

# Noise Suppression in Speech Signal Using Butterworth Filter (MATLAB)

Author: Inaam Ahmed

This is a GUI for Filtering. The user has an option to choose either human voice signal for 5 complete seconds or a loaded signal on system. The loaded signal needs to be a .wav format. I have employed Additive White Gaussian Noise (AWGN) and the filtering for the same is provided using a Butterworth. A sample .wav file has been provided to the user.

Following steps are recommended to complete the experiment

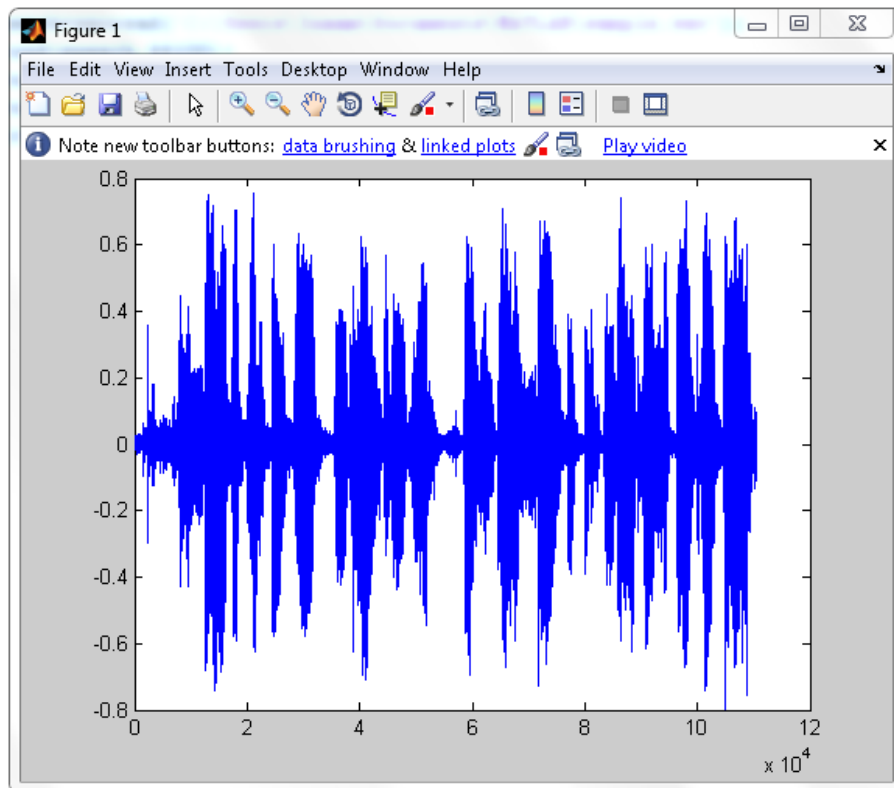
- Click on "load a sound file"(Radio Button) and hit load the sound file(push button).
- Click on "play original" to listen to the loaded file.
- Click on "Add noise" to add a AWGN noise.
- Filter the noisy signal using either Butterworth/Chebychev Filter.
- See the waveform of the signal using the Plot signal.
- Alternately you can follow the same procedure with your voice signal by clicking on "Speak into Mike"

Load the wav file using wavread built in method from current matlab directory.

```
>> speech=wavread('C:\Users\Inaam\Documents\MATLAB\sample.wav');
```

