些参数外的参数随机分布是已知的
- 针对通信信号，按照模型信息是否确定和参数信息是否已知，可将信号大致分为四类

模型类型	模型信息	参数信息	例子	注释
类型 1	确定性	已知	$\cos\left(2\pi n+\pi/2\right)$	完全已知
类型 2	确定性	未知	$\cos\left(2\pi n+\phi\right)$	$\phi\in\left[-\pi,\pi\right)$
类型 3	随机性	已知	$\cos\left(2\pi n+\Phi\right)$	Φ 的分布已知 $\Phi\sim\mathcal{U}\left(-\pi,\pi\right)$
类型 4	随机性	未知	$\cos\left(2\pi n+\Phi\right)$	Φ 的分布未知

信号建模的注意事项

- 信号模型应尽可能反映物理现实，但也不能过于复杂，以至于算法设计变得非常困难；
- 信号的数学模型可是完全已知的确定性模型，或是带有未知参数的确定性模型，也可以是分布信息完全已知的随机模型，或是分布信息未知的随机模型；
- 必须在线估计参数的信号模型往往需要设计自适应算法进行处理；
- 在实际中假定信号完全已知是有风险的，最好假定一些参数是未知的并在线估计，虽然算法性能有可能下降，但鲁棒性显著提高

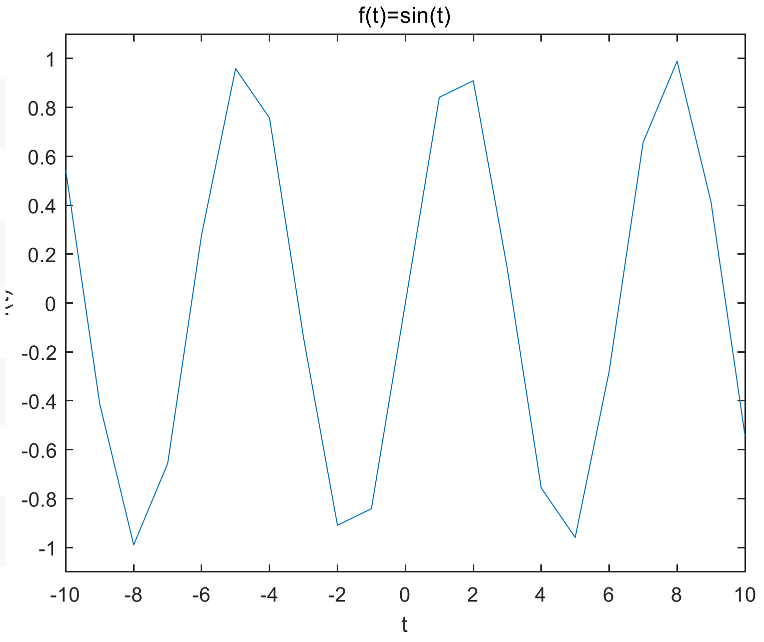
连续时间信号在MATLAB上的表示方法

- MATLAB是在计算机上运行，所有工作都是在离散时间上完成的，因此，严格意义上来讲，MATLAB并不能处理连续信号
- 直觉上，用连续信号在等时间间隔点的样值来近似地表示连续信号，当取样时间间隔足够小时，这些离散的样值就能较好地近似出连续信号
- 方法一：抽样表示法
  - ✓ 实现方法：对连续信号采样，将连续信号表示为样点的向量或矩阵
  - ✓ 优点：信号处理简单易行
  - ✓ 缺点：采样间隔过小，增加数据量；
采样间隔过大，可能损失信号的关键特征
- 方法二：符号表示法
  - ✓ 实现方法：借助于Symbolic Math Toolbox的符号运算，将连续信号表示为符号
  - ✓ 优点：信号表示简单直观
  - ✓ 缺点：过于复杂的信号不易处理，计算时间过长，有可能无符号解

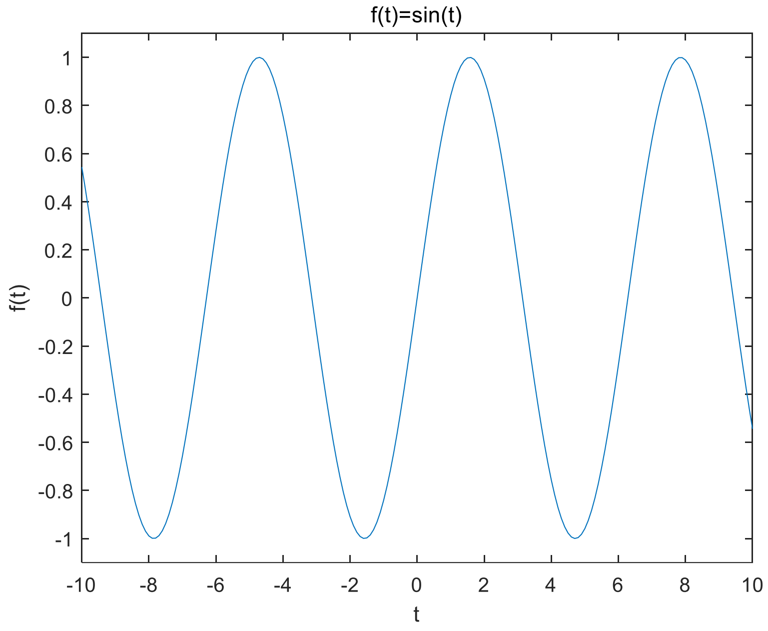
连续信号的表示方法一：抽样表示法

- 对于连续时间信号$f(t)$，用两个向量f和 t来表示
  - ✓ f为连续信号$f(t)$在向量t所定义的时间点上的样值
  - ✓ t是形如$t = t_1 : p : t_2$时间范围向量
  - ✓ t_1为信号起始时间, t_2为终止时间, p为时间间隔
- 例：定义$f(t) = \boldsymbol{sin(t)}$

```
t=-10:p:10;
f=sin(t);
plot(t,f);
title('f(t)=sin(t)');
xlabel('t');
ylabel('f(t)');
axis([-10,10,-1.1,1.1]);
```



$p = 1$



p=**0.1**

✓ 注意：时间间隔的选择

连续信号的表示方法二：符号表示法

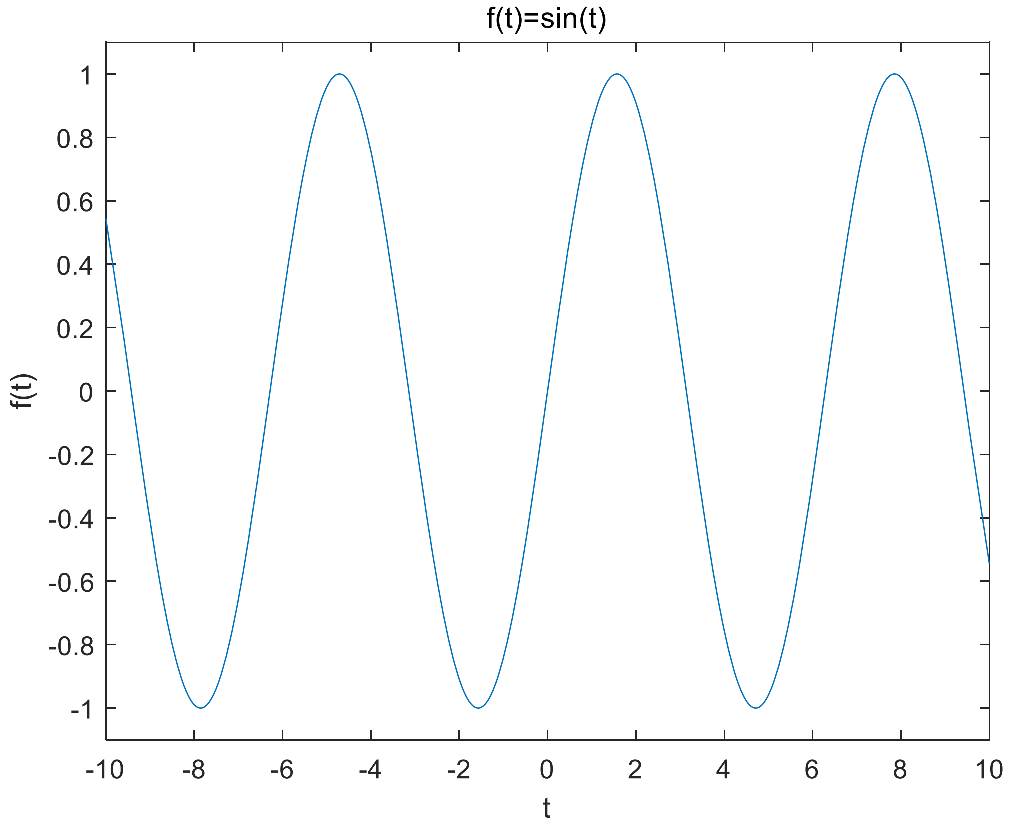
- Symbolic Math Toolbox 是MATLAB进行符号运算的工具箱
- 实现1：先定义时间符号 t ，再用函数关系定义符号函数 f

```
syms t;
f=sin(t);
fplot(f,[-10,10]);
```

- 实现2：先写出函数表达式的字符串，然后将字符串转成符号表达式

```
f=str2sym('sin(t)');
fplot(f,[-10,10]);
```

✓ 注意：使用str2sym()定义符号表达式（早期版本是sym()）

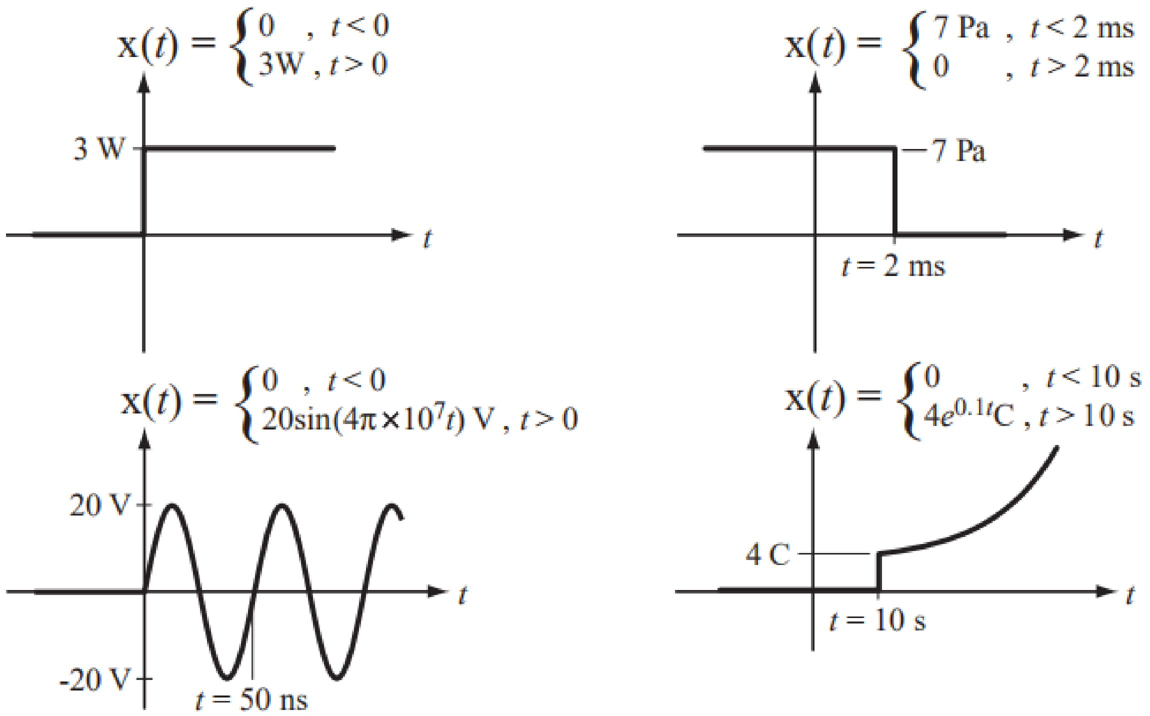


通信仿真中常见连续时间信号——单位阶跃信号(1/2)

