Framing - dividing the audio signal into short time segments (20-40 ms, 25 ms is standard)

Example:

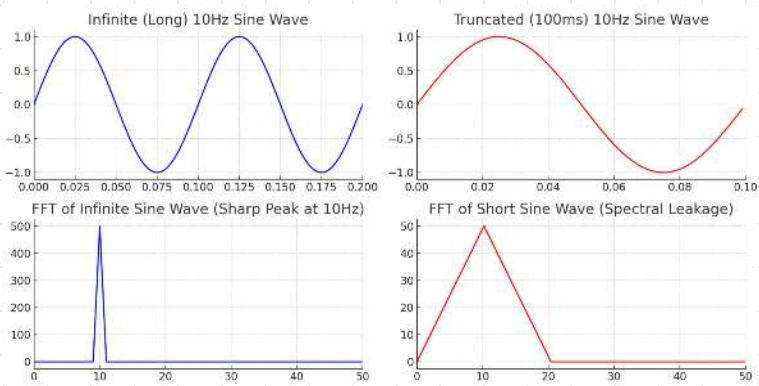
- given a 10s song sampled at 16 000 Hz we get 160 000 samples
- instead of processing all at once, we break it into 25 ms frames (400 samples per frame)
- frames overlap (e.g. 50%) to capture smooth transitions, this is called **frame step** (usually 10 ms)

this allows **analysis of short-term characteristics** of music

when we divide an audio signal into frames, each frame is finite

this creates artificial discontinuities at the edges of the frames, which introduces unwanted frequency artifacts - **spectral leakage**

- sharp transitions at the edges of frames - sudden jumps in waveform - add extra frequency components that weren't originally present in the signal



Windowing

- used to reduce spectral leakage
- we multiply each frame by a window function to smooth the edges before applying the Fourier Transform

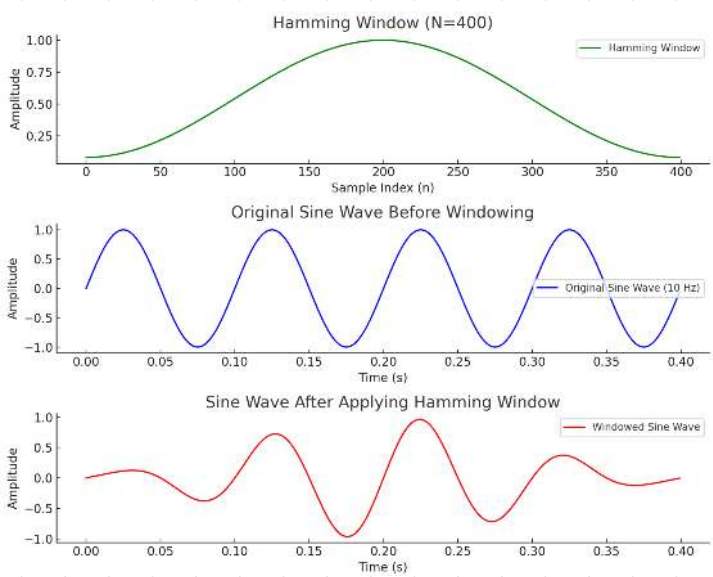
Hamming window function

- use cases: speech processing, music analysis

$$w[n] = 0.54 - 0.46 \cdot \cos\left(\frac{2\pi n}{N-1}\right)$$

n – discrete time index of the window function, ranging from 0 to N-1

e.g., we have a frame of 400 samples, n = 0, 1, ..., 399



Fourier Transform

$$F(f) = \int_{-\infty}^{\infty} f(t)\, e^{-2\pi i f t}\, dt$$

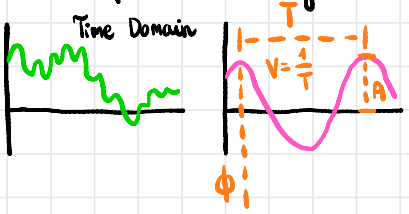
Inverse Fourier Transform

$$F(t) = \int_{-\infty}^{\infty} F(f)\, e^{2\pi i f t}\, dv$$

a transform - a mapping between domains

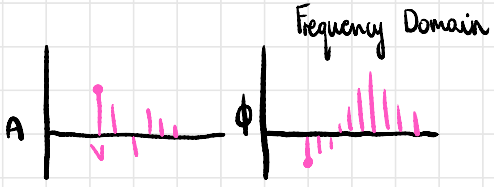
Time $f(t)$ $\Leftrightarrow$ Frequency $F(f)$ $f = \frac{\omega}{2\pi}$ [Hz]

any continious signal in the time domain can be represented by a sum of sinusoids



