

课程名称：信号与系统

学生信息：20337268 张文沁

实验内容：

## 实 验 四

给定一个连续时间信号：
$$f(t) = \begin{cases} \frac{1}{2}[1 + \cos(t)], & 0 \leq |t| \leq \pi \\ 0, & |t| > \pi \end{cases},$$

(1) 画出这个信号的波形和它的频谱。

(2) 当采样周期分别满足  $T=1$ ,  $T=\pi/2$ ,  $T=2$  时, 分别画出三个采样信号  $f_p(n)$  和他们各自的频谱, 并对结果给出解释。

(3) 使用截止频率  $\omega_c = 2.4$  的理想低通滤波器从  $f_p(n)$  重建信号  $f_r(t)$ 。当采样周期分别是  $T=1$  和  $T=2$  时, 画出重建信号  $f_r(t)$  及其频谱, 并画出  $f_r(t)$  和原始信号  $f(t)$  之间的绝对误差, 并对结果给出解释。

### 实验过程：

1. 绘制波形和频谱：

波形的绘制：设置分段函数和使用 plot() axis() 函数即可实现

频谱的绘制：傅里叶变换绘制频谱

代码及注释：

1. 波形图

```
close all;
clc;
x=-3*pi:0.1*pi:3*pi;
y=0.*(x<-pi)+(1+cos(x))/2.*(x>=-pi & x<=pi)+0.*(x>pi);
plot(x,y)
```

2. 频谱图：

```
sym t          %声明变量
R=pi/20;       %取样间隔
t = -3*pi:R:3*pi; %三角脉冲函数x取值范围
Ft = sym(0.*(t<-pi)+(1+cos(t))/2.*(t>=-pi & t<=pi)+0.*(t>pi));
w1=10*pi;      %取要计算的频率范围
```

