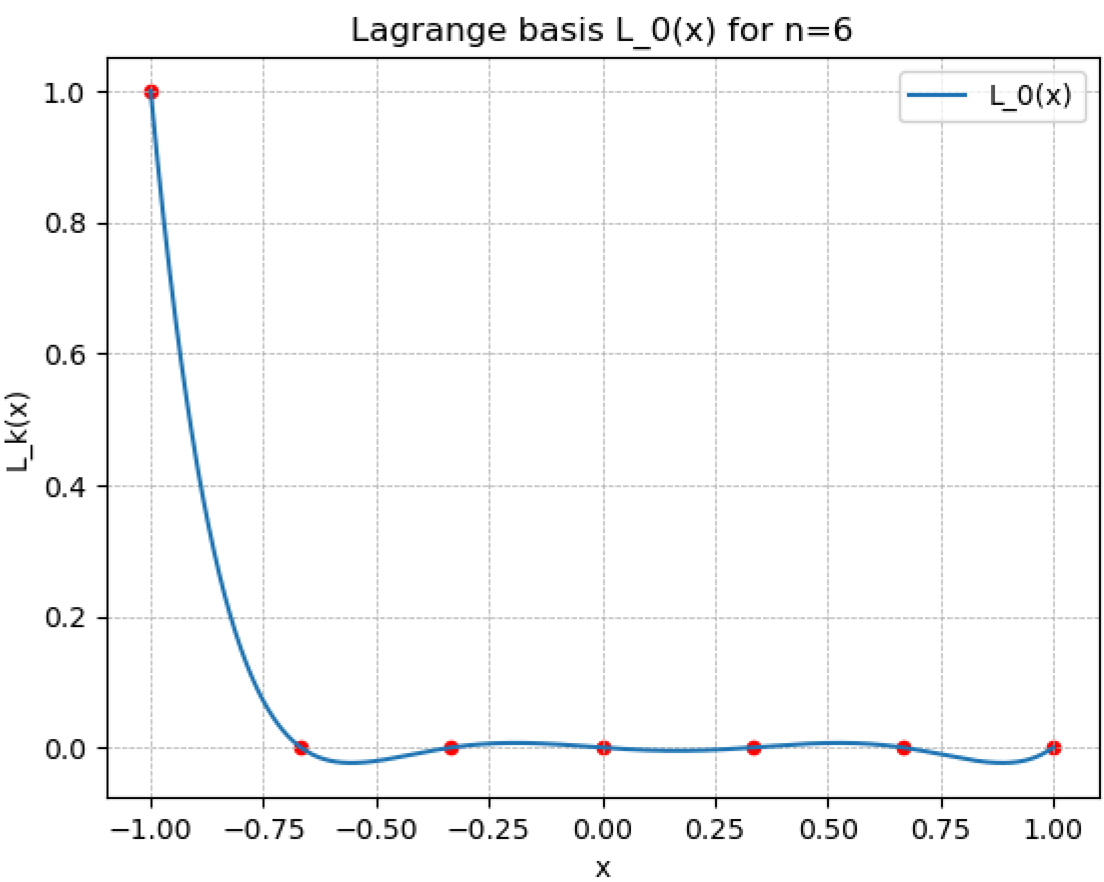
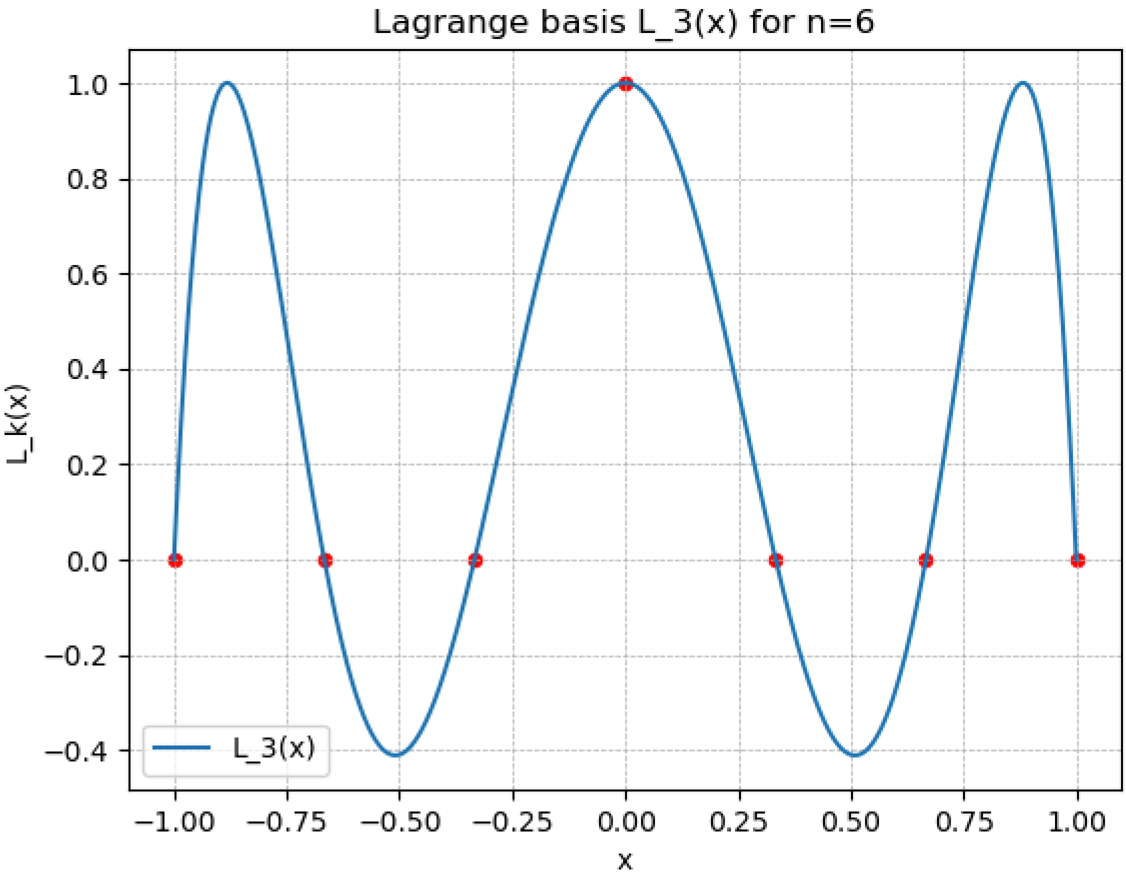
Computational Physics I - Polynomial Interpolation

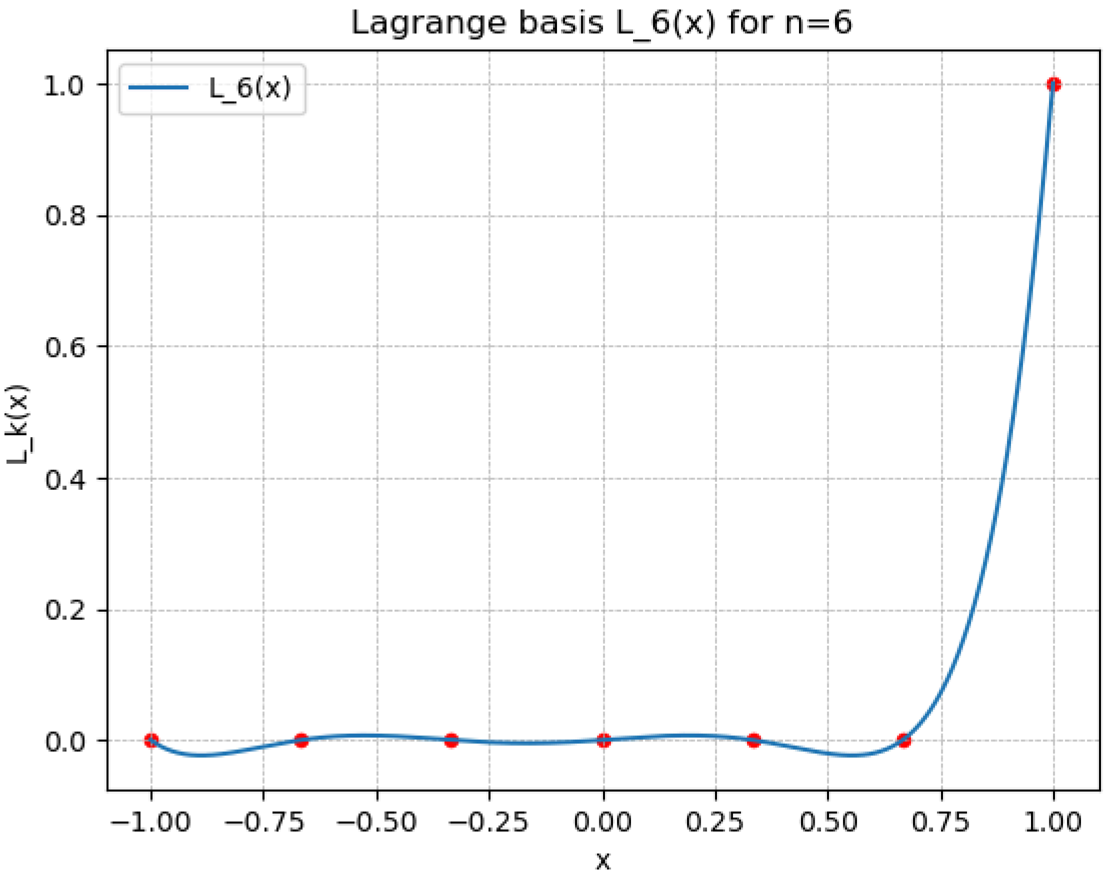
Giovanni Filomeno - K12315325

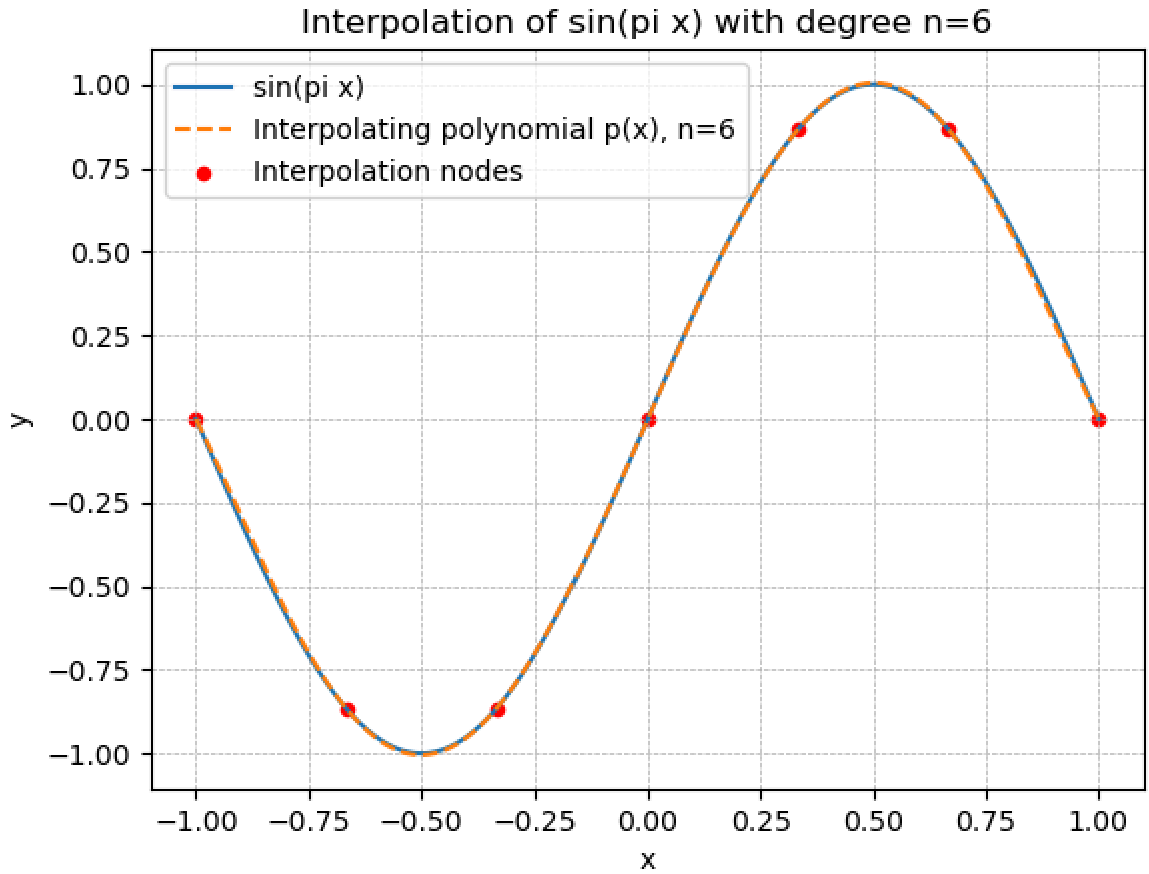
In [4]:
```
import numpy as np
import matplotlib.pyplot as plt

def equidistant_nodes(n: int):
    """Equidistant grid as per x_j = 2j/n - 1"""
    j = np.arange(n + 1)
    return 2 * j / n - 1

def lagrange_basis(x, nodes, k):
    """Lagrange basis polynomial L_k(x)"""
    x = np.asarray(x, dtype=float)
    n = len(nodes) - 1
    L = np.ones_like(x)
    xk = nodes[k]
    for j in range(n + 1):
        if j != k:
            L *= (x - nodes[j]) / (xk - nodes[j])
    return L

def interpolate_lagrange(x_eval, nodes, y_nodes):
    """Interpolating polynomial p(x)= sum f(x_k) * L_k(x)"""
    p = np.zeros_like(x_eval)
    for k in range(len(nodes)):
        p += y_nodes[k] * lagrange_basis(x_eval, nodes, k)
    return p

def f_fun(x):
    """First function"""
    return np.sin(np.pi * x)

def g_fun(x):
    """Second function"""
    return 1.0 / (1.0 + (3.0 * x)**2)

# Setup
x_plot = np.linspace(-1, 1, 1000)
nodes_list = [6, 10, 14]   # degrees to test

for n in nodes_list:
    nodes = equidistant_nodes(n)

    # (a) Plot some L_k(x)
    ks = [0, n//2, n]
    for k in ks:
        plt.figure()
        Lk = lagrange_basis(x_plot, nodes, k)
        plt.plot(x_plot, Lk, label=f"L_{k}(x)")
        plt.scatter(nodes, (nodes == nodes[k]).astype(int), s=20, color='
        plt.title(f"Lagrange basis L_{k}(x) for n={n}")
        plt.xlabel("x")
        plt.ylabel("L_k(x)")
        plt.grid(True, linestyle="--", linewidth=0.5)
```

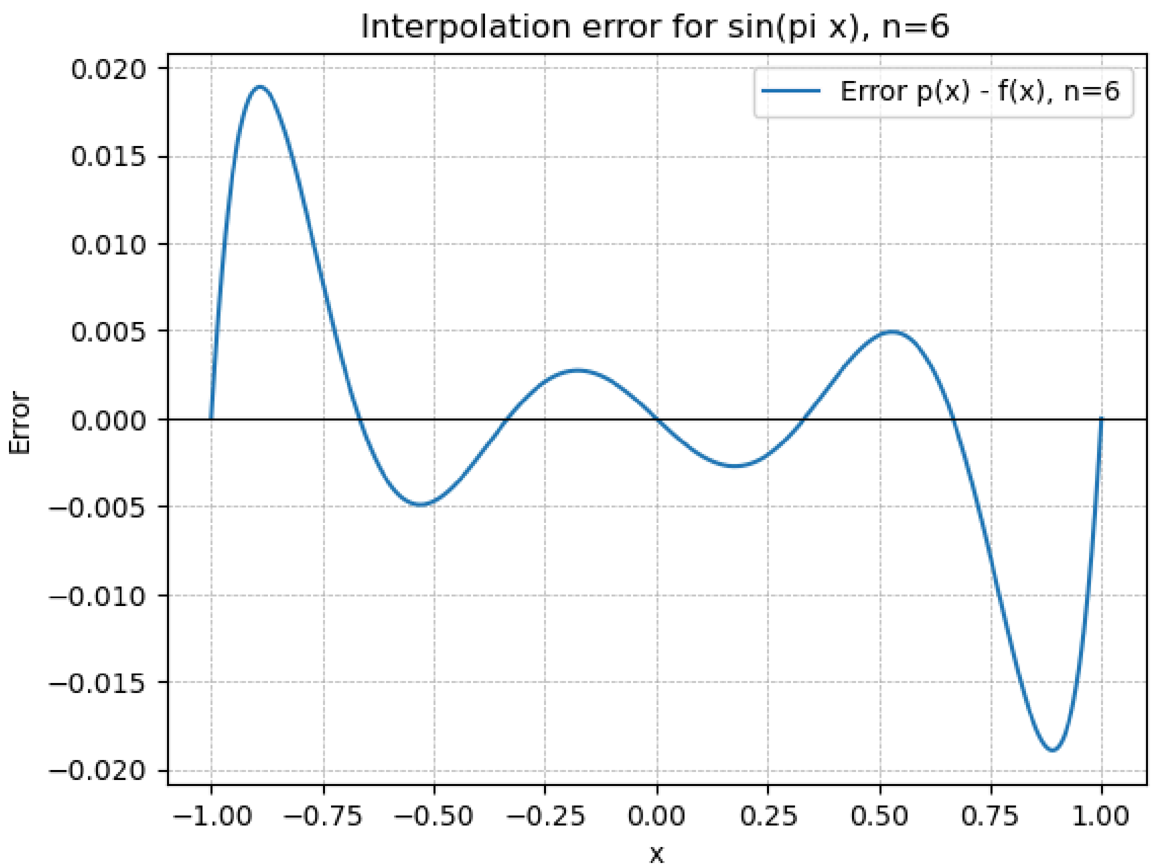
```python
        plt.legend()
        plt.show()

    # (b) Interpolation + Error
    for name, fun in [("sin(pi x)", f_fun), ("1 / (1 + (3x)^2)", g_fun)]:
        y_nodes = fun(nodes)
        p_vals = interpolate_lagrange(x_plot, nodes, y_nodes)
        f_vals = fun(x_plot)

        # Plot function vs interpolant
        plt.figure()
        plt.plot(x_plot, f_vals, label=f"{name}")
        plt.plot(x_plot, p_vals, "--", label=f"Interpolating polynomial p
        plt.scatter(nodes, y_nodes, color='red', s=20, label="Interpolati
        plt.title(f"Interpolation of {name} with degree n={n}")
        plt.xlabel("x")
        plt.ylabel("y")
        plt.grid(True, linestyle="--", linewidth=0.5)
        plt.legend()
        plt.show()

        # Plot interpolation error
        err = p_vals - f_vals
        plt.figure()
        plt.plot(x_plot, err, label=f"Error p(x) - f(x), n={n}")
        plt.axhline(0.0, color="black", linewidth=0.8)
        plt.title(f"Interpolation error for {name}, n={n}")
        plt.xlabel("x")
        plt.ylabel("Error")
        plt.grid(True, linestyle="--", linewidth=0.5)
        plt.legend()
        plt.show()
```

