CSCI 2950-P Homework 1: Belief Propagation, Inference, & Factor Graphs

Brown University, Spring 2013

Homework due at 11:59pm on March 1, 2013

In this problem set, we focus on the problem of computing marginal distributions in graphical models, possibly given observations of the variables at some nodes. We will use factor graph representations of the target models, and implement the sum-product variant of the belief propagation algorithm to compute marginal distributions. To understand the details of the sum-product algorithm, we recommend Sec. 4.2 of Jordan's "An Introduction to Probabilistic Graphical Models", as well as "Factor Graphs and the Sum-Product Algorithm" by Kschischang, Frey, & Loeliger, *IEEE Trans. Information Theory* 47, pp. 498-519, 2001.

We have provided Matlab code which implements a data structure to store the graph adjacency structure, and numeric potential tables, defining any discrete factor graph. We recommend (but do not require) that you use these same data structures for your own sum-product implementation. You may also use other programming languages if you wish, but if so you will be responsible for translating the Matlab-compatible data we provide into other formats. *In all cases, you must write your own novel implementations of the inference algorithms specified below, not copy code from existing software packages.*

Question 1:

a) *Implement the sum-product algorithm. Your code should support an arbitrary factor graph linking a collection of discrete random variables. Use a parallel message update schedule, in which all factor-to-variable messages are updated given the current variable-to-factor messages, and then all variable-to-factor messages given the current factor-to-variable messages. Initialize by setting the variable-to-factor messages to equal 1 for all states. Be careful to normalize messages to avoid numerical underflow.*

b) *Write code which explicitly computes a table containing the probabilities of all joint configurations of the variables in a factor graph. Also write code which sums these probabilities to compute the marginal distribution of each variable. Such "brute force" inference code is of course inefficient, and will only be computationally tractable for small models.*

c) *Create a small, tree-structured factor graph linking four variable nodes. Use this model to verify that the algorithms implemented in parts (a) and (b) are consistent with each other, and compute correct marginal distributions. Design your factor graph to validate many aspects of your code, including variables with different numbers of states, and factors of* *varying orders. Clearly describe the experimental evidence you use to verify that your implementations are correct.*

We now apply our sum-product inference code to the *ALARM (A Logical Alarm Reduction Mechanism)* network, an early example of an expert system designed for hospital patient monitoring. Figure 1 illustrates this network as a directed graphical model. To allow your sum-product implementation to be easily applied to this model, we have provided a Matlab script which constructs a factor graph representation of the ALARM network, in which one factor node links each variable to its parents.

In some of the questions below, we consider smaller networks containing a subset of the nodes in the full ALARM network. You will need to construct the factor graphs for these sub-models yourself, either by modifying the provided script which builds the full network, or by editing the already-loaded factor graph data structure. Remember that in a directed graphical model, an unobserved variable with no children can be exactly marginalized by simply removing its associated factor and variable nodes from the graph. Also, when some of the variables adjacent to a factor node are observed, this induces a lower-order factor node whose potential table is a "slice" of the full potential table.

All variables in the ALARM network are discrete, and take 2 to 4 states indexed by positive integers. For some questions below, we ask you to report means of certain marginal distributions. Means are sensible for the discrete variables in this model, because all variables are defined on an ordinal scale in which larger values correspond to greater degrees of presence or severity of a measurement or diagnosis.

Question 2:

For all questions below, run the sum-product algorithm until the maximum change in message values between iterations is 10^{-6}, where one iteration updates all messages once. If sum-product fails to converge after 500 iterations, report this and discuss possible explanations.

