Analiza în domeniul Timp și în domeniul Frecvență

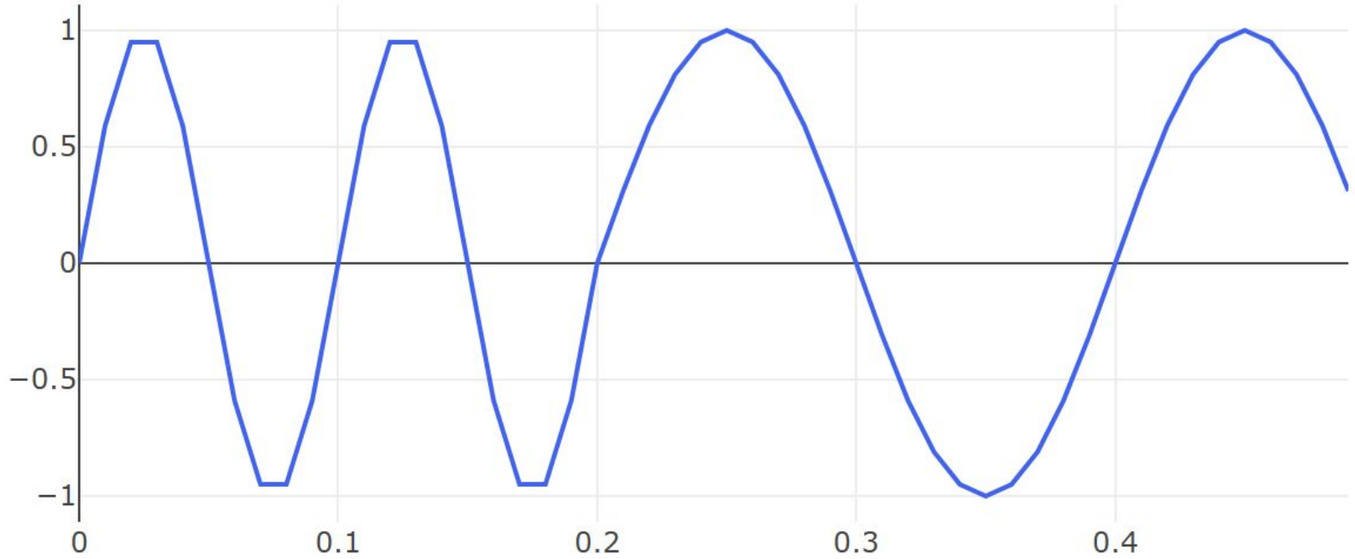
Un semnal digital, *x[n]*, reprezintă amplitudinea unei unde sonore analogice la puncte discrete în timp, rezultând într-o secvență de numere. Aceasta este cunoscută sub denumirea de **reprezentare în domeniul timp** (*time-domain representation*), răspunzând direct la întrebarea: „Care a fost amplitudinea semnalului la un moment specific?".

Deși esențială, această perspectivă prezintă **limitări semnificative** pentru recunoașterea vorbirii. În secțiunile următoare, vom explora cum putem depăși aceste limitări prin analiza în domeniul frecvență și prin combinarea celor două perspective.

Domeniul Timp

Când reprezentăm grafic amplitudinea unui semnal audio digital în funcție de timp, obținem o **formă de undă** (*waveform*). Această vizualizare este utilă pentru a observa caracteristici generale precum **intensitatea sonoră** (amplitudine mare) și **momentele de tăcere** (amplitudine aproape de zero).

Graficul de mai jos arată amplitudinea unui semnal pe parcursul unei fracțiuni de secundă.



Ce putem vedea

- Porțiunile mai puternice ale semnalului
- Zonele de liniște sau tăcere
- Dinamica generală a amplitudinii

Ce NU putem vedea

- Diferența dintre sunete precum „s" și „ș"
- Conținutul spectral (de frecvență)
- Blocurile de construcție ale vorbirii

Concluzie

Sunetele distincte sunt caracterizate de conținutul lor spectral, nu doar de amplitudine. Un model de **machine learning** are nevoie de o reprezentare mult mai bogată pentru a învăța blocurile de construcție ale vorbirii.

