

**MAC0317/MAC5920**

**Introdução ao Processamento de Sinais Digitais**

**Capítulo 4: Convolução e filtragem**

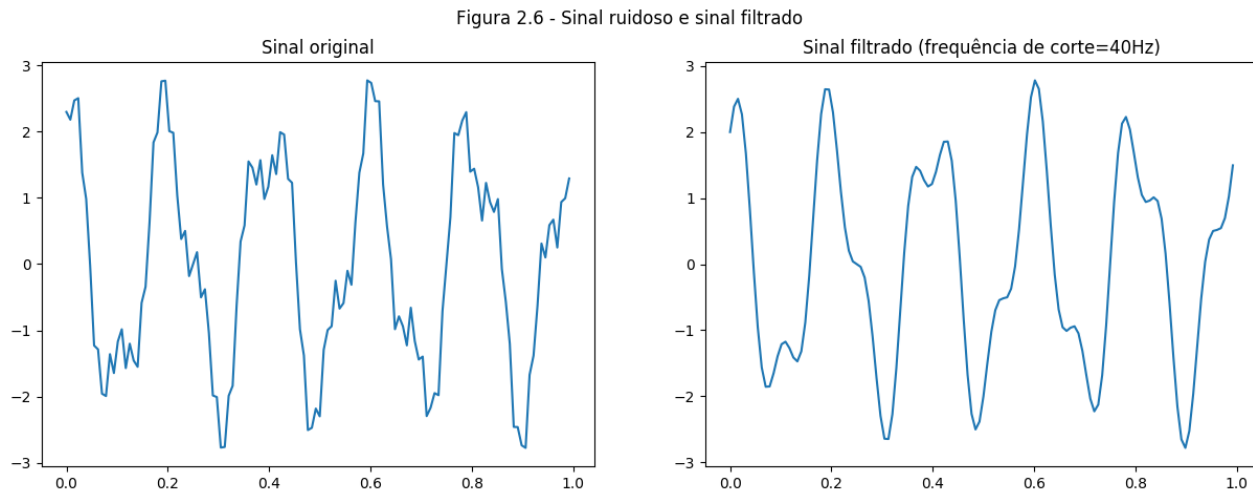
**Seção 4.2: Convolução unidimensional**

## Seção 4.2.1: Exemplo de filtragem passa-baixas e remoção de ruído

Na seção 2.3 vimos um exemplo de filtragem do sinal

$x(t) = 2 \cos(2\pi 5t) + 0.8 \sin(2\pi 12t) + 0.3 \cos(2\pi 47t)$ , onde as componentes de alta frequência foram removidas diretamente do espectro fazendo  $X_k \leftarrow 0, \forall k \geq 40$ :

```
In [8]: f, ax = plt.subplots(1,2,figsize=(15,5))
ax[0].set_title("Sinal original");ax[0].plot(t,x)
ax[1].set_title("Sinal filtrado (frequência de corte=40Hz)");ax[1].plot(t,w)
f.suptitle("Figura 2.6 - Sinal ruidoso e sinal filtrado");plt.show()
```



## Métodos de filtragem no domínio do tempo

Queremos considerar como esse tipo de operação poderia ser realizada no domínio do tempo, através de manipulações das amostras  $x_n$ . Considere o seguinte exemplo, chamado de *filtro da média*: dado um sinal  $x$ , construímos um novo sinal  $w$  pela equação

$$w_n = \frac{1}{2}x_n + \frac{1}{2}x_{n-1}, \quad \forall n$$

No caso de um vetor  $x \in \mathbb{R}^N$ , essa equação apresenta uma dificuldade quando  $n = 0$ :

$$w_0 = \frac{1}{2}x_0 + \frac{1}{2}x_{-1}.$$

Faremos como Fourier, e consideraremos  $x$  um período de um sinal que se repete a cada  $N$  amostras, de tal forma que

$$x_{-1} = x_{N-1};$$

**Observação:** coincidência ou não, Python usa a mesma convenção!

## Resposta em frequência do filtro

Podemos usar a equação do filtro  $w_n = \frac{1}{2}x_n + \frac{1}{2}x_{n-1}$  para estimar como o filtro se comporta em relação a entradas simples. Por exemplo, se

