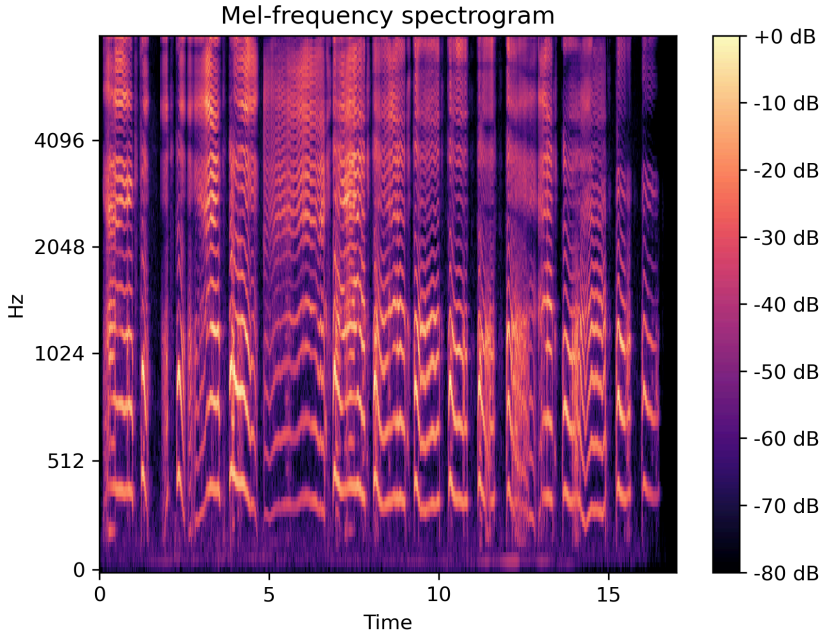
Problem 1:

(a) Explain the audio content shown on the spectrogram.

- Harmonic Structure: The repeating vertical patterns represent the characteristic of baby crying.
- Variability in Intensity: The spectrogram shows significant variability in intensity over time. This variability corresponds to the volume of the crying changing over time—louder cries appear brighter and quieter moments appear darker.
- Temporal Features: The spacing between the bright vertical bands may provide information about the rhythm of the crying.



(b) Please derive the IF of x2.

(2000*T/pi, (50.0*T*cos(2.5*T**2) + 2000)/(2*pi))

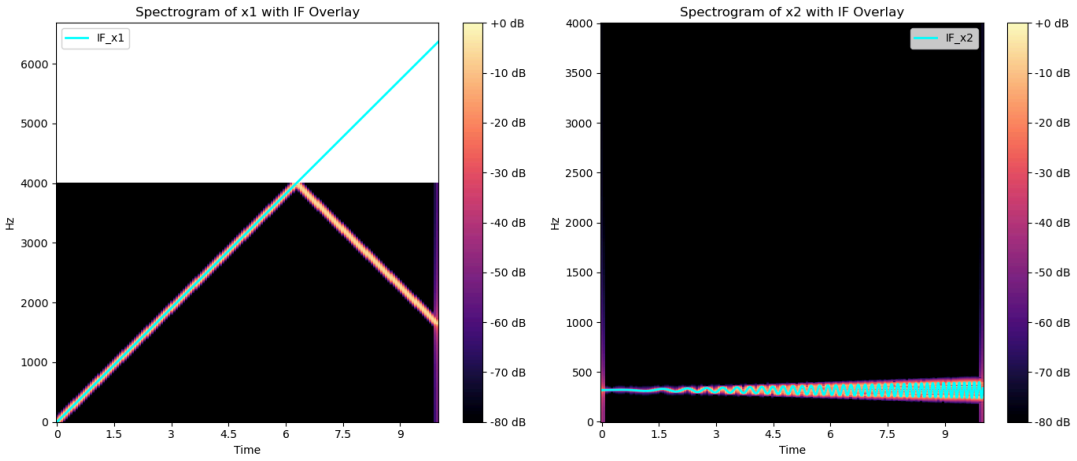
(c) Describe sound:

The pitch of x1.wav gets higher over time but suddenly decreases , whereas the pitch of x2.wav will have a more complex sound with a pitch that not only gets higher over time but also fluctuates, giving a warbling effect. The reason why the pitch of x1.wav suddenly decreases may be 'Aliasing or Folding' : If the signal frequency exceeds the Nyquist frequency at some point , aliasing could occur, where higher frequencies are misinterpreted as lower frequencies.