□ 单位阶跃信号的数学定义

$$\boldsymbol{u}(t)=\begin{cases}1 & t\geq 0\\ 0 & t<0\end{cases}$$

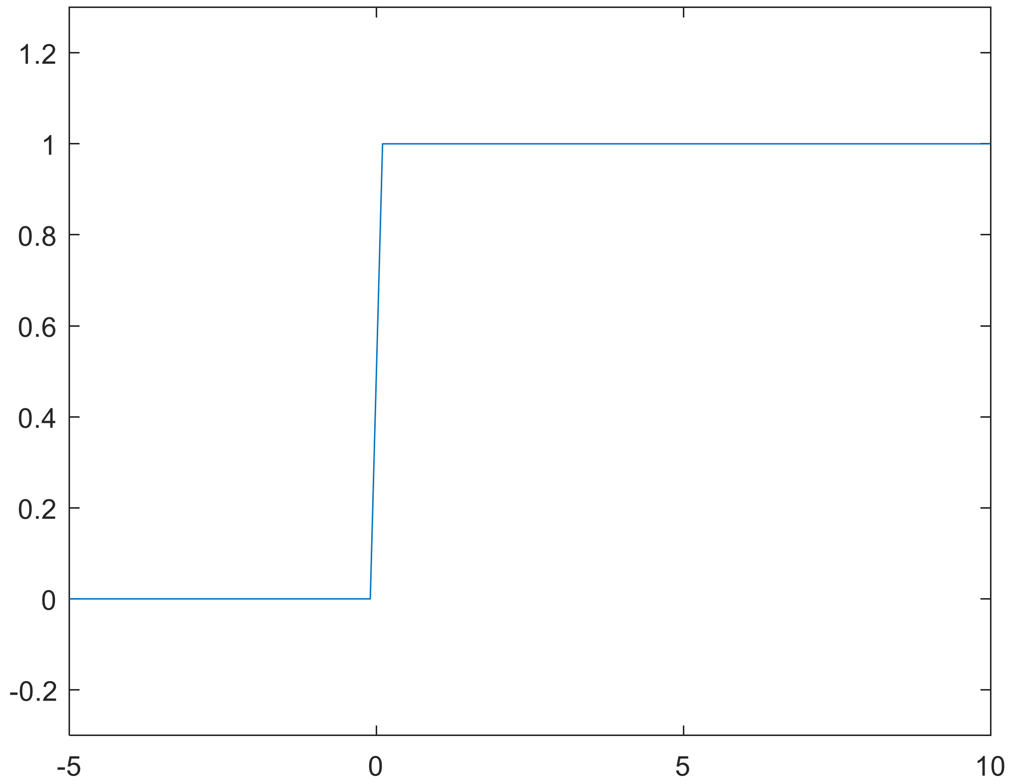
- ✓ 单位阶跃信号是信号分析的基本信号之一， 有着十分重要的作用，常用于简化信号的时域表示
- ✓ 根据单位阶跃信号的特性可用来表示时限信号和单边信号



通信仿真中常见连续时间信号——单位阶跃信号(2/2)

- MATLAB使用函数heaviside(t)定义单位阶跃函数
- 例：用矢量法绘制单位阶跃函数

```
t=-5:0.1:10;
u=heaviside(t);
plot(t,u);
ylim([-0.3 1.3]);
```



- heaviside(t)函数可以在符号计算中使用

```
f=str2sym('heaviside(t)');
fplot(t,u);
ylim([-0.3 1.3]);
```

- 思考：如何定义$u(t-t_0)=\begin{cases}1 & t\geq t_0\\ 0 & t<t_0\end{cases}$？

通信仿真中常见连续时间信号——单位冲击信号(1/2)

- 单位冲击信号的数学定义为

$$\begin{cases} \int_{-\infty}^{\infty} \delta(t) = 1 \\ \delta(t) = 0, t \neq 0 \end{cases}$$

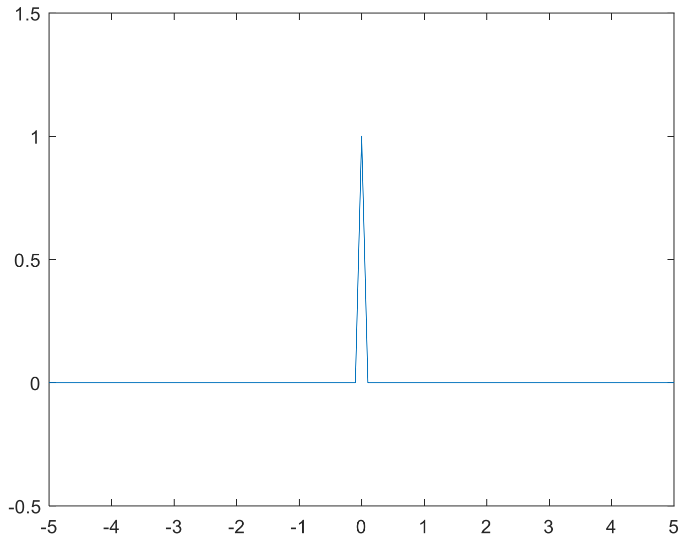
  - ✓ 定义表明该信号除原点以外，处处为零且面积为一
  - ✓ 单位冲击函数是信号与系统的基本信号之一，是信号分析的基础
- 严格说来MATLAB是不能表示单位冲击信号的，Matlab提供了Dirac delta （狄拉克δ）函数dirac()，可用于矢量法和符号法表示信号
- 例：用符号运算，验证dirac函数的积分为一

```
syms t
d=dirac(t);
int(d,t,-inf,inf)
```

通信仿真中常见连续时间信号——单位冲击信号(2/2)

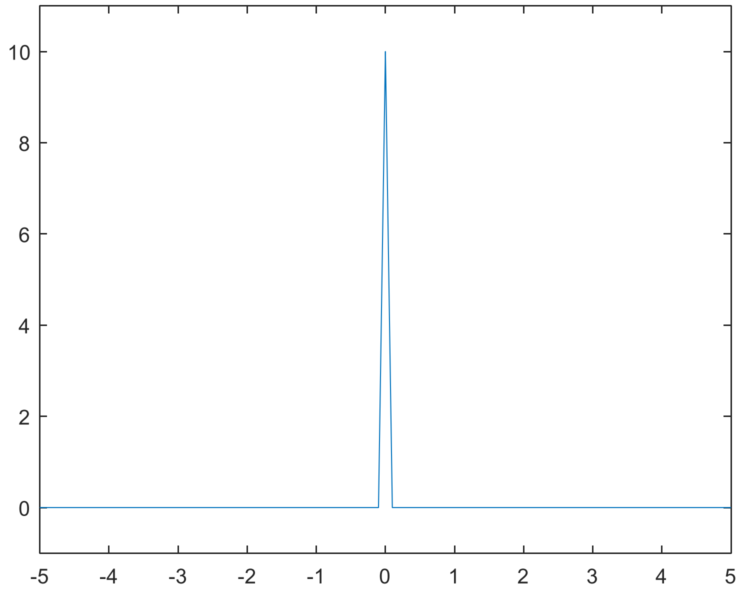
- 由于dirac函数在0处取无穷大，不利于绘图
  - ✓ 方法一：使用高斯调制正弦脉冲gauspuls函数（Gaussian-modulated sinusoidal RF pulse）

```
t=-5:.1:5;
s=gauspuls(t)
plot(t,s)
```



  - ✓ 方法二：根据画图的时间间隔，配合符号函数sign来画

```
p=0.1;
x=-5:p:5;
y=dirac(x);
y=1/p*sign(y);
plot(x,y);
```



通信仿真中常见连续时间信号——实值正弦信号

- 实值正弦信号的数学表达式为

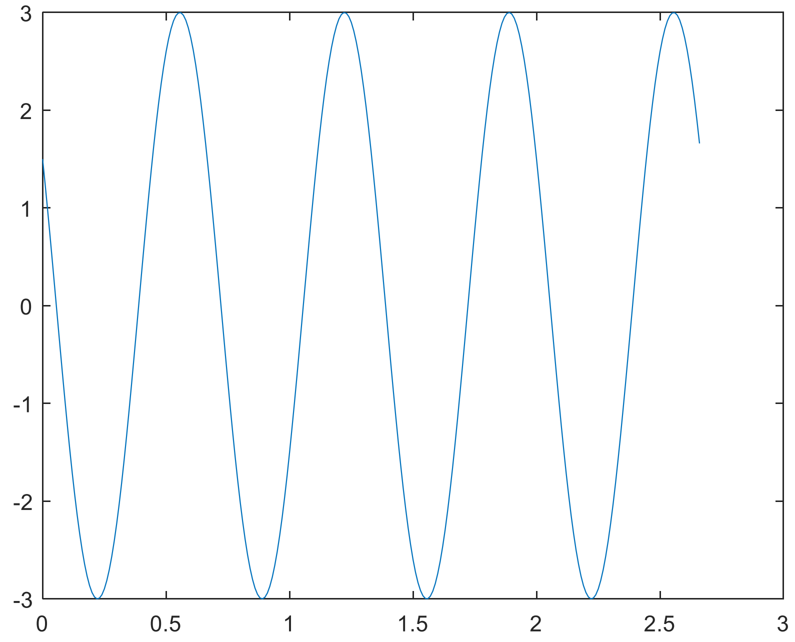
$$x(t)=A\cos(\Omega t+\theta)=A\cos\left(\frac{2\pi t}{T}+\theta\right)=A\cos(2\pi ft+\theta)$$

✓ A正弦信号的振幅；Ω是角频率，以rad/s表示；θ是相位（以rad表示）

✓ 正弦信号是周期性信号，其周期T由$2\pi/\Omega$ 给出；频率f（单位为赫兹）由$f=1/T$或$f=\Omega/2\pi$定义

- 例：绘制四个周期的 $x(t)=3\cos\left(3\pi t+\frac{\pi}{3}\right)$

```
A=3;
omega=3*pi;
thita=pi/3;
T= 2*pi/omega;
t=0:0.01:4*T;
x=A*cos(omega*t+thita);
plot(t,x);
```



通信仿真中常见连续时间信号——指数信号

- **指数信号的数学表达式为**

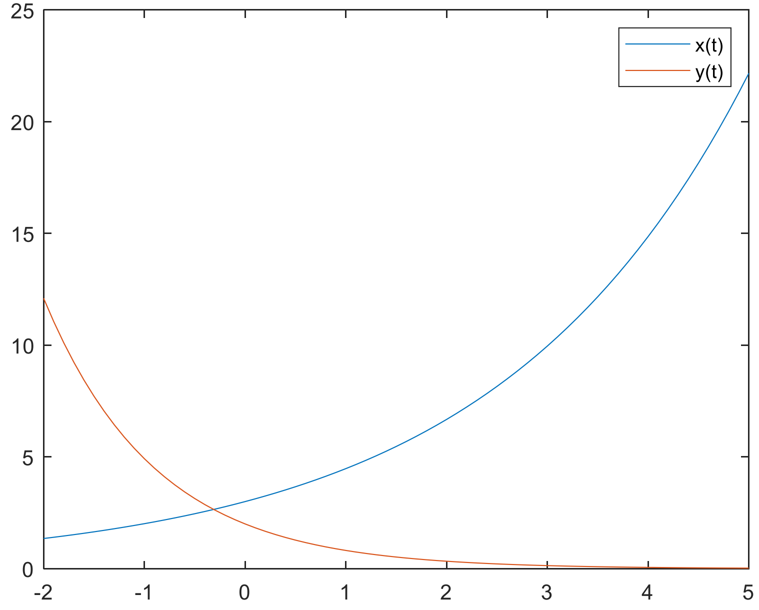
$$x(t) = Ae^{bt} = A\exp(bt)$$

✓ **若**$b>0$，$x(t)$**是一个递增函数；而**$b<0$，$x(t)$**是一个递减函数**

✓ **在**$b=0$**时，信号取**$x(t) = A$

- **例：在时间间隔[-2,5]内，绘制信号**$x(t) = 3e^{0.4t}$**和**$y(t) = 2e^{-0.9t}$

```
t=-2:.1:5;
x=3*exp(0.4*t);
y=2*exp(-0.9*t);
plot(t,x,t,y,':');
legend('x(t)','y(t)');
```



通信仿真中常见连续时间信号——复指数信号

- 复指数信号的数学表达式为

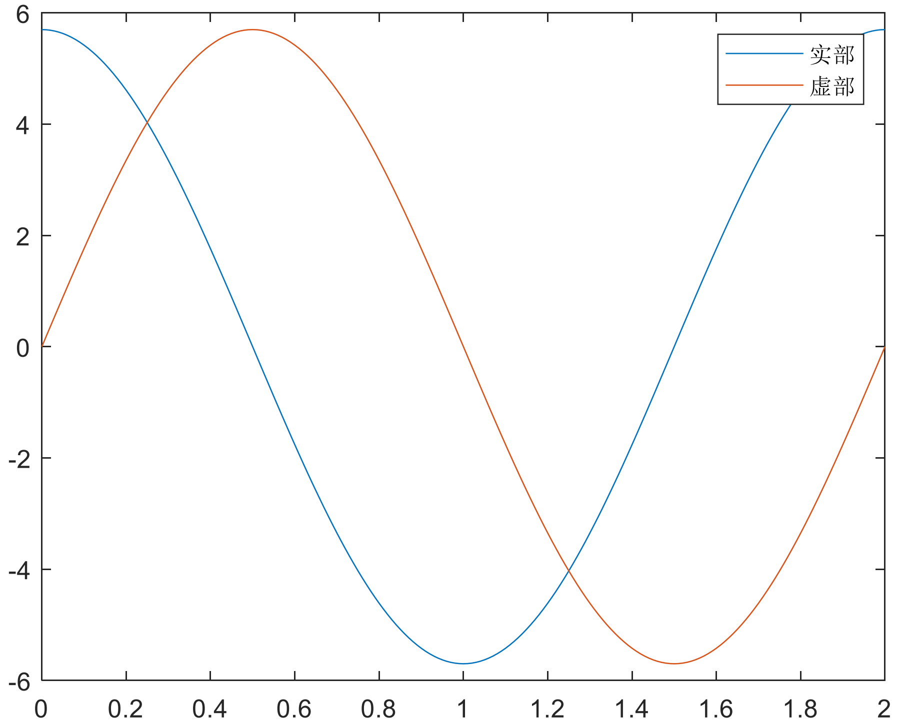
$$x(t) = Ae^{j(\Omega t+\theta)} = A\left(\cos\left(\Omega t+\theta\right)+j\sin\left(\Omega t+\theta\right)\right)$$

