

Matlab 第 11 次作业参考答案

第一题 $f(x) = x^2, x \in [-\pi, \pi]$

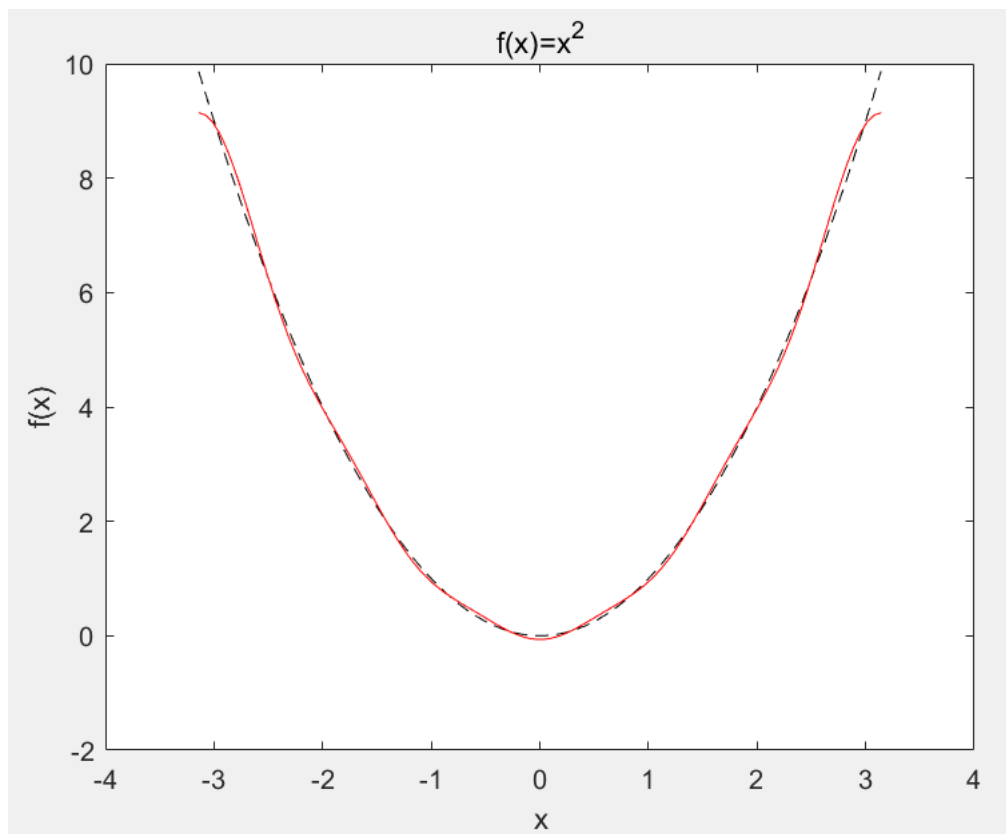
自行编写 M 文件实现对 $f(x)$ 的 Fourier 级数展开，展开阶数 $N = 5$ ，在同一幅

图中画出 $f(x)$ 与 Fourier 级数的部分和 $S_N(x) = \sum_{k=0}^N a_k \cos(kx) + b_k \sin(kx)$ 。要求给

出图像及代码，注意图像基本要素。

hw11_1.m

```
T=2*pi; w=2*pi/T;
syms x;
f=x^2;
a0=1/T*int(f,x,[-T/2 T/2]);
an=zeros(5,1);bn=zeros(5,1);
for i=1:5
    an(i)=2/T*int(f*cos(i*w*x),x,[-T/2 T/2]);
    bn(i)=2/T*int(f*sin(i*w*x),x,[-T/2 T/2]);
end
f=matlabFunction(f);
t=linspace(-pi,pi,100);
plot(t,f(t),'k--');
hold on;
Sn=a0;
for j=1:5
    Sn=Sn+an(j)*cos(j*t)+bn(j)*sin(j*t);
end
plot(t,Sn,'r-');
xlabel 'x';ylabel 'f(x)';title 'f(x)=x^2';
```