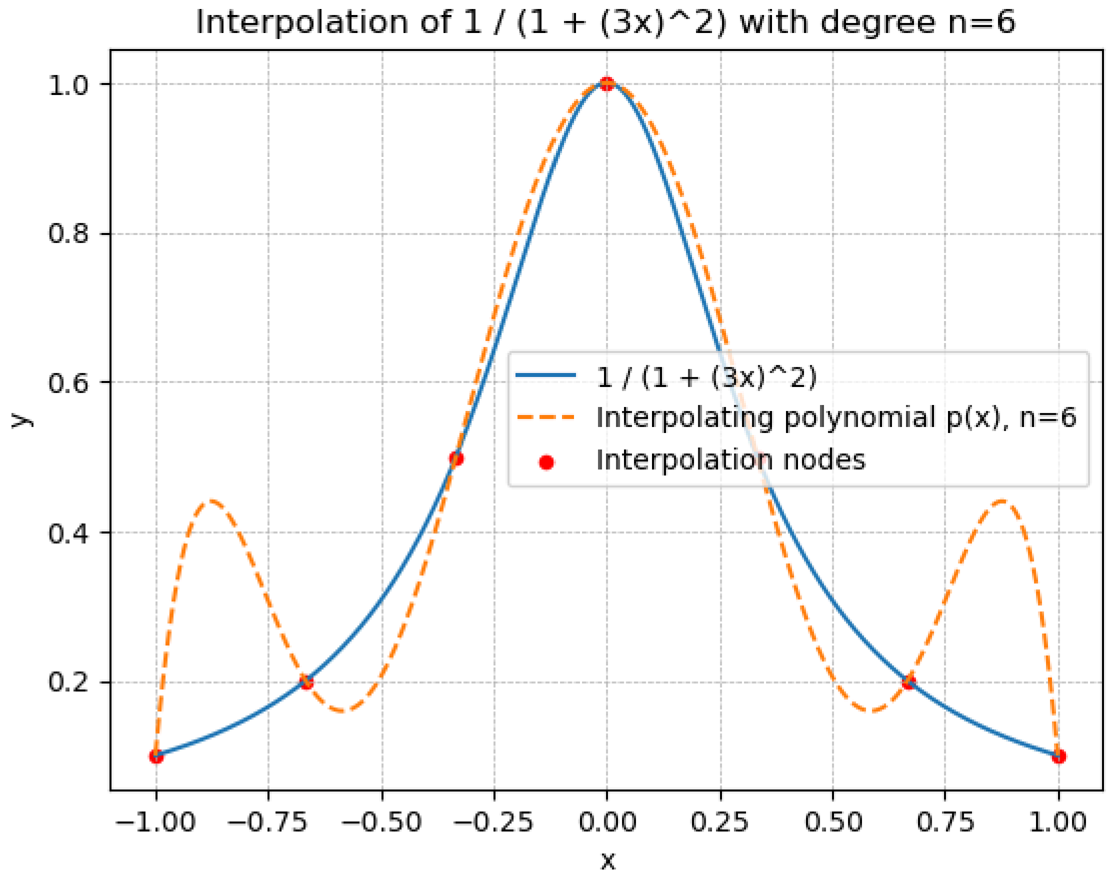


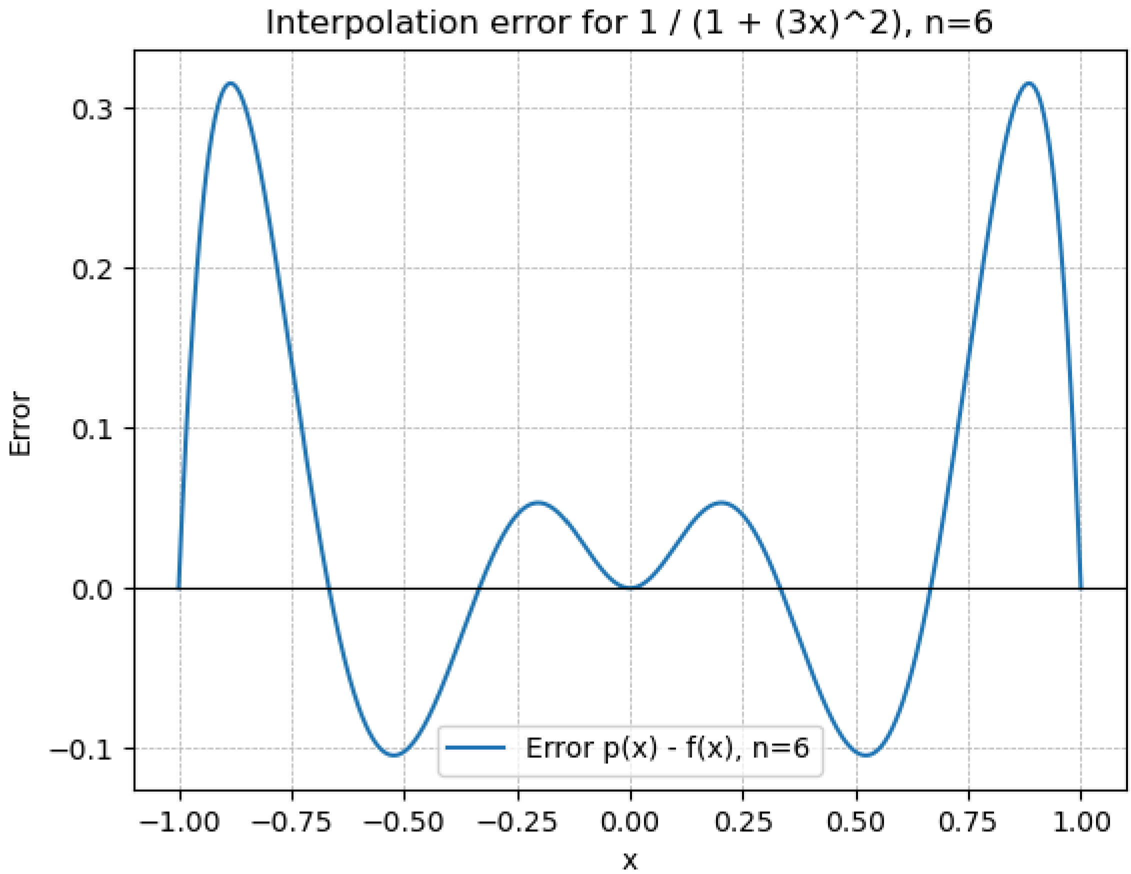

Lagrange basis L_3(x) for n=6
L_3(x)
L_k(x)
x

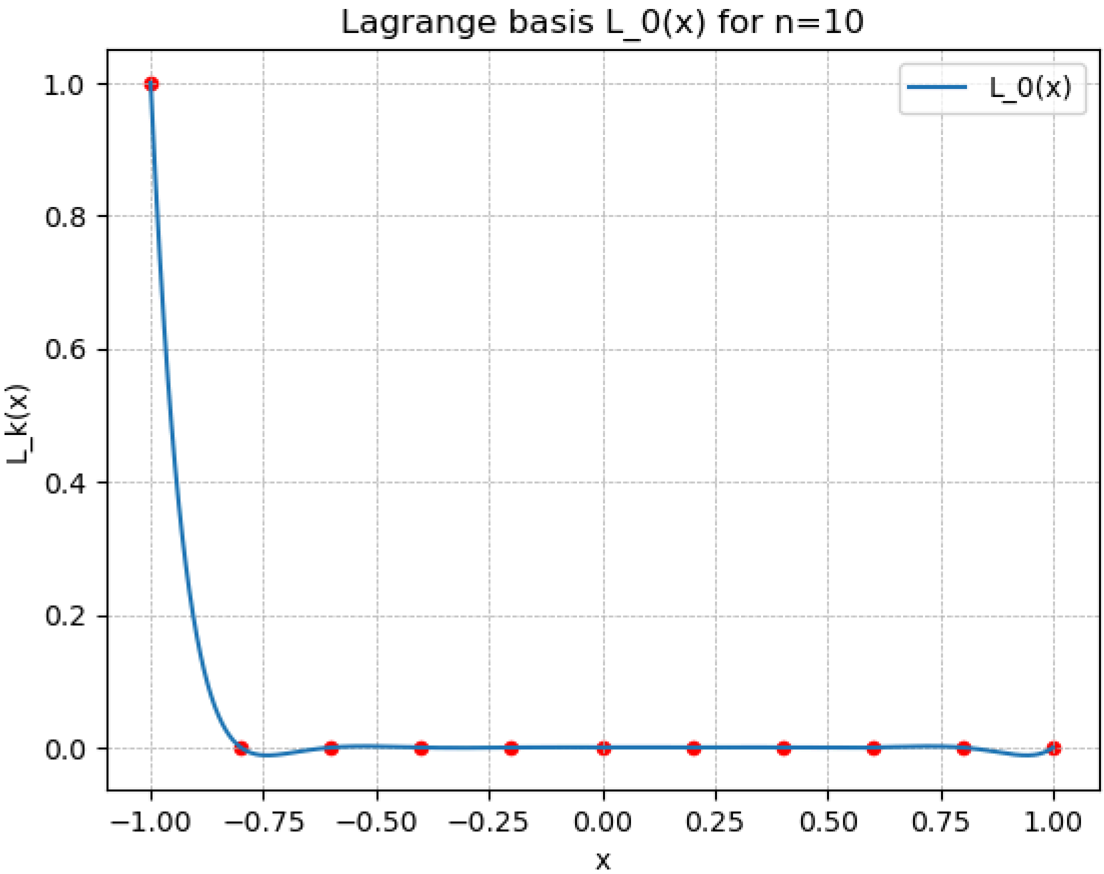

Lagrange basis L_6(x) for n=6
L_6(x)
L_k(x)
x


Interpolation of sin(pi x) with degree n=6
sin(pi x)
Interpolating polynomial p(x), n=6
Interpolation nodes
y
x
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

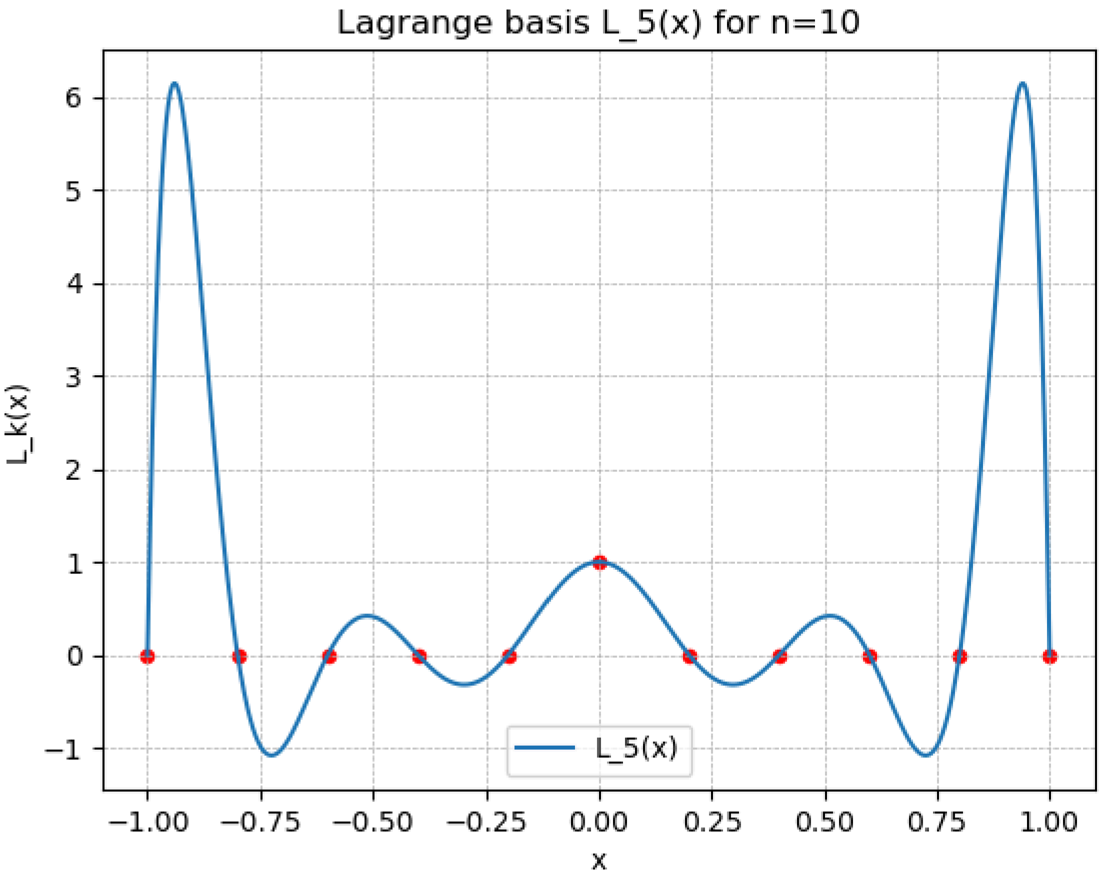

Interpolation error for sin(pi x), n=6
Error p(x) - f(x), n=6
Error
x
0.020
0.015
0.010
0.005
0.000
−0.005
−0.010
−0.015
−0.020
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00


Interpolation of 1 / (1 + (3x)^2) with degree n=6
1 / (1 + (3x)^2)
Interpolating polynomial p(x), n=6
Interpolation nodes
y
x
1.0
0.8
0.6
0.4
0.2
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

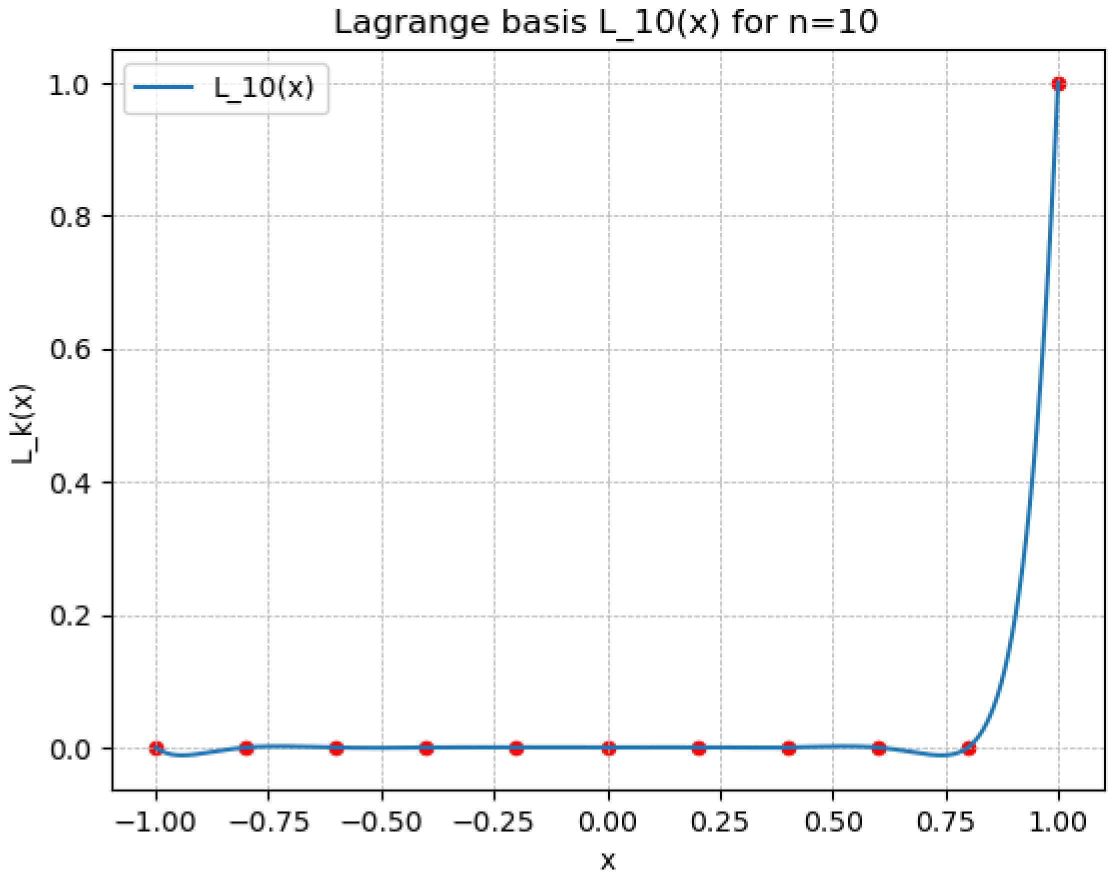

Interpolation error for 1 / (1 + (3x)^2), n=6
Error p(x) - f(x), n=6
Error
x
0.3
0.2
0.1
0.0
−0.1
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00



