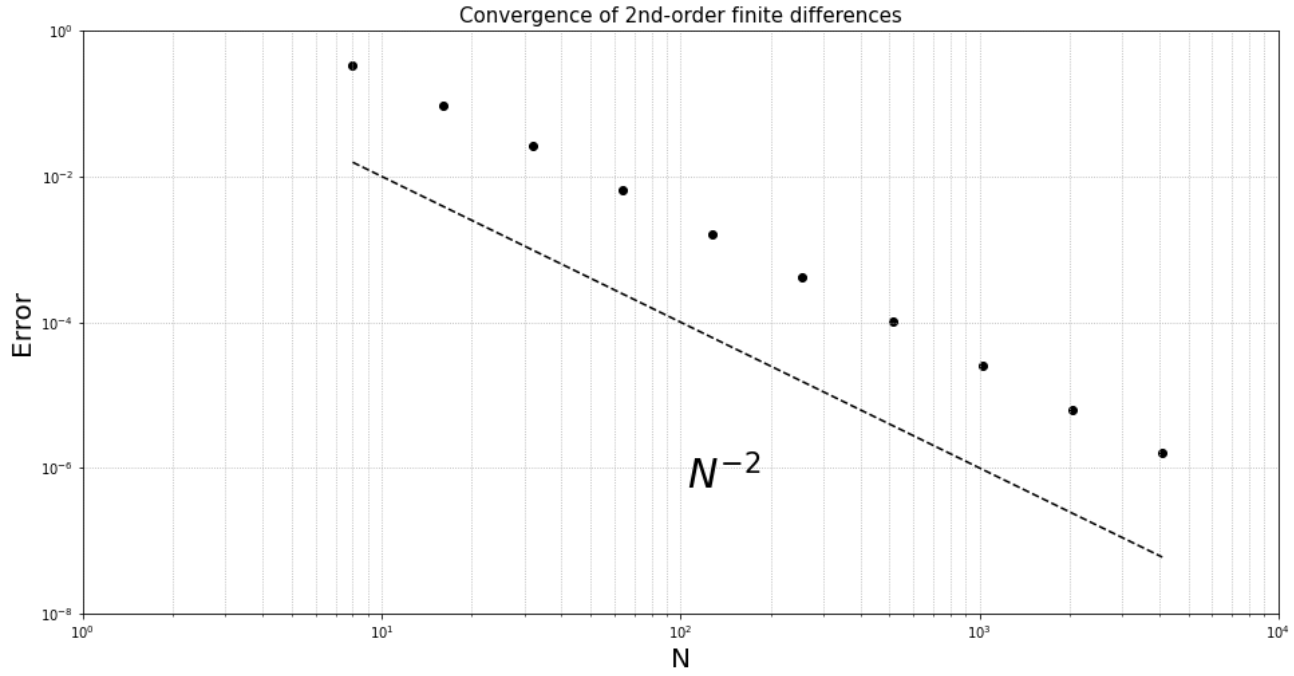
```
import numpy as np
from scipy.sparse import coo_matrix as coo
from matplotlib import pyplot as plt

N_vec = np.power(2, range(3,12 +1))
plt.figure(figsize=(16,8))

for N in N_vec:

  h = 2 * np.pi / N
  x = - np.pi + np.matrix(range(1,N + 1)).T * h
  u = np.power(np.e,np.sin(x))
  uprime = np.matrix(np.array(np.cos(x)) * np.array(u))

  e = np.ones(N)
  row = np.array(range(0,N))
  col = np.array(range(1, N))
  col = np.append(col,0)
  data = 1/2 * e
  D = coo((data, (row,col)), shape=(N,N)).toarray()

  matrix = D
  matrix = 1/h * (matrix - matrix.T)

  error_matrix = matrix * u - uprime
  error = error_matrix.sum(axis=1).max()

  plt.scatter(N,error,color='black')

plt.plot(N_vec,1/ N_vec ** 2,color='black',linestyle = '--')
plt.ylim(10**(-8), 10**0)
plt.xlim(10**0,10**4)
plt.xscale('log')
plt.yscale('log')
plt.grid(which='both',linestyle=':')
plt.text(105, 5*10**(-7), r'$N^{-2}$',fontsize=30)
plt.xlabel('N',fontsize=20)
plt.ylabel('Error',fontsize=20)
plt.title('Convergence of 2nd-order finite differences',fontsize = 15)

plt.show()
```


Convergence of 2nd-order finite differences
Error
N
N^{-2}
10^{0}
10^{-2}
10^{-4}
10^{-6}
10^{-8}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}