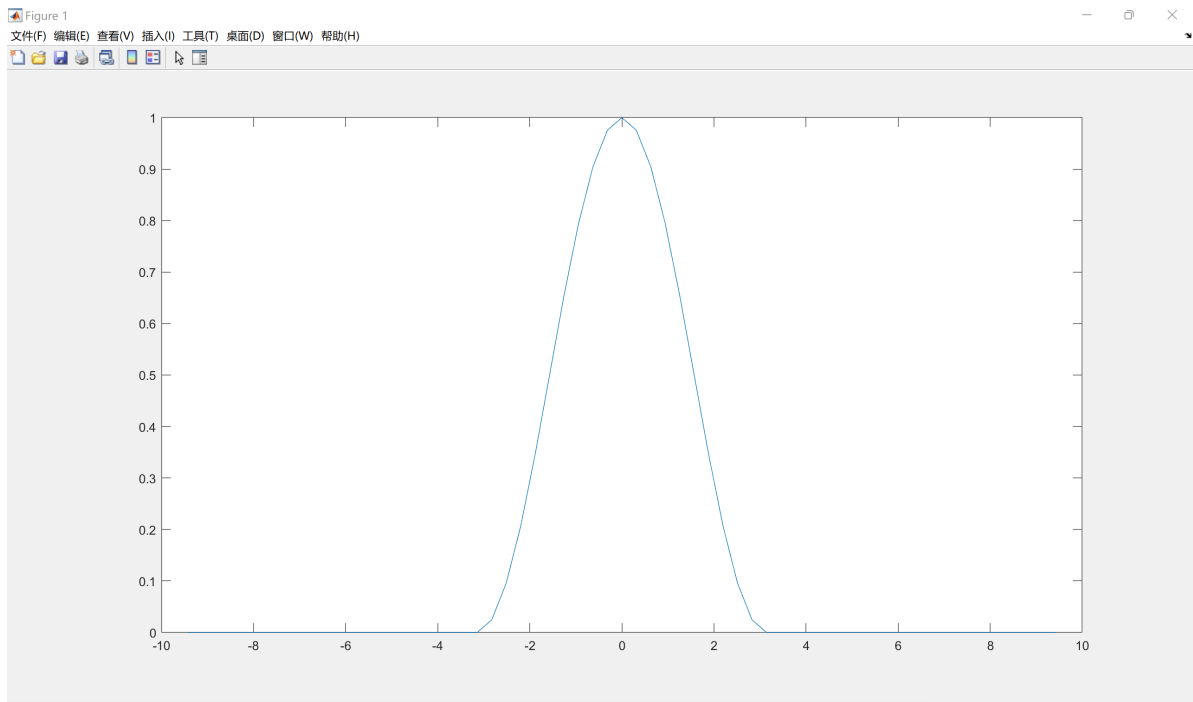
```

M=500;           %计数器
k=0:M;
w=k*w1/M;        %频域采样数为 M, w 为频率正半轴的采样点
Fw=Ft*exp(-1j*t'*w)*R; %求傅氏变换 F(jw)
w=[-flip1r(w),w(2:501)];
Fw=[flip1r(Fw),Fw(2:501)];
subplot(1,3,2);
plot(t,Ft);
grid;
xlabel('t') ;
ylabel('f(t)');
subplot(1,3,3);
plot(w,Fw);
grid;
xlabel ('w');
ylabel ('F(w)');
title('f(t)的频谱图');

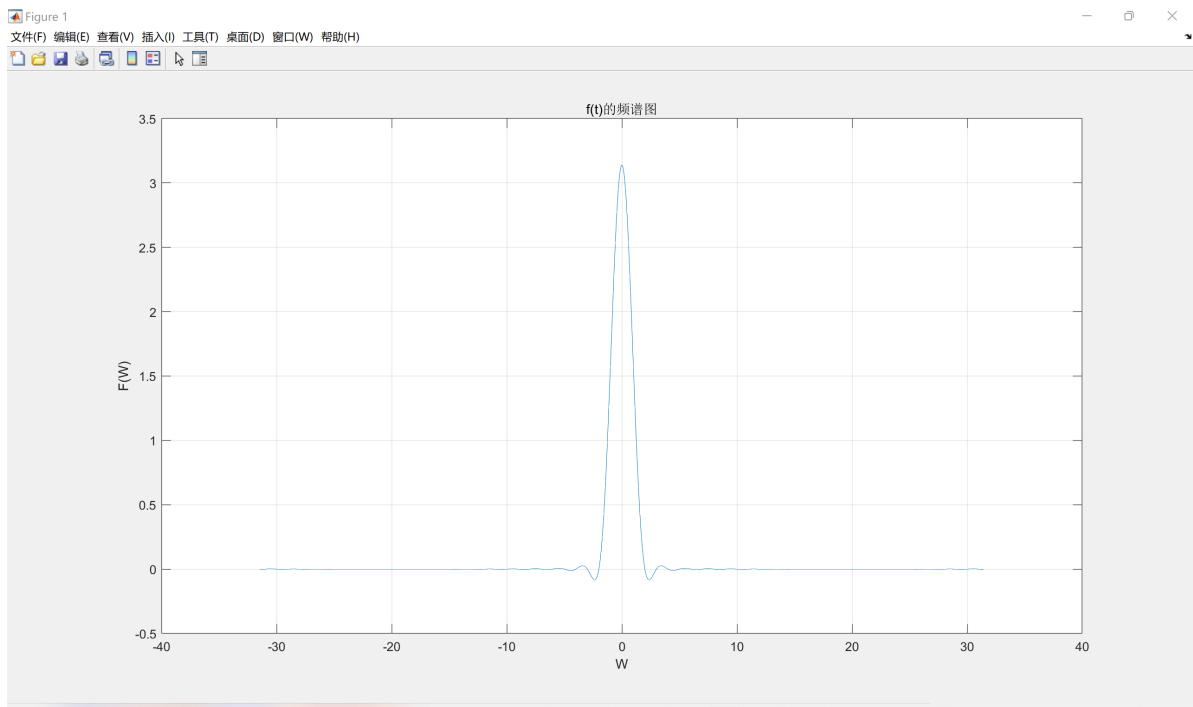
```

