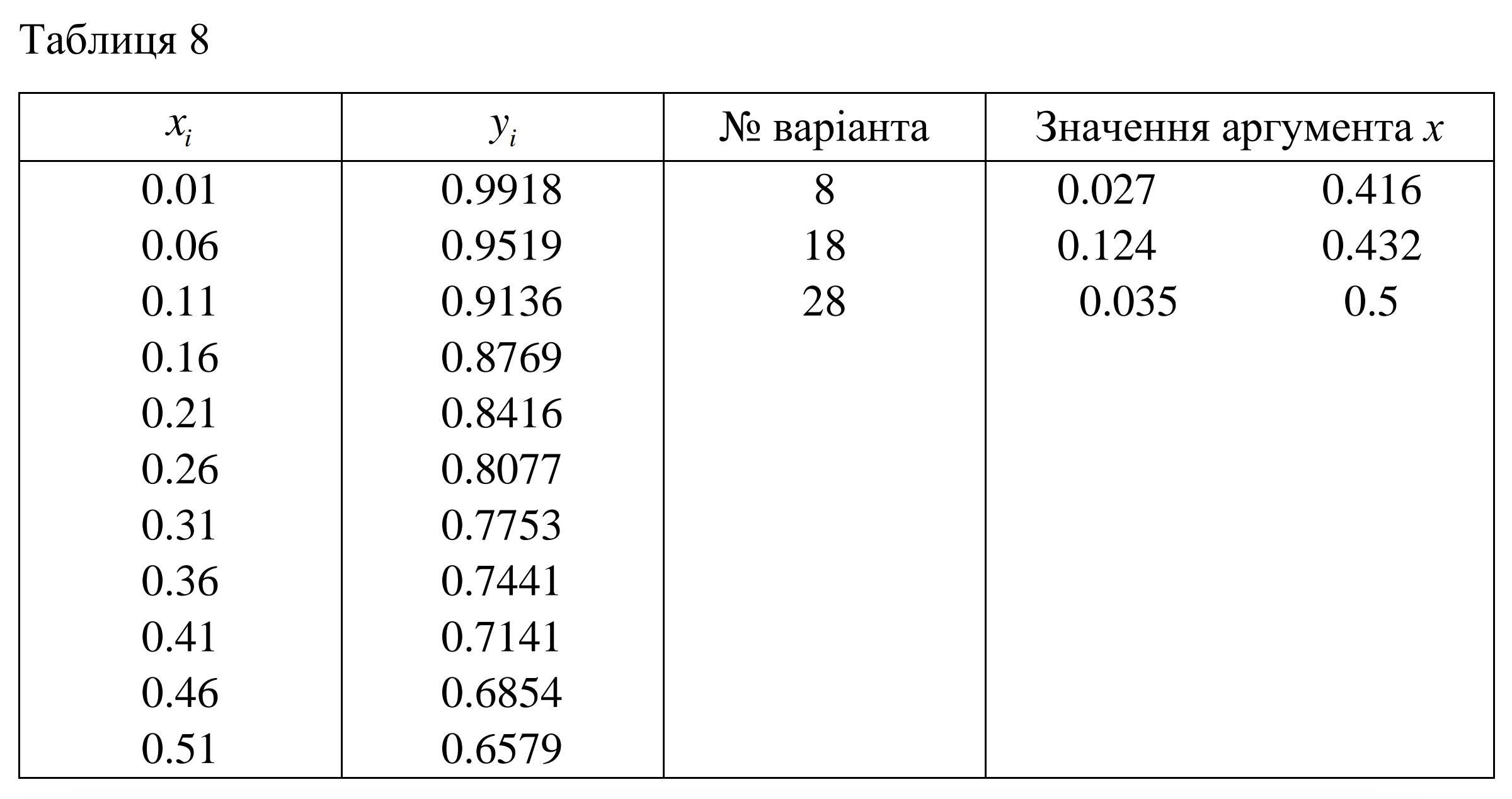
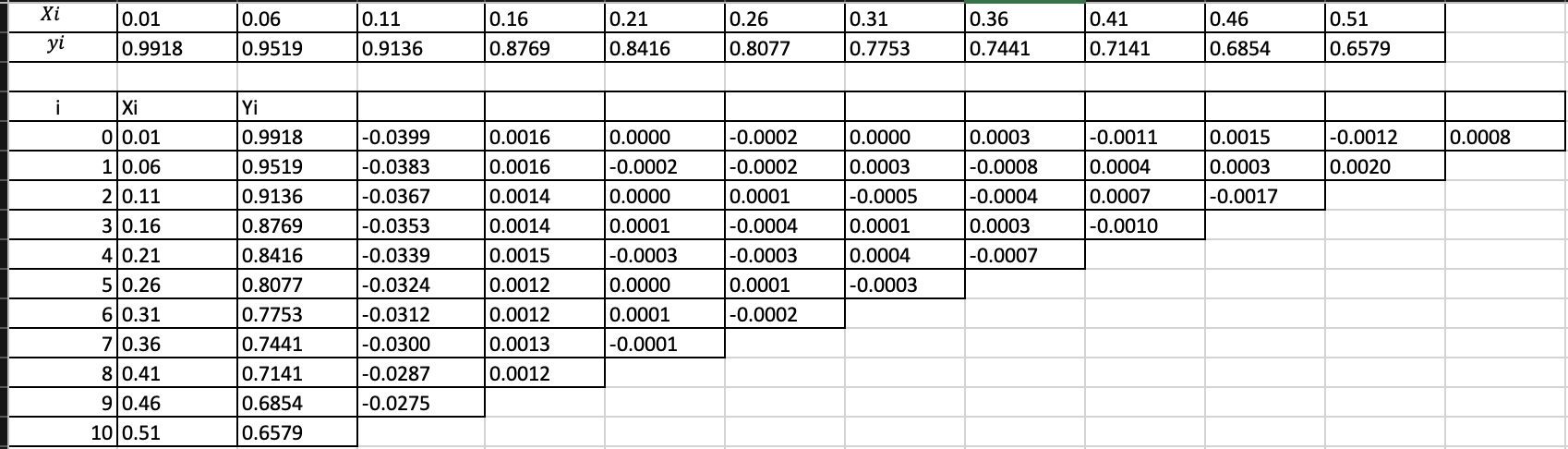
Іванов Кирил  
ФІТ 2-8





