

**Digital Processing of
Electrical Signals
from
the Brain and the Heart**

—

Project proposed (in preparation)

Pablo Laguna

Leif Sörnmo

**Lund University, Lund, Sweden,
Zaragoza University, Zaragoza, Spain**

April 24, 2018

All rights reserved. No part of this book may be reproduced, in any form or by any means, without permission in writing from the authors.

Chapter 3

Project 3.1 (Estimate EEG parameters)

The objective of this project is to get trends of the EEG indexes on a EEG recording.

The recording is in the file *slp.mat* ready to be load with MATLAB. In that file there are an actual EEG signal sampled at 250 Hz (from polisomnographic database). When loading the file there appear two variables: *slp01* and *st*.

- The variable *slp01* contains 5 minutes of EEG signal in a matrix format (60000,2) where the first column are the sample index and in the second the EEG value. The 5 minutes do not belong to the (0,5) minutes interval of the original recording at the database, and because of that the sample index do not start by zero.
- The variable *st* contains the time instants at which several sleep patrons appear (according to the annotations at the original database??). This variable has dimension (8,2) where in the first column appears the sample index associated to a 8 different sleep stages beginning at that instant, and at the second column a code denoting the sleep stage according to the following notation.

W: subject awake; **1:** sleep stage 1; **2:** sleep stage 2; **3:** sleep stage 3; **4:** sleep stage 4; **R:** REM stage; and **M:** subject moving (uncertain sleep stage)

It is proposed to estimate on overlapped moving windows the three Hjorth Parameters $\mathcal{H}_0(n)$, $\mathcal{H}_1(n)$, $\mathcal{H}_2(n)$ so a Hjorth trend can be obtained for the recording, and the spectral purity index $\Gamma_{\text{SPI}}(n)$. Analyse how this parameter change with the different sleep stages as denoted in the variable *st*. The overlapping windows (L) can be taken as 4, 8, 40 seconds, (1000, 2000, and 10000 samples) with 0.5 second displacement.

These Hjorth moments are defined assuming Gaussianity in the data. If we want to explore until what extend this can be true it can be estimated univariate high-order moments like and skewness (S_{mc}) denoting asymmetry in the PDF and kurtosis (K_{mc}) denoting the peakedness in the same time windows as before. Those parameters can be estimated [?] as

$$S_{mc}(n) = \frac{\sum_{k=n-L+1}^n \frac{(x(k) - \bar{x}(n))^3}{L}}{\left[\sum_{k=n-L+1}^n \frac{(x(k) - \bar{x}(n))^2}{L} \right]^{3/2}} \quad (3.1)$$

and

$$K_{mc}(n) = \frac{\sum_{k=n-L+1}^n \frac{(x(k) - \bar{x}(n))^4}{L}}{\left[\sum_{k=n-L+1}^n \frac{(x(k) - \bar{x}(n))^2}{L} \right]^2} - 3 \quad (3.2)$$

where

$$\bar{x}(n) = \frac{1}{L} \sum_{k=n-L+1}^n x(k) \quad (3.3)$$

Project 3.2 (Spectral analysis of EEG during photo stimulation)

Background: Patients with suspected epilepsy usually undergo an EEG investigation which involves photo stimulation. This technique is used to trigger epileptic potentials by repeated light flashes (possibly resulting in a repetitive spike-wave pattern which occur about 3/second if the patient suffers from epilepsy). The lamp, which is positioned approximately 15 centimeters from the patient's face, flashes at a rate that varies from 10 to 30 Hz.

The present project: In the present case, a one-channel EEG is first recorded without photo stimulation, and then followed by recordings obtained at stimulation frequencies of 10, 20 and 30 Hz; thus, four recordings were made in total. All recordings have a length of four seconds and were sampled at a rate of 256 Hz with an amplitude resolution of 1 microVolt. The electrodes were positioned such that cortical center which processing visual information is beneath (posterior part of the parietal lobe). The aim of the project is to analyze the EEG signals spectrally such that the effect of photo stimulation can be adequately assessed. Spectral analysis should be done by using both non-parametric and parametric techniques.

