# Speech Emotion Recognition with Multi-Modal Analysis

MP4ファイルから音声を抽出し、感情分析・文字起こし・話者分析・音声メトリクス分析を行う Jupyter Notebook版です。

# **◎** 主要機能

- 🧏 感情分析(7種類の感情)
- P OpenAl Whisperによる音声文字起こし
- 點 話者分析(pyannote.audio)
- セグメント自動分割(無音・時間・話者ベース)

# ● 必要なライブラリのインポート

```
In [1]: import os
        import warnings
        from typing import Dict, List, Tuple, Optional
        import pandas as pd
        from IPython.display import display, HTML, Audio
        import matplotlib.pyplot as plt
        import seaborn as sns
        from rich.progress import Progress
        from rich.console import Console
        # Suppress warnings
        warnings.filterwarnings("ignore", message="Passing `gradient_checkpointing` to a
        warnings.filterwarnings("ignore", message="Some weights of.*were not initialized
        warnings.filterwarnings("ignore", message="You should probably TRAIN this model"
        warnings.filterwarnings("ignore", message="Conversion of an array with ndim > 0
        # Import our custom modules
        from audio extractor import AudioExtractor
        from audio_segmenter import AudioSegmenter
        from audio metrics import AudioMetricsAnalyzer
        from emotion analyzer import EmotionAnalyzer
        from result_display import ResultDisplay
        from transcription import TranscriptionAnalyzer
        from speaker_diarization import SpeakerDiarization
        print(" ✓ ライブラリの読み込み完了")
```

#### ライブラリの読み込み完了

# ❖ 分析設定

以下のセルで分析したいファイルと設定を指定してください。

```
In [6]: # ======= 分析設定 =======
       # ファイルパス設定
       INPUT_FILE = "assets/pantene_marketing.mp4" # F 分析したいMP4ファイルのパスを指述
       OUTPUT AUDIO = None # 音声ファイルを保存したい場合はパスを指定(例: "extra
       # 分析タイプ設定(True/Falseで有効/無効を切り替え)
       ENABLE_EMOTION_ANALYSIS = True # 🦻 感情分析
       ENABLE_TRANSCRIPTION = True
                                       # 🍃 文字起こし
        ENABLE_SPEAKER_DIARIZATION = True # # 話者分析
       ENABLE_AUDIO_METRICS = True # 🎜 音声メトリクス
       # セグメント設定
       SEGMENT_MODE = True # にノヘントラボルを見かれる。
SPEAKER_SEGMENTS = True # 話者変更時にセグメント分割

MAX_SEGMENT_DURATION = 15.0 # 最大セグメント長(秒)

MIN_SEGMENT_DURATION = 3.0 # 最小セグメント長(秒)

FORCE TIME_SPLIT = False # 時間ベース分割を強制
       SEGMENT MODE = True
                                     # セグメント分析を有効にする
       # モデル設定
       EMOTION_MODEL = "japanese" # 感情分析モデル (whisper/sensevoice/japanese)
WHISPER_MODEL_SIZE = "base" # Whisperモデルサイズ (tiny/base/small/medium/L
TRANSCRIPTION_LANGUAGE = "auto" # 文字起こし言語 (auto/ja/en/etc)
       # GPU設定
       USE GPU = True
                                       # GPUを使用する
       # HuggingFace設定 (話者分析に必要)
                                       # HuggingFaceトークンを設定(環境変数からも取得可
       HF TOKEN = None
       if HF_TOKEN is None:
           HF_TOKEN = os.getenv('HF_TOKEN')
       # 表示設定
       SHOW_DETAILED_EMOTIONS = True # 詳細な感情スコア表示
       SHOW_DETAILED_METRICS = True # 詳細な音声メトリクス表示
SHOW_DETAILED_SPEAKERS = True # 詳細な話者分析表示
       print(f" > 入力ファイル: {INPUT FILE}")
       print(f" 場 感情分析: {' ☑' if ENABLE_EMOTION_ANALYSIS else 'Х'}")
       print(f"點 話者分析: {'☑' if ENABLE_SPEAKER_DIARIZATION else 'X'}")
       print(f" □ セグメント分析: {' \ ' if SEGMENT MODE else ' \ '}")
       ☆ 設定完了
       📂 入力ファイル: assets/pantene marketing.mp4
       🦆 感情分析: 🔽
       🍃 文字起こし: 🛂
      點 話者分析: ☑

₁ 音声メトリクス: ☑
       セグメント分析:
```

### ≦ 音声抽出

```
In [3]: # 入力ファイルの存在確認

if not os.path.exists(INPUT_FILE):
    raise FileNotFoundError(f"入力ファイルが見つかりません: {INPUT_FILE}")
```

```
# 最低1つの分析タイプが有効であることを確認
if not any([ENABLE_EMOTION_ANALYSIS, ENABLE_TRANSCRIPTION, ENABLE_SPEAKER_DIARIZ
    raise ValueError("最低1つの分析タイプを有効にしてください")
# 音声抽出
print(" 音 音声抽出を開始...")
extractor = AudioExtractor()
audio_path = extractor.extract_audio_from_mp4(INPUT_FILE, OUTPUT_AUDIO)
if OUTPUT_AUDIO:
    print(f" ☑ 音声ファイルを保存: {audio_path}")
else:
    print(f"☑ 音声抽出完了: {audio_path}")
# 抽出した音声を再生可能な形で表示
if os.path.exists(audio_path):
    print("\n ● 抽出した音声:")
    display(Audio(audio_path))
≦ 音声抽出を開始...
☑ 音声抽出完了: /tmp/tmpfn1llbgw.wav
● 抽出した音声:
```

# 🧠 分析器の初期化

0:00 / 3:24

```
In [7]: print("● 分析器を初期化中...")
       # 結果表示器
       display_handler = ResultDisplay()
       console = Console()
       # 感情分析器
       emotion_analyzer = None
       if ENABLE EMOTION ANALYSIS:
           print(" 場 感情分析器を初期化...")
           emotion analyzer = EmotionAnalyzer(model type=EMOTION MODEL, use gpu=USE GPU
       # 音声メトリクス分析器
       metrics_analyzer = None
       if ENABLE AUDIO METRICS:
          metrics_analyzer = AudioMetricsAnalyzer()
       # 文字起こし分析器
       transcription analyzer = None
       if ENABLE TRANSCRIPTION:
           print(" 文字起こし分析器を初期化...")
           transcription_analyzer = TranscriptionAnalyzer(
              model_size=WHISPER_MODEL_SIZE,
              use_gpu=USE_GPU,
              language=TRANSCRIPTION_LANGUAGE
           )
```

```
# 話者分析器
 speaker_analyzer = None
 if ENABLE_SPEAKER_DIARIZATION:
     print("點 話者分析器を初期化...")
     speaker_analyzer = SpeakerDiarization(
         use gpu=USE GPU,
         hf_token=HF_TOKEN
 print(" / 分析器の初期化完了")
分析器を初期化中...
🍢 感情分析器を初期化...
Using device: cuda
Loading model: Wav2Vec2 (Japanese-specific)
Model ID: Bagus/wav2vec2-xlsr-japanese-speech-emotion-recognition
Some weights of Wav2Vec2ForSequenceClassification were not initialized from the m
odel checkpoint at Bagus/wav2vec2-xlsr-japanese-speech-emotion-recognition and ar
e newly initialized: ['classifier.bias', 'classifier.weight', 'projector.bias',
'projector.weight']
You should probably TRAIN this model on a down-stream task to be able to use it f
or predictions and inference.
```

- 音声メトリクス分析器を初期化...
- ▶ 文字起こし分析器を初期化...