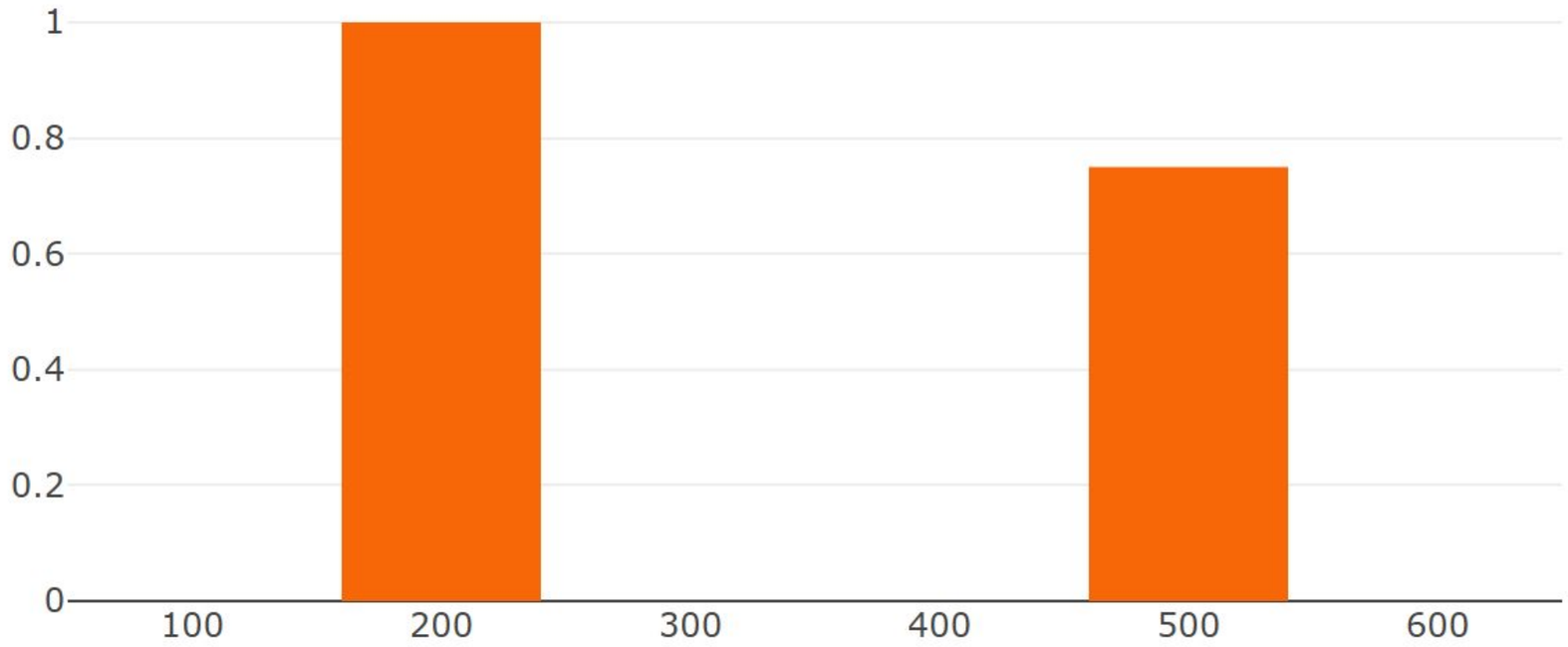
Domeniul Frecvență

Domeniul frecvență oferă o perspectivă fundamental diferită: în loc de amplitudine pe timp, reprezentăm **energia semnalului în raport cu frecvența**. Răspunde la întrebarea: „Ce frecvențe sunt prezente și cât de dominante sunt?".

Pentru a trece din domeniul timp în domeniul frecvență, utilizăm un instrument matematic numit ***Transformata Fourier***. Pentru semnalele digitale, utilizăm echivalentul acesteia, *Transformata Fourier Discretă* (DFT - Discrete Fourier Transform), care este cel mai adesea calculată folosind un algoritm eficient numit T*ransformata Fourier Rapidă* (FFT - Fast Fourier Transform).

- Transformata Fourier descompune un semnal în unde sinusoidale și cosinusoidale de frecvențe diferite care, însumate, reconstruiesc semnalul original. Pentru analiza vorbirii, suntem interesați în primul rând de **magnitudine** – cât de mult din fiecare frecvență este prezentă.

