第七章　　卷积与滤波

7.1 卷积

7.2 差分方程与卷积

7.1 卷 积

- 卷积运算是一种数学算法；
- 卷积是数字信号的一个基本工具，可用于提取信号或图像的特征，它提供了差分方程之外的又一种滤波实现方法；
- 卷积运算在人工智能领域，卷积神经网络（CNN）扮演着重要角色。

卷积，一种神奇的运算，以独特的方式为我们揭示了信号背后更深层次的联系，在各个领域中施展着它的魔法，改变着我们对世界的认知和体验。

在音频处理中，卷积则是灵动的调音师。如，我们想要模拟在不同空间环境下的声音效果，就可以利用卷积将原始音频信号与对应空间的脉冲响应进行运算。这样，原本单调的声音就能拥有在音乐厅的空旷回响，让我们足不出户，就能沉浸在各种奇妙的听觉场景中。

在自然语言处理领域，卷积神经网络（CNN）也借助卷积的力量，从海量的文本中提取关键信息。在分析一篇新闻报道时，CNN 可以通过卷积快速识别出报道的主题、情感倾向，为信息筛选和分析提供了强大助力。

7.1 卷积

卷积是数字信号的一个基本工具，它提供了差分方程之外的又一种滤波实现方法。

要理解卷积。首先把输入信号表示为一系列脉冲函数之和：

$$x[n] = \sum_{k=-\infty}^{\infty} x[k]\delta[n-k]$$

- 对于每个输入脉冲函数，输出为脉冲响应，因此由输入信号得到的输出为

$$y[n]=\sum_{k=-\infty}^{\infty}x[k]h[n-k]=x[n]*h[n]$$

上式的右侧为卷积的运算

卷积运算的本质是一种特殊的积分变换，通过两个函数 x 和 h 生成第三个函数。

- 卷积运算的概念来源于17世纪的瑞士数学家欧拉(Leonhard Euler)。
- 欧拉完成了一本关于积分学和以定积分求解微分方程的巨著，这部作品对卷积运算的概念和发展有着重要的影响。
- 卷积运算遵循第一性原理，从平移对称性中推导出来。
- 卷积运算最初由欧拉提出，并在后续的世纪里通过不同领域的研究者的发展和应用，逐渐成为数学、信号处理、深度学习和人工智能等领域中不可或缺的工具。

根据脉冲函数定义：

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

从脉冲响应出发

输入信号$x[n]$可以表示成：

$$x[n]=x[0]\delta[n]+x[1]\delta[n-1]+x[2]\delta[n-2]+......+x[N]\delta[n-N]$$

可以表示成下面式子

$$x[n]=\sum_{k=0}^{N}x[k]\delta[n-k]$$

根据脉冲响应与脉冲函数之间的关系

若输入为脉冲： $\delta[n-k]$ $\Rightarrow$ 对应的输出为： 脉冲响应 $h[n-k]$

若输入为： $x[n]$ $\Rightarrow$ 对应的输出为： $y[n]$

这样可以得到

$$y[n]=\sum_{k=0}^{N}x[k]h[n-k]$$

$$y[n]=\sum_{k=-\infty}^{\infty}x[k]h[n-k]\Rightarrow\sum_{k=-\infty}^{\infty}h[k]x[n-k]$$

