EECS16A
Shazam

Welcome! We'll be starting at Berkeley Time.

Today's Agenda

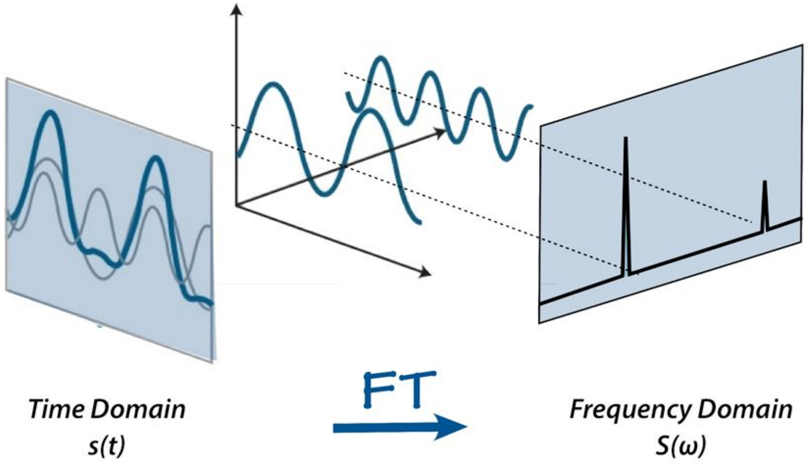
- What is the Fourier Transform?
- DFT Overview
- How does Shazam recognize songs?

Semester Outline

- Shazam
- Imaging
- Acoustic Positioning
- Voice Recognition

What is the Fourier Transform?

- Many interpretations and many uses—but for us, it’s a way to decompose a signal into its frequency components!
- This is because we can decompose any arbitrary signal into a combination of sinusoids (sines and cosines).



DFT (Discrete Fourier Transform) Cont.

$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot e^{-j\frac{2\pi}{N}kn}, \quad k = 0, 1, \ldots, N-1$$

- X(k): the Fourier coefficients
- x(n): our discrete signal at timestep n
- N: number of points in the signal
- Let’s break this down!

DFT Example (N=4)

$$X(k) = \sum_{n=0}^{3} x(n) \cdot \boxed{e^{-j\frac{2\pi}{4}kn},} \qquad k = 0, 1, 2, 3$$

- For each k, we have the following complex exponentials:

$$[1, 1, 1, 1](k = 0)$$

$$[1, e^{-j\frac{2\pi}{4}}, e^{-j\frac{2\pi}{4}*2} e^{-j\frac{2\pi}{4}*3}](k = 1)$$

$$[1, e^{-j\frac{2\pi}{4}*2}, e^{-j\frac{2\pi}{4}*4} e^{-j\frac{2\pi}{4}*6}](k = 2)$$

$$[1, e^{-j\frac{2\pi}{4}*3}, e^{-j\frac{2\pi}{4}*6} e^{-j\frac{2\pi}{4}*9}](k = 3)$$

DFT Example (N=4)

- Simplifying the fractions, we get this:

$$[1, 1, 1, 1](k = 0)$$

$$[1, e^{-j\frac{\pi}{2}}, e^{-j\pi}, e^{-j\frac{3\pi}{2}}](k = 1)$$

$$[1, e^{-j\pi}, e^{-j2\pi}, e^{-j3\pi}](k = 2)$$

$$[1, e^{-j\frac{3\pi}{2}}, e^{-j3\pi}, e^{-j\frac{9\pi}{2}}](k = 3)$$

DFT Example (N=4)

