Cycle-corrected Belief Propagation

Yaroslav Bulatov BULATOVY@MYSTRANDS.COM
MyStrands, 760 SW Madison, STE 106, Corvallis, OR 97333, USA

Keywords: belief propagation, self-avoiding walk tree

Abstract

We provide a way to modify loopy belief propagation to correct for the error introduced by cycles up to length k. The method is based on bottom-up dynamic programming reformulation of the Self-Avoiding Walk Tree representation of marginal. k=2 recovers loopy belief propagation, "k=size of the largest cycle in the graph" provides exact inference at higher cost.

1. Introduction

Sokal and Scott (2005) and later Weitz (2006) demonstrated a way to represent occupation probability of a node in a general hard-core gas model as the occupation probability of a root node in a tree structured hard-core model. Jung and Shah (2007) extended this to general binary graphical models, Nair and Tetali extended this approach to general n-state graphical models (2007). Suppose we wanted to compute all marginals using self-avoiding walk tree representation. Straightforward approach of constructing a self-avoiding walk tree rooted at each node will have redundant computation. To get rid of the redundancy we can identify shared pieces of computation, and compute all the marginals in a bottom up fashion. The result is a procedure similar to loopy belief propagation, except that instead of a message for every edge, we have a message for every self-avoiding walk on the graph. We can trade off between efficiency and accuracy by substituting messages corresponding to shorter walks into the update equations. For example, if the update equation calls for message on the walk (1,2,3), we can replace it with the message corresponding to the walk (2,3).

Preliminary work. Under review by the International Workshop on Mining and Learning with Graphs (MLG). Do not distribute.

1.1. Example: Binary Ising model

Consider the following simple structure

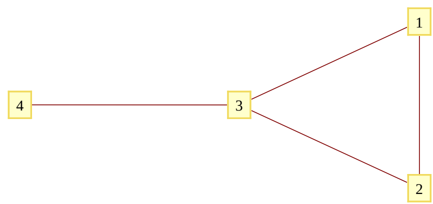

Figure 1. Log(base 10) of error as a function of largest cycle length corrected

Suppose the states are denoted by variables y_1, y_2, y_3, y_4 which take on values 1 and -1 and the probability of a particular configuration is as follows

$$P(y_1, y_2, y_3, y_4) \propto \exp(y_1 y_2 + y_2 y_3 + y_1 y_3 + y_3 y_4 + y_1 x_1 + y_2 x_2 + y_3 x_3 + y_4 x_4) \quad (1)$$

Using the self-avoiding walk tree representation we get the following expressions for the 1/2 log-odds of Y_3 (Eq.2) and Y_4 (Eq.3)

$$f(x_4) + \underbrace{x_3 + f(x_2 + f(x_1 + 1)) + f(x_1 + f(x_2 - 1))}_{c} \quad (2)$$

$$x_4 + f(\underbrace{x_3 + f(x_2 + f(x_1 + 1)) + f(x_1 + f(x_2 - 1))}_{c}) \quad (3)$$

Where $f(x) = \text{arctanh}(\tanh 1 \tanh x)$

You can see that the part marked c is the same in both equations. This part can be computed once and reused. In example above, c has the following interpretation – it is the "log-odds of the root node of a tree rooted at 3 which does not contain node 4". More generally, consider the following binary Ising spin-glass model

$$P(y_1, \ldots, y_n) \propto \exp\Big(\sum_{(ij)\in G} J_{ij} y_i y_j + \sum_{i=1}^{n} x_i y_i\Big) \quad (4)$$

In order to compute marginal probabilities (also known as mean magnetizations) for every node in the model efficiently, we can introduce a message for every self-avoiding walk in the model, and relate them by the following update equation

$$\begin{aligned} m_{p_1,\ldots,p_{l-1},p_l} &= f_{p_{l-1},p_l}\Big(\sum_{q\in\sigma} m_{p_1,\ldots,p_l,q} \\ &\quad x_{p_l} - \sum_{q\in C^+} J_{p_l,q} + \sum_{q\in C^-} J_{p_l,q}\Big) \quad (5) \end{aligned}$$

Here $f_{ij}(x) = \operatorname{arctanh}(\tanh J_{ij} \tanh x)$, σ denotes all self-avoiding 1-step continuations of a self-avoiding walk starting with nodes $p_1, \ldots, p_l$, C^+ indicates all non-backtracking continuations of a self-avoiding walk which create a loop with current node (p_l) being larger than the first node in the loop (ie, 1,2,3,1 would be one such loop). C^- is same as before, but for loops with current node being smaller than the first node in the loop (ie, 1,3,2,1 would be one such loop).

