DSP:
Fourier Transform

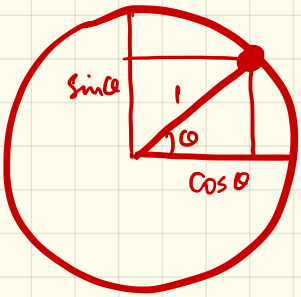
⊙ Abstractions, complex no., Rotation, $e^{j\theta}$

⊙ Model multiplicity → count / numbers.

⊙ Model opposite → I have \$5
How much should I add to make \$0? Ans: −5

Numbers & neg. numbers are all abstractions of reality

Now, How to model rotation?



We have $\cos\theta$ and $\sin\theta$.
Of course we need to express their relation in some way. We cannot just say

$\cos\theta + \sin\theta$ ✗

or $\langle \cos\theta, \sin\theta \rangle$

Which still does not allow algebra.

OK, we can say $\vec{x}\cos\theta + \vec{y}\sin\theta$

But we don't have to say $\vec{x}$, we just need to make $\vec{y}$ become $\perp$ to $\vec{x}$.

How? Well, one way is to say that $\vec{y}\cdot\vec{y}$ should be $-1.\vec{x}$.

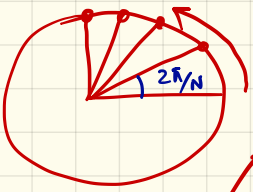
i.e., $y^2 = -1$ $\therefore y = \sqrt{-1} = j$

$\therefore$ Rotation is a vector of $\cos\theta + j\sin\theta$

Aha, this is exactly $= e^{j\theta}$ ⇒ models rotation

Fourier Matrix : An orthogonal Basis

Now, what would rotation at different freq. look like?

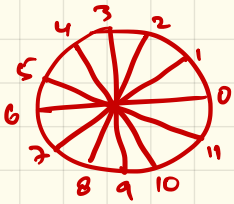


Say we want to model every freq. in N discrete steps.

One cycle/s

$$
\begin{array}{c}
 \\ n=0 \\ n=1 \\ n=2 \\ \vdots \\ n=N-1
\end{array}
\underbrace{
\begin{array}{c}
\text{freq } 0 \\
\begin{bmatrix} e^{j0} \\ e^{j0} \\ e^{j0} \\ \vdots \\ e^{j0} \end{bmatrix}
\end{array}
\begin{array}{c}
\text{freq. } f_1 \\
\begin{bmatrix} e^{j0\cdot\frac{2\pi}{N}} \\ e^{j1\cdot 2\pi/N} \\ e^{j2\cdot 2\pi/N} \\ \vdots \\ e^{j(N-1)\cdot\frac{2\pi}{N}} \end{bmatrix}
\end{array}
\begin{array}{c}
\text{freq } 2f_1 \\
\begin{bmatrix} e^{j0\cdot\left(2\left(\frac{2\pi}{N}\right)\right)} \\ e^{j1\cdot\left(2\cdot\frac{2\pi}{N}\right)} \\ e^{j2\left(2\cdot\frac{2\pi}{N}\right)} \\ \vdots \\ e^{j(N-1)\left(2\cdot\frac{2\pi}{N}\right)} \end{bmatrix}
\end{array}
\cdots
\begin{array}{c}
\text{freq } mf_1 \\
\begin{bmatrix} e^{j0\left(m\frac{2\pi}{N}\right)} \\ e^{j\cdot 1\cdot\left(m\frac{2\pi}{N}\right)} \\ e^{j\cdot 2\cdot\left(m\frac{2\pi}{N}\right)} \\ \vdots \\ e^{j(N-1)\left(m\frac{2\pi}{N}\right)} \end{bmatrix}
\end{array}
\begin{array}{c}
\text{freq } (N-1)f \\
\vdots
\end{array}
}_{\text{Fourier observed each column is orthogonal.}}
$$

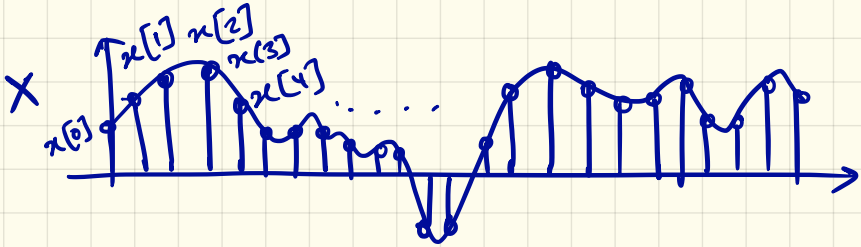
$f_0 \cdot f_1 = 0 \Rightarrow$ you can immediately see because



