

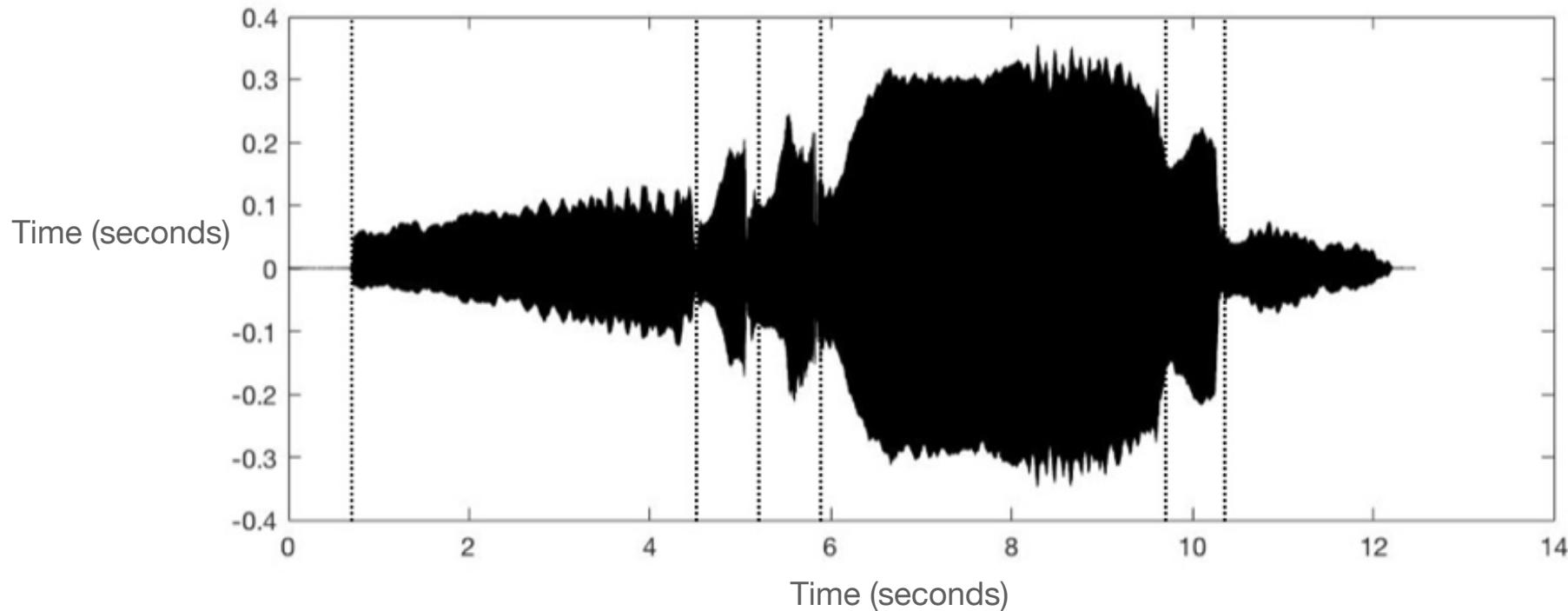
PD Workshop on Audio Analysis Tools

March 11, 2019
Johanna Devaney

<https://github.com/jcdevaney/audioAnalysisWorkshop>

Signal Processing Basics

Time-domain representation

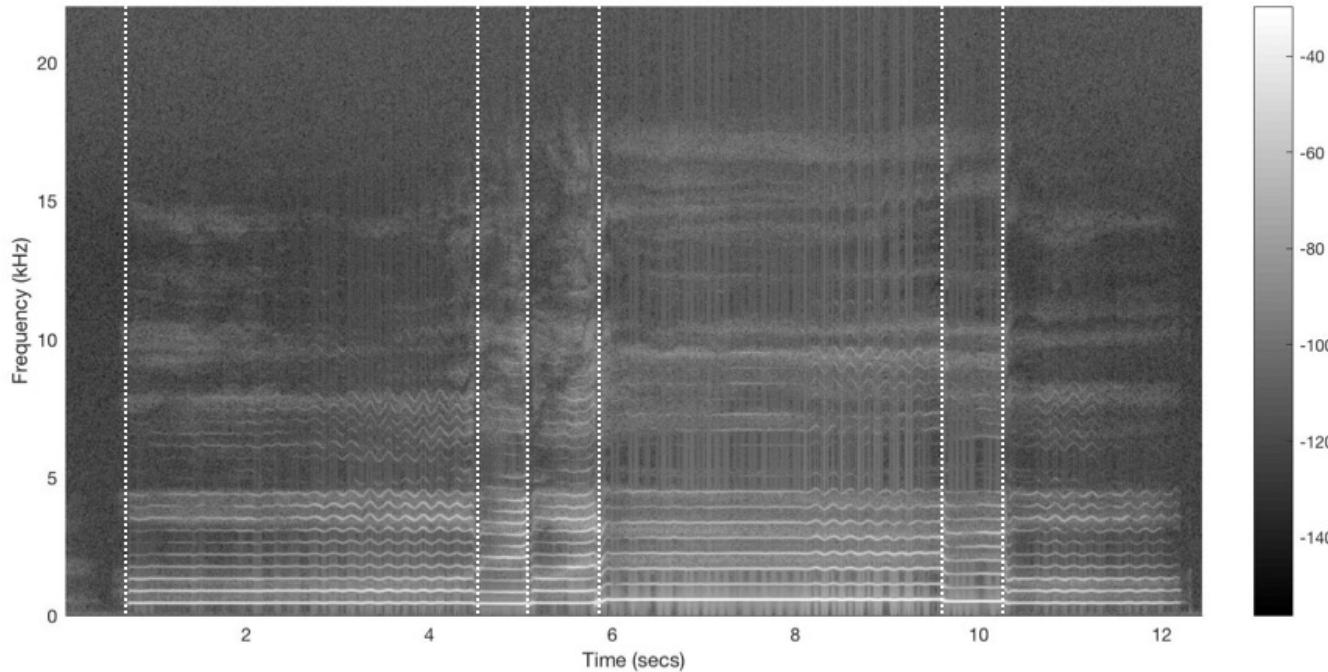


Time domain representation of an audio recording of the opening phrase of Schubert's "Ave Maria". The vertical dotted lines indicate the boundaries of the notes in the score below



Signal Processing Basics

Frequency-domain representation



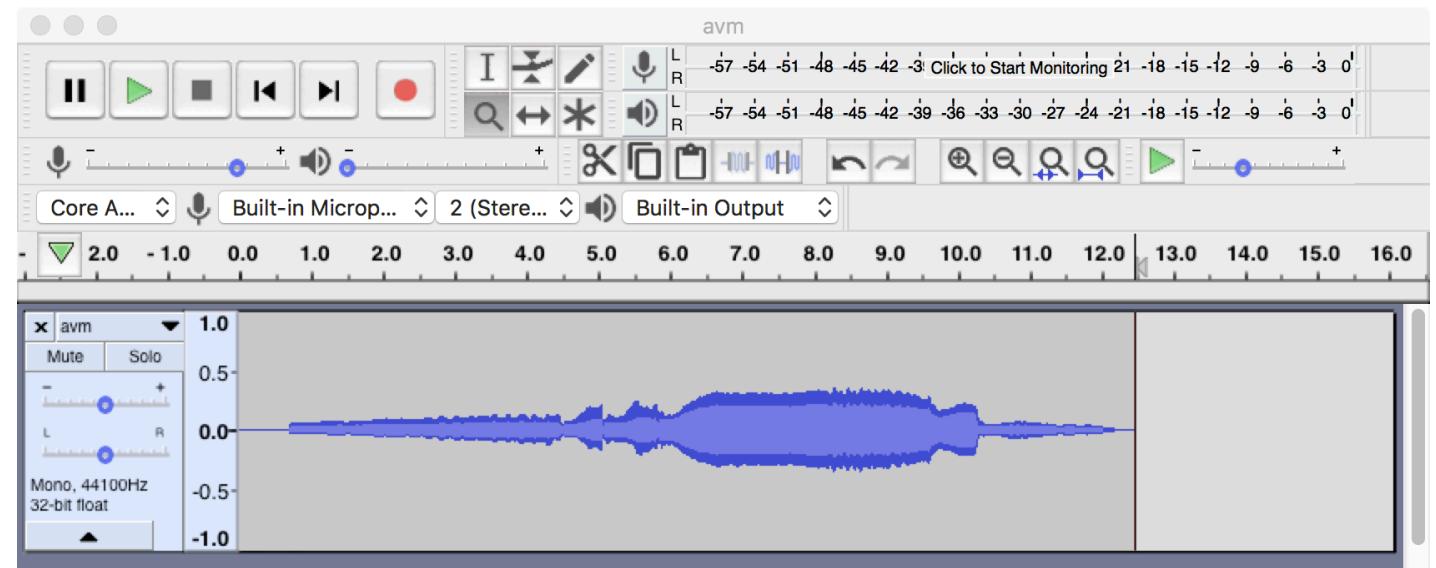
Spectrographic (frequency-domain) representation of an audio recording of the opening phrase of Schubert's “Ave Maria”.



