2. Analyzing audio files

Working with audio files (External resource) (1.0 points possible)

the sample rate, plot them, and take a sample of your sound signal to analyze.

To read the data from the audio file and save the data and sample rate as the output variables y and Fs respectively, use the command

```
[y,Fs] = audioread('audiofilename.wav') % (Note the quotations around the file name)

% y is the data from your file
% Fs is the sample rate
% (We will be using .wav files in this problem,
% but this works for any standard audio file type.)
```

The thing about sound signals is that they change in time. When we want to analyze a piece of a sound signal, we typically need to trim the beginning and end, and crop our data to find some representative sample in the middle which we can then analyze (using Fourier analysis methods). (Because the sound signal we are working with here is of one tone, we only need to trim the beginning and end where there is no sound. However, in general, determining the window of what to sample is itself part of the challenge.)

Suppose I want to clip off the first 10% of the sound signal, and then sample 30% of the signal. The way to do this would be the following:

```
L = length(y); %Find the length of the audio signal
y = y(round(L*0.1):round(L*0.4)); %Pick the range from the ten percent to 40 percent range
% Use the function round(), floor(), or ceil() to guarantee that your range is given by integers!
```

In this problem, you will

1. read an audio file,
2. plot the file (using the RUN CODE button)
3. determine a reasonable range to sample from
4. create a new sound signal sampled from the original data, and submit for a grade!

Script

Save Reset MATLAB Documentation (https://www.mathworks.com/help/)

```
%The following reads the data file, and plots the file
[y,Fs] = audioread('1803_musicdata_guitar1.wav');
%sound(y,Fs)  % Command which doesn't work here in edX

%Because the sound signals have 2 channels, we will just analyze the first channel
y = y(:,1);

figure(1)
plot(y)
set(gca,'fontsize',18)

% Find the length of y
L = length(y);
```

```
% Determine starting and ending points of your clipping by looking at plot 1
% Note these must be integers that are not 0!
index1 = 3*10^4; % CHANGE THIS NUMBER
index2 = 12*10^4; % CHANGE THIS NUMBER

newsignal = y(index1:index2);  % Clips sound signal to new signal that starts at index1 and goes to index2
figure(2)
plot(newsignal)
set(gca,'fontsize',18)
```

Run Script

Assessment: All Tests Passed

Submit

- **Are y and Fs correct?**
- **index1 large enough? index 2 small enough?**
- **newsignal correct?**

Output

2. Analyzing audio files

Hide Discussion

Topic: Unit 1: Fourier Series / 2. Analyzing audio files

Add a Post

Show all posts ▼

by recent activity ▼

There are no posts in this topic yet.

✖

Learn About Verified Certificates