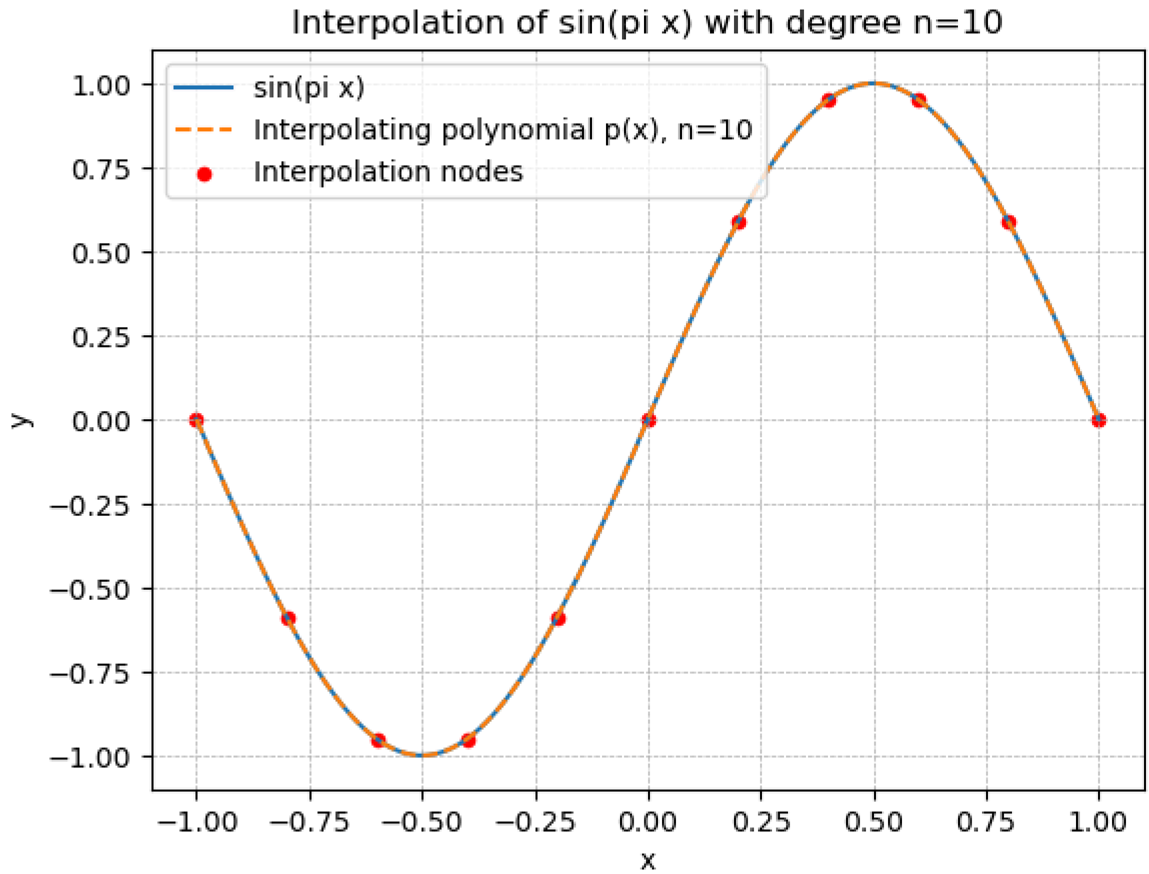
Lagrange basis L_0(x) for n=10
L_0(x)
L_k(x)
x
1.0
0.8
0.6
0.4
0.2
0.0
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00



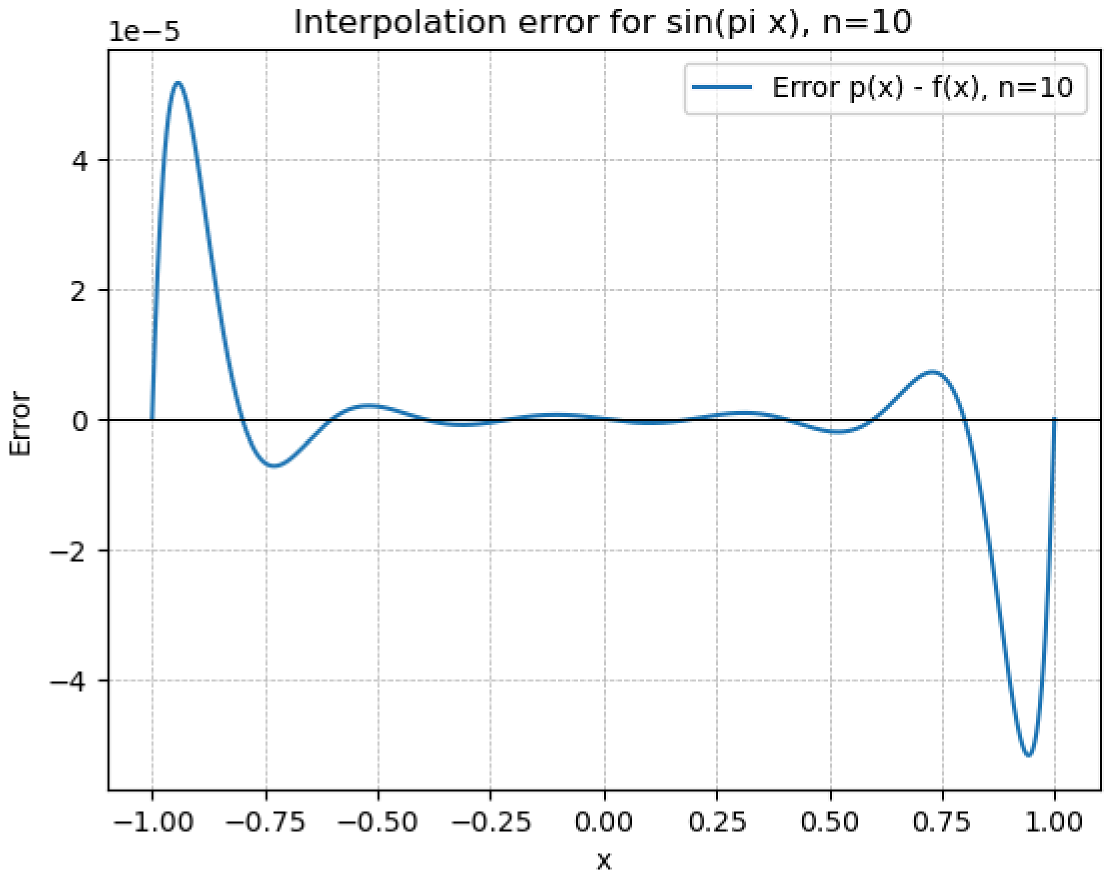
Lagrange basis L_5(x) for n=10
L_5(x)
L_k(x)
x
6
5
4
3
2
1
0
−1
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

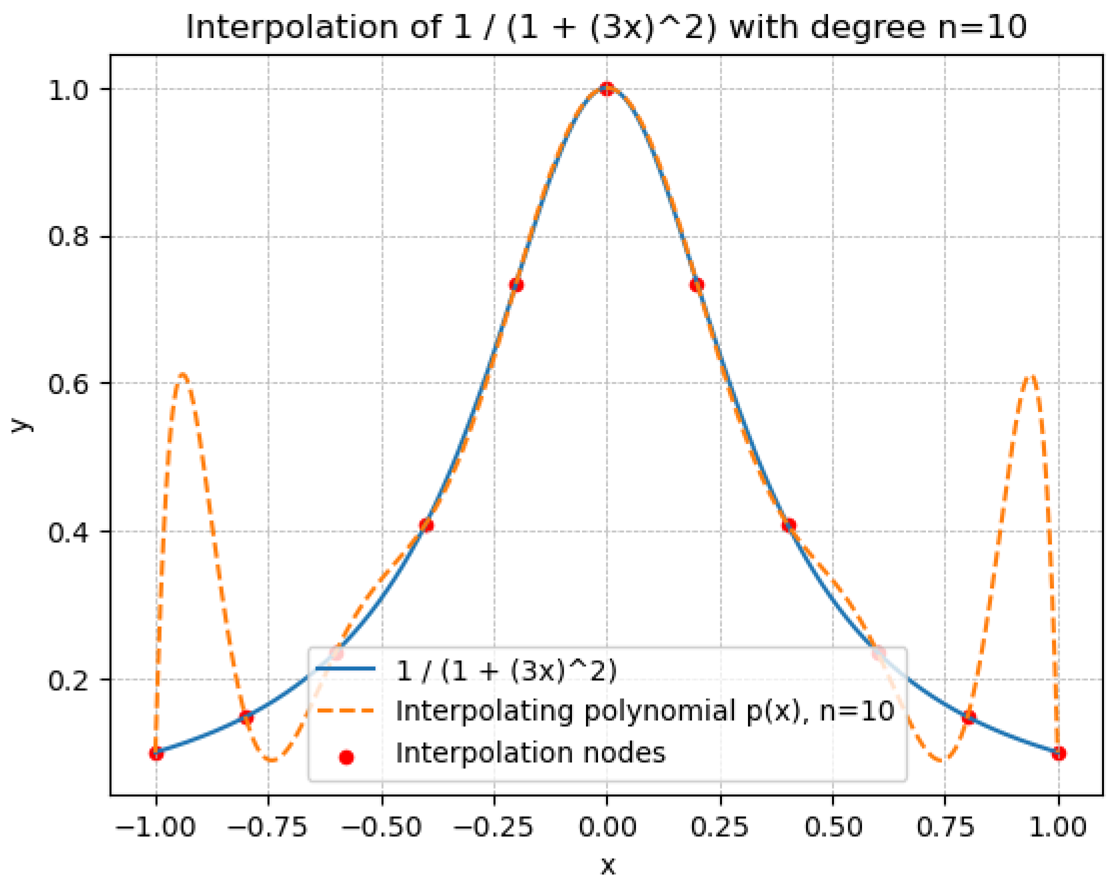


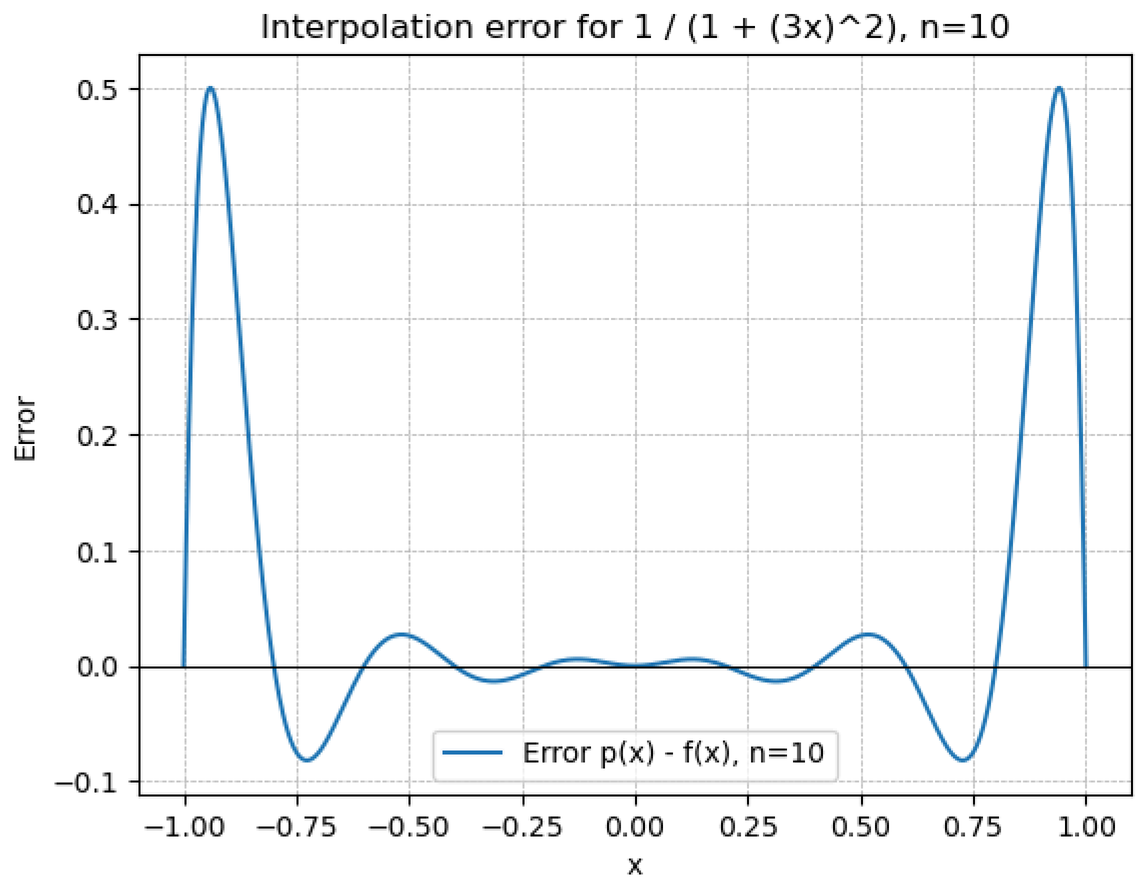
Lagrange basis L_10(x) for n=10
L_10(x)
L_k(x)
1.0
0.8
0.6
0.4
0.2
0.0
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
x



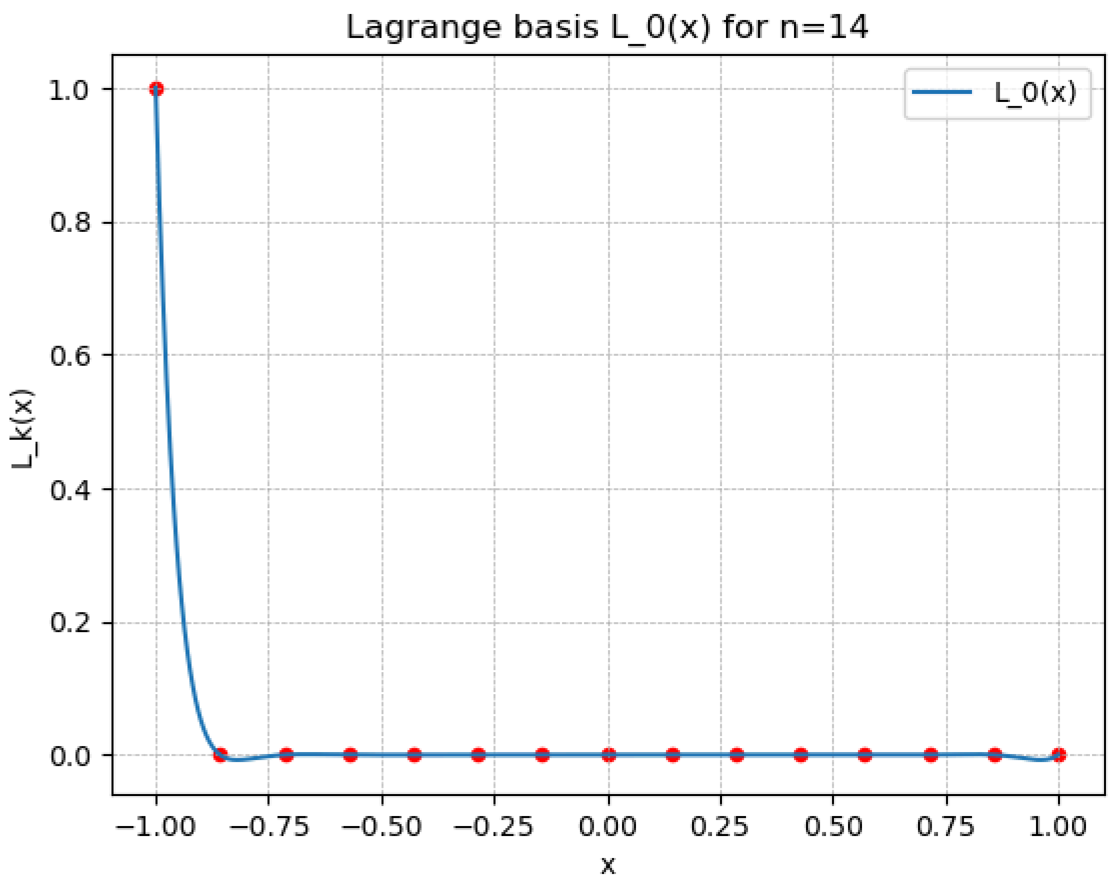
Interpolation of sin(pi x) with degree n=10
sin(pi x)
Interpolating polynomial p(x), n=10
Interpolation nodes
y
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
x


Interpolation error for sin(pi x), n=10
1e−5
Error p(x) - f(x), n=10
4
2
0
−2
−4
Error
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
x

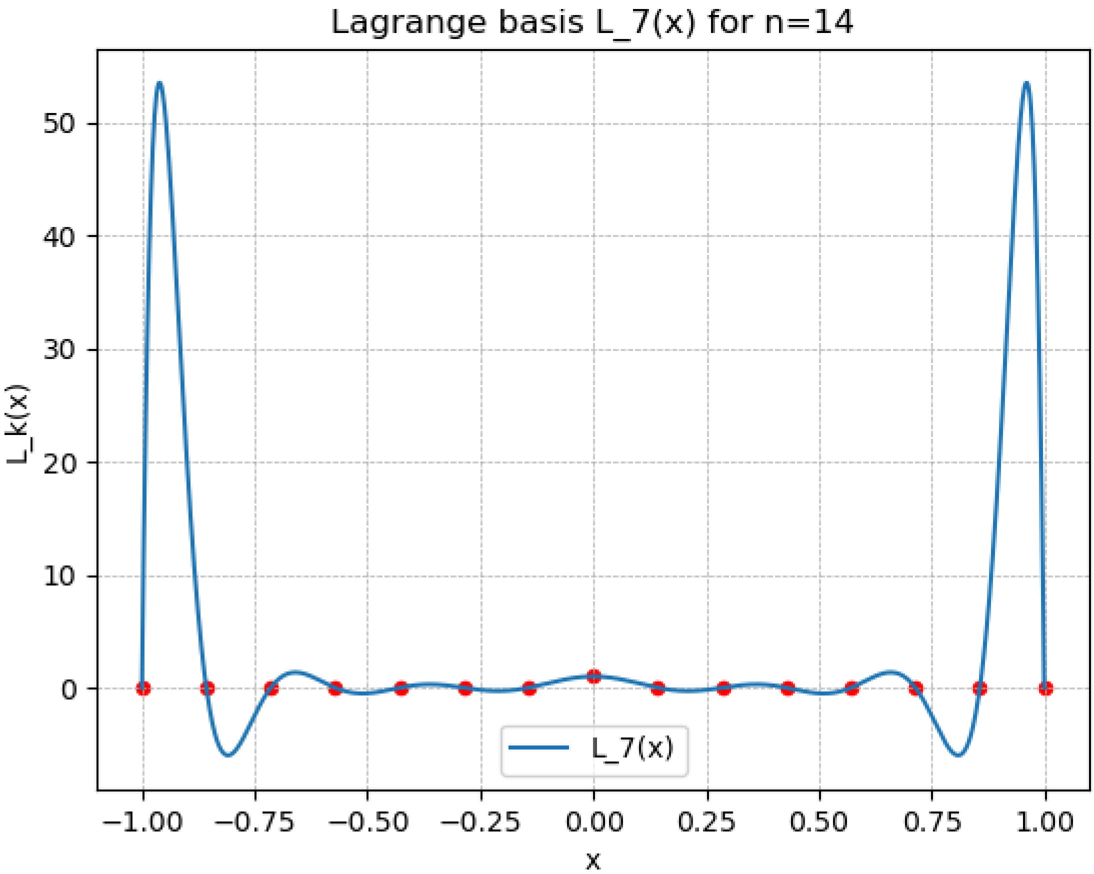

Interpolation of 1 / (1 + (3x)^2) with degree n=10
1.0
0.8
0.6
0.4
0.2
y
1 / (1 + (3x)^2)
Interpolating polynomial p(x), n=10
Interpolation nodes
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
x