Note that if the length of the longest cycle in the model is k, then messages corresponding to self-avoiding walks longer than k are redundant. More specifically,

$$m_{p_1,\ldots,p_k,p_{k+1}} = m_{p_2,\ldots,p_{k+1}} \quad (6)$$

To get the non-redundant set of equations, simply replace every redundant message in the equation 5 with it's shortened version, lets call this procedure "truncation".

Using this procedure, here's the full set of update equations for the model in (1).

$$\begin{aligned} m_1 &= x_1 + m_{12} + m_{13} \\ m_2 &= x_2 + m_{21} + m_{23} \\ m_3 &= x_3 + m_{31} + m_{32} + m_{34} \\ m_4 &= x_4 + m_{43} \\ m_{13} &= f(x_3 + m_{34} + m_{132}) \\ m_{12} &= f(x_2 + m_{123}) \\ m_{21} &= f(x_1 + m_{213}) \\ m_{23} &= f(x_3 + m_{34} + m_{231}) \\ m_{31} &= f(x_1 + m_{312}) \\ m_{32} &= f(x_2 + m_{321}) \\ m_{34} &= f(x_4) \\ m_{43} &= f(x_3 + m_{31} + m_{32}) \\ m_{123} &= f(x3 - 1 + m_{34}) \\ m_{132} &= f(1 + x_2) \\ m_{213} &= f(-1 + x_3 + m_{34}) \\ m_{231} &= f(1 + x_1) \\ m_{312} &= f(-1 + x_2) \\ m_{321} &= f(1 + x_1) \end{aligned}$$

Value of log-odds at of y_1 is $2m_1$. Note that if we truncate each message to it's last 2 vertices, resulting equation will be equivalent to loopy belief propagation. This suggests a natural sequence of approximations – truncate each message to the last p nodes. When p is equal to k, result will be exact. When p is 2, the resulting procedure is equivalent to loopy belief propagation, which gives result equivalent to Bethe-Peierls approximation when it converges.

1.2. Experiment

Consider the graphical model in the figure 2

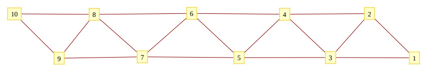

Figure 2. Skip-chain of length 5

Suppose it represents an Ising model with interaction strength +1 for adjacent nodes, and +1 potential for each node. We can construct a sequence of approximations by taking a full set of update equations according to (5) and then truncating messages in those equations as in (6) to various lengths. We then iterate resulting equations 150 times, Figure (4) shows logarithm (base 10) of squared error over all marginals (marginal log-odds compared to approximate log-odds) as a function of the truncation length (ie, largest cycle length corrected).

If we use the same model as before, but with anti-ferromagnetic interactions, the errors may cancel out, and increasing order of approximation can actually slightly increase error as seen below

If we go one step further, and correct for all cycles up to length 10, the result will be identical to exact marginalization.

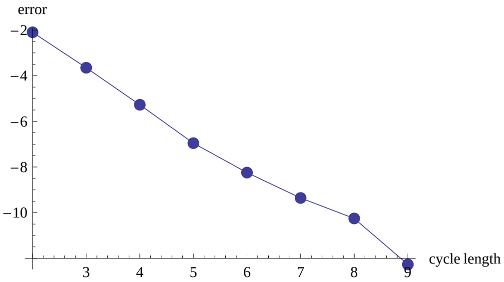

Figure 3. Log(base 10) of error as a function of largest cycle length corrected

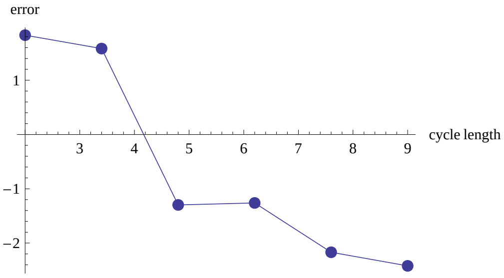

Figure 4. Log(base 10) of error as a function of largest cycle length corrected for anti-ferromagnetic potentials