Loading Whisper base model on cuda...

- 點 話者分析器を初期化...
- Loading speaker diarization pipeline on cuda...
- ☑ Successfully loaded pyannote/speaker-diarization-3.1
- 分析器の初期化完了

### ・ 音声セグメント化

```
In [8]: segments = None
       diarization result = None
       if SEGMENT MODE:
          print(" 🔄 音声セグメント化を開始...")
          # セグメンター初期化
          segmenter = AudioSegmenter(
              min silence len=int(1.0 * 1000), # 1秒 -> ms
              silence_thresh=-40,
              min segment len=int(MIN SEGMENT DURATION * 1000),
              max segment len=int(MAX SEGMENT DURATION * 1000),
              force time split=FORCE TIME SPLIT
          )
          # 話者ベースのセグメント分割の場合、事前に話者分析を実行
          if SPEAKER_SEGMENTS and speaker_analyzer:
              try:
                 diarization result = speaker analyzer.diarize audio(audio path)
                 print(f"☑ 話者分析完了: {diarization_result.get('num_speakers', 0)}.
              except Exception as e:
                 print(f"▲ 話者分析失敗: {e}")
                 print(" 🔄 無音/時間ベースのセグメント分割にフォールバック")
                 diarization result = None
          # セグメント分割実行
```

```
segments = segmenter.segment_audio(audio_path, diarization_result)
                 if not segments:
                                 raise RuntimeError("セグメントが生成されませんでした。音声が短すぎるか無音のF
                 print(f" ✓ セグメント分割完了: {len(segments)}個のセグメントを生成")
                 # セグメント情報を表示
                 segment_info = []
                 for i, (_, start_time, end_time) in enumerate(segments):
                                  duration = end_time - start_time
                                  segment info.append({
                                                   "t^2 + t^2 + t^2
                                                   "開始時間": f"{start_time:.1f}s",
                                                  "終了時間": f"{end_time:.1f}s",
                                                  "長さ": f"{duration:.1f}s"
                                 })
                 df segments = pd.DataFrame(segment info)
                 print("\n tグメント情報:")
                display(df_segments)
else:
                 print(" > フルオーディオ分析モード")
```

- 音声セグメント化を開始...
- 話者分析を実行(セグメント分割用)...

/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si te-packages/pyannote/audio/utils/reproducibility.py:74: ReproducibilityWarning: T ensorFloat-32 (TF32) has been disabled as it might lead to reproducibility issues and lower accuracy.

It can be re-enabled by calling

- >>> import torch
- >>> torch.backends.cuda.matmul.allow\_tf32 = True
- >>> torch.backends.cudnn.allow\_tf32 = True

See https://github.com/pyannote/pyannote-audio/issues/1370 for more details.

warnings.warn(

/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si te-packages/pyannote/audio/models/blocks/pooling.py:104: UserWarning: std(): degr ees of freedom is <= 0. Correction should be strictly less than the reduction fac tor (input numel divided by output numel). (Triggered internally at /pytorch/ate n/src/ATen/native/ReduceOps.cpp:1839.)

std = sequences.std(dim=-1, correction=1)

☑ 話者分析完了: 2人の話者を検出

Using speaker-based segmentation (72 speaker segments found)

- ☑ セグメント分割完了:20個のセグメントを生成
- セグメント情報:

	セグメント	開始時間	終了時間	長さ
0	1	1.7s	5.7s	4.1s
1	2	6.2s	10.8s	4.5s
2	3	11.2s	15.7s	4.5s
3	4	17.8s	24.4s	6.6s
4	5	42.4s	46.0s	3.6s
5	6	44.5s	47.5s	3.0s
6	7	48.6s	54.6s	6.0s
7	8	55.1s	60.5s	5.5s
8	9	63.4s	66.4s	3.0s
9	10	68.7s	73.0s	4.4s
10	11	74.4s	80.3s	5.8s
11	12	83.7s	93.7s	10.0s
12	13	94.3s	98.3s	4.0s
13	14	127.7s	135.3s	7.5s
14	15	139.1s	144.6s	5.6s
15	16	144.8s	148.9s	4.1s
16	17	149.3s	154.8s	5.5s
17	18	158.0s	161.2s	3.2s
18	19	162.7s	166.8s	4.0s
19	20	166.8s	175.9s	9.1s