怎么得到的呢

令

$m=n-k,$那么　$k=n-m$

带入上述式中

$$y[n]=\sum_{m=-\infty}^{\infty}x[n-m]h[m]$$

所以卷积可以表示成

$$y[n]=\sum_{k=-\infty}^{\infty}x[k]h[n-k]$$

$$y[n]=\sum_{k=-\infty}^{\infty}h[k]x[n-k]$$

所以卷积可以表示成

$$y[n]=\sum_{k=-\infty}^{\infty}x[k]h[n-k]$$

$$y[n]=\sum_{k=-\infty}^{\infty}h[k]x[n-k]$$

卷积定理

两个函数在时域上的卷积等于两者在频域上的相乘

时域上卷积 ⟶ 频域上乘积

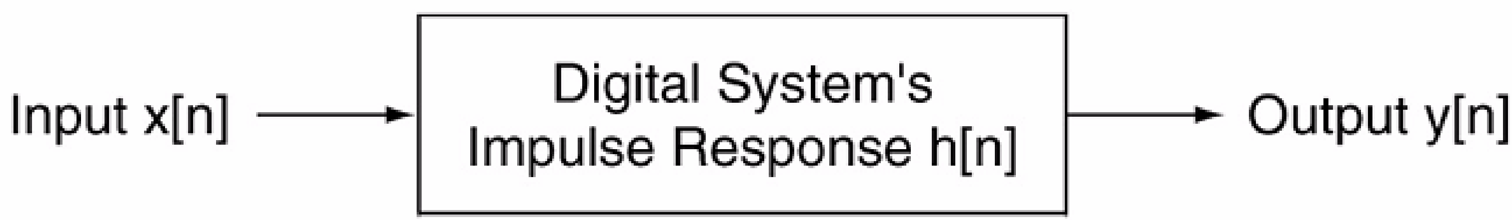
$$y[n]=\sum_{k=-\infty}^{\infty}x[k]h[n-k]=x[n]*h[n]$$

这一性质不仅在理论上具有重要意义，而且在实践中提供了将时域问题转换为频域问题进行处理的有效方法，从而简化了复杂的时域卷积运算，提高信号处理效率。

卷积定理

从数学原理来讲，傅里叶变换建立了时域和频域之间的联系。设两个函数分别为$f(t)$和$g(t)$，它们的卷积表示为$(f*g)(t)$，对这两个函数分别进行傅里叶变换得到$F(\omega)$和$G(\omega)$。根据卷积定理，时域卷积$(f*g)(t)$的傅里叶变换，就等于频域中$F(\omega)$和$G(\omega)$的乘积。

系统输出效果取决于输入和系统脉冲响应的卷积



例1

假设有一个简单的离散信号 $x[n]$，为$x[n]$=[1,2,3,4,5,4,3,2,1]

0, 1, 2, 3, 4, 5, 6, 7, 8

脉冲响应（卷积核）是： $h[n]$=[1,−1]

0, 1

卷积运算过程

$$y[n]=\sum_{k=-\infty}^{\infty}x[k]h[n-k]=\sum_{k=0}^{1}x[k]h[n-k]$$

卷积核长度 M=2，信号长度为 9。逐步计算每个输出样本 $y[n]$。

计算步骤

- n=0 时，计算 $y[0]$

$y[0]=x[0]\cdot h[0]+x[1]\cdot h[-1]$

因为 $h[-1]$ 不存在，所以： $y[0]=1\cdot 1+0=1$ $\longrightarrow$ $[1 \quad 0]\times\begin{bmatrix}1\\-1\end{bmatrix}$

- n=1 时，计算 $y[1]$

$y[1]=x[0]\cdot h[1]+x[1]\cdot h[0]$

$y[1]=1\cdot(-1)+2\cdot 1=-1+2=1$ $\longrightarrow$ $[2 \quad 1]\times\begin{bmatrix}1\\-1\end{bmatrix}$

- n=2 时，计算 y[2]

$y[2]=x[1]\cdot h[1]+x[2]\cdot h[0]$

$y[2]=2\cdot(-1)+3\cdot 1=-2+3=1$ $\longrightarrow$ $[3 \quad 2]\times\begin{bmatrix}1\\-1\end{bmatrix}$

- n=3 时，计算 y[3]

$y[3]=x[2]\cdot h[1]+x[3]\cdot h[0]$

$y[3]=3\cdot(-1)+4\cdot 1=-3+4=1$

- n=4 时，计算 y[4]

$y[4]=x[3]\cdot h[1]+x[4]\cdot h[0]$

$y[4]=4\cdot(-1)+5\cdot 1=-4+5=1$

- N=5 时，计算 y[5]

$y[5]=x[4]\cdot h[1]+x[5]\cdot h[0]$

$y[5]=5\cdot(-1)+4\cdot 1=-5+4=-1$

- n=6 时，计算 y[6]