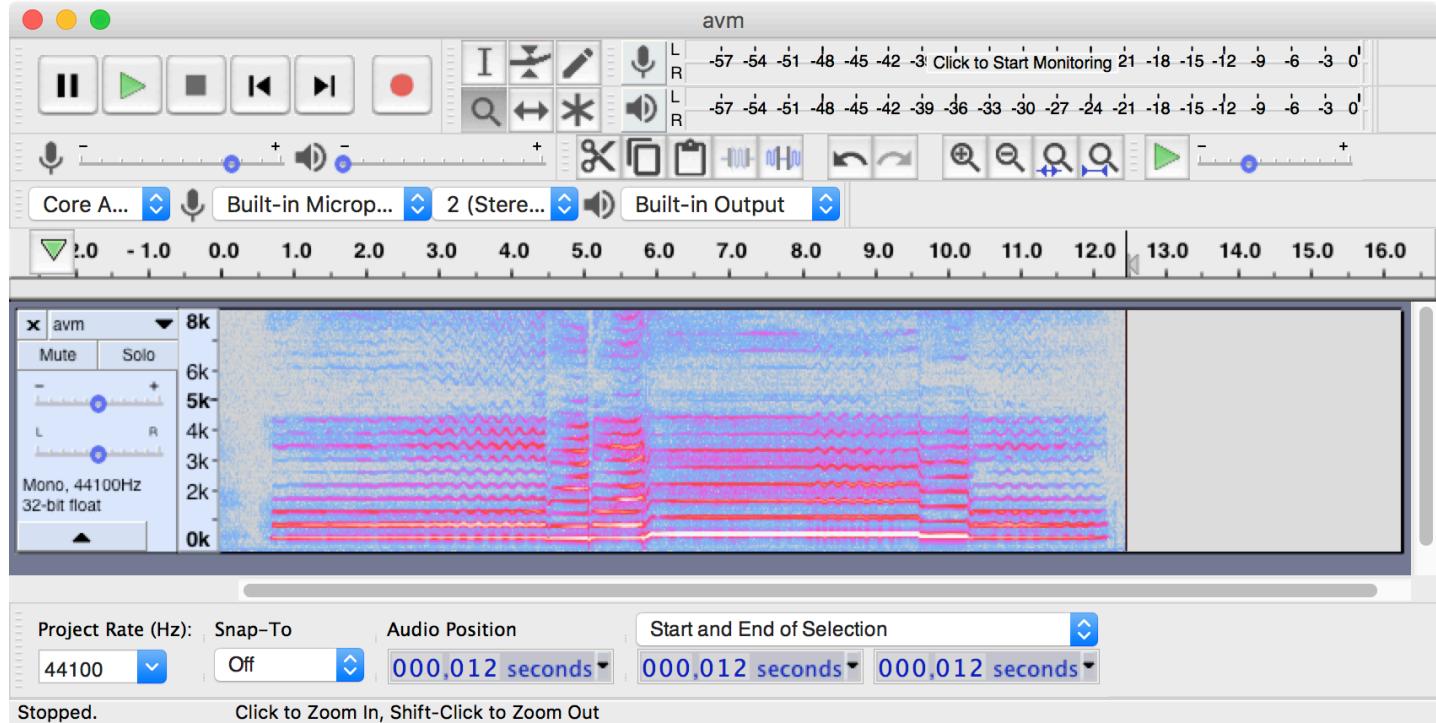
Audacity

Representations

Time-domain

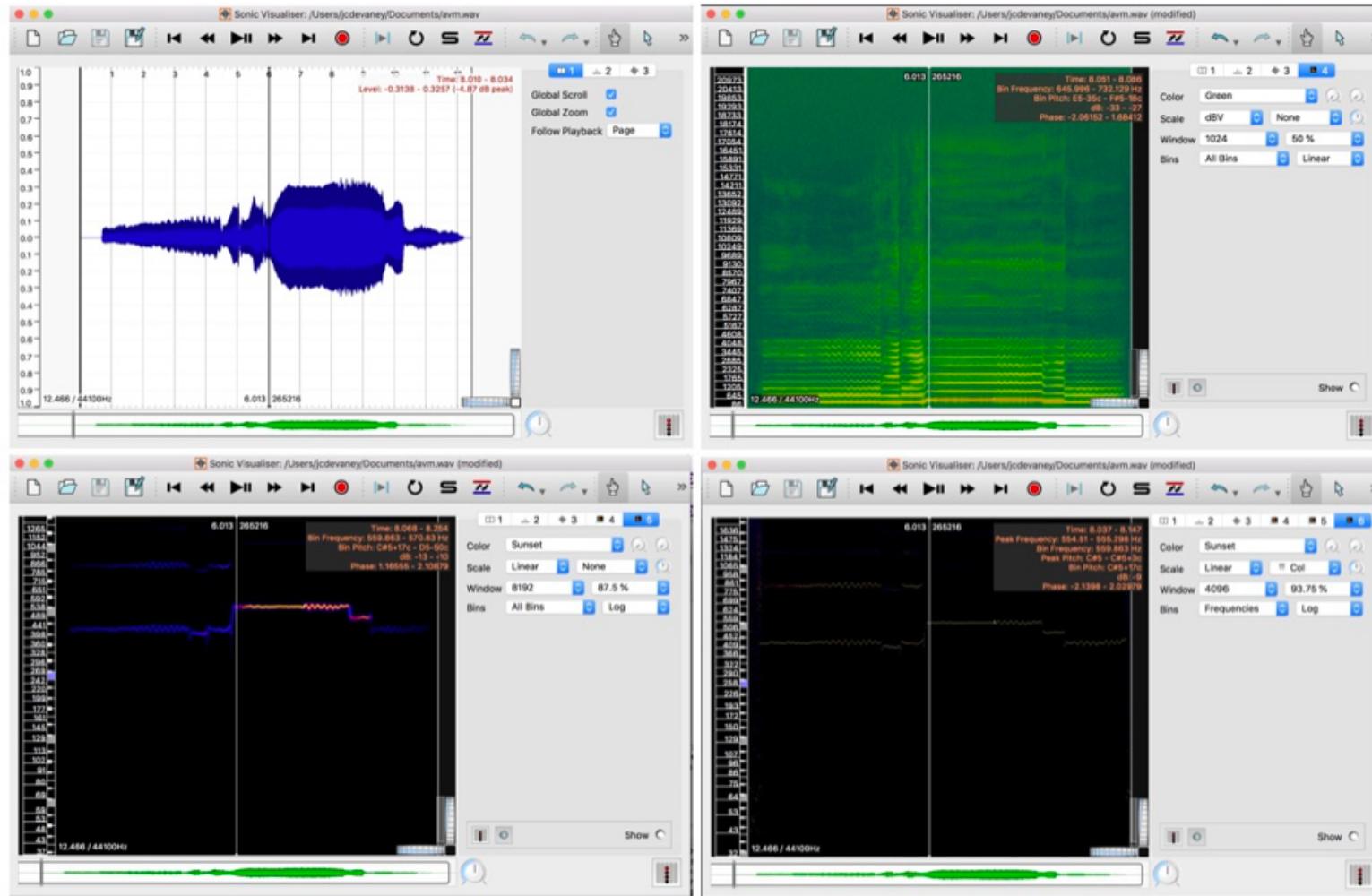


Frequency-domain



Sonic Visualiser

Representations

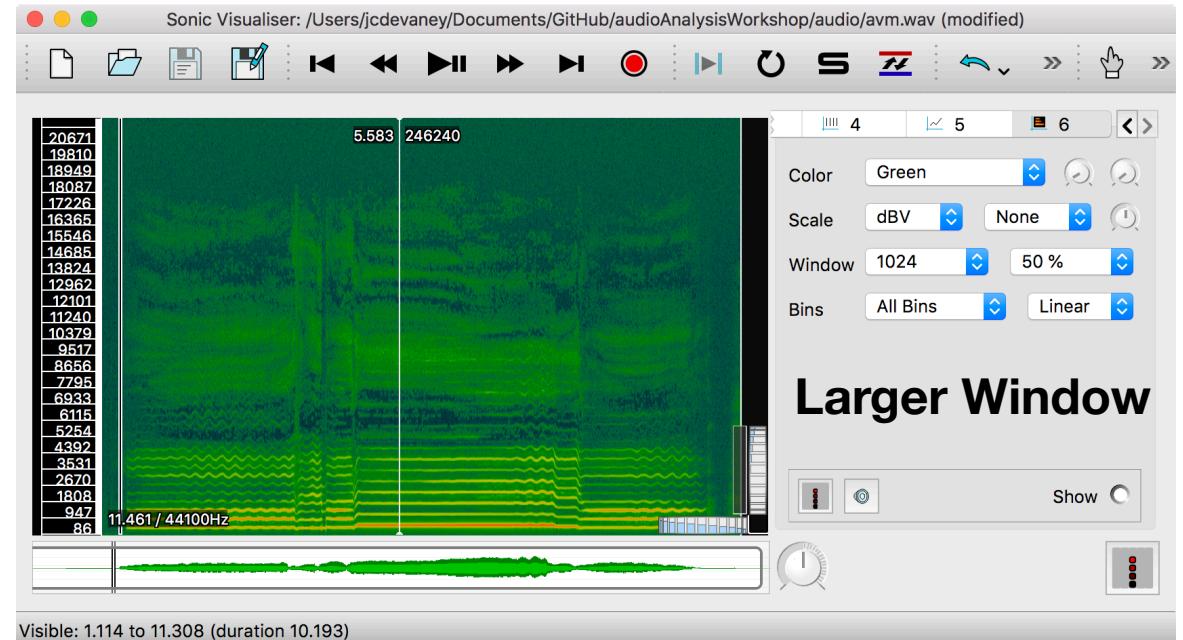


The top-left panel is a time-domain waveform, the top-right panel is a spectrogram, the bottom-left panel is a melody-range spectrogram, and the bottom-right spectrogram is a peak-frequency spectrogram.

