Signal Processing 1TE651 2019
– Project Description –
Reconstruction of ECG Signals

Carl Bondesson, Joel Berglund, Martin Hellkvist, Ayça Özçelikkale
Signals and Systems, Uppsala University

Contents

1 Introduction

This project is motivated by the 2010 PhysioNet challenge: "Mind the Gap" [1, 2]. It focuses on reconstructing of ECG signals.

1.1 PhysioNet Challenges

PhysioNet is an online forum which facilitates the exchange of recorded biomedical signals and open-source software for the analysis of these signals. It is a part of a collaboration program which aims to simulate research in the study of cardiovascular and other biomedical signals. The participating institutions of this collaborative program include Beth Israel Deaconess Medical Center/Harvard Medical School, Boston University's Center for Polymer Studies, Division of Health Sciences and Technology of Harvard University-Massachusetts Institute of Technology, and McGill University's Centre for Nonlinear Dynamics in Physiology and Medicine [3].

Every year, PhysioNet announces a different "challenge." Each challange focuses on a different issue that is of interest to biomedical research community. Researchers from different institutions submit their solutions. These solutions are compared according to their reconstruction performance and the winner is announced. The resulting research papers are presented in academic conferences and journals.

1.2 "Mind the Gap"

This course project is based on the 2010 PhysioNet Challenge: "Mind the Gap" [1, 2]. This challenge focuses on reconstruction of missing parts of biomedical signals.

In intensive care units (ICUs), having clear measurements of patient physiological signals, such as ECG or PLETH signals, is of great importance. On the other hand, disturbances can make these physiological signals unreadable or completely lost for some periods of time. This makes the accurate monitoring of the condition of the patients difficult.

To solve this problem, one can attempt to reconstruct[1], i.e. estimate, the lost signal values by utilizing other correlated signals which has reliable data during these periods. This reconstruction can be done using various methods. During PhysioNet 2010 challenge, the most successful methods are found to be either neural network based methods or adaptive filtering methods [2, Fig. 2], [4]. As such, we will investigate usage of adaptive filters and the relevant trade-offs in this project.

2 Problem Set-up

We will consider a scaled down version of the PhysioNet challenge. We will focus on the reconstruction of ECG signals. Here, "reconstruction" of a signal means "estimation" of signal values. In Part A, we will use Recursive Least Squares (RLS). In Part B, you will propose your own solution.

2.1 Background

The data we will use for this project has been provided as Matlab mat files in the course folder under StudentPortal. The data consists of only ECG signals. Data for eight patients is supplied.

[1]Throughout this document, the terms "reconstruction" and "estimation" of a signal is used interchangeably

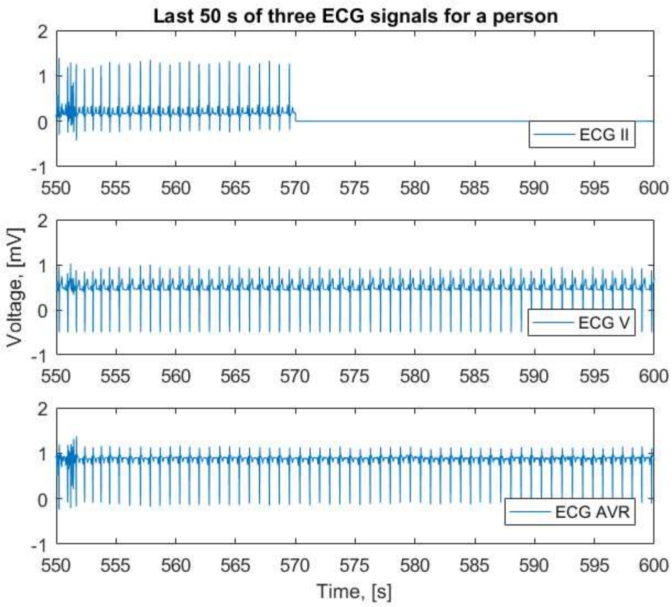


Figure 1: An illustration of the three ECG signals of interest. The zeroed part of the first signal is the part to be reconstructed.

The ECG signal *ECG II* will be estimated using the other ECG signals *ECG V* and *ECG AVR*. The ECG signal *ECG II* is called target signal throughout this document. The *ECG V* and *ECG AVR* signals encompass recordings of a total of 10 minutes. The last 30 seconds of the target signal, which was originally of the same length as the *ECG V* and *ECG AVR* signals, is removed. This last 30 second is the part that we assume to be missing. An illustration of these signals is provided in Fig. 1.