↓

can be described by its amplitude, frequency and phase



signal in the time domain

$x \in \mathbb{R}^{28\,200} = 3.525\,s \times 8000\,Hz$

(28 200: sample length; 8000 Hz: sample rate)

Why Fourier Transform?

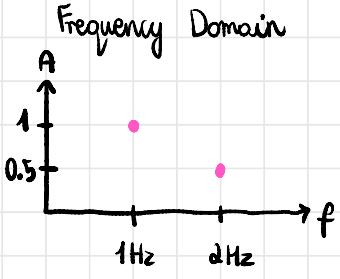
- we are trying to transform the signal into sth that is easier to work with

- sinuses are the only wave form that doesn't change shape when subjected to LTI

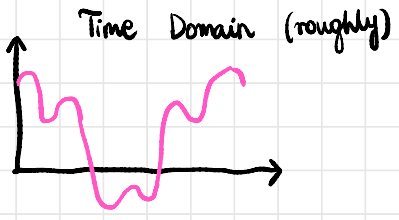
Linear Time-Invariant System

linear; system's response doesn't change over time
predictable, stable, easy to analyze

MATH



$$f(t) = \cos(2\pi t) + 0.5\cos(4\pi t)$$



general formula (points)

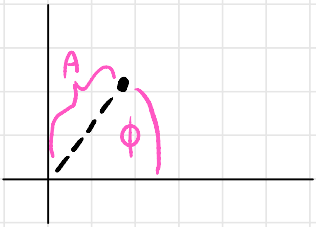
$$f(t) = \sum_{v=-\infty}^{\infty} A(f)\cos(2\pi f t)$$

general formula (continuous)

$$f(t) = \int_{-\infty}^{\infty} A(f)\cos(2\pi f t)\,df$$

(without phase information)

phase and amplitude can be described by a complex number



$$A = \sqrt{real^2 + Im^2}$$

$$\tan(\phi) = \frac{Im}{real}$$

Euler's formula $e^{it} = \cos(t) + i \cdot \sin(t)$

with phase information we have $F(v)\, e^{2\pi i v t}$

real part of $F(v)$ is an even function and the imaginary part is odd

Discrete Fourier Transform

- in practical applications we deal with discrete signals sampled at regular intervals

$$X_k = \sum_{n=0}^{N-1} x_n \cdot e^{-i \frac{2\pi}{N} k n}$$

N - number of samples

- requires $O(N^2)$ complex multiplications, which is very slow for large signals

Fast Fourier Transform

DFT given by

$$\hat{f}_k = \sum_{n=0}^{N-1} f_n \omega_N^{-nk}, \quad k = 0, 1, \dots, N-1, \quad \omega_N = e^{\frac{2\pi i k}{N}}$$

written out explicitly is

$$\hat{f}_0 = f_0 \omega_N^{-0\cdot 0} + f_1 \omega_N^{-1\cdot 0} + \dots + f_{N-1} \omega_N^{-(N-1)\cdot 0} \qquad k=0$$

$$\vdots$$

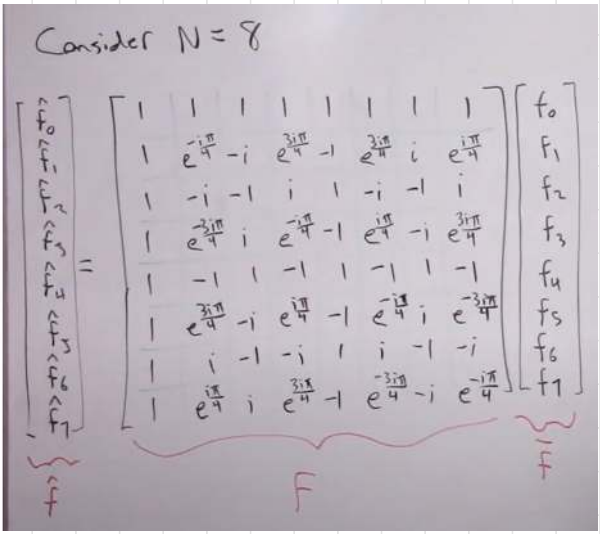
$$\hat{f}_{N-1} = f_0 \omega_N^{-0\cdot(N-1)} + f_1 \omega_N^{-1\cdot(N-1)} + \dots + f_{N-1} \omega_N^{-(N-1)\cdot(N-1)} \qquad k = N-1$$