$y[6]=x[5]\cdot h[1]+x[6]\cdot h[0]$

$y[6]=4\cdot(-1)+3\cdot 1=-4+3=-1$

- n=7 时，计算 y[7]

$y[7]=x[6]\cdot h[1]+x[7]\cdot h[0]$ → $\begin{bmatrix}2 & 3\end{bmatrix}\times\begin{bmatrix}1\\-1\end{bmatrix}$

$y[7]=3\cdot(-1)+2\cdot 1=-3+2=-1$

- n=8 时，计算 y[8]

$y[8]=x[7]\cdot h[1]+x[8]\cdot h[0]$

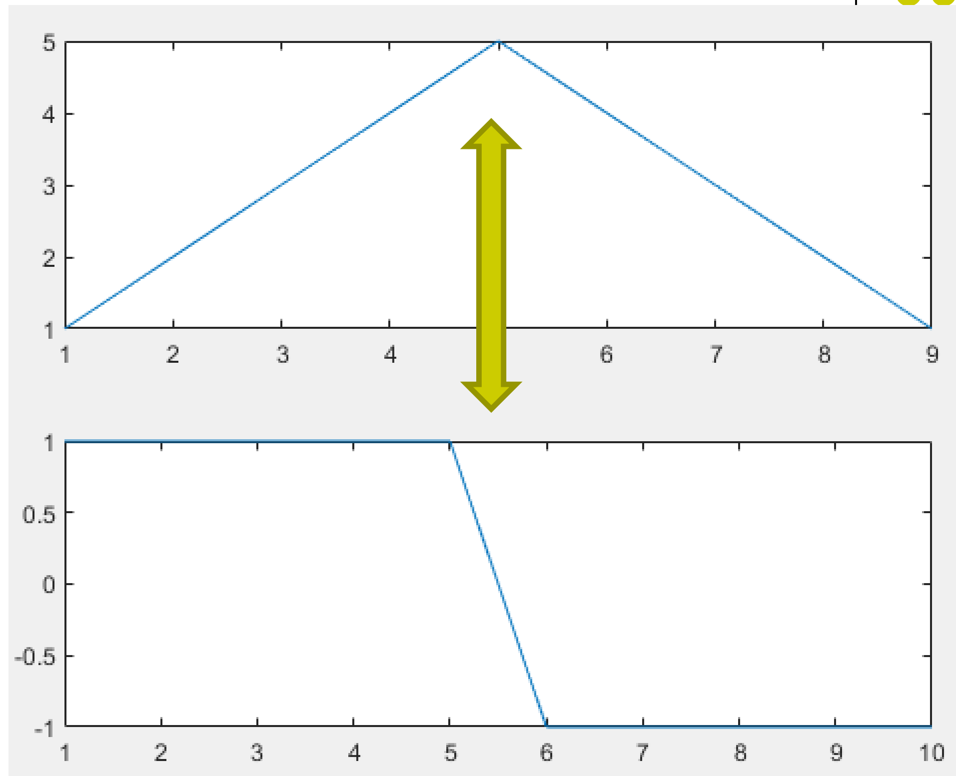
$y[8]=2\cdot(-1)+1\cdot 1=-2+1=-1$

- n=9 时，计算 y[9]

$y[9]=x[8]\cdot h[1]+x[9]\cdot h[0]$

因为 x[9] 不存在，所以:

$y[9]=1\cdot(-1)+0=-1$



这个卷积核［1，-1］能够突出信号中的高频成分（信号突变部分），因此，这个简单的卷积核可以有效地增强信号中的高频部分。

输出信号

最终的输出信号 y[n] 为:

$y[n]=[1,1,1,1,1,-1,-1,-1,-1,-1]$

例2　观察某城市一周内的日气温变化趋势

1　2　7

测得日气温数据（系列1），一维数组为：{20，22，24，25，28，27，26}

使用卷积核 kernel = [0.5,0.8,0.5] 中心点的值0.8对应输入值20，卷积核中心放在输入信号对应计算点的位置，卷积结果为该点的输出值

进行一维卷积运算，在位置1：0 x 0.5 + 20 x 0.8 + 22 x 0.5 = 27.0 → $[22 \quad 20 \quad 0]\times\begin{bmatrix}0.5\\0.8\\0.5\end{bmatrix}$

