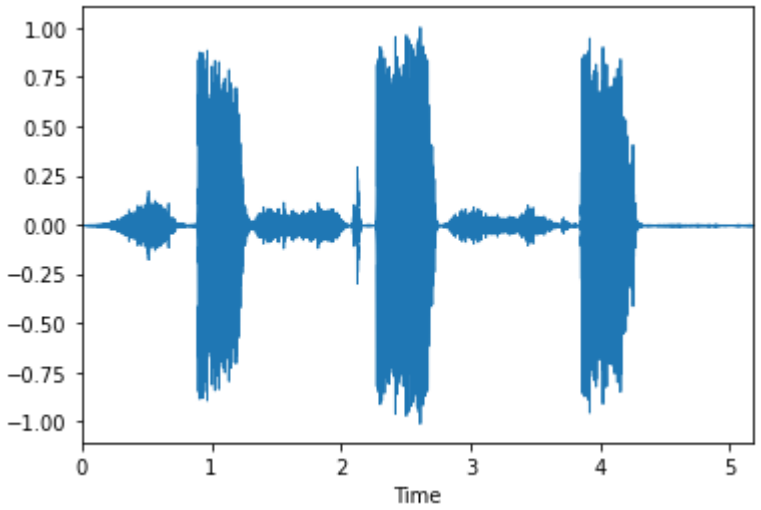
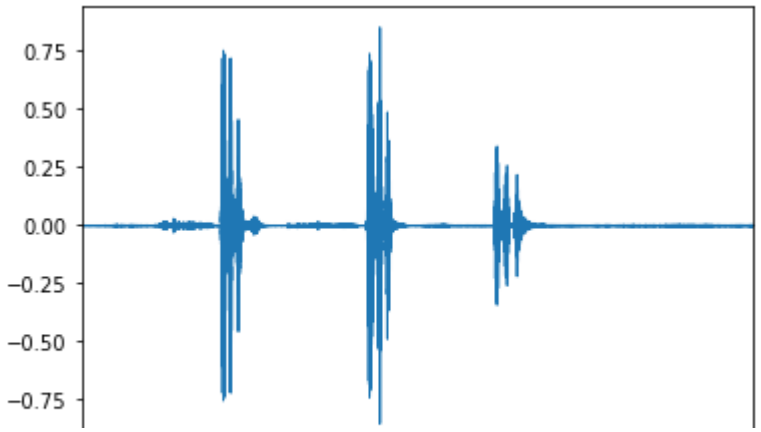
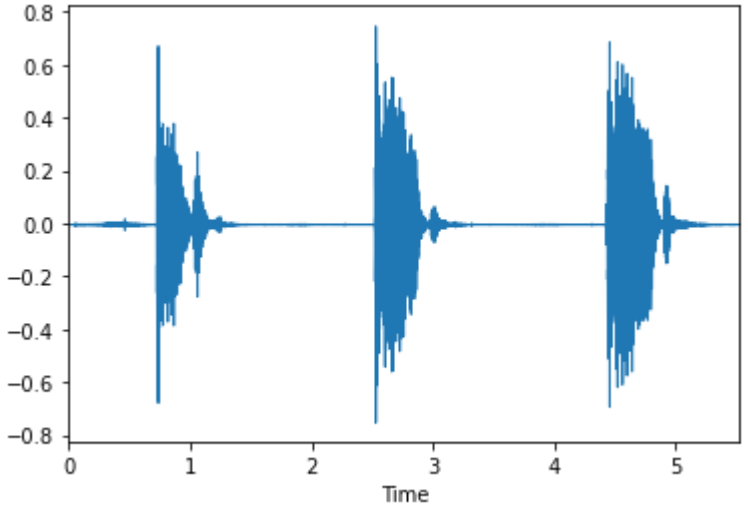
```
import librosa
import librosa.display

filename = '/content/cough-1.wav'
y, sr = librosa.load(filename)
# trim silent edges
whale_song, _ = librosa.effects.trim(y)
librosa.display.waveplot(whale_song, sr=sr);
```




```
import librosa
import librosa.display

filename = '/content/Covid cough.wav'
y, sr = librosa.load(filename)
# trim silent edges
whale_song, _ = librosa.effects.trim(y)
librosa.display.waveplot(whale_song, sr=sr);
```

