P1. Fourier Transform / Fast FT. Ryan.

Fourier transform. f integrable function

$$\hat{f}(\xi) = \int_{-\infty}^{+\infty} e^{-2\pi i \xi x} f(x)\, dx.$$

spatial domain (time.) $\underset{\text{IFT}}{\overset{\text{FT}}{\rightleftarrows}}$ frequency domain.

IFT.

$$f(x) = \int_{-\infty}^{+\infty} e^{2\pi i \xi x} \hat{f}(\xi)\, d\xi.$$

Each value of $\hat{f}(\xi)$ corresponds to a wave $\underbrace{\hat{f}(\xi)}_{\text{amplitude}} \cdot e^{2\pi i \xi x}$.

frequency.

P2.

FFT.

DFT. (Assume the func is compact support).

Given $x_0, x_1, \cdots, x_{n-1}$. $\qquad n = 2^p$.

DFT: $$\hat{x}_j = \sum_{k=0}^{n-1} e^{-2\pi i \frac{jk}{n}} x_k .$$

In matrix form.

$$\hat{X} = F_n X.$$

$$\begin{pmatrix} \hat{x}_0 \\ \vdots \\ \\ \hat{x}_{N-1} \end{pmatrix} = \begin{bmatrix} \\ e^{-2\pi i jk/N} \\ \shortparallel \\ \omega_N^{jk} \end{bmatrix} \begin{bmatrix} x_0 \\ \vdots \\ \\ x_{N-1} \end{bmatrix}.$$

Reorganize x into $\begin{bmatrix} X_e \\ X_o \end{bmatrix}$.

$$X_e = \begin{bmatrix} x_0 \\ x_2 \\ \vdots \end{bmatrix} \qquad X_o = \begin{bmatrix} x_1 \\ x_3 \\ \vdots \end{bmatrix}.$$

even $\quad$ odd

$$F_n = \begin{bmatrix} F_{11} & F_{12} \\ F_{21} & F_{22} \end{bmatrix}.$$

(P3)

△ F_{11} $\quad 0 \le j', k' < \frac{N}{2}$

$$(F_{11})_{j',k'} = e^{-\frac{2\pi i}{N}(j' \cdot 2k')} = e^{-2\pi i \frac{j'k'}{\left(\frac{N}{2}\right)}}$$

So $\quad F_{11} = F_{N/2}$.

△ F_{21}. $\quad 0 \le j', k' < \frac{N}{2}$

$$(F_{21})_{j'+\frac{N}{2}, k'} = e^{-\frac{2\pi i}{N}\left(j'+\frac{N}{2}\right) \cdot 2k'}$$

$$= e^{-2\pi i k'} e^{-\frac{2\pi i j' k'}{\left(\frac{N}{2}\right)}}$$

$$= e^{-\frac{2\pi i j' k'}{\left(\frac{N}{2}\right)}}$$

So $\quad F_{21} = F_{N/2}$.

(P4)

$\triangle$ F_{12} . $0 \le j', k' < \frac{N}{2}$

$$(F_{12})_{j', 2k'+1} = e^{-\frac{2\pi i}{N}(j' \cdot (2k'+1))}$$

$$= e^{-\frac{2\pi i \, j' \cdot 2k'}{N}} \cdot e^{-\frac{2\pi i}{N} j'}$$

$$= e^{-\frac{2\pi i}{N} j'} \cdot e^{-\frac{2\pi i \, j'k'}{\frac{N}{2}}}.$$

$$F_{12} = \begin{pmatrix} e^{-\frac{2\pi i}{N}0} & & & \\ & e^{-\frac{2\pi i}{N}1} & & \\ & & \ddots & \\ & & & e^{-\frac{2\pi i}{N}(\frac{N}{2})} \end{pmatrix} \quad F_{\frac{N}{2}}.$$

$$= \Omega_{N/2} \, F_{N/2}.$$

(P5).

△ $F_{22} \quad 0 \le j', k' < \frac{N}{2}$

$$\left(F_{22}\right)_{j'+\frac{N}{2},\, 2k'+1}$$

$$= e^{-\frac{2\pi i}{N}(j'+\frac{N}{2})(2k'+1)}$$

$$= e^{-\frac{2\pi i}{(\frac{N}{2})} j' \cdot k'} \cdot \underbrace{e^{-2\pi i k'}}_{1} \cdot e^{-\frac{2\pi i}{N} \cdot j'} \cdot \underbrace{e^{-2\pi i \cdot \frac{1}{2}}}_{-1}$$

$$= -e^{-\frac{2\pi i}{N} j'} \cdot e^{-2\pi i j' k' / (\frac{N}{2})}$$

$$F_{22} = -\Omega_{N/2} F_{N/2}.$$

△ $$F_N = \begin{pmatrix} F_{11} & F_{12} \\ F_{21} & F_{22} \end{pmatrix} = \begin{pmatrix} F_{N/2} & \Omega_{N/2} F_{N/2} \\ F_{N/2} & -\Omega_{N/2} F_{\frac{N}{2}} \end{pmatrix}.$$

$$= \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 1 & \\ & \Omega_{N/2} \end{pmatrix} \begin{pmatrix} F_{N/2} & \\ & F_{N/2} \end{pmatrix}.$$