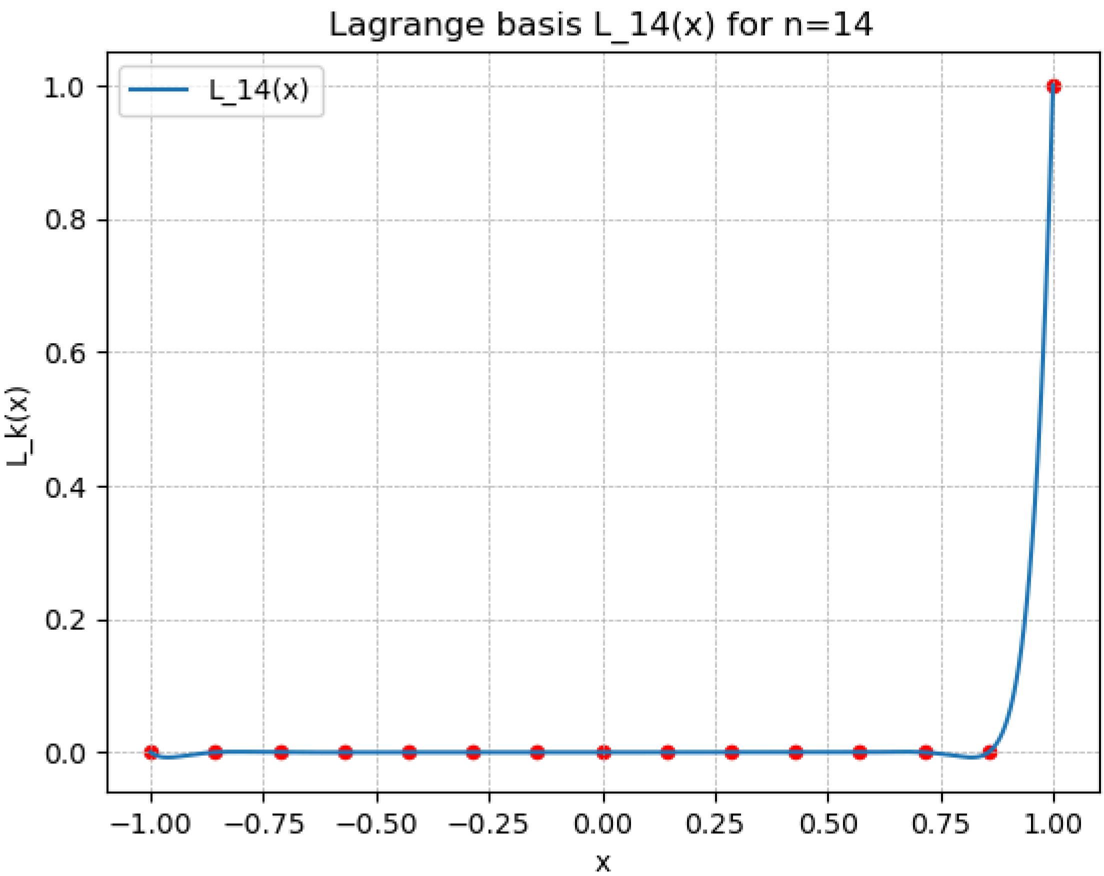
Interpolation error for 1 / (1 + (3x)^2), n=10
Error
x
Error p(x) - f(x), n=10
0.5
0.4
0.3
0.2
0.1
0.0
−0.1
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00



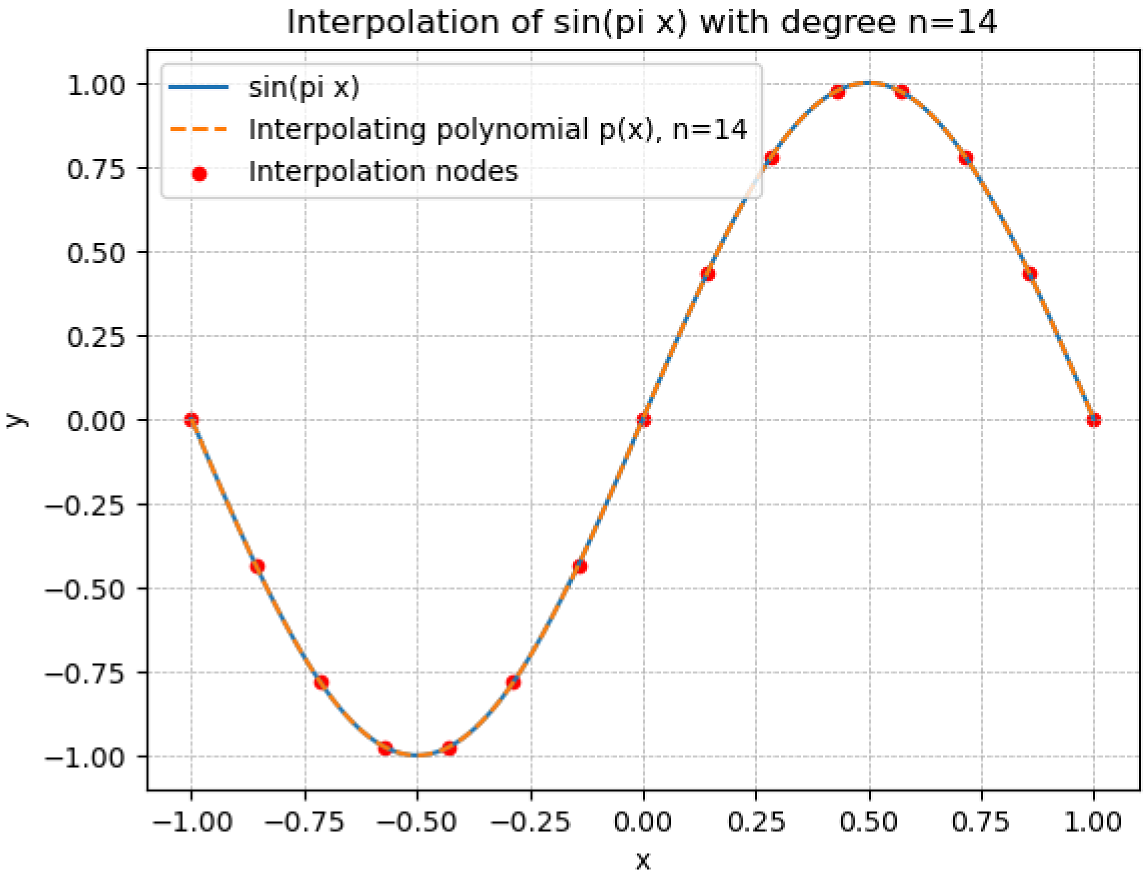
Lagrange basis L_0(x) for n=14
L_k(x)
x
L_0(x)
1.0
0.8
0.6
0.4
0.2
0.0
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00



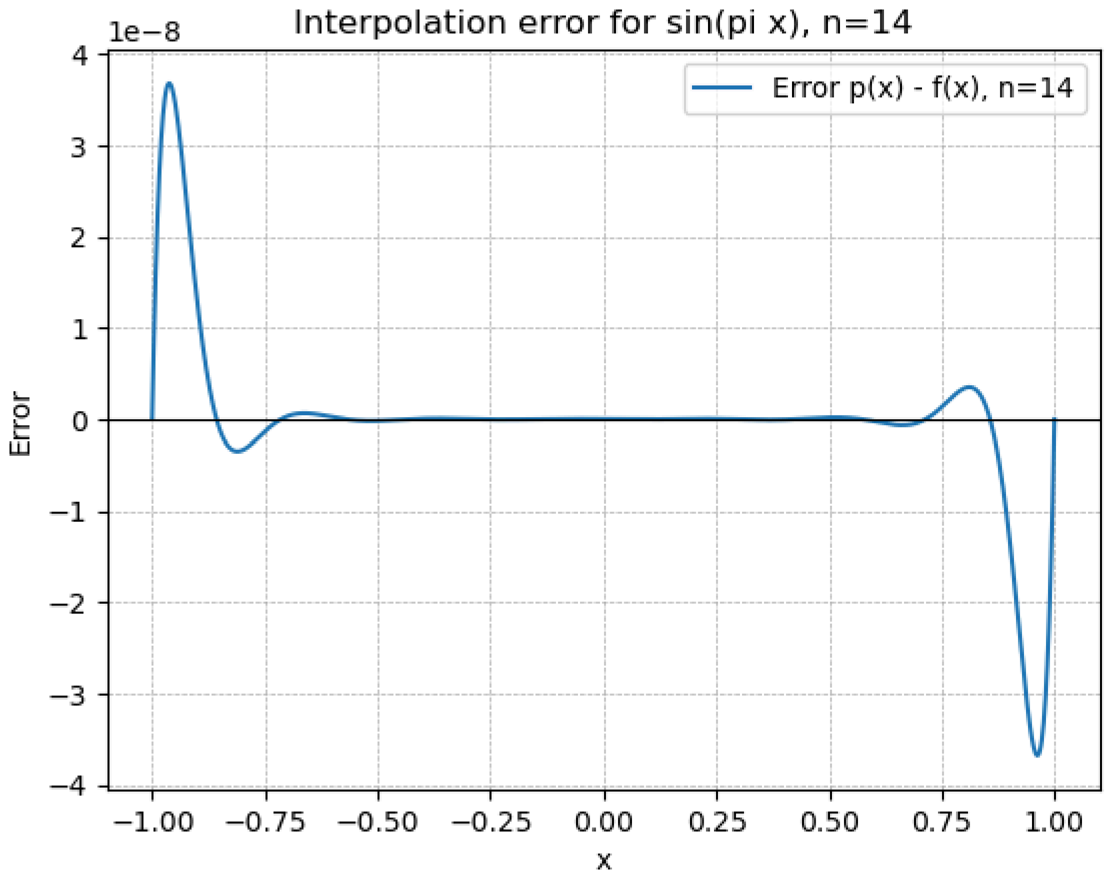
Lagrange basis L_7(x) for n=14
L_7(x)
L_k(x)
x
50
40
30
20
10
0
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00



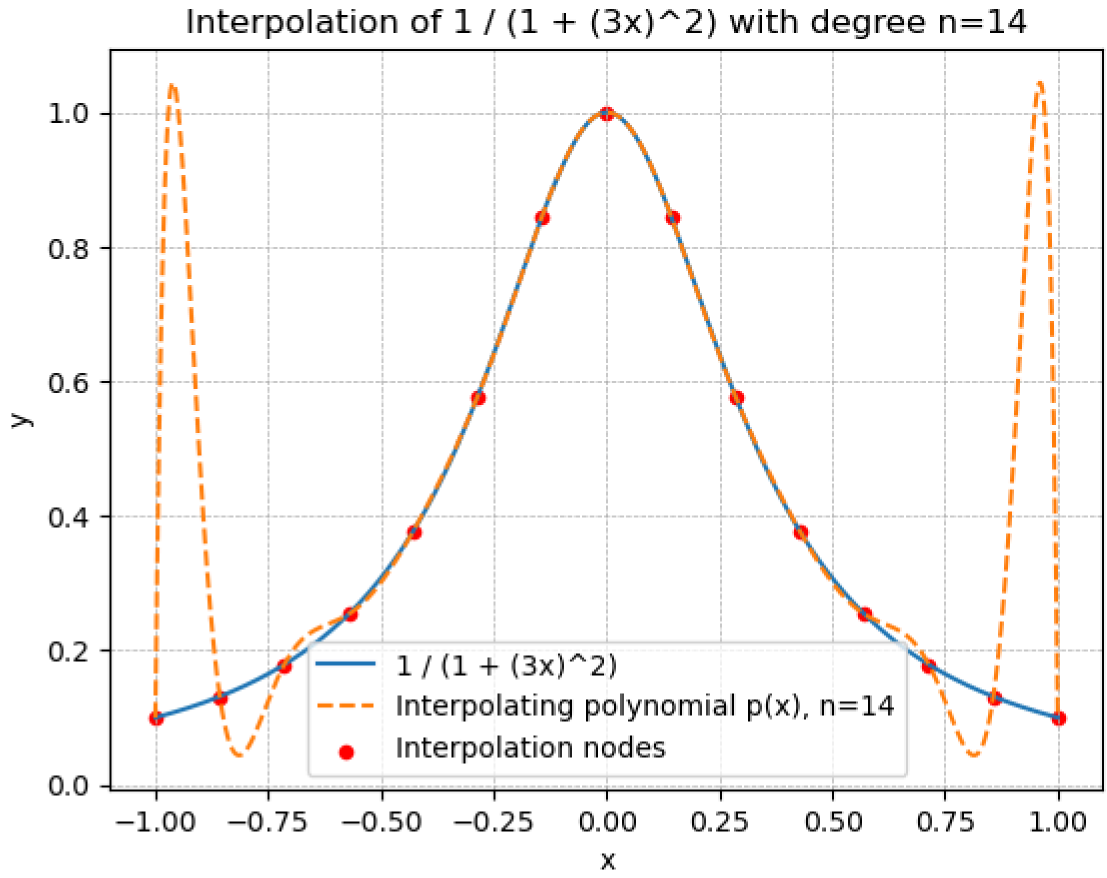
Lagrange basis L_14(x) for n=14
L_14(x)
L_k(x)
x
1.0
0.8
0.6
0.4
0.2
0.0
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

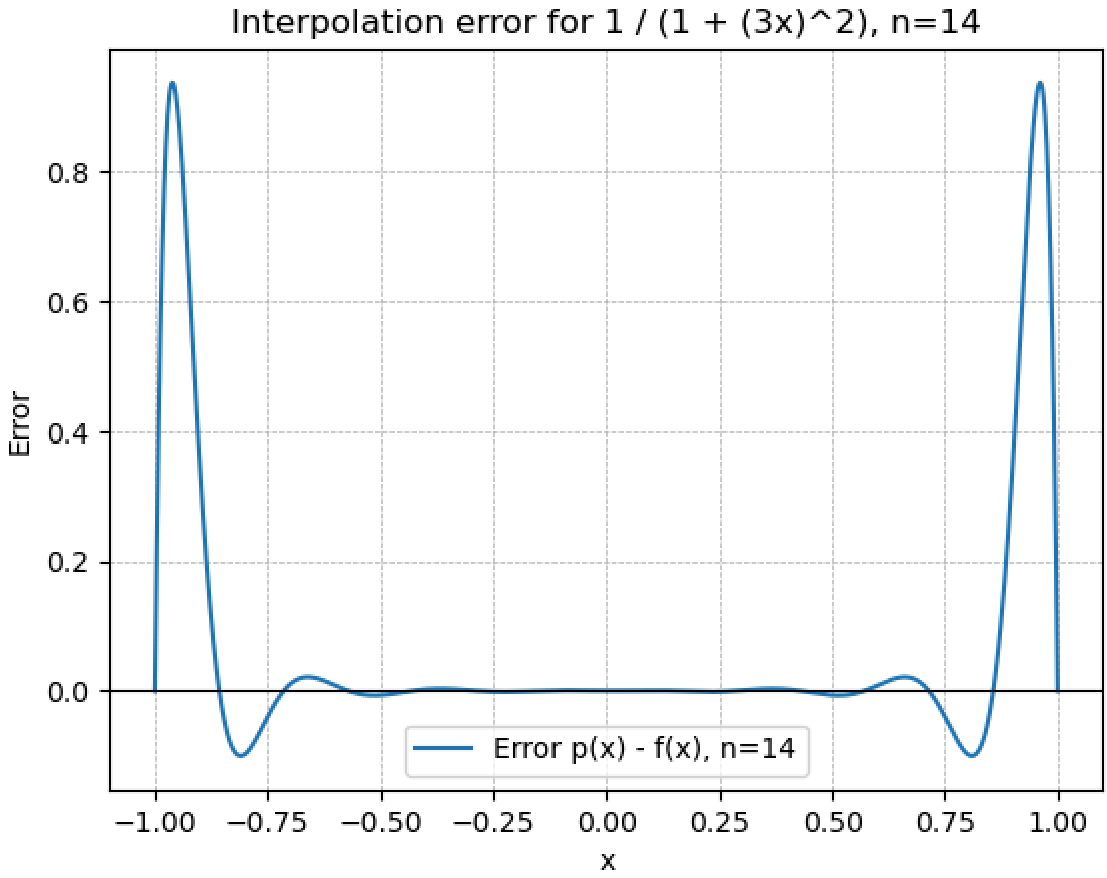


Interpolation of sin(pi x) with degree n=14
sin(pi x)
Interpolating polynomial p(x), n=14
Interpolation nodes
y
x