To make it possible for you to evaluate the goodness of your reconstructions, the missing part of *ECG II* is provided under a separate variable. This data will be used for "testing" purposes.

2.2 Part A: RLS

Our aim is to reconstruct the missing part of the target signal $x_T[n] =$*ECG II* signal using $x_1[n] =$ *ECG V* and $x_2[n] =$*ECG AVR*. We will use the recursive least-squares (RLS) algorithm.

Assume that $x_T[n]$ can be written as a linear combination of samples of $x_1[n]$ and $x_2[n]$. Hence, we have the following signal model

$$x_T[n] = \sum_{i=0}^{N} a_i x_1[n-i] + \sum_{i=0}^{M} b_i x_2[n-i] + w[n]$$

where a_i and b_i are unknown coefficients to be determined and $w[n]$ is unknown disturbance. N and M are unknown.

The solution approach we adopt here is as follows:

1. Using the first 9.5 minutes of $x_T[n]$, $x_1[n]$, $x_2[n]$, train the RLS filter and estimate the coefficients a_i and b_i.
2. Freeze the estimates $\hat{a}_i$ and $\hat{b}_i$ at the end of 9.5 minutes.
3. Using the last 30 seconds of $x_1[n]$ and $x_2[n]$, form an estimate of $x_T[n]$ for the last 30 seconds.

$$\hat{x}_T[n] = \sum_{i=0}^{N} \hat{a}_i x_1[n-i] + \sum_{i=0}^{M} \hat{b}_i x_2[n-i]$$

We can evaluate the goodness of our estimate using the true value of last 30 seconds of *ECG II* provided in the course folder. See also Section 2.4.

2.3 Part B

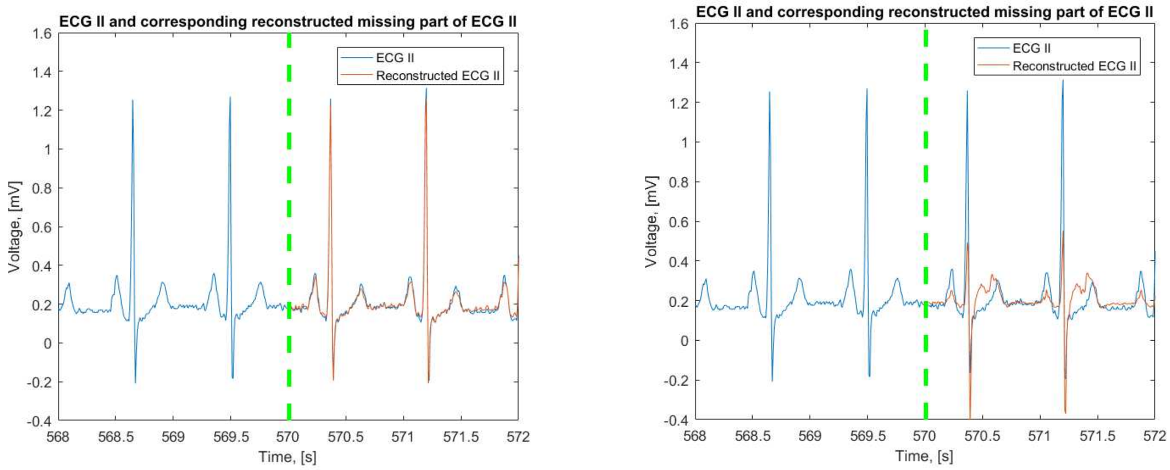
In Part B, your task is to suggest another method for this reconstruction problem and investigate its performance. You also need to compare the performance of the suggested method with RLS. Choose a method that allows real-time operation, i.e. do not use the future values of $x_1[n]$ and $x_2[n]$ while estimating the last 30 seconds of $x_T[n]$.

- **As long as you justify your choice, you can use any of the methods we have covered in the class except plain Least Mean Squares (LMS) and the Normalized LMS (NLMS).** Although plain LMS and the NLMS are applicable to this problem, we prohibit the usage of these methods for Part B to encourage you to explore other ideas.
- Performance of the solution you have proposed will not be used to grade your work. Hence, you can investigate a solution which you expect to be inferior to the RLS.