结果：

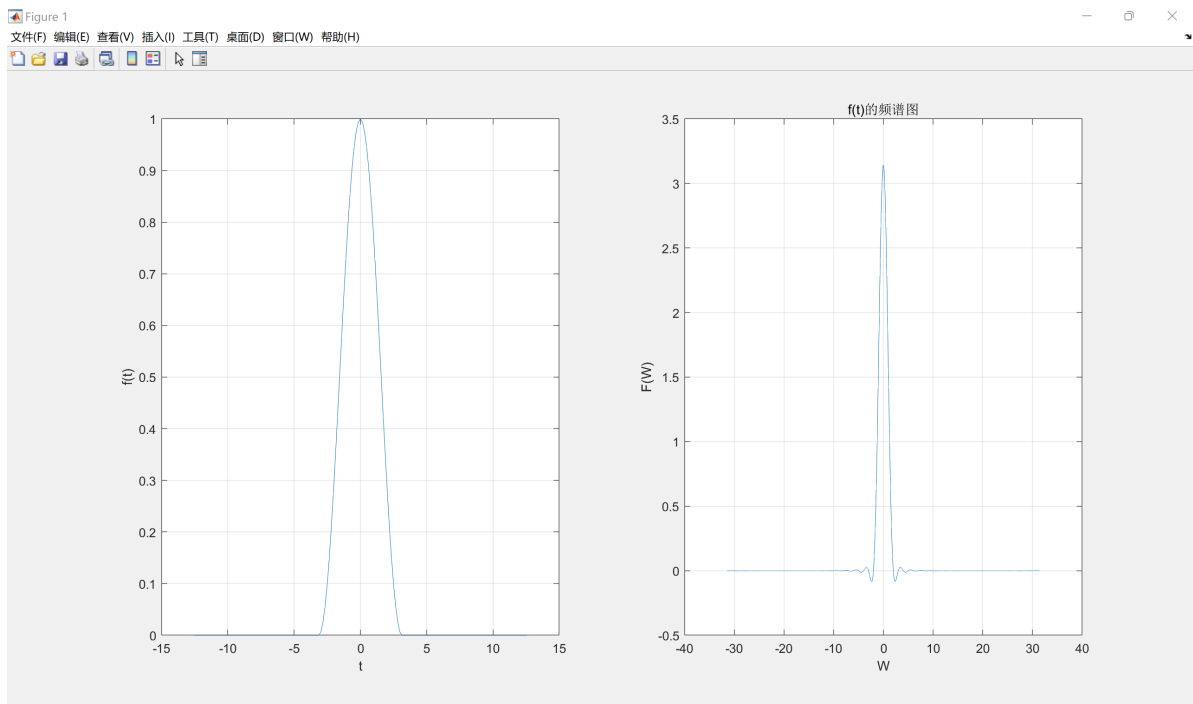
波形图：



频谱图：



对照图像：



2.采样：

只需要保留原图像上对应的采样点处的值即可完成采样

代码及注释如下：

```
t=(-3*pi:pi/20:3*pi);    %定义信号的时间范围
x=0.*(t<-pi)+(1+cos(t))/2.*(t>=-pi & t<=pi)+0.*(t>pi);    %生成信号

N=10;                    %定义采样点数
dt=pi/2;                %采样间隔，采样间隔其实就可以理解为是采样信号的周期
T=(-(N/2):N/2)*dt;      %定义采样的每个时间点