在位置2：20 x 0.5 + 22 x 0.8 + 24 x 0.5 = 39.,6

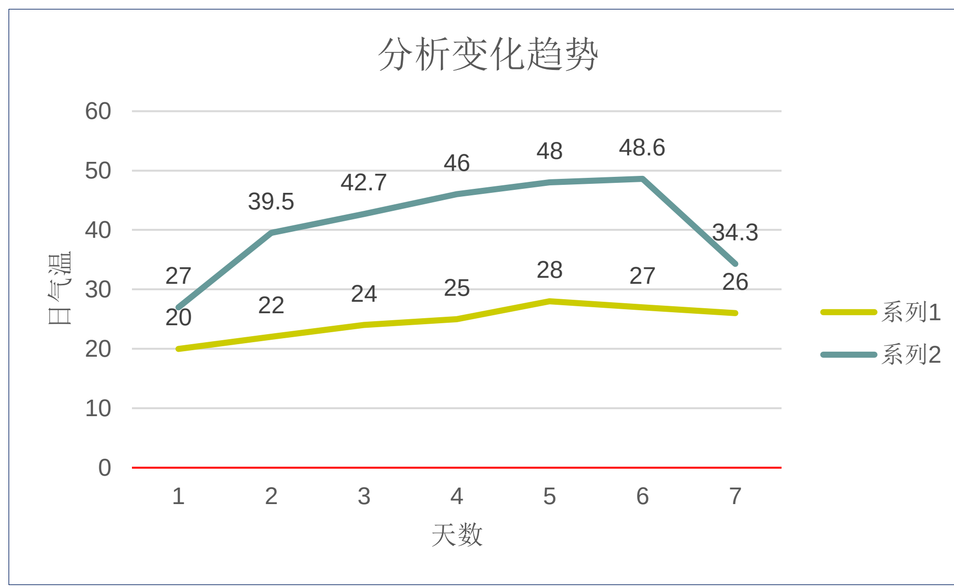
以此类推

在位置7： 27 x 0.5 + 26 x 0.8 + 0 x 0.5 = 34.3 → $[0 \quad 26 \quad 27]\times\begin{bmatrix}0.5\\0.8\\0.5\end{bmatrix}$