a) *Consider a sub-network containing four "cause" variables (PULMEMBOLUS, INTUBATION, VENTTUBE, KINKEDTUBE) and four "effect" variables (PAP, SHUNT, PRESS, VENTLUNG). To construct this graph, condition on observing VENTMACH=4, DISCONNECT=2, and (exactly) marginalize all other descendants. Use sum-product to compute the means of the four cause variables, and compare your answer to that produced by your enumeration-based inference code.*

b) *Take the network from part (a), and also condition on observing SHUNT=2, PRESS=4. Use sum-product to compute the means of the four cause variables, and compare your answer to that produced by your enumeration-based inference code.*

c) *Now consider a different sub-network containing the nodes INTUBATION, VENTTUBE, KINKEDTUBE, and VENTLUNG. To construct this graph, condition on observing VENTMACH=4, DISCONNECT=2, PRESS=4, MINVOL=2, and (exactly) marginalize all other descendants. Use sum-product to compute the means of the four unobserved nodes, and compare your answer to that produced by your enumeration-based inference code.*

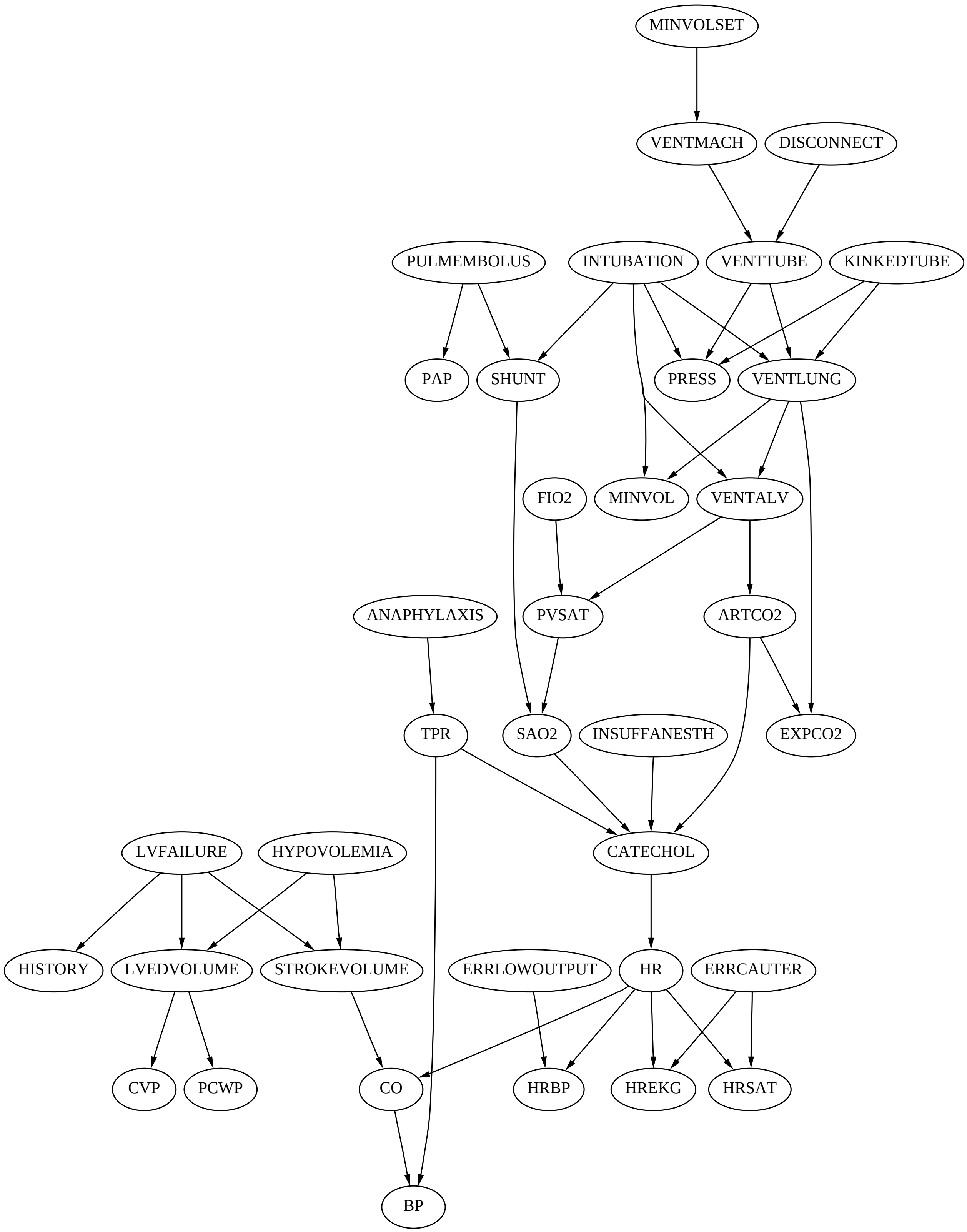

Figure 1: The ALARM network represented as a directed graphical model.