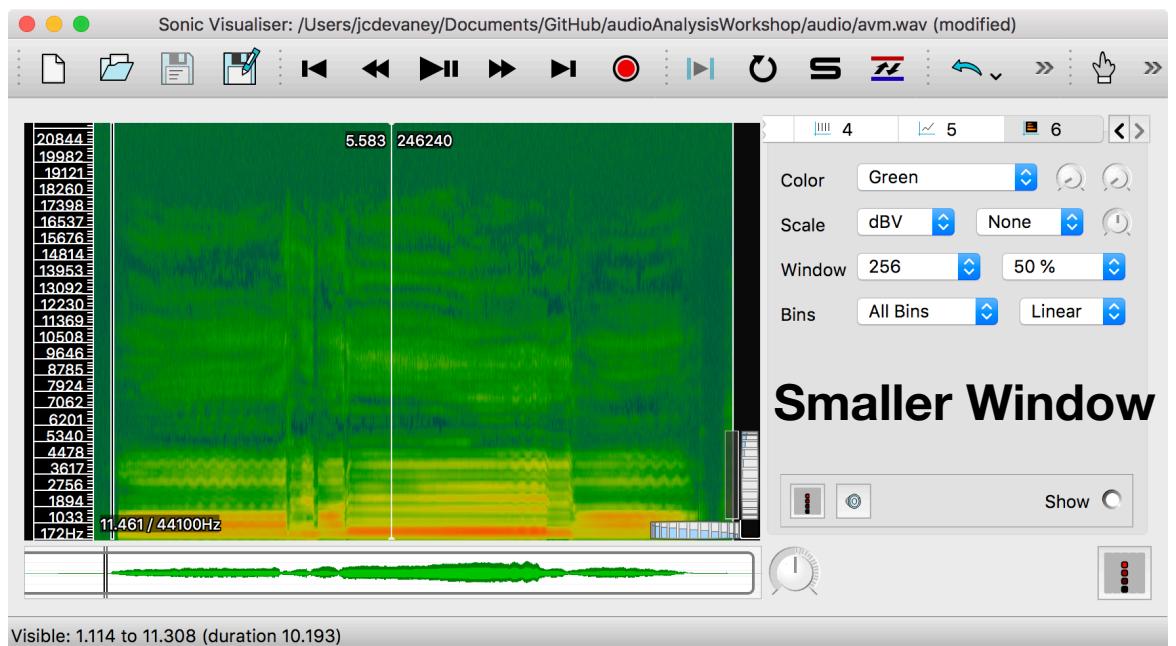
Signal Processing Basics

Spectrogram settings

A spectrogram is a series of spectral analysis (fast Fourier transforms) laid out in temporally overlapping “tiles”

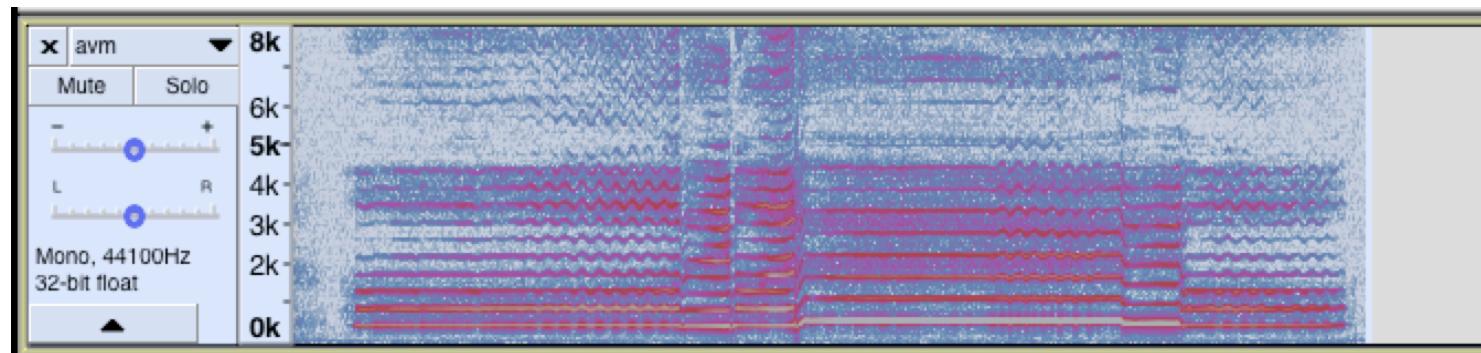


Each “tile” has a window size, which determines how much of the audio signal analyzed

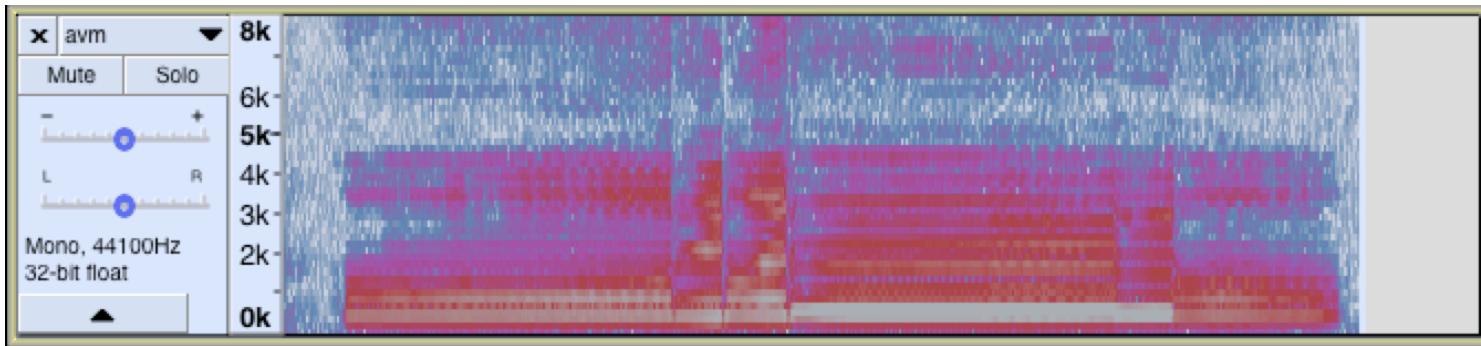


Signal Processing Basics

Spectrogram settings



Larger Window



Smaller Window

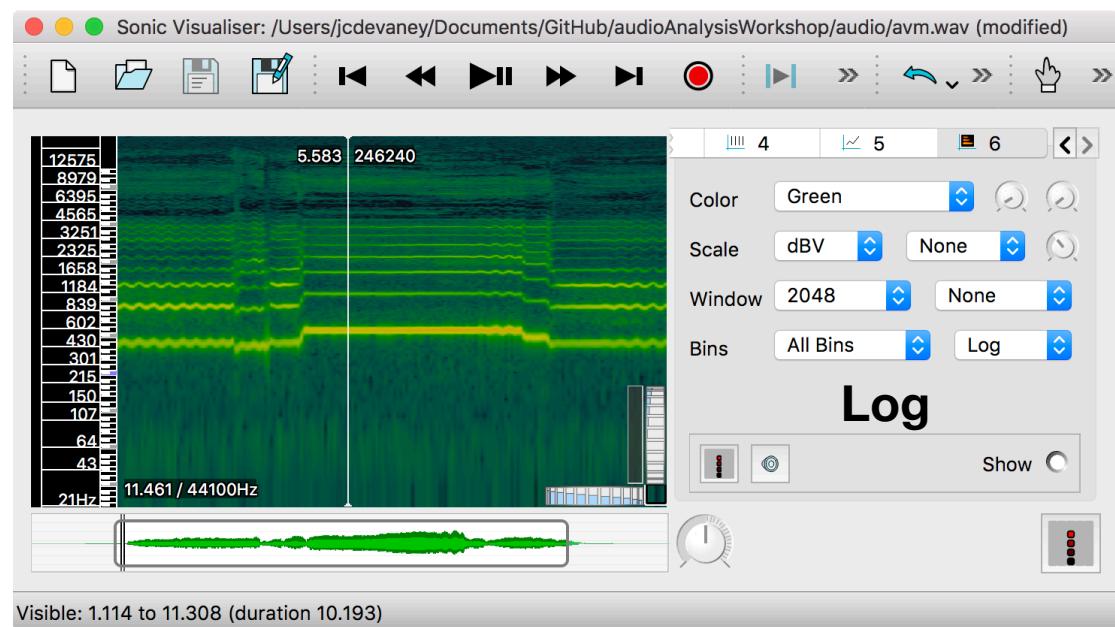
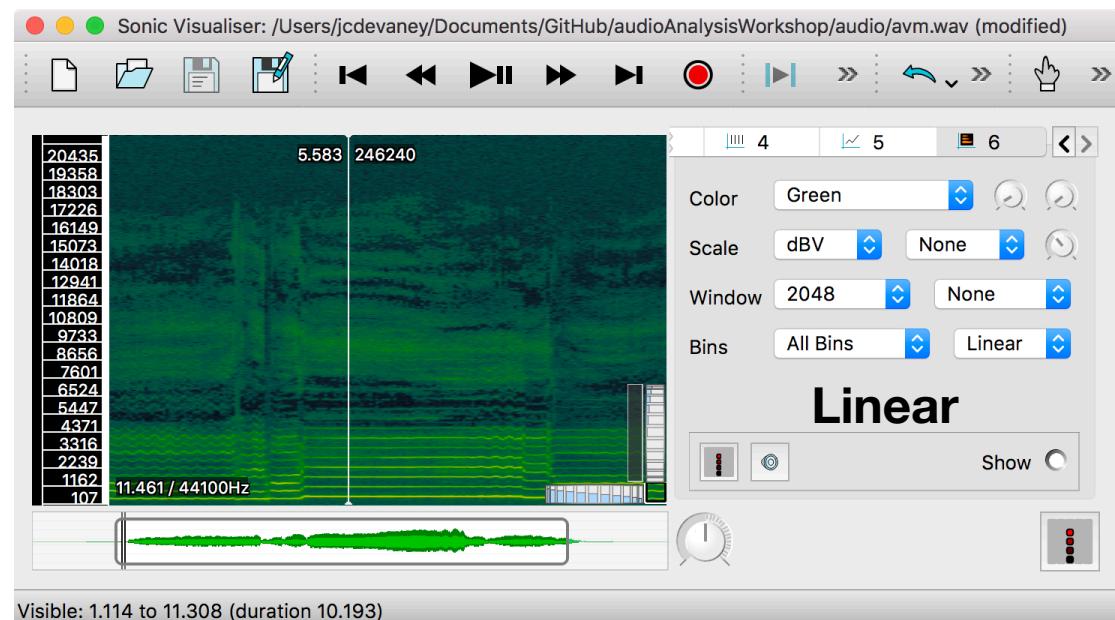
Signal Processing Basics