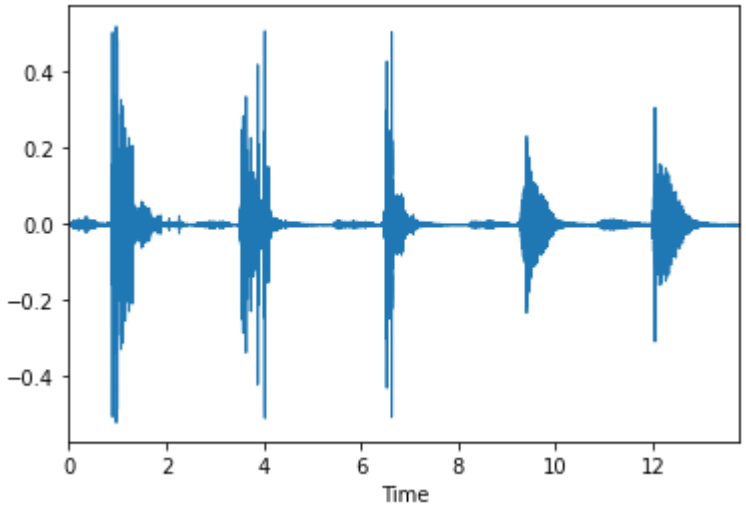



```
import librosa
import librosa.display

filename = '/content/cough_8PmvbJ4U3o_1587970252151.wav'
y, sr = librosa.load(filename)
# trim silent edges
whale_song, _ = librosa.effects.trim(y)
librosa.display.waveplot(whale_song, sr=sr);
```

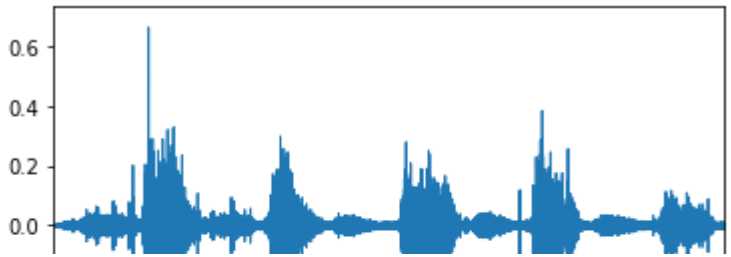



```
import librosa
import librosa.display
```

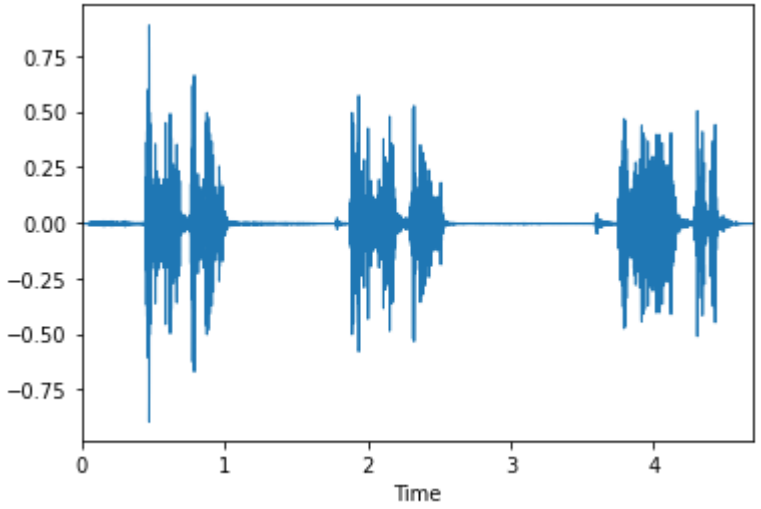
```
filename = '/content/breaths_0c4dx8rU5G_1586982341272.wav'
y, sr = librosa.load(filename)
# trim silent edges
whale_song, _ = librosa.effects.trim(y)
librosa.display.waveplot(whale_song, sr=sr);
```




```
import librosa
import librosa.display

filename = '/content/breaths_rHKBZMvTCd_1589986348444.wav'
y, sr = librosa.load(filename)
# trim silent edges
kk, _ = librosa.effects.trim(y)
librosa.display.waveplot(kk, sr=sr);
```