第二题

(1) 编写 M 文件求解向量 $[0, 1, 0, -1, 0, 1, 0, -1]$ 的离散傅里叶变换 (DFT)，要求给出结果和代码。

(2) 编写 M 文件求解向量 $[1, -i, 1, i]$ 的离散傅里叶逆变换 (inverse DFT)，要求给出结果和代码。

(1)

```
0.0000 + 0.0000i  
-0.0000 - 0.0000i  
0.0000 - 1.4142i  
0.0000 + 0.0000i  
0.0000 + 0.0000i  
0.0000 + 0.0000i  
-0.0000 + 1.4142i  
0.0000 - 0.0000i
```

hw11_2_1.m

```
clear;  
x=[0;1;0;-1;0;1;0;-1];  
n=length(x);  
omega=exp(-2*pi*1i/n);  
j=0:n-1;  
k=j';  
F=1/sqrt(n)*omega.^(k*j);  
y=F*x;  
disp(y);
```

(2)

```
1.0000 + 0.0000i  
1.0000 - 0.0000i  
1.0000 - 0.0000i  
-1.0000 + 0.0000i
```

hw11_2_2.m

```
clear;  
y=[1;-1i;1;1i];  
n=length(y);  
omega=exp(-2*pi*1i/n);  
j=0:n-1;  
k=j';  
F=1/sqrt(n)*omega.^(-k*j);  
x=F*y;  
disp(x);
```

第三题 $x = 0, 1, 2, \dots, 15$

自行编写函数文件 `myfft.m` 实现快速傅里叶变化 (FFT)。分别用 `myfft` 函数与 MATLAB 内置的 `fft` 函数求解 x 的快速傅里叶变化并进行比较。要求给出结果和代码。

```
1.0e+02 *  
  
1.2000 + 0.0000i    1.2000 + 0.0000i  
-0.0800 + 0.4022i   -0.0800 + 0.4022i  
-0.0800 + 0.1931i   -0.0800 + 0.1931i  
-0.0800 + 0.1197i   -0.0800 + 0.1197i  
-0.0800 + 0.0800i   -0.0800 + 0.0800i  
-0.0800 + 0.0535i   -0.0800 + 0.0535i  
-0.0800 + 0.0331i   -0.0800 + 0.0331i  
-0.0800 + 0.0159i   -0.0800 + 0.0159i  
-0.0800 + 0.0000i   -0.0800 + 0.0000i  
-0.0800 - 0.0159i   -0.0800 - 0.0159i  
-0.0800 - 0.0331i   -0.0800 - 0.0331i  
-0.0800 - 0.0535i   -0.0800 - 0.0535i  
-0.0800 - 0.0800i   -0.0800 - 0.0800i  
-0.0800 - 0.1197i   -0.0800 - 0.1197i  
-0.0800 - 0.1931i   -0.0800 - 0.1931i  
-0.0800 - 0.4022i   -0.0800 - 0.4022i
```