The EEG signal is contain in the file *eegfoto.mat*, and when load form MATLAB it appear a variable *eeg* dimensioned 1024x4, where in the first column in the control EEG without stimulation and the second, third and fourth columns the EEG from 10, 20 and 30 Hz stimulation rate respectively (is this correct Leif?)

The following tasks should be addressed:

- 1: Plot all four signals and comment on their characteristics.
- 2: Analyze each of the signals by using a non-parametric spectral technique (i.e., based on the FFT) by using some different choice of windows and interval segmentation.
- 3: Analyze each of the signals by using some parametric spectral technique which assumes AR modeling for some different model orders.
- 4: The choice of sampling rate is important in 3 and does not have to be the original 256 Hz. Discuss which sampling rates that would be appropriate, and then analyze the signals spectrally using such a sampling rate. The Matlab function `decimate` can be used.
- 5: Trend removal (that is, removal of the very slow variations in the signal) is an important processing step in parametric spectral analysis in order to avoid that the DC component obscures higher frequencies. Use the Matlab functions `butter` together with `filtfilt` to accomplish appropriate trend removal.

You are free to use any of the functions in Matlab which you consider as suitable for the tasks above.

Project 4.1 (Noise reduction of somatosensory evoked potentials)

This project studies the value of different methods for ensemble-based noise reduction of evoked potentials. Since the signal-to-noise ratio is very poor for the ensemble of interest, it is necessary to make use of signal processing techniques. Classical ensemble averaging assumes that the signal waveform is constant from response to response; this assumption must, however, be modified here since there are strong reasons to believe that the signal morphology is changing due to a drug being injected at a certain time instant.

The acquired signal is sampled at a rate of 3200 Hz and contains 2840 stimuli. Each evoked (somatosensory) response has a duration of 80 milliseconds (256 samples), from which 40 milliseconds are before the stimuli and the other 40 milliseconds are after stimulus. The responses has been concatenated with the other response to produce a single signal containing 727040 samples. The data is stored as a Matlab mat-file call *ev.mat*. When loaded from MATLAB it appear the variable *ev* dimensioned (727040,1) containing the data.

The drug is injected after stimulus 1420, and, consequently, the signal morphology is likely to change after that particular stimulus.

1. Determine the ensemble average for successive subgroups of the data. Try different size of the subgroup, like 50, 100, 200 and comment on the results.
2. Determine the weighted average for successive subgroups of the data by estimating the noise variance from the interval prior to the stimulus.
3. Determine the exponential average for a number of different weights $\alpha = 0.2, 0.05, 0.01, 0.005$.
4. Determine the estimation by truncated Fourier series taking the first 12 harmonics.
5. Determine the estimation by truncated Fourier series taking the first 12 harmonics, but now making a adaptive estimation by the block LMS algorithm and using same μ values as α were use in the exponential average.
6. Compare, at suitable time instants (like after 100, 1420, 1425, 1700 and 2840 stimuli), the morphology of the estimated signals from items 1, 2 and 3 above, and describe any differences (or similarities) that you may observe.

Project 4.2 (Confidence interval in somatosensory evoked potentials)

This project studies the confidence in the evoked potential after estimated by ensemble averaging. To study this we can plot the superimposed waveforms for a number N of observation to average. Then we can plot, on other graph, the estimated mean together with the mean plus the standard deviation and the mean minus the standard deviation. Still other more meaningful graph is to plot the confidence limits together with the mean value. The confidence limits can be estimated as the threshold value that when crossed by the estimated mean, the probability that the evoked response still is not present is lower than 5%. this graph can be estimated by making used of the two-tailed t student test. It is suggest to evaluate the required graphs for a number of averaged observations, M , ranging from $M=10, 20, 50$ and 100 . The two-tailed testing t student distributions for the respective degrees of freedom, $d_f=(M-1)$, and a confidence interval lower than 5%, $t_{0.05}(d_f)$, are: $t_{0.05}(d_f)=2.262, 2.093, 2.009$, and 1.984 , respectively.