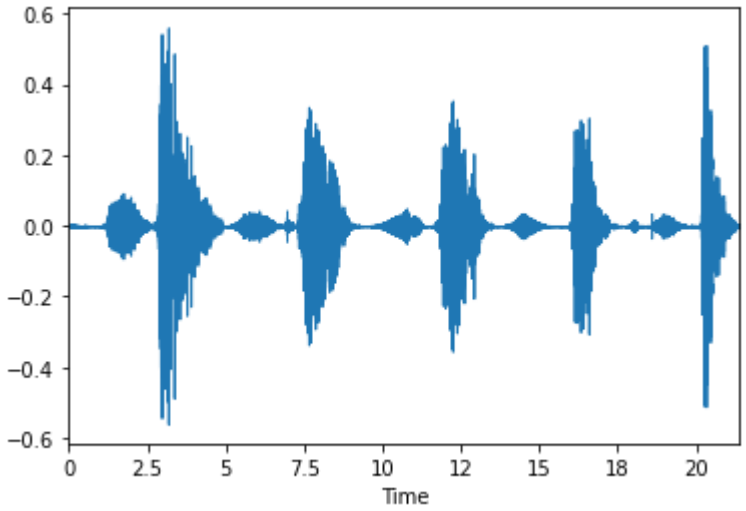



```
import librosa
import librosa.display

filename = '/content/cough_2aSAZx0fOr_1586937599126_mono.wav'
y, sr = librosa.load(filename)
# trim silent edges
kk, _ = librosa.effects.trim(y)
librosa.display.waveplot(kk, sr=sr);
```



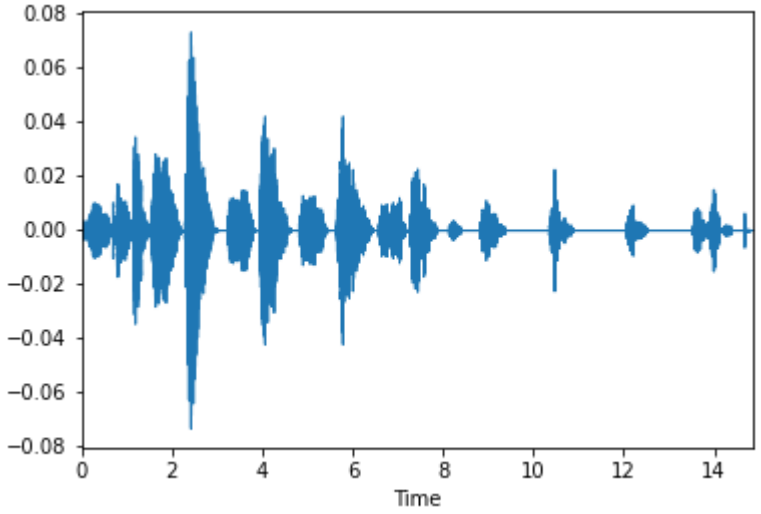

```
import librosa
import librosa.display

filename = '/content/breaths_2aSAZx0fOr_1587017671806.wav'
y, sr = librosa.load(filename)
# trim silent edges
```

```
kk, _ = librosa.effects.trim(y)
librosa.display.waveplot(kk, sr=sr);
```

