Loopy Propagation

1 Tree Structured MRFs

Consider an acyclic graph (a tree) whose nodes are random variables and where there is function Γ_X defined on the paossible values of X for each node X and for each pair X, Y, connected by an edge there is function $\Gamma_{X,Y}$ defined on a pair of a value for X anf a value for Y. We let $N(X)$ be the set of nodes Y such that the tree contains a branch from X to Y. This gives a Markov Random Field where for any variable assignment ρ we have the following probability of ρ.

$$Z(\rho) = (\Pi_X \ \Gamma_X(\rho(X))) \ \left(\Pi_{X,Y:\ X \in N(Y)} \Gamma_{X,Y}(\rho(X), \rho(Y))\right)$$

$$P(\rho) = \frac{Z(\rho)}{\sum_\rho Z(\rho)}$$

For such trees there is a particularly simple algorithm for computing the probability of values of variables. We can select a node R as the root of the tree. Relative to this root each node has a well defined parent (nearer the root) and well defined children (farther from the root). Of course different choices of the root change which neighbor is the parent. For $Y \in N(X)$ we write $T_Y(X)$ to be the subtree rooted at X in the case where Y is taken to be the parent of X. Equivalently, we have $W \in T_Y(X)$ if the path from W to X does not go through Y. We always have $X \in T_Y(X)$. We define $Z_{X,Y}(x)$ as follows.

$$Z_{X,Y}(x) = \sum_{\rho:\ \mathrm{dom}(\rho) = T_Y(X),\ \rho(X) = x} \begin{array}{l} \Pi_{W \in T_Y(X)} \ \Gamma_X(\rho(X)) \\ \Pi_{W,W' \in T_Y(X), W' \in N(W)} \ \Gamma_{W,W'}(\rho(W), \rho(W')) \end{array}$$

These quantities can be computed efficiently as follows.

$$Z_{X,Y}(x) = \Gamma_X(x) \Pi_{W \in N(X):\ W \neq Y} \sum_w \Gamma X, W(x,w) Z_{W,X}(w)$$

In the base case we have that X is a leaf node and there is no W in $N(X)$ other than the parent Y. For the root variable R we define $Z_R(r)$ as follows.

$$Z_R(r) = \sum_{\rho:\ \rho(R) = r} Z(\rho)$$

This can be computed efficiently as follows.

$$Z_R(r) = \Gamma_R(r) \Pi_{X \in N(R)} \sum_x \Gamma_{X,R}(x,r) Z_{X,R}(x)$$

We then have the following where $P_R(r)$ is the probability that node R is assigned value r.

$$P_R(r) = \frac{Z_R(r)}{\sum_r Z_R(r)}$$

In these equations the renormalization is done at the end. But the renormalization can be done as we go. We define $P_{X,Y}(x)$ as follows.

$$P_{X,Y}(x) = \frac{Z_{X,Y}(x)}{\sum_x Z_{X,Y}(x)}$$

The variables $P_{X,Y}(x)$ can be computed directly as follows.

$$Q_{X,Y}(x) = \Gamma_X(x)\Pi_{W \in N(X):\, W \neq Y} \sum_w \Gamma_{X,W}(x,w) P_{W,X}(w) \quad (1)$$

$$P_{X,Y}(x) = \frac{Q_{X,Y}(x)}{\sum_x Q_{X,Y}(x)} \quad (2)$$

Equation (1) is a definition. Equation (2) can be proved as follows.

$$\begin{aligned}
Q_{X,Y}(x) &= \Gamma_X(x)\Pi_{W \in N(X):\, W \neq Y} \sum_w \Gamma_{X,W}(x,w) P_{W,X}(w) \\
&= \Gamma_X(x)\Pi_{W \in N(X):\, W \neq Y} \sum_w \Gamma_{X,W}(x,w) \frac{Z_{W,X}(w)}{\sum_{w'} Z_{W,X}(w')} \\
&= \left(\frac{1}{\Pi_{W \in N(X):\, W \neq Y} \sum_{w'} Z_{W,X}(w')}\right) Z_{X,Y}(x) \\
&= C Z_{X,Y}(x) \\
\frac{Q_{X,Y}(x)}{\sum_x Q_{X,Y}(x)} &= \frac{Z_{X,Y}(x)}{\sum_x Z_{X,Y}(x)} = P_{X,Y}(x)
\end{aligned}$$

Similarly we can then compute $P_R(r)$ as follows.

$$Q_R(r) = \Gamma_R(r)\Pi_{X \in N(R)} \sum_x \Gamma_{R,X}(r,x) P_{X,R}(x) \quad (3)$$

$$P_R(r) = \frac{Q_R(r)}{\sum_r Q_R(r)} \quad (4)$$

2 Loopy MRFs

Here we are interested in using equations (1) and (2) on loopy graphs, i.e., graphs that are not trees. We initialize the values $P_{X,Y}(x)$ randomly and then recompute these values again and a gain using equations (1) and (2). There is no selected root and we compute all values of the form $P_{X,Y}(x)$. In practice this often converges and gives approximately correct values of $P_X(x)$ at all nodes X as computed by (3) and (4). This is called loopy belief propagation, or simply loopy propogation. The efficiency of the iteration can be improved by using the following equations.

$$M_{X,W}(x) = \sum_w \Gamma_{X,W}(x,w) P_{W,X}(w) \tag{5}$$

$$Q_X(x) = \Gamma_X(x) \Pi_{W \in N(X)} M_{X,W}(x) \tag{6}$$

$$Q_{X,Y}(x) = \frac{Q_X(x)}{M_{X,Y}(x)} \tag{7}$$

$$Q_{X,Y} = \sum_x Q_{X,Y}(x) \tag{8}$$

$$P_{X,Y}(x) = \frac{Q_{X,Y}(x)}{Q_{X,Y}} \tag{9}$$

The run time for a single iteration is dominated by (5) which takes time proportional to the number of four-tuples $\langle X, x, W, w \rangle$ with $W \in N(X)$.