Message-Passing Algorithm for Marginal-MAP Estimation *

Jiarong Jiang, Piyush Rai, Hal Daumé III
{*jiang,piyush,hal*} *@cs.utah.edu*
School of Computing, University of Utah

1 Introduction

We consider an inference setting for probabilistic graphical models where the graph consists of two type of nodes. On one (the *max* nodes), we want to compute the MAP estimates; on the other (the *sum* nodes) we want to compute the marginals. More formally, let $p_\theta(x, z) = \exp\left[\langle\theta, F(x, z)\rangle - A(\theta)\right]$ be the exponential family probability distribution defined by a graphical model, where $F(x, z)$ is the enumeration of all nodes (x represents max nodes and z sum nodes) and all edges between sum nodes, max nodes and connecting sum and max nodes. The log partition function $A(\theta) = \log \sum_{x,z} \exp[\langle\theta, F(x, z)\rangle]$. In our setting, we only care about the *maximum a posteriori* (MAP) estimates for x nodes, but sometimes the marginals on z are also probably useful. The former problem can be written formally as the following maximization problem: $\arg\max_x \sum_z p_\theta(x, z)$. A typical example is a latent variable model where the xs are variables we really care about, and the zs are often nuisance or latent variables.

In such settings, the Expectation-Maximization algorithm [1] can be an obvious choice if the *marginal* posterior in the E-step can be computed in closed-form. If not, then the E-step can use Monte-Carlo simulation. However, EM can still be prone to getting stuck in a local maxima. Alternatively, [2] proposed an MCMC-based algorithm for *direct* maximization of marginal posterior distributions by introducing an artificially augmented probability model, whose sampling gives marginal-MAP estimates of the variables of interest. However, this approach also suffers from the local maxima problem. To deal with this issue, [3] proposed a Sequential Monte Carlo based approach (and similar to simulated annealing) which is much less sensitive to initialization than EM and MCMC algorithms. In our work, we take a different approach and show how message-passing algorithms for graphical models can be used to obtain marginal-MAP estimates in a variational framework [4]. We also show connections of our algorithm to the generalized EM algorithm.

2 Marginal-MAP Estimation using Message-Passing

Sum-product and max-product are the basic algorithms for computing marginals and MAP estimates respectively, in probabilistic graphical models. For our setting where we want marginals for one set of nodes and MAP estimates for the other set, one can still run sum-product or max-product algorithms over the graph, and choose the assignments according to the maximum of sum or max marginal values for each random variable. However, using max-product only will inevitably ignore the effect of sum-nodes z, whereas using sum-product only will ignore the effect of max nodes x.

In this work, we present a hybrid algorithm (which is based on using a mix of sum and max messages) where the outgoing message of a node is decided by the type of the node. The updates are as follows:

- Message from *sum* node t: $M_{ts}(x_s) \leftarrow \kappa \sum_{x'_t \in \chi_t} \{\exp[\theta_{st}(x_s, x'_t) + \theta_t(x'_t)] \prod_{v \in N(t) \backslash s} M_{vt}(x_t)\}$

- Message from *max* node t: $M_{ts}(x_s) \leftarrow \kappa \max_{x'_t \in \chi_t} \{\exp[\theta_{st}(x_s, x'_t) + \theta_t(x'_t)] \prod_{v \in N(t) \backslash s} M_{vt}(x_t)\}$

---

*Topic: graphical models   Preference: oral

When the messages converge, the node marginals are obtained from $p(x_s; \theta) = \kappa \exp\{\theta_t(x_t)\} \prod_{t \in N(s)} M_{ts}(x_s)$ and κ is a normalization constant. For those max nodes, the assignment of node x_s is given by $\arg\max_{x_i \in \mathcal{X}_s} p(x_s(x_i))$.