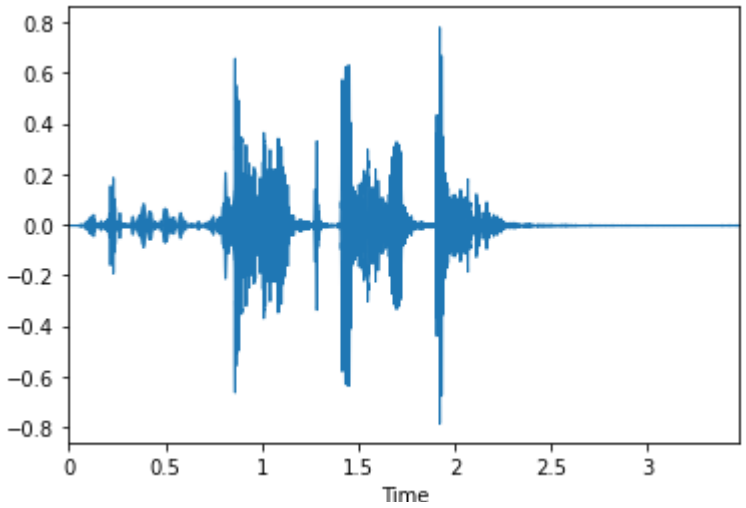



```
import librosa
import librosa.display

filename = '/content/breaths_1fxxVyop57_1589091470099.wav'
y, sr = librosa.load(filename)
# trim silent edges
kk, _ = librosa.effects.trim(y)
librosa.display.waveplot(kk, sr=sr);
```

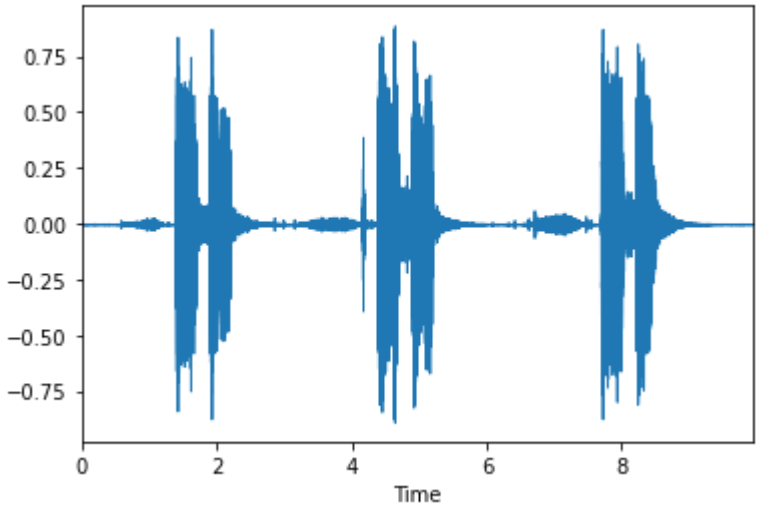

```
import librosa
import librosa.display

filename = '/content/breaths_mYtwaX7NGZ_1588578339313.wav'
y, sr = librosa.load(filename)
# trim silent edges
kk, _ = librosa.effects.trim(y)
librosa.display.waveplot(kk, sr=sr);
```



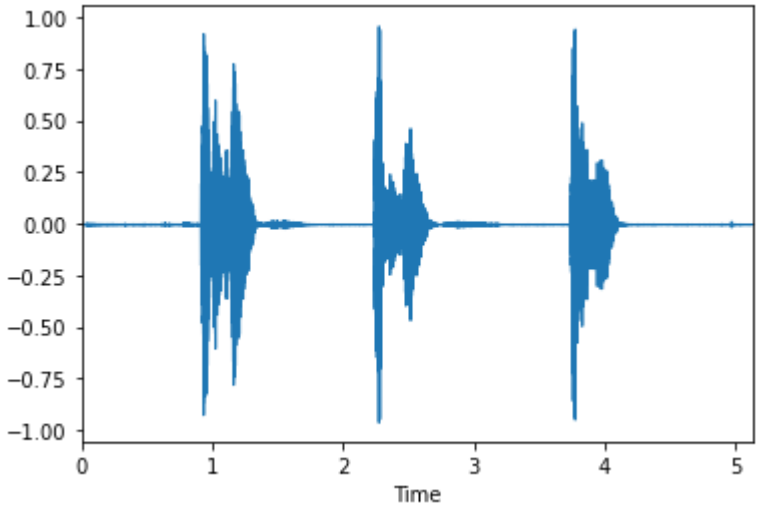

```
import librosa
import librosa.display

filename = '/content/cough_6hVSzUlymW_1588001367930.wav'
y, sr = librosa.load(filename)
# trim silent edges
kk, _ = librosa.effects.trim(y)
librosa.display.waveplot(kk, sr=sr);
```