✓ 复指数信号是周期性信号，其周期T由$2\pi/\Omega$ 给出

- 例：绘制信号的实部和虚部

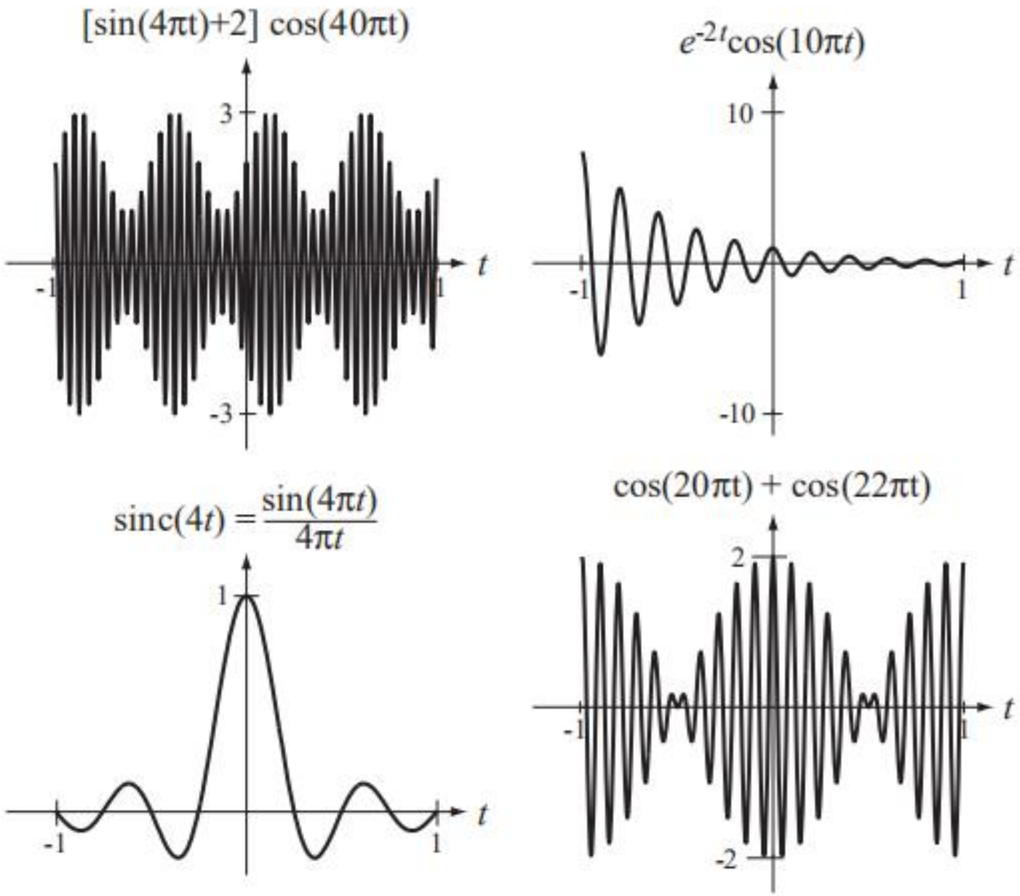
$$y(t) = 2e^{j\left(\pi t+\frac{\pi}{3}\right)}$$

```
t=0:0.01:2;
y_re=real(2*exp(1i*pi*t+pi/3));
y_im=imag(2*exp(1i*pi*t+pi/3));
plot(t,y_re,t,y_im);
legend('实部','虚部');
```



通信仿真中常见连续时间信号——复合函数信号

- 一个单一的数学函数可以完全描述一个信号，但通常一个函数不足以精确描述实际中的信号
- 通过复合两个或多个函数可实现任意信号，复合的方式通常是函数的和、差、积、商等



通信仿真中常见连续时间信号——方波脉冲信号

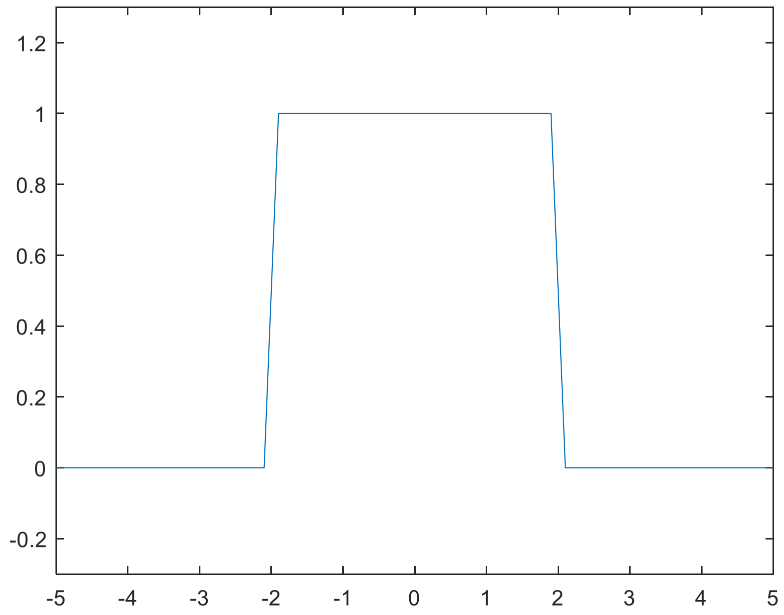
- **方波脉冲信号的数学定义为**

$$pT(t) = u\left(t+\frac{T}{2}\right) - u\left(t-\frac{T}{2}\right) = \begin{cases} 1, & -T/2 \le t \le T/2 \\ 0, & \text{elsewhere} \end{cases}.$$

  - ✓ **T是方波的脉宽**

- **在MATLAB中，可以通过heaviside函数的符合函数实现**
  - ✓ **例：绘制T=4的方波脉冲信号**

```matlab
t=-5:.1:5;
u1=heaviside(t+2);
u2=heaviside(t-2);
p=u1-u2;
plot(t,p);
ylim([-0.3 1.3]);
```



课堂小练习

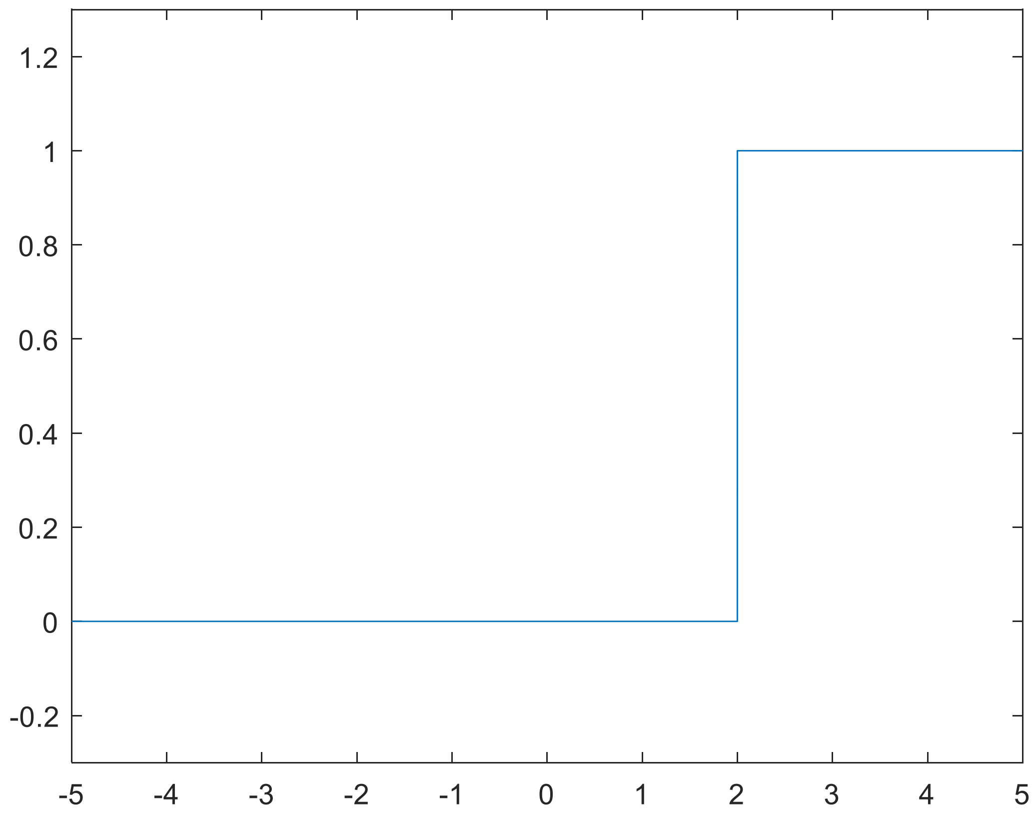
- 用矢量法和符号法绘制$u(t-2)$

- 方法1

```
t=-5:0.1:10;
u=heaviside(t-2);
plot(t,u);
ylim([-0.3 1.3]);
```

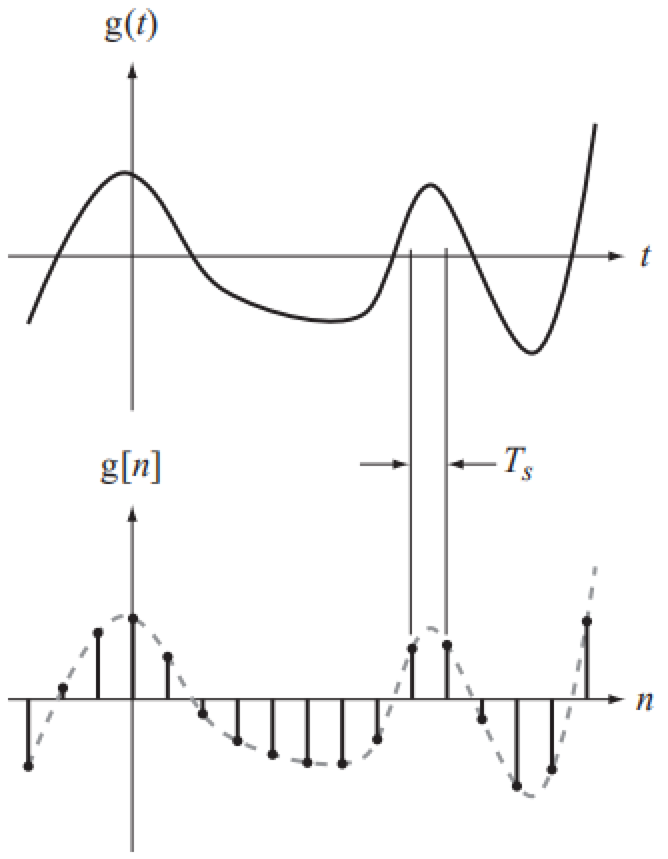
- 方法2

```
f=str2sym('heaviside(t-2)');
fplot(f);
ylim([-0.3 1.3]);
```



离散时间信号在MATLAB上的表示方法

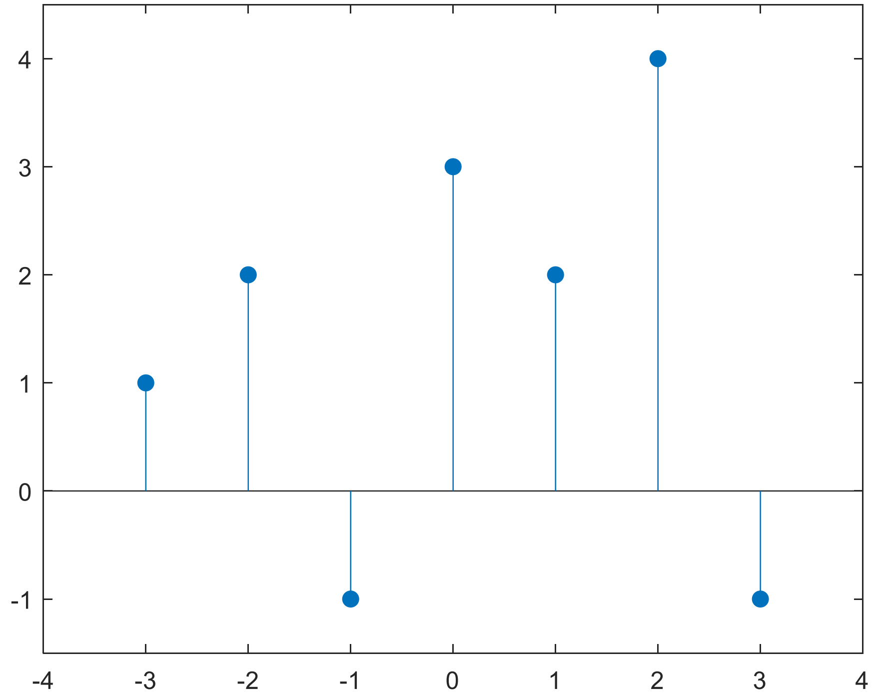
- 连续时间信号和离散时间信号的根本区别在于，离散时间信号中随时间推移出现的信号值是可数的，而连续时间信号中的信号值是不可数的
- 离散时间信号最常见的例子是通过对连续时间信号进行采样而获得的
- 离散时间信号用$x(k)$表示,其中变量k为整数，代表离散的采样时间点
  - ✓ $x(k) = \{\cdots, x(-2), x(-1), x(0), x(1), x(2), \cdots\}$
- 在MATLAB中,用一个向量f即可表示一个有限长度的序列,但这样的向量并没有包含其对应的时间序号信息，因此，完整地表示离散信号需要用两个向量
  - ✓ k = [-3,-2,-1,0,1,2,3] 或是k = -3:3;
  - ✓ x = [1,2,-1,3,2,4,-1]