d) *Discuss the results in parts (a-c), explaining cases in which sum-product produces exact marginals, and cases in which they only approximate the true marginals.*

e) *We now consider a much larger sub-network. To construct it, condition HYPOVOLEMIA, HR, INTUBATION, KINKEDTUBE, and VENTALV to take their smallest possible values. Use the sum-product algorithm to compute the means of the diagnostic variables LVFAILURE, ANAPHYLAXIS, INSUFFANESTH, PULMEMBOLUS, and DISCONNECT. Do you expect sum-product to compute exact marginal distributions for this graph? Why or why not?*

f) *Finally, we consider the full ALARM network with no observations. Using the sum-product algorithm, compute the means of the following 8 diagnostic variables: LVFAILURE, HYPOVOLEMIA, ANAPHYLAXIS, INSUFFANESTH, PLUMEMBOLUS, INTUBATION, DISCONNECT, KINKEDTUBE. Are these estimates equal to the true means?*

g) *Consider the full ALARM network, and condition the following observation variables to their smallest possible values: HISTORY, CVP, PCWP, BP, HRBP, HREKG, HRSAT, EXPCO2, MINVOL. Use the sum-product algorithm to compute the means of the 8 diagnostic variables from part (f).*

h) *Consider the network and observations from part (g), but change HRBP, HREKG, and HRSAT to take their largest possible values. Again use the sum-product algorithm to compute the means of the 8 diagnostic variables, and discuss the most significant changes.*

For a graph with cycles like the ALARM network, the sum-product algorithm may not always compute the correct marginal distributions. In the following question we explore the computational complexity of the elimination algorithm, which is guaranteed to find exact marginals with additional computational expense.

Question 3:

Consider the undirected graphical model in Figure 2. This 3×3 grid is a small example of the spatial lattices which are often used in image processing and computer vision.

a) *Consider the elimination algorithm from Chap. 3 of Jordan's "An Introduction to Probabilistic Graphical Models". Suppose we compute the marginal of node 1 via the following elimination ordering: $\{5, 2, 6, 8, 4, 9, 7, 3, 1\}$. Sketch the sequence of graphs formed. What is the largest clique that is produced?*

b) *Suppose we instead use the following elimination ordering: $\{9, 7, 3, 6, 8, 5, 2, 4, 1\}$. Sketch the sequence of graphs formed. What is the largest clique that is produced? Which ordering is more computationally efficient?*

c) *Using intuition from these examples, give a reasonable ($\ll n^2$) upper bound on the treewidth of an $n \times n$ grid. Explain your answer.*

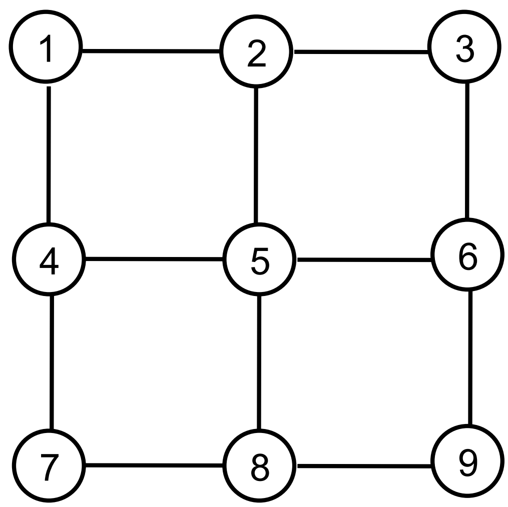

Figure 2: An undirected graphical model of two-dimensional spatial dependencies among nine variables arranged in a 3×3 grid.