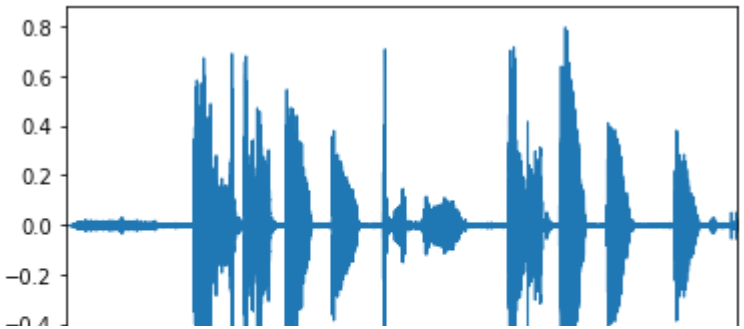

```
import librosa
import librosa.display

filename = '/content/cough_Fxiwf2vW03_1587798762166.wav'
y, sr = librosa.load(filename)
# trim silent edges
kk, _ = librosa.effects.trim(y)
librosa.display.waveplot(kk, sr=sr);
```



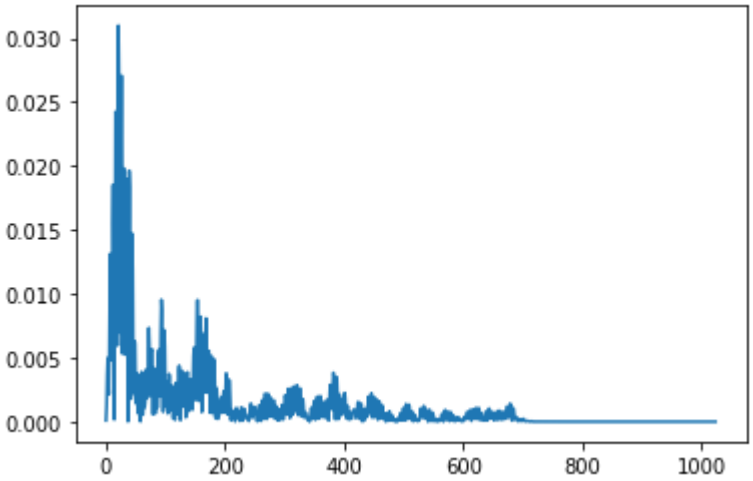
```
import librosa
import librosa.display

filename = '/content/cough_yfW1c2NlCx_1586974588221.wav_aug_amp2.wav'
y, sr = librosa.load(filename)
# trim silent edges
kk, _ = librosa.effects.trim(y)
librosa.display.waveplot(kk, sr=sr);
```