Код

import numpy as np

from math import factorial

import matplotlib.pyplot as plt

# задані точки

x = np.array ([0.1, 0.6, 0.11, 0.16, 0.21, 0.26, 0.31, 0.36, 0.41, 0.46, 0.51]) # задаємо x генератором списків

y = np.array([0.9918, 0.9519, 0.9136, 0.8769, 0.8416, 0.8077, 0.7753, 0.7441, 0.7141, 0.6854,0.6579])

# перша інтерполяційна формула

def first\_interpolation(x, y, x0):

n = len(x)

f = np.zeros((n, n))

f[:, 0] = y

for j in range(1, n):

for i in range(n - j):

f[i, j] = (f[i+1, j-1] - f[i, j-1]) / (x[i+j] - x[i])

ans = 0

for j in range(n):

prod = f[0, j]

for i in range(j):

prod \*= (x0 - x[i])

ans += prod

return ans

# друга інтерполяційна формула

def second\_interpolation(x, y, x0):

n = len(x)

f = np.zeros((n, n))

f[:, 0] = y

for j in range(1, n):

for i in range(n - j):

f[i, j] = (f[i+1, j-1] - f[i, j-1]) / (x[i+j] - x[i])

ans = f[0, 0]

for j in range(1, n):

prod = f[0, j]

for i in range(j):

prod \*= (x0 - x[i])

ans += prod

return ans

# обчислюємо значення функції в точках x = 0.1 та x = 0.9

x1 = 0.1

x2 = 0.9

y1 = first\_interpolation(x, y, x1)

y2 = second\_interpolation(x, y, x2)

print(f"f({x1}) = {y1}")

print(f"f({x2}) = {y2}")

xx = np.linspace(np.min(x), np.max(x), 100)

yy = np.zeros\_like(xx)

for i in range(len(xx)):

yy[i] = second\_interpolation(x, y, xx[i])

plt.plot(x, y, 'o', label='Дані точки')

plt.plot(xx, yy, label='багаточлен Н\*ютона')

plt.title("Графік інтерполяційної функції Ньютона")

plt.xlabel("x")

plt.ylabel("y")

plt.legend()

plt.show()