Dacă am analiza un semnal compus din două unde sinusoidale pure, una la 200 Hz și alta la 500 Hz, spectrul său de frecvență ar arăta astfel:



Această reprezentare este mult mai informativă. Putem vedea clar frecvențele constitutive care alcătuiesc semnalul, un aspect care era ascuns în forma de undă din domeniul timp.

Provocarea semnalelor nestaționare

> Vorbirea este un **semnal nestaționar/dinamic** -- proprietățile sale de frecvență se modifică în timp.

Când pronunțați cuvântul „speech", frecvențele care formează sunetul „s" sunt foarte diferite de cele care formează sunetul „ee". Dacă am aplica o singură transformată FFT pe întregul clip audio, am obține un spectru care **mediază** toate frecvențele sunetelor „s", „p" și „ee".

Ce obținem	Ce pierdem	De ce contează
Frecvențele prezente la nivel global în cuvânt	Informația referitoare la **momentul** în care frecvențele au apărut	Informația temporală este critică ex: „cats" vs. „stack"

Combinarea Timpului și a Frecvenței: STFT

Soluția este să analizăm semnalul în **segmente mici**, cvasi-staționare. Aceasta este sarcina **Transformatei Fourier pe Termen Scurt (STFT)**.

01

Segmentarea (Framing)

Semnalul audio este tăiat în cadre scurte, suprapuse, de 20-30 ms — suficient pentru a capta un singur eveniment fonetic.

02

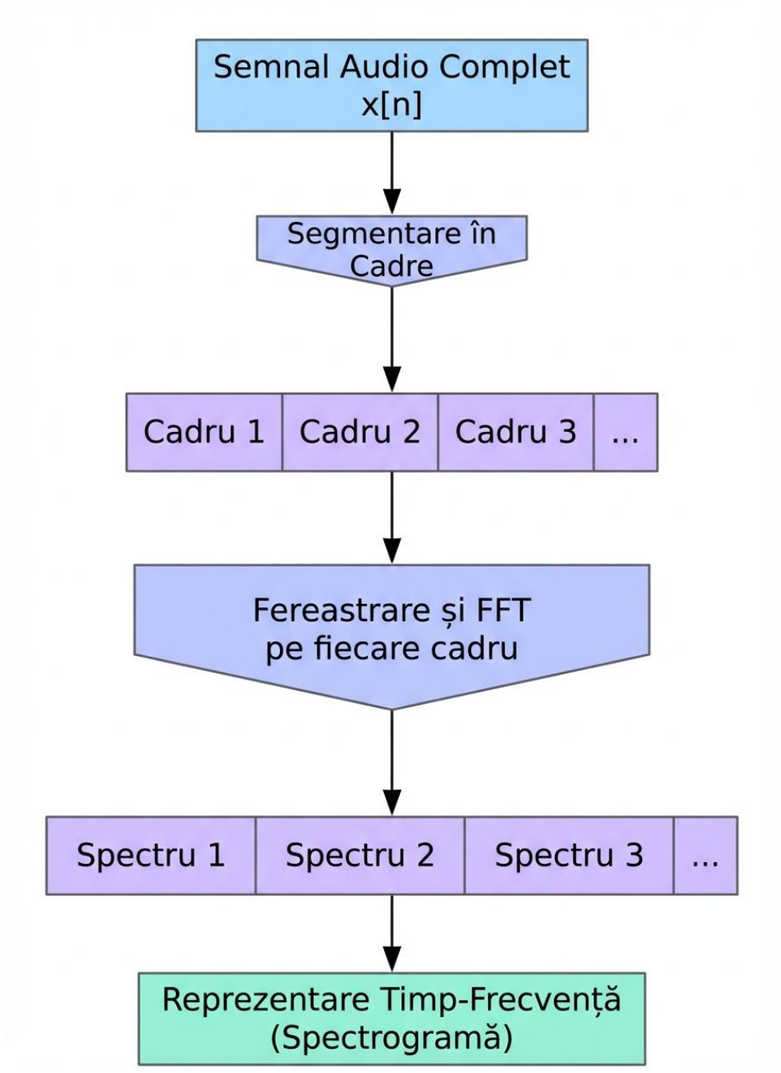
Ferestruirea (Windowing)

Fiecare cadru este înmulțit cu o funcție Hann sau Hamming, atenuând semnalul la margini pentru a reduce artefactele spectrale.