2.3.1 Suggestions from the course staff

In Part B, you are most encouraged to investigate an aspect of this problem that is *interesting for you*. In any case, below are some ideas that you may use if you want to:

- You can use a modified version LMS with Adam optimizer and compare its performance with that of RLS. Adam optimizer is a stochastic optimization method, which has been introduced in [5]. It has been quite influential, for instance Ref [5] has been cited more than 32700 times although this is a recent article that has first appeared online in 2014. Moreover, Adam optimizer is used in various modern machine learning methods, for instance for optimization of neural networks, and has become a standard option in various software platforms, for instance in TensorFlow. A short overview of Adam optimizer is provided in Section 4.
- You can use Kalman filtering and compare its performance with that of RLS.
- The reconstruction performance is only one aspect of the performance of a solution method. Another aspect is computational time. Throughout the course, we have claimed some methods are better than the others in terms of computational time. You can investigate this claim. For instance, you can define a least-squares problem and try to solve the problem directly using LS formulas instead of using RLS.



(a) A "good" reconstruction

(b) A "not so good" reconstruction

Figure 2: Example Reconstructions

- An important aspect of practical implementations is the quantization of filter coefficients and signal values. This aspect becomes more significant if you want to implement these algorithms with low-cost embedded systems with limited memory. Hence, an investigation of the effect of quantization is an interesting line of study here.
- Looking at the plots of ECG signals, we observe that they seem to be periodic. This suggests that a frequency domain analysis or power spectrum analysis may be particularly useful.

Note that the aim is to pick one aspect/idea and investigate it throughly. Do not try to cover all the ideas here. Pick one of these (or another one of your choice) and study it systematically.

2.4 Evaluation of the Reconstruction Performance

Evaluate the goodness of reconstruction performance using both of the following methods:

- Illustrate the performance by comparing your estimate with true values of the reconstructed signal on a plot for 2-3 patients. Example reconstructions are shown in Fig. 2.
- Report the value of the quality functions *Q1* and *Q2* defined below. These quality functions are used to evaluate the competing solutions in the original Physionet's challenge. For the report, *Q1* and *Q2* values for 8 patients should be reported in a table. A table for both Part A and Part B should be provided in your report.

Given the last 30 second part of the target signal, $x_T[n]$, and the estimated, (i.e. reconstructed), signal $\hat{x}_T[n]$, the first quality function is defined as

$$Q1 = 1 - \frac{mse(x_T[n], \hat{x}_T[n])}{var(x_T[n])}, \tag{1}$$

where $mse(x_T[n], \hat{x}_T[n])$ is the mean square error between the missing and the reconstructed part of *ECG II* and the $var(x_T[n])$ is the variance of the missing part. The second quality function is defined as

$$Q2 = \frac{cov(x_T[n], \hat{x}_T[n])}{\sqrt{var(x_T[n])var(\hat{x}_T[n])}}, \tag{2}$$

where $cov(x_T[n], \hat{x}_T[n])$ is the covariance between the missing and the reconstructed part of the target signal.

Both $Q1$ and $Q2$ is expected to take values satisfying $|Q_1| \leq 1$ and $|Q_2| \leq 1$. Here higher values indicate better reconstruction.

Calculation of $Q1$ and $Q2$ values require you to create empirical estimates of the mean-square error, the covariance and the variances. These emprical values can be calculated by changing the expected values in the definitions of these quantities with sample averages. For instance, $cov(x[k], y[k])$ can be calculated as

$$cov(x[k], y[k]) = \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} (x[k] - m_x)(y[k] - m_y) \tag{3}$$

where $\mathcal{K}$ is the set of indices in the interval we're interested in and $|\mathcal{K}|$ is the cardinality of the set K, i.e. the number of indices in $\mathcal{K}$. Here, m_x and m_y are the mean-values over this interval, i.e. $m_x = \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} x[k]$, $m_y = \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} y[k]$.

If the quality function Q1 or Q2 takes a negative value, set it to 0. Does this make sense? Why/why not?

While reporting the Q1 and Q2 values for different patients, you must keep the parameters of your solution fixed. See Section 5 for further discussions.

Note: You are not allowed to use Matlab function `immse()` for calculation of the above quantities. Implementation of this function changes from one Matlab version to another and causes issues regarding reproducibility of the results.

2.5 Hints

- For Part A, you are not allowed to use MATLAB implementation of the RLS filter. If you want, you can start with the RLS implementation provided under the course folder in the StudentPortal under Lecture Notes, Supplementary Material. Note that you will need to modify that implementation. Note that many of the previous course participants thought that it is actually easier to use that m-file only to understand the RLS filter, and then write your own code from scratch for the project.