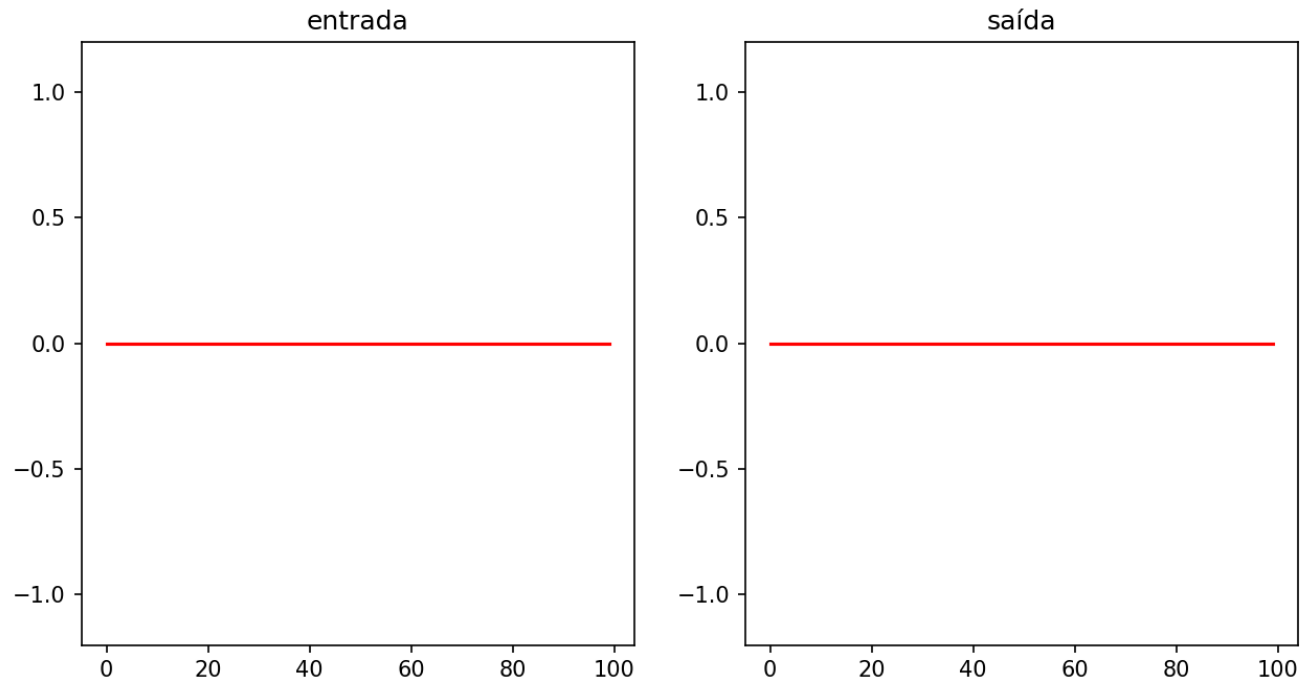
$$x_n = \sin(2\pi qn/N),$$

teremos

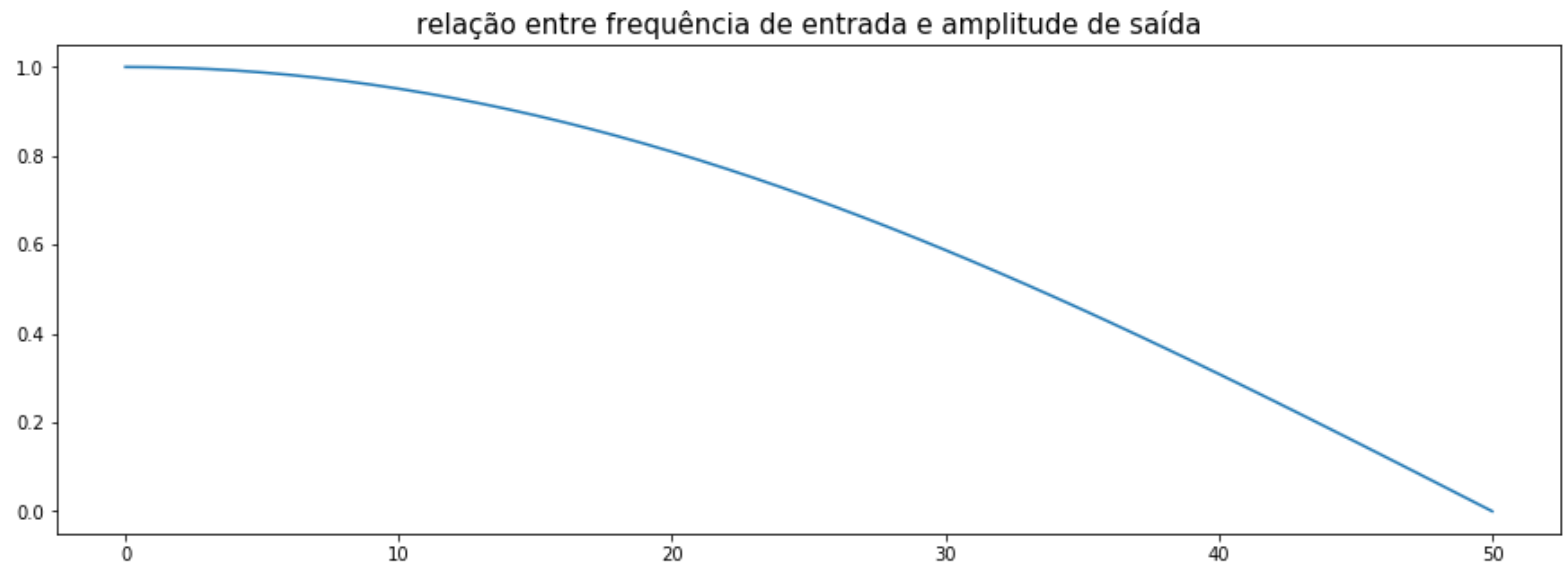
$$\begin{aligned}w_n &= \frac{1}{2}\sin(2\pi qn/N) + \frac{1}{2}\sin(2\pi q(n-1)/N) \\&= \frac{1}{2}\sin(2\pi qn/N) + \frac{1}{2}\overbrace{\sin((2\pi qn/N) - (2\pi q/N))}^{\sin(\alpha-\beta)=\sin\alpha\cos\beta-\cos\alpha\sin\beta} \\&= \frac{1}{2}\sin(2\pi qn/N) + \frac{1}{2}\sin(2\pi qn/N)\cos(2\pi q/N) - \frac{1}{2}\cos(2\pi qn/N)\sin(2\pi q/N) \\&= A\sin(2\pi qn/N) - B\cos(2\pi qn/N)\end{aligned}$$

onde  $A = \frac{1}{2} + \frac{1}{2}\cos(2\pi q/N)$  e  $B = \frac{1}{2}\sin(2\pi q/N)$ .

```
In [6]: # animação com entradas senoidais e saídas do filtro da média (ver no jupyter)
%matplotlib notebook
N=100;k=np.arange(0,N);W=[1]+[0]*(N//2)
fig , ax = plt.subplots(1,2, figsize=(10,5))
ax[0].set_title("entrada");ax[1].set_title("saída")
ax[0].set_ylim([-1.2,1.2]);ax[1].set_ylim([-1.2,1.2])
w = 0*k;line1, = ax[0].plot(k, w, 'r-');line2, = ax[1].plot(k, w, 'r-')
