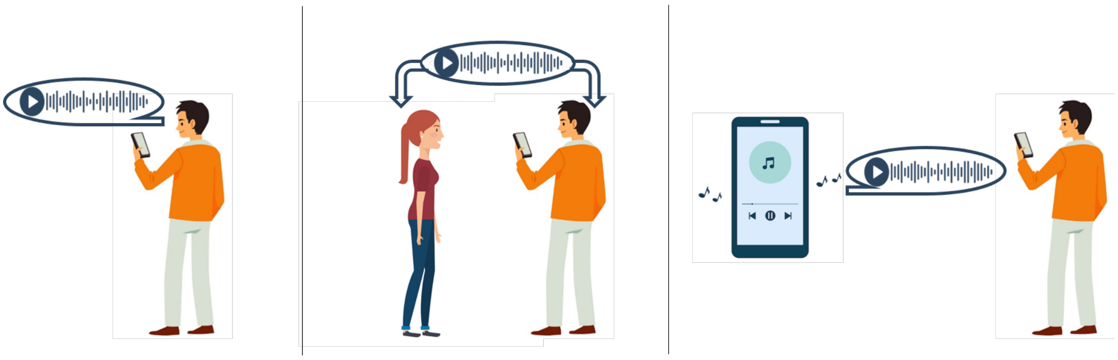
3 Tutorial 3: Time-Frequency Analysis

In this Tutorial, you will use Fourier Transform to analyze an audio data file captured from either one of three different sources using PhyPhox software on your phone (just like what we did last tutorial), as shown in Fig(4).



(a) (b) (c)

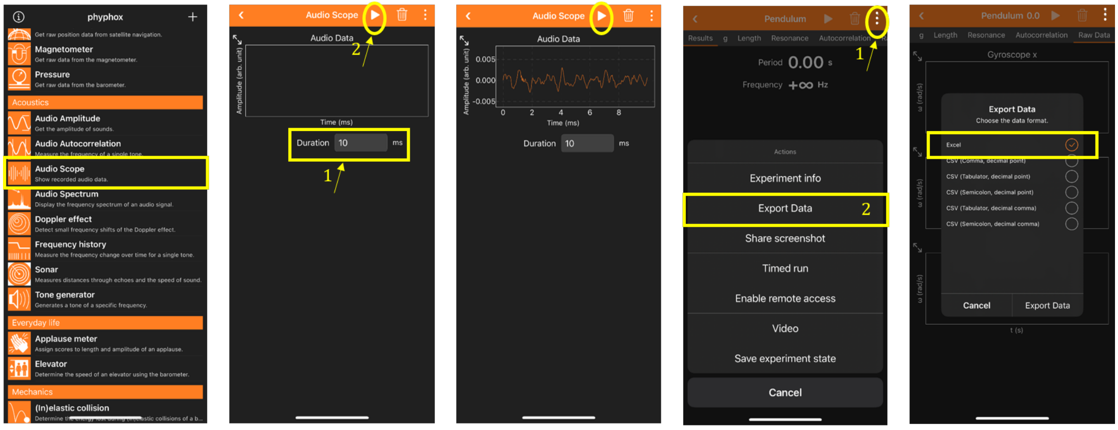
Figure 4: The three options to record an audio data file. (a) Record your speech, (b) record a conversation between you and your friend, or (c) record a tone playing on another device, e.g., your friend's phone, your laptop, etc.

To do this, follow the next steps in order.

1. **Download PhyPhox Software**. Similar to last week's tutorial, there will be no need to have hardware equipment to do the experiment of today's tutorial. PhyPhox will be enough to capture the necessary data. If you missed that last week, Phyphox (`https://phyphox.org/`) is a mobile lab that allows you to use the sensors in your phone to do your experiments. For example, record an audio signal and analyze its frequency and pattern. PhyPhox is available to be downloaded for free on Android and iOS. You can directly install it from Google Play (Android) or the App Store (iOS), and it will only take a couple of minutes. If you haven't done so last week, please download the software for today's experiment.

2. **Experiement Setup**. For this experiment, you need to record an audio signal within a specified time interval using your phone and via PhyPhox software. To do this, you have three options to choose from, as shown in Fig(4). You can either record your own voice speaking (or singing!), as shown in Fig(4, a). Alternatively, you can record a conversation between you and your friend, as shown in Fig(4, b). Or, as shown in Fig(4, c), you can use another device, e.g., your friend's phone or your laptop, to record a playing song. It's totally

up to you whichever option you choose. Once you decide which setup you will be adopting, open PhyPhox on your phone, and choose the "Audio Scope" experiment from the "Acoustics" section, as shown in Fig(5, a). A new screen will be opened showing a blank figure of the signal amplitude as a function of time, as shown in Fig(5, b). Inside this figure, you will observe the audio signal captured from the source. Set the duration to 10 m.sec. Try to minimize the noise as much as possible, but do not worry about this too much, because you will be filtering your signal later on.



(a) (b) (c) (d) (e)

Figure 5: The step to record an audio signal using PhyPhox.

3. **Collect Data**. At this point, you are ready to record your data. When you are ready, press the start button (the right-headed triangle on the top right of the screen), as shown in Fig(5, b). Different patterns will be shown inside the figure that corresponds to the different frequencies your phone receives. Once you decide on a preferred pattern, press stop to end the recording, as shown in Fig(5, c).

4. **Export Data**. To export the data, press the three dots on the top right of your screen and choose the "Export Data" option, as shown in Fig(5, d). Multiple options will appear to you of which format you would like your data to be exported, as shown in Fig(5, e). For simplicity, choose "Excel" as the exported data format. Having said that, Feel free to choose any other format. If you open the Excel file, you should see that your dataset has two columns, the time and the amplitude of the captured audio signal.
   NB: The exported Excel file is in its compatibility mode and you should save it as an "Excel worksheet" to be properly imported to Jupyter Notebook later on.

5. **Analyse Data on Time and Frequency Domain**. As you have the audio data on your laptop by now, you can open a Jupyter Notebook file and start analyzing the signal. First, as usual, you need to import all necessary libraries and then import the audio Excel file. Do all the necessary explanatory data analysis steps to make sure you understand your file. Visualize the signal pattern in both the time and frequency domain.

6. **Filter the Audio Signal using Fourier Transform**.
   - Compute the power spectrum density of the audio signal.
   - Filter out the noise by finding all frequencies that are above a certain threshold and zeroing out all the rest.
   - Visualise the original and the filtered audio signals and their associated power spectrum density.

7. **Create a Spectrogram**. Using the ***original*** audio signal, create a spectrogram using a Gaussian window with different widths: $\sigma = 0.2, 0.5, 1$. What is the difference between the resulting three spectrograms when changing the Gaussian window width? Visualize the three spectrograms along with the original audio signal.

8. **Create a Scaleogram**. Using the ***original*** audio signal, create two scaleograms where each uses one of the continuous wavelets and the other uses one of the discrete wavelets. Visualize both scaleograms along with the original audio signal.

9. **Answer the following questions**:
   - What is the difference between the spectrogram and the scalogram of your audio signal?
   - What do you think would happen if you dampen the audio signal at the end of the recording, i.e., move slowly away from the source? Try this experiment if you still have time in the tutorial and analyze the results.