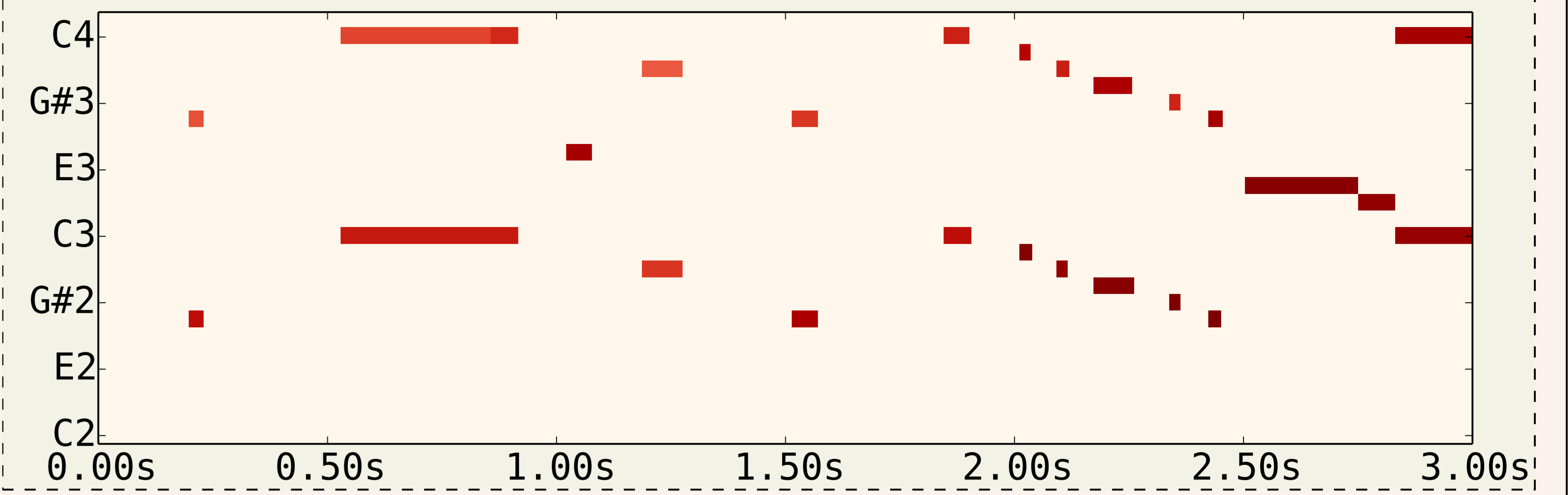


Functionality

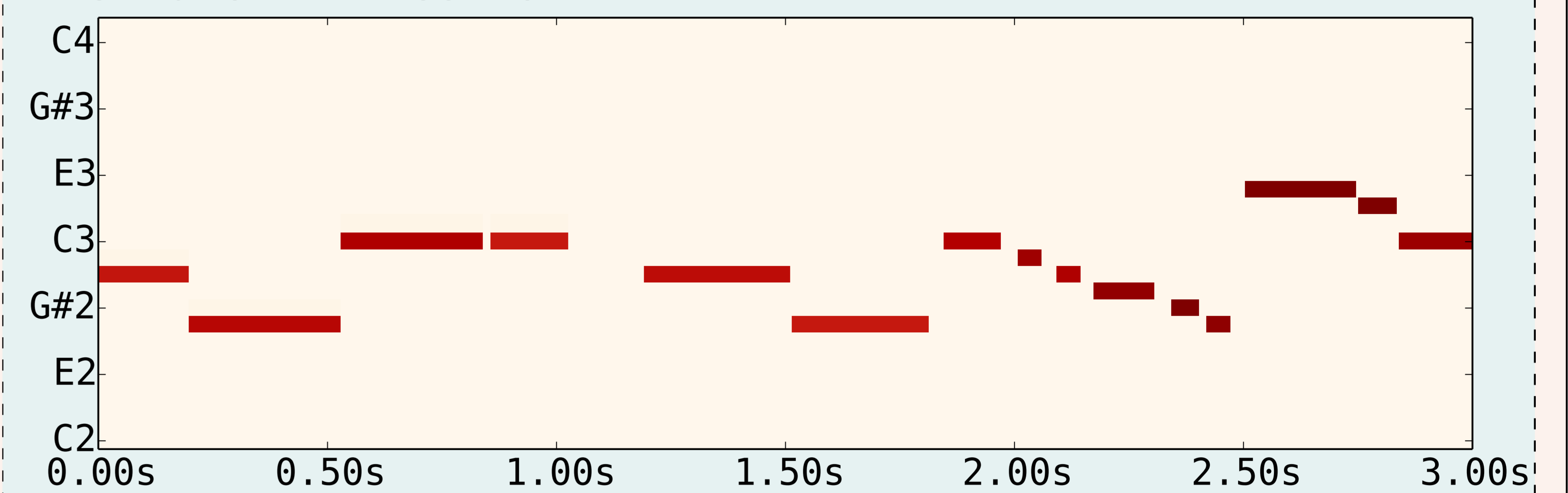
- MIDI file parsing and writing
- Creating MIDI data from scratch
- Information extraction
 - get_tempo_changes : MIDI tempo change events
 - estimate_tempo : Empirical estimate
 - get_beats : Using tempo change events
 - get_onsets : Note-on times
 - get_piano_roll : Instrument- or file-wise
 - get_chroma : Collapses piano roll across octaves
 - etc.
- Synthesis
 - fluidsynth : Using a Soundfont file
 - synthesize : Using a periodic function (e.g. sin)
- Utility functions
 - note_number_to_hz : 81 → 880.0
 - drum_name_to_note_number : 'High Tom' → 50
 - program_to_instrument_name : 20 → 'Reed Organ'
 - etc.

MIDI Data

Instrument 1: Piano



Instrument 2: Cello



•
•
•

```
import pretty_midi
# Load MIDI file into PrettyMIDI object
midi_data = pretty_midi.PrettyMIDI('midi_file.mid')
# Compute the relative amount of each semitone across the entire song, a proxy for key
print [sum(semitone)/sum(sum(midi_data.get_chroma())) for semitone in midi_data.get_chroma()]
# Shift all notes up by 5 semitones
for instrument in midi_data.instruments:
    # Don't want to shift drum notes
    if not instrument.is_drum:
        for note in instrument.notes:
            note.pitch += 5
# Synthesize the resulting MIDI data using sine waves
audio_data = midi_data.synthesize()
```

```
import pretty_midi
# Create a PrettyMIDI object
cello_c_chord = pretty_midi.PrettyMIDI()
# Create an Instrument instance for a cello instrument
cello = pretty_midi.Instrument(program=pretty_midi.instrument_name_to_program('Cello'))
# Iterate over note names, which will be converted to note number later
for note_name in ['C5', 'E5', 'G5']:
    # Retrieve the MIDI note number for this note name
    note_number = pretty_midi.note_name_to_number(note_name)
    # Create a Note instance for this note, starting at 0s and ending at .5s
    note = pretty_midi.Note(velocity=100, pitch=note_number, start=0, end=.5)
    # Add it to our cello instrument
    cello.notes.append(note)
# Add the cello instrument to the PrettyMIDI object
cello_c_chord.instruments.append(cello)
# Write out the MIDI data
cello_c_chord.write('cello-C-chord.mid')
```