for q in range(1,N//2+1):
    W[q] = entrada_saida_filtro(q,k,line1,line2,fig)
W[N//2] = 0 # corrige a última magnitude
```



```
In [7]: %matplotlib inline
plt.figure(figsize=(15,5));plt.plot(W);
plt.title("relação entre frequência de entrada e amplitude de saída", fontsize=15)
plt.show()
```



## Resposta complexa em frequência do filtro

Analogamente, podemos usar a equação do filtro  $w_n = \frac{1}{2}x_n + \frac{1}{2}x_{n-1}$  para estimar como o filtro se comporta em relação às formas básicas de onda de Fourier. Se

$$x_n = e^{i2\pi qn/N}, n = 0, 1, \dots, N - 1$$

onde  $q = 0, 1, \dots, N - 1$  é uma frequência, então

$$\begin{aligned}w_n &= \frac{1}{2}e^{i2\pi qn/N} + \frac{1}{2}e^{i2\pi q(n-1)/N} \\&= \frac{1}{2}e^{i2\pi q(n-\frac{1}{2})/N} \left[ e^{i\pi q \frac{1}{N}} + e^{-i\pi q \frac{1}{N}} \right] \\&= \cos\left(\pi q \frac{1}{N}\right) e^{i2\pi q(-\frac{1}{2})/N} e^{i2\pi qn/N} \\&= H_q x_n\end{aligned}$$

onde  $H_q = \cos\left(\pi q \frac{1}{N}\right) e^{-i\pi q/N}$ .