卷积运算后的结果（系列2）为：{ 10.0，27.0，39.6，42.7，46.0，48.0，48.6，34.3, 13.0 }

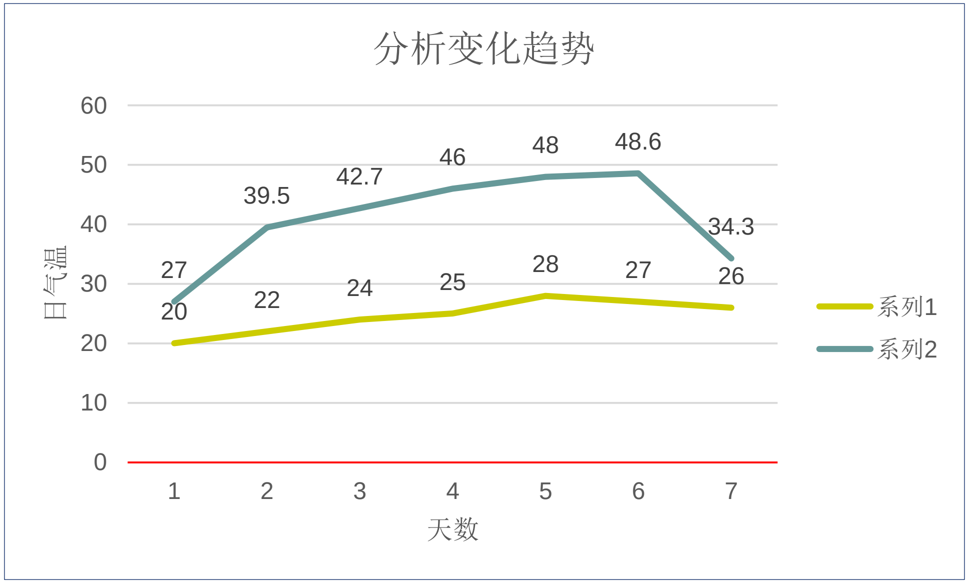
从另一角度进行观察

看到气温的拐点



举例：观察某城市一周内的日气温变化趋势

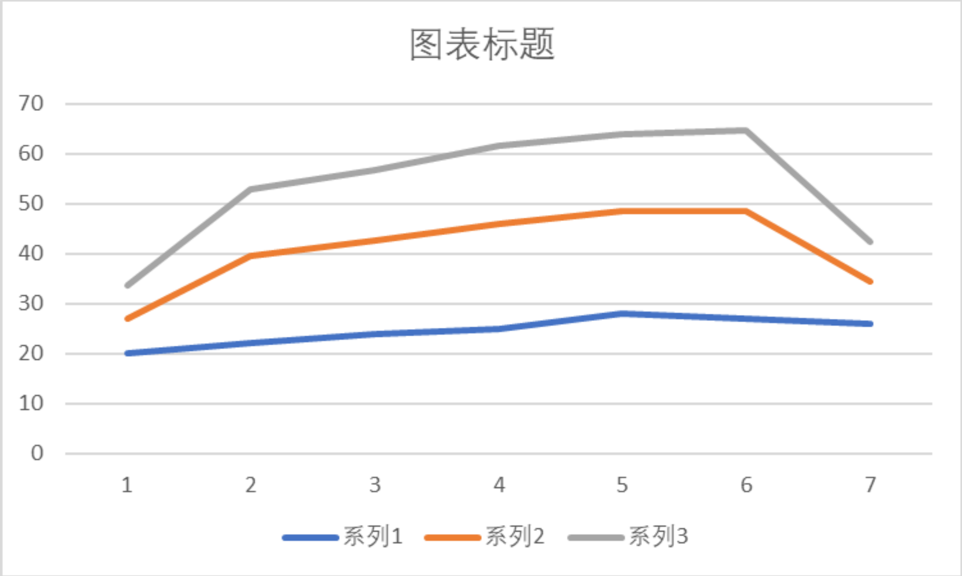
```
% Matlab 代码
% 定义两个一维信号，分别代表两个函数 f 和 g
f = [20, 22, 24, 25, 28, 27, 26];
g = [0.5, 0.8, 0.5]; % 卷积核
% 使用conv2进行卷积运算
h = conv2(f, g, 'same');
% 输出结果
disp(h)
```




```
% 输出结果
disp(h)
   10.0000  27.0000   39.6000   42.7000   46.0000   48.4000   48.6000   34.3000  13.0000
```

举例：观察某城市一周内的日气温变化趋势

```
% Matlab 代码
% 定义两个一维信号，分别代表两个函数 f 和 g
f = [20, 22, 24, 25, 28, 27, 26];
g = [0.8, 0.8, 0.8]; % 卷积核
% 使用conv2进行卷积运算
h = conv2(f, g, 'same');
% 输出结果
disp(h)
```




```
% 输出结果
disp(h)
  16.0000  33.6000   52.8000   56.8000   61.6000   64.0000   64.8000   42.4000  20.8000
```