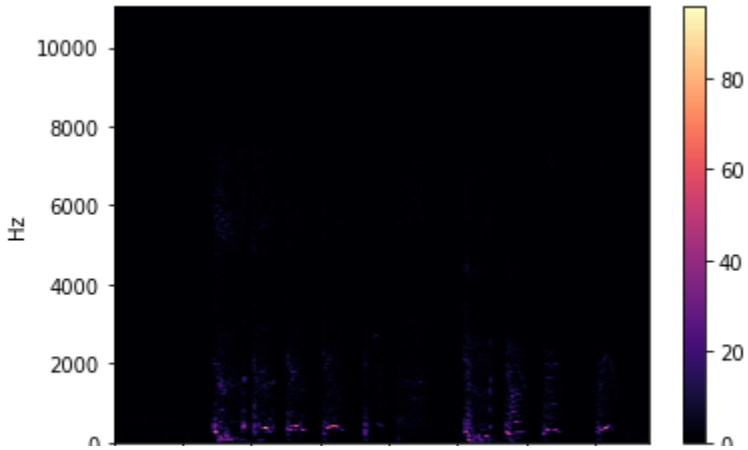

```
import librosa
import librosa.display

filename = '/content/cough_crxRiqIPHi_1587914038132.wav'
y, sr = librosa.load(filename)
# trim silent edges
kk, _ = librosa.effects.trim(y)
librosa.display.waveplot(kk, sr=sr);
```



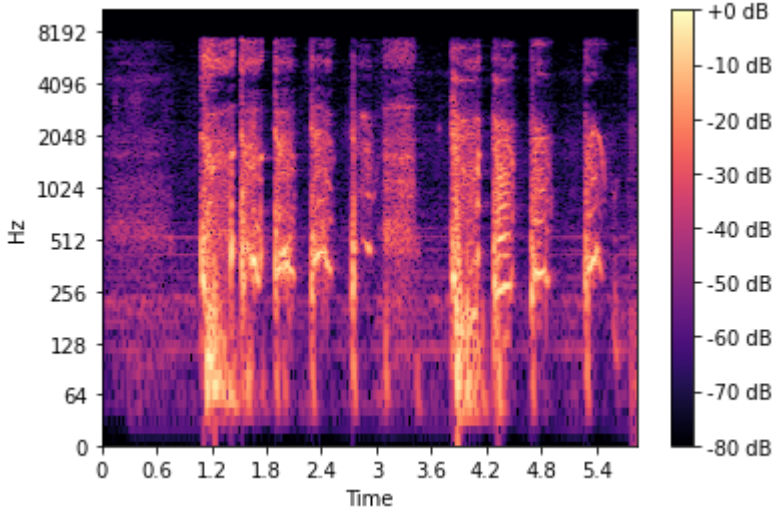

```
import numpy as np
import matplotlib.pyplot as plt

n_fft = 2048
D = np.abs(librosa.stft(kk[:n_fft], n_fft=n_fft, hop_length=n_fft+1))
plt.plot(D);
```




```
hop_length = 256
D = np.abs(librosa.stft(kk, n_fft=n_fft,  hop_length=hop_length))
librosa.display.specshow(D, sr=sr, x_axis='time', y_axis='linear');
plt.colorbar();
```



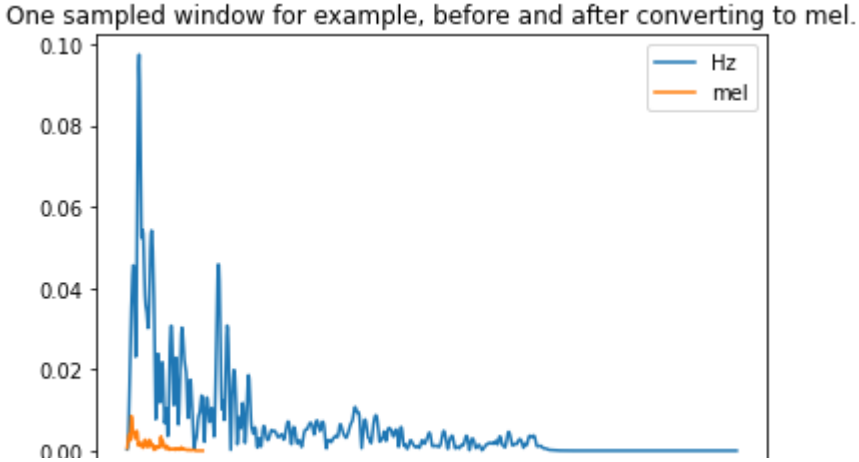

```
DB = librosa.amplitude_to_db(D, ref=np.max)
librosa.display.specshow(DB, sr=sr, hop_length=hop_length, x_axis='time', y_axis=
plt.colorbar(format='%+2.0f dB');
```