**Observe** que o fator  $H_q$ , dependente da frequência  $q$ , equivale a uma atenuação de amplitude

$$|H_q| = \cos\left(\pi q \frac{1}{N}\right)$$

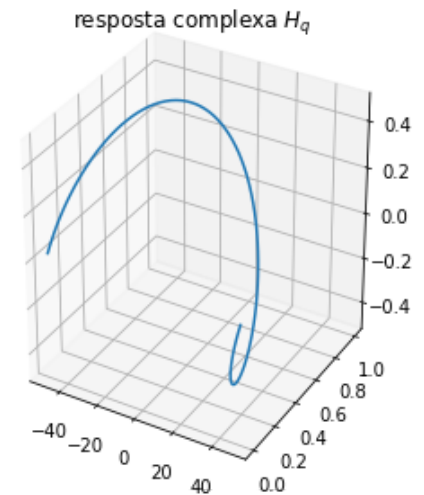
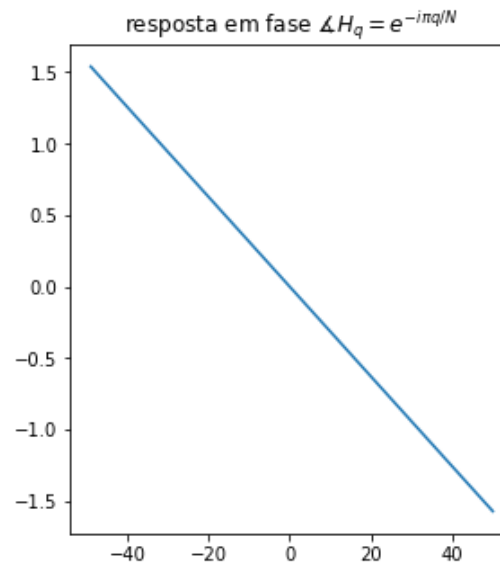
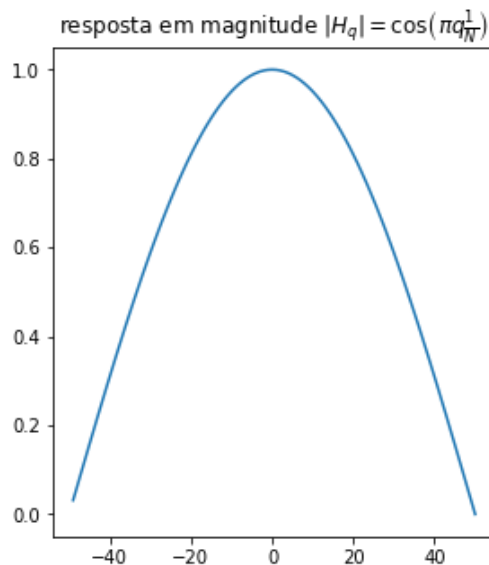
com característica de passa-baixas e uma mudança de fase