03

Aplicarea FFT

Transformata Fourier Rapidă este calculată pe fiecare cadru ferestruit pentru a obține spectrul de frecvență.



Introducere în Spectrograme

Vizualizarea Vorbirii

O formă de undă ascunde informația de frecvență. O Transformată Fourier standard elimină informația temporală. Vorbirea necesită **ambele** dimensiuni simultan.

Forma de Undă	Transformata Fourier	Spectrograma
Amplitudine vs. Timp	Energie vs. Frecvență	Frecvență × Timp × Energie
✅ Când | ❌ Ce frecvențe	❌ Când | ✅ Ce frecvențe	✅ Când | ✅ Ce frecvențe

Spectrograma este o reprezentare vizuală care ilustrează modul în care conținutul de frecvență al unui semnal evoluează în timp unul dintre instrumentele fundamentale în procesarea vorbirii.

Transformata Fourier pe Termen Scurt (STFT)

O spectrogramă este generată prin **STFT**: semnalul este segmentat în cadre scurte, suprapuse, de 20-30 ms. Pentru fiecare cadru, presupunem că semnalul este staționar.

1

Ferestruirea (Windowing)

Semnalul audio este divizat în cadre mici, suprapuse. O funcție Hann sau Hamming este aplicată fiecărui cadru pentru a reduce dispersia spectrală (*spectral leakage*).

2

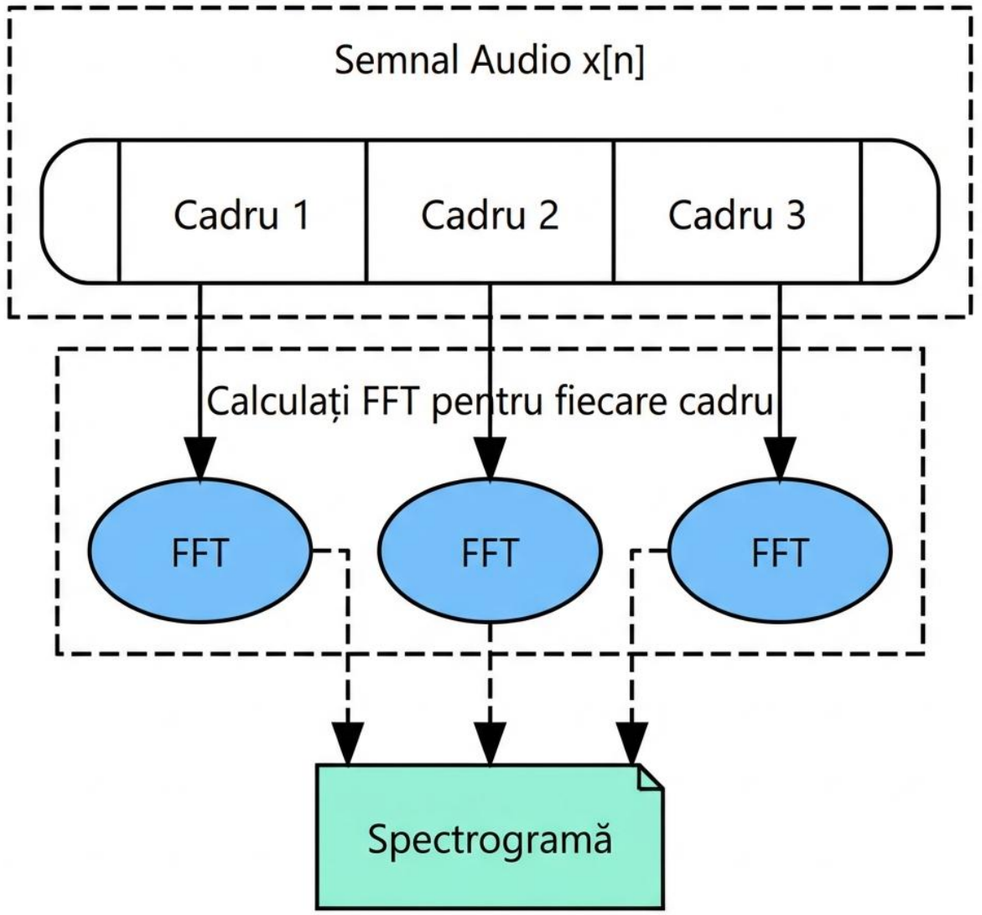
Transformata Fourier (FFT)

FFT este calculată pentru fiecare cadru ferestruit, convertind semnalul din domeniul timp în domeniul frecvență — indicând puterea fiecărei frecvențe prezente.