Spectrogram settings

Spectrograms often default to a linear spacing of frequencies

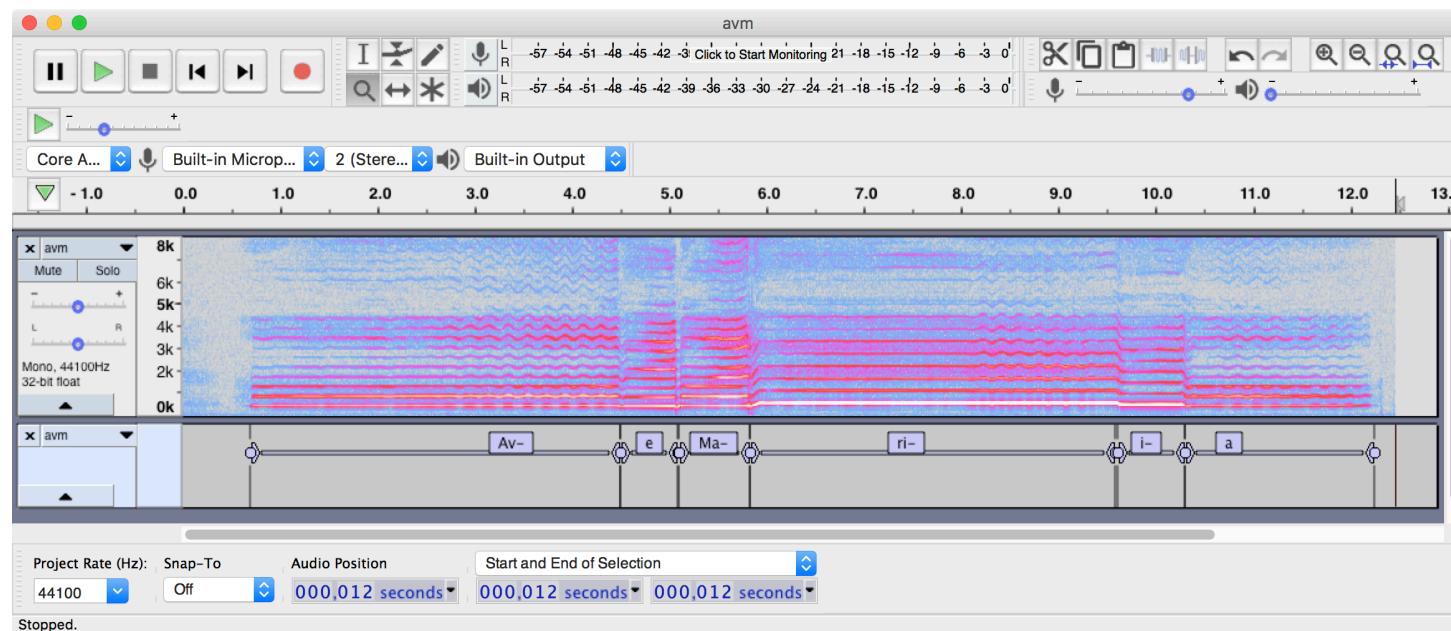
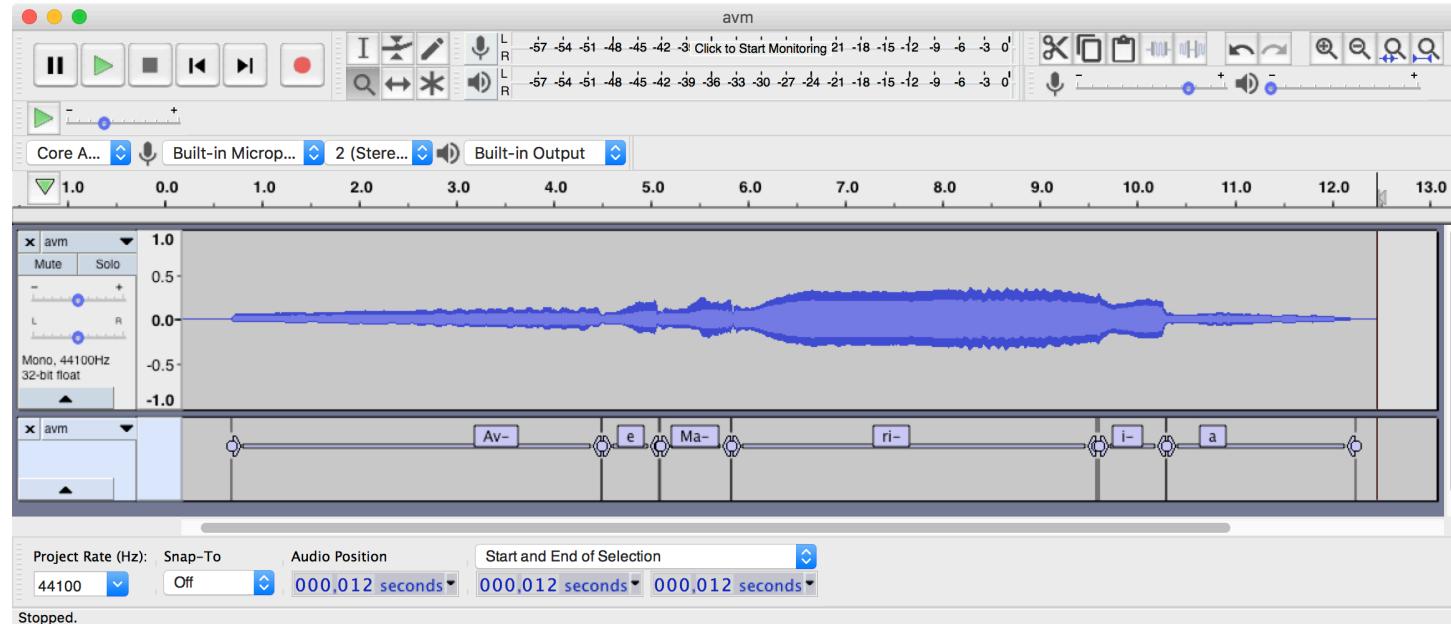
Humans hear frequencies logarithmically (a doubling of frequency is perceived as an octave)

Log-spacing in spectrograms can help observe harmonic and timbral characteristics