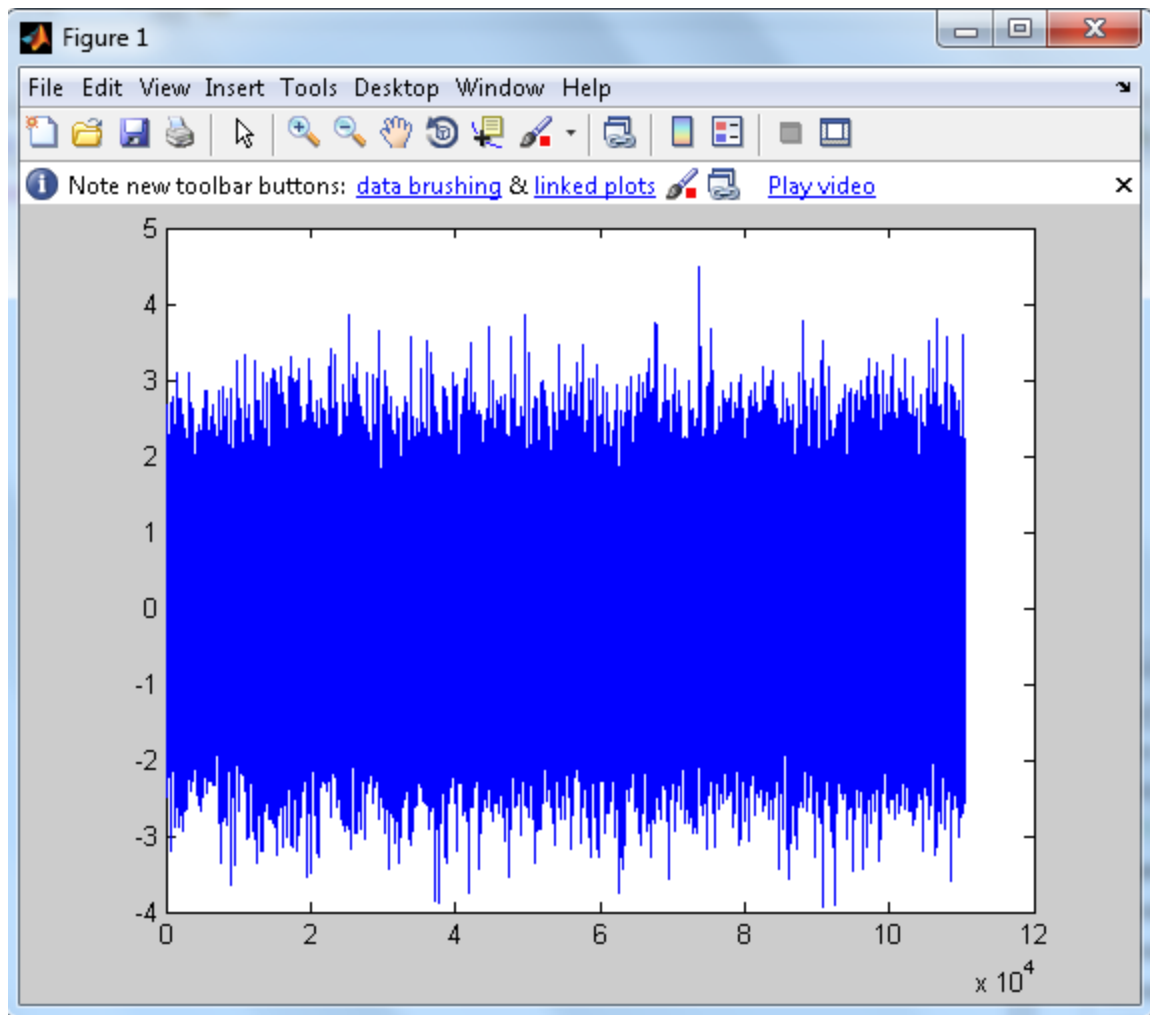
Use sound function to listen wave file using 88100, 44100, 22050 sampling frequencies,

```
>> plot(speech);
```