- **We suggest you first work with Patient 2.** First, perform the reconstruction of the target signal *ECG_II_2.mat* using only *ECG_AVR_2.mat* which has a high correlation with the target signal. Your implementation should provide good results with this data.

- If you just want to pass the project module with minimum effort, it may be beneficial to pursue the LMS with Adam optimizer idea presented in Section 2.3.1 for Part B. This is suggested due to ease of applying LMS with Adam optimizer at this problem, similarities between RLS and LMS.

- Making the signals zero mean before processing may be a good idea. To do this, you can use the MATLAB function `mean` on the signals and subtract this value from every signal sample. Don't forget to save the mean and add this mean value to every sample of the reconstructed signal. Why do you think this preprocessing is suggested?
- If you are going to investigate aspects related to computational time, you can find the Matlab commands `tic`, `toc`, `profile` useful.

3 Adminstrative Issues

3.1 Overview

The project is to be completed in groups of 1-2. Even if you're going to work with the same group you worked on the assignments, you need to register to a new project group.

The project report and Matlab code should be submitted through Studentportal. The project report should be submitted as a **pdf** file and should not exceed **6 pages** excluding the cover page. It should be typeset, i.e. no handwritten reports are allowed. The Matlab code should be uploaded as executable m-files. Your code should be commented appropriately. Please note that if there are no executable m-files, your first submission will be considered as "Fail".

The submission deadline is 17:00, Dec. 10.

3.2 Consultation Hours

Project consultation hours will be held at Week 48-49 according to the following schedule:

W48: 27/11/2019, 12:15 -13:00: Room 72121, KE-CT-AO
W49: 02/12/2019, 15:15 - 17:00: Room 4101, KE-CT-MH
W49: 4/12/2019, 12:15 -13:00: Room 72121, KE-CT-MH
W49: 5/12/2019, 12:15 -13:00: Room 72121, KE-MH

Both rooms are at Ångström. Kalle Ekström (KE) and Carl Tysk (CT) primarily provide support for Part A. Martin Hellkvist (MH) and Ayça Özçelikkale (AO) provide support for both parts.

3.3 Evaluation

In order to pass the project module, you need to get a Pass grade for the code, the report and the oral exam. The grading rubric that will be used by the course staff for evaluating the code is presented in Section 6.

To be able to pass, you need to present a systematic study of the properties of the reconstruction methods you have implemented and connect your findings to the theoretical results. The following are some desirable properties that will be taken into account during evaluation:

- Submitted implementation successfully reconstructs the signals. Provided you get reasonably good reconstruction results, the reconstruction performance will NOT be used to grade your work.
- Report illustrates that authors' implementation successfully reconstructs the signals.

- Parameters of the signal reconstruction methods are explained appropriately.
- Presentation of the findings is clear.
- Findings are illustrated by plots and quantitative results. These findings are discussed throughly.
- Authors show that they are aware of what the important aspects of an estimation problem/method are.
- Authors show that they are aware of what the important aspects of performance in an estimation problem are.
- Conclusions drawn by the authors are supported by the presented results.
- Observed results are compared with the expected results. Discrepancies between these are acknowledged and discussed appropriately.
- Organization and English usage of the report is satisfactory.
- Plots are labelled appropriately.

These aspects will be evaluated both in oral exams and the report whenever applicable. Please also see the general information document on the assignments/projects on the Student Portal to recall the rules about assignments/projects.

Oral exams will be held on Dec. 12-13. A link for signing-up for the oral exam will be provided by e-mail. Please also see the document on the student portal for further information about the oral exams.

4 Appendix: Adam Optimizer

In this section, we provide a short overview of LMS with Adam optimizer [5]. We use the notation in the LMS lecture notes, which can be found on the Student Portal. Define the filter input at iteration n as $\boldsymbol{y}[n]$ and the filter coefficients at iteration n as $\boldsymbol{h}[n]$. Define the cost function as in plain LMS: $f(\boldsymbol{h}[n-1]) = \frac{1}{2}e^2[n]$ where $e[n]$ is the a-priori estimation error $e[n] = d[n] - \boldsymbol{h}^T[n-1]\boldsymbol{y}[n]$. Let $\boldsymbol{g}_n$ denote the gradient of $f(\boldsymbol{h}[n-1])$ with respect to $\boldsymbol{h}[n-1]$. The Adam optimizer is presented in Algorithm 1.

