POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

HOMEWORK REPORT

Homework 1

SCIENTIFIC COMPUTING TOOLS FOR ADVANCED MATHEMATICAL MODELLING

Authors: Andrea Moreschi - Randeep Singh - Daniela Zanotti

Academic year: 2021-2022

---

1. Mathematical formulation of the problem

Anti-tachycardia pacing (ATP) delivers pacing pulses to interrupt a tachyarrhythmia episode and restore normal sinus rhythm. Devices such as the implantable cardioverter-defibrillator (ICD) employ this technique before delivering high-voltage shocks to reduce the patient's sensation of pain and extend the life of the device's battery. To ensure that ATP is effective, these devices must be programmed by selecting the proper pacing pulse width, duration and timing. Note that it is preferred to keep the impulse limited, to reduce battery consumption.
The goal of this homework is to identify an optimal ATP strategy based on a single impulse.
In order to find the optimal timing and duration of the impulse for each different patient, we can decompose the problem into two separate sub-problems:

1. Estimation of the value of the parameter ν_2 for each one of the three patients;
2. Computation of the impulse timing and duration using the three values of ν_2 found above.

After solving the sub-problems we check the robustness of the found solutions with the following approaches:

1. We check whether the timing and duration are robust with respect to small variations of the optimal ν_2, since the latter is estimated using noisy data;
2. We check if small variations of the optimal timing and duration generate a pacing pulse which is still effective, since the ATP may not be able to deliver pacing pulse having the exact computed timing and duration.

Both the sub-problems can be seen as *parameter estimation* problems.

Estimate of ν_2

Since for each patient we have a noisy observation of the ECG, ν_2 can be computed minimizing the mean squared error ($\mathbf{MSE}(\mathbf{x}, \mathbf{y}) = \frac{1}{N}\sum_{i=0}^{N}(x_i - y_i)^2$) between the numerical simulation and the data in the measurement window $(0, 450)$ ms. In particular, letting $\Omega = [0.0116, 0.0124]$ and indicating with τ the time variable for the observing window:

$$\nu_2 = \underset{\nu\in\Omega}{\arg\min} \quad \mathbf{MSE}(ECG_{sim}(\nu), ECG)|_{\tau<450}$$

Estimate of t_{opt} and d_{opt}

Since stopping the tachycardia means having a flat ECG, the optimal impulse timing t_{opt} and duration d_{opt} can be computed by minimizing the mean squared error between the numerical simulation, with fixed value of ν_2, and the flat signal in the observing window $(600, 800)$ ms. In particular, since we would like to have the duration of the impulse as small as possible to preserve the battery, we add a penalization term for it. Letting

$\Omega = [450, 525] \times [0, 10]$, $\lambda = 0.5$ and indicating with τ the time variable for the observing window, we have:

$$(t_{opt}, d_{opt}) = \underset{(t,d)\in\Omega}{\arg\min} \quad \mathbf{MSE}(ECG_{sim}(t,d), 0)|_{600<\tau<800} + \lambda \left(\frac{d}{10}\right)^2$$

Observe that since we are considering normalized ECG, in the loss function we consider also a normalization of the duration d.
To improve even more the duration of the pacing pulse, a *constrained* optimization step can be performed. Fixing the timing estimated in the previous step, let us define $\Omega_d = [0, d_{opt}]$. A new optimal duration $\hat{d}_{opt}$ can be computed as:

$$\hat{d}_{opt} = \underset{d\in\Omega_d}{\arg\min} \quad \mathbf{MSE}(ECG_{sim}(t_{opt}, d), 0)|_{600<\tau<800} + \lambda \left(\frac{d}{10}\right)^2$$

This trick allows to explore more accurately the loss function for small values of the duration. As further improvements for this step, we can consider $\Omega_d = [\epsilon, d_{opt}]$, $\epsilon > 0$ small enough, and a bigger penalization term λ.

2. Methods

Bayesian Optimization

Bayesian optimization (BO) is a powerful global optimization technique used to find a good approximation of the global optimum of a function *f*. It is an iterative method which does not assume any specific form of *f*, since it does not need to compute derivatives.

In particular, BO treats the objective function f as a random function and assign a prior to it, which encapsulate our beliefs about the overall behaviour of the function. Then, the method sequentially gather function evaluations, and at each step the prior random function is updated into a posterior distribution over the objective function, using all the function evaluations collected up to the last step. The points where to acquire evaluations are determined through the *acquisition function*, which is build upon the last updated posterior distribution.
Typically, the prior-posterior updating mechanism is done by means of Gaussian Processes which also provide a useful tool to build the acquisition function, namely the *Upper Confidence Bound* method.