## 6 分析実行

```
In [9]: if SEGMENT_MODE and segments:
           print(" ** セグメント分析を実行中...")
           # 話者分析(セグメント用でまだ実行していない場合)
           if speaker_analyzer and not SPEAKER_SEGMENTS:
              try:
                  print(" 1 話者分析を実行...")
                  diarization_result = speaker_analyzer.diarize_audio(audio_path)
                  print(f"☑ 話者分析完了: {diarization_result.get('num_speakers', 0)}.
              except Exception as e:
                  print(f"▲ 話者分析失敗: {e}")
                  speaker_analyzer = None
                  diarization_result = None
           # 各セグメントの分析実行
           results = []
           metrics results = []
           transcription_results = []
```

```
speaker_results = []
   total_segments = len(segments)
   for i, (segment_path, start_time, end_time) in enumerate(segments):
       print(f"\n i セグメント {i+1}/{total segments} を分析中... ({start time:.
       try:
           # 感情分析
           if emotion_analyzer:
               emotion_scores = emotion_analyzer.analyze_audio(segment_path)
               results.append((start time, end time, emotion scores))
               dominant_emotion = max(emotion_scores.items(), key=lambda x: x[1
               print(f" 🤘 感情: {dominant_emotion}")
           # 音声メトリクス分析
           if metrics_analyzer:
               audio_metrics = metrics_analyzer.analyze_audio_metrics(segment_p
               metrics results.append((start time, end time, audio metrics))
               print(f" 『 ピッチ: {audio_metrics.get('pitch_mean', 0):.0f}Hz"
           # 文字起こし分析
           if transcription_analyzer:
               transcription = transcription_analyzer.transcribe_audio(segment_
               transcription_results.append((start_time, end_time, transcriptio
               preview = transcription.get('preview_text', '[空]')
               print(f" > テキスト: {preview}")
           # 話者分析
           if speaker analyzer and diarization result:
               speaker = speaker_analyzer.get_speaker_for_segment(
                   start_time, end_time, diarization_result
               formatted_speaker = speaker_analyzer.format_speaker_label(speake
               speaker results.append((start time, end time, formatted speaker)
               print(f" ** 話者: {formatted speaker}")
       except Exception as e:
           print(f" X エラー: {e}")
           continue
   # セグメントファイルのクリーンアップ
   segmenter.cleanup segments(segments)
else:
   # フルオーディオ分析
   print(" ** フルオーディオ分析を実行中...")
   if emotion analyzer:
       emotion scores = emotion analyzer.analyze audio(audio path)
       print("☑ 感情分析完了")
       print("▲ フルオーディオモードでは感情分析を有効にしてください")
print("\n ☑ 分析完了!")
```

```
📊 セグメント 1/20 を分析中... (1.7s - 5.7s)
  😼 感情: disgust
  』 ピッチ: 135Hz
Detected language: Japanese
100%| 408/408 [00:00<00:00, 641.29frames/s]
  テキスト: ピアンドジーでパンテ...
  針 話者: 話者A

■ セグメント 2/20 を分析中... (6.2s - 10.8s)
  😽 感情: happy
  』 ピッチ: 159Hz
Detected language: Japanese
100% | 454/454 [00:00<00:00, 1725.84frames/s]
  テキスト: 僕やってた時でそんな...
  針 話者: 話者A
📊 セグメント 3/20 を分析中... (11.2s - 15.7s)
  🛂 感情: happy
  』 ピッチ: 188Hz
Detected language: Japanese
100%| 452/452 [00:00<00:00, 1961.31frames/s]
  ▶ テキスト: 当時のパンテーズ19...
  ●● 話者: 話者A
📊 セグメント 4/20 を分析中... (17.8s - 24.4s)
  🝢 感情: disgust
  』 ピッチ: 167Hz
Detected language: Japanese
100% 659/659 [00:00<00:00, 712.74frames/s]
  テキスト:パンテヌールB5が入...

    話者: 話者A

■ セグメント 5/20 を分析中... (42.4s - 46.0s)
  😽 感情: happy
  』 ピッチ: 229Hz
Detected language: Japanese
100%| 356/356 [00:00<00:00, 1056.74frames/s]
  テキスト:なんか黄色というかゴ...
  點 話者: 話者B
📊 セグメント 6/20 を分析中... (44.5s - 47.5s)
  🝢 感情: disgust
  』 ピッチ: 173Hz
Detected language: Japanese
100%| 302/302 [00:00<00:00, 1407.61frames/s]

  テキスト: そうですよね、気を挨...

  野 話者: 話者A
📊 セグメント 7/20 を分析中... (48.6s - 54.6s)
  😽 感情: happy
  』 ピッチ: 169Hz
Detected language: Japanese
100% | 600/600 [00:00<00:00, 1970.16frames/s]
```

```
テキスト: あれを大体的にやって...
  11 話者: 話者A
👔 セグメント 8/20 を分析中... (55.1s - 60.5s)
  😼 感情: happy
  』 ピッチ: 169Hz
Detected language: Japanese
100%| 546/546 [00:00<00:00, 1489.47frames/s]
  テキスト:元々パンテーヌルB5...

    話者: 話者A

📊 セグメント 9/20 を分析中... (63.4s - 66.4s)
  💺 感情: happy
  ┛ ピッチ: 156Hz
Detected language: Japanese
100%| 300/300 [00:00<00:00, 1812.23frames/s]

 テキスト:第一世界大戦に平手の...

 11 話者: 話者A

    セグメント 10/20 を分析中... (68.7s - 73.0s)

  💺 感情: fearful
  ■ ピッチ: 199Hz
Detected language: Japanese
100% | 435/435 [00:00<00:00, 1452.68frames/s]
  テキスト: そういうセームが入っ...
  野話者:話者A
📊 セグメント 11/20 を分析中... (74.4s - 80.3s)
  😽 感情: happy
  』 ピッチ: 174Hz
Detected language: Japanese
100%| 582/582 [00:00<00:00, 1010.09frames/s]
  テキスト:でもその後パンテンフ...
 野話者: 話者A
📊 セグメント 12/20 を分析中... (83.7s - 93.7s)
  😽 感情: happy
  』 ピッチ: 181Hz
Detected language: Japanese
100%| 1002/1002 [00:00<00:00, 1265.69frames/s]
  テキスト: でもいいってみんなに...
  野話者:話者A

■ セグメント 13/20 を分析中... (94.3s - 98.3s)
  🝢 感情: disgust
  』 ピッチ: 183Hz
Detected language: Japanese
100% 401/401 [00:00<00:00, 1168.27frames/s]

    テキスト: パンテナルB5はもっ...

  野話者:話者A
📊 セグメント 14/20 を分析中... (127.7s - 135.3s)
  🥞 感情: happy
  』 ピッチ: 201Hz
Detected language: Japanese
100% | 751/751 [00:00<00:00, 1886.26frames/s]
```

```
テキスト: これ、実は黄色ではな...
  11 話者: 話者A
📊 セグメント 15/20 を分析中... (139.1s - 144.6s)
  🦖 感情: fearful
  』 ピッチ: 207Hz
Detected language: Japanese
100%| 555/555 [00:00<00:00, 1068.31frames/s]
  ▶ テキスト: コミュニケーションと...
  野 話者: 話者A
📊 セグメント 16/20 を分析中... (144.8s - 148.9s)
  😼 感情: disgust
  ┛ ピッチ: 167Hz
Detected language: Japanese
100% | 406/406 [00:00<00:00, 2148.96frames/s]
  テキスト:細ではないですけど切...
  11 話者: 話者A
■ セグメント 17/20 を分析中... (149.3s - 154.8s)
  💺 感情: angry
  ■ ピッチ: 151Hz
Detected language: Japanese
100% | 553/553 [00:00<00:00, 1462.02frames/s]

  テキスト: その象徴的に見せよっ...

  針 話者: 話者A
■ セグメント 18/20 を分析中... (158.0s - 161.2s)
  № 感情: fearful
  』 ピッチ: 282Hz
Detected language: Japanese
100%| 315/315 [00:00<00:00, 2143.45frames/s]

  テキスト: それずっとやられてる...

  針 話者: 話者B
📊 セグメント 19/20 を分析中... (162.7s - 166.8s)
  😽 感情: happy
  』 ピッチ: 212Hz
Detected language: Japanese
100% | 401/401 [00:00<00:00, 1209.08frames/s]
  🍃 テキスト: CMとかをミリューに...
  11 話者: 話者B

■ セグメント 20/20 を分析中... (166.8s - 175.9s)

  💺 感情: happy
  』 ピッチ: 157Hz
Detected language: Japanese
100%| 907/907 [00:00<00:00, 1401.31frames/s]
```

▶ テキスト: 日本もそれをやらせて...