x1=0.*(T<-pi)+(1+cos(T))/2.*(T>=-pi & T<=pi)+0.*(T>pi);    %对信号进行采样

subplot(311);
plot(t,x);
```

```

ylim([-1 1])
title('原始信号')

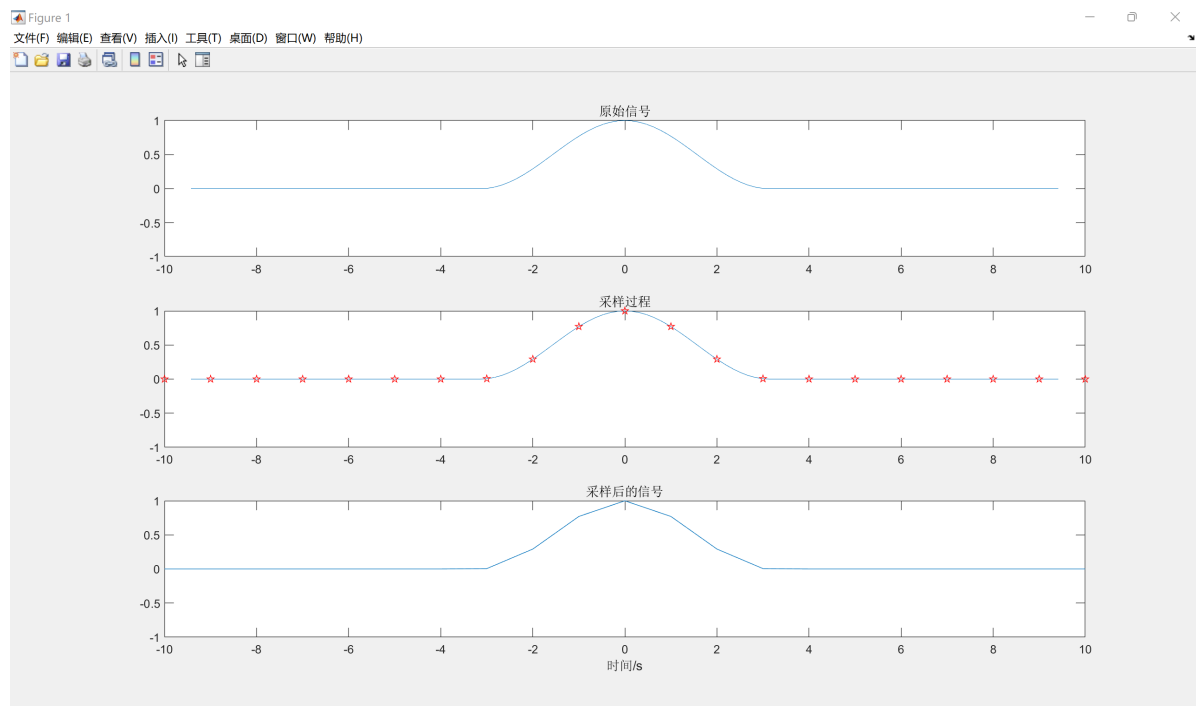
subplot(312)
plot(t,x,T,x1,'rp');
ylim([-1 1]);
title('采样过程')

subplot(313)
plot(T,x1);
ylim([-1 1])
xlabel('时间/s')
title('采样后的信号')

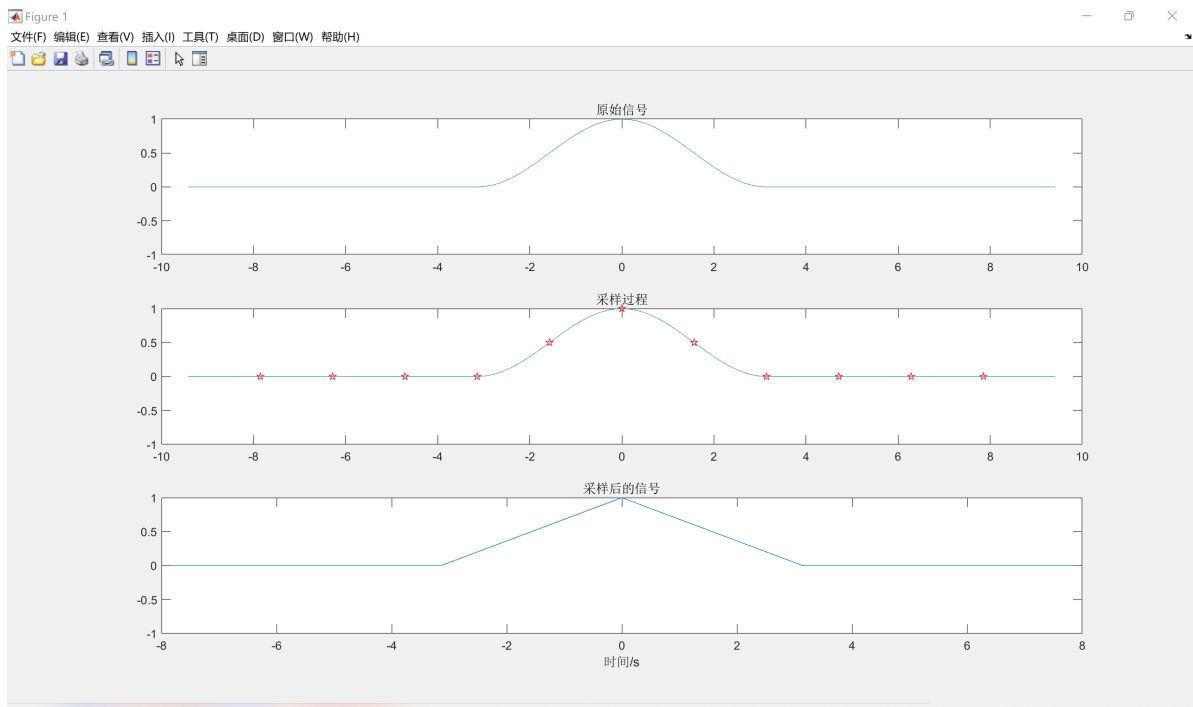
```