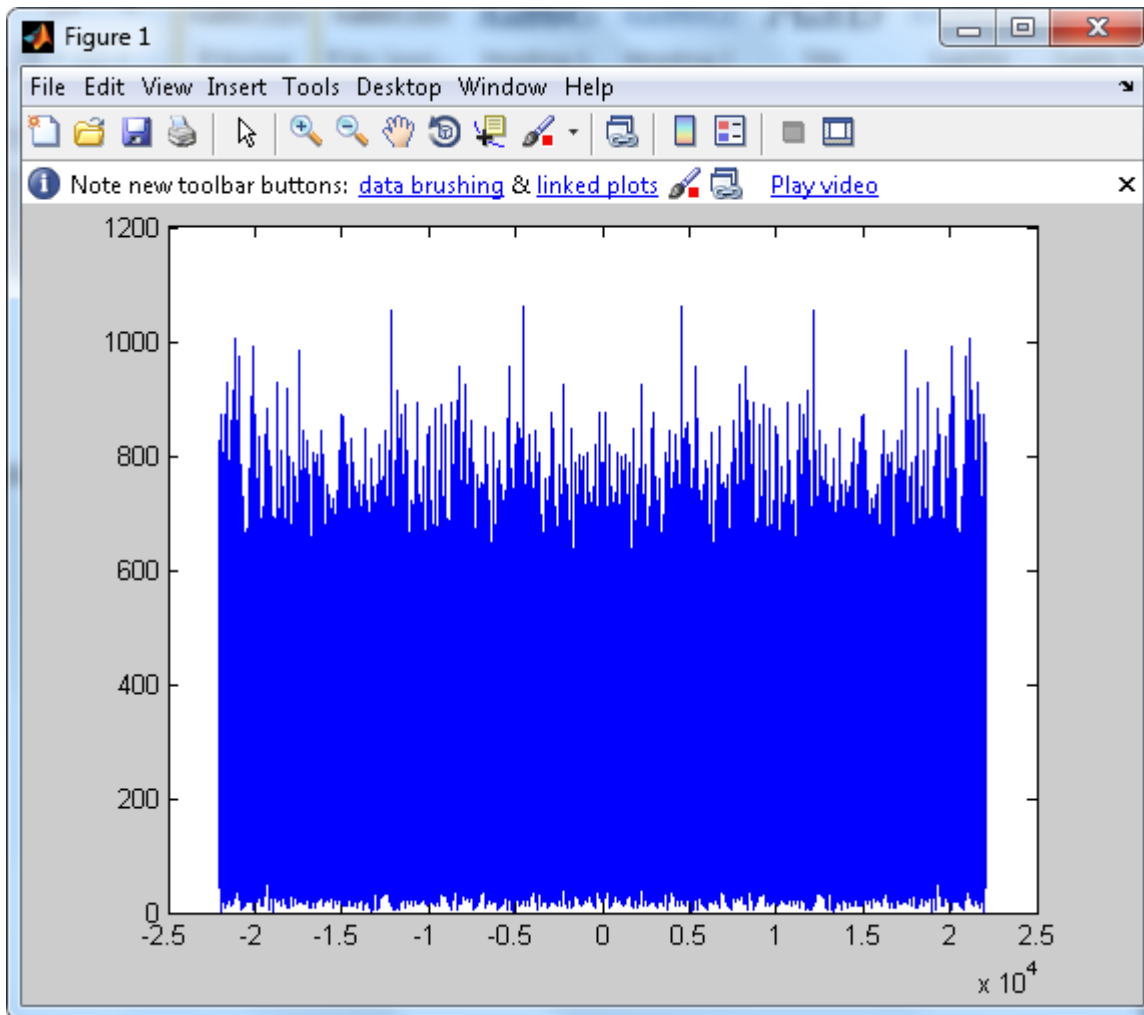
```
>>noise=awgn(speech,0.5);
```

```
>>plot(noise);
```



```
>>fft_of_speech=fft(noise);  
>> shiftedfft=fftshift(fft_of_speech);  
>> frequency=(44100/2)*linspace(-1,1,length(shiftedfft));  
>> plot(frequency,abs(fft_of_speech));
```

Required FFT of speech signal now we can use any filter as filter allows some frequencies and stop some frequencies.