0	cancels	6
1	cancels	7
2	"	8
3	"	9
4	"	10
5	"	11

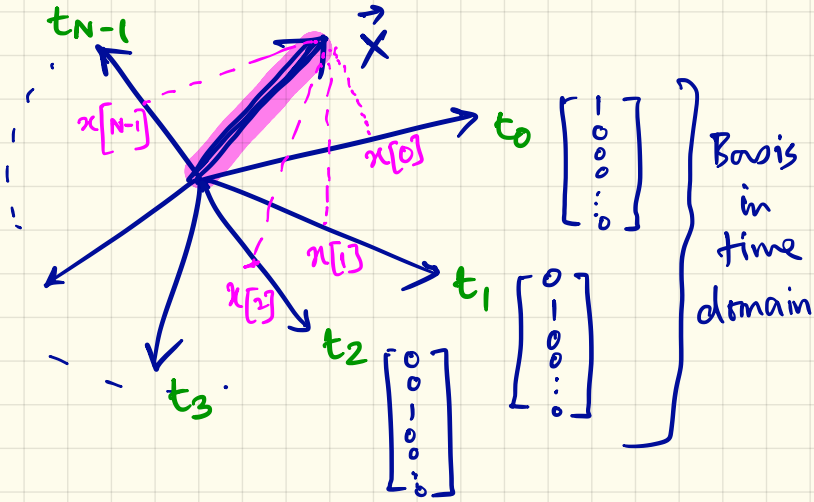
Same holds for any pair.

⊖ Vector Projection to Fourier Basis



$$\vec{X} = \left[x[0], x[1], x[2] \; \ldots \; x[N-1] \right]$$

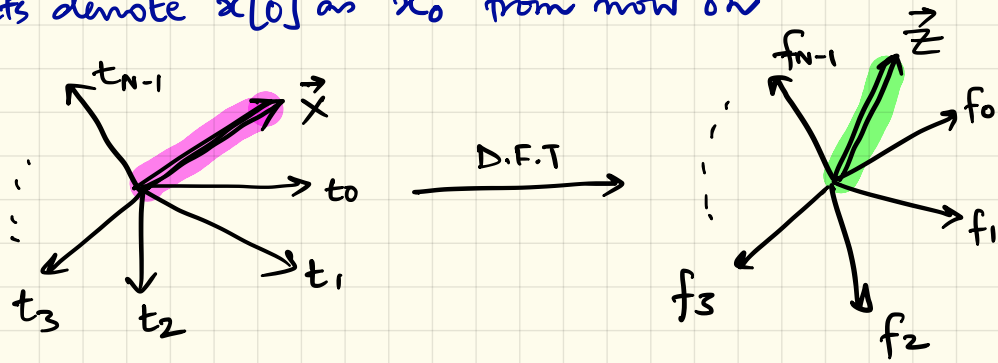
Vector in N dimensional space.



$$\therefore \quad \vec{X} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & & 0 \\ 0 & 0 & 1 & & 0 \\ 0 & 0 & 0 & & \vdots \\ \vdots & \vdots & \vdots & & 1 \\ 0 & 0 & 0 & & \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \\ \vdots \\ x[N-1] \end{bmatrix}$$

(→) Let's denote $x[0]$ as x_0 from now on

(→)



$$\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & & 0 \\ 0 & 0 & 1 & & 0 \\ 0 & 0 & 0 & & \vdots \\ \vdots & \vdots & \vdots & & \\ 0 & 0 & 0 & & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_{N-1} \end{bmatrix} = \underbrace{\begin{bmatrix} | & | & | & & | \\ f_0 & f_1 & f_2 & \cdots & f_{N-1} \\ | & | & | & & | \end{bmatrix}}_{\text{Orthogonal Fourier Basis matrix}} \begin{bmatrix} z_0 \\ z_1 \\ z_2 \\ \vdots \\ z_{N-1} \end{bmatrix}$$

$$\therefore \quad z(m) = F^{-1} \cdot \vec{x}$$

$$z(m) = (F^*)^T \cdot \vec{x}$$

$$\therefore \quad \underbrace{\begin{bmatrix} z_0 \\ z_1 \\ z_2 \\ \vdots \\ z_{N-1} \end{bmatrix} = \begin{bmatrix} \text{———} & f_0^* & \text{———} \\ \text{———} & f_1^* & \text{———} \\ \text{———} & f_2^* & \text{———} \\ \vdots & \vdots & \\ \text{———} & f_{N-1}^* & \text{———} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_{N-1} \end{bmatrix}}_{\text{Discrete Fourier Transform (DFT)}}$$