in matrix form is

$$\begin{bmatrix} \hat{f}_0 \\ \hat{f}_1 \\ \hat{f}_2 \\ \vdots \\ \hat{f}_{N-1} \end{bmatrix} = \underbrace{\begin{bmatrix} \omega_N^{-0\cdot 0} & \omega_N^{-1\cdot 0} & \cdots & \omega_N^{-(N-1)\cdot 0} \\ \omega_N^{-0\cdot 1} & \omega_N^{-1\cdot 1} & \cdots & \omega_N^{-(N-1)\cdot 1} \\ \omega_N^{-0\cdot 2} & \omega_N^{-1\cdot 2} & \cdots & \omega_N^{-(N-1)\cdot 2} \\ \vdots & \vdots & \cdots & \vdots \\ \omega_N^{-0\cdot(N-1)} & \omega_N^{-1\cdot(N-1)} & \cdots & \omega_N^{-(N-1)\cdot(N-1)} \end{bmatrix}}_{\text{(DFT matrix)}} \begin{bmatrix} f_0 \\ f_1 \\ f_2 \\ \vdots \\ f_{N-1} \end{bmatrix}$$

$\hat{f}$ ($N \times 1$) $= F$ ($N \times N$) $\cdot \bar{f}$ ($N \times 1$)

so $\hat{f} = F \cdot \bar{f}$ $\searrow O(N^2)$



$$\bar{f}_R = \begin{bmatrix} \bar{f}_{even} \\ \bar{f}_{odd} \end{bmatrix} = \begin{bmatrix} f_0 \\ f_2 \\ f_4 \\ f_6 \\ f_1 \\ f_3 \\ f_5 \\ f_7 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f_0 \\ f_2 \\ f_4 \\ f_6 \\ f_1 \\ f_3 \\ f_5 \\ f_7 \end{bmatrix}$$

$\bar{f}_R$ $\qquad$ T $\qquad$ $\bar{f}$

R as in reorder

$$\bar{f}_R = T\,\bar{f}$$

$$\hat{f} = \underbrace{F \cdot T^T}_{F_R} \cdot \bar{f}_R \Rightarrow \hat{f} = F_R \cdot f_R$$

$O(N^2)$

$$\hat{f} = F_R \begin{bmatrix} \bar{f}_{even} \\ \bar{f}_{odd} \end{bmatrix}$$

Note that F_R can be written as

$$F_R = \begin{bmatrix} I_{\frac{N}{2}\times\frac{N}{2}} & D_{\frac{N}{2}\times\frac{N}{2}} \\ I_{\frac{N}{2}\times\frac{N}{2}} & -D_{\frac{N}{2}\times\frac{N}{2}} \end{bmatrix} \begin{bmatrix} F_{\frac{N}{2}\times\frac{N}{2}} & 0_{\frac{N}{2}\times\frac{N}{2}} \\ 0_{\frac{N}{2}\times\frac{N}{2}} & F_{\frac{N}{2}\times\frac{N}{2}} \end{bmatrix} \begin{bmatrix} \bar{f}_{even} \\ \bar{f}_{odd} \end{bmatrix}$$

so for N=8 this will be of size 4×4

$N\times N$ $\quad$ $N\times N$ $\quad$ $O((\frac{N}{2})^2)$ $\quad$ $O((\frac{N}{2})^2)$

$2\,O((\frac{N}{2})^2)$

for

$$D_{\frac{N}{2}\times\frac{N}{2}} = \begin{bmatrix} \omega_N^{-0} & 0 & 0 & 0 \\ 0 & \omega_N^{-1} & 0 & 0 \\ 0 & 0 & \omega_N^{-2} & 0 \\ 0 & 0 & 0 & \omega_N^{-3} \end{bmatrix}$$

