

DC Offset

The measured EEG signal on the headset is converted from an unsigned 14 or 16-bit ADC output to a floating point value which is stored by EmotivPRO. To allow measurement of negative values the (floating) DC level of the signal occurs at approximately 4200 uV. Negative voltages are transmitted less than the average level, and positive voltages are transmitted as greater than the average.

In order to remove the DC offset, especially before performing any kind of analysis such as Fast Fourier Transform (FFT) it is necessary to apply some kind of DC offset removal. The simplest method is to subtract the average from the entire data channel, although this is the least accurate. Ideally you should apply a high-pass filter which matches the characteristics of the electronics - that is, you should use a 0.16Hz first order high-pass filter to remove the background signal (this also removes any longer term drift, which is not achieved by the average subtraction method). Another method is to use an IIR filter to track the background level and subtract it an example is shown below in Matlab pseudocode, assuming the first row has been removed from the array `input_data()`:

```
IIR_TC = 256;
```

2 second time constant- adjust as required

```
EEG_data = input_data( : ,3:16 );
```

select out only the EEG data

```
[rows columns]= size(EEG_data);
```

rows= number of data samples, columns= 14

```

AC_EEG_data = zeros(rows, columns);

reserve space for the output data file

back= EEG_data( 1, : );

copy the first row of data into the background file

for r = 2 : rows

back= (back* ( IIR_TC- 1 ) + EEG_data( r,:)) / IIR_TC;

AC_EEG_data = EEG_data( r,:)- back;

end

```

This demonstration code is not efficient in memory and assumes the entire file is available. It is quite straightforward to modify the code to replace the data in the source array rather than making a separate AC-coupled array, and also to run the IIR filter in open-ended form for processing in real time.

Note the vectorised form of the background recalculation at each iteration - each individual channel background average is retained in the relevant column of "back". At each step the running average is re-estimated using the new input value. Note also that the first IIR_TC samples are biased towards the initial value but this settles down after about $2 * \text{IIR_TC}$ samples.

It is very important to remove the background signal before performing an FFT - you should also apply a tapered window function such as a HANNING transform before executing the FFT to ensure there are no wrapping artefacts where the FFT treats the data as an infinitely repeating sequence, and any mismatch between the first and last samples appears as a STEP FUNCTION in the analysis, injecting noise across the spectrum.

[Previous](#)
 Converting EDF files to CSV files

Next
Code Examples

Last updated 3 years ago