野話者:話者A

#### ☑ 分析完了!

### 📊 結果表示(Rich UI)

```
In [10]: print(" 分析結果を表示中...")
        if SEGMENT MODE and segments:
            # セグメント分析結果の表示
            if results or metrics_results or transcription_results or speaker_results:
                display_handler.show_segment_results(
                   results if emotion_analyzer else None,
                   metrics_results if metrics_analyzer else None,
                   transcription_results if transcription_analyzer else None,
                   speaker_results if speaker_analyzer else None
               # 詳細表示
               if SHOW DETAILED METRICS and metrics results:
                   print("\n 』 詳細音声メトリクス:")
                   display_handler.show_audio_metrics_details(metrics_results)
               if SHOW_DETAILED_SPEAKERS and speaker_results:
                   display_handler.show_speaker_summary(speaker_results)
            else:
                print("★ 表示可能な分析結果がありません")
        else:
            # フルオーディオ分析結果の表示
            if emotion_analyzer and 'emotion_scores' in locals():
                display_handler.show_single_result(emotion_scores, SHOW_DETAILED_EMOTION
            else:
                print("X フルオーディオ分析の結果がありません")
```

#### ■ 分析結果を表示中...

🦠 感情・音声・文字起こし・話者タイムライン

時間	」 _ 話者	ー テキスト ー	感情	   信頼度 	ピッチ	音量
1.7s	話者A		嫌悪	29%	135Hz	0.
6.2s	話者A		喜び	28%	159Hz	0 .
11.2s	話者A		喜び	31%	188Hz	0 .
17.8s	話者A		嫌悪	27%	167Hz	0
42.4s	話者B		喜び	29%	229Hz	0.
44.5s	話者A		嫌悪	28%	173Hz	0.
48.6s	│ 話者A │		喜び	30%	169Hz	0 .
55.1s	│ 話者A │		喜び	30%	169Hz	0
63.4s	│ 話者A │		喜び	28%	156Hz	0.
68.7s	話者A		恐怖	28%	199Hz	0.
74.4s	話者A		喜び	31%	174Hz	0.
83.7s	話者A		喜び	29%	181Hz	0.
94.3s	│ 話者A │		嫌悪	27%	183Hz	0
127.7s	話者A		喜び	29%	201Hz	0.
139.1s	話者A		恐怖	26%	207Hz	0.
144.8s	話者A		嫌悪	27%	167Hz	0.
149.3s	│ 話者A │		怒り	30%	151Hz	0.
158.0s	│ 話者B │		恐怖	26%	282Hz	0.
162.7s	話者B		喜び	28%	212Hz	0
166.8s	話者A	<u>I</u>	喜び	28%	157Hz	0.
					_	

#### ※ 統計サマリー

項目	   値	I
総総時間を受ける。 総総のでは、おりますが、ままりますが、ままりますが、ままりますが、ままりますが、ままますが、ままますが、ままますが、まままますが、ままままますが、ままままますが、まままままままま	20   103.9秒 5.2秒   喜び (11 セグメント) 183 Hz   9.1 195.5   0.036   JA 100.0% 2 話者A (17 セグメント)	

#### ■ 詳細音声メトリクス:

#### ■ ピッチ分析

時間	・ 平均ピッチ   ヒ	プッチ範囲   有音	
1.7s 6.2s 11.2s 17.8s 42.4s 44.5s 48.6s 55.1s 63.4s	135 Hz 159 Hz 188 Hz 167 Hz 229 Hz 173 Hz 169 Hz 169 Hz 156 Hz	170 Hz 194 Hz 451 Hz 170 Hz 217 Hz 241 Hz 267 Hz 210 Hz 130 Hz	68.5% 74.7% 71.8% 71.4% 50.8% 64.4% 74.1% 75.2%
68.7s 74.4s 83.7s 94.3s 127.7s 139.1s 144.8s 149.3s 158.0s 162.7s 166.8s	199 Hz 174 Hz 181 Hz 183 Hz 201 Hz 207 Hz 167 Hz 151 Hz 282 Hz 212 Hz	340 Hz 157 Hz 260 Hz 224 Hz 264 Hz 292 Hz 196 Hz 141 Hz 391 Hz 350 Hz 166 Hz	74.1% 68.1% 79.1% 71.9% 73.7% 77.6% 78.5% 62.1% 71.9% 69.1% 75.4% 77.8%

#### 

時間	明度	  清涼度	 
1.7s 6.2s 11.2s 17.8s 42.4s 44.5s 48.6s 55.1s 63.4s 68.7s 74.4s 83.7s 94.3s 127.7s 139.1s 144.8s 149.3s 158.0s 162.7s 166.8s	10.5 7.0 7.7 8.8 9.1 11.4 8.0 8.2 11.0 10.3 7.5 8.2 8.7 10.2 8.8 10.1 9.6 10.6 8.2 8.8	187.9 198.9 193.4 195.6 185.7 192.8 203.0 194.3 199.7 191.3 207.8 194.8 195.3 198.6 206.5 185.8 194.4 193.3 193.5 198.3	2795 Hz 2198 Hz 2272 Hz 2460 Hz 2308 Hz 2639 Hz 2356 Hz 2583 Hz 2657 Hz 2745 Hz 2192 Hz 2351 Hz 2430 Hz 2414 Hz 2776 Hz 2460 Hz 2524 Hz 2159 Hz 2381 Hz

點 詳細話者分析:

**#** 話者分析サマリー

話者	発話時間   第	発話割合   セ	グメント数   平均: -	   します
話者A	93.2秒	89.7%	17	5.5秒
話者B	10.7秒	10.3%	3	3.6秒