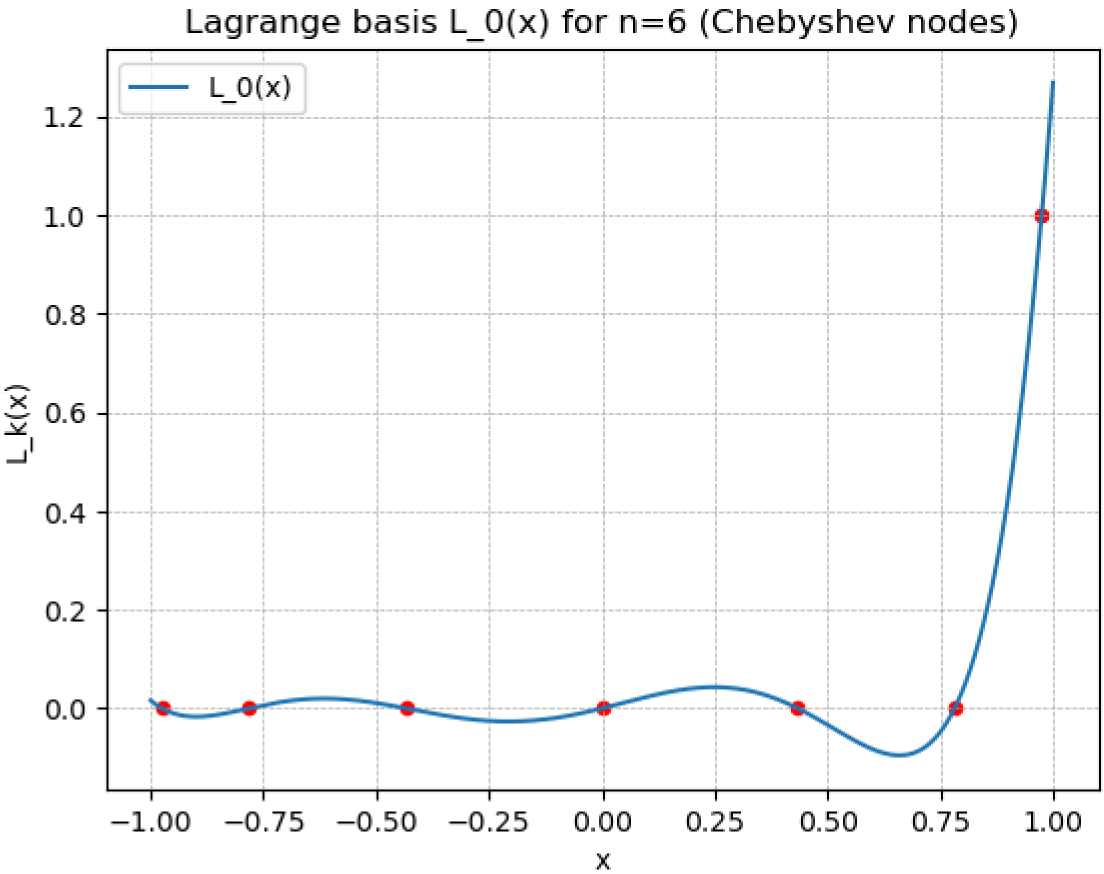
Interpolation error for sin(pi x), n=14
1e−8
Error p(x) - f(x), n=14
Error
x

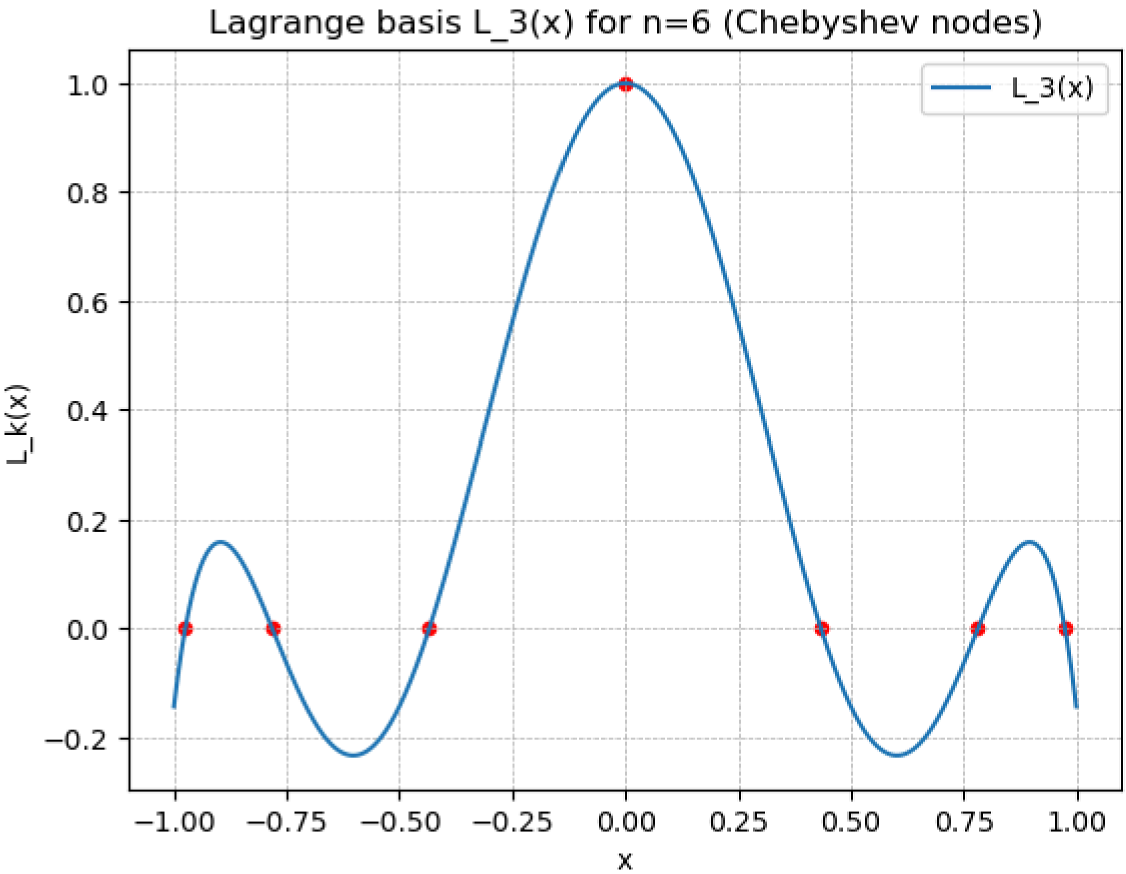


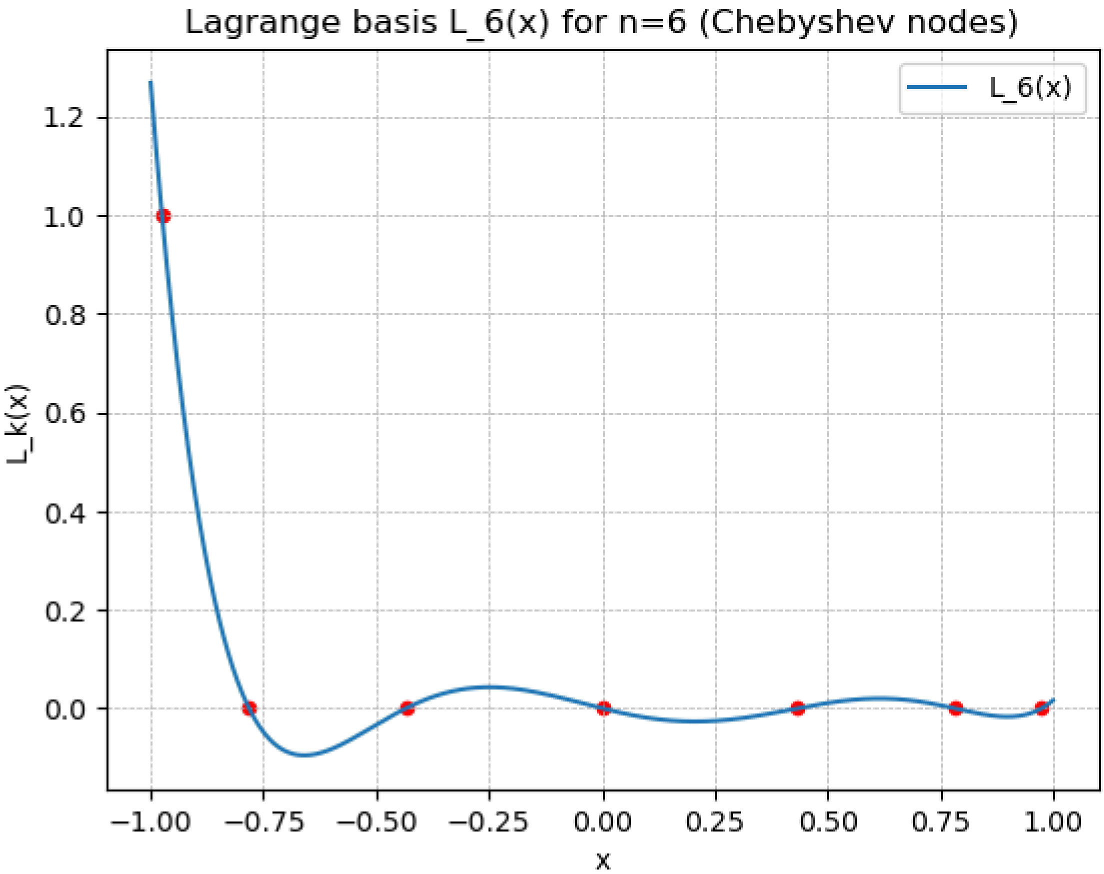


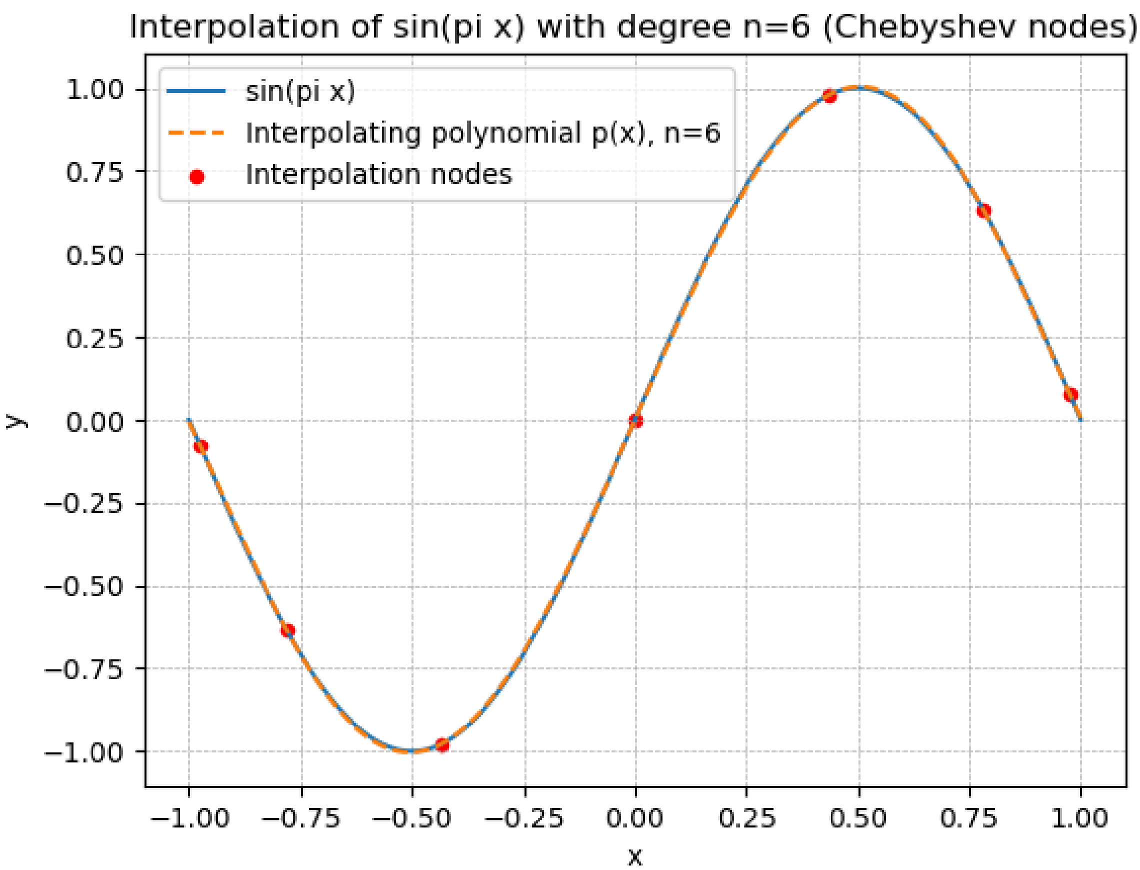
In [5]:
```
# === Second part: Chebyshev nodes (first kind) ===

def chebyshev_nodes_first_kind(n: int):
    """Chebyshev nodes (first kind, Gauss type): x_j = cos(pi*(j+1/2)/(n+
    j = np.arange(n + 1)
```

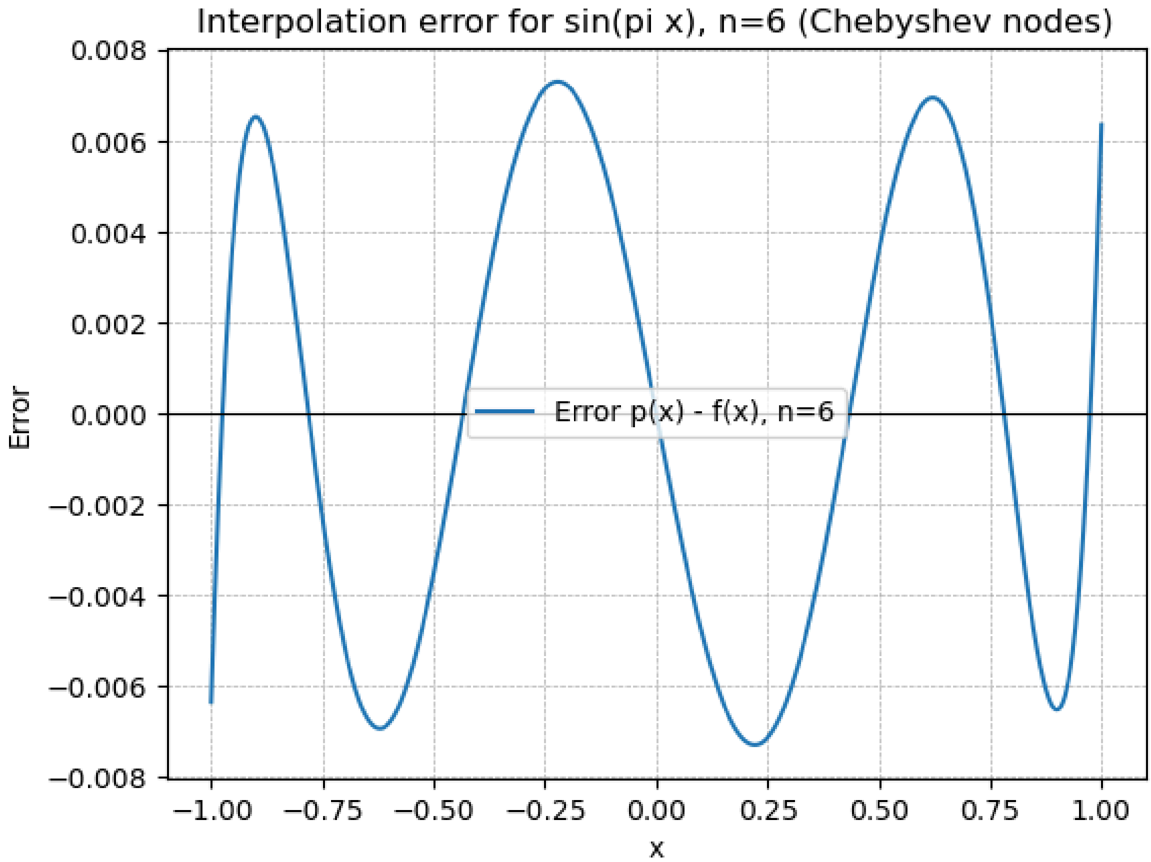
```python
    return np.cos(np.pi * (j + 0.5) / (n + 1))

# Setup
x_plot = np.linspace(-1, 1, 1000)
nodes_list = [6, 10, 14]

for n in nodes_list:
    nodes = chebyshev_nodes_first_kind(n)

    # (a) Plot a few L_k(x)
    ks = [0, n // 2, n]
    for k in ks:
        plt.figure()
        Lk = lagrange_basis(x_plot, nodes, k)
        plt.plot(x_plot, Lk, label=f"L_{k}(x)")
        plt.scatter(nodes, (nodes == nodes[k]).astype(int), s=20, color='
        plt.title(f"Lagrange basis L_{k}(x) for n={n} (Chebyshev nodes)")
        plt.xlabel("x")
        plt.ylabel("L_k(x)")
        plt.grid(True, linestyle="--", linewidth=0.5)
        plt.legend()
        plt.show()

    # (b) Interpolation and comparison
    for name, fun in [("sin(pi x)", f_fun), ("1 / (1 + (3x)^2)", g_fun)]:
        y_nodes = fun(nodes)
        p_vals = interpolate_lagrange(x_plot, nodes, y_nodes)
        f_vals = fun(x_plot)

        # Plot function vs interpolant
        plt.figure()
        plt.plot(x_plot, f_vals, label=f"{name}")
        plt.plot(x_plot, p_vals, "--", label=f"Interpolating polynomial p
        plt.scatter(nodes, y_nodes, color='red', s=20, label="Interpolati
        plt.title(f"Interpolation of {name} with degree n={n} (Chebyshev
        plt.xlabel("x")
        plt.ylabel("y")
        plt.grid(True, linestyle="--", linewidth=0.5)
        plt.legend()
        plt.show()

        # Error plot (optional but insightful)
        err = p_vals - f_vals
        plt.figure()
        plt.plot(x_plot, err, label=f"Error p(x) - f(x), n={n}")
        plt.axhline(0.0, color="black", linewidth=0.8)
        plt.title(f"Interpolation error for {name}, n={n} (Chebyshev node
        plt.xlabel("x")
        plt.ylabel("Error")
        plt.grid(True, linestyle="--", linewidth=0.5)
        plt.legend()
        plt.show()
```