由上可见自行编写的 `myfft` 函数与内置的 `fft` 函数运行结果完全一致

`hw11_3.m`

```
x=(0:15)';  
z=myfft(x);  
y=fft(x);  
disp([z,y]);
```

`myfft.m`

```
function z=myfft(x)  
%z=myfft(x)  
n=length(x);  
m=ceil(log2(n)); N=2^m;  
x=[x;zeros(N-n,1)];  
order=zeros(N,1); order(1)=1;  
X=zeros(N,1); i=1;  
while i<N  
    order(i+1:2*i)=order(1:i)+2^(m-log2(i)-1);  
    i=2*i;  
end  
x_order=x(order(:));
```

```

omega=exp(-pi*1i);
for j=1:N/2
    X(2*j-1:2*j)=[1 1;1 omega]*x_order(2*j-1:2*j);
end
for k=1:m-1
    a=2^(k+1);
    omegaN=exp(-2*pi*1i/a);
    for k1=0:2^(m-k-1)-1
        X1=X(a*k1+1:(a*k1+a/2));
        X2=X(a*k1+a/2+1:a*(k1+1));
        X(a*k1+1:(a*k1+a/2))=X1+omegaN.^(0:a/2-1).'*X2;
        X(a*k1+a/2+1:a*(k1+1))=X1-omegaN.^(0:a/2-1).'*X2;
    end
end
z=X;
end

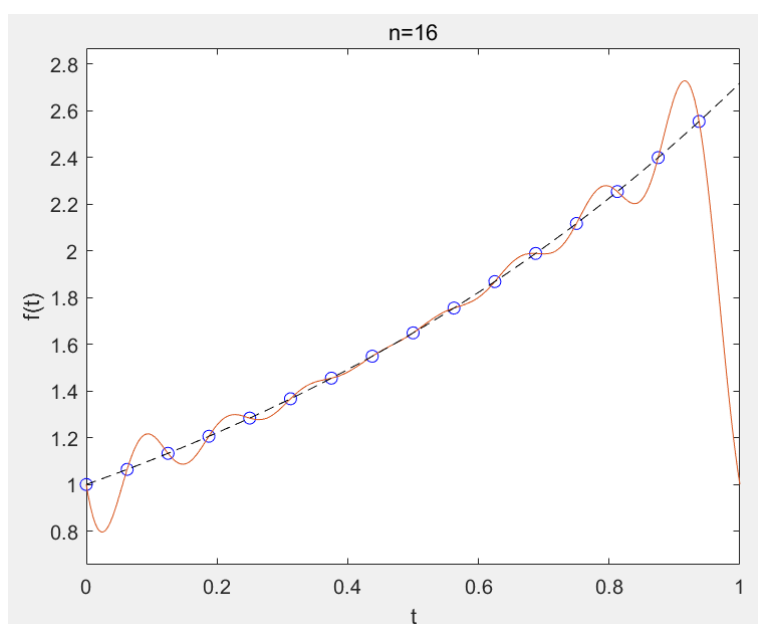
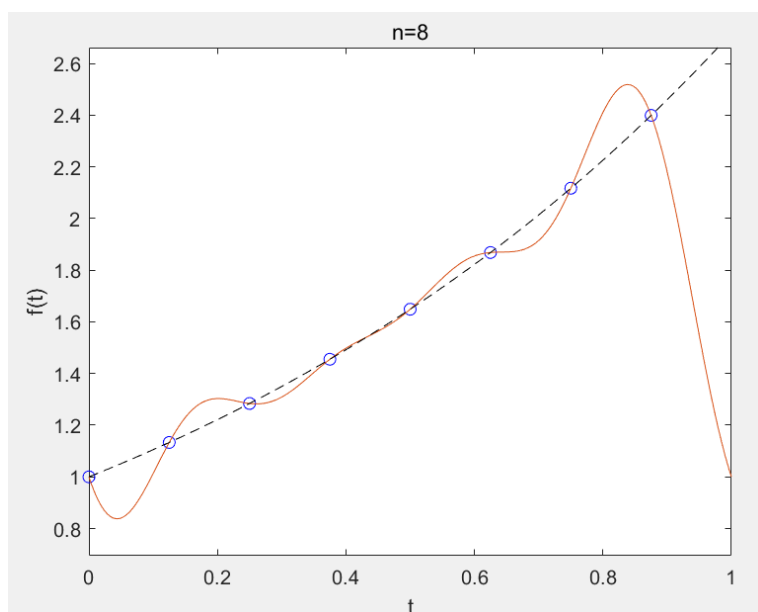
```