Note that, for each sample n , the threshold value at which, when the EP estimate $\hat{s}_a(n)$ is higher than the threshold, the probability of EP absence (mean equal to zero) is lower than 5% can be estimated with the student t distribution,

$$t(n) = \frac{\hat{s}_a(n) - 0}{\hat{\sigma}_v/\sqrt{M}} \quad (4.4)$$

as

$$|\hat{s}_a(n)| > t_{0.05}(d_f)\hat{\sigma}_v(n)/\sqrt{M}. \quad (4.5)$$

The data to be used is the same as in previous project.

Project 4.3 (Alignment in somatosensory evoked potentials)

This project has as objective to evaluate the alignment of the evoked responses prior to averaging. For that it is used the same signal as in previous project.

1. Estimate the new alignment instant for each response using the Woody method.
2. Compare the ensemble average with this realignment and without realignment at the same time instant than in previous project. Make emphasis in the peak values of responses that are used for clinical diagnosis.
3. Make same comparison with the weighted average estimation.
4. To have an amplified idea of the effects of misalignment it can be re-estimated the averages rather than with a realign synchronization with an artificially introduced misalignment δ which takes values according to a random variable Gaussian distributed with variance 1 and 5 ms. Compare the result with previous ones.
5. Woody alignment improves the synchronization instant estimation up to certain SNR. This phenomena can be shown here by adding noise to the original recordings and reestimating the averages. Find the noise level when the Woody alignment rather than improving the average deteriorates it.

Project 4.4 (Wavelet denoising in somatosensory evoked potentials)

This project should illustrate the wavelet denoising applied to EP

Project 5.1 (EMG Amplitude estimate)

In polisomnographics studies it is useful to monitor the EMG of some mussel to certify if the subject is with some muscular activity related to awake ness status.

In the companying file (slp48) it is presented the EMG signal and it is required to estimate the amplitude by a running window RMS value.

Obtain the amplitude estimate for several values of the window length and comment about the trade-off between the variance of the estimate and the robustness of the activity episode detection.

Do that by the Laplacian estimate and the Gaussian. Comment about the suitability of each one for this case.

obtain a running trend of the mean and median frequency and see also How the variance of each one varied across the recording. Comment about the expectation you have for mean frequency in a polisomnographic study.

Project 6.1 (QRS detector)

The objective of this project is to develop a QRS detector, that constitutes the basis of most (if not all) ECG applications. It is required to implement the three typical stages of the QRS detectors, linear filtering, Nonlinear filtering and decision rule.

For this purpose it is supplied a signal coming from a three lead (X, Y, Z) recording, sampled at 250 Hz, and contained 8.8 minutes. The signal is contained in a file called *wallen1.mat*. When loading this file from MATLAB it appears a variable call *wallen1* dimensioned 133088x4. First column denotes the sample index, second the ECG value for lead X, third ECG value for lead Y, and fourth ECG value for lead Z. The signal amplitude is given with a gain of 200 *adu/mV*.

1. Develop a single lead QRS detector, with the proper filters, squaring as non-linear transformation and convolution with squared impulse response plus proper decision rule.
2. Evaluate the results in terms of false positives (FP), and false negatives (FN) at each lead. The reference for the evaluation can be taken by visual inspection on the raw data.
3. develop now a multi-lead QRS detector simply by generalising the non-linear transform so the output is the sum of the squaring of the three lead filtered signals. Compare the results with the single lead results and comment on it.
4. Finally it can be redesign the QRS detector so the first linear filtering stage has a matched filter to the QRS shape estimated as average of the 10 first QRS estimated by previous detectors. Compare the results, both in terms of FP and FN and in precision at the fiducial point estimation evaluated by overplotting the raw signal with the QRS marks.
5. One further improvement can be thought to be using, after the non-linear transformation, a new matched filter matched to the typical QRS non-linear transformed signal rather than the squared impulse response. Implement this stage and comment on its real improvement. shape

Project 6.2 (50 Hz ECG filtering)

The objective of this project is to implement and compare different strategies for 50/60 Hz filtering in the ECG. For that is presented in the file *ecg50.mat* an one minute ECG recording sampled at 1000 Hz. When load the file from MATLAB it appears the variable *ecg50* dimensioned (60000x2). In the first column there is the original 50 Hz free ECG data, and in the second it is a 50 Hz contaminated ECG by adding to the raw data a sinusoid of 40 μV amplitude. This is a typical recording from some surgery unit with the advantage of having the "clean" signal so to evaluated the different filtering techniques.