# 結果をDataFrameとして表示

```
In [11]: if SEGMENT MODE and segments:
             print(" 編集をDataFrameで表示...")
             # 結果をまとめたDataFrameを作成
             df_data = []
             max len = max(
                 len(results) if results else 0,
                 len(metrics_results) if metrics_results else 0,
                 len(transcription_results) if transcription_results else 0,
                 len(speaker_results) if speaker_results else 0
             for i in range(max_len):
                 row = \{\}
                 # 基本情報
                 if results and i < len(results):</pre>
                     start_time, end_time, emotion_scores = results[i]
                     row['開始時間'] = f"{start_time:.1f}s"
                     row['終了時間'] = f"{end_time:.1f}s"
                     row['長さ'] = f"{end_time - start_time:.1f}s"
```

```
# 感情分析結果
           dominant_emotion, confidence = max(emotion_scores.items(), key=lambd
           emotion_jp = display_handler._translate_emotion(dominant_emotion)
           row['感情'] = emotion_jp
           row['信頼度'] = f"{confidence:.1%}"
       elif metrics results and i < len(metrics results):</pre>
           start_time, end_time, _ = metrics_results[i]
           row['開始時間'] = f"{start time:.1f}s"
           row['終了時間'] = f"{end_time:.1f}s"
           row['長さ'] = f"{end_time - start_time:.1f}s"
       elif transcription_results and i < len(transcription_results):</pre>
           start_time, end_time, _ = transcription_results[i]
           row['開始時間'] = f"{start_time:.1f}s"
           row['終了時間'] = f"{end_time:.1f}s"
           row['長さ'] = f"{end_time - start_time:.1f}s"
       elif speaker_results and i < len(speaker_results):</pre>
           start_time, end_time, _ = speaker_results[i]
           row['開始時間'] = f"{start_time:.1f}s"
           row['終了時間'] = f"{end time:.1f}s"
           row['長さ'] = f"{end_time - start_time:.1f}s"
       # 音声メトリクス
       if metrics_results and i < len(metrics_results):</pre>
           _, _, audio_metrics = metrics_results[i]
           row['ピッチ(Hz)'] = f"{audio_metrics.get('pitch_mean', 0):.0f}"
           row['音量'] = f"{audio_metrics.get('rms_mean', 0):.3f}"
           row['明度'] = f"{audio_metrics.get('brightness_score', 0):.1f}"
       # 文字起こし
       if transcription results and i < len(transcription results):</pre>
           _, _, transcription = transcription_results[i]
           row['テキスト'] = transcription.get('preview_text', '[空]')
           row['言語'] = transcription.get('language', 'unknown').upper()
       # 話者
       if speaker_results and i < len(speaker_results):</pre>
           _, _, speaker = speaker_results[i]
           row['話者'] = speaker
       df_data.append(row)
   if df data:
       df_results = pd.DataFrame(df_data)
       print("\n <a> 分析結果一覧:")</a>
       display(df_results)
       # CSVとして保存するオプション
       if save_csv.lower() in ['y', 'yes', 'はい']:
           csv_filename = f"analysis_results_{os.path.splitext(os.path.basename
           df_results.to_csv(csv_filename, index=False, encoding='utf-8')
           print(f"☑ 結果を保存: {csv filename}")
   else:
       print("★ 表示可能なデータがありません")
else:
   print(" Z フルオーディオモードではDataFrame表示はサポートされていません")
```

- 📋 分析結果一覧:

	開始時間	終了時 間	長さ	感 情	信頼 度	ピッチ (Hz)	音量	明 度	テキスト	言語	話 者
0	1.7s	5.7s	4.1s	嫌 悪	29.1%	135	0.022	10.5	ピアンドジーでパ ンテ	JA	話 者A
1	6.2s	10.8s	4.5s	喜 び	28.4%	159	0.036	7.0	僕やってた時で そんな…	JA	話 者A
2	11.2s	15.7s	4.5s	喜 び	30.8%	188	0.032	7.7	当時のパンテー ズ19	JA	話 者A
3	17.8s	24.4s	6.6s	嫌悪	27.3%	167	0.037	8.8	パンテヌールB5 が入	JA	話 者A
4	42.4s	46.0s	3.6s	喜 び	28.9%	229	0.023	9.1	なんか黄色とい うかゴ	JA	話 者B
5	44.5s	47.5s	3.0s	嫌 悪	27.9%	173	0.030	11.4	そうですよね、気 を挨	JA	話 者A
6	48.6s	54.6s	6.0s	喜 び	29.6%	169	0.047	8.0	あれを大体的に やって	JA	話 者A
7	55.1s	60.5s	5.5s	喜 び	29.7%	169	0.040	8.2	元々パンテーヌ ルB5	JA	話 者A
8	63.4s	66.4s	3.0s	喜 び	27.8%	156	0.036	11.0	第一世界大戦 に平手の…	JA	話 者A
9	68.7s	73.0s	4.4s	恐 怖	28.2%	199	0.035	10.3	そういうセームが 入っ	JA	話 者A
10	74.4s	80.3s	5.8s	喜 び	31.1%	174	0.050	7.5	でもその後パンテ ンフ	JA	話 者A
11	83.7s	93.7s	10.0s	喜 び	29.5%	181	0.039	8.2	でもいいってみん なに	JA	話 者A
12	94.3s	98.3s	4.0s	嫌 悪	26.8%	183	0.044	8.7	パンテナルB5は もっ	JA	話 者A
13	127.7s	135.3s	7.5s	喜 び	28.5%	201	0.043	10.2	これ、実は黄色 ではな	JA	話 者A
14	139.1s	144.6s	5.6s	恐 怖	26.1%	207	0.051	8.8	コミュニケーション と	JA	話 者A
15	144.8s	148.9s	4.1s	嫌悪	26.8%	167	0.036	10.1	細ではないです けど切	JA	話 者A
16	149.3s	154.8s	5.5s	怒 り	29.5%	151	0.025	9.6	その象徴的に見せよっ	JA	話 者A
17	158.0s	161.2s	3.2s	恐 怖	26.5%	282	0.022	10.6	それずっとやられ てる	JA	話 者B
18	162.7s	166.8s	4.0s	喜 び	27.8%	212	0.025	8.2	CMとかをミリュー に…	JA	話 者B
19	166.8s	175.9s	9.1s	喜 び	28.4%	157	0.039	8.8	日本もそれをや らせて	JA	話 者A