## 1. $T=1$

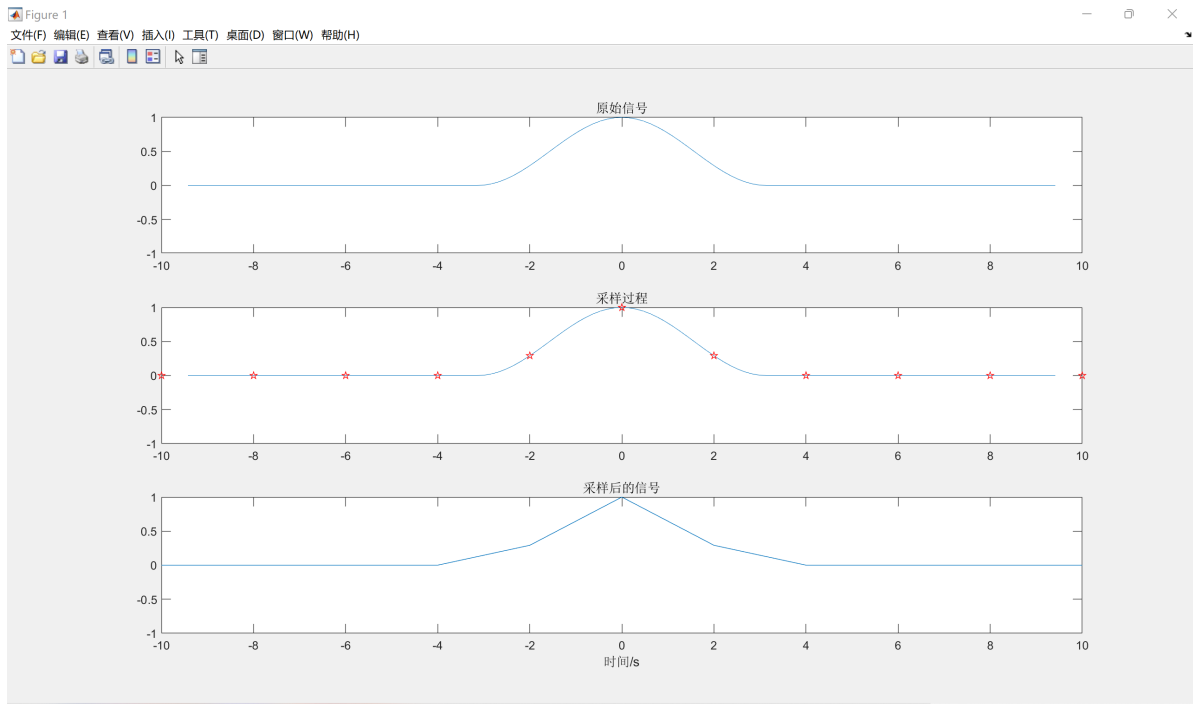
采样间隔为1，在整数点处取函数值，并连接就得到了采样图像，下面两个相同，不多做解释。



## 2. $T=\pi/2$



### 3. $T=2$



### 3. 低通滤波:

因为是理想低通滤波，所以只需要将图像在所给区间进行截断即可

代码:

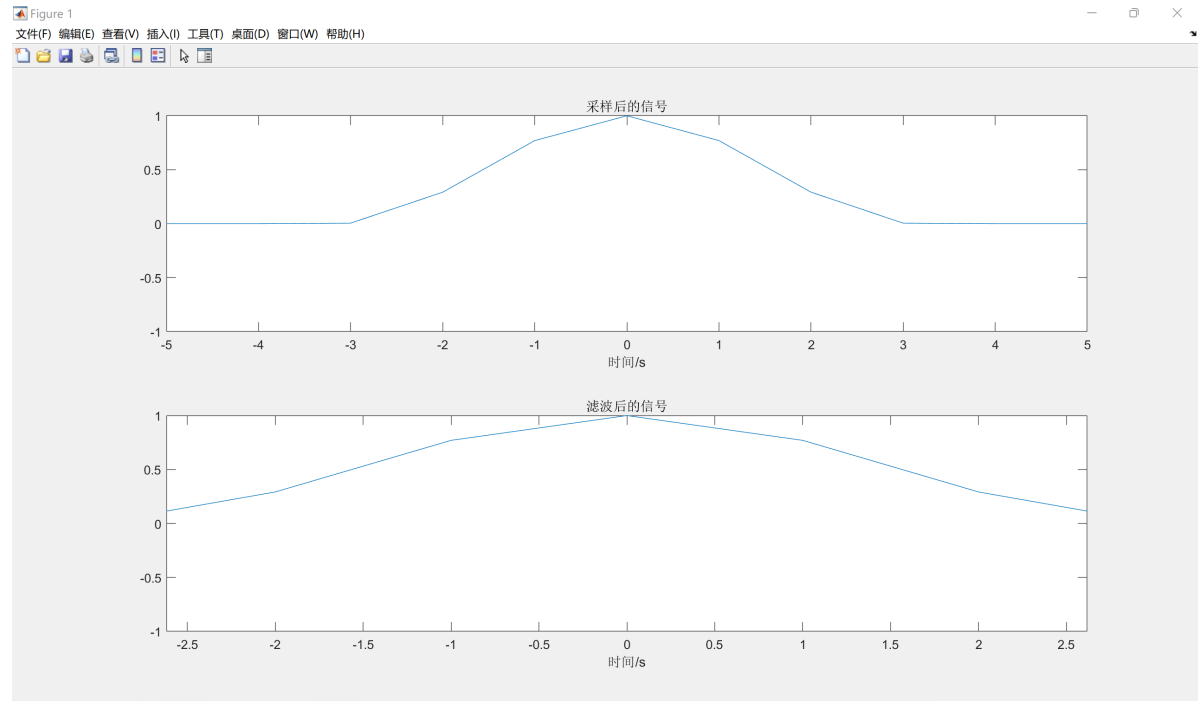
```
w=2.4;
tc=(2*pi)/w;    %截断周期
subplot(211)
plot(T,x1);
ylim([-1 1])
xlabel('时间/s')
title('采样后的信号')

subplot(212)
plot(T,x1);
```

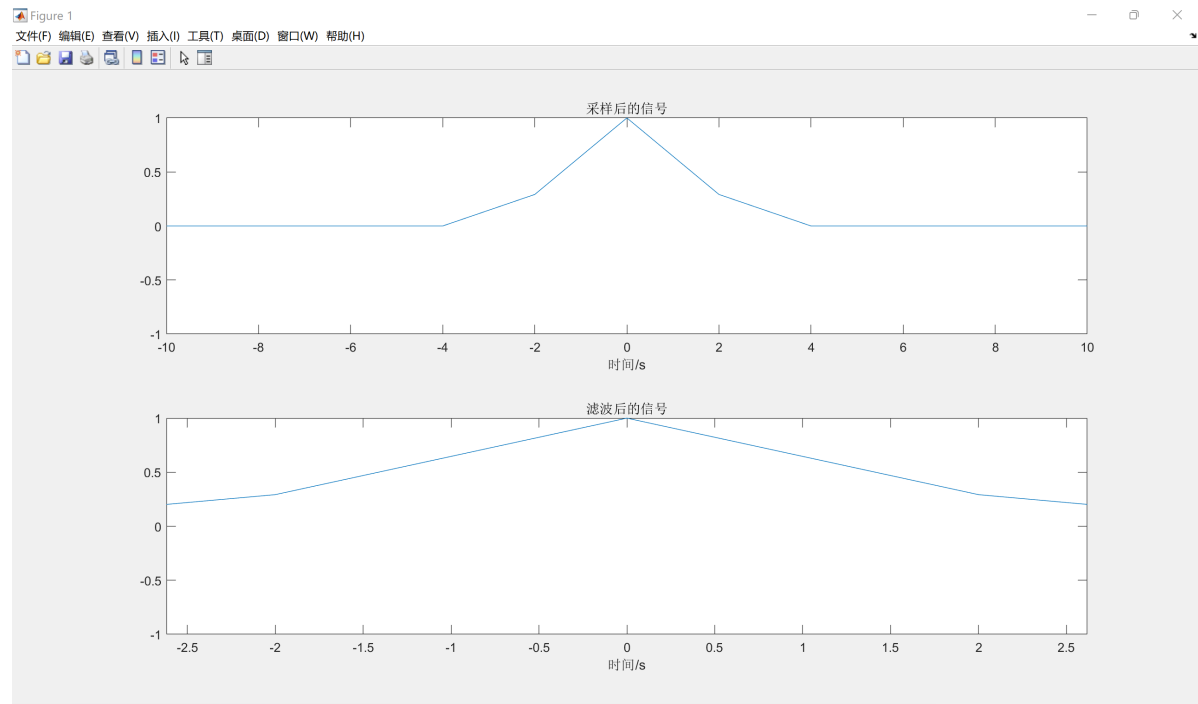
```
axis([-tc tc -0.1 10]);
ylim([-1 1])
xlabel('时间/s')
title('滤波后的信号')
```