DFT Equations

Consider 3D case :

$$\begin{bmatrix} z_0 \\ z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} — & f_0^* & — \\ — & f_1^* & — \\ — & f_2^* & — \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix}$$

$$z_0 = e^{-j\frac{2\pi}{N}.0.0}.x_0 + e^{-j\frac{2\pi}{N}.0.1}.x_1 + e^{-j\frac{2\pi}{N}.0.2}x_2$$

$$z_1 = e^{-j\frac{2\pi}{N}.1.0}.x_0 + e^{-j\frac{2\pi}{N}.1.1}x_1 + e^{-j\frac{2\pi}{N}.1.2}.x_2$$

$$z_2 = \underbrace{e^{-j\frac{2\pi}{N}.2.0}x_0 + e^{-j\frac{2\pi}{N}.2.1}.x_1 + e^{-j\frac{2\pi}{N}.2.2}.x_2}$$

DFT :

$$Z(m) = \sum_{n=0}^{N-1} x[n]\, e^{-j\frac{2\pi}{N}.m.n}$$

IDFT :

$$x[n] = \sum_{m=0}^{N-1} Z(m)\, e^{j\frac{2\pi}{N}.m.n}$$

Scaling by $\frac{1}{\sqrt{N}}$

Note that dot.product = Projection only when the dot product is normalized.

Now, for DFT, observe that each column of F is not a unit vector, e.g., $[1\ 1\ 1]^T$.

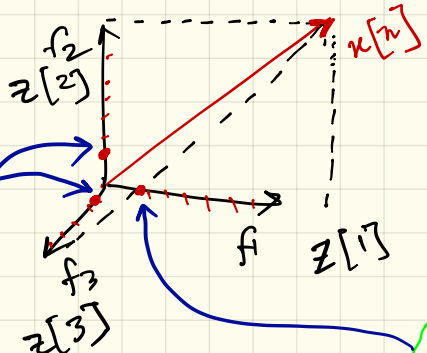
$\therefore$ The DFT needs to be scaled by the length of the vector $1/\sqrt{N}$

i.e., DFT $X_m = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x[n]\, e^{-j\frac{2\pi}{N}mn}$

IDFT $x_n = \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} X_m\, e^{j\frac{2\pi}{N}mn}$

⊛ IDFT : Apply Parseval's synthesis Identity

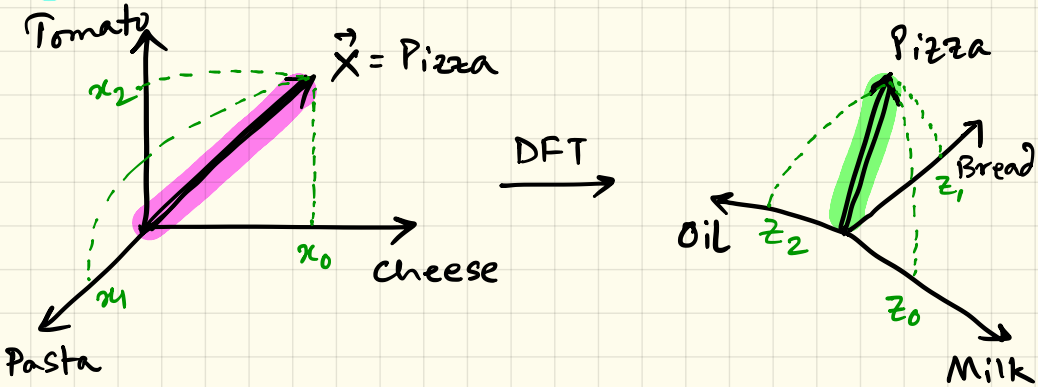
$$x[n] = \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} Z(m) e^{j\frac{2\pi}{N} \cdot mn}$$



⇒ Add up the projections multiplied by the corresponding basis vector.

$$x[1] = Z(1) e^{j\frac{2\pi}{N} \cdot 1 \cdot n} + Z(2) e^{j\frac{2\pi}{N} \cdot 2n} + Z(3) e^{j\frac{2\pi}{N} \cdot 3n}$$

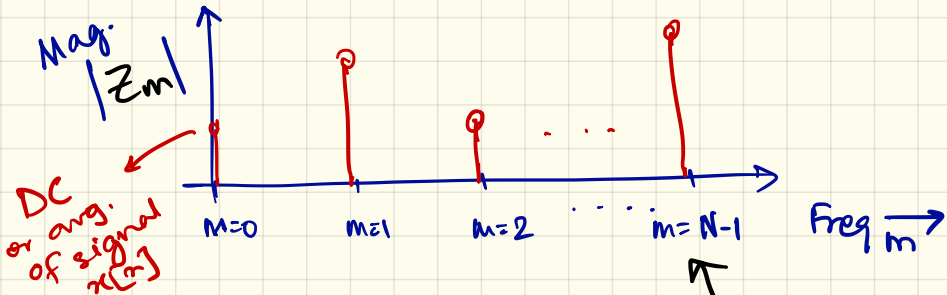
③ Analogy:



