

Variational Methods for Graphical Models*

Patrick Pletscher†

May 14, 2006

In the past 10 years variational methods have gained quite a bit of popularity as an approximate method for probabilistic inference in the context of graphical models. In this summary of [JGJS99] we mainly look at their theoretical foundations and discuss shortly two applications: the QMR-DT database and the Factorial Hidden Markov Model.

Keywords: Variational methods, Bayesian networks, graphical models, probabilistic inference.

1 Introduction

Given a graphical model, as e.g. the one shown in Figure 1, we would like to compute the conditional probability distribution over the values of the “hidden” nodes H , given the values of the “evidence” nodes E :

$$P(H|E) = \frac{P(H, E)}{P(E)}$$

This particular problem is known as (probabilistic) *inference*. However as for most interesting problems in Computer Science, it is known to be a \mathcal{NP} -hard

*a summary of [JGJS99]

†pat@student.ethz.ch

problem [Coo87]. Thus, as in other areas, the research focus has shifted from finding exact algorithms, towards finding good approximation schemes. Nevertheless there are important special instances of graphical models, e.g. trees, where exact algorithms are efficient. And, as we will see, even in the framework of variational methods, we want to use exact algorithms.

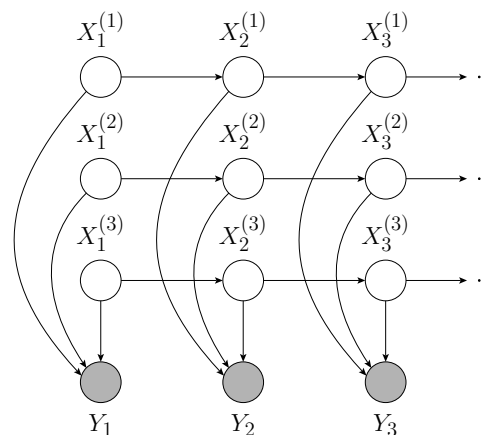


Figure 1: A factorial HMM with three chains. The observed nodes are colored gray.

A popular exact method, we shortly want to discuss, is the *junction tree* algorithm. It works in 3 steps:

1. moralization (marry parents of all nodes and drop edge orientation).

2. triangulation (add new edges, such that all 4-cycles have a chord).
3. Build a junction tree (arrange cliques of the graph in a tree, such that if a node appears in any two cliques, it appears in all cliques that lie on the path between the two cliques).

The time complexity of the junction tree algorithm is exponential in the size of the maximum clique.

For the graphical model, shown in Figure 1, the junction tree algorithm has running time $\mathcal{O}(N^{M+1}T)$, where N denotes the number of states in each chain, M is the number of chains (in the example $M = 3$) and finally T is the length of the time series. Thus we get a running time exponential in the number of chains. This should illustrate that for certain graphical models approximate methods are needed.

2 Variational Methods

The key idea behind variational methods is to transform the graphical model into a simplified one, in which inference is efficient. Thereby making use of the numerical representation of the joint probability, i.e. there might exist nodes that are “nearly” independent. Exact algorithms are unable to explore such facts.

The general principle of *convex duality* provides us with a scheme how to translate a concave function (or in our case a conditional probability distribution) into an optimization problem. This goes as follows:

$$f(\mathbf{x}) = \min_{\boldsymbol{\lambda}} \{\boldsymbol{\lambda}^T \mathbf{x} - f^*(\boldsymbol{\lambda})\} \quad (1)$$

where $f^*(\mathbf{x})$ can be obtained from the following dual expression:

$$f^*(\boldsymbol{\lambda}) = \min_{\mathbf{x}} \{\boldsymbol{\lambda}^T \mathbf{x} - f(\mathbf{x})\} \quad (2)$$

Equation (1) tells us that for a given \mathbf{x} we can find a linear function that is exactly the function value $f(\mathbf{x})$ at \mathbf{x} and “above” or equal to $f(\mathbf{x})$ everywhere else. Equation (2) can be simply understood as finding the “right” intercept, for a given $\boldsymbol{\lambda}$, which boils down to finding the minimal vertical deviation between $\boldsymbol{\lambda}^T \mathbf{x}$ (notice that this function goes through the origin) and $f(\mathbf{x})$, a graphical illustration is given in Figure 2. In physics this kind of transformation is known as a Legendre transformation, which was pointed out by Cyril Stark, a colleague of mine (whom I would like to thank for the fruitful discussion).

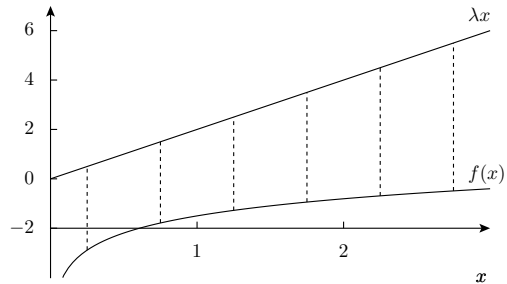


Figure 2: Minimizing across the deviations between λx and $f(x)$.