离散时间信号的绘制

- 与连续时间信号不同，离散时间信号无法用符号运算来表示
- 由于在MATLAB中，矩阵的元素个数是有限的，因此无法表示无限序列
- 在绘制离散信号波形时，要使用专门绘制离散数据的stem命令，而不是plot命令
  - ✓ 对于定义的二向量f和k，用stem函数 stem(k,x,’filled’)
  - ✓ 例

```
k=-3:3;
x=[1,2,-1,3,2,4,-1];
stem(k,x,'filled');
axis([-4,4,-1.5,4.5]);
```



通信仿真中常见离散时间信号——单位脉冲序列

单位脉冲序列的定义为

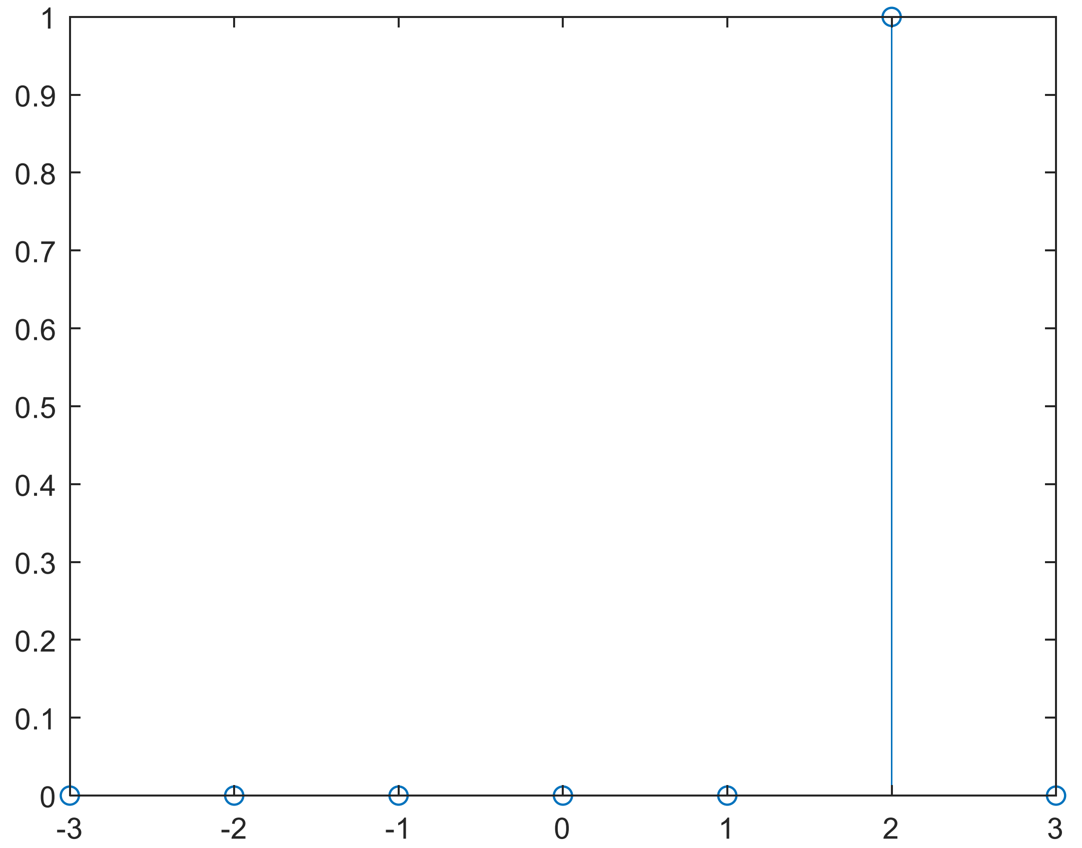
$$\delta[n-n_0]=\begin{cases}1, & n=n_0\\ 0, & n\neq n_0\end{cases}.$$

方法一：

```
n=-3:3;
d=gauspuls(n-2);
stem(n,d);
```

方法二：

```
n=-3:3;
d=(n==2);
stem(n,d);
```



通信仿真中常见离散时间信号——单位阶跃序列

- 单位阶跃序列的数学定义为

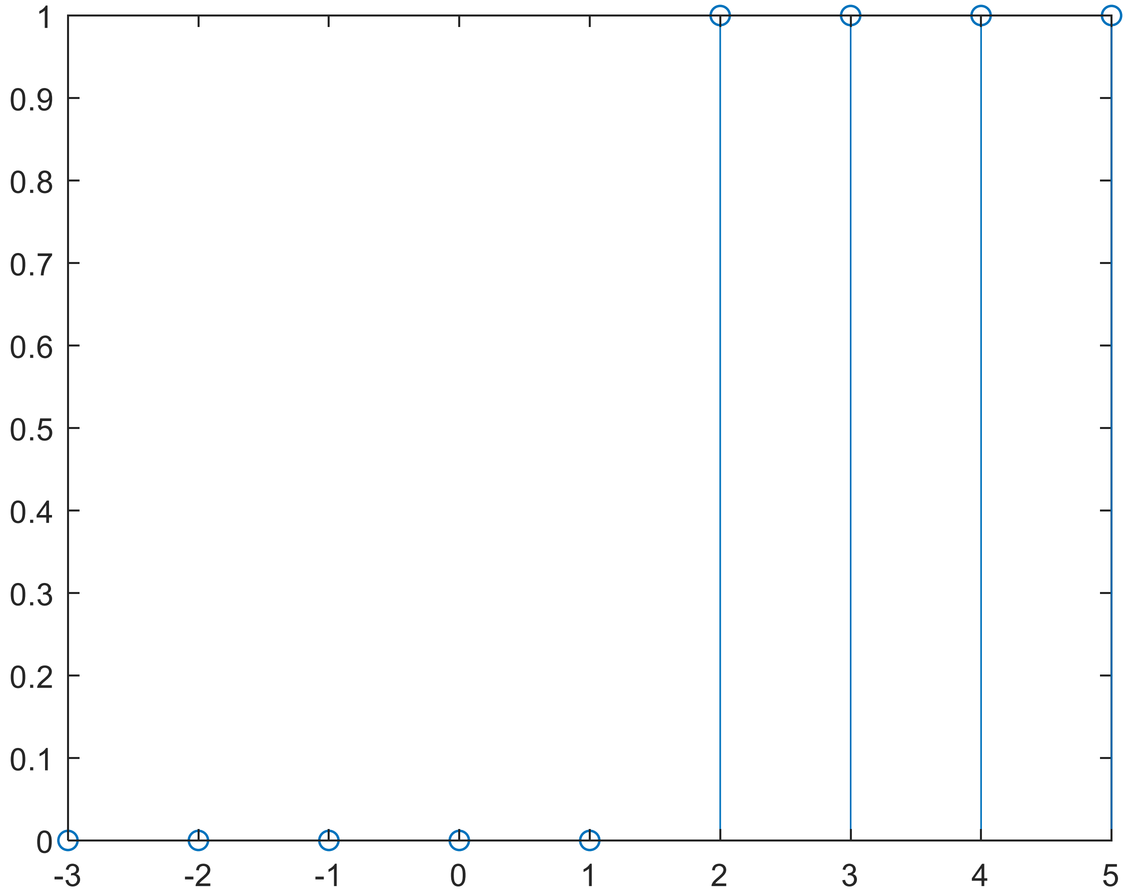
$$u[n-n_0]=\begin{cases}1, & n \ge n_0 \\ 0, & n < n_0\end{cases}$$

- 方法一

```
n=-3:5;
n0=2;
u=(n>=n0)
stem(n,u)
```

- 方法二

```
n0=2; n1=-3; n2=5;
n=n1:n2;
k=length(n);
u=zeros(1,k);
u(1, n0-n1+1:end)=1;
stem(n,u)
```