# ■ 可視化(matplotlib/seaborn)

```
In [12]: if SEGMENT MODE and (results or metrics results):
            print(" 編 結果を可視化中...")
            # 図のスタイル設定
            plt.style.use('default')
            sns.set_palette("husl")
            # 感情分析結果の可視化
            if results:
                fig, axes = plt.subplots(2, 2, figsize=(15, 10))
                fig.suptitle('場 感情分析結果', fontsize=16, fontweight='bold')
                # 感情の時系列変化
                emotions timeline = []
                times_timeline = []
                confidences_timeline = []
                for start_time, end_time, emotion_scores in results:
                    dominant_emotion, confidence = max(emotion_scores.items(), key=lambd
                    emotion_jp = display_handler._translate_emotion(dominant_emotion)
                    emotions_timeline.append(emotion_jp)
                    times timeline.append(start time)
                    confidences_timeline.append(confidence)
                # 1. 感情分布 (円グラフ)
                emotion_counts = pd.Series(emotions_timeline).value_counts()
                axes[0, 0].pie(emotion_counts.values, labels=emotion_counts.index, autop
                axes[0, 0].set_title('感情分布')
                # 2. 信頼度の時系列変化
                axes[0, 1].plot(times_timeline, confidences_timeline, 'o-', linewidth=2,
                axes[0, 1].set_title('信頼度の時系列変化')
                axes[0, 1].set_xlabel('時間 (秒)')
                axes[0, 1].set_ylabel('信頼度')
                axes[0, 1].grid(True, alpha=0.3)
                # 3. 感情別の信頼度分布
                emotion confidence df = pd.DataFrame({
                    '感情': emotions_timeline,
                    '信頼度': confidences timeline
                sns.boxplot(data=emotion_confidence_df, x='感情', y='信頼度', ax=axes[1,
                axes[1, 0].set title('感情別信頼度分布')
                axes[1, 0].tick_params(axis='x', rotation=45)
                # 4. セグメント長と信頼度の関係
                segment_lengths = [end_time - start_time for start_time, end_time, _ in
                axes[1, 1].scatter(segment_lengths, confidences_timeline, alpha=0.6, s=5
                axes[1, 1].set title('セグメント長と信頼度の関係')
                axes[1, 1].set_xlabel('セグメント長 (秒)')
                axes[1, 1].set ylabel('信頼度')
                axes[1, 1].grid(True, alpha=0.3)
                plt.tight_layout()
                plt.show()
```

```
# 音声メトリクスの可視化
   if metrics results:
       fig, axes = plt.subplots(2, 2, figsize=(15, 10))
       fig.suptitle(' 🎜 音声メトリクス分析結果', fontsize=16, fontweight='bold')
       metrics data = {
           'time': [],
           'pitch': [],
           'volume': [],
           'brightness': [],
           'clarity': []
       }
       for start_time, end_time, audio_metrics in metrics_results:
           metrics_data['time'].append(start_time)
           metrics_data['pitch'].append(audio_metrics.get('pitch_mean', 0))
           metrics_data['volume'].append(audio_metrics.get('rms_mean', 0))
           metrics_data['brightness'].append(audio_metrics.get('brightness_scor
           metrics data['clarity'].append(audio metrics.get('clarity score', 0)
       # 1. ピッチの時系列変化
       axes[0, 0].plot(metrics_data['time'], metrics_data['pitch'], 'o-', color
       axes[0, 0].set_title('ピッチの時系列変化')
       axes[0, 0].set_xlabel('時間 (秒)')
       axes[0, 0].set_ylabel('ピッチ (Hz)')
       axes[0, 0].grid(True, alpha=0.3)
       # 2. 音量の時系列変化
       axes[0, 1].plot(metrics_data['time'], metrics_data['volume'], 'o-', colo
       axes[0, 1].set title('音量の時系列変化')
       axes[0, 1].set_xlabel('時間(秒)')
       axes[0, 1].set_ylabel('音量 (RMS)')
       axes[0, 1].grid(True, alpha=0.3)
       # 3. 明度と清涼度の散布図
       axes[1, 0].scatter(metrics_data['brightness'], metrics_data['clarity'],
       axes[1, 0].set title('明度 vs 清涼度')
       axes[1, 0].set_xlabel('明度')
       axes[1, 0].set ylabel('清涼度')
       axes[1, 0].grid(True, alpha=0.3)
       # 4. 音声メトリクスの分布
       metrics df = pd.DataFrame({
           'ピッチ': metrics_data['pitch'],
           '音量': [v * 1000 for v in metrics_data['volume']], # スケール調整
           '明度': metrics_data['brightness'],
           '清涼度': metrics data['clarity']
       metrics_df.boxplot(ax=axes[1, 1])
       axes[1, 1].set_title('音声メトリクス分布')
       axes[1, 1].tick_params(axis='x', rotation=45)
       plt.tight layout()
       plt.show()
else:
   print(" 可視化するデータがありません")
```

■ 結果を可視化中...

```
/tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 24863 (\N{CJK UNIFIED
IDEOGRAPH-611F}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
/tmp/ipykernel 908212/3325136464.py:54: UserWarning: Glyph 24773 (\N{CJK UNIFIED
IDEOGRAPH-60C5}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 20998 (\N{CJK UNIFIED
IDEOGRAPH-5206}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 24067 (\N{CJK UNIFIED
IDEOGRAPH-5E03}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
/tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 21916 (\N{CJK UNIFIED
IDEOGRAPH-559C}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel 908212/3325136464.py:54: UserWarning: Glyph 12403 (\N{HIRAGANA LET
TER BI}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 23244 (\N{CJK UNIFIED
IDEOGRAPH-5ACC}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
/tmp/ipykernel 908212/3325136464.py:54: UserWarning: Glyph 24746 (\N{CJK UNIFIED
IDEOGRAPH-60AA}) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 24656 (\N{CJK UNIFIED
IDEOGRAPH-6050}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
/tmp/ipykernel 908212/3325136464.py:54: UserWarning: Glyph 24598 (\N{CJK UNIFIED
IDEOGRAPH-6016}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 24594 (\N{CJK UNIFIED
IDEOGRAPH-6012}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 12426 (\N{HIRAGANA LET
TER RI}) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 26178 (\N{CJK UNIFIED
IDEOGRAPH-6642}) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel 908212/3325136464.py:54: UserWarning: Glyph 38291 (\N{CJK UNIFIED
IDEOGRAPH-9593}) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 31186 (\N{CJK UNIFIED
IDEOGRAPH-79D2}) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel 908212/3325136464.py:54: UserWarning: Glyph 20449 (\N{CJK UNIFIED
IDEOGRAPH-4FE1}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 38972 (\N{CJK UNIFIED
IDEOGRAPH-983C}) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel 908212/3325136464.py:54: UserWarning: Glyph 24230 (\N{CJK UNIFIED
IDEOGRAPH-5EA6}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel 908212/3325136464.py:54: UserWarning: Glyph 12398 (\N{HIRAGANA LET
TER NO}) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel 908212/3325136464.py:54: UserWarning: Glyph 31995 (\N{CJK UNIFIED
IDEOGRAPH-7CFB}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
```

```
/tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 21015 (\N{CJK UNIFIED
IDEOGRAPH-5217}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 22793 (\N{CJK UNIFIED
IDEOGRAPH-5909}) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 21270 (\N{CJK UNIFIED
IDEOGRAPH-5316}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 21029 (\N{CJK UNIFIED
IDEOGRAPH-5225}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 12475 (\N{KATAKANA LET
TER SE}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
/tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 12464 (\N{KATAKANA LET
TER GU}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel 908212/3325136464.py:54: UserWarning: Glyph 12513 (\N{KATAKANA LET
TER ME}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 12531 (\N{KATAKANA LET
TER N)) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 12488 (\N{KATAKANA LET
TER TO}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 38263 (\N{CJK UNIFIED
IDEOGRAPH-9577}) missing from font(s) DejaVu Sans.
  plt.tight layout()
/tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 12392 (\N{HIRAGANA LET
TER TO}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 38306 (\N{CJK UNIFIED
IDEOGRAPH-95A2}) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel 908212/3325136464.py:54: UserWarning: Glyph 20418 (\N{CJK UNIFIED
IDEOGRAPH-4FC2}) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 127917 (\N{PERFORMING
ARTS ) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel 908212/3325136464.py:54: UserWarning: Glyph 26512 (\N{CJK UNIFIED
IDEOGRAPH-6790}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 32080 (\N{CJK UNIFIED
IDEOGRAPH-7D50}) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel_908212/3325136464.py:54: UserWarning: Glyph 26524 (\N{CJK UNIFIED
IDEOGRAPH-679C}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 24863 (\N{CJK UNIF
IED IDEOGRAPH-611F}) missing from font(s) DejaVu Sans.
  fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 24773 (\N{CJK UNIF
IED IDEOGRAPH-60C5}) missing from font(s) DejaVu Sans.
  fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
```

```
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 20998 (\N{CJK UNIF
IED IDEOGRAPH-5206}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 24067 (\N{CJK UNIF
IED IDEOGRAPH-5E03}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 21916 (\N{CJK UNIF
IED IDEOGRAPH-559C}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 12403 (\N{HIRAGANA
LETTER BI}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 23244 (\N{CJK UNIF
IED IDEOGRAPH-5ACC}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 24746 (\N{CJK UNIF
IED IDEOGRAPH-60AA}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 24656 (\N{CJK UNIF
IED IDEOGRAPH-6050}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 24598 (\N{CJK UNIF
IED IDEOGRAPH-6016}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 24594 (\N{CJK UNIF
IED IDEOGRAPH-6012}) missing from font(s) DejaVu Sans.
 fig.canvas.print figure(bytes io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 12426 (\N{HIRAGANA
LETTER RI}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 20449 (\N{CJK UNIF
IED IDEOGRAPH-4FE1}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 38972 (\N{CJK UNIF
IED IDEOGRAPH-983C}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 24230 (\N{CJK UNIF
IED IDEOGRAPH-5EA6}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 12398 (\N{HIRAGANA
LETTER NO}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 26178 (\N{CJK UNIF
IED IDEOGRAPH-6642}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
```

```
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 31995 (\N{CJK UNIF
IED IDEOGRAPH-7CFB}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 21015 (\N{CJK UNIF
IED IDEOGRAPH-5217}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 22793 (\N{CJK UNIF
IED IDEOGRAPH-5909}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 21270 (\N{CJK UNIF
IED IDEOGRAPH-5316}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 38291 (\N{CJK UNIF
IED IDEOGRAPH-9593}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 31186 (\N{CJK UNIF
IED IDEOGRAPH-79D2}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 21029 (\N{CJK UNIF
IED IDEOGRAPH-5225}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 12475 (\N{KATAKANA
LETTER SE}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 12464 (\N{KATAKANA
LETTER GU}) missing from font(s) DejaVu Sans.
 fig.canvas.print figure(bytes io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 12513 (\N{KATAKANA
LETTER ME}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 12531 (\N{KATAKANA
LETTER N}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 12488 (\N{KATAKANA
LETTER TO}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 38263 (\N{CJK UNIF
IED IDEOGRAPH-9577}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 12392 (\N{HIRAGANA
LETTER TO}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 38306 (\N{CJK UNIF
IED IDEOGRAPH-95A2}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
```