结果：

**T=1**



**T=2**



#### 4.误差估计

使用sinc()函数进行函数复原，原理为插值函数，之后做减法可以得到误差

代码如下：

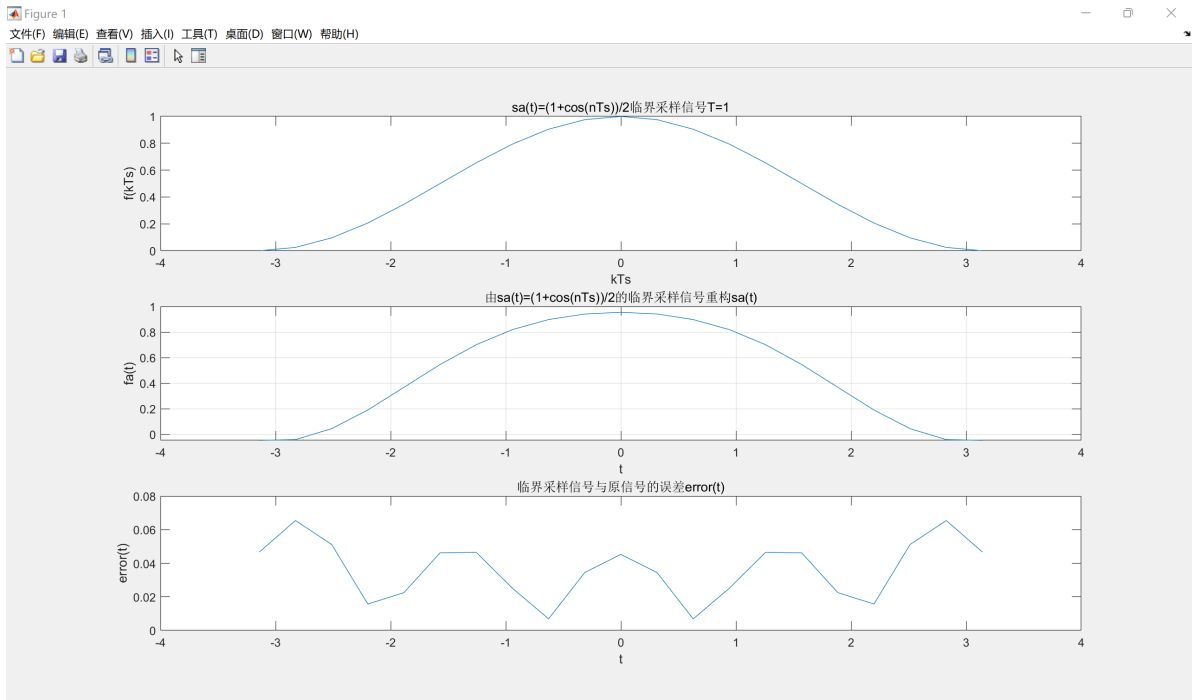
```

wc=2.4; %滤波器截止频率
Ts=2; %采样间隔
n=-5:5; %时域采样点数
nTs=n*Ts; %时域采样点
f=(1+cos(nTs))/2; %信号f(nTs)的表达式
t=-pi:pi/10:pi;
fa=f*Ts*wc/pi*sinc((wc/pi)*(ones(length(nTs),1)*t-nTs'*ones(1,length(t))))); %信号重构
error=abs(fa-((1+cos(t))/2)); %求重构信号与原信号的误差
t1=-pi:pi/10:pi;
f1=(1+cos(t1))/2;
subplot(3,1,1);
plot(t1,f1);