通信仿真中常见离散时间信号——实指数序列

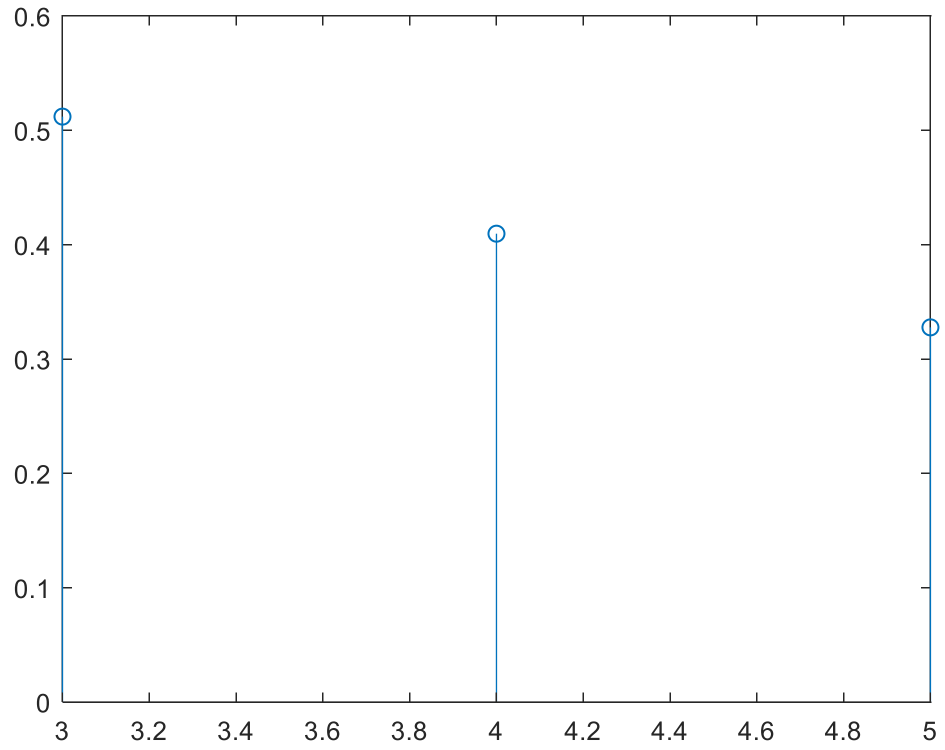
- **实指数序列的数学表达式为**

$$x[n] = a^n \qquad a \in \mathbb{R}$$

✓ $|a| > 1$为扩散序列， $|a| < 1$为收敛序列

- **示例**

```
n =3:5
a1 =0.8;
x1=a1.^n;
stem(n,x1);
```



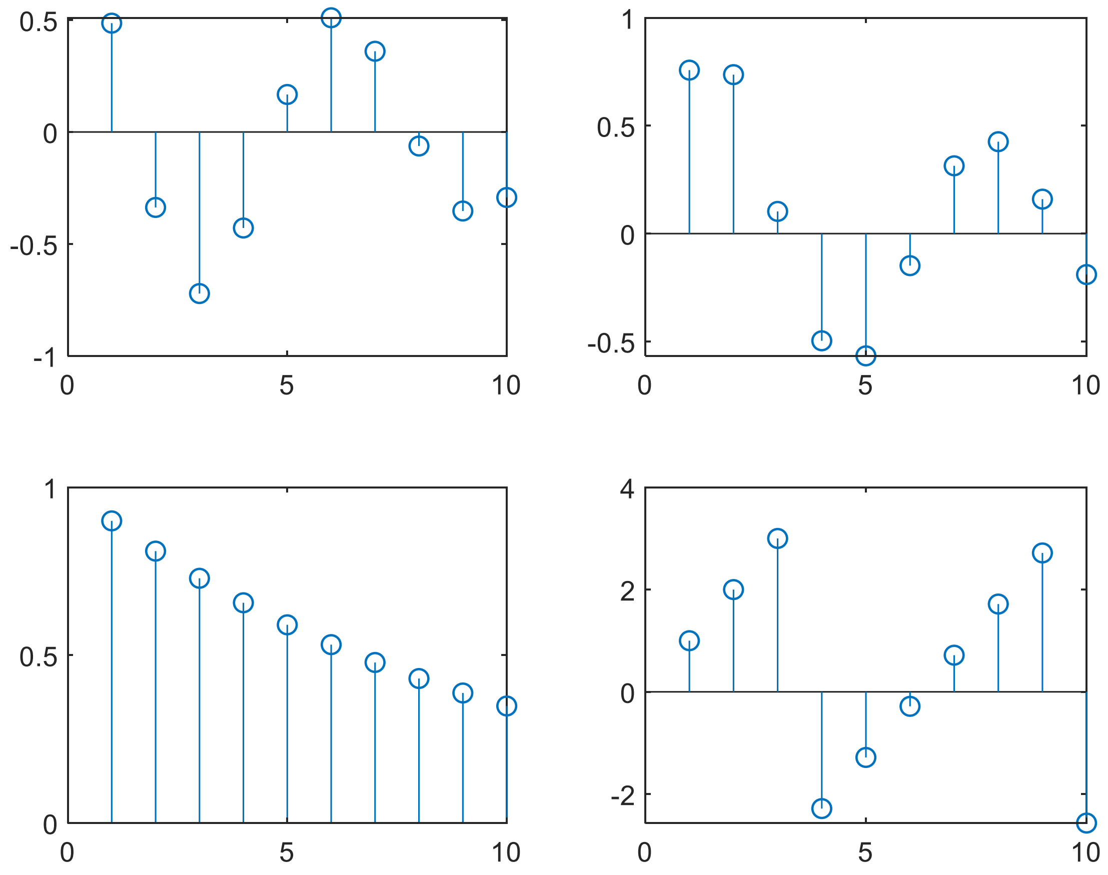
通信仿真中常见离散时间信号——复指数序列

□ 复指数序列的数学表达式为

$$x[n] = r^n e^{jwn} \qquad r \in \mathbb{R}$$

□ 示例

```matlab
n=1:10;
r=0.9;
w=1;
x=(r.^n).*exp(j*w*n);
subplot 221
stem(n,real(x));
subplot 222
stem(n,imag(x))
subplot 223
stem(n,abs(x))
subplot 224
stem(n,angle(x))
```



通信仿真中常见离散时间信号——正弦序列

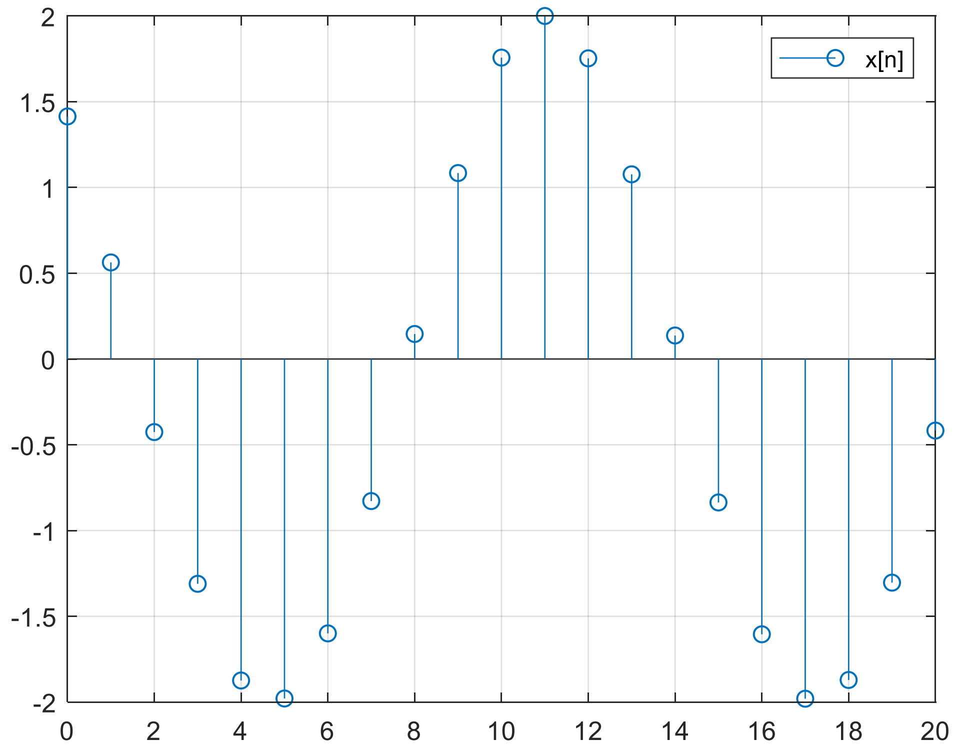
正弦序列的数学表达式为

$$x[n]=A\cos(\omega n+\varphi) \qquad x[n]=A\sin(\omega n+\varphi)$$

- ✓ 其中A是幅度，ω是角频率，φ是相位

示例

```
n=0:20;
x=2*cos(1/2*n+pi/4);
stem(n,x)
legend('x[n]')
grid
```



MATLAB提供的信号生成函数

函数	功能
chirp	扫频余弦波
diric	Dirichlet或周期 sinc函数
gauspuls	高斯正弦脉冲信号
gmonopuls	高斯单脉冲
pulstran	脉冲串
rectpuls	非周期方波
sawtooth	锯齿或三角波
sinc	Sinc函数
square	方波
tripuls	非周期三角波

连续时域信号的时域运算和变换（1/2）

- 相加：连续信号的相加， 是指两信号的对应时刻值相加
  - ✓ $x(t) = x_1(t) + x_2(t)$
  - ✓ 符号方式：x=x1+x2
  - ✓ 矢量方式：x=x1+x2
- 相乘：指两信号的对应时刻值相乘
  - ✓ $x(t) = x_1(t) \times x_2(t)$
  - ✓ 符号方式：x=x1*x2
  - ✓ 矢量方式：x=x1.*x2
- 位移：也称平移
  - ✓ 对连续信号$x(t)$，若有常数$t_0 > 0$，延时信号$x(t - t_0)$是将原信号沿正t轴方向平移时间t_0，而$x(t + t_0)$是将原信号沿负t 轴方向移动时间t_0
  - ✓ 符号方式：
    - ➢ y=subs(x,t,t-t0)； y=subs(x,t,t+t0)

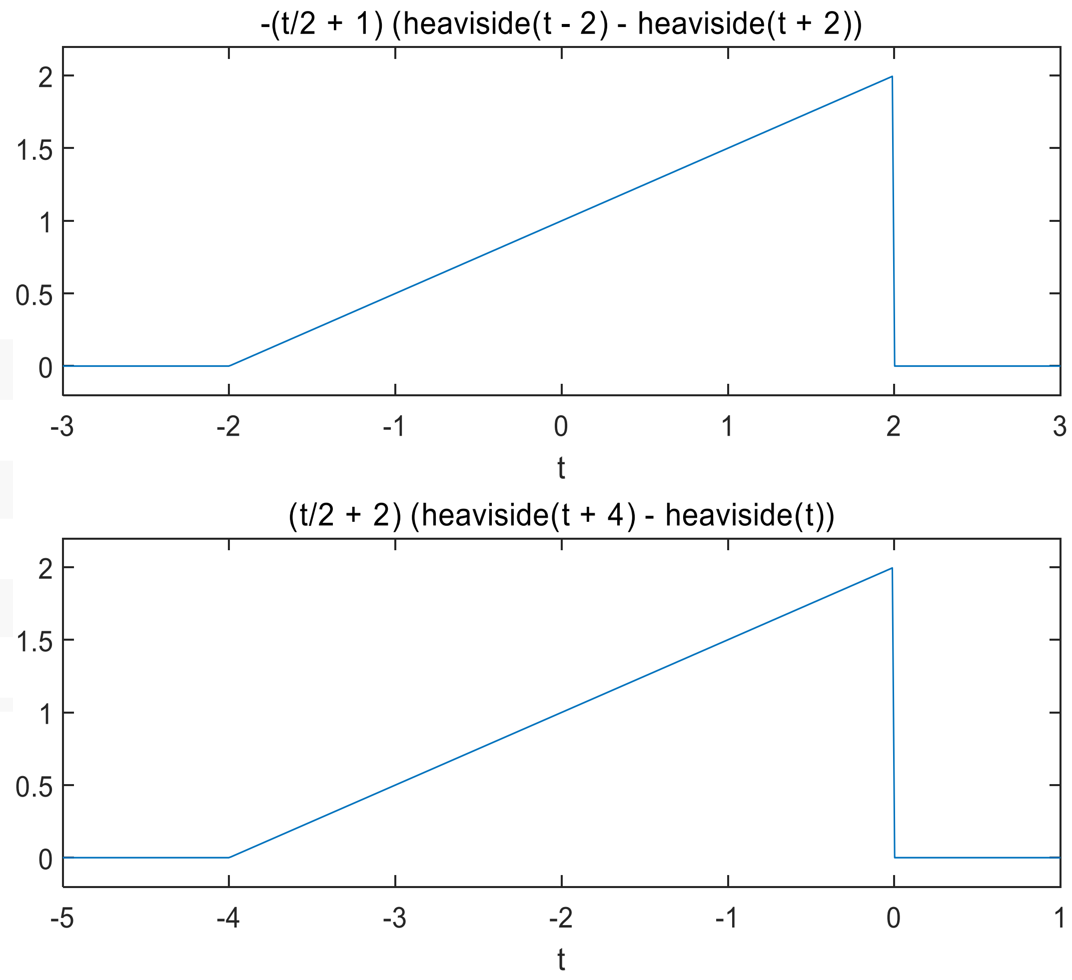
连续时域信号的时域运算和变换（2/2）

- 反折：$x'(t) = x(-t)$
  - ✓ 将信号以纵坐标为轴反折，即将信号$x(t)$中的自变量t换为$-t$
  - ✓ 符号方式：y=subs(x,t,-t)
- 尺度变换:$x'(t) = x(at)$
  - ✓ 将信号的横坐标进行展宽或压缩变换
  - ✓ 将信号$x(t)$中的自变量t换为at， 当$a > 1$时， 信号$x(at)$以原点为基准，沿横轴压缩到原来的$1/a$；当$0 < a < 1$时,信号$x(at)$将沿横轴展宽至原来的$1/a$倍
  - ✓ 符号方式： y=subs(x,t,a*t)
- 倒相:$x'(t) = -x(t)$
  - ✓ 将信号$x(t)$ 以横轴为对称轴对折得到$-x(t)$
  - ✓ y=-x

课堂小练习

- 设信号$f(t)=\left(1+\frac{t}{2}\right)[u(t+2)-u(t-2)]$,用Matlab求$f(t+2)$, 并绘出其时域波形
  - ✓ ex5TimeSignal.m
  - ✓ 提示：str2sym、subs、fplot

```
clear
syms t
f=str2sym('(t/2+1)*(heaviside(t+2)-heaviside(t-2))')
subplot(2,1,1),ezplot(f,[-3,3])
y1=subs(f,t,t+2)
subplot(2,1,2),ezplot(y1,[-5,1])
```



离散序列的时域运算和变换(1/3)

- 对离散序列来说，序列相加、相乘是将两序列对应时间序号的值逐项相加或相乘，平移、反折、及倒相变换与连续信号的定义完全相同
- 但与连续信号不同， 在MATLAB中离散序列的时域运算和变换不能用符号运算来实现， 必须用向量表示的方法
- 离散序列相加
  - ✓ $x[n]=x1[n]+x2[n]$
  - ✓ simSeqAdd.m

```matlab
function [x,n] = SeqAdd(x1,n1,x2,n2)
% 实现序列相加 x[n] = x1[n]+x2[n]
% 输入参数
%  x1 : 定义在时间矢量 n1 上的序列 1
%  x2 : 定义在时间矢量 n2 上的序列 2 (n2 可与 n1 不同)
% 输出参数
%  x: 定义在时间矢量 n 上的序列和

n=min(min(n1),min(n2)):max(max(n1),max(n2)); % 生成序列和的时间矢量
s1=zeros(1,length(n));s2=s1;                  %根据序列和的时间矢量，分配存储空间
s1(((n>=min(n1))&(n<=max(n1))==1))=x1;        % 将 x1 按照时间矢量 n 复制到 s1
s2(((n>=min(n2))&(n<=max(n2))==1))=x2;        % 将 x2 按照时间矢量 n 复制到 s2
x=s1+s2;                                      % 时间矢量对齐后序列相加
end
```

离散序列的时域运算和变换(2/3)

- **离散序列相乘**
  - ✓ $x[n] = x1[n] * x2[n]$
  - ✓ **simSeqMult.m**

```matlab
function [x,n] = SeqMult(x1,n1,x2,n2)
% 实现序列相乘 x[n] = x1[n]+x2[n]
% 输入参数
%  x1 : 定义在时间矢量 n1 上的序列 1
%  x2 : 定义在时间矢量 n2 上的序列 2 (n2 可与 n1 不同)
% 输出参数
%  x: 定义在时间矢量 n 上的序列积

n=min(min(n1),min(n2)):max(max(n1),max(n2)); % 生成序列积的时间矢量
s1=zeros(1,length(n));s2=s1;                 %根据序列积的时间矢量，分配存储空间
s1(((n>=min(n1))&(n<=max(n1))==1))=x1;        % 将 x1 按照时间矢量 n 复制到 s1
s2(((n>=min(n2))&(n<=max(n2))==1))=x2;        % 将 x2 按照时间矢量 n 复制到 s2
x=s1.*s2;                                     % 时间矢量对齐后序列相乘
end
```