2. Multi-variate models

Similar approach can be applied to multi-variate graphical models. Nair/Tetali (2007) have showed how to extend the self-avoiding walk tree approach to handle more than two states. Their construction, called correlation decay tree, involves fixing some loop closing nodes in the self-avoiding walk tree to take on the same value as the opening node. Here again we can save shared pieces of computation. For a model with s states we get the following set of update equations in 1/2 log-odds parameterization

$$M_{p_1,\dots,p_n}(v_1,\dots,v_n) = \sum_{\sigma} m_{p_1,\dots,p_n,q}(v_1,\dots,v_n) + \sum_{i\in C^+} \log \frac{\psi^{p_n,p_i}_{v_n,v_i}}{\psi^{p_n,p_i}_{s,v_i}} + \sum_{i\in C^-} \log \frac{\psi^{p_n,p_i}_{v_n,s}}{\psi^{p_n,p_i}_{s,s}} \qquad (7)$$

$$m_{p_1,\dots,p_n}(v_1,\dots,v_{n-1}) = \frac{1}{2}\log\left(\frac{\sum_j^s \psi^{p_{n-1}p_n}_{jv_{n-1}} \exp(2M_{p_1\dots p_n}(v_1\dots j))}{\sum_j^s \psi^{p_{n-1}p_n}_{js} \exp(2M_{p_1\dots p_n}(v_1\dots j))}\right) \qquad (8)$$

C^+,C^- and σ here have the same interpretation as in (5). $\psi^{a,b}$ indicates the potential matrix for edge from a to b. For a matrix A, A_{ab} indicates entry in position (a,b). i indexes the state, since we are dealing with log-odds, $i \in (1\dots s-1)$. Variables $v_1 \dots v_n$ index states of previously visited nodes. Again, because of degree of freedom removed by log-odds parameterization, they take on values $(1\dots s-1)$. Note, if we have two states, then v and i variables in (7,8) can only assume one value, so we can drop them. Using a potential matrix corresponding to the transfer matrix for the simple Ising spin glass as in (1) we will recover equations in (5). For example, for the model in (1), the potential matrix corresponding to edge connecting variables y_1 and y_2 is

$$\begin{pmatrix} e^{1+x_1/2+x_2/2} & e^{-1+x_1/2-x_2/2} \\ e^{-1-x_1/2+x_2/2} & e^{1-x_1/2-x_2/2} \end{pmatrix} \qquad (9)$$

Log-odds of state i for node k is double the value of $M_k(i)$ where M is defined using equations (7,8).

M messages are temporary and they can be substituted into (8), so to analyze the complexity it is sufficient to look at m messages. Note that if we only allow messages for self-avoiding walks of length 2 (loopy belief propagation), then we will have a message for every edge, and each message will have $s-1$ entries. If we allow self-avoiding walks of length 3, then we get matrix-valued messages, each with $(s-1)^2$ entries. More generally, correcting for loops up to length k can introduce up to d^k messages, each having $(s-1)^k$ entries, where d is maximum degree of the graph. However, we can do better with graphs with few loop interactions. Note that in equations (7,8), the only time the value of previous node (v_i) is being used is during convolution step, where the most recent node value is needed. Also when the walk corresponding to current message can complete a loop, this calls for a value of a node visited earlier in the walk. If only some nodes can be a part of the loop for given walk, we can reduce size of messages. For instance, consider message corresponding to the walk $1,2,3,4$. Suppose that the only way that the walk starting with $1,2,3,4$ can terminate with a loop is by traversing nodes in following order $1,2,3,4,2$. This means that our message only needs to keep track of the value of nodes 2 and 4, so we can use a matrix-valued message instead of 4-dimensional tensor valued one. For a graph G with n nodes, consider the following measure

cc(G) is maximum k such that there are k cycles $C_1,\dots,C_k \in G, k = 1,\dots,n$, such that $C_1 \cap \dots \cap C_k$ contains a path of length k.

The size of the largest message needed to provide loop corrections for a gaph G will be $(s-1)^{(cc(G)+1)}$. Note, that if the family of graphs has bounded tree-width, then this measure is also bounded (Kawarabayashi, 2008)

2.1. Code

Code used to obtain the graph in the experiment section and implementation of update equations (7,8) for multi-variate models is available at
`yaroslavvb.com/research/reports/mlg08`

References

Jung, K., & Shah, D. (2007). Inference in binary pairwise markov random fields through self-avoiding walks.

Kawarabayashi, K. (2008). personal communication.

Nair, C., & Tetali, P. (2007). The correlation decay (cd) tree and strong spatial mixing in multi-spin systems.

Scott, A., & Sokal, A. (2005). The repulsive lattice gas, the independent-set polynomial, and the lovasz local lemma. *Journal of Statistical Physics*, *118*, 1151–1261.

Weitz, D. (2006). Counting independent sets up to the tree threshold. *STOC '06: Proceedings of the thirty-eighth annual ACM symposium on Theory of computing* (pp. 140–149). New York, NY, USA: ACM.