

# Assessment of Piano Performances

## 1. Correct Pitch - Pitch-Based Evaluation

Correct Pitch (정확도 %로 표기)

$$Pitch\ Accuracy = \frac{Correct\ Pitches}{Total\ Pitches} * 100$$

수식 1. 피치 정확도

```
import mido

# Step 1. 피치 정확도 구하기 ~ 메서드로 구현

def calculate_pitch_accuracy(input_midi_path, target_midi_path, limit_offset = 0):
    """
    인풋 MIDI와 정답 MIDI의 pitch 정확도 계산 - Yeong-Min Ko

    Args:
        input_midi: 인풋 MIDI 파일
        target_midi: 타겟 MIDI 파일

    Return:
        pitch 정확도
    """

    # MIDI 파일 읽기
    input_midi = mido.MidiFile(input_midi_path)
    target_midi = mido.MidiFile(target_midi_path)

    # 인풋 MIDI와 타겟 MIDI의 음표 정보를 추출
    input_notes = [(msg.note, msg.time) for msg in input_midi if msg.type == 'note_on']
    target_notes = [(msg.note, msg.time) for msg in target_midi if msg.type == 'note_on']

    # 피치 정확도 계산
```

```

total_accuracy = 0
for input_note in input_notes:
    for target_note in target_notes:
        if abs(input_note[0] - target_note[0]) < limit_offset:
            total_accuracy += 1
            break

# 전체 음표 수와 정확한 음표 수를 이용하여 정확도 계산
pitch_accuracy = (total_accuracy / len(target_notes)) * 100

return pitch_accuracy

# 메인 시작 부분
if __name__ == "__main__":
    # Step 2_1. Mido 라이브러리로 정답 데이터와 같은 인풋 데이터 읽기
    input_midi_path1 = 'Prelude1.mid'
    target_midi_path1 = 'Prelude1.mid'
    limit_offset = 5 # 정확도를 판단할 때 허용되는 피치 차이
    accuracy1 = calculate_pitch_accuracy(input_midi_path1, target_midi_path1, limit_offset)

    # Step 2_2. Mido 라이브러리로 정답 데이터와 다른 인풋 데이터 읽기
    input_midi_path2 = 'Prelude1.mid'
    target_midi_path2 = 'Fugue1.mid'
    limit_offset = 1
    accuracy2 = calculate_pitch_accuracy(input_midi_path2, target_midi_path2, limit_offset)

    # Step 2_3. Mido 라이브러리로 정답 데이터와 다른 인풋 데이터 읽기(피치 한계 약간 찾을 때)
    input_midi_path3 = 'Prelude1.mid'
    target_midi_path3 = 'Fugue1.mid'
    limit_offset = 10
    accuracy3 = calculate_pitch_accuracy(input_midi_path3, target_midi_path3, limit_offset)

    # Step 2_4. Mido 라이브러리로 정답 데이터와 다른 인풋 데이터 읽기(피치 한계 많이 찾을 때)
    input_midi_path4 = 'Prelude1.mid'
    target_midi_path4 = 'Fugue1.mid'
    limit_offset = 1000
    accuracy4 = calculate_pitch_accuracy(input_midi_path4, target_midi_path4, limit_offset)

    # Step 3. 각각의 정확도 출력
    print(f'Pitch Accuracy testCase 1: {accuracy1:.2f}%')
    print(f'Pitch Accuracy testCase 2: {accuracy2:.2f}%')
    print(f'Pitch Accuracy testCase 3: {accuracy3:.2f}%')
    print(f'Pitch Accuracy testCase 4: {accuracy4:.2f}%')

```

```
# Step 3. 각각의 정확도 출력
```

```
print(f'Pitch Accuracy testCase 1: {accuracy1:.2f}%')  
print(f'Pitch Accuracy testCase 2: {accuracy2:.2f}%')  
print(f'Pitch Accuracy testCase 3: {accuracy3:.2f}%')  
print(f'Pitch Accuracy testCase 4: {accuracy4:.2f}%')
```