- **离散序列反折**
  - ✓ $x[n] = x1[-n]$
  - ✓ **simSeqFold.m**

```matlab
function [x,n] = SeqFold(x1,n)
% 实现序列反折 x[n] = x1[-n]
% 输入参数
%  x1 : 定义在时间矢量 n 上的序列
% 输出参数
%  x: 定义在时间矢量 n 上的反折后的序列

x = fliplr(x1);
n = -fliplr(n);
end
```

离散序列的时域运算和变换(3/3)

- **离散序列平移**
  - ✓ $x[n] = x1[n - n0]$
  - ✓ **simSeqShift.m**

```
function [x,n] = SeqShift(x1,n1,n0)
% 实现序列平移 x[n] = x1[n1-n0]
% 输入参数
%  x1 : 定义在时间矢量 n1 上的序列
% 输出参数
%  x: 定义在时间矢量 n 上的平移后的序列

n = n1+n0;      %调整时间矢量
x = x1;         %信号矢量原样复制
end
```

- **离散序列倒相**
  - ✓ $x[n] = -x1[n]$
  - ✓ **simSeqRev.m**

```
function [x,n] = SeqRev(x1,n)
% 实现序列倒相 x[n] = -x1[n]
% 输入参数
%  x1 : 定义在时间矢量 n 上的序列
% 输出参数
%  x: 定义在时间矢量 n 上的倒相后的序列

x =-x1;
end
```

信号的能量和功率（1/2）

- 连续时间信号
  - ✓ 能量：

$$E_x=\int_{-\infty}^{\infty}|x(t)|^2dt$$

  - ✓ 功率：

一般信号

$$P_x=\lim_{T\to\infty}\frac{1}{T}\int_{-T/2}^{T/2}|x(t)|^2dt$$

周期信号

$$P_x=\frac{1}{T}\int_{t_0}^{t_0+T}|x(t)|^2dt=\frac{1}{T}\int_T|x(t)|^2dt$$

- 离散时间信号
  - ✓ 能量：

$$E_x=\sum_{n=-\infty}^{\infty}|x(n)|^2$$

  - ✓ 功率：

一般信号

$$P_x=\lim_{N\to\infty}\frac{1}{2N+1}\sum_{n=-N}^{n=+N}|x(n)|^2$$

指定$N0\sim N1$区间的信号

$$P_x=\lim_{N\to\infty}\frac{1}{N_1-N_0+1}\sum_{n=N_0}^{n=N_1}|x(n)|^2$$

信号的能量和功率（2/2）

□ 有限长序列的能量与功率

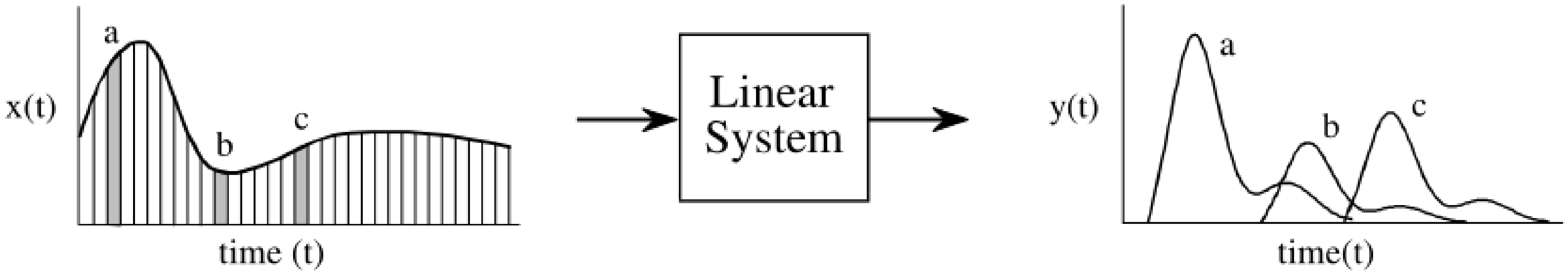
✓ simSeqEngPower

```
% 演示信号能量和功率计算
close all;clearvars;

Fsam=100;
Tsam=1/Fsam;
t=0:Tsam:10-Tsam;
x=cos(2*pi*10*t);

[E,P]=SeqEngPower(x,Tsam)
```

```
function [E,P]=SeqEngPower(x,ts)
% 有限长度序列的能量与功率
% 输入参数
%  x ：信号序列
%  ts：信号序列采样周期
% 输出参数
%  E:  有限长度序列的能量
%  P:  有限长度序列的功率
 E=(sum(abs(x).^2))*ts;
 P=(sum(abs(x).^2))/length(x);
end
```

理解信号卷积

- 卷积的概念与线性系统的输入和输出有关
  - ✓ 首先，输入信号可以分解成一组脉冲，每个脉冲都可以看作是一个缩放和移位的delta函数



  - ✓ 其次，每个脉冲产生的输出是脉冲响应的缩放和移位版本
  - ✓ 第三，通过将这些标度和移位的脉冲响应相加，可以找到整个输出信号
- 如果我们知道一个系统的脉冲响应，那么我们就可以计算出任何可能的输入信号的输出，这意味着我们对这个系统了如指掌

连续时域信号的卷积

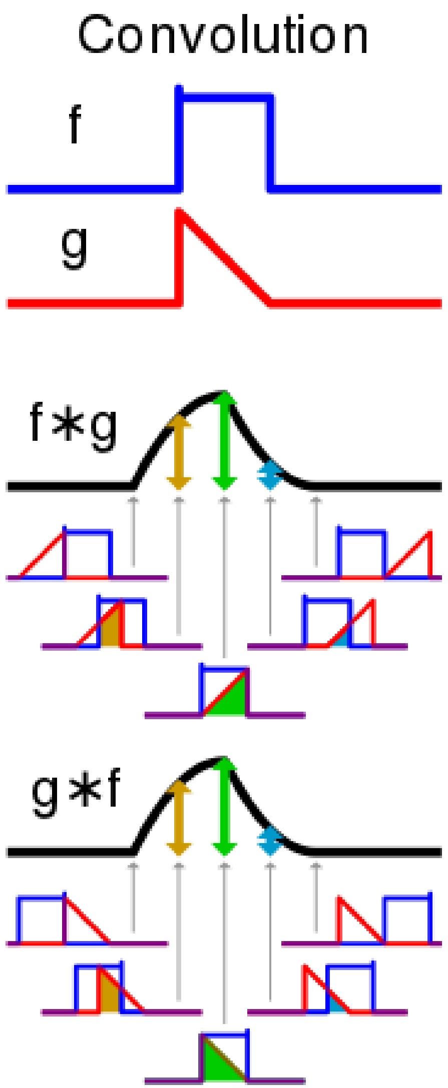
- 为了研究建立模型，数学家把这一点发挥到极致，把输入段变得无限狭窄，把情况变成一个微积分问题
  - ✓ 这个积分很不简单，它描述了一个动态过程，表达了系统不断衰减同时又不断受到激励的综合结果
- 研究结果

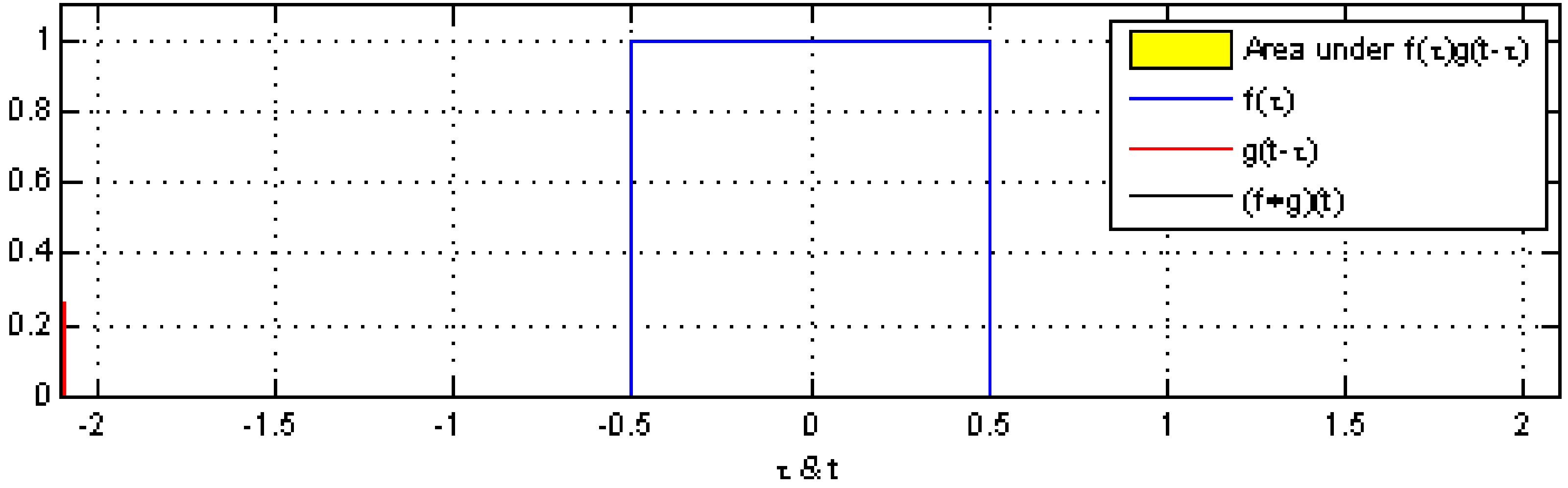
$$y(t)=f(t)*g(t)=\int_{-\infty}^{+\infty}f(\tau)g(t-\tau)d\tau=\int_{-\infty}^{+\infty}f(t-\tau)g(\tau)d\tau$$

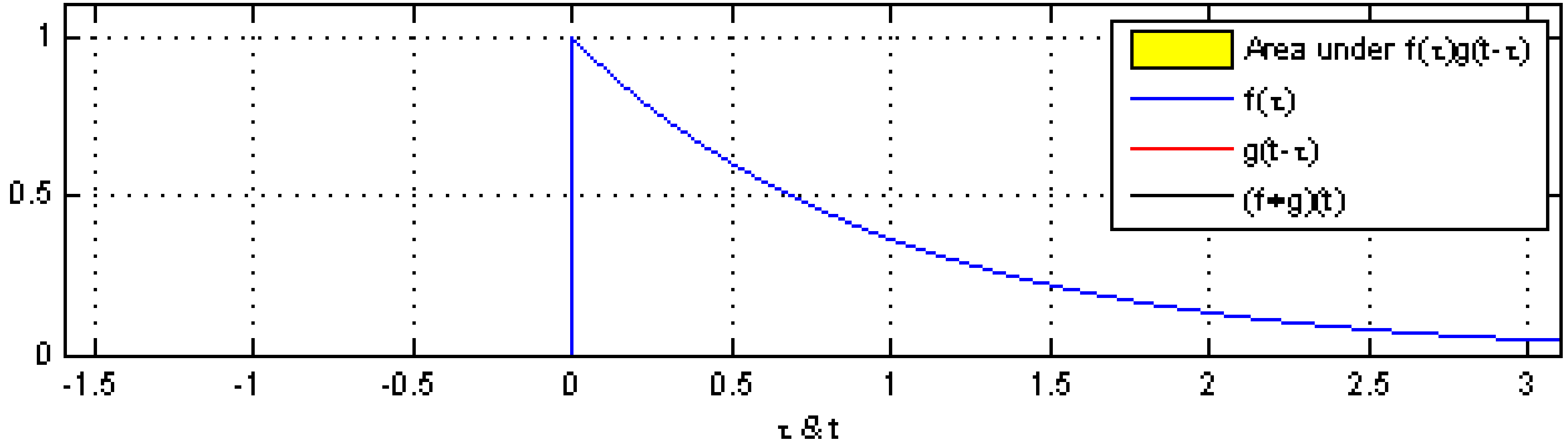
  - ✓ 大致方法是将激励信号进行反转（这是最不能让人理解的部分，将一个以时间为自变量的信号反转，难道是要让时光倒流？），然后与系统相乘，再积分
  - ✓ 请你记住：不要试图通过公式的计算过程来了理解卷积的物理意义。我们要做的是代入公式计算就好了
- 数学家们从纯数学函数的角度出发，给这一算法起了一个貌似形象的名字“卷积”

连续时间信号卷积计算的图形化解释

$$y(t) = f(t) * g(t) = \int_{-\infty}^{+\infty} f(\tau)g(t-\tau)d\tau = \int_{-\infty}^{+\infty} f(t-\tau)g(\tau)d\tau$$