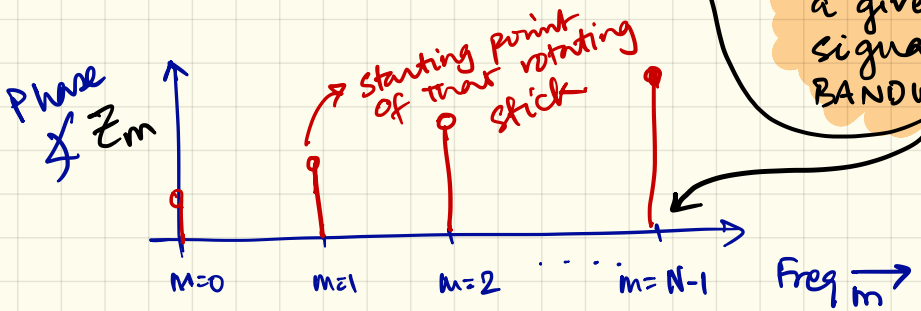
- Cook Italian food with 2 different "Basis ingredients"
- DFT helps go from one basis to another.

Visualizing DFT, Bandwidth & Resolution

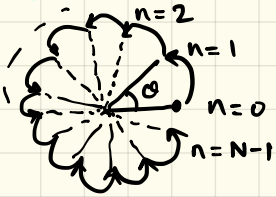
- Z_m is a complex vector



Max freq. of a given signal = BANDWIDTH



Translating to real-world frequencies



Say sampling freq. = f_s

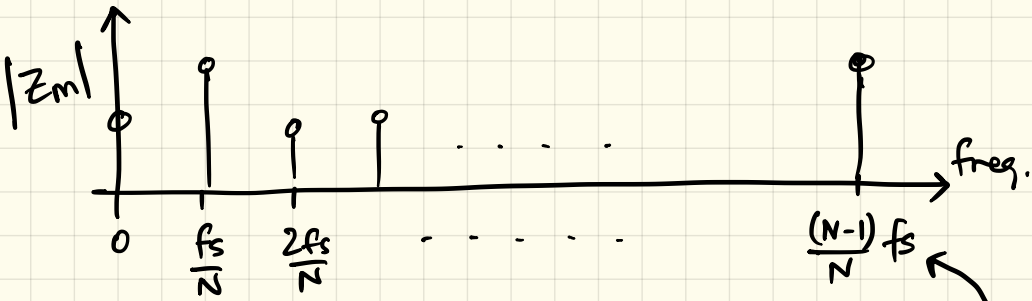
Now, slowest freq = N samples/cycle

N samples take $N \cdot \frac{1}{f_s}$ time

1 cycle takes N/f_s time

$\therefore$ Slowest freq. $= \frac{f_s}{N}$ Hz.

→ ∴ Increasing freq. @ $0, \frac{f_s}{N}, \frac{2f_s}{N}, \frac{3f_s}{N} \ldots \frac{(N-1)f_s}{N}$



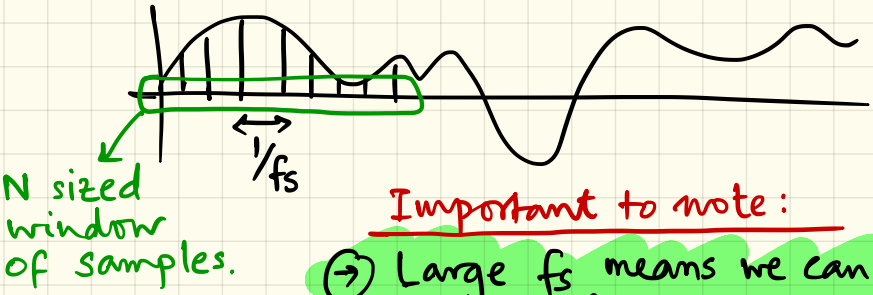
$\Downarrow$

$\approx f_s$ for large N

→ Thus, when analyzing a given signal, we have two knobs.

