Algorithm 13.6 Efficient min-sum message passing for untruncated 1-norm energies

Procedure Msg-Truncated-1-Norm (c // Parameters defining the pairwise factor
                                 h_i(x_i) // Single-variable term in equation (13.36)

for x_j = 1, . . . , K - 1
    r(x_j) ← min[h_i(x_j), r(x_j - 1) + c]

for x_j = K - 2, . . . , 0
    r(x_j) ← min[r(x_j), r(x_j + 1) + c]

return (r)

Exercise 13.17

Consider the task of passing a message over an edge X_i—X_j in a metric MRF; our goal is to make the message passing step more efficient by exploiting the metric structure. As usual in metric MRFs, we consider the problem in terms of energies; thus, our message computation takes the form:

\[
\delta_{i \rightarrow j}(x_j) = \min_{x_i} (\epsilon_{i,j}(x_i, x_j) + h_i(x_i)),
\]

where \( h_i(x_i) = \epsilon_i(x_i) + \sum_{k \neq j} \delta_{i \rightarrow j}(x_k) \). In general, this computation requires \( O(K^2) \) steps. However, we now consider two special cases where this computation can be done in \( O(K) \) steps.

a. Assume that \( \epsilon_{i,j}(x_i, x_j) \) is an Ising energy function, as in equation (4.6). Show how the message can be computed in \( O(K) \) steps.

b. Now assume that both \( X_i, X_j \) take on values in \{0, . . . , K - 1\}. Assume that \( \epsilon_{i,j}(x_i, x_j) \) is a nontruncated 1-norm, as in equation (4.7) with \( p = 1 \) and dist_{max} = \( \infty \). Show that the algorithm in algorithm 13.6 computes the correct message in \( O(K) \) steps.

c. Extend the algorithm of algorithm 13.6 to the case of a truncated 1-norm (where dist_{max} < \( \infty \)).

Exercise 13.18

Consider the use of the branch-and-bound algorithm of appendix A.4.3 for finding the top \( K \) highest-probability assignments in an (unnormalized) distribution \( \tilde{P}_\Phi \) defined by a set of factors \( \Phi \).

a. Consider a partial assignment \( y \) to some set of variables \( Y \). Provide both an upper and a lower bound to \( \log \tilde{P}_\Phi(y) \).

b. Describe how to use your bounds in the context of a branch-and-bound algorithm to find the MAP assignment for \( \tilde{P}_\Phi \). Can you use both the lower and upper bounds in your search?

c. Extend your algorithm to find the \( K \) highest probability joint assignments in \( \tilde{P}_\Phi \). Hint: Your algorithm should find the assignments in order of decreasing probability, starting with the MAP. Be sure to reuse as much of your previous computations as possible as you continue the search for the next assignment.