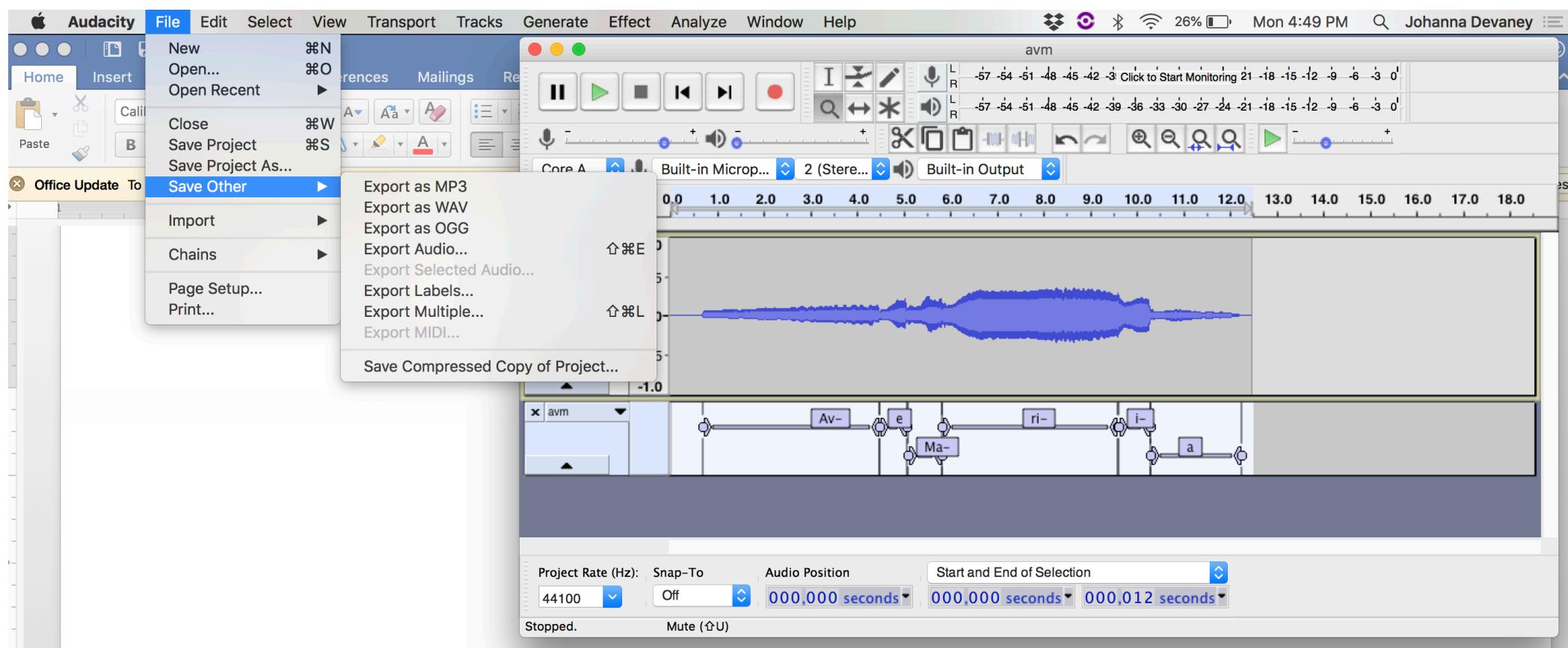
Audacity

Annotations



Audacity

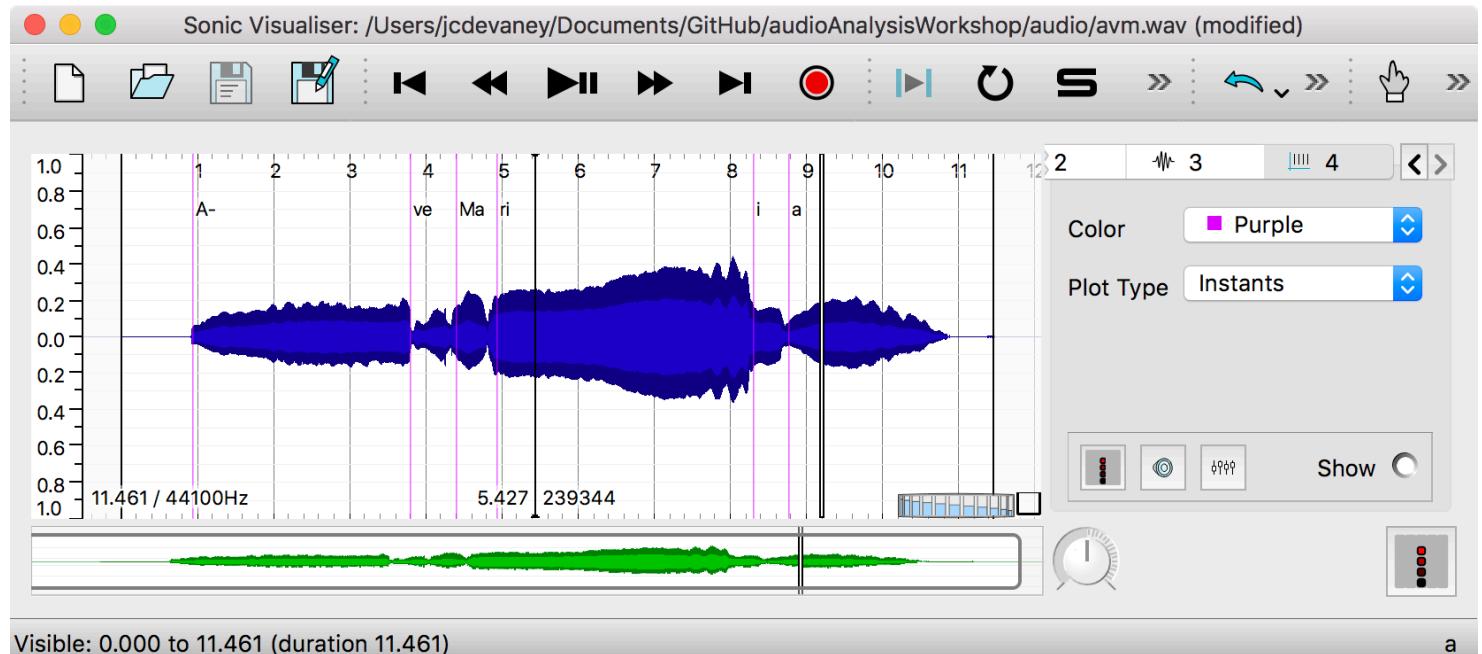
Annotations



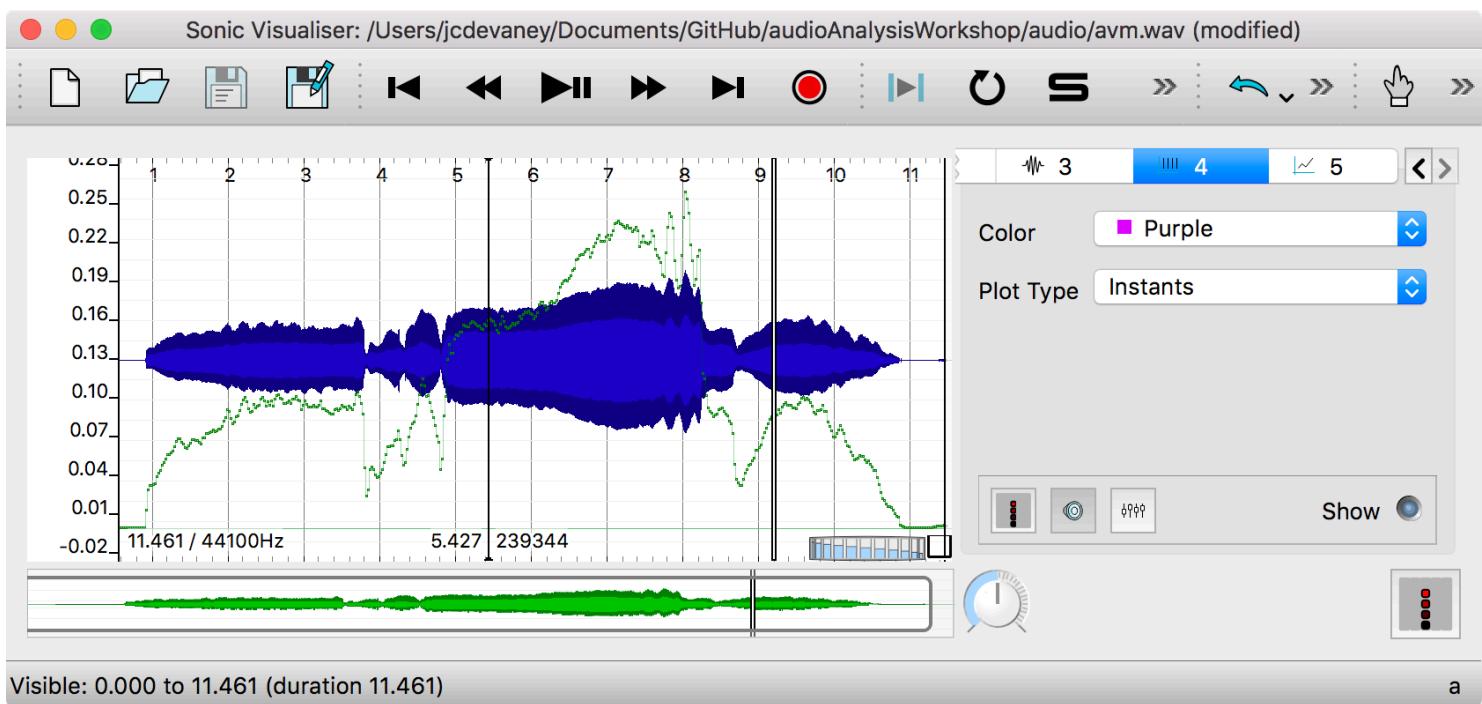
Sonic Visualiser

Annotations

Time Instants
(time points)



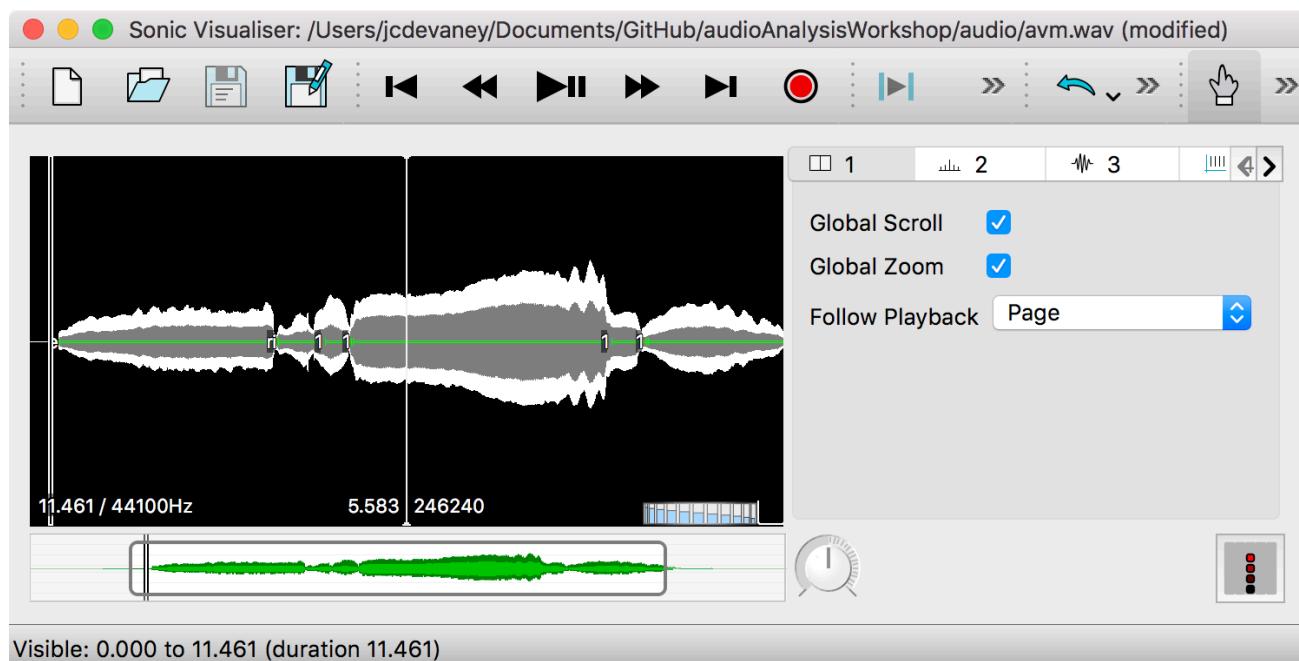
Time Values
(time points + values)