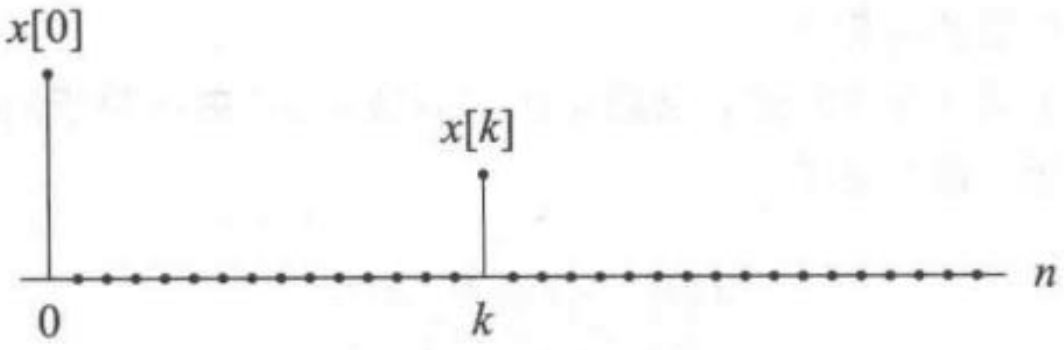




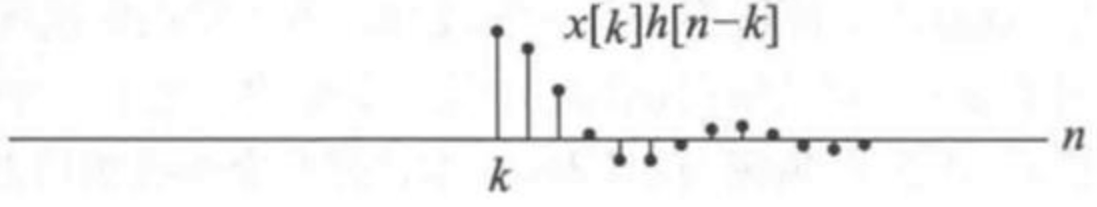
离散时间序列的卷积和

- 在MATLAB中，使用符号计算不易直接计算时间连续信号的卷积，通常对连续时间信号进行采样，然后转化为离散时间序列的卷积和
- 离散时间序列$x(k)$和$h(k)$的卷积和定义为

$$y(k)=x(k)*h(k)=\sum_{i=-\infty}^{\infty}x(i)h(k-i)$$

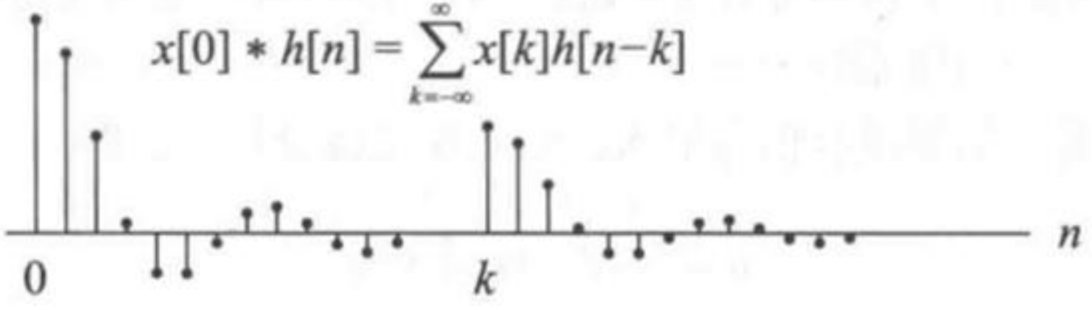


(a) 由两个脉冲组成的输入信号



(c) 第二个脉冲的响应，相比第一个脉冲的响应，位置向右移动了 k

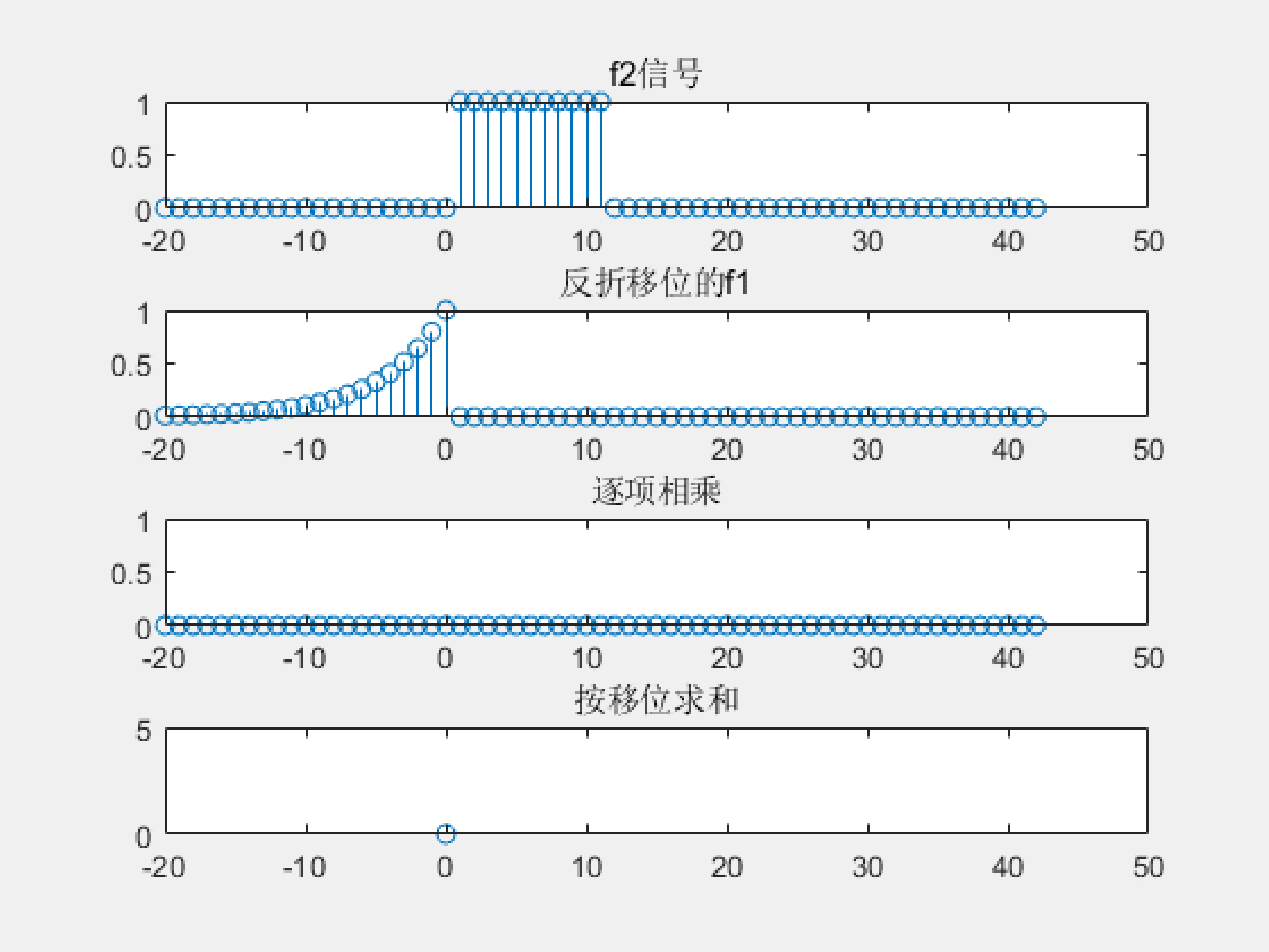
(b) 第一个脉冲的响应



(d) 叠加起来就是离散卷积运算

离散时间信号卷积和计算的图形化解释

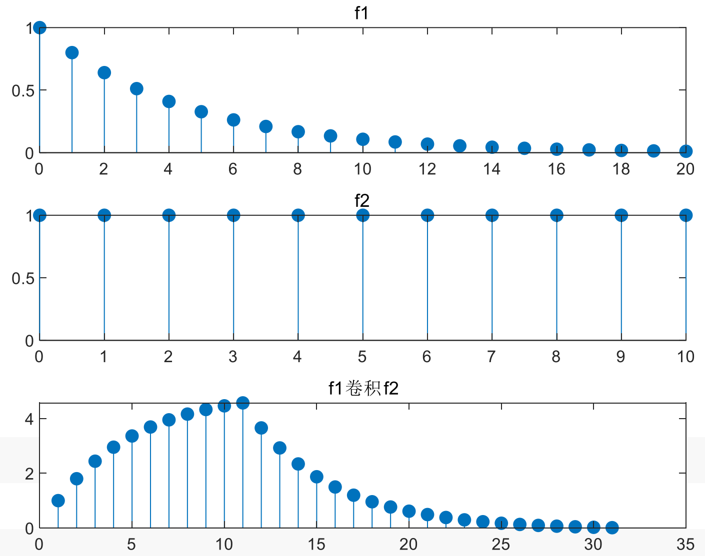
$$y(k)=x(k)*h(k)=\sum_{i=-\infty}^{\infty}x(i)h(k-i)$$



ex3TimeSignal.m

简单序列的卷积和实现

- 在MATLAB中，卷积函数conv默认两个信号的时间序列从n＝0开始，y对应的时间序号也从n＝0开始
  - ✓ f=conv(f1,f2)
  - ✓ f1:序列$f_1(k)$的非零样值点向量
  - ✓ f2:序列$f_2(k)$的非零样值点向量
  - ✓ f:序列卷积后的所有非零样值点行向量



- 例：知两个信号序列：f1＝0.8^n (0<＝n<＝20)、f2＝u(n) (0<＝n<＝10),求两个序列的卷积和
  - ✓ ex3conv.m

```
n1=0:20;
f1=0.8.^n1;
n2=0:10;
f2=(n2>=0);
f3=conv(f1,f2);
subplot(3,1,1),
stem(n1,f1,'filled');%绘制脉冲杆图，且圆点处用实心圆表示
title('f1')
subplot(3,1,2)
stem(n2,f2,'filled');%绘制脉冲杆图，且圆点处用实心圆表示
title('f2')
subplot(3,1,3)
stem(f3,'filled');%绘制脉冲杆图，且圆点处用实心圆表示
title('f1卷积f2')
```

一般序列的卷积和实现

- 如果信号不是从0开始，则编程时必须用两个数组分别确定波形的幅度样值和其对应的时间向量

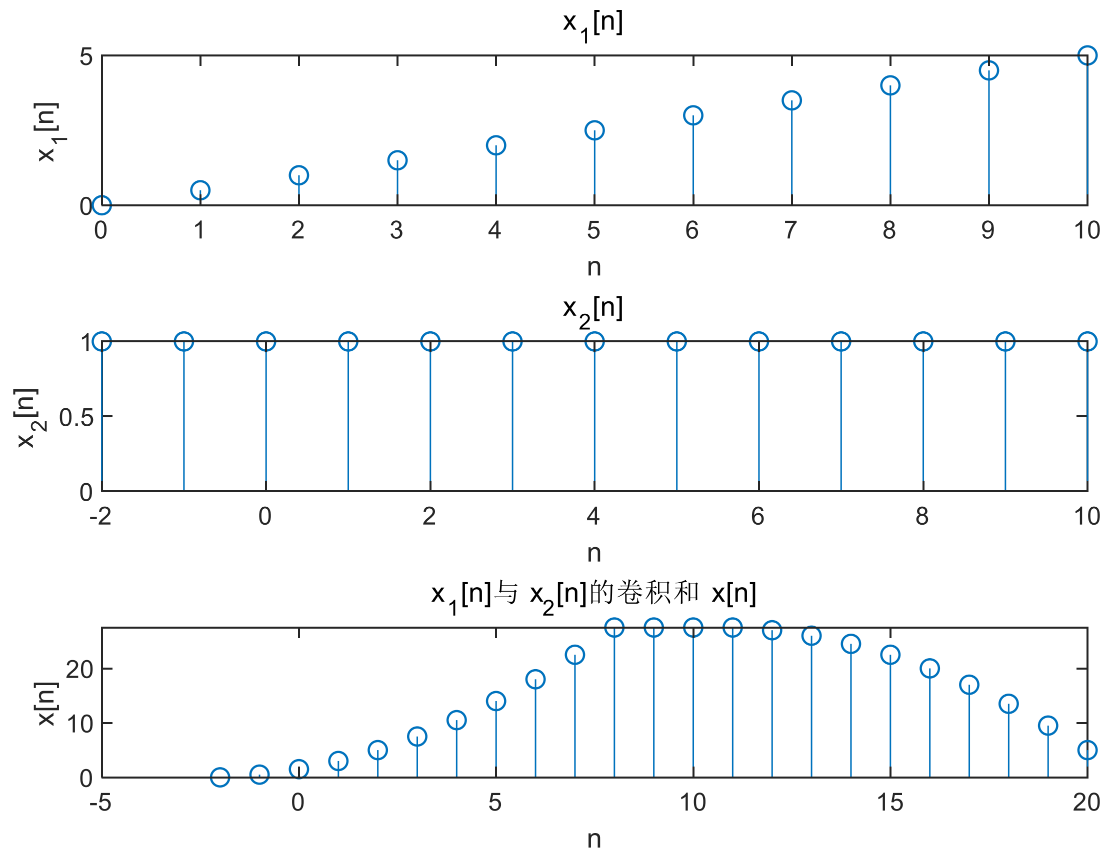
- 设序列$x_1(k)$在区间$n_1 \sim n_2$非零，$x_2(k)$在区间$m_1 \sim m_2$非零，则$x_1(k)$的时域宽度为$L_1 = n_2 - n_1 + 1$，$x_2(k)$的时域宽度为$L_2 = m_2 - m_1 + 1$
- 由卷积和的定义可得，序列$x(k) = x_1(k) * x_2(k)$的非零区间从$n_1 + m_1$开始，长度为$L = L_2 + L_1 - 1$,所有非零区间为$n_1 + m_1 \sim n_1 + m_1 + L - 1$
  - ✓ simTimeDiscreteConv.m
  - ✓ TimeDiscreteConv.m

```
function [x,n]=TimeDiscreteConv(x1,n1,x2,n2)
%计算序列的卷积 x[n]=x1[n]*x2[n]
% 输入参数
% x1: x1[n]的非零样值向量及其对应的时间矢量 n1;
% x2: x2[n]的非零样值向量及其对应的时间矢量 n2;
% 输出参数
% x: 卷积结果序列 x[n]及其对应的时间矢量 n

x=conv(x1,x2);  %计算序列 x1 与 x2 的卷积和 x
k0=n1(1)+n2(1); %计算序列 x 非零样值的起点位置
L=length(x1)+length(x2)-1; %计算卷积和 x 的非零样值的宽度
n=k0:k0+L-1;    %确定卷积和 x 非零样值的序号向量
end
```

