Course mini-project: Addendum to the description

This note provides a correction to the last two points that were asked in the project description:

5) Estimate the (normalized) mean squared error (MSE) $\mathbb{E}(\frac{1}{N}\sum_{i=1}^{N}(\hat{x}_i - X_i)^2)$ *reached by your algorithm, by running the whole thing multiple times for various realizations of* Y*. Draw the result as a function of the parameter* $\lambda > 0$*.*

6) Is there a limiting value $\lambda_c > 0$ *below which your algorithm does not perform better than a purely random guess?*

The fact is: if you run the Metropolis algorithm (or Glauber dynamics) as described, then if you draw the MSE as a function of λ, you will find something surprising! We let you discover it... The following fact will help you understand what you see: for any given Y, the energy function $H_Y(x)$ is symmetric in x: $H_Y(-x) = H_Y(x)$, which implies that this function is insensible to a global sign flip of the whole vector x.

What turns out to be more interesting is to draw both the *distribution* of the sqarred error for different values of λ, as well as the *standard deviation* of the squared error (SDSE) as a function of λ. What do you observe?

As a reminder:

$$\begin{cases} \text{squared error: } \mathrm{SE} = \frac{1}{N}\sum_{i=1}^{N}(\hat{x}_i - X_i)^2 \\ \text{mean squared error: } \mathrm{MSE} = \mathbb{E}(\mathrm{SE}) \\ \text{standard deviation of the squared error: } \mathrm{SDSE} = \sqrt{\mathbb{E}(\mathrm{SE}^2) - \mathbb{E}(\mathrm{SE})^2} \end{cases}$$

Needless to say, one does not compute the true MSE or SDSE in pratice, but just estimates of these quantities.

Some other precisions about the project:

- You should draw the ground truth vector X_0 at random, with i.i.d. uniform components in $\{-1, +1\}$.
- Irrespective of the schedule you choose for simulated annealing, the number of iterations in the algorithm should at least scale with the size N of the data, if you want to see something interesting for large N. You have the choice to increase it further if you want to get even better results at large N (but this might come of course at the expense of larger computation times).
- Contrary to what was stated in the problem description, the influence of the initial condition on the performance of the algorithm is perhaps not so important, if you happen to have a good scheduling for the simulated annealing. But you are again free to choose among all possible strategies here.