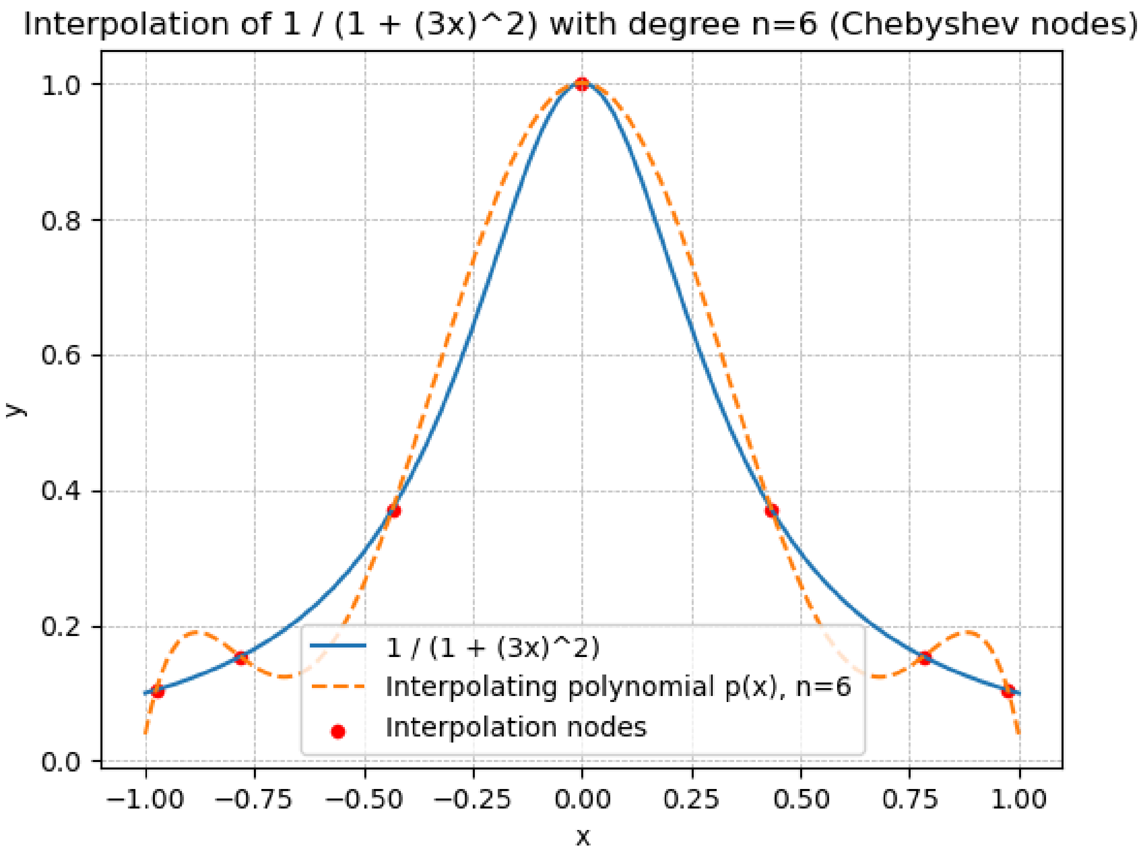

Lagrange basis L_0(x) for n=6 (Chebyshev nodes)
L_0(x)
L_k(x)
x
1.2
1.0
0.8
0.6
0.4
0.2
0.0
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

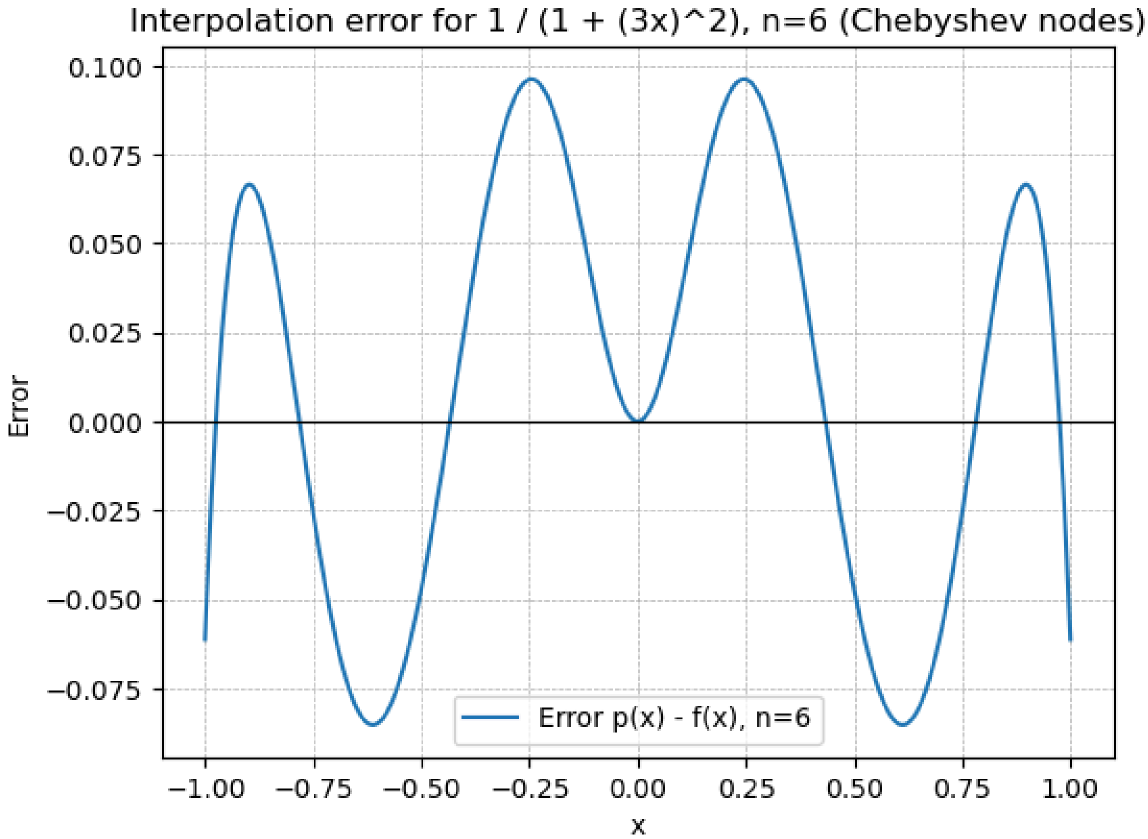

Lagrange basis L_3(x) for n=6 (Chebyshev nodes)
L_3(x)
L_k(x)
x
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

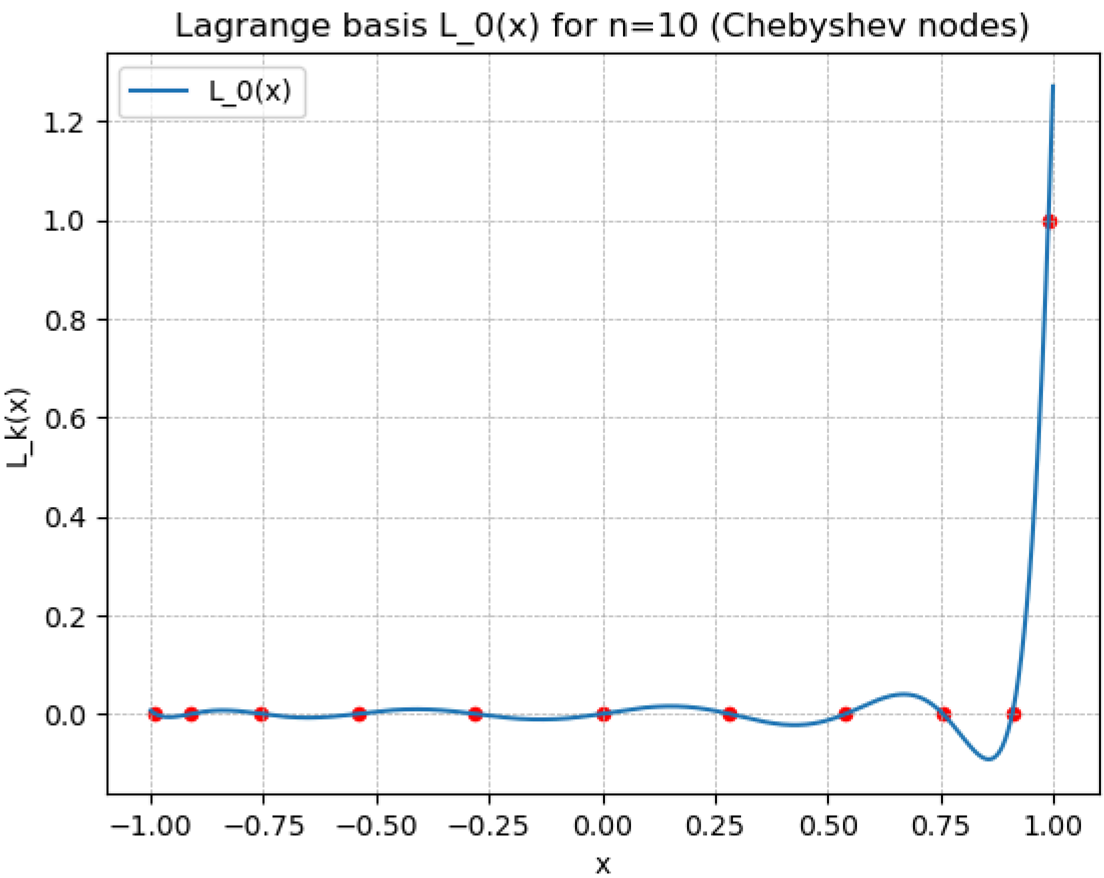

Lagrange basis L_6(x) for n=6 (Chebyshev nodes)
L_6(x)
L_k(x)
x
1.2
1.0
0.8
0.6
0.4
0.2
0.0
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

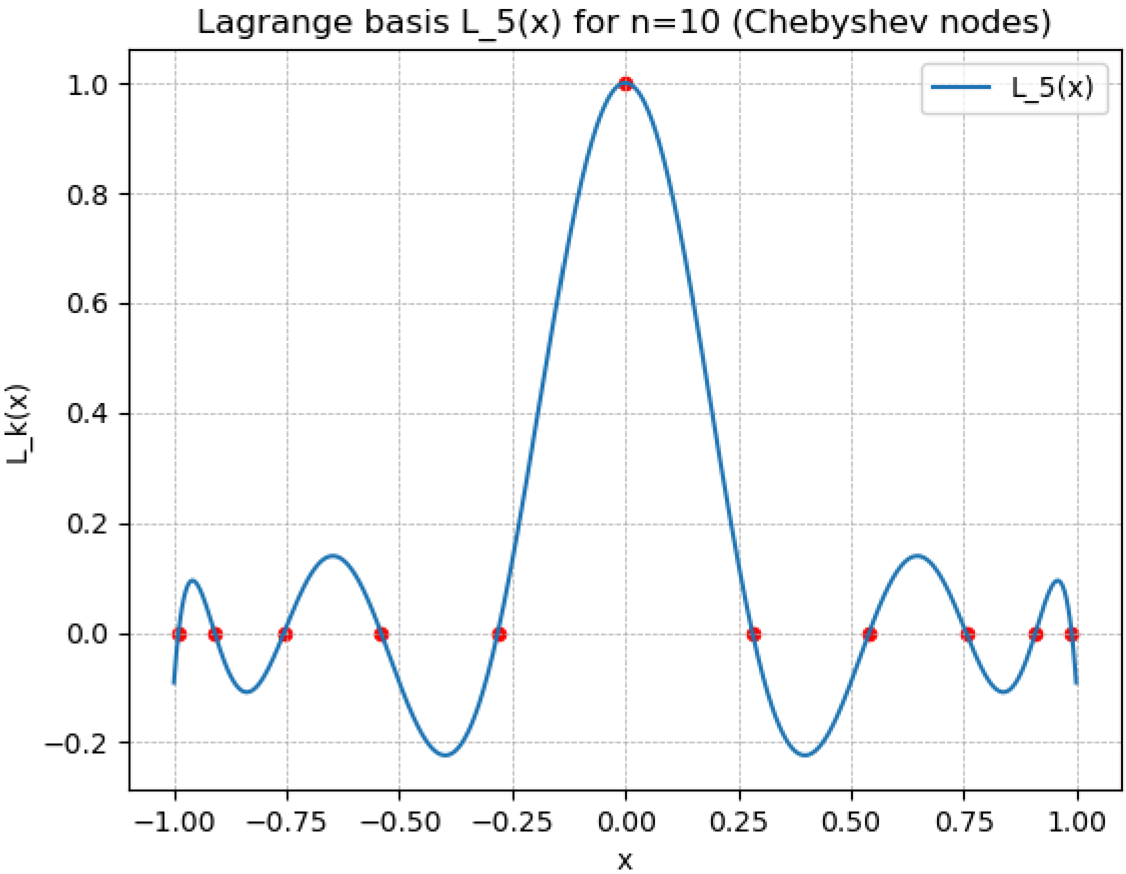

Interpolation of sin(pi x) with degree n=6 (Chebyshev nodes)
sin(pi x)
Interpolating polynomial p(x), n=6
Interpolation nodes
y
x
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

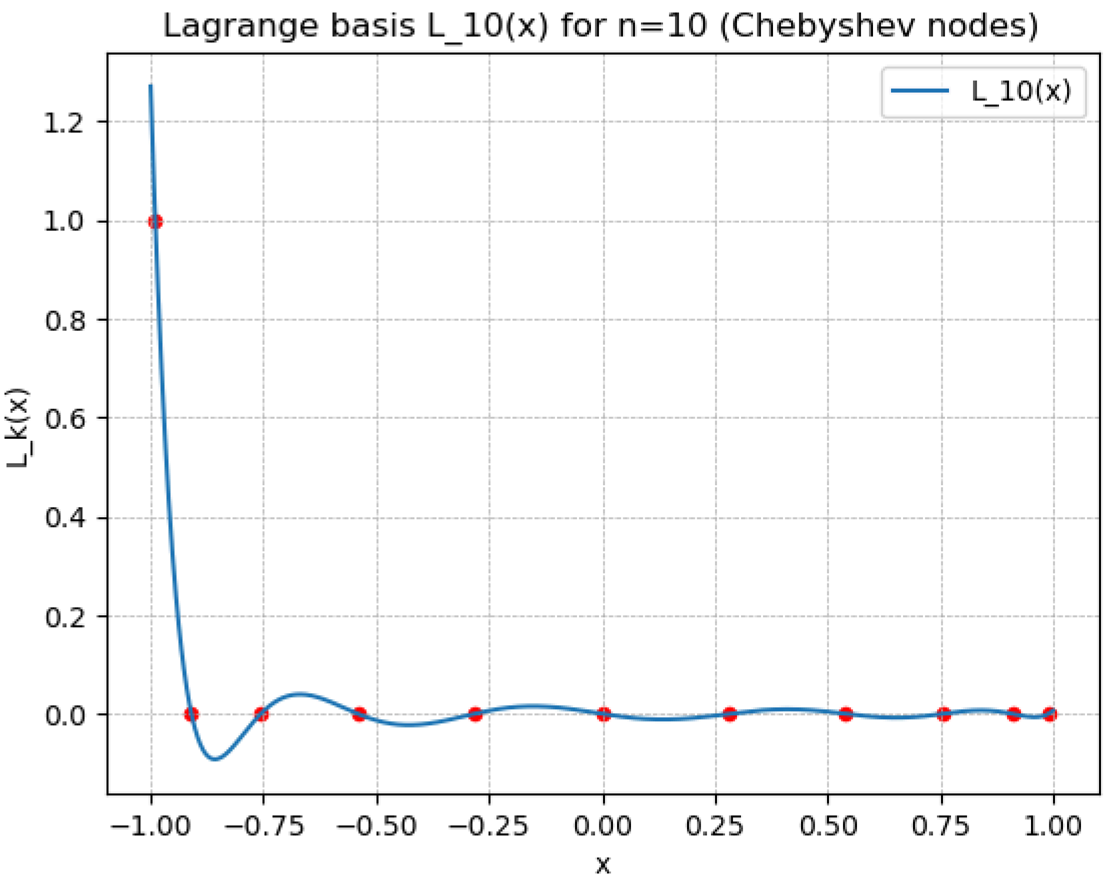

Interpolation error for sin(pi x), n=6 (Chebyshev nodes)
Error
x
Error p(x) - f(x), n=6
0.008
0.006
0.004
0.002
0.000
−0.002
−0.004
−0.006
−0.008
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

