Stochastic Optimization

This exercise sheet is about applying simulated annealing and a deterministic approximation thereof to a toy problem.

8.1 Simulated Annealing (5 points)

Simulated annealing can be used to optimize a *cost function* $E : \boldsymbol{s} \to \mathbb{R}$ where the state $\boldsymbol{s}$ is a set of discrete state variables $s_i \in \{-1, +1\},\ i = 1, \ldots, N$. For a fully connected "network" with $N = 6$ binary nodes, this means that $\boldsymbol{s} \in \{-1, +1\}^6$, and we will use the cost ("energy")

$$E(\boldsymbol{s}) = -\frac{1}{2} \sum_{i,j=1}^{N} w_{ij} s_i s_j,$$

where $w_{ij} = w_{ji} \in \mathbb{R}$, and $w_{ii} = 0$.
The probability that the network is in a state $\boldsymbol{s}$ with energy $E(\boldsymbol{s})$ is given by

$$P(\boldsymbol{s}) = \frac{1}{Z} \exp\big(-\beta E(\boldsymbol{s})\big),$$

where the *partition function* Z guarantees $P(\boldsymbol{s})$ to be a valid probability mass function and is given as the sum over all possible configurations, i.e., $Z = \sum_{\boldsymbol{s}} \exp\big(-\beta E(\boldsymbol{s})\big)$.

Write a program that finds the optimal configuration $\boldsymbol{s}$ for a given set of weights $\boldsymbol{W}$ as follows (try to find reasonable parameters β_0 and τ to have a rather efficient implementation if possible):

Initialization:

- set $\beta_0, \tau > 1$, and t_{max} (make sure β_0 is small enough), try both $M = 1$ and $M = 500$
- set initial state $\boldsymbol{s}$ randomly; $\boldsymbol{W}$ arbitrary, but symmetrically and with zero diagonal

Optimization: for each iteration $t = 0, ..., t_{max}$

- repeat the following M times (state update loop):
  - select node i randomly
  - determine the energies for the two options of state s_i and compute their difference

    $$E_{s_i} = -\frac{1}{2} \sum_{j \in \mathcal{N}_i} w_{ij} s_i s_j \qquad \text{and} \qquad E_{-s_i} = -E_{s_i} \qquad \to \qquad \Delta E = E_{-s_i} - E_{s_i}$$

    where $\mathcal{N}_i$ is the set of neighbors of node i, i.e. here the set of all other nodes
  - flip state s_i with probability $P(s_i \to -s_i) = (1 + e^{\beta_t \Delta E})^{-1}$
- increase β using $\beta_{t+1} = \tau \beta_t$

Plotting:

- Plot the temperature $T_t = \frac{1}{\beta_t}$ and the energy $E(\boldsymbol{s})$ over the iterations $t = 0, \ldots, t_{max}$.
- Show the energy $E(\boldsymbol{s})$ for all possible 2^6 states using a bar plot. The sequence of the states is not relevant. Additionally, plot the probabilities $P(\boldsymbol{s})$ for different β as a bar plot. Choose the $\{\beta\}$ in a way, that the probability distributions differ discernibly.

8.2 Mean-Field Annealing (5 points)

Mean-field annealing is a deterministic approximation of simulated annealing. During optimization each node i corresponds to the first moment of s_i and thus takes continuous instead of binary values. The states s_i represent the mean with respect to the factorized distribution $Q(\boldsymbol{s}) \approx P(\boldsymbol{s})$.

Consider again a fully connected network with $N = 6$ nodes. The cost (energy) function remains the same:

$$E(\boldsymbol{s}) = -\frac{1}{2} \sum_{i,j=1}^{N} w_{ij} s_i s_j,$$

where the $w_{ij} \in \mathbb{R}$ are symmetric, and $w_{ii} = 0$. The approximated probability of a state $\boldsymbol{s}$ is now given by

$$Q(\boldsymbol{s}) = \frac{1}{Z_Q} \exp\Big(-\beta \sum_j e_j s_j \Big).$$

Write a program that finds the optimal configuration $\boldsymbol{s}$ of the network for given weights $\boldsymbol{W}$. It should execute the following steps:

Initialization:

- β_0 small enough, $\tau > 1$, set t_{max}, ε small enough
- initial state $\boldsymbol{s}$ randomly, $\boldsymbol{W}$ from exercise above

Optimization: for each iteration $t = 0, ..., t_{max}$

- repeat the following until convergence, i.e., $|\boldsymbol{e}_{\text{new}} - \boldsymbol{e}_{\text{old}}| < \varepsilon$
  - for $i = 1, \ldots, N$:
    - compute mean-field

      $$e_i = -\sum_{j \in \mathcal{N}_i} w_{ij} s_j$$

      where $\mathcal{N}_i$ is the set of neighbors of node i, i.e. here the set of all other nodes.
    - update the state using $s_i = \tanh(-\beta e_i)$
- Increase β using $\beta_{t+1} = \tau \beta_t$

Plotting:

- Plot the temperature $T_t = \frac{1}{\beta_t}$ and the energy E_t over the iterations $t = 0, \ldots, t_{max}$.

Comparison: for simulated and mean-field annealing compare:

- the required number of iterations until convergence
- the total runtime (in dependence of different choices of M, τ, ε)
- the final result $\boldsymbol{s}$ after iteration $t_{\max}$

Total points: 10