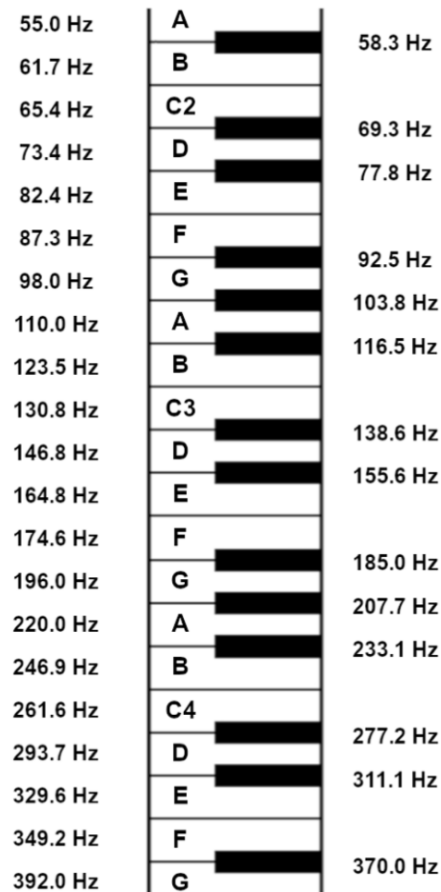
```
Pitch Accuracy testCase 1: 100.00%  
Pitch Accuracy testCase 2: 71.56%  
Pitch Accuracy testCase 3: 73.56%  
Pitch Accuracy testCase 4: 73.56%
```

- offset은 작게 하면 정확도가 올라감, 너무 크다해서 완전 틀린 것 까지 맞다고 하지는 않음

## ~~제일 처음에 잘못 생각한 방법~~

- 주파수(frequency): 단위 시간 1초 동안 얼마나 많이 진동하는가를 의미
  - 주파수가 높아지면 소리가 높아짐, 낮아지면 소리가 낮아짐
- 피치(pitch): 진동이 1초 동안 몇 번 발생했는가를 의미
- 주파수와 피치는 매우 밀접한 관계가 있으며 주파수를 통해 피치를 구할 수 있음
  - 이외에도 피아노 노트별로 이미 해당 주파수가 구해져 있음

피치 데이터를 이용하는 방법(고민중)



- 주파수를 이용하여 피치를 구하는 방법

$$P = 69 + 12 \times \log_2\left(\frac{f}{440}\right)$$

- P: 피치(pitch), f: 주파수(frequency)
- 약간의 오차가 있으므로 주파수가  $\pm 15\text{Hz}$  이내인 경우 올바르게 연주된 것으로 간주
- Correct Pitch = (정답 데이터의 단위 프레임에 해당하는 피치 값 == 입력 데이터의 단위 프레임에 해당하는 피치 값)의 개수 / (정답 데이터의 단위 프레임 개수)
  - 전제 조건: 정답 데이터와 입력 데이터(사용자의 연주 데이터)를 같은 단위로 프레임을 나눠야 함(연주 시작점만 맞춰주고 만약 연주 시간을 다르게 한다면 입력 데이터는 정답 데이터에 맞게 자르고 뒷 부분을 버릴 것인지?)
  - 예를 들면, 정답 데이터의 프레임 수는 총 100 프레임이라고 가정

- 이때 정답 데이터와 인풋 데이터의 단위 프레임에서 연주되는 피치가 같은 경우가 95개라면 0.95라는 값이 도출 (  $\text{Correct Pitch} = 95 / 100$  )

# Pseudo Code

```
# Step 1. 정답 데이터와 인풋 데이터의 미디 데이터를 추출
# Step 2. 정답 데이터와 인풋 데이터의 프레임별 피치를 배열에 삽입
# Step 3. 추출한 음표의 주파수를 계산
# Step 4. 주파수가  $\pm 10\text{Hz}$  이내인 경우 올바르게 연주된 것으로 간주합니다.
# Step 5. 올바르게 연주된 음표의 비율을 계산합니다.
```

References: **Automatic Assessment of Piano Performances Using Timbre and Pitch Features**(<https://www.mdpi.com/2079-9292/12/8/1791>)