

Fourier Cosine Series

Kurt Bryan and SIMIODE

A Matlab script to compute the first few terms of the Fourier cosine expansion of a function $f(x)$ defined on an interval $0 \leq x \leq L$.

Define the interval and function

```
clear;
L = 2;
f = @(x) x - x.^2/3
```

Choose number n of Fourier cosine coefficients to compute and set up array "a[k]" to hold coefficients

```
n = 5
a = zeros(1,n+1); %Note a(k+1) is coefficient for cos(k*pi*x/L)
```

Compute coefficients numerically

```
syms x;
for k=1:n+1
    a(k) = 2/L*integral(@(x) f(x).*cos((k-1)*pi*x/L),0,L);
end
```

Form the Fourier cosine approximation

```
fcosapp = a(1)/2+sum(a(2:n+1).*cos((1:n)*pi*x/L));
```

Plot and compare to $f(x)$

```
xr = L*linspace(0,1);
fxr = subs(fcosapp,x,xr);
plot(xr,fxr, '-r',xr,f(xr),'-b') %Cosine approximation in red, function in blue
```

Alternatively, perform integrations symbolically, if possible

```
syms x;
aa = sym('b',[1,n+1]);
for k=1:n+1
    aa(k) = 2/L*int(f(x)*cos((k-1)*pi*x/L),[0,L]);
end
```

Form Fourier cosine series

```
fcosappsym = aa(1)/2+sum(aa(2:n+1).*cos((1:n)*pi*x/L));
```

Plot function $f(x)$ and Fourier approximation

```
fplot(f(x),[0,L], '-b');
hold on
fplot(fcosappsym,[0,L], '-r');
hold off
```