On Fourier Transforms of Non-Uniform Data

Aarav Rastogi

Motivation

- The FFT computes the DFT in $N \log N$ time instead of N^2 time, but it needs uniform samples
- Real data is often non-uniform

How can we get a fast algorithm for non-uniform data?

The Paper and Today's Topic

Contents of the Paper

1. Introduction
2. Overview of the traditional FFT
3. Dutt and Rokhlin's Gridding Method
4. The Fast Multipole Method
5. Taylor Series Approximation
6. Applications
7. Appendix: Alternative FFT proof

Today's Content

1. Introduction
2. Overview of the traditional FFT
3. Dutt and Rokhlin's Gridding Method
4. Applications

Feel free to check out the paper if you're interested in the other methods.

The DFT and Naïve Computation

The DFT

$$X(j) = \sum_{k=0}^{N-1} x(k) W^{jk} \tag{1}$$

for $j = 0, \ldots, N-1$, and $W = e^{2i\pi/N}$

Naïve computation uses N multiplications and N summations which gives us a complexity of $O(N^2)$

Cooley and Tukey developed a faster, less expensive method to evaluate this in the 1960s called the Fast Fourier Transform (FFT): it relies on the fact that W is the principal nth root of unity.

Divide and Conquer

We write $N = r_1 r_2$, giving us

$$\begin{aligned} j &= j_1 r_1 + j_0, \qquad j_0 = 0, \ldots, r_1 - 1, \qquad j_1 = 0, \ldots, r_2 - 1 \\ k &= k_1 r_2 + k_0, \qquad k_0 = 0, \ldots, r_2 - 1, \qquad k_1 = 0, \ldots, r_1 - 1. \end{aligned} \tag{2}$$

We can then write the DFT equation in terms of j_0, j_1, k_0, and k_1:

$$X(j_0, j_1) = \sum_{k_0} \sum_{k_1} x(k_0, k_1) W^{jk_1 r_2} W^{jk_0}. \tag{3}$$

This still requires N^2 operations

The Inner Sum

Since $W^N = 1$,

$$W^{jk_1r_2} = W^{j_1r_1k_1r_2}\,W^{j_0k_1r_2} = \underbrace{W^{j_1k_1N}}_{=1}\,W^{j_0k_1r_1} = W^{j_0k_1r_1}. \tag{4}$$

The inner sum x below no longer depends on j_1

$$x_1(j_0, k_0) = \sum_{k_1} x(k_1, k_0) W^{j_0k_1r_1}, \tag{5}$$

which requires only Nr_1 computations (there are only r_1 values of k_1 and we have to sum over N).

The Outer Sum

Substituting in the inner sum, we get

$$X(j_1, j_0) = \sum_{k_0} x_1(j_0, k_0) W^{k_0(j_0 + j_1 r_1)}. \tag{6}$$

This only requires Nr_2 operations, giving us a total complexity of $N(r_1 + r_2)$.
Doing this infinitely, we get $N = r_1 r_2 \dots r_m$. If $r_i = 2$, we get the radix-2 formulation, which has a complexity $O(N \log N)$.

Questions

Any questions about the FFT before we move on?

Why does the FFT fail for nonuniform sampling?

The original DFT formula:

$$X(j) = \sum_{k=0}^{N-1} x(k) W^{jk} \tag{7}$$

Remember that

$$W^{jk_1 r_2} = W^{j_0 k_1 r_2} \tag{8}$$

because $W^{Nk} = 1$. This only applies if k is an integer, which isn't true for nonuniform samples.

Therefore, we can't apply the FFT algorithm.

Re-framing the Problem

Our original problem can be reframed into:

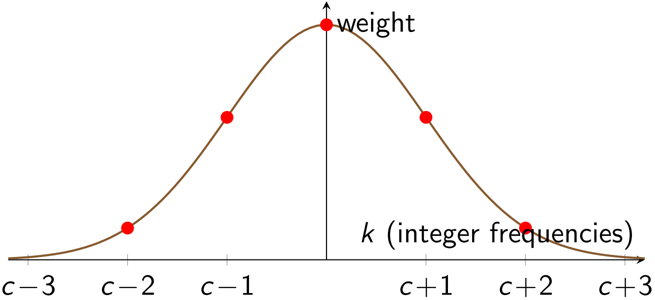
1. We take e^{icx} for some $c \notin \mathbb{Z}$
2. Unlike the DFT, $e^{ic(x+2\pi)} \neq e^{icx}$

Dutt and Rokhlin (1993) invented an algorithm to solve this by approximating these non-uniform points.

Changing to a Gaussian

Instead of taking e^{icx} as a point, they take it as a distribution:

$$\phi(x) = e^{-bx^2} e^{icx} \tag{9}$$



Note- we actually take $\lfloor c \rfloor$ for the distribution

How does the Gaussian help?

Using the Gaussian, we can estimate how much of each integer frequency is 'in' the actual value.

Estimation of $\phi(x)$

We do this using

$$\sum_{k=-\infty}^{\infty} \rho_k e^{ikx}, \tag{10}$$

where ρ_k is the weight for each k,

$$\rho_k = \frac{1}{2\sqrt{b\pi}} e^{-(c-k)^2/4b} \tag{11}$$

Error

One of the results that proved by Dutt and Rokhlin is that:

$$\left| \phi(x) - \sum_{k=-\infty}^{\infty} \rho_k e^{-ikx} \right| < e^{-b\pi^2} \left(4b + \frac{70}{9} \right), \tag{12}$$

where the right hand side represents the maximum possible error.
In practice, because we can't take this sum, we limit it using a $q \in \mathbb{Z}$

$$\left| \phi(x) - \sum_{k=\lfloor c \rfloor - q/2}^{\lfloor c \rfloor + q/2} \rho_k e^{ikx} \right| < e^{-b\pi^2}(4b + 9), \tag{13}$$

but this gives us a larger error.

Application: The Riemann Zeta Function

$$\zeta(s) = \sum_{n=1}^{\infty} \frac{1}{n^s}$$

- Verifying the Riemann Hypothesis up to height T (where $0 \leq t \leq T$) requires evaluating $\zeta(\frac{1}{2} + it)$ at enormous numbers of points t.
- The Riemann–Siegel formula reduces each evaluation to a finite sum of $\sim \sqrt{t}$ terms — but doing this independently at N points costs $O(N\sqrt{T})$.
- In 1988, Odzylko and Schönhage came up with an algorithm that evaluates this at many uniform t that is much quicker and used it to verify the Riemann Hypothesis till 10^{20}
- When finding zeroes, a uniform grid can often be slower than using a non-uniform sampling method

Other Application Areas

- Signal processing with unequal spectral resolution (Oppenheim et al., 1971).
- Electrochemistry: spectrally accurate Ewald summation for particle simulations (Lindbo & Tornberg, 2011).
- Observational astronomy: irregular sampling of interferometric and photometric data (Thompson & Bracewell, 1974; Meisel, 1978).

Thank you!

Feel free to ask me any questions or email me at
aaravrastogi10@gmail.com