Gaussian Process

A Gaussian process is defined as a probability distribution over functions *y(x)* such that the set of values of *y(x)* evaluated at an arbitrary set of points $x_1, ..., x_N$ jointly have a Gaussian distribution.

A key advantage of Gaussian processes is that the joint distribution over N variables $y_1, ..., y_N$ is completely specified by the second-order statistics, namely the mean and the covariance. The covariance of *y(x)* (assuming, without loss of generality, zero mean function) is evaluated at any two values of x and is given by the kernel function:

$$\mathbb{E}[y(x_n)y(x_m)] = k(x_n, x_m)$$

We can define the kernel function directly. The one that we will use later is the Matern kernel, that is given by

$$k(x_i, x_j) = \frac{1}{\Gamma(\nu)2^{\nu-1}} \left(\frac{\sqrt{2\nu}}{l} d(x_i, x_j)\right)^{\nu} K_{\nu}\left(\frac{\sqrt{2\nu}}{l} d(x_i, x_j)\right)$$

where:

- l is the length scale
- $d(x_i, x_j)$ is the Euclidean distance
- $K_\nu(\cdot)$ is a modified Bessel function
- $\Gamma(\cdot)$ is the Gamma function
- ν is a parameter controlling the smoothness of the learned function.
  The smaller ν, the less smooth the approximated function is. Important intermediate values are $\nu = 1.5$ (once differentiable functions) and $\nu = 2.5$ (twice differentiable functions).

Gaussian Process Upper Confidence Bound

Assume that our goal is to maximize a function f over a domain D. However, we can only observe noisy estimates of the function:

$$y_t = f(x_t) + \varepsilon_t \quad \text{where } \varepsilon_t \overset{iid}{\sim} N(0, \sigma^2 I) \tag{1}$$

We can approach this problem by modeling the underlying function $f(\cdot)$ as a Gaussian process, with a mean function $\mu(\cdot)$ capturing our estimate of the function across the domain and a variance function $\sigma(\cdot)^2$ capturing our uncertainty in this estimate.

Algorithm 1 GP - UCB

Input: Function to maximize f; Function input space D; GP prior $\mu_0 = 0,\ \sigma_0,\ k(x, x') \leq 1\,\forall x,\ x'$
  for $t = 1, 2, ...$ **do**
    Choose $x_t = \arg\max_{x \in D} \mu_{t-1}(x) + \sqrt{\beta_t}\sigma_{t-1}(x)$
    Examine $y_t = f(x_t) + \varepsilon_t,\ \varepsilon_t \overset{iid}{\sim} N(0, \sigma^2 I)$
    Perform Gaussian process Bayesian update with $(x_t,\, y_t)$ to obtain μ_t and σ_t
  end for

β_t is an increasing function in t and captures the number of standard deviations in the upper confidence bounds [1]. Actually, in most cases β_t is just a constant and for us $\beta_t = 2.576$.

Our implementation

We have decided to minimize the loss functions defined before through a Bayesian optimization approach (in practice the algorithm will maximize the loss function with changed sign), constructing the posterior distribution of the loss by means of a Gaussian Process, using the Matern 2.5 kernel.
We perform a few completely random iterations sampling uniformly in the parameter space and then we sample from a Sobol' sequence in order to explore the space as much as possible.
When the optimization starts, at each step a Gaussian Process is fitted to the known samples, and the next point to be explored is determined by the posterior distribution combined with an exploration strategy, in particular the *Upper Confidence Bound* (UCB).
This trade-off between *exploration* and *exploitation* is one of the best way to minimize the number of steps required to reach the optimal value of the parameters.
See Algorithm 2 for a detailed explanation.

Algorithm 2 Bayesian optimization

Input: Loss f, N_{rand}, N_{Sobol}, N, Parameter space Ω
Output: x_{best}

```
y_best = +∞
Initialize a Gaussian Process as prior distribution for f
for i = 1 to N_rand do
    Select x_i randomly in Ω
    Compute loss function y_i ← f(x_i)
    Update the Gaussian Process with the new data (x_i, y_i)
end for
for i = N_rand + 1 to N_rand + N_Sobol do
    Select x_i from a Sobol' sequence in Ω
    Compute loss function y_i ← f(x_i)
    Update the Gaussian Process with the new data (x_i, y_i)
end for
for i = N_rand + N_Sobol + 1 to N_rand + N_Sobol + N do
    Select x_i maximizing the UCB
    Compute loss function y_i ← f(x_i)
    Update the Gaussian Process with the new data (x_i, y_i)
    if y_i < y_best then
        x_best ← x_i
        y_best ← y_i
    end if
end for
```

3. Numerical results

Estimate of ν_2

The estimated ν_2 for the three patients are respectively:

- 0.01234
- 0.01191
- 0.01171

The loss functions predicted by the Gaussian Process are reported below.