3

Alăturarea (Stacking)

Spectrele de frecvență din toate cadrele sunt dispuse unul lângă altul, formând o imagine 2D.

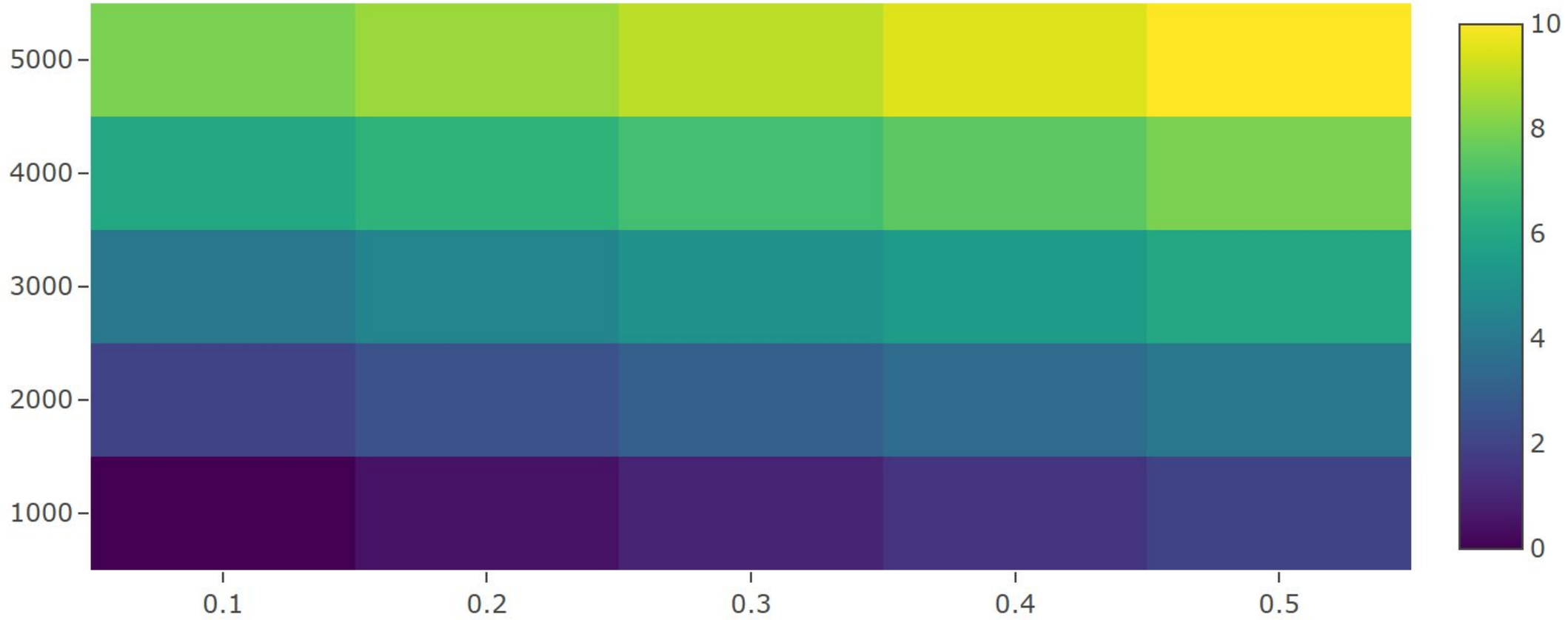


Procesul de creare a unei spectrograme prin STFT

Rezultatul este o reprezentare tridimensională: axa *x* = timp, axa *y* = frecvență, iar culoarea/intensitatea = amplitudinea sau puterea unei frecvențe la un moment dat.

Interpretarea Vorbirii într-o Spectrogramă

O spectrogramă transformă un semnal audio într-o imagine, permițându-ne să vizualizăm tiparele distincte ale vorbirii. Să analizăm componentele cuvântului „speech":



Vocalele -- Formanți

Barele orizontale întunecate pe spectrogramă sunt **formanții** — frecvențele de rezonanță ale tractului vocal. Tiparul și spațierea lor definesc diferitele sunete vocalice. Vocala /i/ din „speech" are un set specific de frecvențe ale formanților.

