

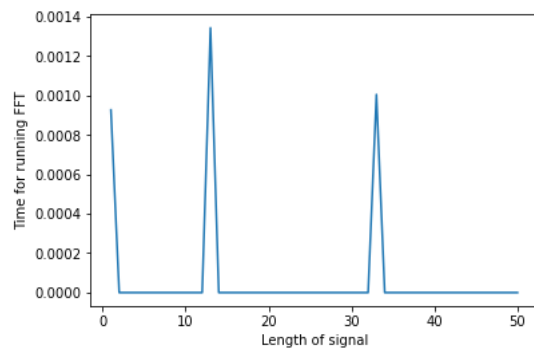
Afshar - Q3

این سوال در مورد عملکرد

fft and conv function

در قسمت اول زمان محاسبه افافتی بر حسب طول سیگنال ورودی به آن بدست آمده است که نمودار آن در خروجی چاپ شده

```
1 #part 1 and 2
2 import numpy as np
3 from scipy.fftpack import fft
4 import time
5
6 N_values = np.arange(1, 51)
7 times = np.zeros(len(N_values))
8
9 for i, N in enumerate(N_values):
10     x = np.random.rand(N)
11     start_time = time.time()
12     fft(x, N)
13     times[i] = time.time() - start_time
14
15 # Plot the results
16 import matplotlib.pyplot as plt
17 plt.figure()
18 plt.plot(N_values, times)
19 plt.xlabel('Length of signal')
20 plt.ylabel('Time for running FFT')
21 plt.show()
22
```



part3

به ازای طول سیگنال هایی که توانی از دو هستند عملکرد بهینه تری خواهد داشت

part4

در این قسمت تابع کانولوشن بررسی شده است. یکبار با استفاده از این تابع و بار دیگر با استفاده از توابع

fft and ifft

بررسی شده است و خروجی ها به صورت زیر میباشند

```
1 def conv_python(x, h):
2     N1 = len(x)
3     N2 = len(h)
4     N = N1 + N2 - 1
```

```

5     y = np.zeros(N)
6     for n in range(N):
7         for k in range(N1):
8             if (n-k) < N2 and (n-k) >= 0:
9                 y[n] += x[k]*h[n-k]
10    return y
11

```

```

1 import numpy as np
2 from scipy.fftpack import fft, ifft
3 import time
4
5 L_values = [50, 1000]
6
7 for L in L_values:
8     x = np.random.normal(0, 1, L)
9     h = np.random.uniform(0, 1, L)
10
11    # Using numpy.convolve function
12    start_time = time.time()
13    conv_numpy = conv_python(x, h)
14    time_numpy = time.time() - start_time
15
16    # Using FFT method
17    N = int(2*np.ceil(np.log2(L + L - 1)))
18    start_time = time.time()
19    conv_fft = ifft(fft(x, N) * fft(h, N))
20    time_fft = time.time() - start_time
21
22    print("For L = {}".format(L))
23    print("Time for numpy.convolve function: {:.6f} s".format(time_numpy))
24    print("Time for FFT method: {:.6f} s".format(time_fft))
25    print()
26

```

```

For L = 50:
Time for numpy.convolve function: 0.007992 s
Time for FFT method: 0.000000 s

For L = 1000:
Time for numpy.convolve function: 1.618336 s
Time for FFT method: 0.000998 s

```