Butterworth Filtering Function:

```
function varargout = proj_filtering(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @proj_filtering_OpeningFcn, ...
                  'gui_OutputFcn',  @proj_filtering_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

end
```

```
function proj_filtering_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;
guidata(hObject, handles);
end
```

```
function varargout = proj_filtering_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;
set(handles.speak_into_mike, 'Value',0);
set(handles.load_sound_file, 'Value',0);
end
```

```
function speak_into_mike_Callback(hObject, eventdata, handles)
set(handles.speak_into_mike, 'Value',1)
set(handles.load_sound_file, 'Value',0)
set(handles.speak, 'Visible', 'on');
set(handles.sound_file, 'Visible', 'off');
end
```

```
function load_sound_file_Callback(hObject, eventdata, handles)
set(handles.load_sound_file, 'Value',1)
set(handles.speak_into_mike, 'Value',0)
set(handles.sound_file, 'Visible', 'on');
set(handles.speak, 'Visible', 'off');
end
```

```
function speak_Callback(hObject, eventdata, handles)
r = audiorecorder(22050, 16, 1);
disp('speak for 5 seconds');
handles.r=r;
recordblocking(r,5);
myspeech=getaudiodata(r, 'double');
freq=22100;
wavwrite(double(myspeech),freq, 'myvoice.wav');
original_sound=wavread('myvoice.wav');
handles.original_sound=original_sound;
guidata(hObject,handles);
end
```

```
function sound_file_Callback(hObject, eventdata, handles)
[filename,pathname]=uigetfile('*.wav', 'Select an image File');
[input_song,freq]=wavread(fullfile(pathname,filename));
```

```
wavwrite(double(input_song),freq,'song.wav');
original_sound=wavread('song.wav');
handles.original_sound=original_sound;
guidata(hObject,handles);
end
```

```
function play_original_Callback(hObject, eventdata, handles)
freq=22100;
original_sound=handles.original_sound;

    sound(original_sound,freq)
handles.freq=freq;
guidata(hObject,handles);
end
```

```
function add_noise_Callback(hObject, eventdata, handles)
original_sound=handles.original_sound;
adding_noise=awgn(original_sound,10,'measured');
freq=22100;
sound(adding_noise,freq);
    wavwrite(double(adding_noise),freq,'noisy.wav');
    noisy_signal=wavread('noisy.wav');
    handles.noisy_signal=noisy_signal;
    guidata(hObject,handles);
end
```

```
% --- Executes on button press in butterworth.
function butterworth_Callback(hObject, eventdata, handles)
freq=22100;
freq=handles.freq;
noisy_signal=handles.noisy_signal;
[b,a] = butter(5,0.9, 'low');
    butterworth_filtered_signal= filtfilt(b, a, noisy_signal);
    handles.butterworth_filtered_signal=butterworth_filtered_signal;
    sound(butterworth_filtered_signal,freq);
    guidata(hObject,handles);
end
```

```
function Chebychev_Callback(hObject, eventdata, handles)
freq=22100;
freq=handles.freq;
noisy_signal=handles.noisy_signal;
[b,a] = cheby2(5,20,0.9, 'low');
    chebchev_filtered_signal = filtfilt(b, a, noisy_signal);
    sound(chebchev_filtered_signal,freq);
```