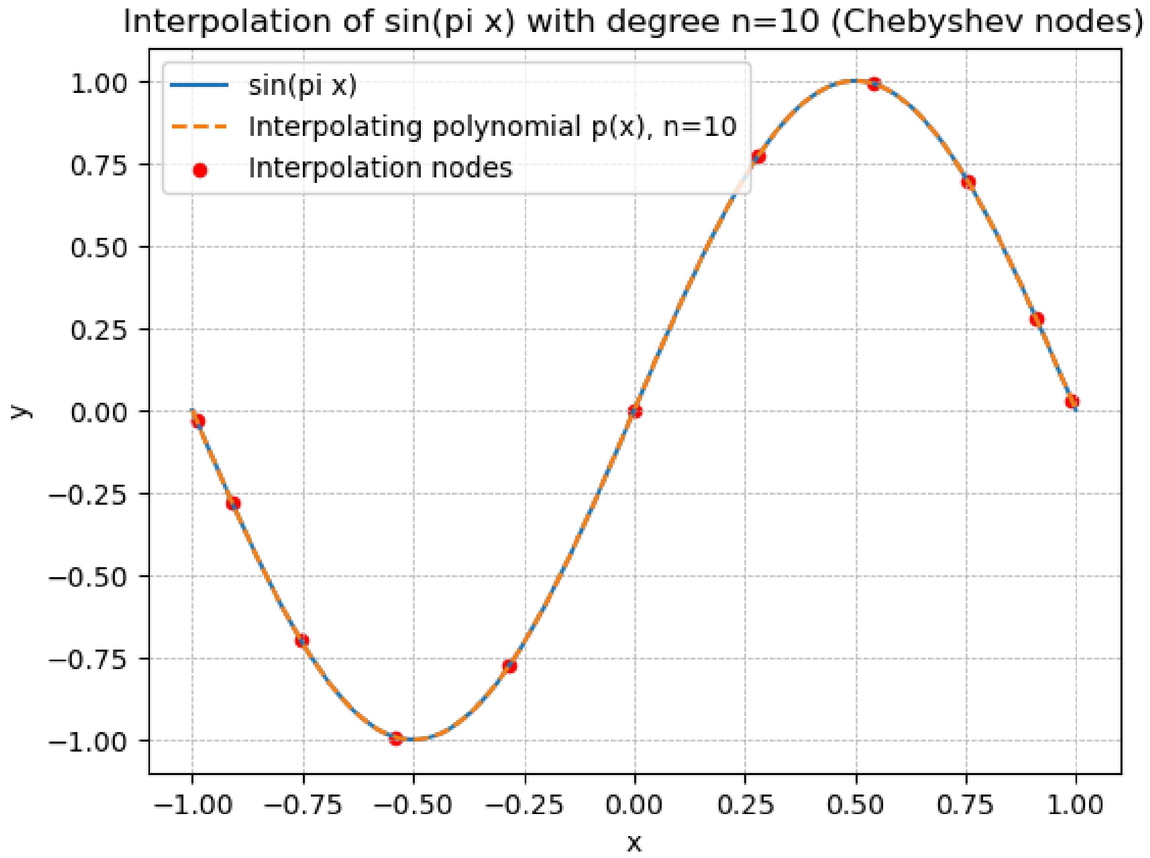

Interpolation of 1 / (1 + (3x)^2) with degree n=6 (Chebyshev nodes)
y
x
1 / (1 + (3x)^2)
Interpolating polynomial p(x), n=6
Interpolation nodes
1.0
0.8
0.6
0.4
0.2
0.0
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

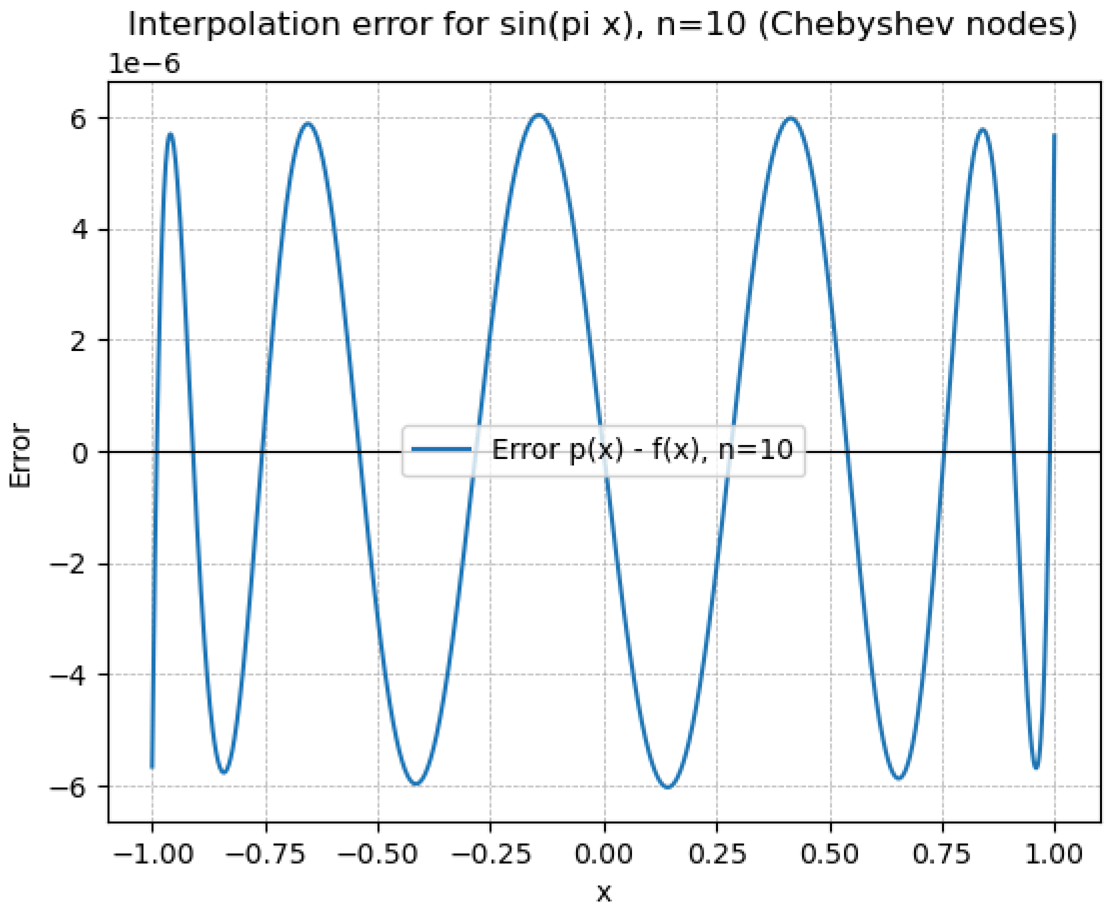

Interpolation error for 1 / (1 + (3x)^2), n=6 (Chebyshev nodes)
Error
x
Error p(x) - f(x), n=6

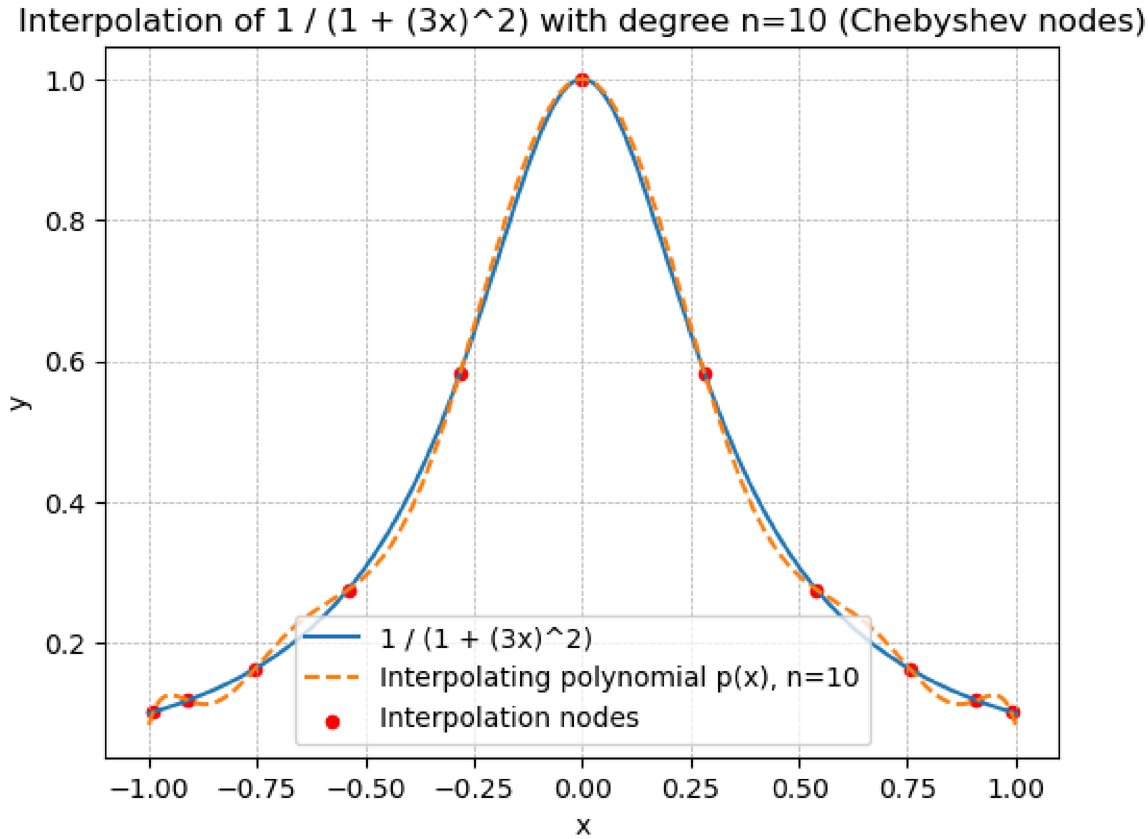

Lagrange basis L_0(x) for n=10 (Chebyshev nodes)
L_k(x)
x
L_0(x)

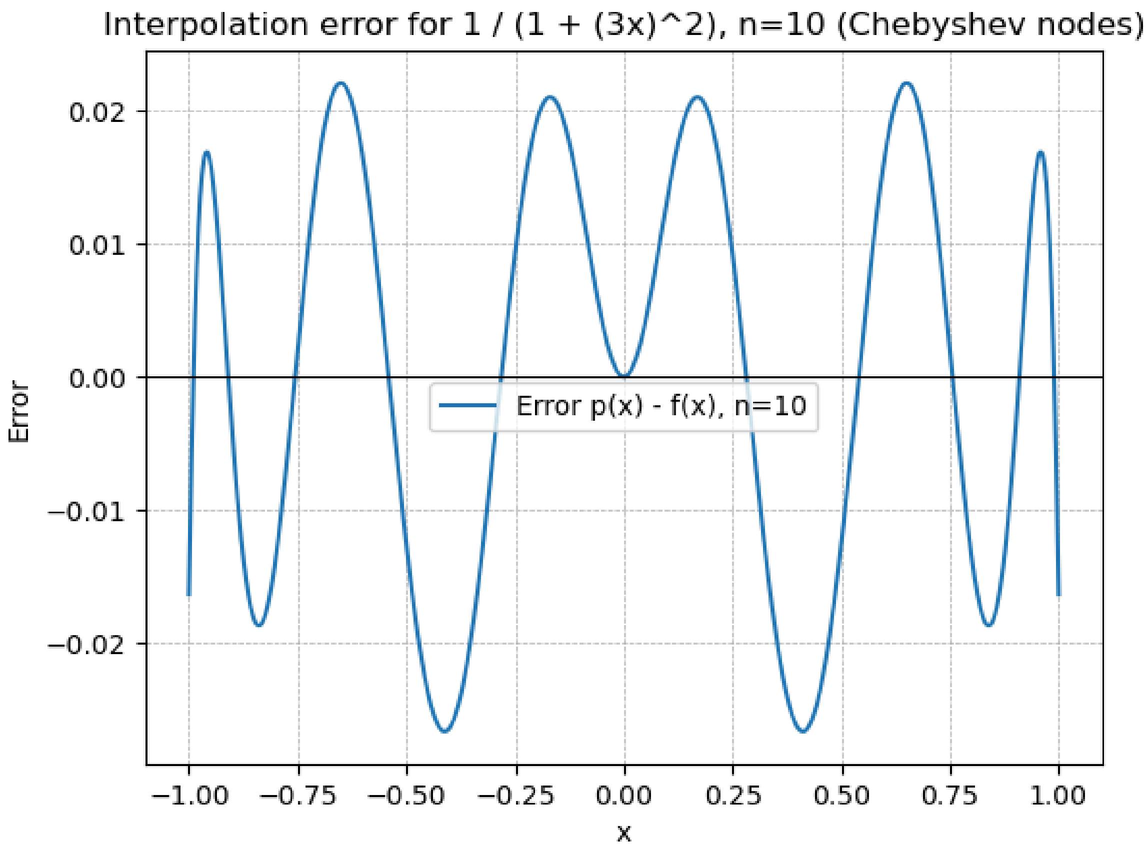

Lagrange basis L_5(x) for n=10 (Chebyshev nodes)
L_5(x)
L_k(x)
x
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

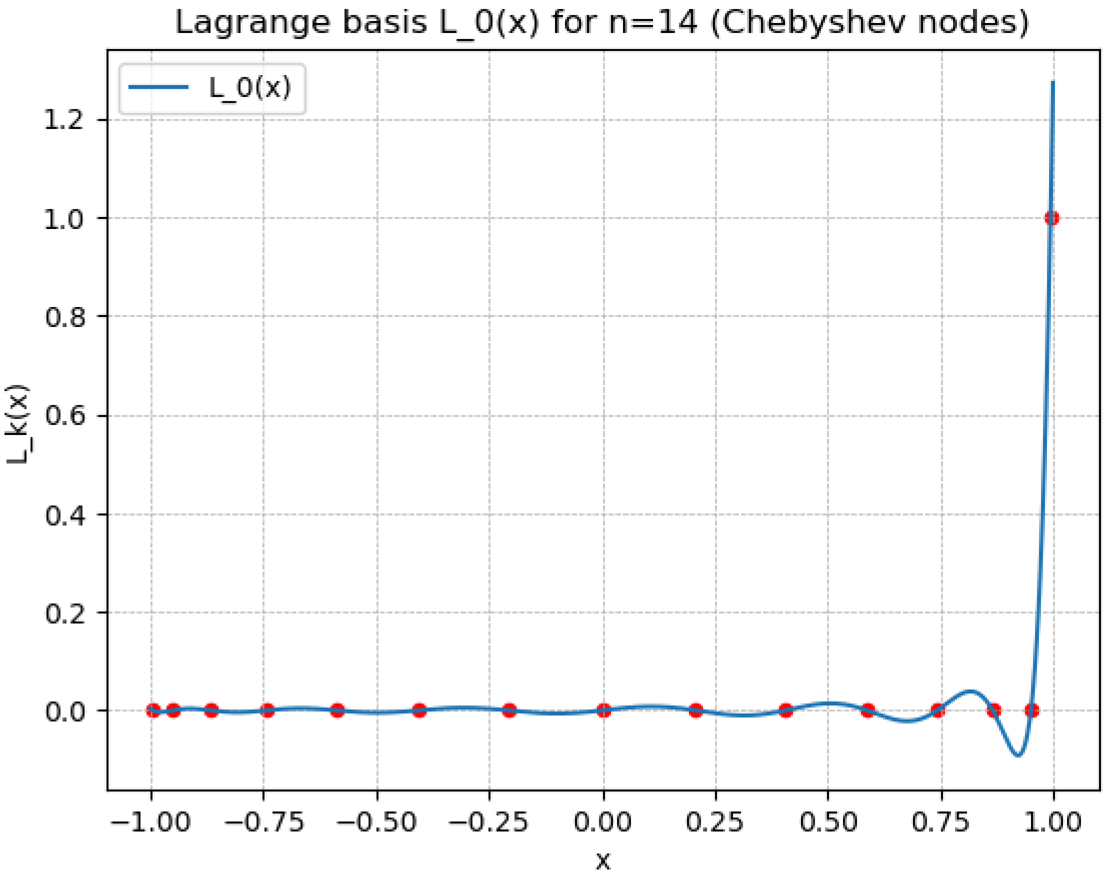

Lagrange basis L_10(x) for n=10 (Chebyshev nodes)
L_10(x)
L_k(x)
x
1.2
1.0
0.8
0.6
0.4
0.2
0.0
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00


Interpolation of sin(pi x) with degree n=10 (Chebyshev nodes)
sin(pi x)
Interpolating polynomial p(x), n=10
Interpolation nodes
y
x

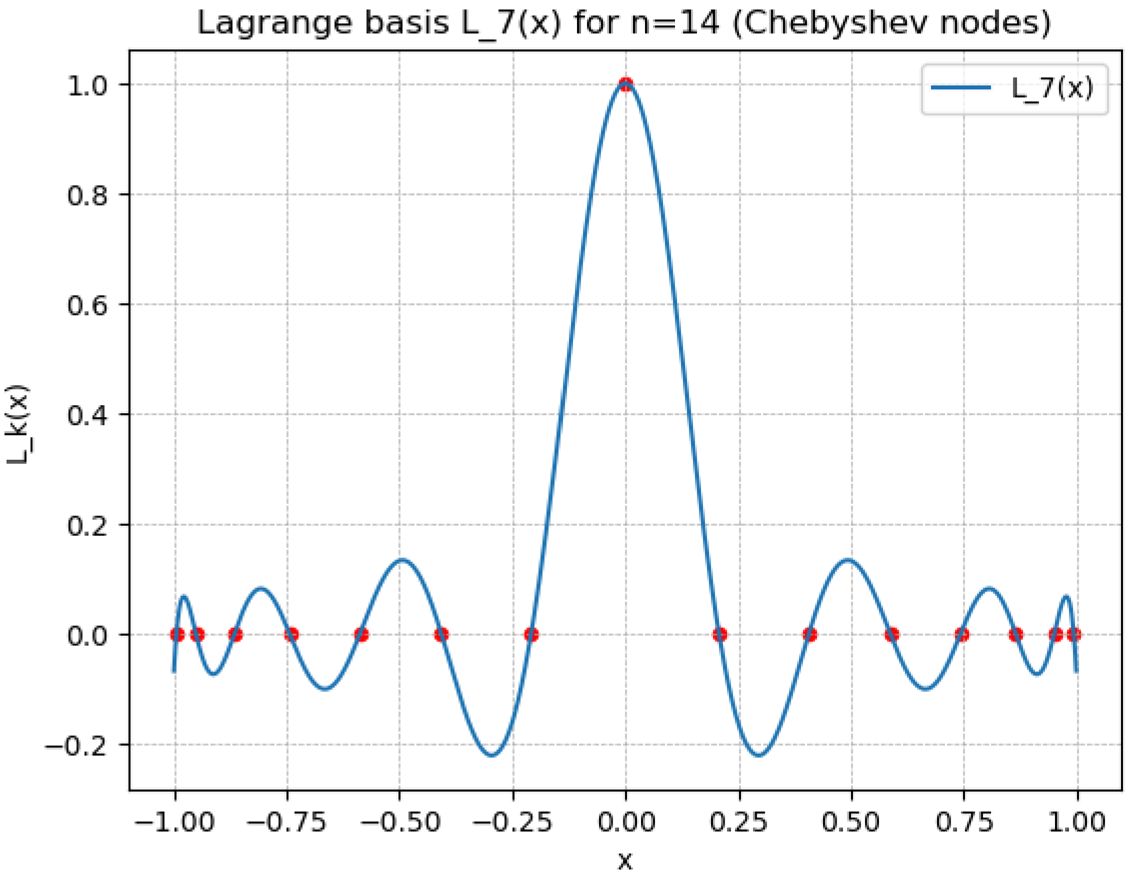

Interpolation error for sin(pi x), n=10 (Chebyshev nodes)
1e−6
Error p(x) - f(x), n=10
Error
x