1. Implement and filter the signal with a FIR with two complex-conjugate zeros.
2. Repit the filtering with a IIR filter that includes a pair of complex-conjugate poles at the same angle as the zeros and close to the unit circle.
3. Filter the signal by the non-linear strategy proposed in the text for this application.
4. Implement the estimation by subtraction method for 50/60 Hz removal
5. Compare the results in terms of total RMS and in term of local RMS around the final part of the QRS where the ringing of the filter makes bigger impact. comment on the results.

Project 6.3 (QRS complex representation using the KLT expansion)

In this project, the usefulness of the Karhunen-Love (KL) expansion is studied for representing different QRS complex morphologies. The material consists of heart beats selected both from healthy individuals and from those with different pathologies; in total 50, three lead, recordings resulting in 150 beats. Note that lead-dependent KL representation is not of interest here. Each beat contains 400 samples and is obtained at a sampling rate of 1000 Hz. The QRS are align with a cross-correlation prior to segment it.

The data is in the file *ekg50pt.mat*. When load with MATLAB it appear a variable call *tbeat* dimensioned 150x400 where each row is a QRS complex. Every consecutive 3-beat set belong to one complex (ordered as lead X, Y, Z, in scending order).

1. The correlation matrix for the ensemble of signals is estimated (note that this matrix should not be estimated as if the data is assumed to be a stationary process **What does it means Leif**).
2. Determine the eigenvalues and eigenvectors and present these quantities.
3. Present statistics which characterize the error when using the KL representation. Determine the mean square error when using different numbers of basis functions. In addition, the error when representing individual waveforms should be studied such that the number of basis functions is determined required for obtaining a certain predetermined maximum error between the original and reconstructed signal.
4. Determine the effect of correcting for the DC level before KL representation.
5. Make some basic considerations on how suitable the KL expansion is for data compression (see the attached article "KLT-Based Quality Controlled Compression of Single-Lead ECG" by T. Blanchett, G. C. Kember, and G. A. Fenton).
6. repit the analysis making the KL representation particularized for each lead, mean one covariance matrix for each lead set. Does the result improve in terms of representation capability? why?

Project 6.4 (Baseline wander attenuation by linear filtering)

The aim of the project is to study the performance of linear filtering for removal of baseline variations and to what degree it will distort the ST segment of the cardiac cycle.

The project makes use of an ECG signal, originally without any baseline variations, to which rather drastic variations have been artificially added. Thus, the signal resulting from baseline removal can be compared to the original signal and the performance easily assessed. The other ECG signal also contains baseline variations but included in a fully natural way! The signal can be reached by loading the file *ecg_bw.mat*. It appears two variables: *ecg* dimensioned 60000x2 (first column original ECG sampled at 1000 Hz with no baseline variation, and second column same ECG with artificially introduced baseline variation) and *ecg_lf* dimensioned 290882x1 where it appear an ECG with baseline wander from natural origin when recording.

1. Decide which filter structure to be implemented, i.e., FIR or IIR structure, if decimation and interpolation should be used of the original sampling rate, and if forward or forward/backward filtering should be used.
2. The performance of the filter is presented as an error (e.g., RMS) between the original and the corrected signal. It is also of interest to compare the ST levels at a certain time instant of these two signals. The results should be presented for various parameter settings, e.g., different cut-off frequencies of the high pass filter. Hint: Matlab offers excellent functions for decimation (DECIMATE) and interpolation (INTERP) and these should, of course, be used.