te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 20418 (\N{CJK UNIF IED IDEOGRAPH-4FC2}) missing from font(s) DejaVu Sans.

fig.canvas.print\_figure(bytes\_io, \*\*kw)

/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/site-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 127917 (\N{PERFORM ING ARTS}) missing from font(s) DejaVu Sans.

fig.canvas.print\_figure(bytes\_io, \*\*kw)

/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 26512 (\N{CJK UNIF IED IDEOGRAPH-6790}) missing from font(s) DejaVu Sans.

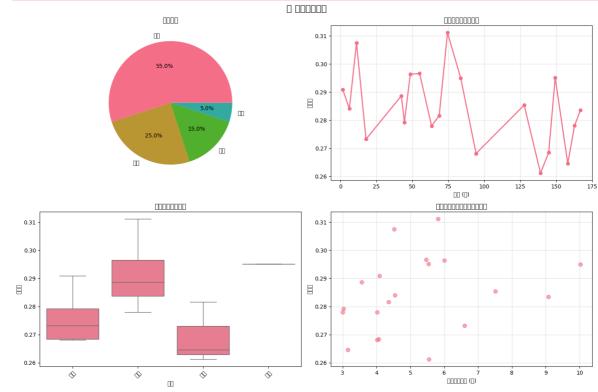
fig.canvas.print\_figure(bytes\_io, \*\*kw)

/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/site-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 32080 (\N{CJK UNIF IED IDEOGRAPH-7D50}) missing from font(s) DejaVu Sans.

fig.canvas.print\_figure(bytes\_io, \*\*kw)

/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 26524 ( $\N{CJK\ UNIF\ IED\ IDEOGRAPH-679C}$ ) missing from font(s) DejaVu Sans.

