

# 信号与系统实验报告

内容：卷积的 **Matlab** 求法

20337268 张文沁

## 1. 简述实验目的及实验原理：

(1) 目的：

- ① 掌握卷积和的计算机编程方法，利用 **MATLAB** 实现两个离散序列的卷积和
- ② 利用卷积和求离散系统的响应，观察、分析系统的时域特性

(2) 原理：

两个离散序列卷积和的定义为

$$f(k) = f_1(k) * f_2(k) = \sum_{i=-\infty}^{\infty} f_1(i) \cdot f_2(k-i) \quad (1)$$

定义式可以看作是：将序列  $f_2(i)$  的时间轴反并将其移位  $k$  个样本，然后将移位后的  $f_2(k-i)$  乘以  $f_1(i)$  并在  $i$  上将所得到的乘积序列相加。这种说法直接来自离散时间系统的线性和时不变性质。信号  $f_1(k)$  可以看成是由延时和加权脉冲的线性叠加所构成，因为一个 LTI 系统能够用它对单个脉冲的响应来表示，那么一个 LTI 系统的输出就应该相对于系统对构成  $f_1(k)$  的每一个延时和加权脉冲的叠加。在数学上，这个结果就是卷积和。

## 2. 实验内容及结果分析：

1) 附上源程序清单，要求可读性好，必要处要加注释

a. 练习 2、3 代码：调用教程所给函数

```
f1 = ones(1,4);
k1 = 0:3;
f2 = ones(1,3);
k2 = 0:2;
[f,k] = dconv(f1,f2,k1,k2)
```

```
f1 = ones(1,100);
k1 = 0:99;
f2 = ones(1,3);
k2 = 0:2;
[f,k] = dconv(f1,f2,k1,k2)
```

b. 连续函数卷积

```
1. function [f,k]=sconv(f1,f2,k1,k2,p)
2. %f 卷积和序列 f(k)对应的非零样值向量
3. %k 序列 f(k)的对应序号向量
4. %f1 序列 f1(k)向量
5. %f2 序列 f2(k)向量
6. %k1 序列 f1(k)自变量
7. %k2 序列 f2(k)自变量
8. f=conv(f1,f2);%计算序列 f1 与 f2 的卷积和 f
```