→ Sampling freq. f_s

→ No. of FFT points N



Important to note:

→ Large f_s means we can see ti'll large freq. components f_s

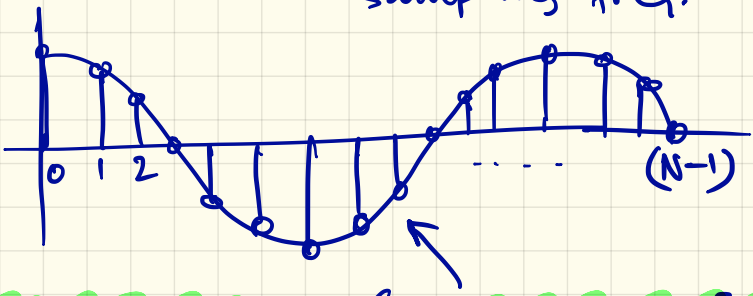
→ Large N implies we have better resolution of f_s (i.e., f_s/N)

3 Examples of DFT : Cos, $e^{j\theta}$, Sin

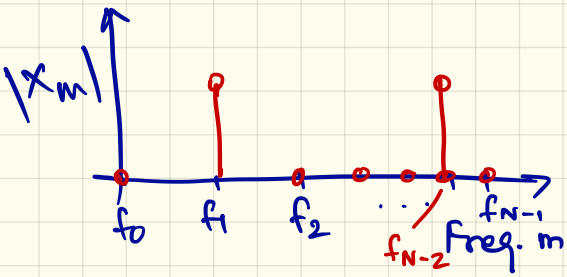
Everyday signal $\cos 2\pi f_m t$

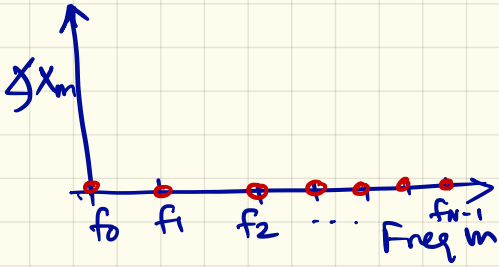
Discrete sampled signal $x[n] = \cos 2\pi f_m n t_s$

where $t_s = \frac{1}{f_s}$ is sampling interval, f_s is sampling freq.

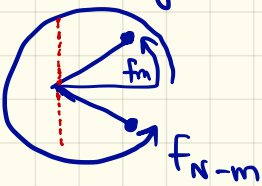


Now what is the DFT ($x[n] = \cos 2\pi f_1 n t_s$)



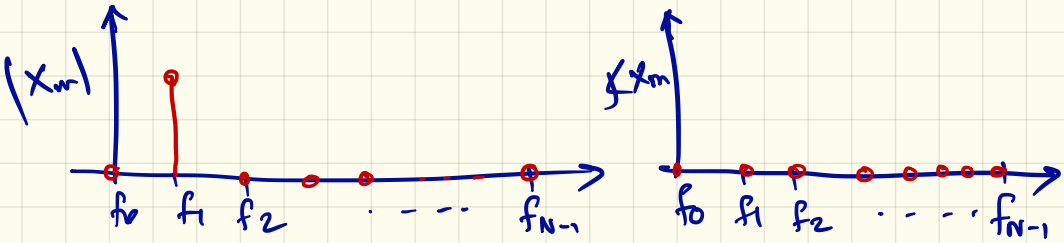


Real signals always symmetric because the imaginary component needs to be cancelled out by a much faster moving rotation



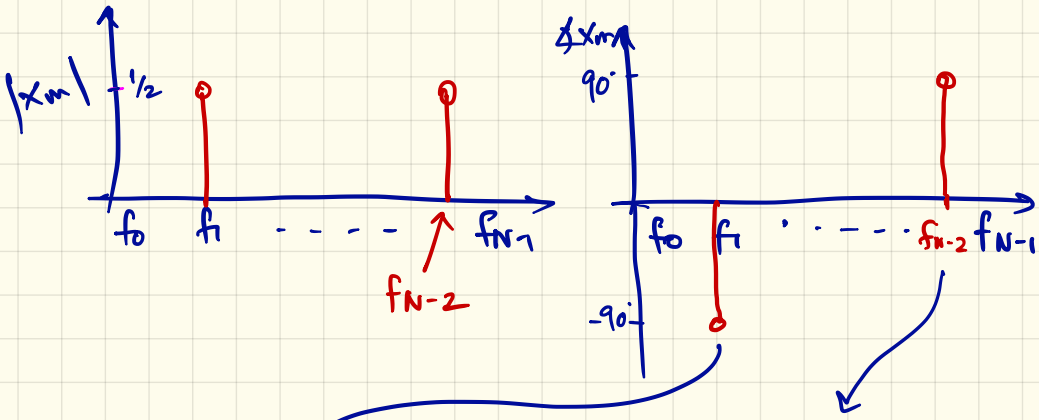
Also, phase ZERO because all rotations start from zero.