深度神经网络（LeNet卷积网络）

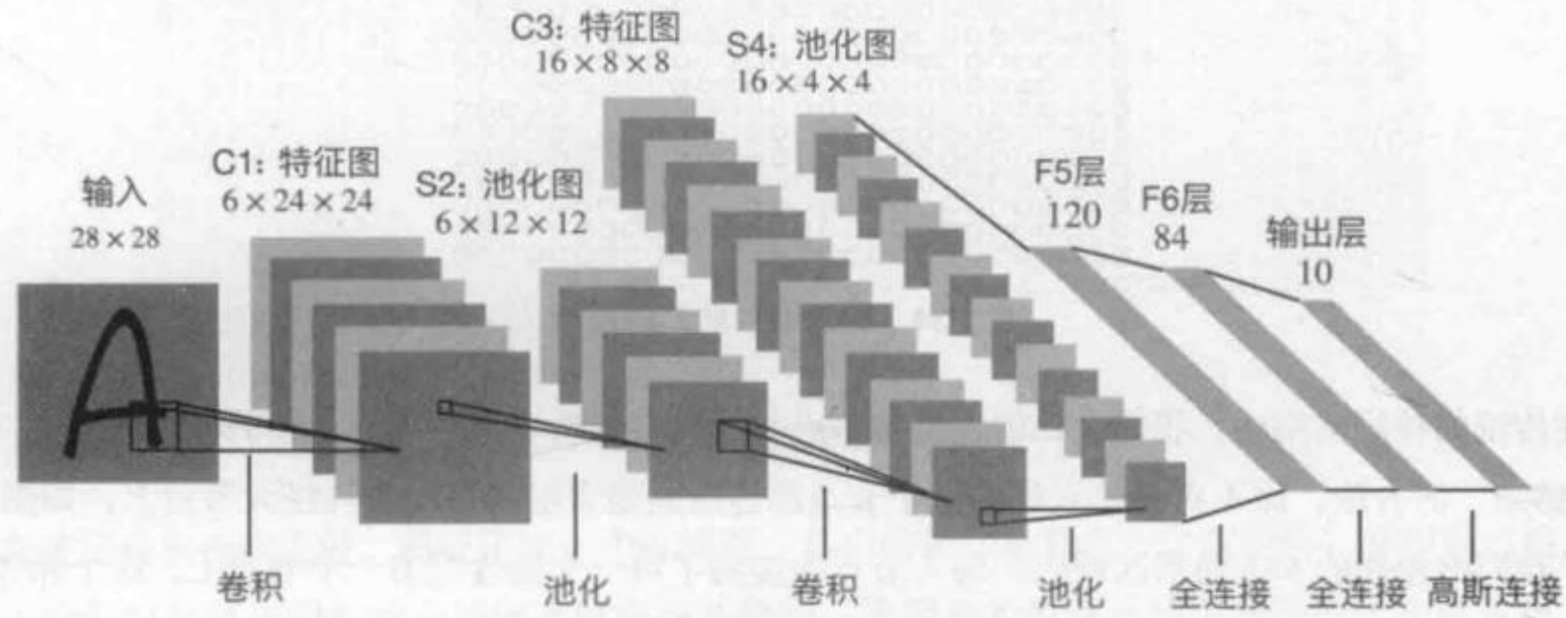
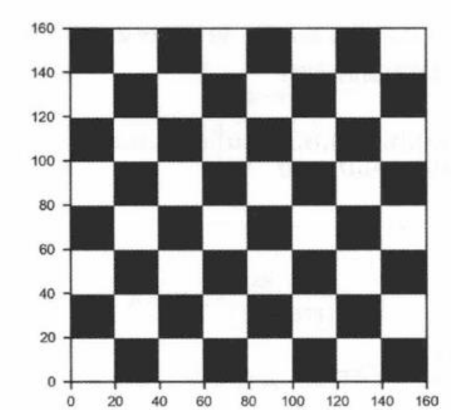


图 3-33　LeNet 结构示意图

互联网发展至今，已经存储了海量的网络图片，但是这些图片被形象地称为互联网的“暗物质”，因为现在的计算机还难以分类或识别这些非结构性的图片数据。在早期的图像识别研究中，使用人工提取的特征造成识别效果不佳。卷积神经网络的出现给图像识别领域带来了崭新的风气，如今，CNN 图像识别技术的正确率已经可以达到人类水平。卷积神经网络的兴起大大促进了深度学习研究的发展。

