CAP 6617 ADVANCED MACHINE LEARNING

NAME: SANKET KUMAR   UF ID: 4513-3352

FACTOR GRAPHS AND THE SUM-PRODUCT ALGORITHM

AIM: This paper provides a tutorial introduction to factor graphs and the sum-product algorithm, a simple way to understand a large number of seemingly different algorithms that have been developed in computer science and engineering.

- **A factor graph** is a bipartite graph that expresses the structure $g(x_1, \ldots, x_n) = \prod f_j(x_j)$ where j belongs to a discrete index set and $f_j(X_j)$ is a function having elements of subsets of $\{x_1, \ldots, x_n\}$ as arguments. A factor graph has a node for each variable x_i, a node for each function f_j and an edge connecting variable to function node if and only if x_i is an argument of f_j. The summary of a specific variable is given by the summation of the function over all the variables except the variable under consideration.
- **The sum-product algorithm** is a message passing algorithm for performing inference on graphical models. It calculates the marginal distribution for each unobserved node and conditional distribution for each observed node. A message from the variable node to the factor node is the product of the messages from all other neighboring factor nodes except the recipient. A message from a factor node to a variable node is the product of the internal factor with messages from all other nodes, marginalized over all variables except the variable under consideration. Upon convergence, the estimate joint marginal distribution of the set of variables belonging to one factor is proportional to the product of the factor and the messages from the variables:
$$\mu_{f \to x}(x) =, \sum_{\sim\{x\}} \left(f(X) \prod_{y \in n(f) \setminus \{x\}} \mu_{y \to f}(y) \right)$$
- In probabilistic modelling, a factor graph can be used to represent the joint p.m.f. of the variables that comprise the system. In behavioral modelling, system behavior is specified in the set-theoretic terms by specifying which particular configurations of variables are valid. A factor graph can represent the "indicator" function for the given behavior.
- Some well-known algorithms may be viewed as special cases of sum-product algorithm.
  Forward/Backward algorithm is an inference algorithm for hidden Markov models which computes posterior marginals of all hidden variables given a sequence of observations. A factor graph can be drawn for *a posteriori* joint p.m.f given an observation, that represents the factorization of the function. Since the proportionality of the marginal probabilities is the same as in sum-product algorithm and if the factor graph is cycle-free, marginal functions can be computed by applying sum-product algorithm to the factor graph.
  The Viterbi algorithm is a DP algorithm for finding the most likely sequence of hidden states, called the Viterbi path, that results in a sequence of observed events. The codomain of a global function represented by the factor graph may be any semiring with two operations "sum" and "product". In Maximum Likelihood Sequence Detection (MLSD), the "product" becomes "sum" and "max" becomes "min". Now applying the sum-product algorithm in the min-sum semiring yields the same message flow as in the forward/backward algorithm. So, the basic operation here is "minimum of sums" rather than "sum of products".
  Kalman Filtering is an algorithm that uses a series of measurements observed over time and estimates a joint probability distribution over the variables for each timeframe. The updates used by a Kalman filter are similar to any cycle-free factor graph in which all distributions are Gaussian. The operation of sum-product algorithm in such a graph can be regarded as a generalized Kalman filter, and in a graph with cycles as an iterative approximation to the Kalman filter. The results of sum-product algorithm in a factor graph with cycles cannot in general be interpreted as exact function summaries. For factor graphs with cycles, we assume that messages are synchronized with a global discrete-time clock. A message sent from node v at time i will be a function only of the local function at v and the messages received at v prior to i. A message passing schedule is implemented which specifies messages to be passed during each clock tick.
- Transformations may be applied to a factor graph to modify it into a more convenient form, for example: from cycles to cycle-free, but at the expense of increasing the complexity of local functions. Factor graphs provide a natural graphical description of a global function into a product of local functions. Factor graphs can be applied to a wide range of application areas.