

Hw4

Существует два типа интерполяции: Лагранжа и Ньютона. Но, на самом деле, это одно и то же (одинаковые полиномы). Однако, метод Лагранжа удобно применять, когда узлы интерполяции фиксированы и интерполируется не одна, а несколько функций. А метод Ньютона удобен при интерполяции одной и той же функции, в которой число узлов интерполяции меняется.

При интерполировании функции также появляются и погрешности-ошибки приближения функции интерполяционным полиномом. Для минимизации погрешности мы должны выбирать оптимальные узлы-нули многочлена Чебышева.

Интерполяция функции $1/(1-x)$

```
<def f(x):
    return 1/(1-x)

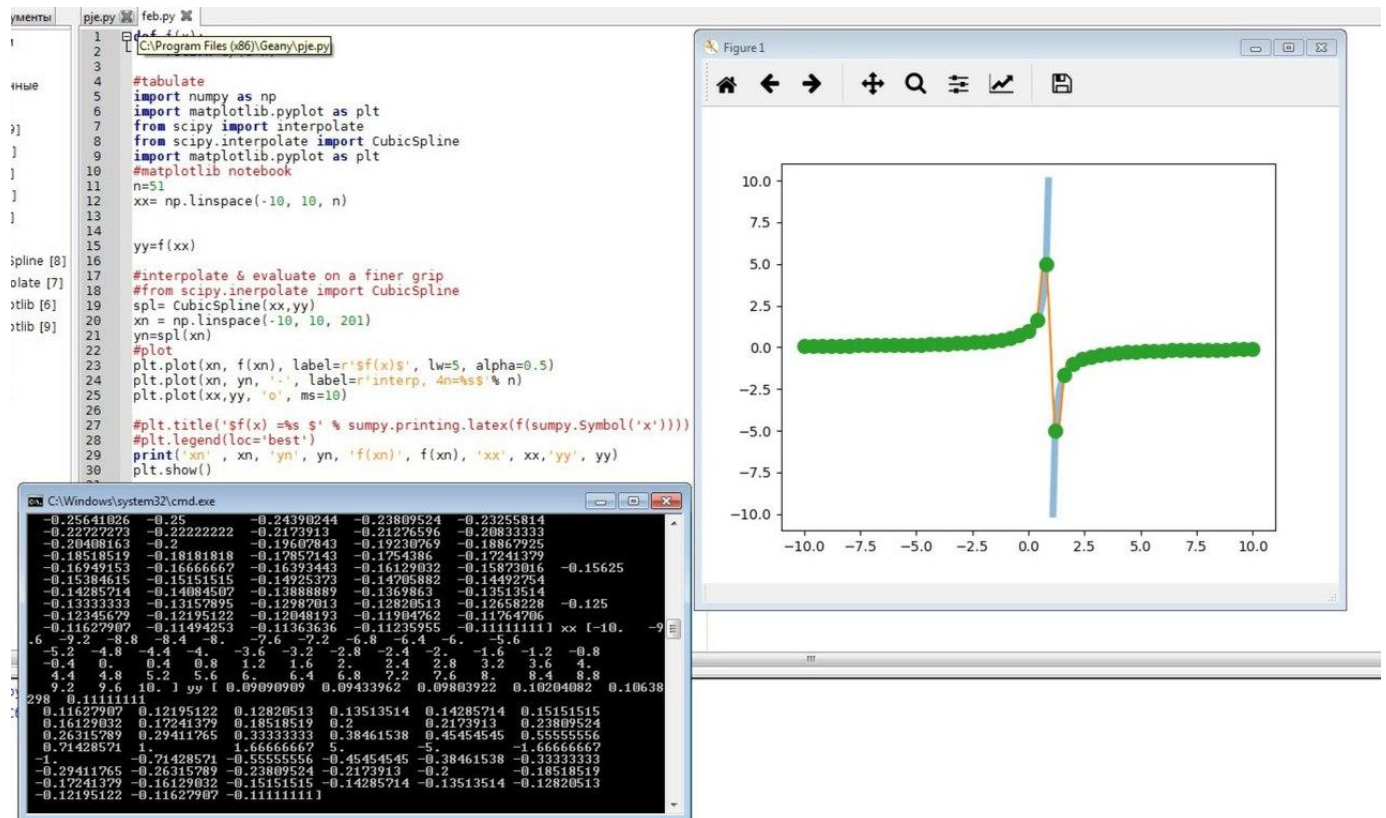
import numpy as np
import matplotlib.pyplot as plt
from scipy import interpolate
from scipy.interpolate import CubicSpline
import matplotlib.pyplot as plt

n=51
xx= np.linspace(-10, 10, n)
yy=f(xx)

spl= CubicSpline(xx,yy)
xn = np.linspace(-10, 10, 201)
yn=spl(xn)

plt.plot(xn, f(xn), label=r'$f(x)$', lw=5, alpha=0.5)
plt.plot(xn, yn, '-', label=r'interp, 4n=%s$'% n)
plt.plot(xx,yy, 'o', ms=10)

print('xn' , xn, 'yn', yn, 'f(xn)', f(xn), 'xx', xx, 'yy', yy)
plt.show()
```



Мы видим, что при интерполяции разрывной функции, подобранная функция сама неразрывна, поэтому график нужной нам функции получается неточный.