输入 **输出**



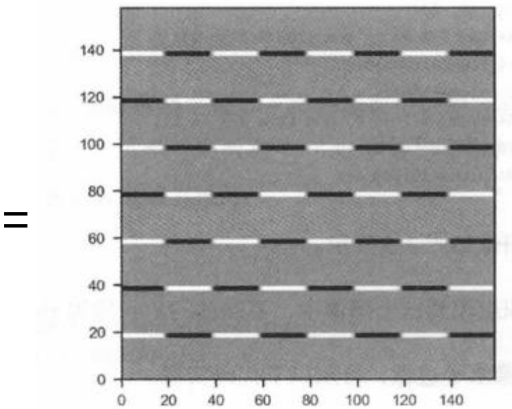
可用来检测水平边缘：

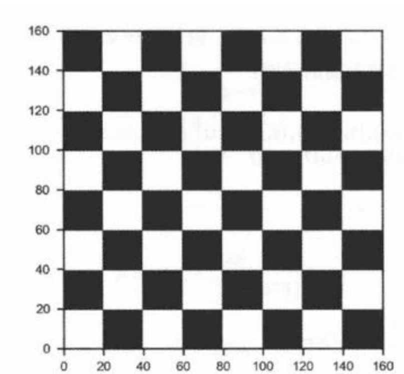
X

$$\mathcal{J}_H = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & 1 & 1 \end{bmatrix}$$

卷积核

=





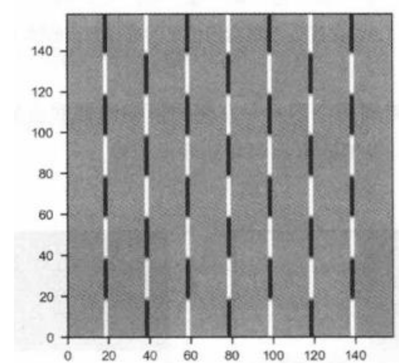
检测垂直边缘：

X

$$\mathcal{J}_V = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$$

卷积核

=



7.2 差分方程与卷积

- 滤波器的差分方程的一般表达为

$$\sum_{k=0}^{N} a_k y[n-k] = \sum_{k=0}^{M} b_k x[n-k]$$

- 卷积的表达为

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k] = x[n]*h[n]$$

卷积只与输入有关，为非递归关系。

例3 将下面差分方程表示卷积的形式

$$y[n]=0.25y[n-1]+x[n]$$

解：

根据卷积的定义，首先求系统的脉冲响应

若系统的输入是脉冲函数$\delta[n]$，则系统的输出就是脉冲响应 $h[n]$

所以

$$x[n]=\delta[n],\qquad y[n]=h[n]$$

这样脉冲响应为：

$$h[n]=0.25h[n-1]+\delta[n] \Longrightarrow \left[\begin{array}{lllll} n= & 0, & 1, & 2, & 3\ldots \\ h[n]= & 1, & 0.25, & 0.25^2, & 0.25^3\ldots \end{array}\right.$$

上述带入卷积定义

$$y[n]=\sum_{k=0} h[k]x[n-k]$$

$$=h[0]x[n]+h[1]x[n-1]+h[2]x[n-2]+\ldots\ldots$$

$$=0.25^0x[n]+0.25^1x[n-1]+0.25^2[2]x[n-2]+0.25^3x[n-3]+\ldots\ldots$$

$$=\sum_{k=0}^{\infty}0.25^k x[n-k]$$

Matlab 卷积运算

```
h = [1, 2, 3, 4];

x = [0, -1, -4];

% 使用 conv 进行卷积运算

%h = conv(x, h, 'same');% 输出结果位数与x一样

h = conv(h, x);% 输出结果不受限制

%h = conv(x, h);% 输出结果不受限制,h*x=x*h

% 输出结果

disp(h)
```