fig.canvas.print\_figure(bytes\_io, \*\*kw)



```
/tmp/ipykernel_908212/3325136464.py:109: UserWarning: Glyph 26178 (\N{CJK UNIFIED
IDEOGRAPH-6642}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
/tmp/ipykernel 908212/3325136464.py:109: UserWarning: Glyph 38291 (\N{CJK UNIFIED
IDEOGRAPH-9593}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:109: UserWarning: Glyph 31186 (\N{CJK UNIFIED
IDEOGRAPH-79D2}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:109: UserWarning: Glyph 12500 (\N{KATAKANA LE
TTER PI}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
/tmp/ipykernel_908212/3325136464.py:109: UserWarning: Glyph 12483 (\N{KATAKANA LE
TTER SMALL TU}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel 908212/3325136464.py:109: UserWarning: Glyph 12481 (\N{KATAKANA LE
TTER TI}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:109: UserWarning: Glyph 12398 (\N{HIRAGANA LE
TTER NO}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
/tmp/ipykernel 908212/3325136464.py:109: UserWarning: Glyph 31995 (\N{CJK UNIFIED
IDEOGRAPH-7CFB}) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel_908212/3325136464.py:109: UserWarning: Glyph 21015 (\N{CJK UNIFIED
IDEOGRAPH-5217}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
/tmp/ipykernel 908212/3325136464.py:109: UserWarning: Glyph 22793 (\N{CJK UNIFIED
IDEOGRAPH-5909}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:109: UserWarning: Glyph 21270 (\N{CJK UNIFIED
IDEOGRAPH-5316}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:109: UserWarning: Glyph 38899 (\N{CJK UNIFIED
IDEOGRAPH-97F3}) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel_908212/3325136464.py:109: UserWarning: Glyph 37327 (\N{CJK UNIFIED
IDEOGRAPH-91CF}) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel 908212/3325136464.py:109: UserWarning: Glyph 26126 (\N{CJK UNIFIED
IDEOGRAPH-660E}) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel_908212/3325136464.py:109: UserWarning: Glyph 24230 (\N{CJK UNIFIED
IDEOGRAPH-5EA6}) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel 908212/3325136464.py:109: UserWarning: Glyph 28165 (\N{CJK UNIFIED
IDEOGRAPH-6E05}) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel_908212/3325136464.py:109: UserWarning: Glyph 28092 (\N{CJK UNIFIED
IDEOGRAPH-6DBC}) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel 908212/3325136464.py:109: UserWarning: Glyph 22768 (\N{CJK UNIFIED
IDEOGRAPH-58F0}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:109: UserWarning: Glyph 12513 (\N{KATAKANA LE
TTER ME}) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel 908212/3325136464.py:109: UserWarning: Glyph 12488 (\N{KATAKANA LE
TTER TO}) missing from font(s) DejaVu Sans.
 plt.tight_layout()
```

```
/tmp/ipykernel_908212/3325136464.py:109: UserWarning: Glyph 12522 (\N{KATAKANA LE
TTER RI}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:109: UserWarning: Glyph 12463 (\N{KATAKANA LE
TTER KU}) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel_908212/3325136464.py:109: UserWarning: Glyph 12473 (\N{KATAKANA LE
TTER SU}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:109: UserWarning: Glyph 20998 (\N{CJK UNIFIED
IDEOGRAPH-5206}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:109: UserWarning: Glyph 24067 (\N{CJK UNIFIED
IDEOGRAPH-5E03}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
/tmp/ipykernel_908212/3325136464.py:109: UserWarning: Glyph 127925 (\N{MUSICAL NO
TE}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
/tmp/ipykernel 908212/3325136464.py:109: UserWarning: Glyph 26512 (\N{CJK UNIFIED
IDEOGRAPH-6790}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
tmp/ipykernel_908212/3325136464.py:109: UserWarning: Glyph 32080 (\N{CJK UNIFIED
IDEOGRAPH-7D50}) missing from font(s) DejaVu Sans.
  plt.tight layout()
tmp/ipykernel_908212/3325136464.py:109: UserWarning: Glyph 26524 (\N{CJK UNIFIED
IDEOGRAPH-679C}) missing from font(s) DejaVu Sans.
  plt.tight_layout()
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 12500 (\N{KATAKANA
LETTER PI ) missing from font(s) DejaVu Sans.
  fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 12483 (\N{KATAKANA
LETTER SMALL TU}) missing from font(s) DejaVu Sans.
  fig.canvas.print figure(bytes io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 12481 (\N{KATAKANA
LETTER TI}) missing from font(s) DejaVu Sans.
  fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 12398 (\N{HIRAGANA
LETTER NO}) missing from font(s) DejaVu Sans.
  fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 26178 (\N{CJK UNIF
IED IDEOGRAPH-6642}) missing from font(s) DejaVu Sans.
  fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 31995 (\N{CJK UNIF
IED IDEOGRAPH-7CFB}) missing from font(s) DejaVu Sans.
  fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 21015 (\N{CJK UNIF
IED IDEOGRAPH-5217}) missing from font(s) DejaVu Sans.
  fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 22793 (\N{CJK UNIF
IED IDEOGRAPH-5909}) missing from font(s) DejaVu Sans.
  fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
```

```
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 21270 (\N{CJK UNIF
IED IDEOGRAPH-5316}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 38291 (\N{CJK UNIF
IED IDEOGRAPH-9593}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 31186 (\N{CJK UNIF
IED IDEOGRAPH-79D2}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 38899 (\N{CJK UNIF
IED IDEOGRAPH-97F3}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 37327 (\N{CJK UNIF
IED IDEOGRAPH-91CF}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 28165 (\N{CJK UNIF
IED IDEOGRAPH-6E05}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 28092 (\N{CJK UNIF
IED IDEOGRAPH-6DBC}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 24230 (\N{CJK UNIF
IED IDEOGRAPH-5EA6}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 26126 (\N{CJK UNIF
IED IDEOGRAPH-660E}) missing from font(s) DejaVu Sans.
 fig.canvas.print figure(bytes io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 22768 (\N{CJK UNIF
IED IDEOGRAPH-58F0}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 12513 (\N{KATAKANA
LETTER ME}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 12488 (\N{KATAKANA
LETTER TO}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 12522 (\N{KATAKANA
LETTER RI}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 12463 (\N{KATAKANA
LETTER KU}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 12473 (\N{KATAKANA
LETTER SU}) missing from font(s) DejaVu Sans.
 fig.canvas.print_figure(bytes_io, **kw)
/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/si
```

te-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 20998 (\N{CJK UNIF IED IDEOGRAPH-5206}) missing from font(s) DejaVu Sans.

fig.canvas.print\_figure(bytes\_io, \*\*kw)

/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/site-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 24067 (\N{CJK UNIF IED IDEOGRAPH-5E03}) missing from font(s) DejaVu Sans.

fig.canvas.print\_figure(bytes\_io, \*\*kw)

/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/site-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 127925 (\N{MUSICAL NOTE}) missing from font(s) DejaVu Sans.

fig.canvas.print\_figure(bytes\_io, \*\*kw)

/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/site-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 26512 (\N{CJK UNIF IED IDEOGRAPH-6790}) missing from font(s) DejaVu Sans.

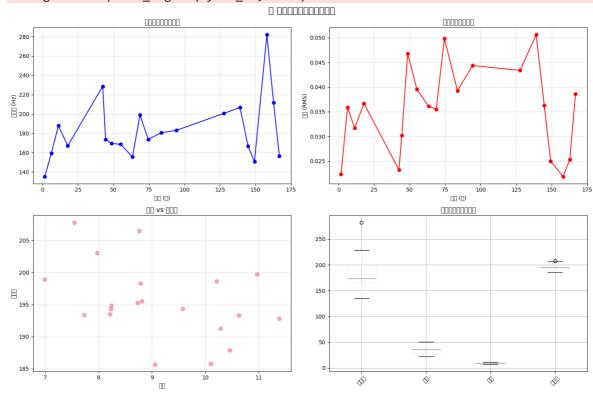
fig.canvas.print\_figure(bytes\_io, \*\*kw)

/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/site-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 32080 (\N{CJK UNIF IED IDEOGRAPH-7D50}) missing from font(s) DejaVu Sans.

fig.canvas.print\_figure(bytes\_io, \*\*kw)

/home/nnao45/ghq/github.com/nnao45/speech-emotion-whisper/.venv/lib/python3.12/site-packages/IPython/core/pylabtools.py:170: UserWarning: Glyph 26524 (\N{CJK UNIF IED IDEOGRAPH-679C}) missing from font(s) DejaVu Sans.

fig.canvas.print\_figure(bytes\_io, \*\*kw)



### 🖊 クリーンアップ

```
In []: # 一時ファイルのクリーンアップ
if not OUTPUT_AUDIO and os.path.exists(audio_path):
    os.unlink(audio_path)
    print(f" ✓ 一時音声ファイルを削除: {audio_path}")

print(" ☑ 分析完了!")
print("\n ※ お疲れ様でした!")
```

# ▶ 使用方法のまとめ

### 

- 1. 設定セルで分析したいファイルと有効にしたい機能を指定
- 2. 順番にセルを実行して分析を実行
- 3. 結果をRich UI、DataFrame、グラフで確認

### 🐡 主要な設定項目

- INPUT FILE:分析対象のMP4ファイル
- ENABLE\_\*:各分析機能の有効/無効
- SEGMENT MODE: セグメント分析の有効/無効
- SPEAKER SEGMENTS:話者変更時のセグメント分割
- HF\_TOKEN: HuggingFaceトークン(話者分析用)

### トラブルシューティング

- HuggingFaceエラー: HF\_TOKEN を正しく設定
- GPUメモリエラー: USE\_GPU = False でCPUモードに切り替え
- セグメント生成エラー: FORCE\_TIME\_SPLIT = True で時間ベース分割を試す

### ■ 出力について

- Rich UI: カラフルな表形式での結果表示
- DataFrame: パンダスデータフレームでの構造化データ
- **可視化**: matplotlib/seabornによるグラフ表示
- **CSV出力**: 結果のCSVファイル保存