$$\angle H_q = -\pi q/N$$

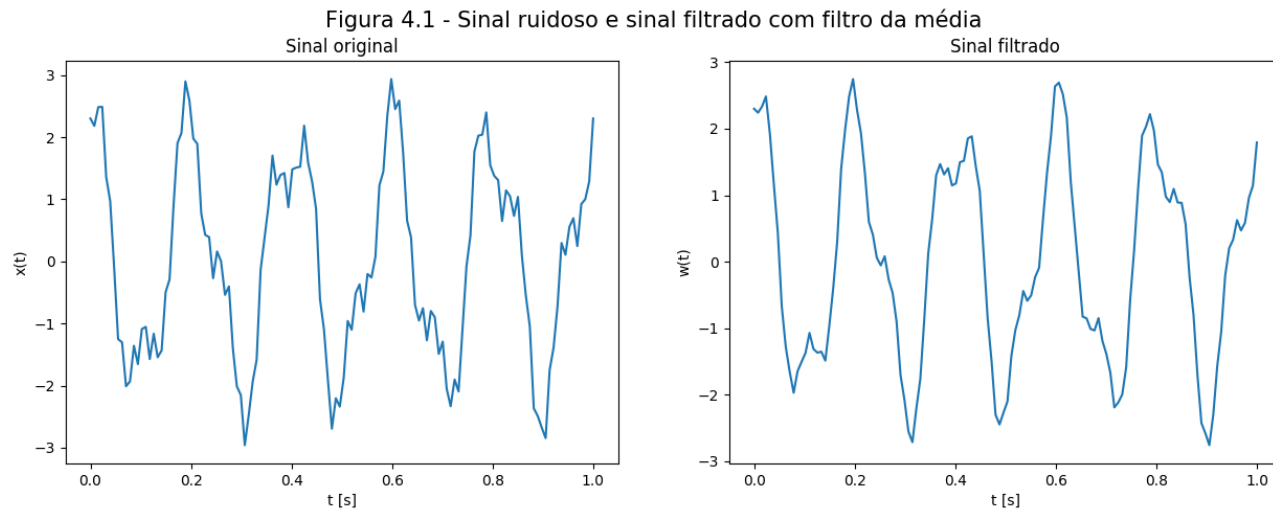
que corresponde a uma rotação.



```
In [8]: q = np.arange(-N/2+1,+N/2+1);Hq = np.cos(m.pi*q/N)*np.exp(-1j*m.pi*q/N)
fig = plt.figure(figsize=(15,5))
ax = fig.add_subplot(1, 3, 1);ax.plot(q,np.abs(Hq));ax.set_title(r"resposta em
magnitude  $|H_q| = \cos\left(\pi q \frac{1}{N}\right)$ ")
ax = fig.add_subplot(1, 3, 2);ax.plot(q,np.angle(Hq));ax.set_title(r"resposta e
m fase  $\angle H_q = e^{-i\pi q/N}$ ")
ax = fig.add_subplot(1, 3, 3, projection='3d');ax.plot3D(q,Hq.real,Hq.imag);ax.
set_title(r"resposta complexa  $H_q$ ")
plt.show()
```

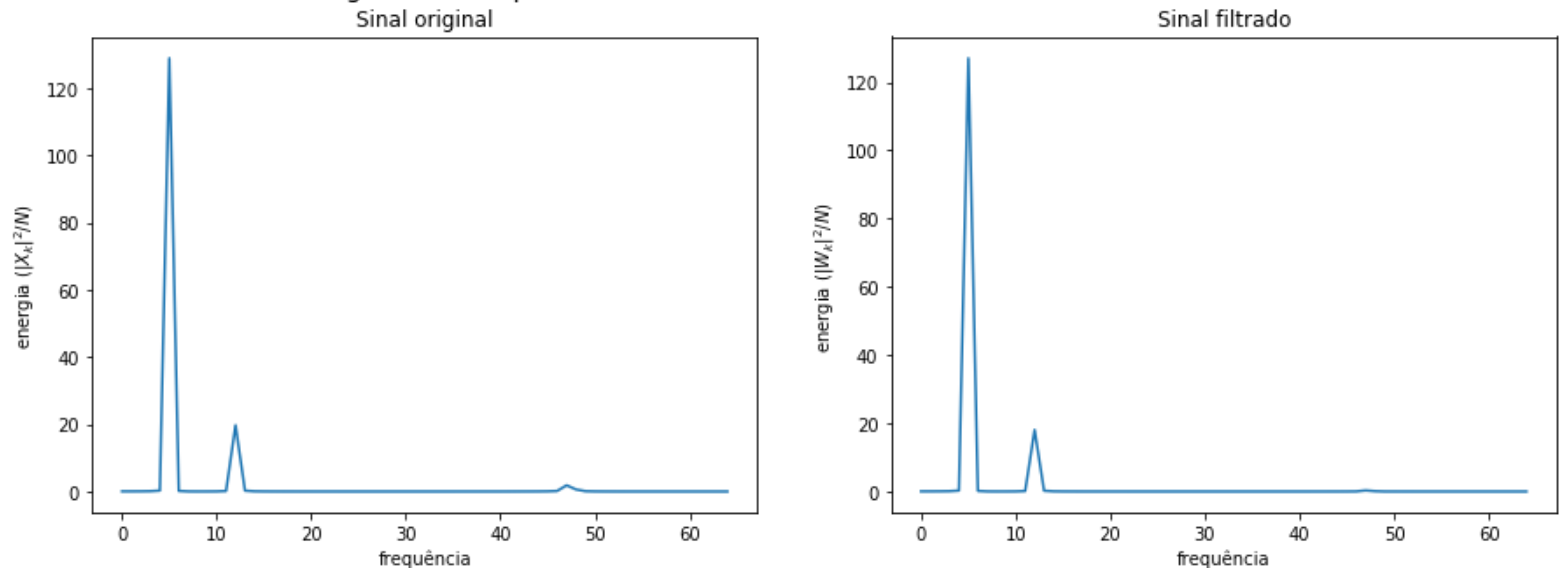


```
In [9]: # Exemplo 4.1: Aplicando o filtro da média na função do exemplo na seção 2.3.
T=1;N=128;t=np.linspace(0,T,N);x = 2*np.cos(2*m.pi*5*t) + 0.8*np.sin(2*m.pi*12*
t) + 0.3*np.cos(2*m.pi*47*t)
w = 0.5*(x+np.roll(x,1)) # saída do filtro da média.
fig, ax = plt.subplots(1,2,figsize=(15,5))
ax[0].set_title("Sinal original");ax[0].plot(t,x);ax[0].set_xlabel('t [s]');ax[
0].set_ylabel('x(t)')
ax[1].set_title("Sinal filtrado");ax[1].plot(t,w);ax[1].set_xlabel('t [s]');ax[
1].set_ylabel('w(t)')
fig.suptitle("Figura 4.1 - Sinal ruidoso e sinal filtrado com filtro da média",
fontsize=15)
plt.show()
```