- We can also simplify further by using $e^{-j2\pi} = e^0 = 1$

$$[1, 1, 1, 1](k = 0)$$

$$[1, e^{-j\frac{\pi}{2}}, e^{-j\pi}, e^{-j\frac{3\pi}{2}}](k = 1)$$

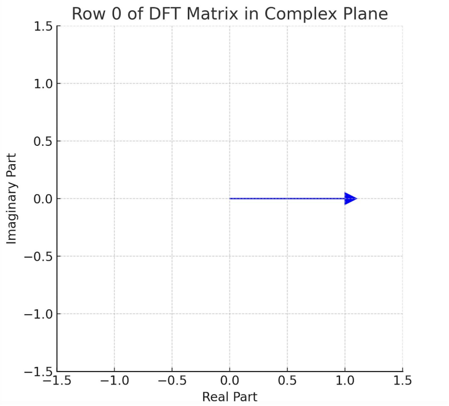
$$[1, e^{-j\pi}, 1, e^{-j\pi}](k = 2)$$

$$[1, e^{-j\frac{3\pi}{2}}, e^{-j\pi}, e^{-j\frac{\pi}{2}}](k = 3)$$

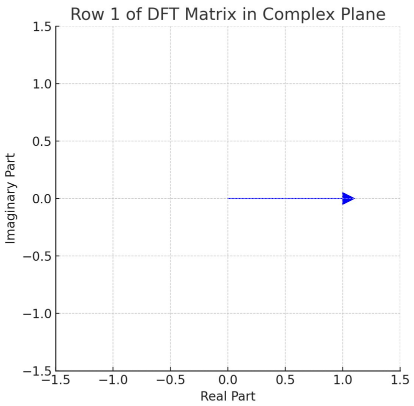
DFT Example (N=4)

- What do these vectors look like for each timestep?
- We can see that k is our “frequency”.

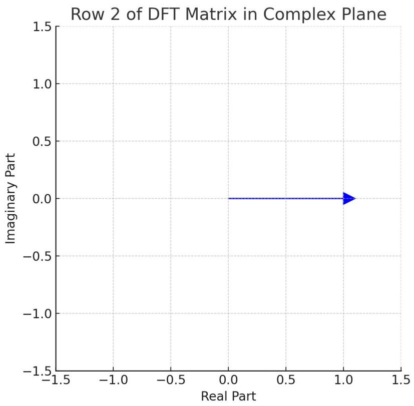
$$[1, 1, 1, 1] (k = 0)$$



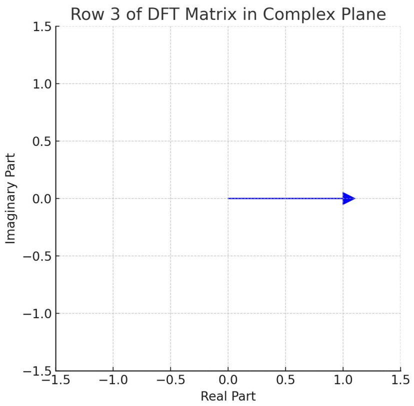
$$[1, e^{-j\frac{\pi}{2}}, e^{-j\pi}, e^{-j\frac{3\pi}{2}}] (k = 1)$$



$$[1, e^{-j\pi}, 1, e^{-j\pi}] (k = 2)$$



$$[1, e^{-j\frac{3\pi}{2}}, e^{-j\pi}, e^{-j\frac{\pi}{2}}] (k = 3)$$



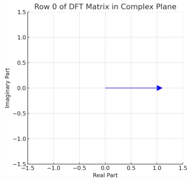
DFT Example (N=4)

$$X(k) = \sum_{n=0}^{3} x(n) \cdot e^{-j\frac{2\pi}{4}kn}, \quad k = 0, 1, 2, 3$$

- Taking a look at the equation, we find that we're doing a dot product between x, our signal, and each of the four vectors we just derived.

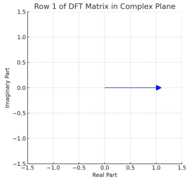
DFT Example (N=4)

$[1, 1, 1, 1](k = 0)$



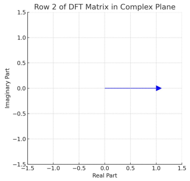
$\cdot\ x(n)(= [x(0), x(1), x(2), x(3)]) \rightarrow X(0)$

$[1, e^{-j\frac{\pi}{2}}, e^{-j\pi}, e^{-j\frac{3\pi}{2}}](k = 1)$



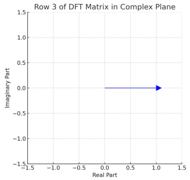
$\cdot\ x(n)(= [x(0), x(1), x(2), x(3)]) \rightarrow X(1)$