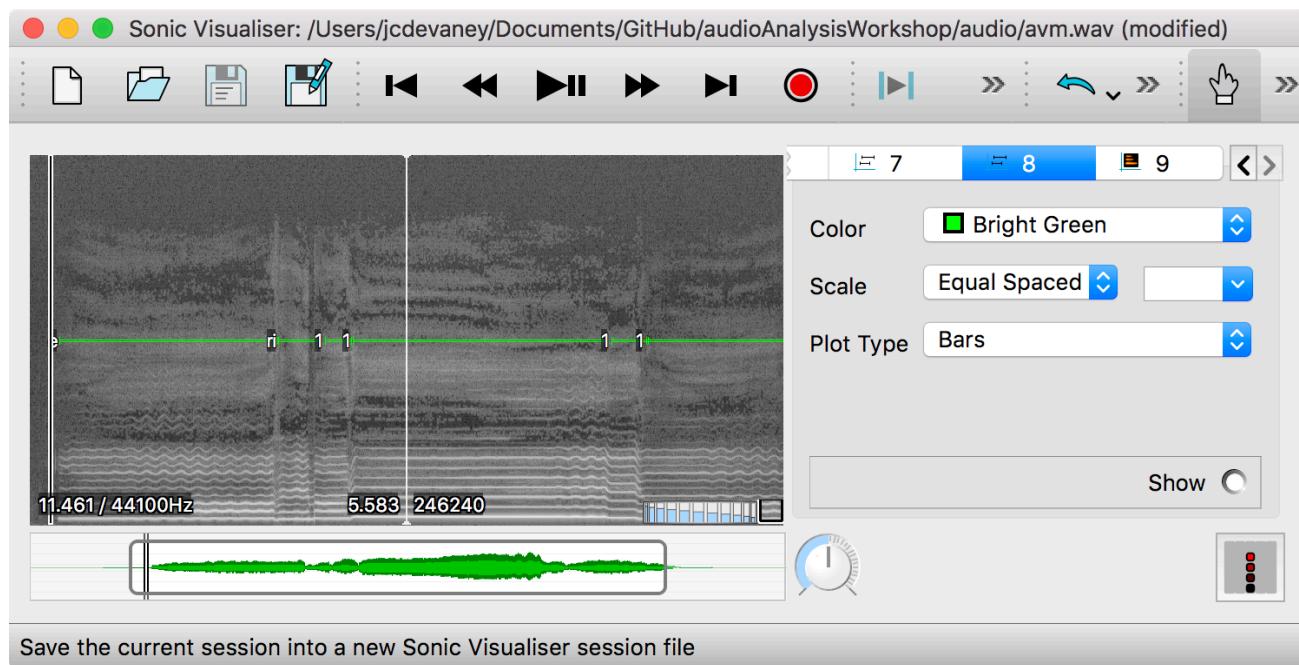
Sonic Visualiser

Annotations

Regions
(Time-domain
overlay)

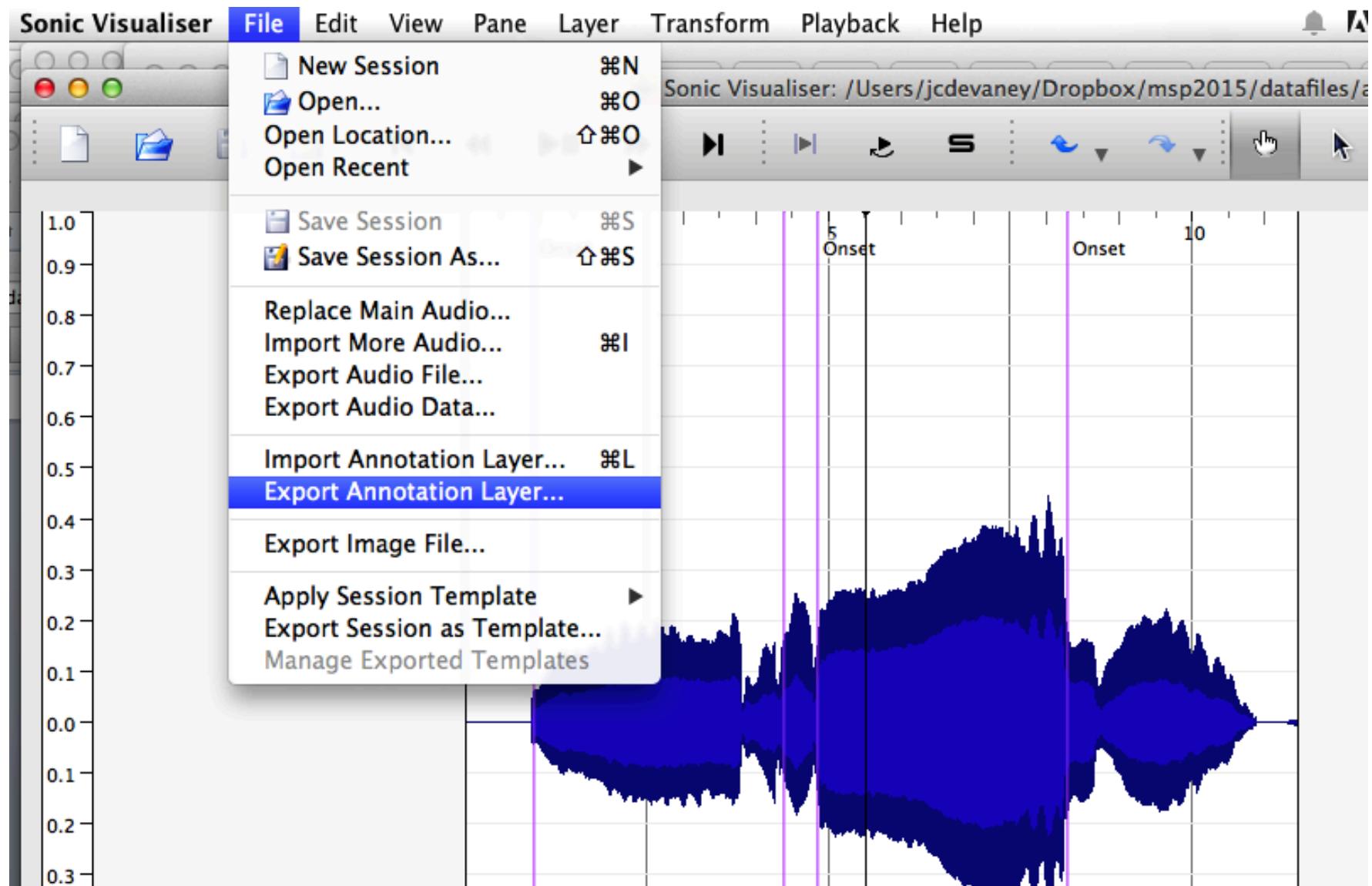


Regions
(Frequency-
domain
overlay)