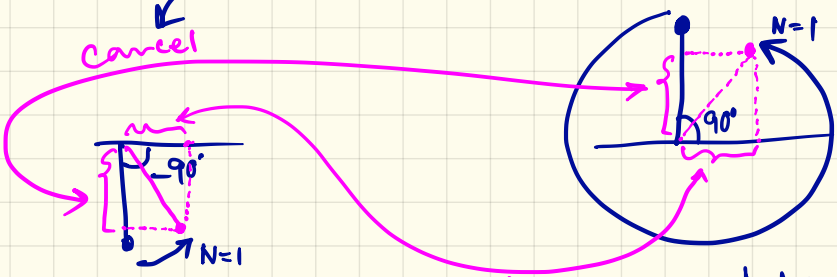
(→) What is DFT($x[n] = e^{j2\pi f_1 n t_s}$) ?



Just one stick rotating at f_1 is good enough to create this signal.

(→) Now, what is DFT($x[n] = \sin 2\pi f_1 n t_s$) ?



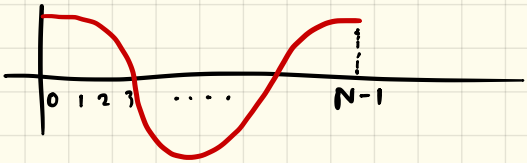


add up to 2x but multiplied by $x_m = 1/2$

⊖ Signals containing full cycles

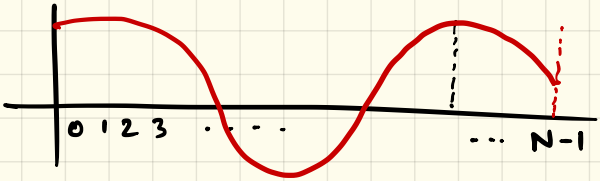
When we took the DFT of the $\cos(2\pi f_0 n t_s)$ signal, the implicit assumption is that we took N points that exactly covers integer no. of $\cos(\cdot)$ cycles.

$x =$



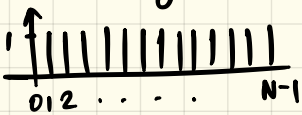
⊖ If we take N samples that only covers part of the signal, say

$y =$



Then, we can no longer form this signal with just the f_0 freq. component.

⊖ Instead, this signal y can be viewed as a truncated x signal, truncated by a pulse train of N samples



$$\therefore \quad \mathrm{DFT}(y) = \mathrm{DFT}(x \cdot \text{PulseTrain})$$
$$\neq \mathrm{DFT}(x)$$

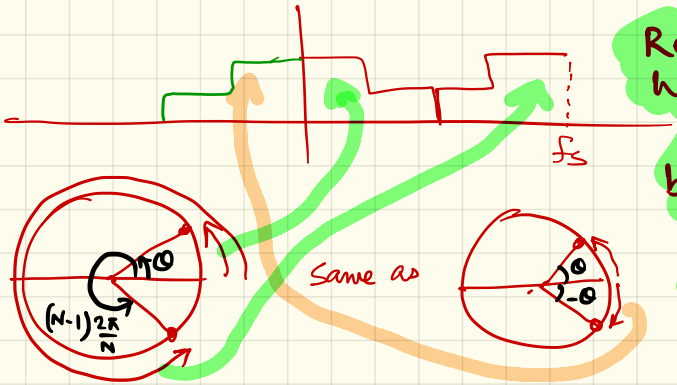
... So be careful.

⊛ Properties of DFT

① DFT is linear :

$$DFT(x[n] + y[n]) = DFT(x[n]) + DFT(y[n])$$
$$= X_m + Y_m$$

② For real signals, DFT symmetric (negative freq.)

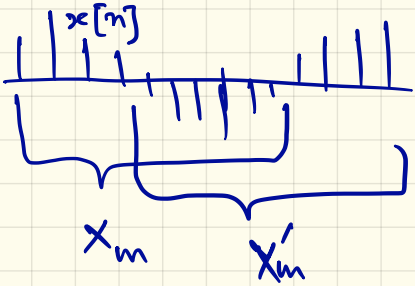


Real signals have symm. spectrogram because imaginary part cancelled by faster moving stick.

So this is equivalent to taking the mirror image and pasting it on the (-)ve freq. axis.

Intuitively, the second stick is rotating in the clockwise direction to cancel the imaginary parts of the real signal.