$\omega_N = e^{\frac{2\pi i}{N}}$ $\quad N=8:\ e^{\frac{\pi i}{4}}$

$\omega_N^0 = 1$

$\omega_N^{-1} = e^{-\frac{i\pi}{4}}$

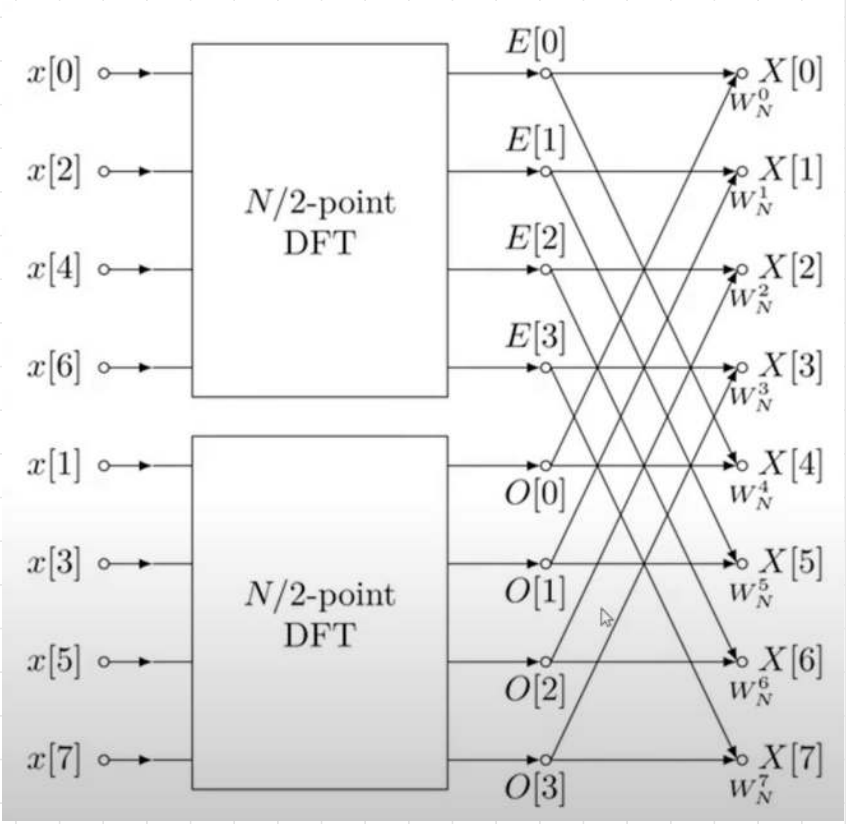
$\omega_N^{-2} = -i$

$\omega_N^{-3} = e^{-\frac{3i\pi}{4}}$

$$F_{\frac{N}{2}\times\frac{N}{2}} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & \omega_{\frac{N}{2}}^{-1\cdot1} & \omega_{\frac{N}{2}}^{-2\cdot1} & \omega_{\frac{N}{2}}^{-3\cdot1} \\ 1 & \omega_{\frac{N}{2}}^{-1\cdot2} & \omega_{\frac{N}{2}}^{-2\cdot2} & \omega_{\frac{N}{2}}^{-3\cdot2} \\ 1 & \omega_{\frac{N}{2}}^{-1\cdot3} & \omega_{\frac{N}{2}}^{-2\cdot3} & \omega_{\frac{N}{2}}^{-3\cdot3} \end{bmatrix}$$

$\omega_{\frac{N}{2}} = e^{\frac{2\pi i}{\frac{N}{2}}}$, $\quad N=8:\ e^{\frac{\pi i}{2}}$

FFT Butterfly diagram



as such, the FFT approach is $O(N \log_2 N)$

Periodogram estimate of the power spectrum

if the Fourier transform is defined by

$$\tilde{X}_i(k) = \sum_{n=0}^{N-1} f_n w_N^{-nk}, \quad k = 0, 1, \ldots, N-1, \quad w_N = e^{\frac{2\pi i k}{N}}$$