Having introduced this variational framework (variational in the sense, that we have introduced new parameters $\boldsymbol{\lambda}$, which have to be optimized), we can now apply it to graphical models. Given conditional probabilities $P(S_i|S_{\pi(i)})$ and their variational representations $P^U(S_i|S_{\pi(i)}, \boldsymbol{\lambda}_i^U)$, which provide an upper bound, we can write the joint probability

S as:

$$P(S) = \prod_i P(S_i | S_{\pi(i)}) \\ \leq \prod_i P^U(S_i | S_{\pi(i)}, \lambda_i^U)$$

The same holds true for marginalization, where E and H are a distinct partition of S :

$$P(E) = \sum_{\{H\}} P(H, E) \\ \leq \sum_{\{H\}} \prod_i P^U(S_i | S_{\pi(i)}, \lambda_i^U)$$

We choose the variational forms, such that the summation over H can be carried out efficiently. Notice that we end up with the approximate, variational distribution of interest, which we still have to minimize with respect to the variational parameters λ . One can derive similar rules for lower bounds, which are especially important for the computation of $P(H|E) = P(H, E)/P(E)$.

This method can now be applied in (at least) two ways to graphical models: either the *sequential* approach, where the algorithm determines on its own which nodes to transform, or the *block* approach, where the author of the graphical model selects some “difficult” sub-graph together with a graph, which should replace the selected graph. In both cases we then use an exact inference algorithm on the simplified graph.

Here we only discuss the block approach in slightly more detail: We wish to approximate the conditional probability $P(H|E)$. To do so, we introduce an approximating family $Q(H|E, \lambda)$, where λ are variational parameters. We now select the variational parameters, such

that the Kullback-Leibler (KL) divergence, $D(Q||P)$ is minimized

$$\lambda^* = \arg \min_{\lambda} D(Q(H|E, \lambda) || P(H|E)),$$

where for any probability distributions $Q(S)$ and $P(S)$ the KL divergence is defined as follows:

$$D(Q||P) := \sum_{\{S\}} Q(S) \ln \frac{Q(S)}{P(S)}.$$

In the area of graphical models the distribution P is often only given up to a parameter vector θ (e.g. the transition probability in a Hidden Markov Model), and we then usually not only need to solve the inference problem, but as well find an estimation for θ . The minimization of the KL-divergence (expectation step) is then often interleaved with a maximization step for θ , which is nothing but a special instance of an EM algorithm [DLR77].

3 Two Applications

3.1 QMR-DT database

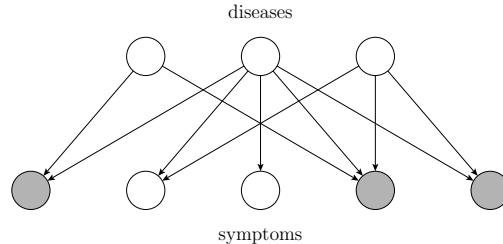


Figure 3: The QMR-DT graphical model.

If we apply variational methods to a node of a QMR-DT graphical model, we see that it accounts to a decoupling of the node from the model.

3.2 Factorial Hidden Markov Model

For the FHMM, as in Figure 1, we notice that, to make inference tractable, it is sufficient to decouple the chains, as the inference for the chains is efficient. Thus we select as an approximating distribution the FHMM where all the “vertical” edges are dropped, see Figure 4. This might seem foolish, as the “output” Y_i would then be completely independent of the chain’s state. However the variational parameters become interdependent when we optimize the KL-divergence. This can in some way be seen as the dependence introduced by the moralization step in exact algorithms.

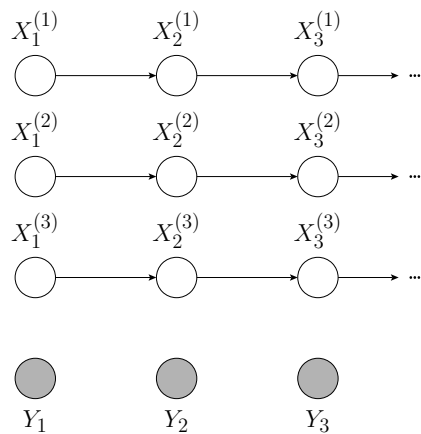


Figure 4: Variational approximation of the FHMM.

However, as far as I can see, this new dependence is not fully understood yet (or at least in the paper).

4 Discussion

The variational framework has shown to be quite successful and has been applied to many diverse graphical models, for example in text understanding (Latent Dirichlet Allocation [BNJ03]), analysis of medical data (QMR-DT) or Hidden

Markov Models, which show up in many areas, e.g. bio-informatics or theoretical Computer Science.

However there’s no ultimate recipe yet, how to apply the variational methods out-of-the-box to a given graphical model (except the sequential approach, which has other deficiencies). Often it is best, if the “designer” knows which nodes make exact methods intractable and who then decides on a tractable probability distribution for these nodes (block approach).

References

- [BNJ03] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.
- [Coo87] G. F. Cooper. Probabilistic inference using belief networks is \mathcal{NP} -hard. (KSL-87-27), 1987.
- [DLR77] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39:1–38, 1977.
- [JGJS99] Michael I. Jordan, Zoubin Ghahramani, Tommi Jaakkola, and Lawrence K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233, 1999.