③ DFT of shifted signal is original DFT with just a phase shift.



$$x'[n] = x[n+k]$$

i.e., $x'[n]$ is shifted by k samples. Then

$$y[n] = x[n+k]$$

$$\therefore \quad Y_m = \sum_{n=0}^{N-1} x[n+k]\, e^{-j\frac{2\pi}{N}.m.n}$$

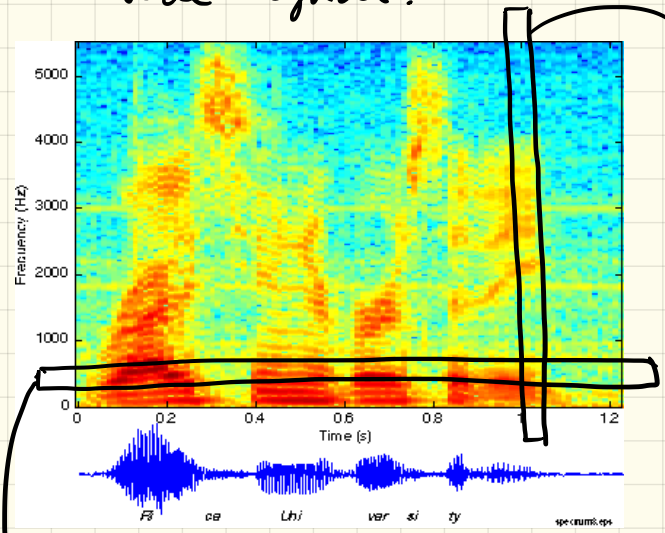
$$= \sum_{n=0}^{N-1} x[n+k]\, e^{-j\frac{2\pi}{N}.m.(n+k)} \cdot e^{j\frac{2\pi}{N}.mk}$$

$$\boxed{Y_m = \underbrace{e^{j\frac{2\pi}{N}.m.k}}_{\text{means a phase shift only to } X_m.} X_m}$$

$$X'_m = \underbrace{e^{j\frac{2\pi}{N}.k.m}}_{\text{means only phase shift}} X_m$$

Real spectrogram

Voice signal.

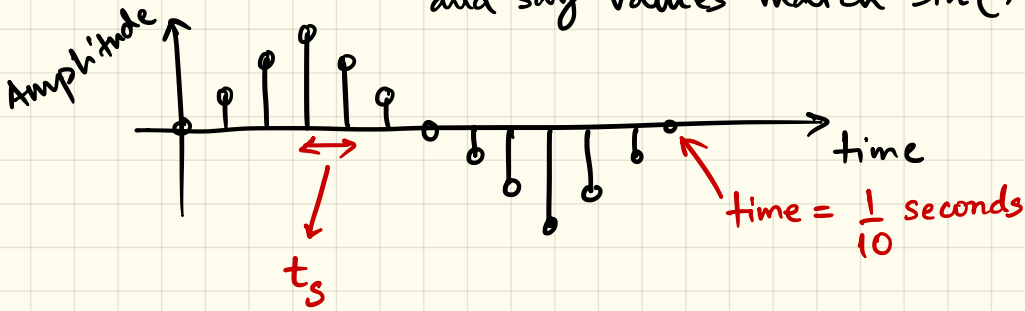


shows various freq. components at this time instant

shows how a specific freq. comes and goes over time.

Nyquist's Sampling

- Say you are given the following discrete samples and asked to reconstruct the analog signal... and say values match $\sin(\cdot)$



- We can say freq $f = 10$ Hz

$\therefore$ Analog signal $= \sin(2\pi f n t_s)$

- But is this the only signal that fits these values?

Observe: $\sin(2\pi f n t_s + 2\pi m)$ also fits the values given, if m = integer

i.e., $\sin\left(2\pi\left(f_0 + \frac{m}{n t_s}\right) n t_s\right)$ fits values

Choosing m as multiple of n $\left(\frac{m}{n} = k\right)$

$$\sin\left(2\pi\left(f_0 + k \cdot \frac{1}{t_s}\right) n t_s\right)$$

$$= \sin\left(2\pi\left(f_0 + k f_s\right) n t_s\right)$$ ... f_s is sampling freq.

→ This means, many other frequencies, separated by kf_s from the core freq. f fits the given set of points.

$\Downarrow$

called "ALIASED" frequencies.

ie., $(f + kf_s)$ are aliases of f for different values of k.

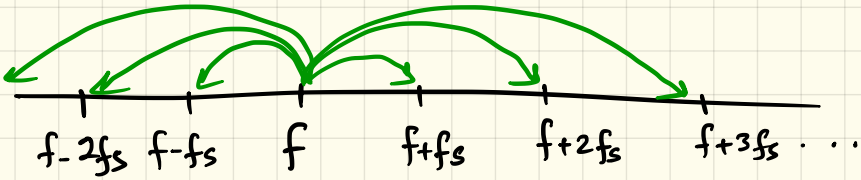
$\Downarrow$

If I sample a signal at f_s, and therefore get to see freq. components of $\frac{f_s}{N}, \frac{2f_s}{N}, \frac{3f_s}{N} \ldots \frac{N-1}{N} f_s$,

actually, each of these freq. components could have aliased freq. from other freq contained in the signal.