$[1, e^{-j\pi}, 1, e^{-j\pi}](k = 2)$



$\cdot\ x(n)(= [x(0), x(1), x(2), x(3)]) \rightarrow X(2)$

$[1, e^{-j\frac{3\pi}{2}}, e^{-j\pi}, e^{-j\frac{\pi}{2}}](k = 3)$



$\cdot\ x(n)(= [x(0), x(1), x(2), x(3)]) \rightarrow X(3)$

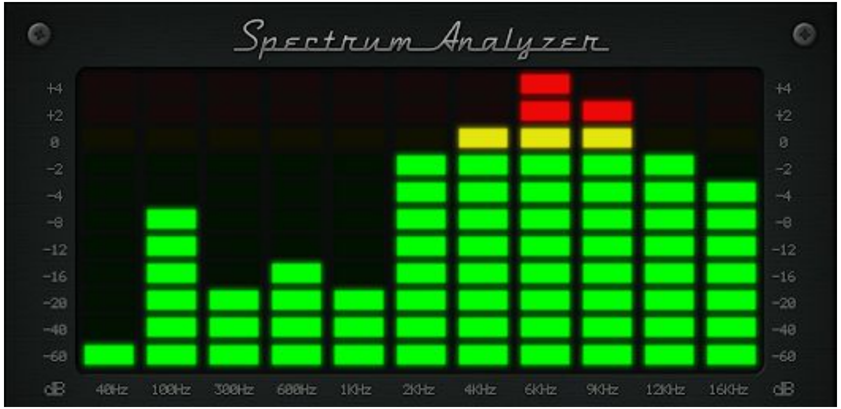
Our Fourier coefficients!

What do the Fourier Coefficients Tell Us?

- Intuitively, we're using the dot product to calculate the "similarity" between our signal and the vectors of each frequency.
- Thus, we can say that the Fourier coefficients are telling us how much contribution each frequency component has in our signal.
- Another way to look at this is that it's "pulling" out the sinusoidal components in our signal (remember, we said that all signals can be decomposed into combinations of sinusoids.)
- Notably, by way of how the DFT is set up, the vector of Fourier coefficients is the same length as our signal.

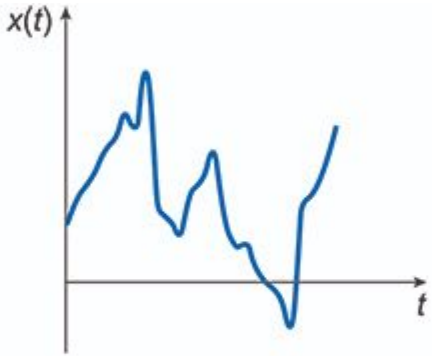
Terminology

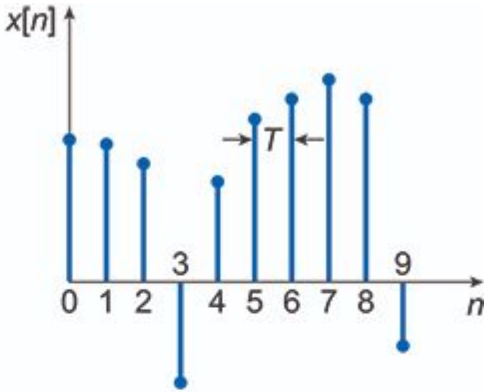
- We call the vector of Fourier coefficients the “DFT of our signal x.”
- We also call the Fourier transform of a signal the “spectrum” of a signal.
- e.g.) ever seen these spectrum analyzers / plots for songs? They’re performing the Fourier transform on your songs in real-time.



What Makes the DFT “Discrete?”

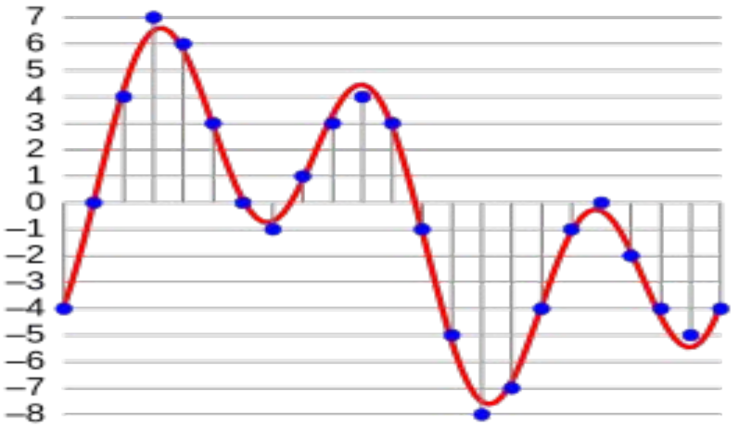
- The data types that we have been working with is discrete, meaning that there are separate values for each discretized timestep.
- The DFT is discrete in both the frequency and time domain.





Why is the DFT Useful for Us?

- All the data we work with on computers is digital (discrete)
- That means the time series data in question can be represented as a 1-d vector, which is what the DFT uses.
  - e.g.) mono channel audio data:



What does Shazam do?