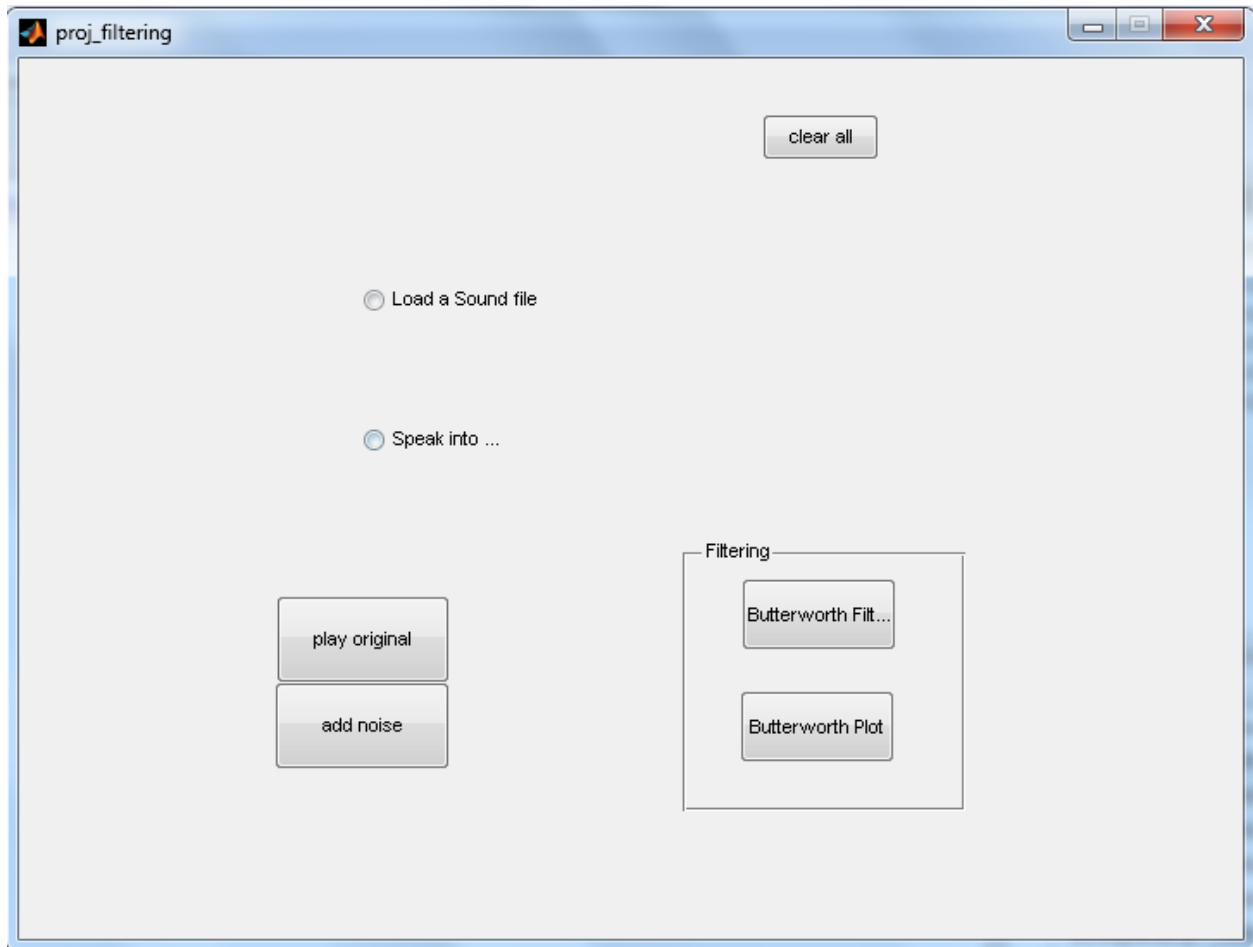
```
handles.chebchev_filtered_signal=chebchev_filtered_signal;
guidata(hObject,handles);
end
```

```
function clear_Callback(hObject, eventdata, handles)
clear all;
clc;
end
```

```
function chebychev_plot_Callback(hObject, eventdata, handles)
chebchev_filtered_signal=handles.chebchev_filtered_signal;
original_sound=handles.original_sound;
figure(1)
subplot(211),plot(original_sound),title('Original Sound Plot');
subplot(212),plot(chebchev_filtered_signal),title('Chebychev Filtered Signal');
end
```

```
function butterworth_plot_Callback(hObject, eventdata, handles)
butterworth_filtered_signal=handles.butterworth_filtered_signal;
original_sound=handles.original_sound;
figure(2)
subplot(211),plot(original_sound),title('Original Sound Plot');
subplot(212),plot(butterworth_filtered_signal),title('Butterworth Filtered Signal');
end
```

```
>> proj_filtering
//shows the GUI Form below,
```



Click on load sound file radio button.