Project 6.5 (Baseline wander attenuation by linear time-varying filtering)

The objective of this project is to implement and evaluate time-varying method for baseline attenuation of the ECG.

For that it is provided in the file *lb117.mat* two minutes of an ECG (sample 250 Hz) with natural baseline variation together with their corresponding QRS position marks. When loading the file it appear two variables: *lb117* dimensioned 30000x3 where in the first column are the sample index, in the second the ECG belonging to lead II and in the third the ECG belonging to lead V2, the second variable is call *lb117_qrs* is dimensioned 107x1 and contains the samples at which each QRS occurs as given by a QRS detector.

1. Implement a time-varying filter with cut-off frequency function of the heart rate.
2. implement a time-varying filter using the cubic-spline technique. Use as knot and average of the signal in the interval (-80, -60) ms prior to each QRS mark.
3. Evaluate these two method adding different sinusoids at areas with no evident baseline wander and then proceeding as in previous project, also comparing with a linear time-invariant filter.
4. Make trend of the ST for every segment, measuring at a fix distance from the QRS complex.

Project 6.6 (DC in ECG)

The AZTEC or SAPA compressors had the property of keeping constant the reconstruction error ε across the ECG signal $x(n)$. However, this property does not imply that the clinical information is retained to the same extent at different areas of the cardiac cycle. For instance at large ε the fast-changing large-amplitude areas as the QRS can still have a diagnostically good performance but the low-amplitude areas as the P wave can already have disappeared and then clinically unacceptable. One way to deal with this limitation is to define a tolerance error ε function inversely proportional to the time distant between the selected retained sample points. This was addressed in the TRIM compressor ?? by first applying a turning point strategy to select samples to be retained and after applying a recursive partitioning compressor that basically breaks each interval at the point where there is more distance from the signal to the interpolated line. This is applied recursively until the maximum error lies below a threshold.

In this project, the ECG is first studied with SAPA-2 algorithm with $\varepsilon=0.1$ mV, to obtain the J time intervals first partitioned, and after apply a recursive partitioning compressor with the following varying threshold [?].

$$\delta\varepsilon(j) = \begin{cases} -kT_j + \varepsilon & T_j < \frac{\varepsilon-0.01}{k} \\ 0.01, & T_j > \frac{\varepsilon-0.01}{k} \end{cases}$$

where T_j ($j = 1, \dots, J$) is the length of the segment to be partitioned, and k is a constant with value $k = 0.3$ mV/s. Play with different k and ε and see what is the effect on the reconstructed signal, and in the compression ratio performance expressed as beats/channel/second.

For that it is provided in the file *ECGDC.mat* two minutes of an ECG (sample 250 Hz) from record *e0106* of the European ST-T database. When loading the file it appear two variables: *sig* dimensioned 30001x2 where in each column is one lead (MLIII and V3), and *heasig* that contains the information provided in the header at the original database.

Project 7.1 (Generating heart rate series from the IPFM)

The objective of this project is to generate heart rate series that follow the IPFM model under some particular input $m(t)$. For that it is required:

1. generate a IPFM models implementation
2. generate the heart rate when the input to the IPFM model $m(t)$ is a summation of three sinusoids at frequencies 0.1, 0.2, and 0.35 Hz. The mean heart rate should be 70 bpm. The maximum amplitude value of $m(t)$ should be 0.2 and the three sinusoids should have the same relative contribution in $m(t)$.
3. Once the heart rate series is obtained generate the heart rhythm representation $d_{IT}(k)$, $d_{IIT}(k)$, $d_{IF}(t)$, $d_{IIF}(t)$, $d_E(t)$, $d_{LE}(t)$, and $d_{HT}(t)$ and plot them.
4. Evaluate for each previous heart rhythm representation the power spectrum by a non-parametric method. There where the signal is not evenly sampled use linear interpolation to generate a evenly sampled signal and then use the PSD estimation. Comment and compare the results in view if the original PSD of $m(t)$.
5. Repeat the same procedure than before by changing the mean heart rate to 40 bpm. What has change and what explains this changes?