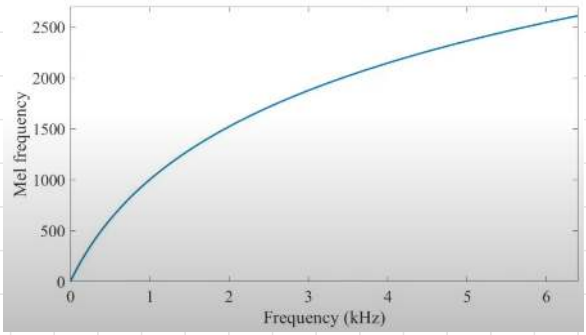
then $$P_i(k) = \frac{1}{N} \left| \tilde{X}_i(k) \right|^2$$

- we take absolute value to get rid of complex numbers we get rid of redundant complex conjugate symmetry
- it is an estimate of the power spectral density - how the power is distributed across different frequency components

so we get $\frac{\text{FFT size}}{2} + 1$ coefficients

Mel-scale

- for perceptual relevance of pitch



- logarithmic scale
- 1000 Hz = 1000 Mel

$$m = 2595 \cdot \log_{10}\left(1 + \frac{f}{700}\right)$$

$$f = 700\left(10^{m/2595} - 1\right)$$

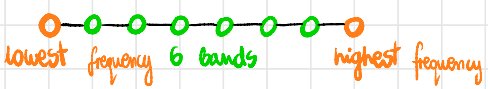
Mel bands

- a hyperparameter (usually between 20 and 40)
  26 is standard

constructing Mel filter banks

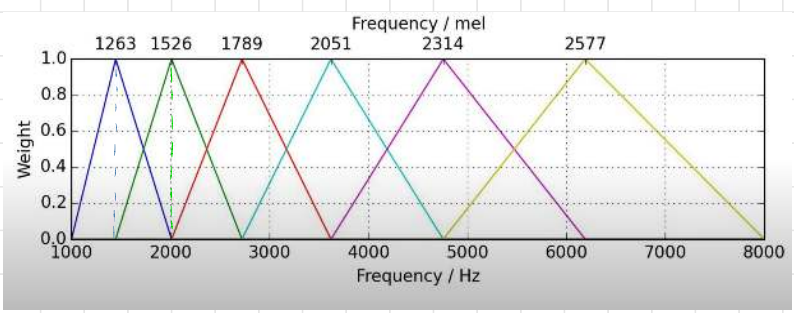
1. Take the lowest and highest frequency that we want to consider and convert to Mel (e.g., 300 - 8000 Hz)
   for speech upper f is limited to 4000 Hz
2. Create # bands equally spaced points

e.g., for Mel bands number 6



3. Conver these points back to Hertz
4. Round to nearest frequency bin $bin = \lfloor \frac{nfft+1}{\text{sampling rate}} \cdot f \rfloor$

   number of FFT coefficients
5. Create triangular filters - for each band construct the triangular response



triangular filter response $H_i(k) = \begin{cases} 0 & \text{if } k < bin_{i-1} \\ \frac{k - bin_{i-1}}{bin_i - bin_{i-1}} & \text{if } bin_{i-1} \leq k < bin_i \\ \frac{bin_{i+1} - k}{bin_{i+1} - bin_i} & \text{if } bin_i \leq k < bin_{i+1} \\ 0 & \text{if } k \geq bin_{i+1} \end{cases}$

the shape of Mel filter bank $T = (\#\ bands, \frac{FFT\ size}{2} + 1)$

the shape of the power spectrum $P = (\frac{FFT\ size}{2} + 1, \#\ frames)$

Mel spectogram = TP of a shape $\#\ bands \times \#\ frames$

e.g.
bands = 26
FFT size = 512

the filter bank is in the form of 26 vectors of length 257; each vector is mostly zeroes, but is non-zero for a certain section of the spectrum

example $sr = 16\,kHz$, $nfft = 512$

- lower bound: 300 Hz is 401.25 Mels
- upper bound: 8000 Hz is 2834.99 Mels
- we will choose number of bands = 10

we need 10 additional points spaced out equally between 401.25 & 2834.99

m(i) = 401.25, 622.50, 843.75, 1065.00, 1286.25, 1507.50, 1728.74, 1949.99, 2171.24, 2392.49, 2613.74, 2834.99

- convert back to Hz

h(i) = 300, 517.33, 781.90, 1103.97, 1496.04, 1973.32, 2554.33, 3261.62, 4122.63, 5170.76, 6446.70, 8000

