TU Delft
Faculty of Electrical Engineering, Mathematics, and Computer Science
Circuits and Systems Group

ET4350 Applied Convex Optimization

ASSIGNMENT

Change Detection in Time Series Model

1 Context

In statistical signal processing, change point detection tries to identify time instances when the probability distribution of a stochastic process or time series changes. The problem in this assignment concerns both detecting whether or not a change has occurred, or whether several changes might have occurred, and identifying the times of any such changes.

This exercise consists of two parts: (a) formulate the step detection problem as a suitable convex optimization problem; and (b) implement the change detector. In a group of 2 students, make a short report (4-5 pages; pdf file) containing the required Matlab scripts, plots, and answers. Also, prepare a short presentation to explain your results and defend your choices.

Dataset explanation

Consider the following scalar autoregressive (AR) time-series model

$$y(t+3) = a(t)y(t+2) + b(t)y(t+1) + c(t)y(t) + v(t); \quad v(t) \sim \mathcal{N}(0, 0.5^2).$$

The assumption is that the AR coefficients are piecewise constant and change infrequently. Given $y(t), t = 1, \ldots, T$, the problem is to estimate $a(t)$, $b(t)$, and $c(t)$, but by taking into account the structure of the AR coefficients (i.e., that they are piecewise constant). The dataset `change_detection.mat` includes observations for $T = 300$ samples as well as the true AR coefficients.

2 Assignment

You will have to answer the following questions:

1. (2 pts) Formulate the change detection problem as an optimization problem. Suggest a suitable convex approximation (i.e., derive a convex relaxed problem) if the true problem is not convex. Motivate the proposed formulation as well as the relaxation.

2. (2 pts) Implement the proposed convex optimization problem in your favorite off-the-shelf solver (e.g., `CVX`, `SeDuMi`, or `YALMIP`). How does this ready-made software solve your problem? Comment on the number of iterations, CPU time, and algorithm the ready-made solver uses.

3. (5 pts) Implement a low-complexity algorithm (e.g., projected (sub)gradient descent for the above problem, or provide a first-order algorithm to solve the primal and dual problems). Compare the obtained results with the solutions from the off-the-shelf solver. Comment on the number of iterations, CPU time, and convergence of your low-complexity algorithm.

4. (1 pt) Make a short presentation explaining your results clearly in 5 minutes.

3 Consultant

ir. M.A. (Mario Alberto) Coutino
Tel: +31 15 27 84327
Email: m.a.coutinominguez@tudelft.nl
Room: HB 17.070
Office hours: By prior appointment only.