Load the sound file

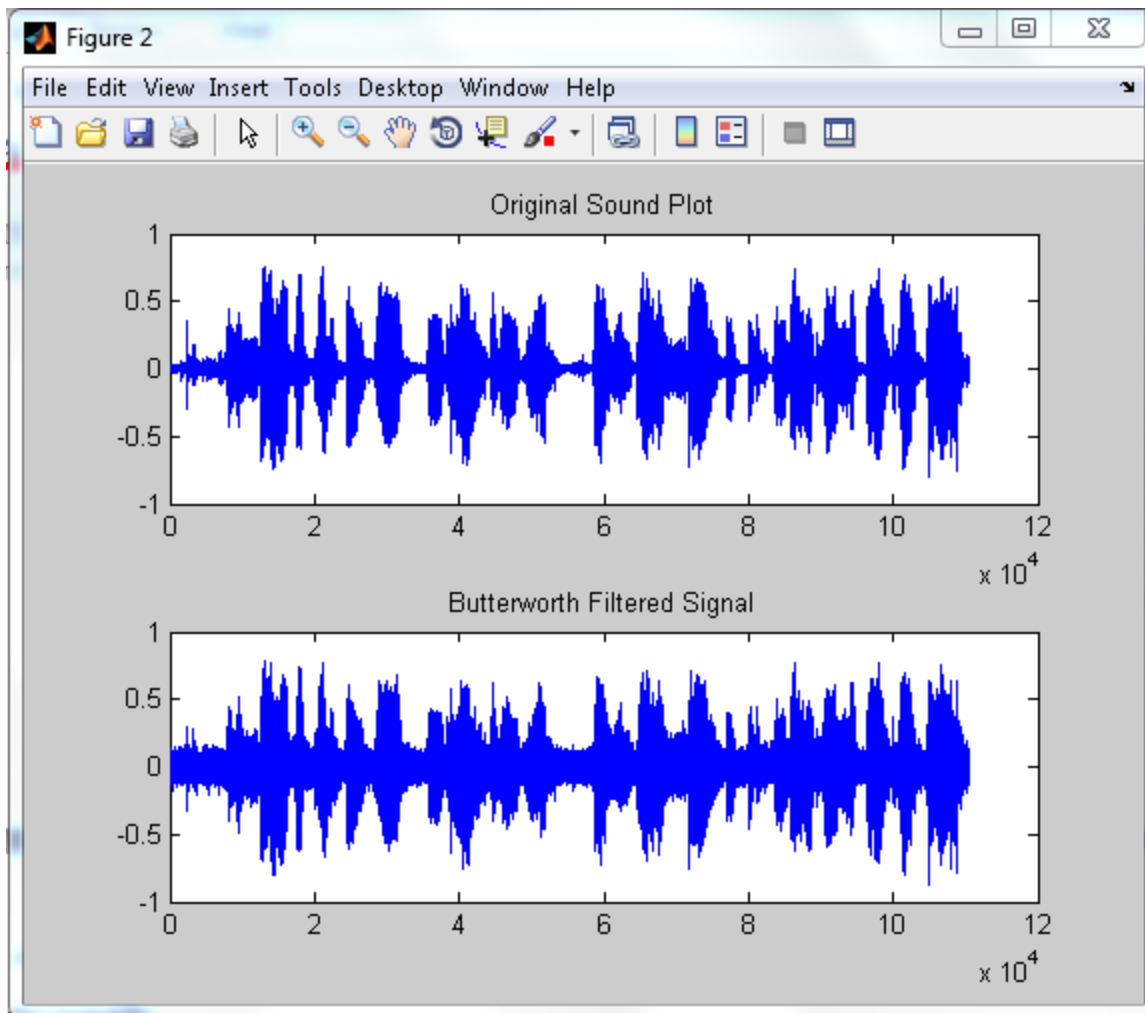
Play original file

Now play noisy version

Click on Butterworth Filtering button that sound will redunct the noise to some extent

Click on Butterworth Plot





Although Butterworth removes the noise to some extent but noise is still remaining due to frequency samples same at both noise and speech ordinates.