Ay190 – Worksheet 06
Chatarin (Mee) Wong-u-railertkun
Date: January 30, 2014

1 Discrete Fourier Transform

1.1 Compare Results from `dft` and `fft`

With the input as a vector x, `numpy.arange(10)`, table 1 shows result from two methods. We can see that our method yields the result very close to those from `numpy.fft`

Method	`dft(x)`	`fft(x)`
Index 0	45. +0.00000000e+00j	45. +0.j
Index 1	-5. +1.53884177e+01j	-5.+15.38841769j
Index 2	-5. +6.88190960e+00j	-5. +6.8819096j
Index 3	-5. +3.63271264e+00j	-5. +3.63271264j
Index 4	-5. +1.62459848e+00j	-5. +1.62459848j
Index 5	-5. +3.53452967e-14j	-5. +0.j
Index 6	-5. -1.62459848e+00j	-5. -1.62459848j
Index 7	-5. -3.63271264e+00j	-5. -3.63271264j
Index 8	-5. -6.88190960e+00j	-5. -6.8819096j
Index 9	-5. -1.53884177e+01j	-5.-15.38841769j

Table 1: Comparing results from two methods of Fourier transform with input of (0, 1, 2, ..., 9)

1.2 `dft` computational time

From figure 1, we can see that the computational time $\propto N^{1.5}$ instead of the expected N^2.

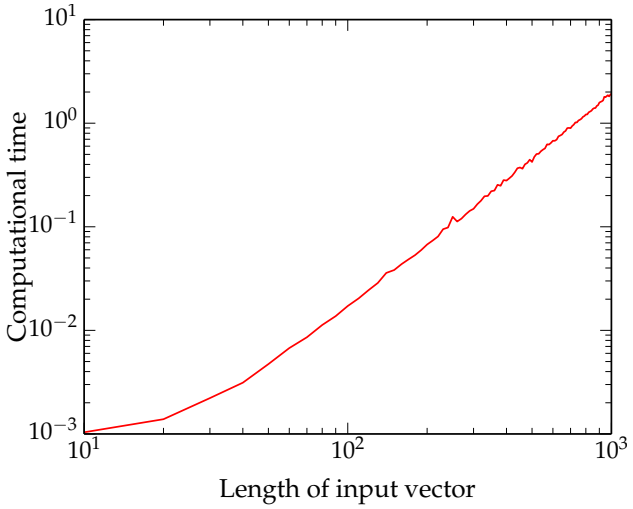

Figure 1: Plot the computational time of `dft` with respect to length of input vector.

1.3 Compare Computational Time

We can see from figure 2 that computational time for `fft(x)` increases much slower than that of `dft(x)`.

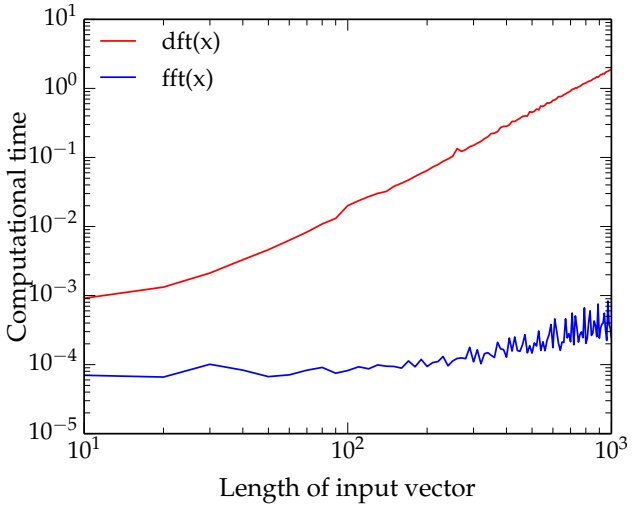

Figure 2: Comparing the increase of computation with length of input vector, for two different methods of Fourier transform.