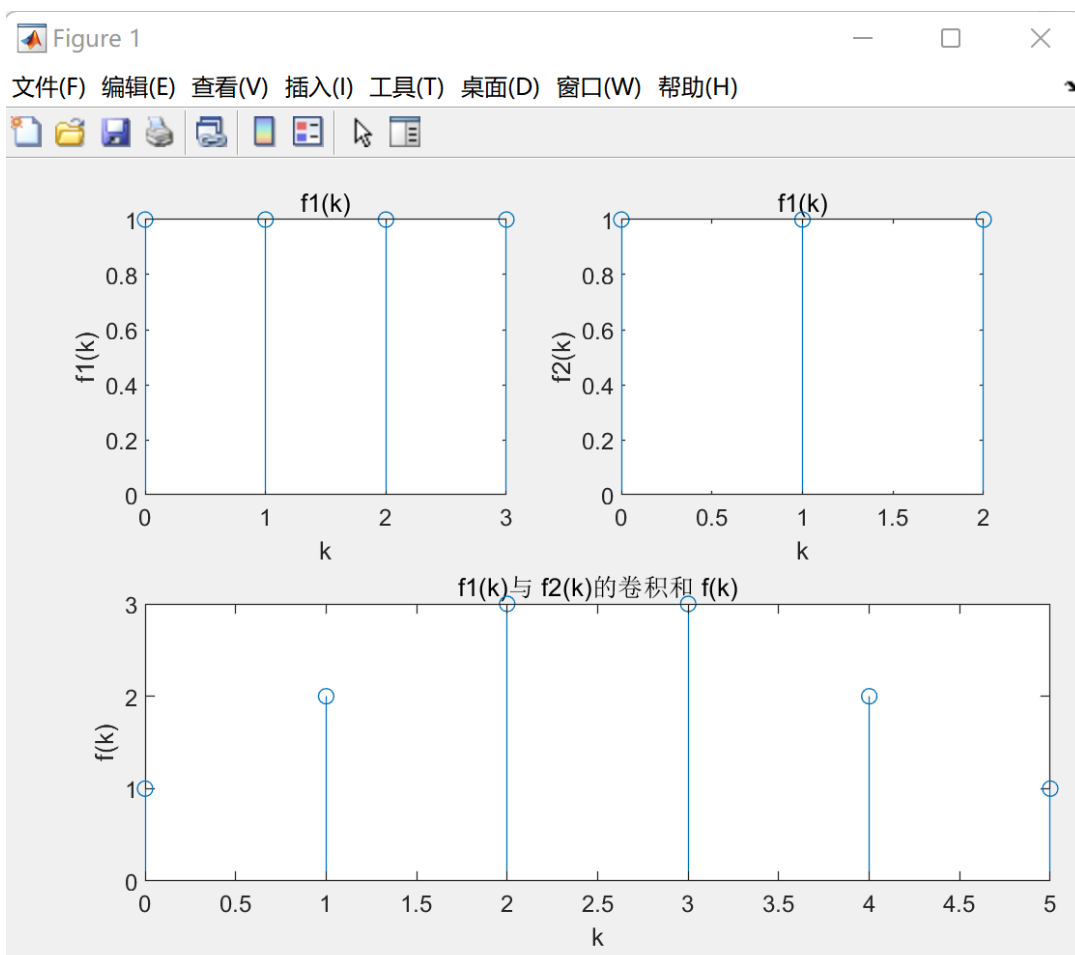
```

9.     f=f*p;
10.    k0=k1(1)+k2(1);%计算序列 f 非零样值的起点位置
11.    k3=length(f1)+length(f2)-2;%计算卷积和 f 非零样值的宽度
12.    k=k0:p:k3*p;%确定卷积和 f 非零样值的序号向量
13.    subplot(2,2,1);
14.    plot(k1,f1);%在子图 1 绘序列 f1(k)的波形
15.    title('f1(t) exp(-6*k1)');
16.    xlabel('t');
17.    ylabel('f1(t)');
18.    subplot(2,2,2);
19.    plot(k2,f2);%在子图 2 绘序列 f2(k)的波形
20.    title('f2(t) exp(-3*k2)');
21.    xlabel('t');
22.    ylabel('f2(t)');
23.    subplot(2,2,3);
24.    plot(k,f);%在子图 3 绘序列 f(k)的波形
25.    h=get(gca,'position');
26.    h(3)=2.5*h(3);
27.    set(gca,'position',h);
28.    title('f(t)=f1(t)*f2(t)');
29.    xlabel('t');
30.    ylabel('f(t)');

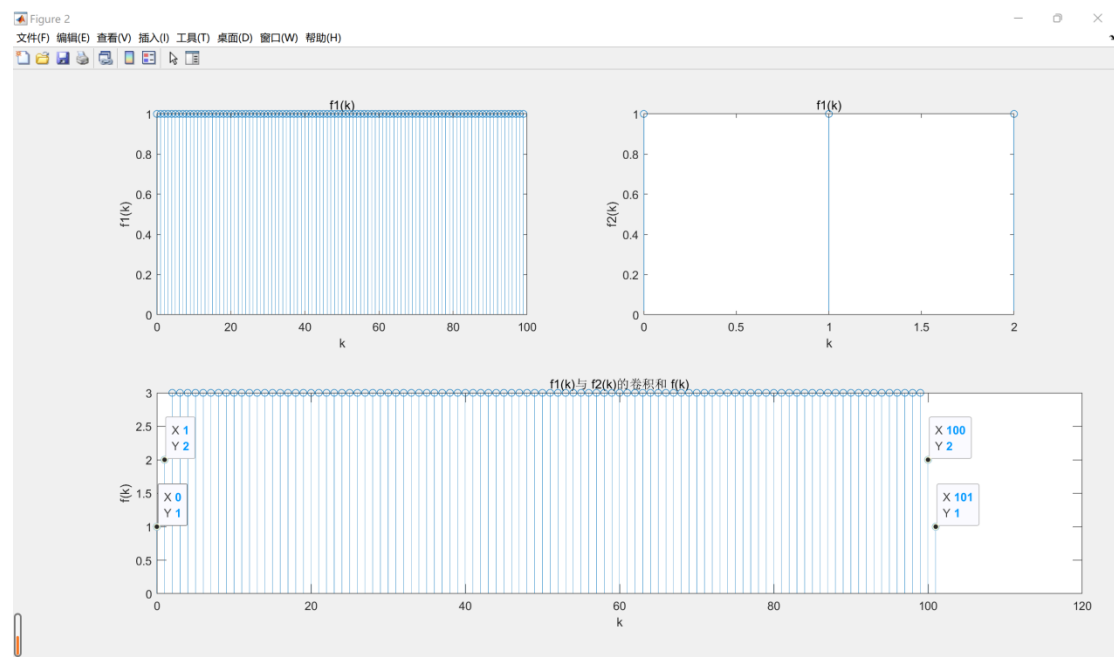
```