3 Generalized EM based Interpretation

It turns out that the hybrid (sum-product/max-product) message-passing algorithm we propose turns out to be an instance of the generalized EM algorithm. To see this, consider the conditional: $p_\theta(x \mid z) = \frac{\exp[\langle \theta, F(x,z) \rangle - A(\theta)]}{\sum_x \exp[\langle \theta, F(x,z) \rangle - A(\theta)]} = \exp[\langle \theta, F(x,z) \rangle - B_z(\theta)]$, where $B_z(\theta) = \sum_x \exp[\langle \theta, F(x,z) \rangle]$. The same yields $p_\theta(z \mid x) = \exp[\langle \theta, F(x,z) \rangle - C_x(\theta)]$, where $C_x(\theta) = \sum_z \exp[\langle \theta, F(x,z) \rangle]$. Now, (generalized) EM works by maximizing: $\mathbb{E}_{z \sim p_\theta(z \mid x)} \log p_\theta(x \mid z)$. By plugging in the expression for $\log p_\theta(x \mid z)$ and dropping irrelevent terms, this leads to:

$$\sum_{x,i} \left[\theta_{x,i} + \sum_{z,j} \mu_z(j)\theta_{xz,ij} \right] x(i) + \sum_{x_1x_2,ij} \theta_{x_1x_2,ij} x_1(i) x_2(j) + Const. \tag{1}$$

Regarding the "E" step, we need to compute these marginals. Following from above, we want the marginals of $p_\theta(z \mid x)$. This can be done by just fixing the x values at their MAP solutions and running sum-product on the zs with no additional modifications (that is, the "messages" coming out from x nodes are just 100% confident messages at their MAP value). For "M" step, it is nothing but running max product with potentials on x nodes modified according to equation (1). It is possible to only run one single message pass for each step and this is equivalent to the hybrid message passing algorithm.

4 Experiments

We compare our hybrid messsage passing algorithm against plain sum-product and plain max-product based MAP estimate on synthetic data. For our preliminary experiments, our synthetic data is a chain of 10 nodes (The results of lattice with 10 nodes are similar).

Fig 1 shows the loss on the assignment of max nodes. As we can see, with the increasing percentage of sum nodes, the accuracy of max-product decreases and sum-product increases, However, our algorithm always results in the smallest loss of the three. Meanwhile, the mean of discrepancy (not shown here) for the three algorithms on marginals compared to the true marginals of $p(z|x)$ when there are 20% sum nodes is 0.4849 for max product, 0.4399 for sum product, 0.4713 for our algorithm. When the percentage of sum nodes increases to 80%, the corresponding means are 1.0693, 0.6501 and 0.6776. The performance of our algorithm on sum nodes beats max product and is close to that of sum product. Moreover, the iterations of our algorithm to converge is no more than the maximum of that of sum and max product and is much less than that of EM.
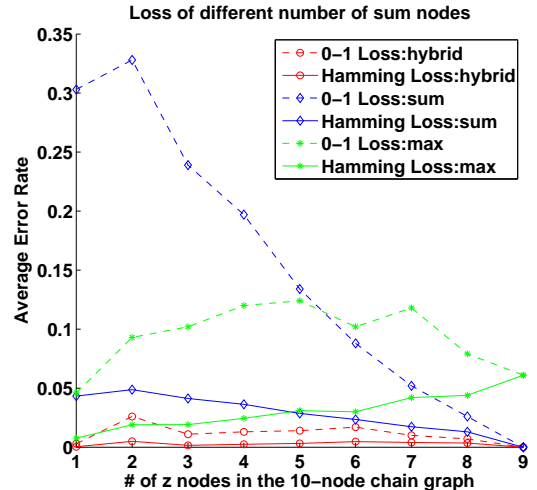

Figure 1: Comparison of various algorithms for MAP estimates

References

[1] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of The Royal Statistica Society*, 1977.

[2] A. Doucet, S. J. Godsill, and C. P. Robert. Marginal Maximum a Posteriori Estimation using Markov Chain Monte Carlo. *Statistics and Computing*, 2002.

[3] A. M. Johansen, A. Doucet, and M. Davy. Particle Methods for Maximum Likelihood Estimation in Latent Variable Models. *Statistics and Computing*, 2008.

[4] M. J. Wainwright and M. I. Jordan. Graphical Models, Exponential Families, and Variational Inference. *Foundations and Trends in Machine Learning*, 1:1–305, 2008.