ME5204: Finite Element Analysis

Department of Mechanical Engineering

IIT Madras

Submitted by – Amit Pathak (MM21S024)

Assignment -1

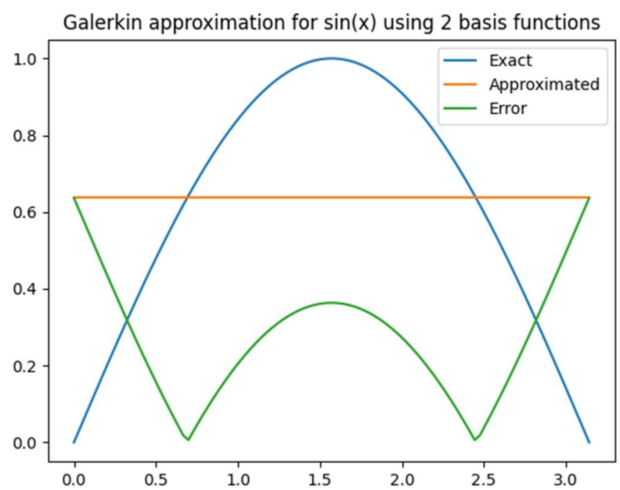
To solve sinx between 0 to pi using Galerkin approximation and plot the actual, approximated and error functions along with the L2.

The python code developed is as follows

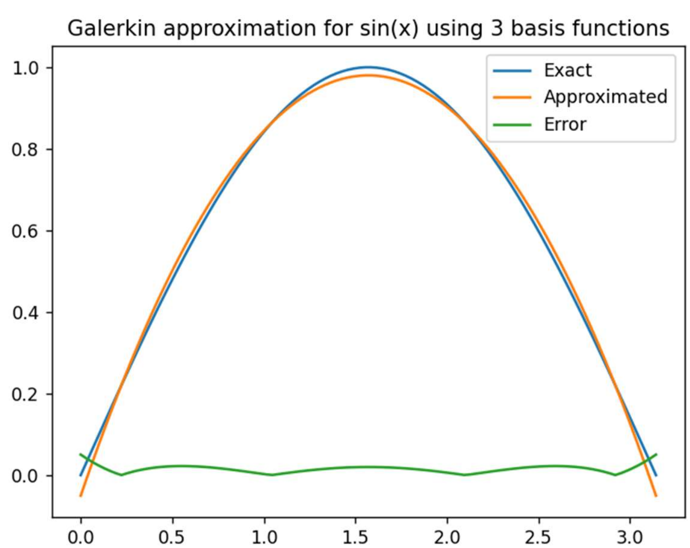
```
import numpy as np

import matplotlib.pyplot as plt


# Define the exact solution and the domain

exact = lambda x: np.sin(x)

x = np.linspace(0, np.pi, 100)


# Define the number of basis functions and the weight function

N = 2 # number of basis functions

w = lambda x: 1


# Generate the basis functions

phi = [lambda x, i=i: x**i for i in range(N)]


# Compute the stiffness matrix

A = np.zeros((N, N))

for i in range(N):

    for j in range(N):

        integrand = lambda x: phi[i](x) * phi[j](x) * w(x)

        A[i,j] = np.trapz(integrand(x), x)


# Compute the load vector

f = np.zeros(N)

for i in range(N):

    integrand = lambda x: exact(x) * phi[i](x) * w(x)

    f[i] = np.trapz(integrand(x), x)


# Solve the system of equations to obtain the coefficients

c = np.linalg.solve(A, f)


# Define the approximated solution

approx = lambda x: sum(c[i] * phi[i](x) for i in range(N))


# Compute the error and the L2 norm
```

```
error = np.abs(exact(x) - approx(x))

L2_norm = np.sqrt(np.trapz(error**2 * w(x), x))


# Plot the actual, approximated, and error functions, along with the L2 norm

plt.plot(x, exact(x), label='Exact')

plt.plot(x, approx(x), label='Approximated')

plt.plot(x, error, label='Error')

plt.legend()

plt.title('Galerkin approximation for sin(x) using {} basis functions'.format(N))

plt.show()

print("L2 norm of the error:", L2_norm)
```

The output of the code are as follows



L2 norm of the error: 0.5456834876194065



L2 norm of the error: 0.030685019773002524