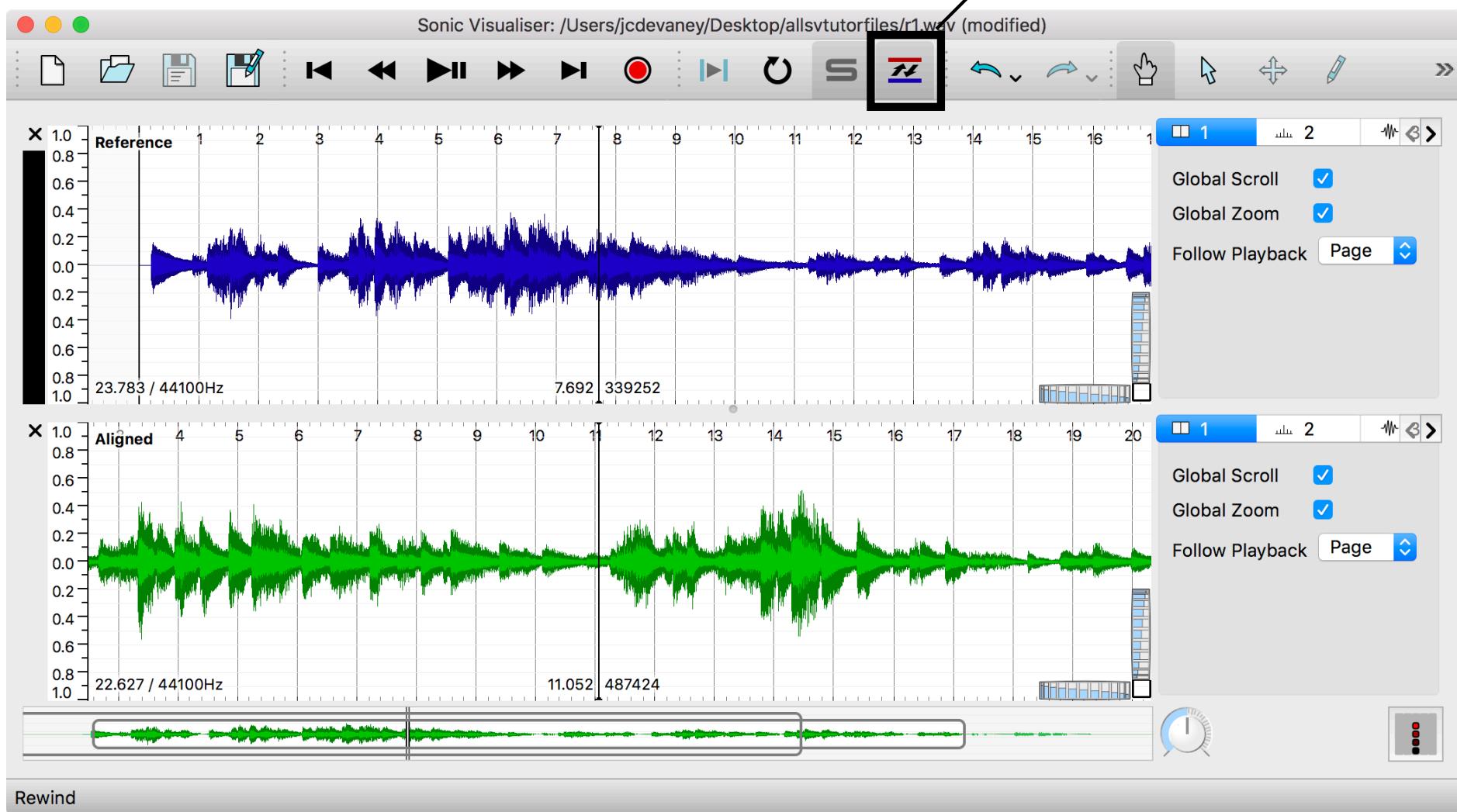
Sonic Visualiser

Annotations



Sonic Visualiser

Time Alignment



Onset Detection

Theory

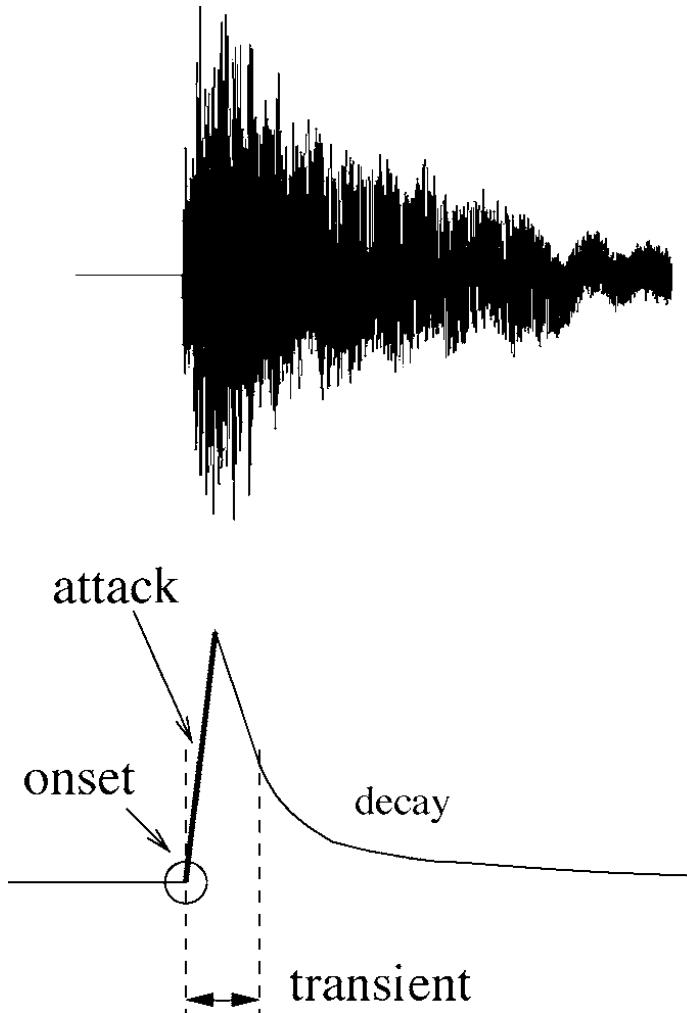


Fig. 1. “Attack,” “transient,” “decay,” and “onset” in the ideal case of a single note.

Onset Detection

Theory

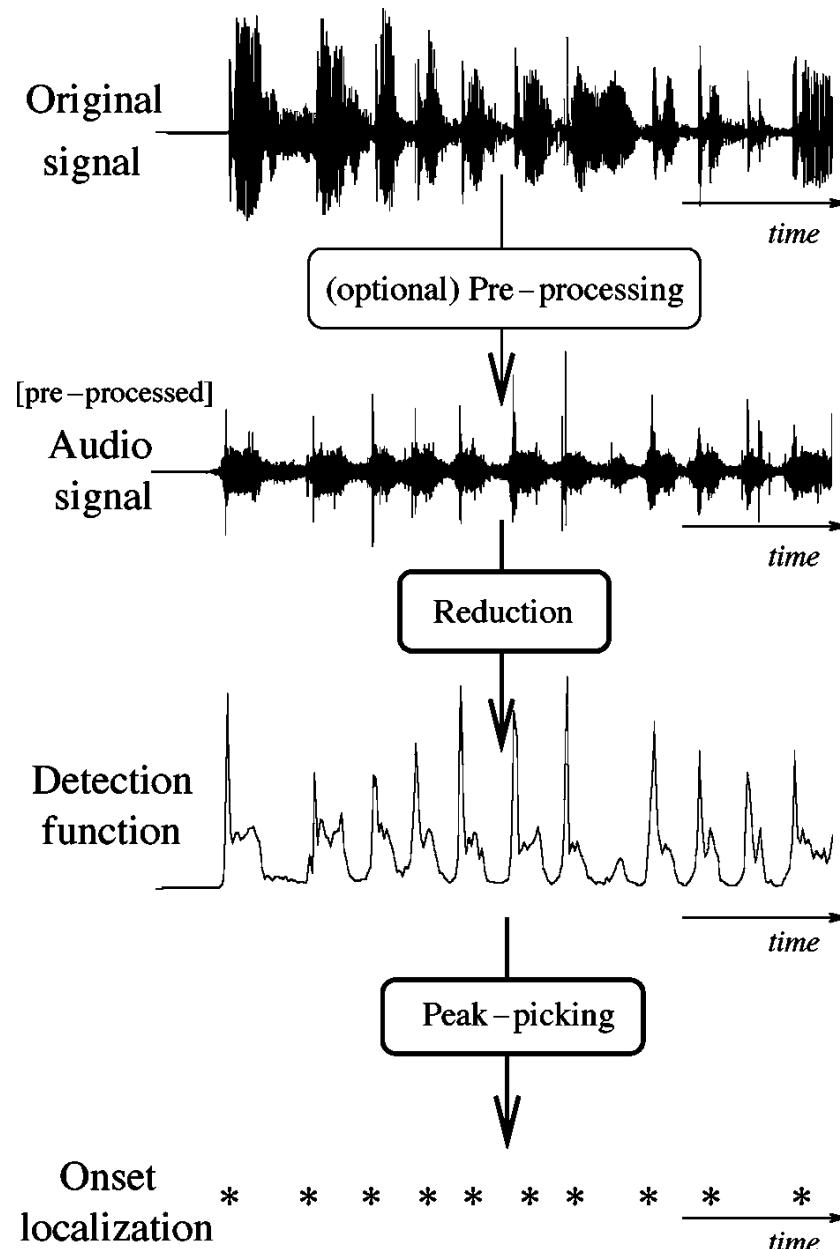


Fig. 2. Flowchart of a standard onset detection algorithm.

Onset Detection

Theory

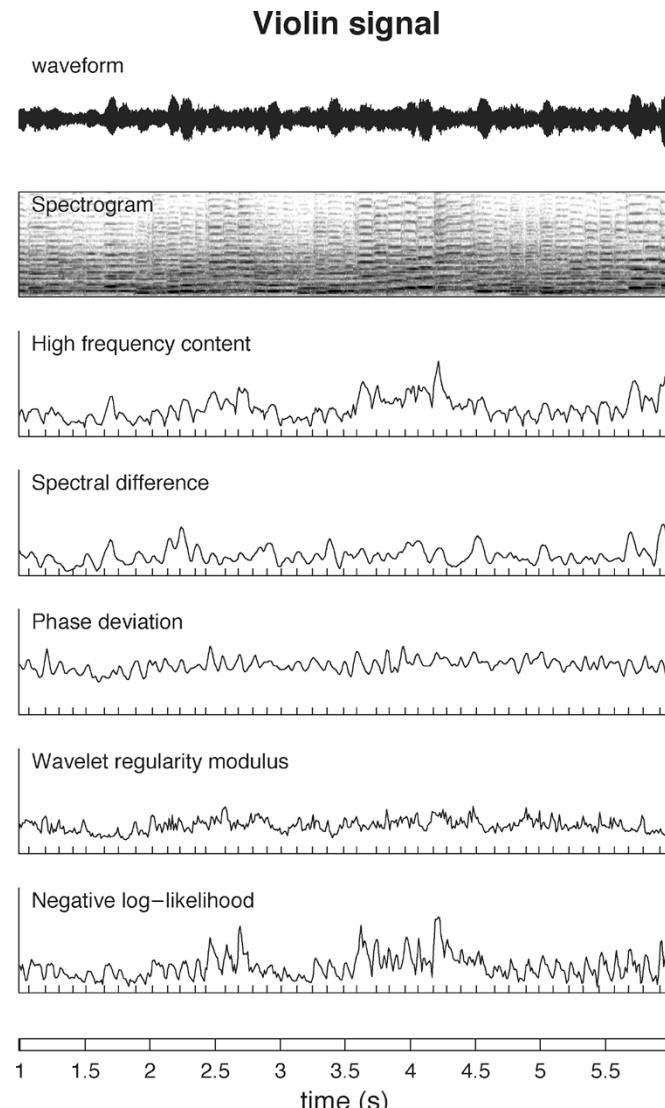


Fig. 4. Comparison of different detection functions for 5 s of a solo violin recording. From top to bottom: time-domain signal, spectrogram, high-frequency content, spectral difference, spread of the distribution of phase deviations, wavelet regularity modulus, and negative log-likelihood using an ICA model. All detection functions have been normalized to their maximum value.