Project 7.2 (Evaluating heart rate variability estimators from AR modeling)

The objective of this project is to evaluate, according to clinical indexes, the different heart rhythm representations.

1. Generate sequences of the modulating signal $m(t)$ by an AR model that emulated a heart rhythm at tilted position

$$m(n) = \sum_{k=1}^p -a_k m(n-k) + v(n) \quad (7.6)$$

where

a_1	a_2	a_3	a_4	a_5	a_6	a_7
-1.8149	2.1365	-2.1703	1.7194	-0.9221	0.5311	-0.3262

and noise standard deviation so the standard deviation of $m(n)$ is 0.2.

These coefficients have been estimated when the sampling frequency for $m(n)$ is 1 Hz. However these sampling frequency is very low to input the IPFM model and generate the heart rate spikes. It is necessary to increase at least up to 16 Hz. This can be done by obtaining the $m(n)$ signal sampled at 1 Hz and after up sampling it up to 16 Hz. This could also be obtained by interpolating the a_k coefficient series and then using a 9x16 model order but this will be less efficient from computational point of view.

2. With this $m(n)$ sequences following and AR model, introduce it in the IPFM model with a mean heart rate of 60 bpm, and generate the heart rhythm representation $d_{IT}(k)$, $d_{IIT}(k)$, $d_{IF}(t)$, $d_{IIF}(t)$, $d_E(t)$, $d_{LE}(t)$, and $d_{HT}(t)$.
3. Compute the PSD as in previous project for the different representations and evaluate the result by computing the mean and variance of the PSD error at the VLF, VF and HF bands between the estimated PSD and the original PSD of the $m(n)$ signal that is used to generate the heart rate. To get these mean and variance it is recommended to run the simulation 50 times with different noise seed.

Project 7.3 (HRV in a actual heart rate series)

The objective of this project is to compute different clinical HRV indexes from actual signals. In the file *qrs.mat* it is stored a variable *qrs* dimensioned 502x1 where there are stored the QRS occurrence marks, expressed in sample units, as given by a QRS detector from a healthy subject ECG recording lasting a bit over 8 minutes. The ECG was sampled at 128 Hz.

1. Obtain the different heart rhythm representation for this heart rate series and obtain the PSD
2. Implement the Lomb PSD estimation method for those representation that are unevenly sampled (except for the d_{HT} signal) and compute that PSD.
3. compute the clinical indexes as given by the power at the LF and HF bands, and comment on the differences from the different methods.
4. Compute also the time domain indexes SDNN, SDANN, rMSSD, pNN50 and the triangular index.

Project 7.4 (Late potentials analysis)

The objective of this project is to develop a system to quantify late potentials from a high resolution ECG. It is propose to implement an averager for QRS complex, then a high-pass filtering of the averaged QRS to estimate the vector magnitude VM and from then estimate the width of the QRS complex.

The ECG signal is contained in the file *n_estep.mat* which when load gives two variables: *n_estep2* dimensioned 180000x3 where each column has 3 minutes of signal sampled at 1000 Hz, column 1, 2, 3 leads X, Y and Z, respectively. The second variable *qrs*, dimensioned 211x1, contains the QRS positions as given by a QRS detector.

1. Make and average of the QRS complex at each lead by aligning with the QRS detector mark, and by applying the Woody alignment method. compare the results.
2. High-pass filter the average leads with a Butterworth bidirectional filter (first half of the QRS forward filtering and last half backwards filtering).
3. Estimate the Vector Magnitude VM
4. From the VM estimate the end of the QRS and the beginning applying the following rule: Estimate the VM variance 170 ms after the QRS ends. Calculaet this variance backwards in 10 ms windows and when it reaches 3 times the one estimated at the end of the QRS establish the QRS end. Similar procedure for the QRS onset. Then give the QRS duration as the distance from the onset to the end.
5. Repeat the procedure by averaging 50, 100 and 200 beats. compare the results.
6. Add noise to the signal before averaging in SNR= 5, 10, and 20 dB, then re-estimate the QRS duration, comment on the results.