示例：一般序列的卷积

□ 程序：simTimeDiscreteConv.m

```
%一般序列的卷积计算演示
close all;clearvars;

n1=0:10;
x1=0.5*n1;
n2=-2:10;
x2=ones(1,length(n2));
[x,n]=TimeDiscreteConv(x1,n1,x2,n2)

subplot(3,1,1)
stem(n1,x1) %在子图 1 绘序列 x1[n]时域波形图
title('x_1[n]')
xlabel('n')
ylabel('x_1[n]')
subplot(3,1,2)
stem(n2,x2) %在图 2 绘序列 x2[n]时波形图
title('x_2[n]')
xlabel('n')
ylabel('x_2[n]')
subplot(3,1,3)
stem(n,x); %在子图 3 绘序列 x[n]的波形图
title('x_1[n]与 x_2[n]的卷积和 x[n]')
xlabel('n')
ylabel('x[n]')
```



连续时域信号卷积的计算方法

- **定义**$x(t)=x_1(t)*x_2(t)=\int_{-\infty}^{\infty}x_1(\tau)x_2(t-\tau)d\tau$
  - ✓ **由数值积分的原理，上述定义可改写为**

$$x(t)=\int_{-\infty}^{\infty}x_1(\tau)x_2(t-\tau)d\tau=\lim_{\Delta\to 0}\sum_{k=-\infty}^{\infty}x_1(k\Delta)x_2(t-k\Delta)\Delta$$

- **当Δ足够小，可以用离散卷积和近似求连续卷积，并在卷积和基础上乘以**

$$\Delta f(n\Delta)=\Delta\sum_{k=-\infty}^{\infty}x_1(k\Delta)x_2(t-k\Delta)$$

- **计算步骤**
  - ✓ **将连续$x_1(t)$和$x_2(t)$以时间隔Δ进行取样， 得到离散序列$x_1(k\Delta)$和$x_2(k\Delta)$**
  - ✓ **构造与 $x_1(k\Delta)$和$x_2(k\Delta)$相对应的时间向量 k_1和k_2（注意,此时k_1和k_2不再是整数， 而是取样时间间隔Δ的整数倍的时间间隔点）**
  - ✓ **调用conv()函数计算卷积积分$x(t)$的近似向量$x(k\Delta)$**
  - ✓ **构造$x(k\Delta)$对应的时间向量k**

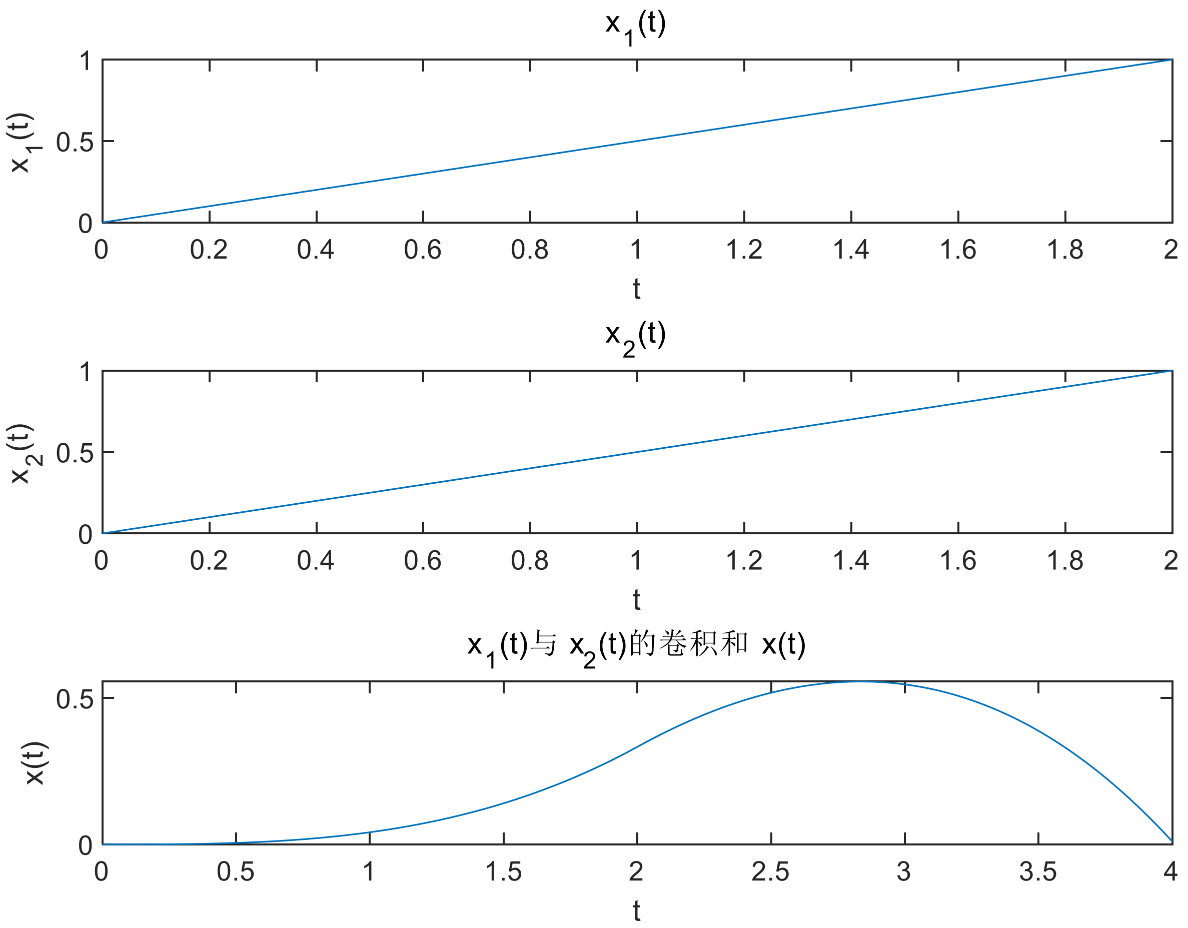
```
function [x,n]=TimeContinuousConv(x1,n1,x2,n2,p)
% 计算时间连续信号抽样的卷积
% 输入参数
% x1: x1[n]的非零样值向量及其对应的时间矢量 n1;
% x2: x2[n]的非零样值向量及其对应的时间矢量 n2;
% p:  采样时间间隔
% 输出参数
% x: 卷积结果序列 x(n)及其对应的时间矢量 n

[x,n]=TimeDiscreteConv(x1,n1,x2,n2);
x=x*p;
n=n*p;
end
```

示例：连续时域信号卷积计算的MATLAB实现

◻ 代码实现

✓ simTimeDiscreteConv.m

```
%连续时间信号卷积的数值计算演示
close all;clearvars; clc;

p=0.01;
n1=0:p:2;
x1=0.5*n1;
n2=n1;
x2=x1;
[x,n]=TimeContinuousConv(x1,n1,x2,n2,p);

subplot(3,1,1)
plot(n1,x1) %在子图 1 绘序列 x1(k)时域波形图
title('x_1(t)')
xlabel('t')
ylabel('x_1(t)')
subplot(3,1,2)
plot(n2,x2) %在图 2 绘序列 x2(k)时波形图
title('x_2(t)')
xlabel('t')
ylabel('x_2(t)')
subplot(3,1,3)
plot(n,x); %在子图 3 绘序列 x(k)的波形图
title('x_1(t)与 x_2(t)的卷积和 x(t)')
xlabel('t')
ylabel('x(t)')
```



课后作业

- **信号的相关系数：描述两个信号之间相似性**

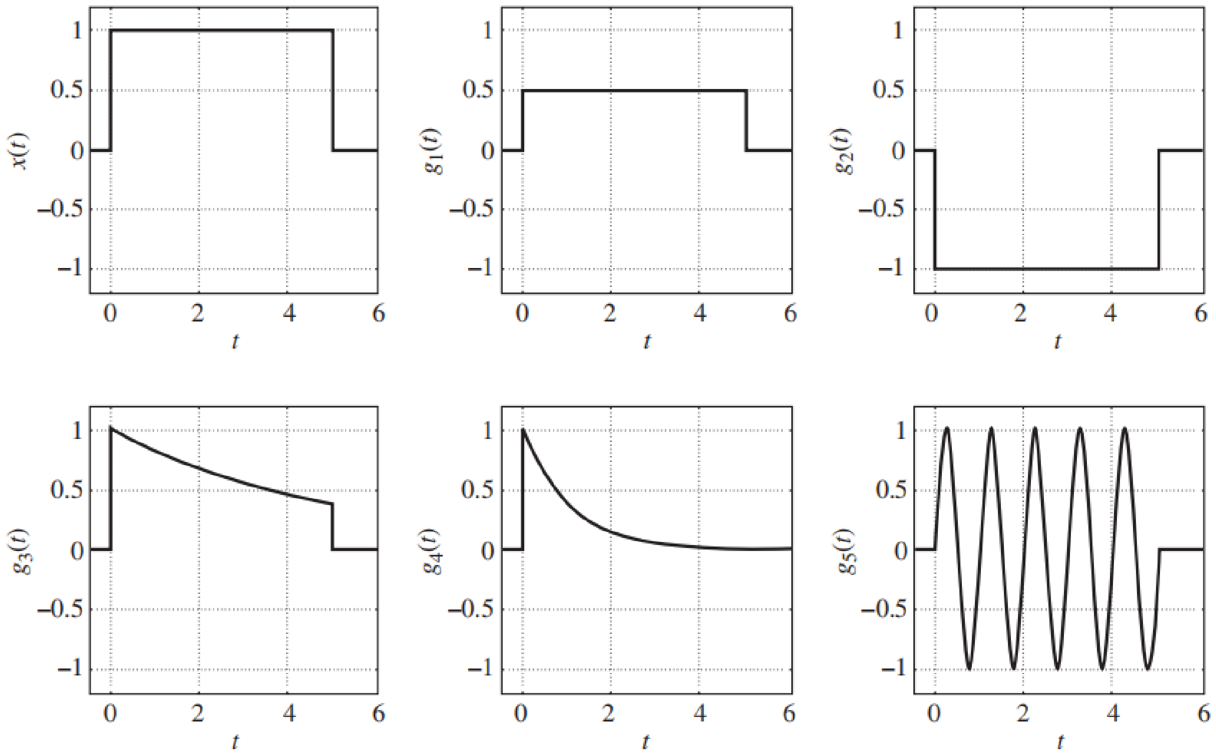
$$\rho = \cos\theta = \frac{<\mathbf{g},\ \mathbf{x}>}{||\mathbf{g}||\ ||\mathbf{x}||}$$

  - ✓ $|\rho| \le 1$，$\rho = 0$:两信号之间正交
  - ✓ 对复信号，相关系数为

$$\rho = \frac{1}{\sqrt{E_g E_x}} \int_{-\infty}^{\infty} g(t) x^*(t)\, dt$$

- **请编写程序计算信号之间的相关系数**
  - ✓ 主程序：hw1.m
  - ✓ 子函数：计算相关系数

```
function rho=correlation_coefficient(x,g,dt)
% 功能：计算信号之间的相关系数
% 输入：x信号样点，g信号样点，dt采样间隔
% 输出：rho 相关系数
```



$g_1\ t = 0.5x\ t$

$g_2\ t = -x\ t$

$g_3\ t = e^{-\frac{t}{5}}x\ t$

$g_4\ t = e^{-t}x\ t$

$g_5\ t = \sin\ 2\pi t\ \ x\ t$

有问题，随便问！