2) 实验结果，包括运行的数值结果或图形

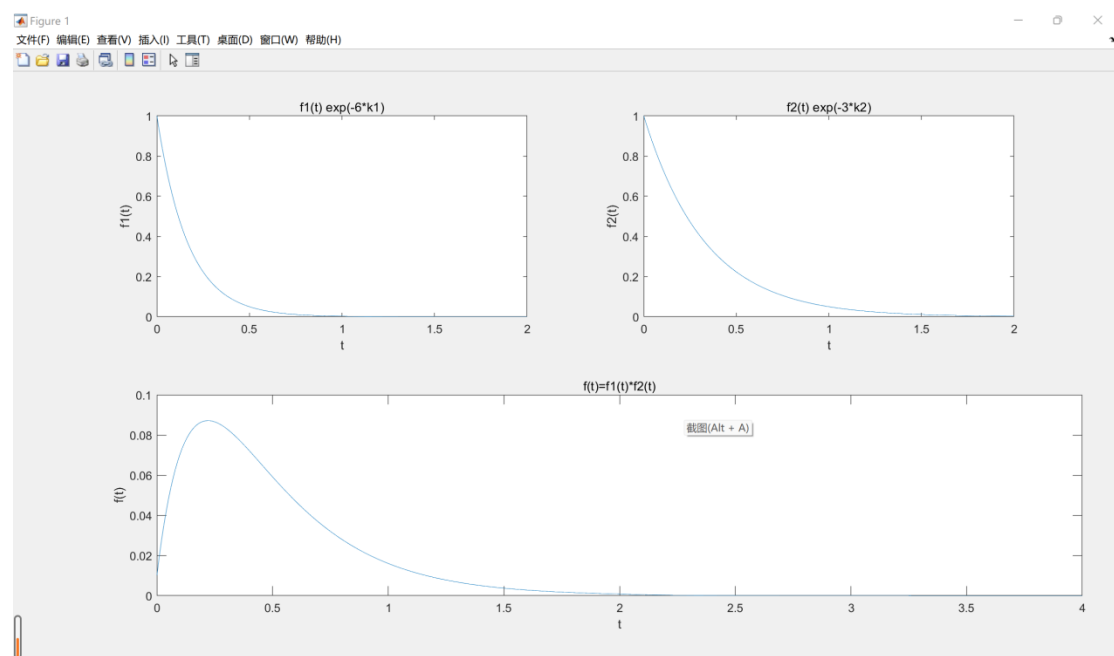
**Exercise2** 运行结果:



### Exercise3 运行结果:



$\exp(-6*k_1)$  和  $\exp(-3*k_2)$  作为例子做连续函数卷积



3) 结果分析, 正确与否, 误差原因

### 3. 简要回答思考题

(1) 观察实验内容 3 的计算结果, 所有计算样值均是真实的吗

答: 都是真实值

(2) 尝试编写连续系统卷积计算的子程序, 由连续信号的时域分解可知, 信号的卷积积分可用信号的分段求和来实现

答: 见上文代码部分

### 4. 简述本次实验的体会和建议

本次实验直观体验了卷积的生成方式和结果，对卷积的理解更加深了，也熟悉了离散型、连续型函数的卷积生成的不同之处。