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

ET4350 Applied Convex Optimization

ASSIGNMENT

Multidimensional Scaling for Localization

1 Context

Multidimensional scaling is a well-known algorithm for localization in wireless sensor networks. In this application example, you are given the time, in minutes, that the NS intercity train takes to travel between some Dutch cities. Given this travel time information, you can use MDS to compute the location of the train stations.

Classical MDS, based on eigenvalue decomposition, can be used to compute these locations; however, as the trains don't travel with constant speed, or in straight lines, the location estimates can be quite erroneous. Convex optimization might be of help here (e.g., you can add box constraints for coordinates).

This exercise consists of two parts: (a) formulate the localization problem as a suitable convex optimization problem; and (b) implement the proposed algorithm. 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

The dataset `mds_train.mat` includes the travel time in minutes between 5 NS train stations. The true location information (saved as `coord`) as well as the all the pairwise distances (saved as `distance`) between the train stations are provided. Names of the NS stations are saved as the variable named `station_index`. To begin with, use the variable `time_matrix` to compute the location of the NS stations using a suitable convex optimization algorithm. Verify if the location estimates are better if you use the true pairwise distances instead of the travel time.

Hint: The classical MDS provides location estimates up to a rotation and translation. To correct the rotation and translation use the Matlab command:

`[Dt,Z] = procrustes(coord, estimated_coodinates)`

Here, the `estimated_coodinates` denote the location estimates from MDS.

2 Assignment

You will have to answer the following questions:

1. (2 pts) Formulate the above MDS 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 Reference

- I. Dokmanic I, R. Parhizkar, J. Ranieri, M. Vetterli, "Euclidean distance matrices: essential theory, algorithms, and applications," IEEE Signal Processing Magazine, vol. 32, no. 6, pp. 12-30, Nov. 2015.

- Z.-Q. Luo, W.-K. Ma, A. Man-Cho So, Y. Ye, and S. Zhang, "Semidefinite Relaxation of Quadratic Optimization Problems," IEEE Signal Processing Magazine, vol. 27, no. 3, May 2010.

4 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.