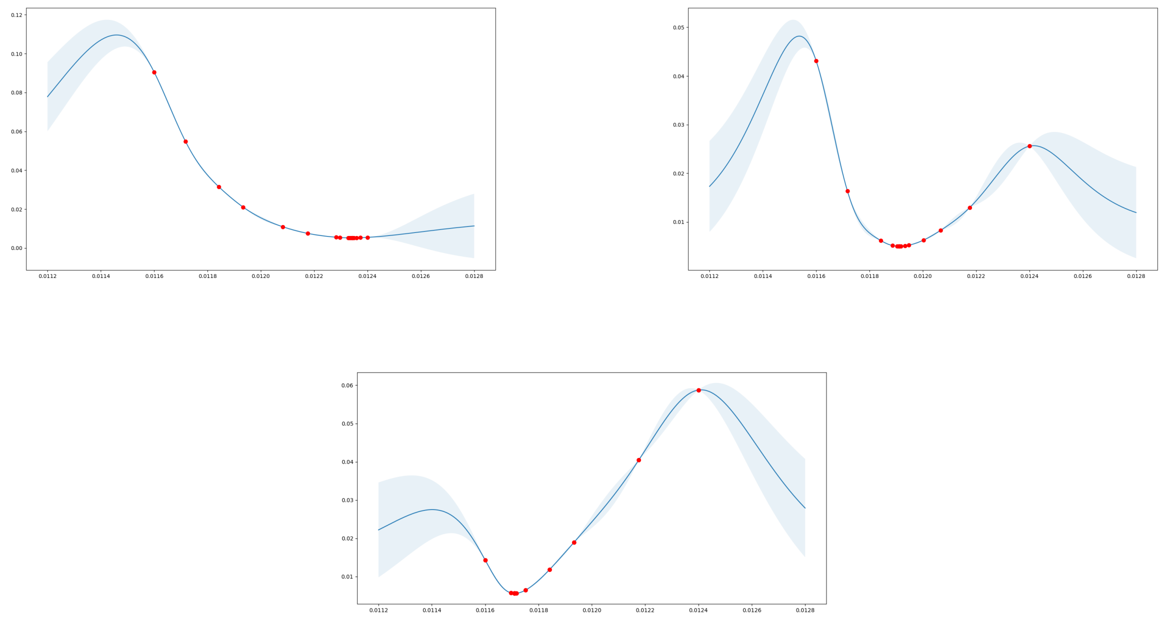


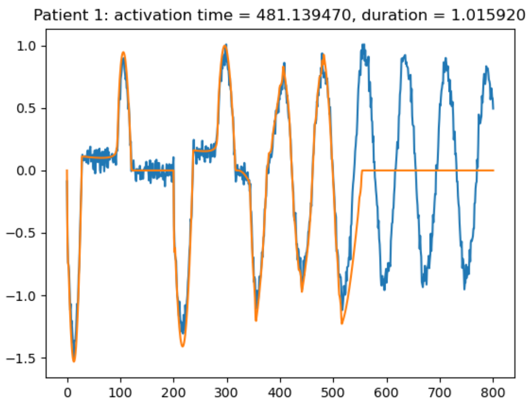
Figure 1: Predicted loss functions for the three patients:

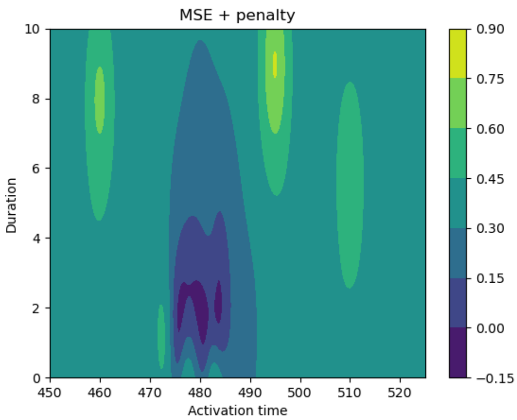
Estimate of t_{opt} and d_{opt}

For each patient we report the predicted impulse timing and duration, together with the plots of the patient's heartbeat (from which we can see the effectiveness of the impulse) and the 2D representation of the loss function fitted by the Gaussian Process.