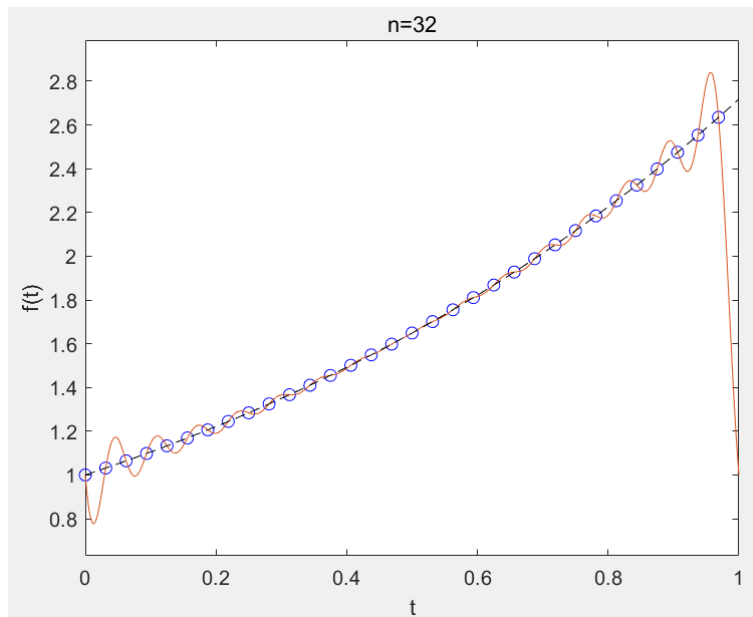
第四题 $f(t) = e^t$, $t \in [0, 1]$

(1) 取 $n=8$ ，取数据点 $(j/n, f(j/n))$, $j=0, 1, \dots, n-1$ ，用 DFT Interpolation Method 对这些数据点进行插值，画出原函数、数据点与插值曲线。要求给出图像和代码。