Matlab 代码

```
%系统差分方程y[n]=0.6y[n-1]+0.08y[n-2]+x[n]
b=1;  %b为x系数
a=[1,-0.6,-0.08];  %a为y系数

ixn=[1,zeros(1,30)];  %产生脉冲信号ixn

%求出系统的脉冲响应hn1
hn1=filter(b,a,ixn);
n=0:length(hn1)-1;
subplot(4,1,1);stem(n,hn1,'.')
title('脉冲响应');
xlabel('n');ylabel('h(n)')

xn=ones(1,30);%系统输入信号xn
subplot(4,1,2)
stem(xn,'.')
title('系统输入');
xlabel('n');ylabel('in(n)')

%由差分方程计算得出的系统输出信号sn1
sn1=filter(b,a,xn); #解差分方程
n=0:length(sn1)-1;
subplot(4,1,3);stem(n,sn1,'.')
title('由差分方程系统输出');
xlabel('n');ylabel('ou(n)')

%由”系统脉冲响应与系统输入信号进行卷积”得到系统输出信号sn2
subplot(4,1,4);
sn2=conv(hn1,xn);  #卷积
stem(0:29,sn2(1:30),'.');
title('脉冲响应与卷积');
xlabel('n');
ylabel('h*in(n)')
```

Python 卷积运行

```
import numpy as np
from scipy import signal
import matplotlib.pyplot as plt

# 建立画图框fig1，在一个框内显示一个曲线或直线
fig1=plt.figure(num=11)

# 建立自变量x
x=np.random.randn(1000)#随机生成输入信号
plt.plot(x) #绘图x

kernel=np.array([1,2,3]) # 定义卷积核

result=signal.convolve(x,kernel) # 进行卷积运算

# 输出结果
fig1=plt.figure(num=12) #显示卷积计算结果图像
print(result)
# 显示结果图像
plt.plot(result)
plt.show()
```

练习

使用Python对前面的（举例：观察某城市一周内的日气温变化趋势）进行分析。

7.2 滑动平均滤波器

- 滑动滤波器是一种非递归滤波器，可以平滑输入信号，保留低频部分，消除高频部分。
-

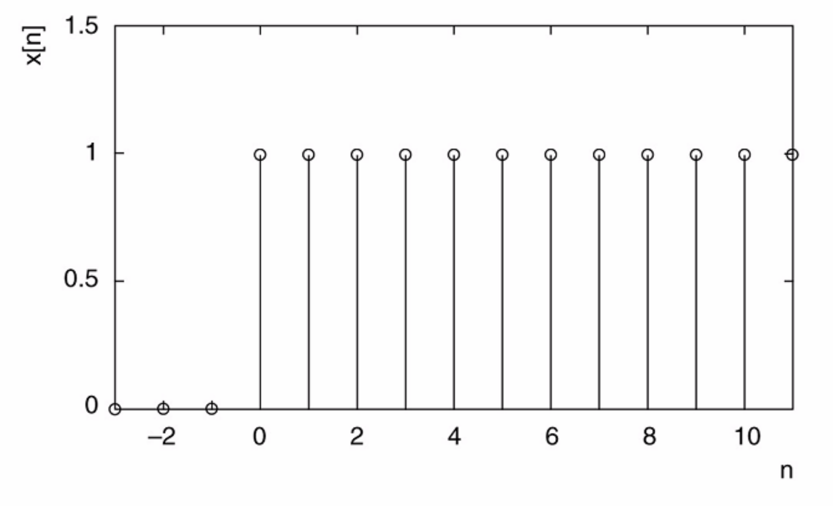
例如：五项滑动平均滤波器的差分方程为

$$y[n]=\frac{1}{5}(x[n]+x[n-1]+x[n-2]+x[n-3]+x[n-4])$$

试对输入信号进行处理

练习

输入信号为：



试用五项滑动平均滤波器对输入信号进行处理

$$y[n]=\frac{1}{5}(x[n]+x[n-1]+x[n-2]+x[n-3]+x[n-4])$$

上述可用 Matlab 或 python 实现

卷积：信号世界的幕后工匠

在信号处理的奇妙天地中，卷积是一位低调却举足轻重的幕后工匠。从数学视角看，它是一种独特运算，将两个函数依特定规则交织融合。

当我们把信号想象成流淌的音符，卷积便是指挥家的指挥棒，巧妙引导音符的节奏与韵律。在音频处理时，卷积化身为调音大师，通过与不同的脉冲响应函数卷积，为声音增添音乐厅的回响、山谷的空灵回音等，丰富着声音的空间感。

卷积就像一把万能钥匙，在各个信号领域解锁无数可能，默默塑造着我们所感知的数字世界。