Fricativele -- Zgomot de înaltă frecvență

Sunetele /s/ şi /ch/ sunt consoane produse prin forțarea aerului printr-un canal îngust. Pe spectrogramă, apar ca un **nor difuz şi neregulat** de energie, dispersat pe frecvențe înalte.

Plozivele -- Explozii de energie

Consoanele /p/, /t/, /k/ sunt create prin blocarea fluxului de aer şi eliberarea sa bruscă. Generează o **explozie scurtă şi intensă** de energie pe spectrogramă.

> Formanții orizontali pentru vocale, norii de zgomot pentru fricative şi exploziile bruște pentru plozive oferă o **semnătură vizuală distinctă** pentru fiecare sunet.

Încărcarea și vizualizarea formelor de undă Audio

Vom scrie cod Python pentru a încărca un fișier audio și a vizualiza forma sa de undă și spectrograma. Vizualizarea datelor audio este un pas esențial pentru **depanare**, **înțelegerea setului de date** și construirea unei intuiții asupra modului în care diferite sunete apar pentru o mașină.

Competență Fundamentală

Exercițiu practic demonstrând conversia undei sonore analogice în semnal digital

Vizualizare Dual

Diferențele dintre reprezentările în domeniul timp și domeniul frecvență

Biblioteca Librosa

Standardul *de facto* pentru analiza audio în Python

Configurarea Mediului de Lucru

Înainte de a începe, asigurați-vă că aveți instalate bibliotecile necesare. **Librosa** se va ocupa de încărcarea și procesarea audio, iar **Matplotlib** oferă baza pentru vizualizare.

```
pip install librosa matplotlib numpy
```

Librosa

Încărcare, procesare și analiză audio

Matplotlib

Vizualizare grafică și generare de ploturi

NumPy

Operații numerice pe tablouri de date

Încărcarea unui Fișier Audio cu Librosa

Funcția `librosa.load()` returnează două elemente esențiale:

y — Seria Temporală

Un tablou NumPy reprezentând forma de undă digitală *x[n]*. Librosa convertește automat semnalul în **mono** și îl reeșantionează la **22.050 Hz**.

sr — Rata de Eșantionare

Câte eșantioane din tabloul *y* corespund unei secunde de audio. Implicit: 22.050 Hz.

```
import librosa

# Definiți calea către fișierul audio
audio_path = 'calea/catre/fisierul_tau_audio.wav'

# Încărcați fișierul audio
y, sr = librosa.load(audio_path)

print(f"Dimensiunea seriei temporale audio: {y.shape}")
print(f"Rata de eșantionare: {sr} Hz")
```

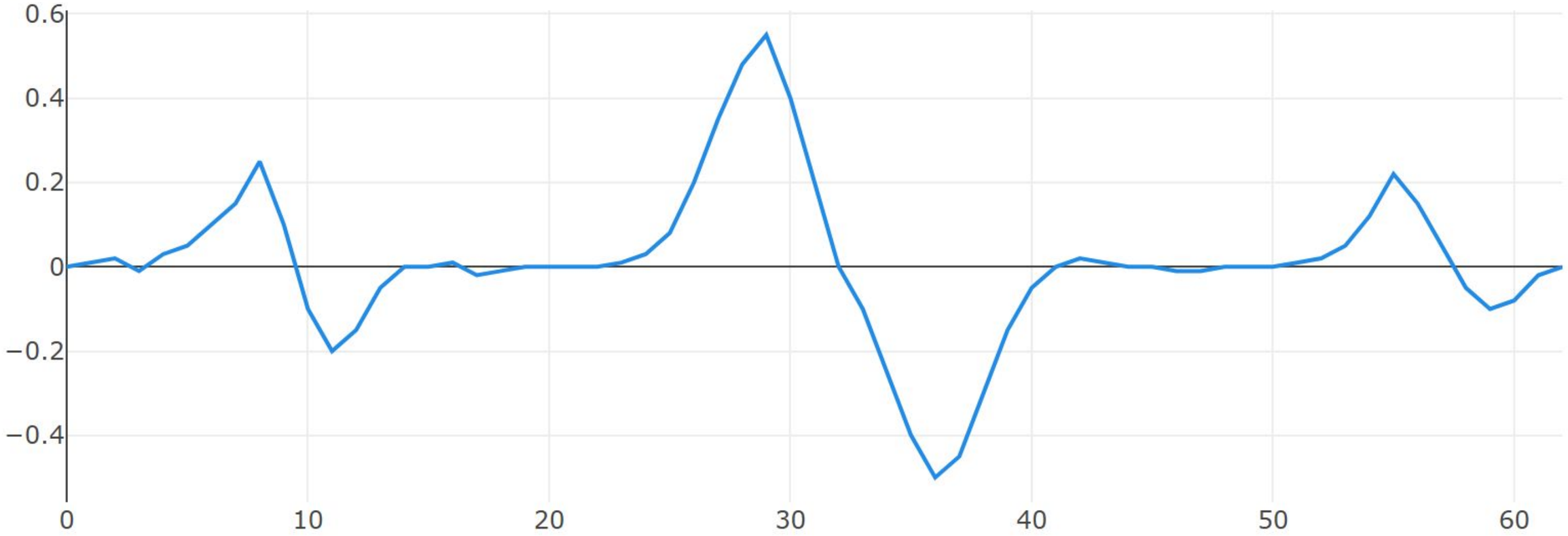
> Pentru un fișier audio de 5 secunde: 5 × 22.050 = **110.250 eșantioane**. Ieșirea va fi: `Dimensiunea seriei temporale audio: (110250,)` și `Rata de eșantionare: 22050 Hz`.