(d) DIscuss two graph:

- Left spectrogram: the IF curve (cyan line) of x1 increases linearly over time, which is consistent with the mathematical description of x1(t). The brightest line in the spectrogram closely follows this IF curve before the 6th second.

- Right spectrogram: the IF curve (cyan line) of x2 contains fluctuations due to the 10*sin(2.5* t**2) in x2.



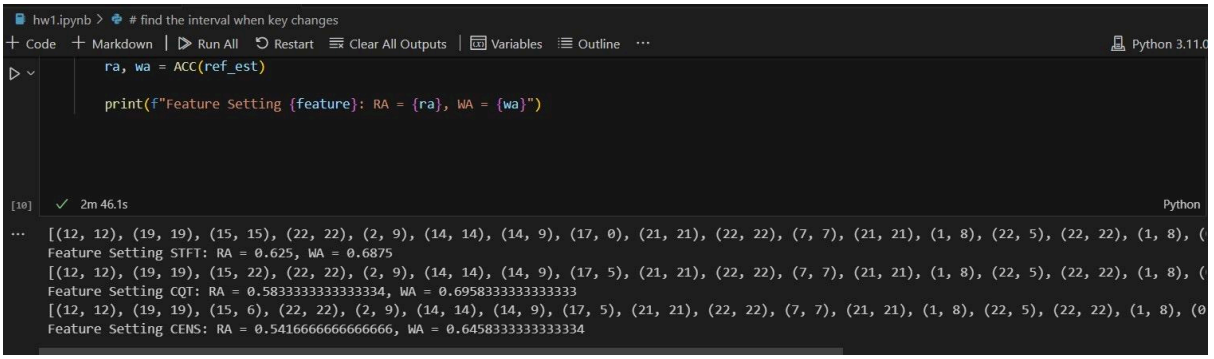
Problem 2:

(a) The Krumhansl-Schmuckler key-finding algorithm for global key detection.

I use early fusion approaches to implement. My experiment results suggest that STFT-based chroma features might be the most suitable for global key detection in this particular dataset, since it has the highest weighted accuracy.

	STFT	CQT	CENS
RA	0.083	0.125	0.083
WA	0.15	0.192	0.137

Note that I run the same code with same input on Window PC and it gives a different outcome:



	STFT	CQT	CENS
RA	0.625	0.583	0.542
WA	0.6875	0.696	0.646

(b) Local Key Detection that outputs the key of the music every 0.1 second.

First, I use the Short time fourier transform method to find the chroma vector, and then detect the key every 0.1 second. This serves as the estimated keys list.

For the reference key list, I use the Schubert_D911-01_SC06.csv and change each time interval in the form of 0.1 second. Since its starting time begins with 0.24, I drop the first three rows in the estimated keys to let its start time begin with 0.3. Also note that I only analyze the song before 50 seconds.

Also, I've tried the parameter: n_fft = 2048, 4096 and 10000, when the n_fft is reduced from 10000 to 2048, the frequency resolution becomes coarser, potentially leading to less accurate representations of the harmonics.

	n_fft = 2048	n_fft = 4096	n_fft = 10000
RA	0.291	0.345	0.414
WA	0.363	0.442	0.511

(c) Apply segmentation measures to local key detection problem

I analyze the key every 1 second, and then use a for loop to find the time interval when the key changes. This serves as the estimated key change interval. But it seems that there is a significant discrepancy between the predicted segments and the ground truth interval.

Over-segmentation	0.11371896753526778
Under-segmentation	0.9838271604938272
Average-segmentation	0.11371896753526778