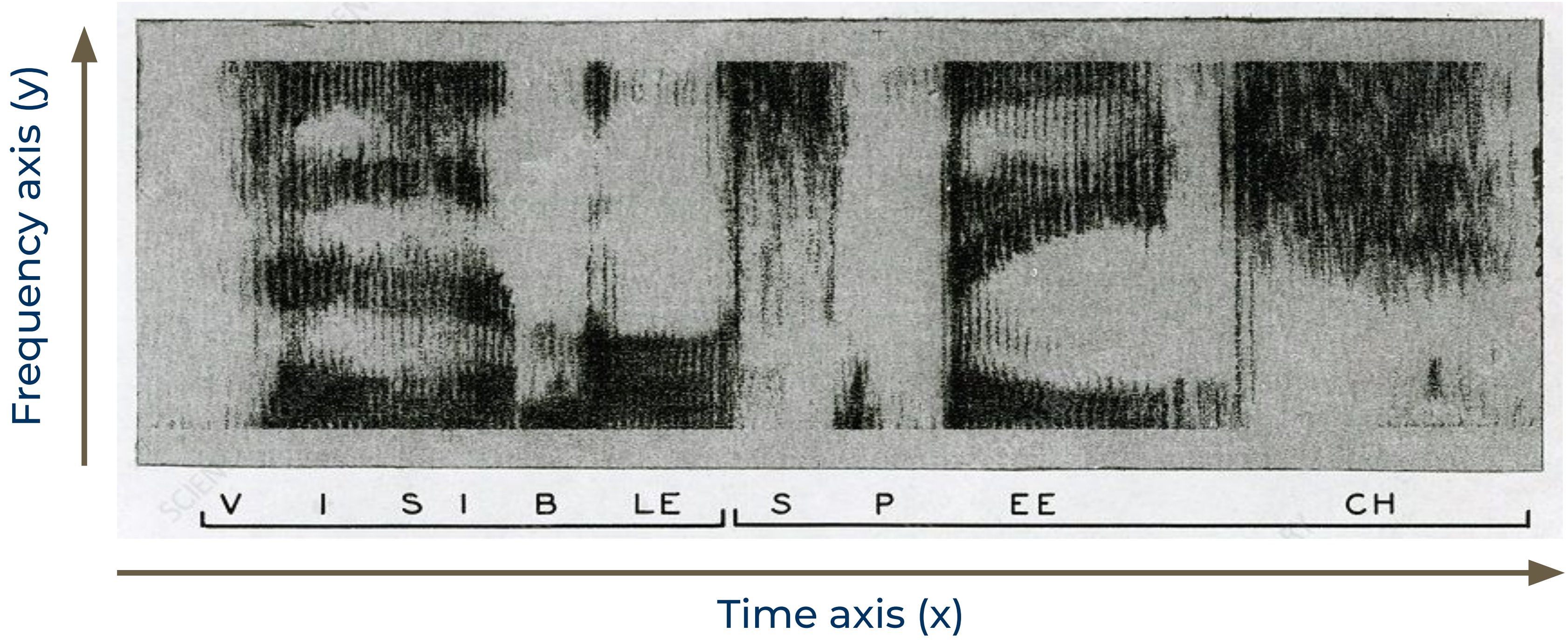
- “Audio fingerprinting”
  - Q: how do we analyze songs so that we can give them a unique “fingerprint” that is unique to a song?
  - A: use spectral analysis! (i.e. analyze the spectrum / DFT of the song)

DFT in Shazam

- DFT is a powerful tool here
- Problem: there’s zero temporal information
    - If you do DFT on the entire song, it gives you the frequency contents of the entire song. It doesn’t tell us about how those contents change throughout the duration of the song.
    - e.g.) a song might be more bass-heavy in certain parts of the song.

Spectrograms

Many DFTs on short snippets of audio!



Audio Fingerprinting Scheme

- We have a spectrogram - a 2D plot that contains values
    - Each value tells us the “intensity” of that frequency component at that time
        - Coordinates: (time, frequency)

1. We calculate a threshold value to find the “peaks” in our spectrogram.
2. We only highlight the peaks in our spectrogram and set everything else to zero.
3. We turn that image into a value that can be used easily by a computer with a hashing algorithm.
4. We have an audio fingerprint!

Audio Fingerprinting Scheme

- We can run the same fingerprinting algorithm on a snippet of a song we would like to identify.
- We can then compare the value we get, and compare it with a pre-existing saved value from all the songs we've analyzed, and find a match!

Feedback

Please provide feedback with this anonymous feedback form!!

https://tinyurl.com/fb-student-fa24