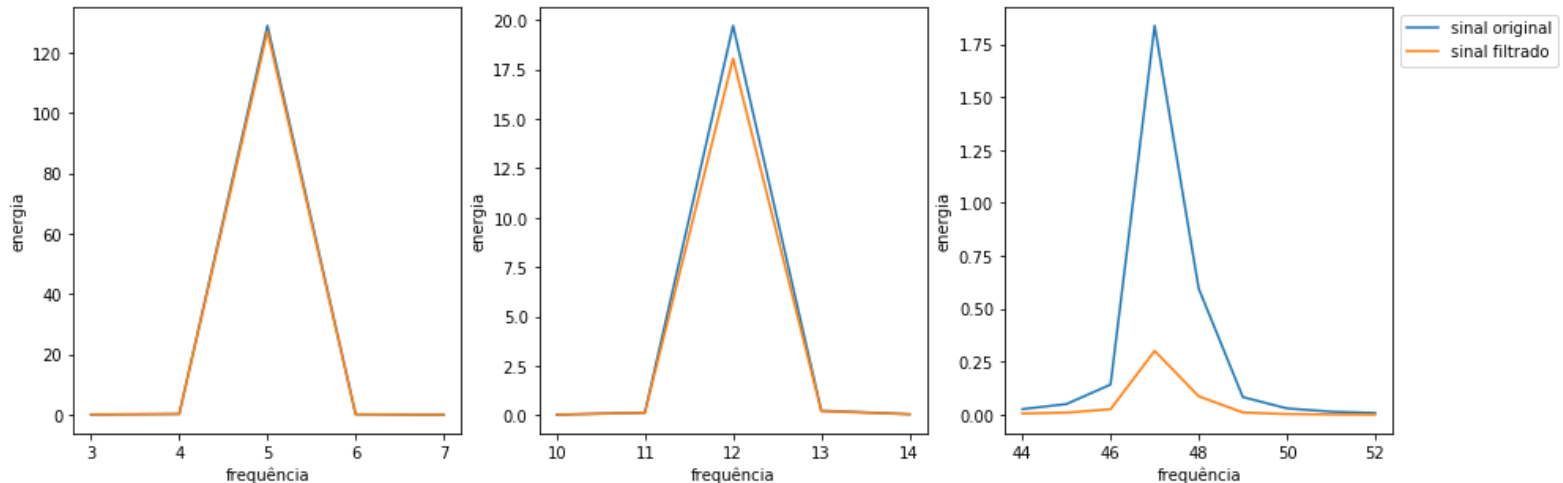


```
In [10]: X = np.fft.fft(x);c = X/N;E = N*abs(c)**2
W = np.fft.fft(w);d = W/N;F = N*abs(d)**2
fig, ax = plt.subplots(1,2,figsize=(15,5))
ax[0].set_title("Sinal original");ax[0].plot(E[:N//2+1]);ax[0].set_xlabel('freq
uência');ax[0].set_ylabel('energia  $(|X_k|^2/N)$ ')
ax[1].set_title("Sinal filtrado");ax[1].plot(F[:N//2+1]);ax[1].set_xlabel('freq
uência');ax[1].set_ylabel('energia  $(|W_k|^2/N)$ ')
fig.suptitle("Figura 4.2 - Espectro do sinal ruidoso e sinal filtrado com filtro
da média", fontsize=15)
plt.show()
```