(P6).

$$\hat{X} = F_N \begin{pmatrix} X_e \\ X_o \end{pmatrix}.$$

$$= \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 1 & \\ & \Omega_{N/2} \end{pmatrix} \begin{pmatrix} F_{N/2} & \\ & F_{N/2} \end{pmatrix} \begin{pmatrix} X_e \\ X_o \end{pmatrix}.$$

$$= \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 1 & \\ & \Omega_{N/2} \end{pmatrix} \begin{pmatrix} F_{N/2} X_e & \\ & F_{N/2} X_o \end{pmatrix}.$$

1 size n FFT $\Rightarrow$ 2 size $N/2$ FFT.

☆ Algorithm

△ func $y = FFT(x, n)$. — $T(n)$.

if $(n=1)$

$y = x$. — $O(1)$.

else.

$\omega_n = e^{-2\pi i/n}$. — $O(1)$

$X_e = \begin{bmatrix} x_0 \\ x_2 \\ \vdots \end{bmatrix}$ $X_o = \begin{bmatrix} x_1 \\ x_3 \\ \vdots \end{bmatrix}$ — " $O(n)$ "

$\operatorname{diag}(\Omega_n) \longrightarrow$ $d = [1, \omega_n, \omega_n^2, \cdots]$ — $O(n)$

$y_T = FFT(X_e, n/2)$ — $T(\frac{n}{2})$

$y_B = FFT(X_o, n/2)$. — $T(\frac{n}{2})$

$y_B = d .* y_B$. — $O(n)$

$y = \begin{bmatrix} y_T + y_B \\ y_T - y_B \end{bmatrix}$ — $O(n)$

end.

Complexity:

$$T(n) = 2T\left(\frac{n}{2}\right) + 3n.$$

$$
\begin{aligned}
T(n) &= 2\left(2T\left(\frac{n}{4}\right) + 3\frac{n}{2}\right) + 3n \\
&= 4T\left(\frac{n}{4}\right) + 2 \cdot 3n \\
&= \cdots \\
&= 2^{\log n}\, T\left(\frac{n}{n}\right) + \cancel{2}\, 3n \log n \\
&\approx 3n \log n = O(n \log n).
\end{aligned}
$$

Applications.

Circulant Matrix.

$$
C = \begin{bmatrix}
c_0 & c_{n-1} & \cdots & c_2 & c_1 \\
c_1 & c_0 & \ddots & & c_2 \\
c_2 & c_1 & \ddots & \ddots & \vdots \\
\vdots & \vdots & \ddots & \ddots & \\
c_{n-1} & c_{n-2} & & \ddots & c_0
\end{bmatrix}
$$

$$Cx = b. \iff c * x = b.$$

$$c = (c_0\ c_1 \cdots c_{n-1})^T.$$

P8

$$b_k = \sum_{i=0}^{n-1} x_i c_{k-i}$$

$$\mathcal{F}(c * x) = \mathcal{F}c \cdot \mathcal{F}x.$$

$$\widehat{(\hat{c} .* \hat{x})} = b. \qquad O(N \log N).$$

$$\widehat{(\hat{b} ./ \hat{c})} = x \qquad O(N \log N)$$

Toeplitz Matrix.

$$A = \begin{bmatrix} a_0 & a_{-1} & \cdots & a_{-n+1} \\ a_1 & \ddots & \ddots & \\ \vdots & \ddots & \ddots & a_{-1} \\ a_{n-1} & & a_1 & a_0 \end{bmatrix}.$$

$$C = \begin{bmatrix} a_0 & a_{-1} & \cdots & a_{-n+1} & a_{n-1} & \cdots & a_1 \\ a_1 & a_0 & a_{-1} & \cdots & a_{-n+1} & a_{n-1} \cdots & a_2 \\ \vdots & & & & & & \\ a_{n-1} & - & - & a_0 & a_{-1} & \cdots & a_{-n+1} \\ a_{-n+1} & \cdots & & & a_0 & & \\ & & & & & & \\ & & & & & & a_0 \end{bmatrix}$$

P9

$Ax = b$

$\iff$ $C\begin{bmatrix} x \\ 0 \end{bmatrix} = \begin{bmatrix} b \\ * \end{bmatrix}$. ✓

~~Can be~~

FFT