Vizualizarea formei de undă în domeniul Timp

Seria temporală *y* conţine amplitudinea audio la fiecare pas de timp discret. Graficul amplitudinii în funcţie de timp este **forma de undă** — ne arată pasajele puternice şi cele silenţioase.

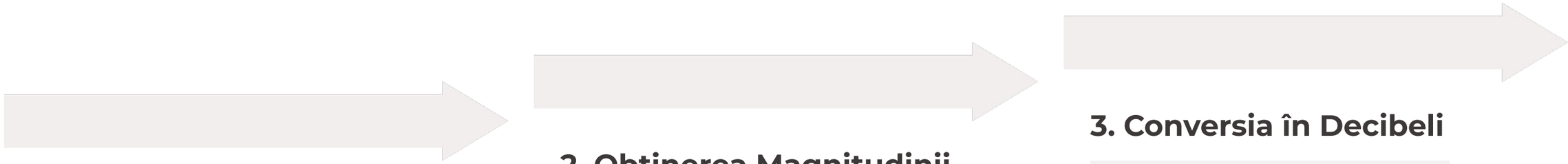
```
import numpy as np
import librosa.display
import matplotlib.pyplot as plt

# Presupunând că 'y' și 'sr' sunt încărcate
fig, ax = plt.subplots(figsize=(10, 4))
librosa.display.waveshow(y, sr=sr, ax=ax)
ax.set_title('Forma de Undă Audio')
ax.set_xlabel('Timp (s)')
ax.set_ylabel('Amplitudine')
plt.tight_layout()
plt.show()
```

Codul generează un grafic în care axa x reprezintă timpul, iar axa y reprezintă amplitudinea. Zonele cu amplitudine absolută ridicată corespund părților mai puternice ale sunetului, în timp ce zonele apropiate de zero sunt silențioase sau liniștite.



Generarea unei Spectrograme

Forma de undă nu ne spune nimic despre conținutul de frecvență. Pentru a vedea ce frecvențe sunt prezente în fiecare moment, calculăm o **spectrogramă** prin STFT.



1. Calcularea STFT

`librosa.stft(y)` returnează o matrice 2D cu valori complexe: rânduri = intervale de frecvență, coloane = cadre de timp.

2. Obținerea Magnitudinii

`np.abs(D)` extrage doar magnitudinea, eliminând faza — suficient pentru vizualizare și majoritatea modelelor ASR.