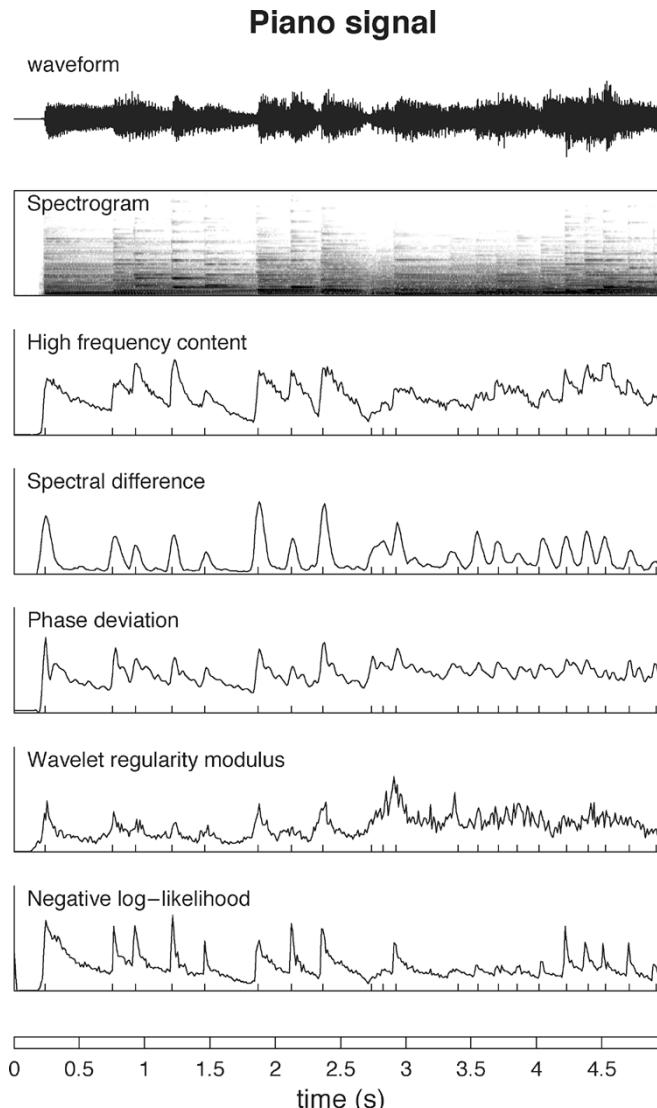


Fig. 5. Comparison of different detection functions for 5 s of a solo piano recording. From top to bottom: time-domain signal, spectrogram, high-frequency content, spectral difference, spread of the distribution of phase deviations, wavelet regularity modulus, and negative log-likelihood using an ICA model. All detection functions have been normalized to their maximum value.

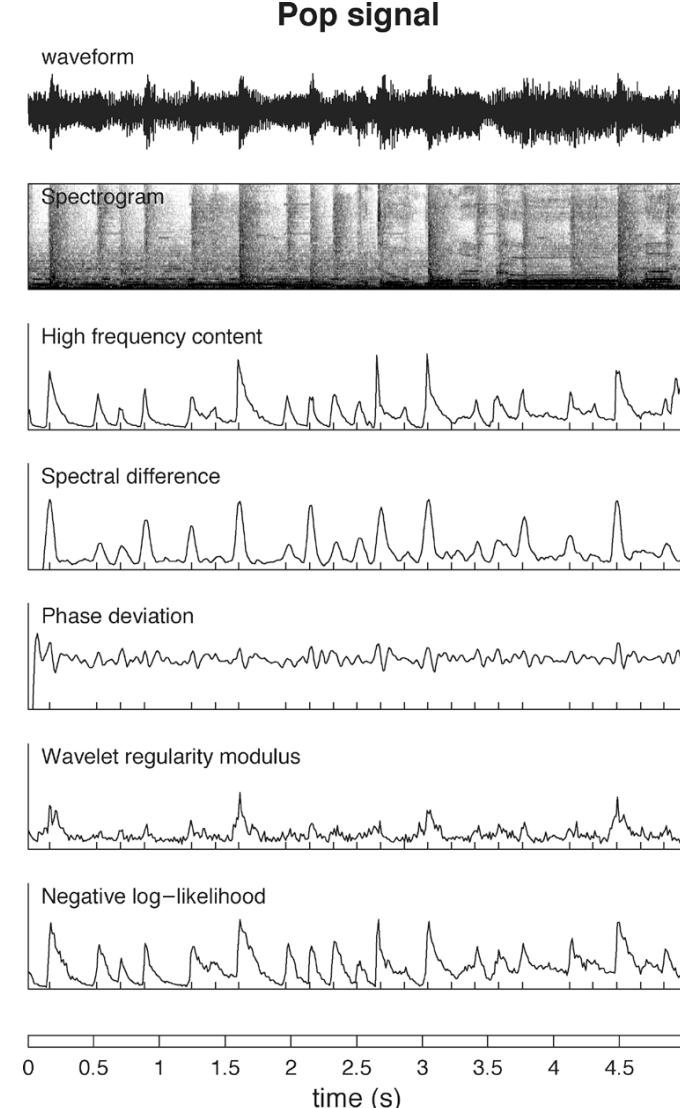


Fig. 6. Comparison of different detection functions for 5 s of a pop song. From top to bottom: time-domain signal, spectrogram, high-frequency content, spectral difference, spread of the distribution of phase deviations, wavelet regularity modulus, and negative log-likelihood using an ICA model. All detection functions have been normalized to their maximum value.

Onset Detection

Sonic Visualiser

- ▶ **University of Alicante: signal approach to reduction**

- pre-processing one-semitone filter bank
- outputs
 - onset detection function
 - note onset estimates
- parameters
 - sensitivity
 - processing option: audio frames per block (window size), window increment (hop)

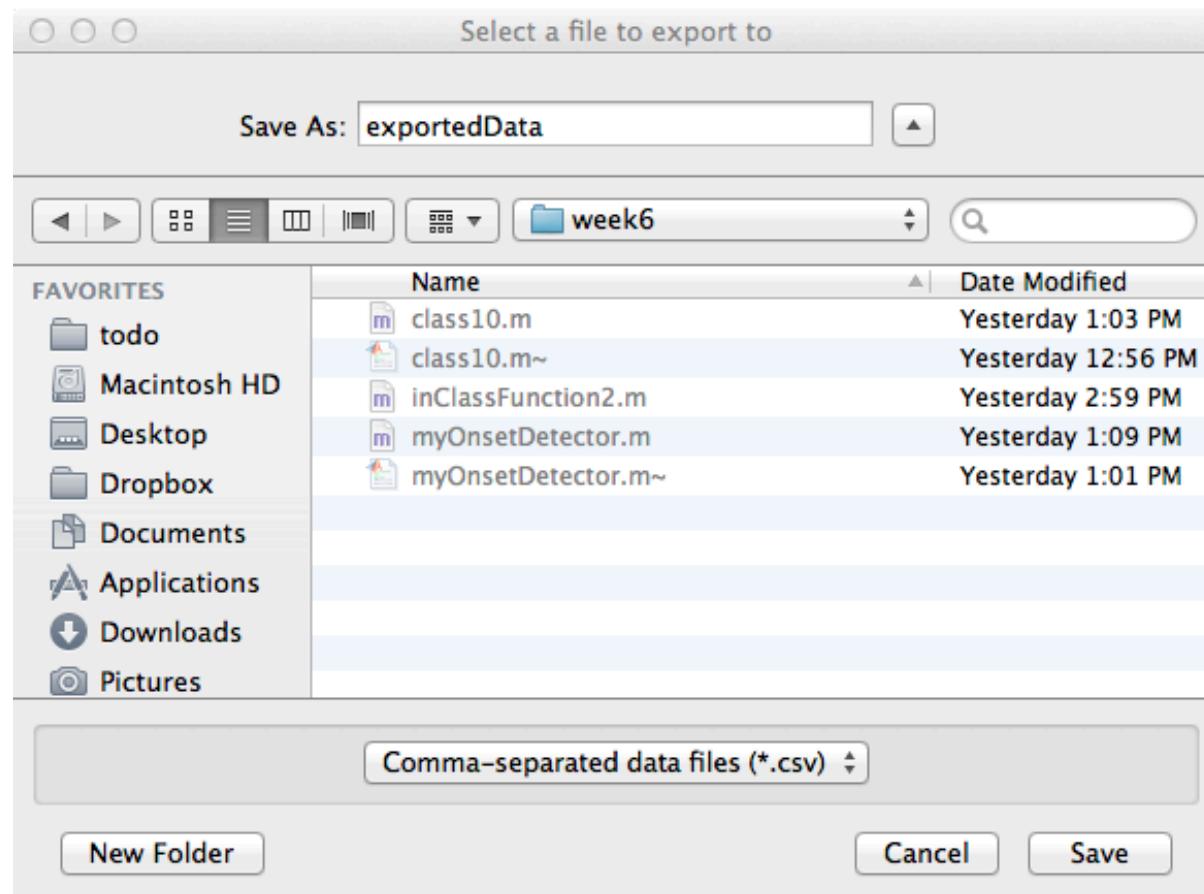
Onset Detection

Sonic Visualiser

- ▶ **Queen Mary: probabilistic approach to reduction**
 - outputs
 - onset detection function
 - smoothed detection function
 - note onset estimates
 - parameters
 - program: general, soft onsets, percussive onsets
 - function type: high-frequency content, spectral difference, phase deviation, complex domain, broadband energy rise
 - processing options: window size, window increment (hop), window shape (Hann, Hamming, Blackman, Blackman-Hanning, Nuttall, Gaussian, Parzen, triangular, rectangular)

Onset Detection

Sonic Visualiser



Sonic Annotator

- ▶ **For each plugin you will need to create a .n3 file, using the -s flag to specify a plugin. This allows you to batch process audio files with a particular plugin with particular settings (although for this assignment just use the default values)**
 - E.g., `./sonic-annotator -s vamp:bbc-vamp-plugins:bbc-energy:rmsenergy > rms.n3`

Sonic Annotator

- ▶ You will then need to put all of the audio file in the same directory and run each .n3 file on the directory using the -r flag to specify the directory, -t flag to specify the .n3 file, and the -w flag to specify csv (which writes to a CSV file)
 - E.g., ./sonic-annotator -r ~/Documents/GitHub/audioAnalysisWorkshop/audio -t rms.n3 -w csv