xlabel('kTs'); ylabel('f(kTs)');
title('sa(t)=(1+cos(nTs))/2临界采样信号T=2');
subplot(3,1,2);
plot(t,fa);
xlabel('t'); ylabel('fa(t)');
title('由sa(t)=(1+cos(nTs))/2的临界采样信号重构sa(t)');
grid on;
subplot(3,1,3);
plot(t,error);
xlabel('t'); ylabel('error(t)');
title('临界采样信号与原信号的误差error(t)');

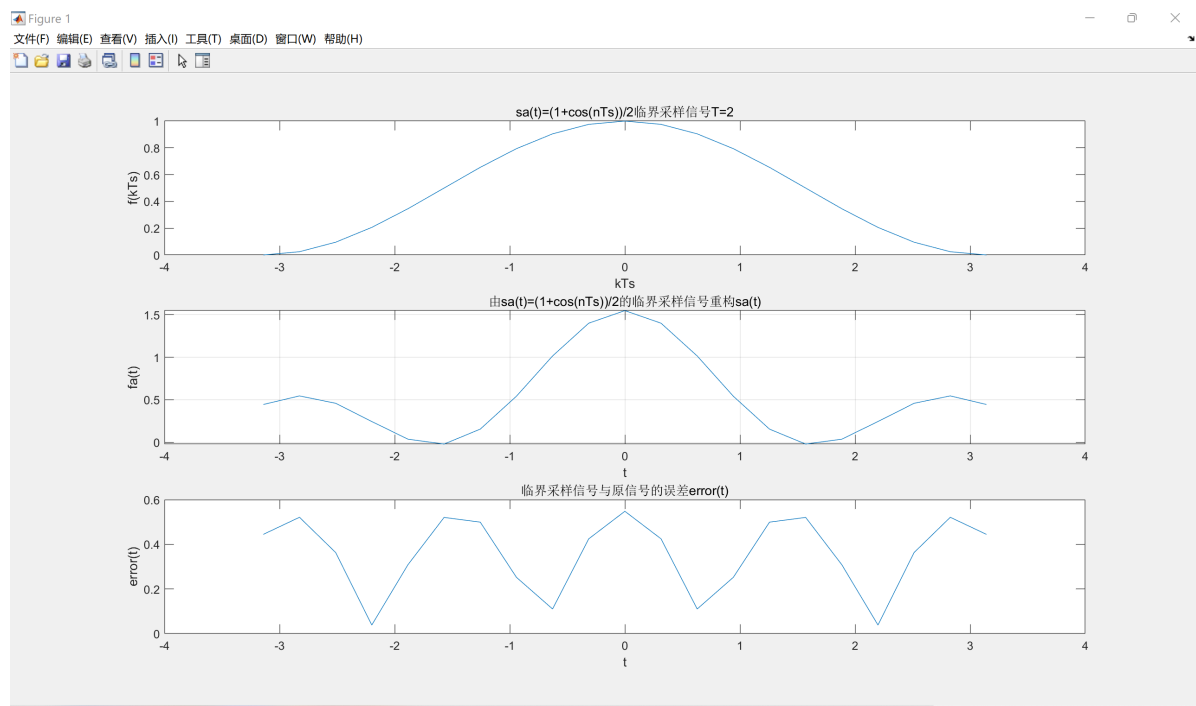
```

结果:

T=1



T=2



分析：

如果要重建效果好，需要取样点尽可能地多，例如  $T=1$  时取得点为  $T=2$  的两倍，最终得到的新函数与原函数的误差则更小。