3. Conversia în Decibeli

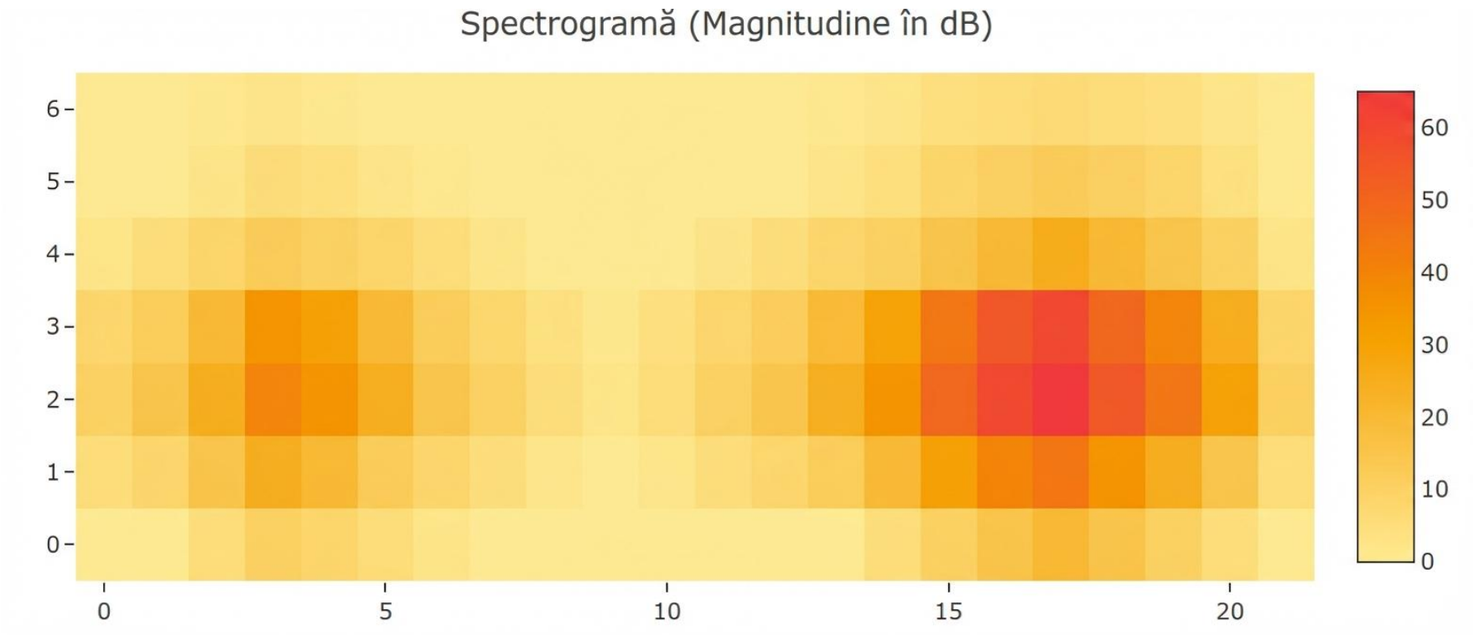
`librosa.amplitude_to_db()` comprimă gama dinamică pe o scală logaritmică, făcând vizualizarea mult mai informativă.

Codul pentru generarea Spectrogramei

```
import numpy as np

# Calculăm Transformata Fourier pe Termen Scurt (STFT)
D = librosa.stft(y)

# Convertim în spectrogramă de magnitudine
S_mag = np.abs(D)

# Convertim la scara de decibeli (dB)
S_db = librosa.amplitude_to_db(S_mag, ref=np.max)

# --- Reprezentarea grafică a spectrogramei ---
fig, ax = plt.subplots(figsize=(10, 4))
img = librosa.display.specshow(
 S_db, sr=sr,
 x_axis='time', y_axis='mel', ax=ax
)
fig.colorbar(img, ax=ax,
 format='%+2.0f dB',
 label='Magnitudine (dB)')
ax.set_title('Spectrogramă Mel')
plt.tight_layout()
plt.show()
```

Spectrograma, hartă termică a vorbirii

Graficul rezultat este o **hartă termică** (*heatmap*): axa *x* = timp, axa *y* = frecvență, iar intensitatea culorii în orice punct *(t, f)* indică energia frecvenței *f* la timpul *t*. Benzile orizontale de energie sunt caracteristice vorbirii și corespund **formanților**.



Ce am învățat

Încărcarea oricărui fișier audio și inspectarea structurii sale în domeniul timp și domeniul frecvență

De ce contează

Vizualizările sunt baza pentru depanare, înțelegerea datelor și construirea intuiției

Ce urmează

Tehnici automate de **feature extraction** — convertirea reprezentărilor vizuale în trăsături numerice pentru modele de deep learning