(2) 再分别取 $n=16$ 和 $n=32$ ，重复 (1) 中操作，对三种情况进行比较。要求给出图像、代码和比较。

(2) 再分别取 $n=16$ 和 $n=32$ ，重复 (1) 中操作，对三种情况进行比较。要求给出图像、代码和比较。





hw11_4_1.m

```

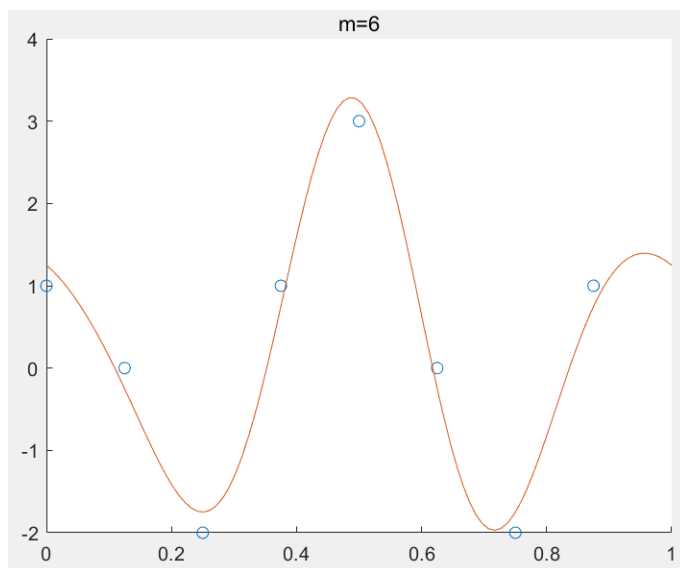
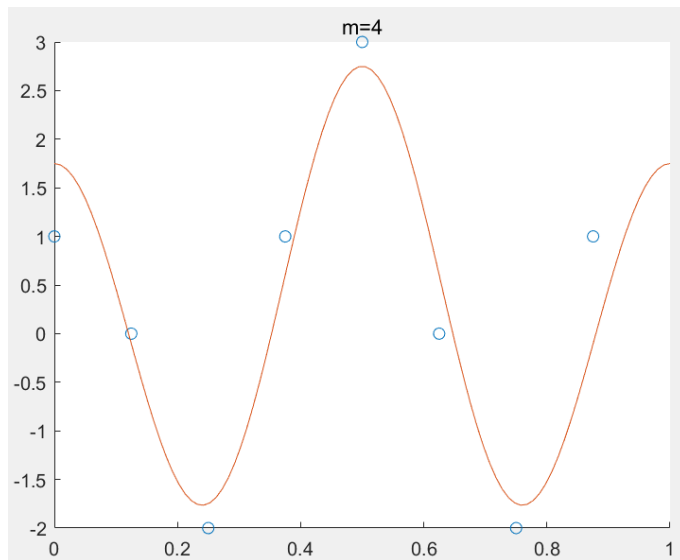
for i=[8,16,32]
    f=@(t)exp(t);
    n=i;
    tj=0:1/n:1-1/n;
    x=f(tj);
    n=length(x);
    y=fft(x)/sqrt(n);
    a=real(y); b=imag(y);
    syms t
    P=a(1)+a(n/2+1)*cos(n*pi*t);
    for k=2:n/2
        P=P+2*(a(k)*cos(2*(k-1)*pi*t)-b(k)*sin(2*(k-1)*pi*t));
    end
    P=P/sqrt(n);
    figure;
    plot(tj,x,'ob');
    hold on
    ezplot(P,[0,1]);
    tt=linspace(0,1,100);
    plot(tt,f(tt),'--k')
    xlabel 't';ylabel 'f(t)';title (['n=',num2str(i)]);
end

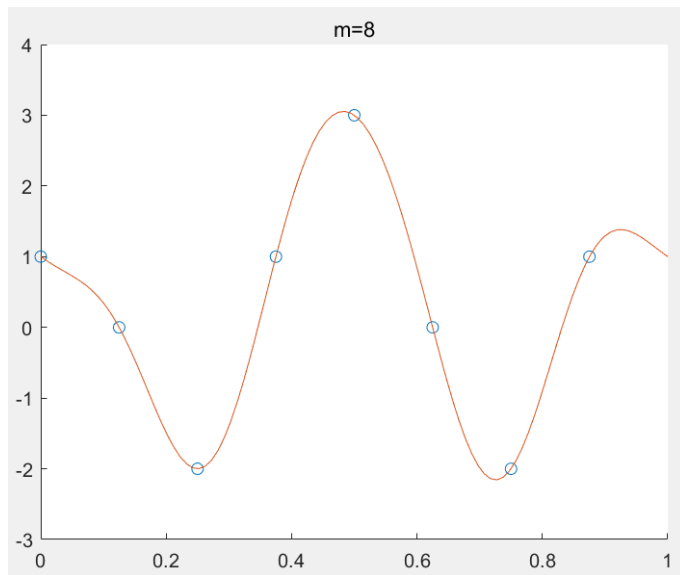
```

第五题

t	0	1/8	1/4	3/8	1/2	5/8	3/4	7/8
y	1	0	-2	1	3	0	-2	1

编写 M 文件，根据以上数据求解阶数分别为 4，6，8 的最小二乘三角逼近函数，分别画出函数曲线和数据点，要求给出图像和代码。





```

clear;
tj=0:1/8:7/8;
x=[1;0;-2;1;3;0;-2;1];
n=length(x);
omega=exp(-2*pi*1i/n);
j=0:n-1;
k=j';
F=1/sqrt(n)*omega.^(k*j);
y=F*x;
a=real(y); b=imag(y);
m=[4,6,8];
t=linspace(0,1,100);
for i=1:3
    P=a(1);
    for k=2:m(i)/2
        P=P+2*(a(k)*cos(2*(k-1)*pi*t)-b(k)*sin(2*(k-1)*pi*t));
    end

    if m(i)<n
        P=P+2*a(m(i)/2+1)*cos(m(i)*pi*t);
    else
        P=P+a(m(i)/2+1)*cos(m(i)*pi*t);
    end
    P=P/sqrt(n);
    figure; hold on
    plot(tj,x,'o',t,P,'-');

    title(['m=',num2str(m(i))]);
end

```