→ To ensure we remove aliasing (which pollutes) we need to understand max freq. component of signal, say f_w.

- Observe that higher the f_s, more separated are the aliased freq. Example



- Now, to avoid pollution for a given signal of bandwidth f_w, we need no. freq. within it to alias to within its bandwidth.

- That is f_w should alias to left of $-f_w$ and $-f_w$ should alias to right of f_w

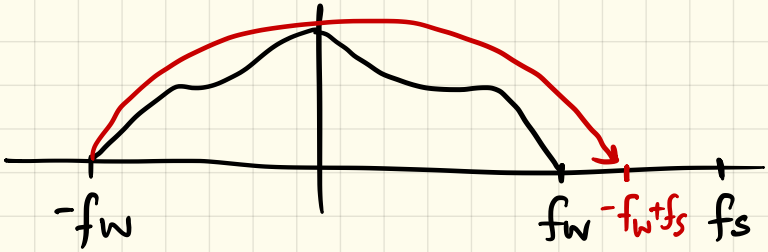
- This implies : $f_w - f_s < -f_w$

or $-f_w + f_s > f_w$

$$\Downarrow$$

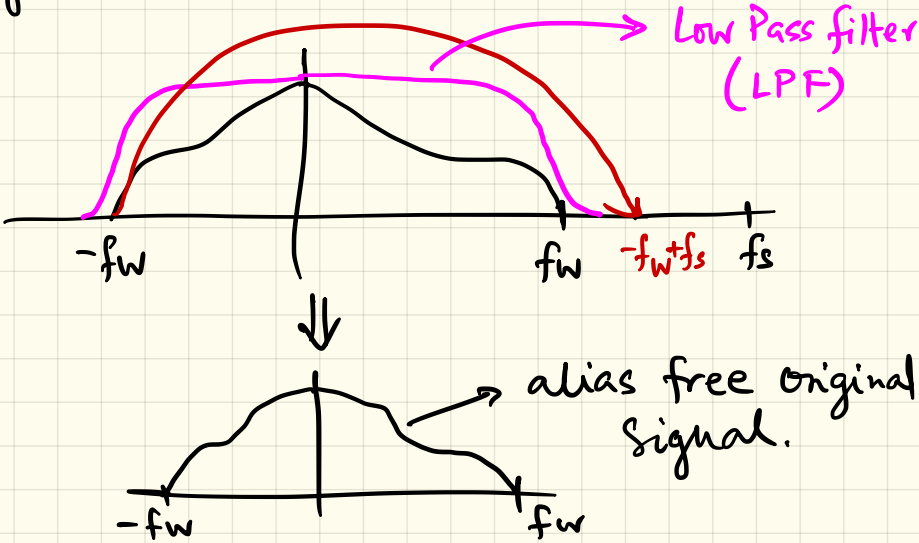
$$\boxed{f_s > 2f_w}$$

i.e., sampling freq. must be twice b/w to have an unpolluted reconstruction of the signal.

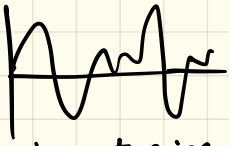


Thus, sampling at f_s may show the alias of $(-f_w+f_s)$.

However, knowing f_w, we can filter out the signal at f_w, thereby removing all the signals outside $[-f_w, f_w]$ giving us a full alias free reconstruction of the signal of interest.



⊖ The signal processing flow.



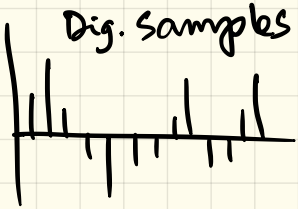
input signal
(e.g., voice)

x_t → **LPF** → **Nyquist sample** →

Filter at f_w (≈ 20 KHz)

Sample at f_s ≥ 40 KHz

$x[n]$

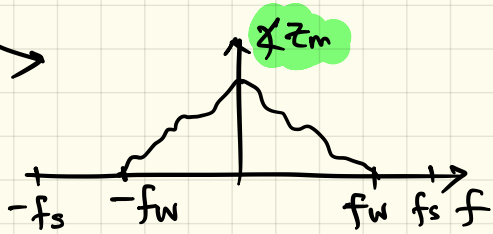


Dig. samples

→ **FFT** →

N point FFT, say $N = 1024$

Z_m → **Visualize** →



Questions?