Figura 4.2 - Espectro do sinal ruidoso e sinal filtrado com filtro da média



```
In [11]: # Como é difícil observar as diferenças dos espectros,
# as 3 figuras a seguir fazem um zoom dos 3 picos dos
# espectros de x (sinal original) e w (sinal filtrado)
fig, ax = plt.subplots(1, 3, figsize=(15,5))
ax[0].plot(np.arange(3,8),E[3:8], label='sinal original');ax[0].plot(np.arange(
3,8),F[3:8], label='sinal filtrado')
ax[1].plot(np.arange(10,15),E[10:15], label='sinal original');ax[1].plot(np.ara
nge(10,15),F[10:15], label='sinal filtrado')
ax[2].plot(np.arange(44,53),E[44:53], label='sinal original');ax[2].plot(np.ara
nge(44,53),F[44:53], label='sinal filtrado')
ax[0].set_xlabel('frequência');ax[1].set_xlabel('frequência');ax[2].set_xlabel('
frequência')
ax[0].set_ylabel('energia');ax[1].set_ylabel('energia');ax[2].set_ylabel('energ
ia')
plt.legend(loc='upper left', bbox_to_anchor=(1, 1))
plt.show()
```



## Formulação matricial do filtro da média

Observe que a equação  $w_n = \frac{1}{2}x_n + \frac{1}{2}x_{n-1}$ ,  $n = 0, 1, \dots, N - 1$  também pode ser escrita como

$$\begin{pmatrix} w_0 \\ w_1 \\ w_2 \\ \vdots \\ w_{N-1} \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & 0 & \dots & \dots & 0 & 1 \\ 1 & 1 & 0 & \dots & \dots & 0 \\ 0 & 1 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & \dots & 1 & 1 & 0 \\ 0 & \dots & \dots & 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_{N-1} \end{pmatrix}$$

A matriz que aparece na equação acima, chamada de *matriz circulante*, representa a operação de filtragem e está associada aos coeficientes da equação do filtro.

## Seção 4.2.2: Convolução

Nesta seção vamos considerar que todos os vetores estão em  $\mathbb{C}^N$  mas podem ser indexados em qualquer índice inteiro através da extensão periódica:

$$x_k = x_{k \bmod N}.$$

### Observação 4.1

Se  $x \in \mathbb{C}^N$  é estendido periodicamente, então somatórios de  $N$  índices seguidos sempre produzirão o mesmo resultado, ou seja,  $\forall m \in \mathbb{Z}$ :

$$\sum_{k=0}^{N-1} x_k = \sum_{k=0}^{N-1} x_{k+m} = \sum_{k=m}^{m+N-1} x_k.$$

## Definição de convolução

**Def. 4.2.1:** Se  $x, y \in \mathbb{C}^N$ , definimos a convolução circular de  $x$  e  $y$  como o vetor  $w \in \mathbb{C}^N$  dado por

$$w_n = \sum_{k=0}^{N-1} x_k y_{n-k}$$

e denotamos essa operação por  $w = x * y$ .

Por exemplo:

$$\begin{array}{cccccc}
 & x_0 & x_1 & x_2 & \dots & x_{N-1} \\
 & \times & \times & \times & \dots & \times \\
 w_0 := & y_0 & \underbrace{y_{-1}}_{y_{N-1}} & \underbrace{y_{-2}}_{y_{N-1}} & \dots & \underbrace{y_{-N+1}}_{y_1} \\
 & \hline
 & x_0 y_0 + x_1 y_{N-1} + \dots + x_{N-1} y_1 & \rightarrow w_0
 \end{array}$$

$$\begin{array}{cccccc}
 & x_0 & x_1 & x_2 & \dots & x_{N-1} \\
 & \times & \times & \times & \dots & \times \\
 w_1 := & y_1 & y_0 & \underbrace{y_{-1}}_{y_{N-1}} & \dots & \underbrace{y_{-(N-2)}}_{y_2} \\
 & \hline
 & x_0 y_1 + x_1 y_0 + \dots + x_{N-1} y_2 & \rightarrow & w_1
 \end{array}$$



em geral

$$\begin{array}{rcccl}
 & x_0 & x_1 & x_2 & \dots & x_{N-1} \\
 w_n = & \times & \times & \times & \dots & \times \\
 & y_n & y_{n-1} & y_{n-2} & \dots & y_{n-(N-1)} \\
 \hline
 & x_0 y_n + x_1 y_{n-1} + \dots + x_{N-1} y_{n-(N-1)} & \rightarrow w_n
 \end{array}$$