- All operations that involve vectors are performed element wise. For instance, the squared gradient $\boldsymbol{g}_n^2$ indicates the vector whose elements are the squared elements of $\boldsymbol{g}_n$.
- Note that the decay rates β_1, β_2 are scaled with the iteration index. For instance, β_2^n is the n^{th} power of β_2.
- Keep in mind that the variable $\boldsymbol{g}_n$ is a vector with the same dimensions as $\boldsymbol{h}[n]$, and so are the variables $\boldsymbol{m}_n, \boldsymbol{v}_n, \hat{\boldsymbol{m}}_n, \hat{\boldsymbol{v}}_n$.
- Start your experiments with the following values $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$. Ref. [5] claims that these values are universally good default choices for these parameters. Do your results support this claim? Why/why not?
- How should you change Algorithm 1 so that it corresponds to plain LMS?

Algorithm 1: Adam optimizer

Initialize: Set the initial values for the filter coefficients $\boldsymbol{h}[0]$. Set the initial values $\boldsymbol{m}_0 = \boldsymbol{v}_0 = \boldsymbol{0}$ for the first and second moment estimates. Initialize the step size α, the decay rates β_1, β_2 and ϵ.

for $n = 1$ *to* N **do**

$\quad \boldsymbol{m}_n = \beta_1 \boldsymbol{m}_{n-1} + (1-\beta_1)\boldsymbol{g}_n$

$\quad \boldsymbol{v}_n = \beta_2 \boldsymbol{v}_{n-1} + (1-\beta_2)\boldsymbol{g}_n^2$

$\quad \hat{\boldsymbol{m}}_n = \boldsymbol{m}_n/(1-\beta_1^n)$

$\quad \hat{\boldsymbol{v}}_n = \boldsymbol{v}_n/(1-\beta_2^n)$

$\quad \boldsymbol{h}[n] = \boldsymbol{h}[n-1] - \alpha\,\hat{\boldsymbol{m}}_n/(\sqrt{\hat{\boldsymbol{v}}_n} + \epsilon)$

end

return $\boldsymbol{h}[N]$

5 Appendix: Further Information About the Parameter Choices

While reporting the Q1 and Q2 values for different patients, you must keep the parameters of your solution fixed. This is due to the fact that in a practical application scenario, we will not have the chance to tailor our system parameters to each patient. Imagine yourself implementing this solution in a ECG monitoring device; the device will not have access to true signal values for the missing part and you will not be there to change the parameters.

Hence, it is desirable that you try different parameters with different patients to see the effect of parameters for different signals. This will allow you to get insights to the problem. Based on these observations, you need to make a reasonable choice about which parameters your solution will use in the end.

To sum up, we would like to have a general idea about reasonable parameters to use but this is not the main of the project. Instead of spending too much time on this aspect, we suggest you reflect on how your observations connect to theory we covered in this class and the other aspects in Section 3.3.

6 Appendix: Grading rubric for the code

Your code should be consistent with the instructions given in this document, such as real-time processing, and the results you have presented in your report. We will perform the first evaluation of the code according to the below rules. All of these are necessary conditions for passing the project.

We check the following for both Part A and Part B whenever applicable:

- There exists Matlab code for the method the report presents.
- There exists Matlab code for calculating scalar Q1 and Q2 values.
- The parameters of the methods and the code for finding Q1/Q2 values are clearly labelled in the code.
- The method uses both signals, i.e. $x_1[n]$ and $x_2[n]$, to create the estimates.

- The method can handle different N and M values (checked by running the code for arbitrary N and M values).
- The reported values in the tables match the values the code produces (checked by running the code for some of the scenarios using the values of the parameters presented in the report).

References

[1] "Physionet challenge: Mind the gap." https://physionet.org/challenge/2010. Accessed 2017-10.

[2] G. B. Moody, "The physionet/computing in cardiology challenge 2010: Mind the gap," in *Computing in Cardiology, 2010*, pp. 305–308, 2010.

[3] A. L. Goldberger, L. A. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "Physiobank, physiotoolkit, and physionet," *Circulation*, vol. 101, no. 23, pp. 215–220, 2000.

[4] I. Silva, "PhysioNet 2010 challenge: a robust multi-channel adaptive filtering approach to the estimation of physiological recordings," in *Computing in Cardiology, 2010*, pp. 313–316, 2010.

[5] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations (ICLR)*, 2015. Latest version available online: https://arxiv.org/abs/1412.6980.