- round those frequencies to the nearest FFT bin

e.g. $\left\lfloor \frac{513 \cdot 300}{16\,000} \right\rfloor = 9$

f(i) = 9, 16, 25, 35, 47, 63, 81, 104, 132, 165, 206, 256

- calculate Mel-filterbanks to obtain matrix $T_{10 \times 257}$
- obtain Mel spectogram $E = T \cdot P$ $E_i(l)$ – amount of energy in filter bank i at frame l

Extracting Mel Frequency Cepstral Coefficients

suppose we had $T \in \mathbb{R}^{26 \times 257}$ filter bank

and $P \in \mathbb{R}^{257 \times 350}$ power spectrum

we obtained $E \in \mathbb{R}^{26 \times 350}$ Mel spectogram

① we take the log of the spectogram

when working with power, not amplitude, take $10 \cdot \log$

$$\log(E) \in \mathbb{R}^{26 \times 350}$$

② Apply Discrete Cosine Transform (DCT) to the log-Mel coefficients

$$C_k = \sum_{n=0}^{N-1} e_n \cos\left[\frac{\pi}{N}(n + 0.5)k\right], \text{ where}$$

C_k - the k^{th} Mel cepstrum coefficient

e_n - the n^{th} log Mel energy

N - number of Mel bands

$k = 0, 1, \dots, N-1$

we get 26 MFCCs for each of the 350 frames

a simple example for one frame:

$N = 4$ bands $\quad e = [2, 1, 0, -1]$

$$c_0 = \sum_{n=0}^{3} x_n \cos\left(\frac{\pi}{4}(n+0.5)\cdot 0\right) = \sum_{n=0}^{3} x_n \cos(0) =$$

$$= \sum_{n=0}^{3} x_n = 2 + 1 + 0 + (-1) = 2$$

$$c_1 = \sum_{n=0}^{3} x_n \cos\left(\frac{\pi}{4}(n+0.5)\cdot 1\right)$$

$n=0$: $\cos\left(\frac{\pi}{8}\right) \approx 0.9239$ $\quad n=1$: $\cos\left(\frac{3\pi}{8}\right) \approx 0.3827$

$n=2$: $\cos\left(\frac{5\pi}{8}\right) \approx -0.3827$ $\quad n=3$: $\cos\left(\frac{7\pi}{8}\right) \approx -0.9239$

$$c_1 = 2(0.9239) + 1(0.3827) + 0(-0.3827) + (-1)(-0.9239) = 3.1544$$

$c_2 = 0$ $\quad c_3 = 0.7654$

$$DCT = [c_0, c_1, c_2, c_3] = [2, 3.15, 0, 0.765]$$

- the first coefficient c_0 captures the average energy
- latter coefficients capture finer details
- we usually keep only the first few coefficients (e.g., the first 12 for 26 bands)
- DCT decorrelates the Mel energies - makes the features independent of each other

note: you can also calculate delta-deltas to capture acceleration

Delta MFCCs

- an estimate of the first derivative of MFCCs over time
- they measure how quickly each MFCC coefficient is changing from one frame to the next

$$\Delta c(t) = \frac{\sum_{n=1}^{N} n\,(c(t+n) - c(t-n))}{2\sum_{n=1}^{N} n^2},$$ where

N - how many frames we look forward / backward (usually 2)

$c(t+n)$ - MFCC value n frames after time t

$c(t-n)$ - MFCC value n frames before time t

simple example for a single coefficient across 5 frames

(for more coefficients simply compute deltas for each of them separately)

frame t	0	1	2	3	4
$c(t)$	2	4	6	8	10

for $N = 1$

- frame 0: cannot compute as frame -1 doesn't exist

  padding strategy: assume missing frame is the same as the closest available frame

- frame 1: $\Delta c(1) = \frac{c(2) - c(0)}{2} = \frac{6-2}{2} = 2$
- frame 2: $\Delta c(2) = 2$
- frame 3: $\Delta c(3) = 2$
- frame 4: analogously to frame 1

frame t	0	1	2	3	4
$c(t)$	2	4	6	8	10
$\Delta c(t)$	2	2	2	2	2

then we concatenate MFCCs and deltas into one feature vector