## Formulação matricial da convolução

Observe que a equação  $w_n = \sum_{k=0}^{N-1} x_k y_{n-k}$ ,  $n = 0, 1, \dots, N-1$  pode ser escrita como

$$w = \begin{pmatrix} y_0 & y_{N-1} & y_{N-2} & \cdots & y_1 \\ y_1 & y_0 & y_{N-1} & \cdots & y_2 \\ y_2 & y_1 & y_0 & \cdots & y_3 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_{N-1} & y_{N-2} & y_{N-3} & \cdots & y_0 \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ \vdots \\ x_{N-1} \end{pmatrix}$$

A matriz  $M_y$  que aparece na equação  $w = M_y x$  é chamada de *matriz circulante para y*. Observe que

$$(M_y)_{k,m} = y_{k-m}.$$

## Exemplo 4.2

o filtro da média corresponde à equação  $w = x * y$  onde

$y_0 = \frac{1}{2}, y_1 = \frac{1}{2}, y_n = 0, \forall n \neq 0, 1$ , e

$$M_y = \frac{1}{2} \begin{pmatrix} 1 & 0 & \dots & \dots & 0 & 1 \\ 1 & 1 & 0 & \dots & \dots & 0 \\ 0 & 1 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & \dots & 1 & 1 & 0 \\ 0 & \dots & \dots & 0 & 1 & 1 \end{pmatrix}$$

como visto anteriormente.

## Propriedades da convolução

**Teorema 4.2.1:** Sejam  $x, y, w \in \mathbb{C}^N$  e  $a, b \in \mathbb{C}$ . Então

1. Linearidade:  $x * (ay + bw) = a(x * y) + b(x * w)$
2. Comutatividade:  $x * y = y * x$
3. Formulação matricial: se  $w = x * y$ , então  $w = M_y x$ , onde

$$M_y = \begin{pmatrix} y_0 & y_{N-1} & \cdots & y_1 \\ y_1 & y_0 & \cdots & y_2 \\ \vdots & \vdots & \ddots & \vdots \\ y_{N-1} & y_{N-2} & \cdots & y_0 \end{pmatrix}$$

e além disso  $M_x M_y = M_{x*y}$

4. Associatividade:  $x * (y * w) = (x * y) * w$
5. Periodicidade: se  $x$  e  $y$  são tratados por extensão periódica, então

$$w_n = \sum_{k=0}^{N-1} x_k y_{n-k}$$

também está definido  $\forall n \in \mathbb{Z}$  e

$$w_n = w_{n \bmod N}$$

## Prova:

1. Exercício 4.5.

2. Seja  $w = x * y$  onde  $w_n = \sum_{k=0}^{N-1} x_k y_{n-k}$ . Então

$$w_n = \sum_{k=0}^{N-1} x_k y_{n-k} = \sum_{l=n}^{n-(N-1)} x_{n-l} y_l = \sum_{l=0}^{-(N-1)} x_{n-l} y_l = \sum_{l=N-1}^0 x_{n-l} y_l = \sum_{l=0}^{N-1} x_{n-l} y_l$$

a última expressão corresponde à definição de  $y * x$ , logo  $w = x * y = y * x$ .

---

## Prova (cont):

3. Já mostramos a construção da matriz, falta só  $M_x M_y = M_{x*y}$ . Mas

$$(M_x M_y)_{k,m} = \sum_{l=0}^{N-1} (M_x)_{k,l} (M_y)_{l,m} = \sum_{l=0}^{N-1} x_{k-l} y_{l-m} = \sum_{r=0}^{N-1} x_{(k-r-m)} y_r$$

$$\begin{aligned} \text{e note que } \sum_{r=0}^{N-1} x_{(k-m)-r} y_r &= \sum_{r=0}^{N-1} y_r x_{(k-m)-r} = (y * x)_{k-m} \\ &= w_{k-m} = (M_w)_{k,m} = (M_{x*y})_{k,m} \end{aligned}$$

Logo  $(M_x M_y)_{k,m} = (M_{x*y})_{k-m}$  e, portanto,  $M_x M_y = M_{x*y}$ .

4.  $x*(y*w) = x*(M_y w) = M_x(M_y w) = (M_x M_y)w = M_{(x*y)} w = (x*y)*w$
5. Exercício 4.6.