Interpolation of 1 / (1 + (3x)^2) with degree n=10 (Chebyshev nodes)
1 / (1 + (3x)^2)
Interpolating polynomial p(x), n=10
Interpolation nodes
y
x
1.0
0.8
0.6
0.4
0.2
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

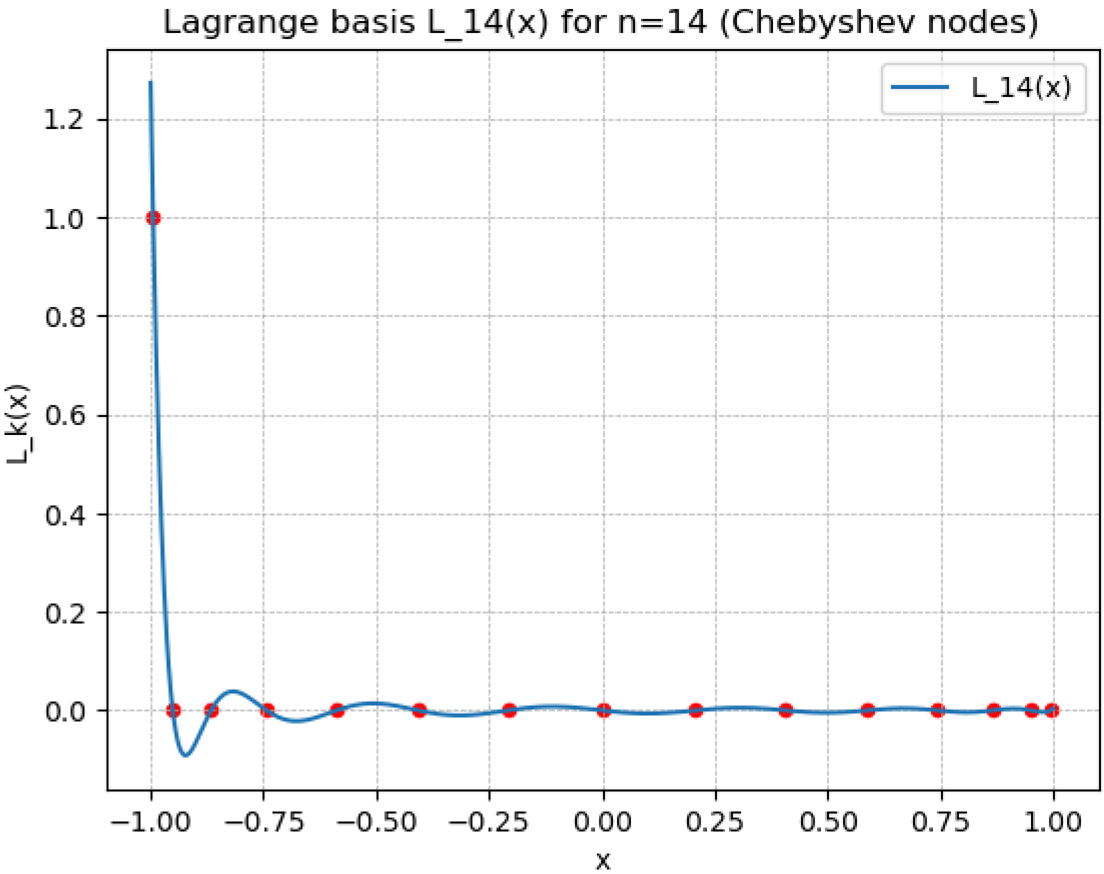

Interpolation error for 1 / (1 + (3x)^2), n=10 (Chebyshev nodes)
Error p(x) - f(x), n=10
Error
x
0.02
0.01
0.00
−0.01
−0.02
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

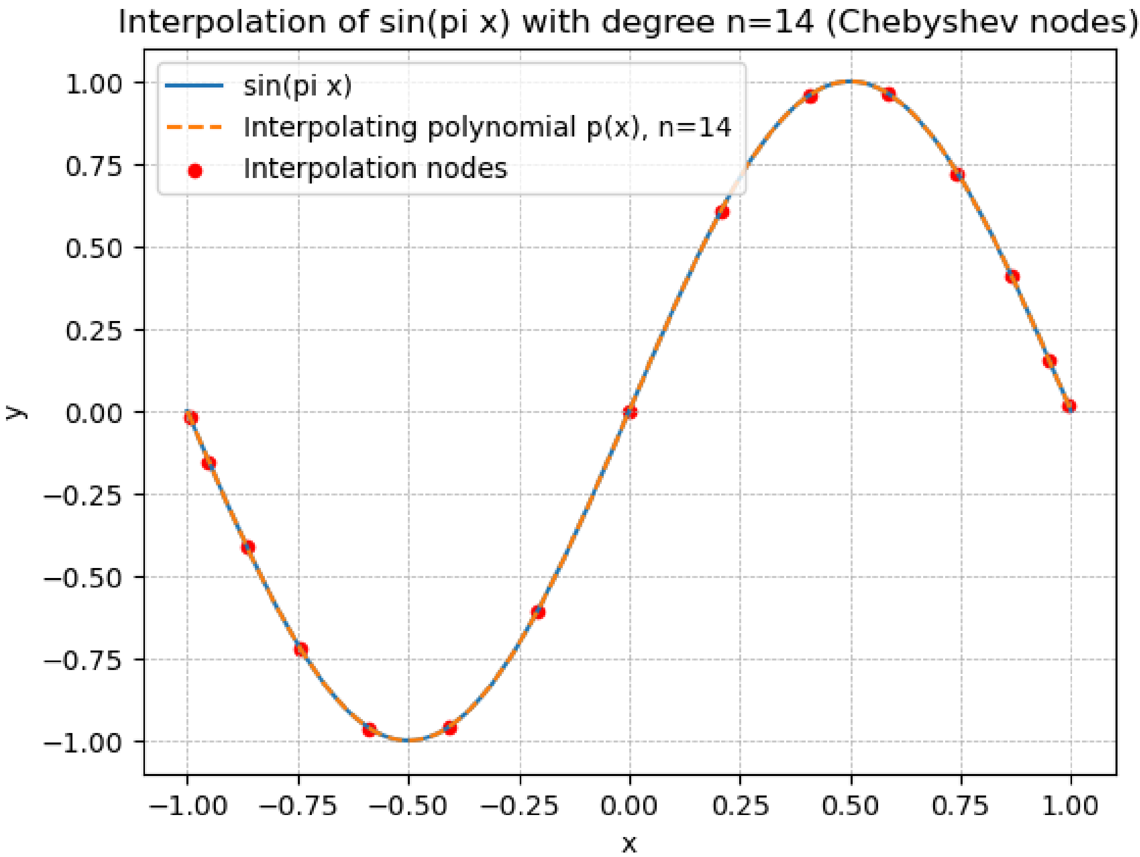


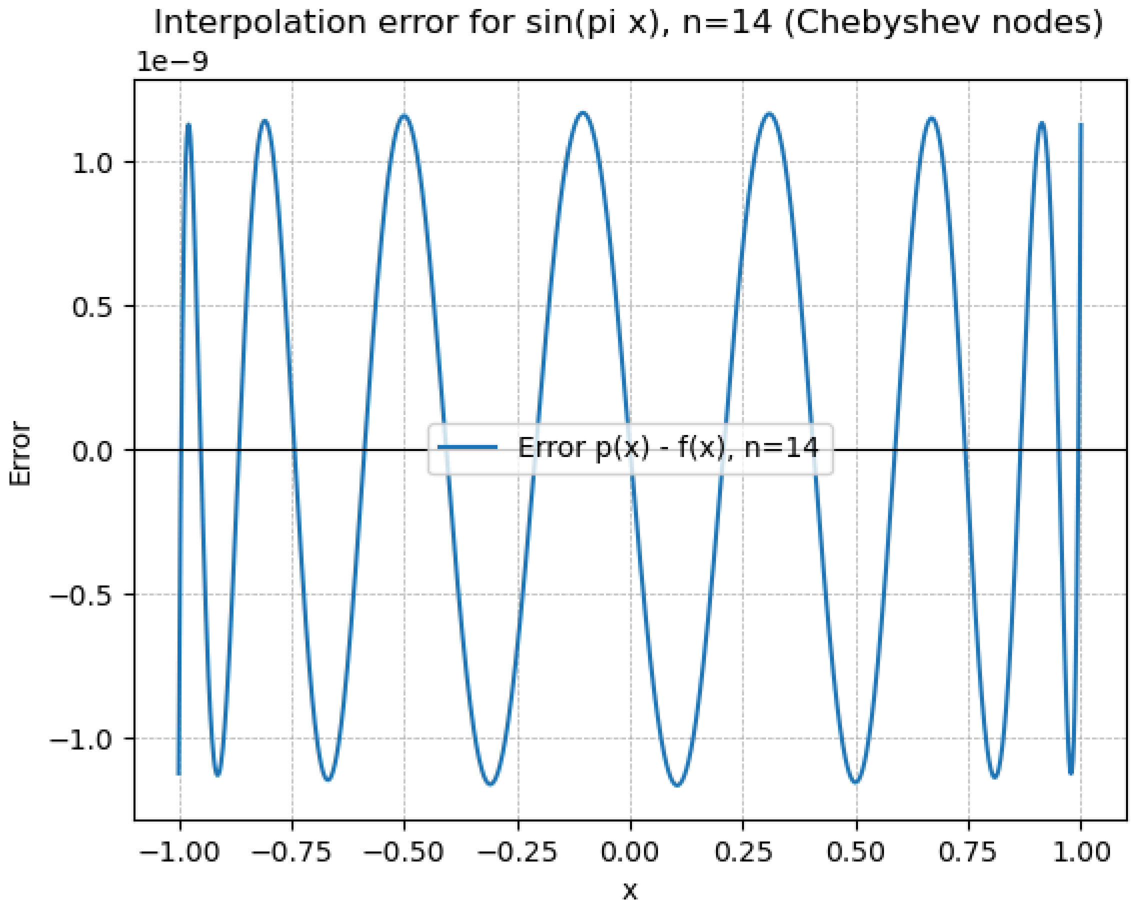
Lagrange basis L_0(x) for n=14 (Chebyshev nodes)
L_0(x)
L_k(x)
x



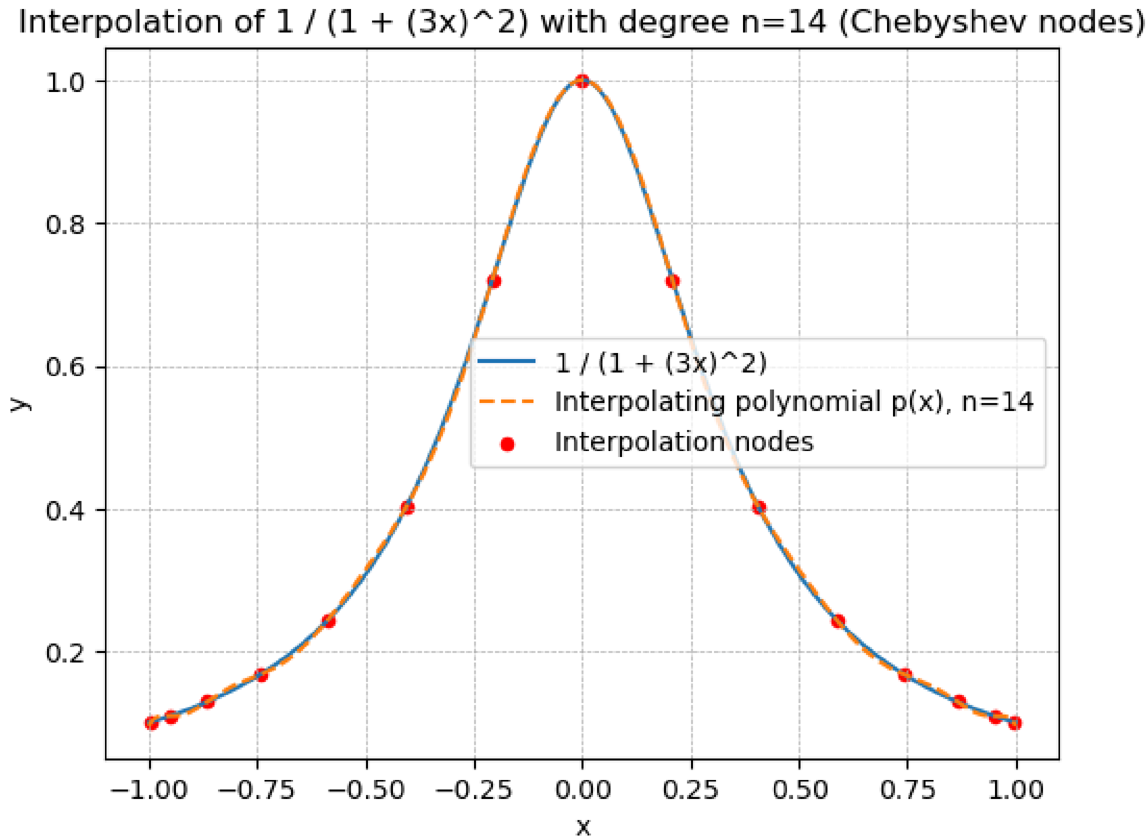
Lagrange basis L_7(x) for n=14 (Chebyshev nodes)
L_7(x)
L_k(x)
x


Lagrange basis L_14(x) for n=14 (Chebyshev nodes)
L_14(x)
L_k(x)
x
1.2
1.0
0.8
0.6
0.4
0.2
0.0
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

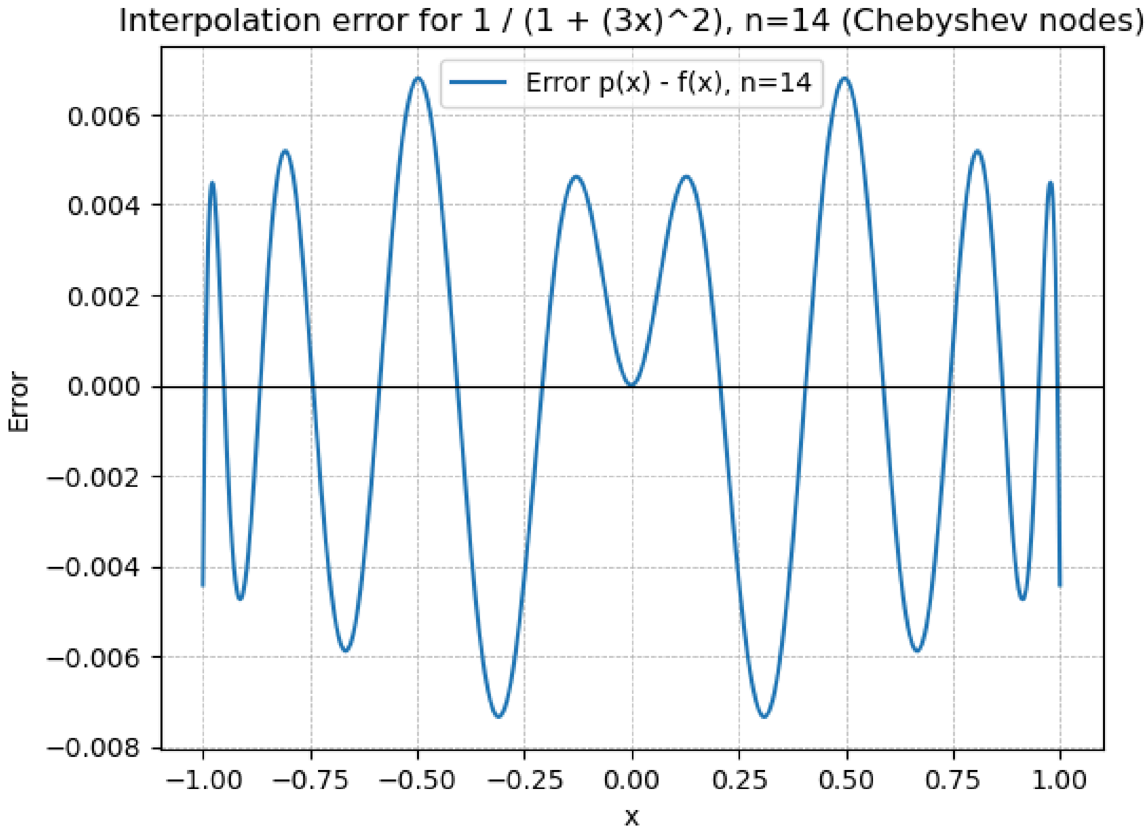

Interpolation of sin(pi x) with degree n=14 (Chebyshev nodes)
sin(pi x)
Interpolating polynomial p(x), n=14
Interpolation nodes
y
x
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00



Interpolation error for sin(pi x), n=14 (Chebyshev nodes)
1e−9
1.0
0.5
0.0
−0.5
−1.0
Error
Error p(x) - f(x), n=14
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
x



Interpolation of 1 / (1 + (3x)^2) with degree n=14 (Chebyshev nodes)
1.0
0.8
0.6
0.4
0.2
y
1 / (1 + (3x)^2)
Interpolating polynomial p(x), n=14
Interpolation nodes
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
x


Interpolation error for 1 / (1 + (3x)^2), n=14 (Chebyshev nodes)
Error p(x) - f(x), n=14
Error
x
0.006
0.004
0.002
0.000
−0.002
−0.004
−0.006
−0.008
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00