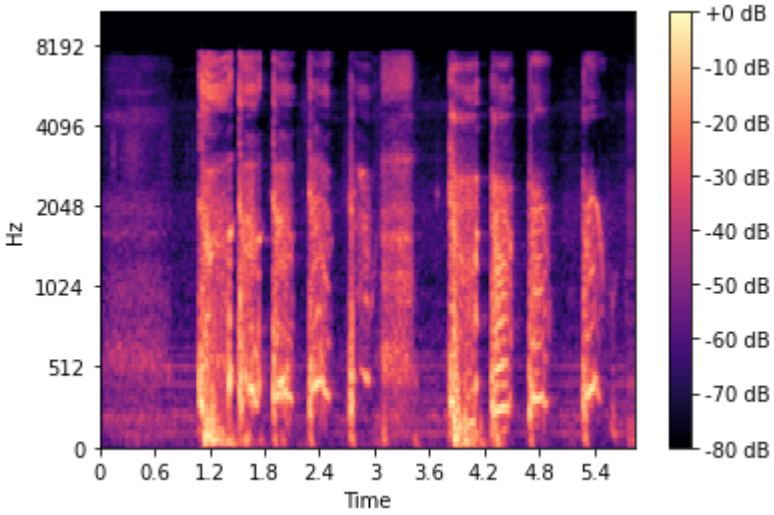

```
n_mels = 128
mel = librosa.filters.mel(sr=sr, n_fft=n_fft, n_mels=n_mels)


plt.figure(figsize=(15, 4));

plt.subplot(1, 3, 1);
```

```
librosa.display.specshow(mel, sr=sr, hop_length=hop_length, x_axis='linear');
plt.ylabel('Mel filter');
plt.colorbar();
plt.title('1. Our filter bank for converting from Hz to mels.');

plt.subplot(1, 3, 2);
mel_10 = librosa.filters.mel(sr=sr, n_fft=n_fft, n_mels=10)
librosa.display.specshow(mel_10, sr=sr, hop_length=hop_length, x_axis='linear');
plt.ylabel('Mel filter');
plt.colorbar();
plt.title('2. Easier to see what is happening with only 10 mels.');

plt.subplot(1, 3, 3);
idxs_to_plot = [0, 9, 49, 99, 127]
for i in idxs_to_plot:
    plt.plot(mel[i]);
plt.legend(labels=[f'{i+1}' for i in idxs_to_plot]);
plt.title('3. Plotting some triangular filters separately.');

plt.tight_layout();
```


```
plt.plot(D[:, 1]);
plt.plot(mel.dot(D[:, 1]));
plt.legend(labels=['Hz', 'mel']);
plt.title('One sampled window for example, before and after converting to mel.');
```




```
S = librosa.feature.melspectrogram(kk, sr=sr, n_fft=n_fft, hop_length=hop_length,
S_DB = librosa.power_to_db(S, ref=np.max)
librosa.display.specshow(S_DB, sr=sr, hop_length=hop_length, x_axis='time', y_axi
plt.colorbar(format='%+2.0f dB');
```




```
# Sanity check that indeed we understood the underlying pipeline
S = librosa.feature.melspectrogram(kk, sr=sr, n_fft=n_fft, hop_length=hop_length,
fft_windows = librosa.stft(kk, n_fft=n_fft, hop_length=hop_length)
magnitude = np.abs(fft_windows)**2
mel = librosa.filters.mel(sr=sr, n_fft=n_fft, n_mels=n_mels)
```

```
assert (mel.dot(magnitude) == S).all()
```