Patient 1

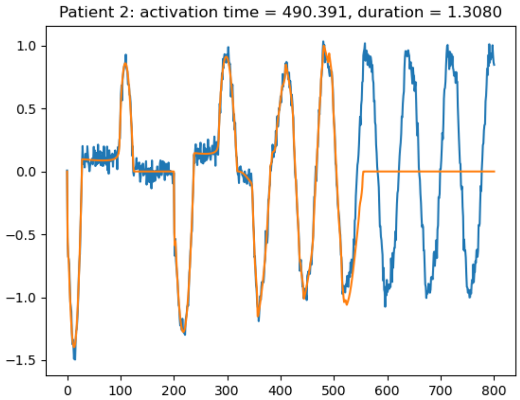
The estimated impulse timing and duration are: [481.139470, 1.015920]

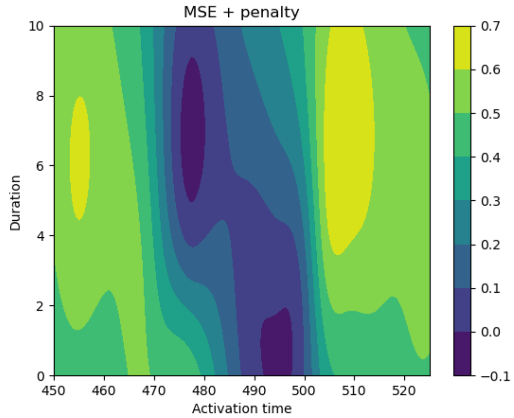




Patient 2

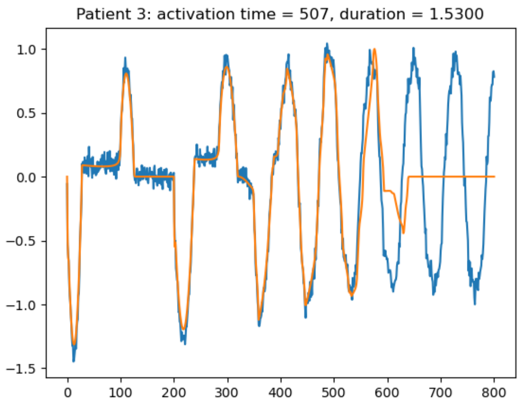
The estimated impulse timing and duration are: [490.3910, 1.3080]

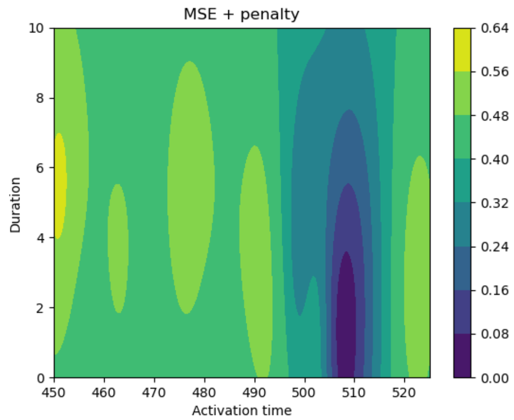




Patient 3

The estimated impulse timing and duration are: [507, 1.5300]





4. Conclusions

As regards the first sub-problem, we observe that our approach works well for all patients and it is quite robust to noisy observation data (ECGs). However, in principle, we should perform smoothing of the input data, by means of a filter (e.g., Savitzky-Golay) or a moving-window, especially if the underlying noise it is not guaranteed to be *white*. Anyhow, we have checked that the timing and duration are robust with respect little variations of ν_2.

Moreover, also the solutions for the second sub-problem are quite strong. We can prove that small variations of the timing and durations (up to the first decimal) provide a pacing pulse which is still able to control the tachycardia.

References

[1] Niranjan Srinivas, Andreas Krause, Sham M. Kakade, and Matthias W. Seeger. Information-theoretic regret bounds for gaussian process optimization in the bandit setting. *IEEE Transactions on Information Theory*, 58(5):3250–3265, may 2012. doi: 10.1109/tit.2011.2182033.

[2] Rogers JM and McCulloch AD. Galerkin finite element model of cardiac action potential propagation. *Trans Biomed Eng.*, 1994.

[3] Fernando Nogueira. Bayesian Optimization: Open source constrained global optimization tool for Python, 2014–. URL `https://github.com/fmfn/BayesianOptimization`.