Factor Graphs and Message Passing

Before we derive some useful results on efficient and distributed algorithms for computing the normalization constant in message passing, we would like to introduce the concept of factor graphs and message passing. A factor graph is a bipartite graph that represents the factorization of a function of n variables. In a factor graph, we have two types of nodes: *variable nodes* (denoted by $X_1, \ldots, X_n$) and *factor nodes* (denoted by $f_1, \ldots, f_m$). Variable nodes represent random variables, while factor nodes represent functions that depend on the variables. More formally, a factor graph defines the non-normalized joint probability function

$$p(x_1, \ldots, x_n) = \prod_{i=1}^{m} f_i(\boldsymbol{x}_{\mathrm{ne}(f_i)}) \,, \tag{1}$$

where $\mathrm{ne}\,(f_i) \subseteq \{1, \ldots, n\}$ denotes the index set of variables that are connected to the factor node f_i, $\boldsymbol{x}_{\mathrm{ne}(f_i)}$ is list of values x_j where $j \in \mathrm{ne}\,(f_i)$. Moreover, $\mathrm{ne}\,(X_j) \subseteq \{1, \ldots, m\}$ denotes the set of factors f_i that are connected to the variable node X_j.

Marginals and Message Passing In message passing, we are interested to compute the non-normalized marginal $p_{X_j}(x_j)$ defined by

$$p_{X_j}(x_j) = \sum_{\{x_1\}} \cdots \sum_{\{x_{j-1}\}} \sum_{\{x_{j+1}\}} \cdots \sum_{\{x_n\}} p(x_1, \ldots, x_n) \tag{2}$$

If the factor graph is a tree, then we can easily compute $p_{X_j}(x_j)$ by recursively applying the distributive law

$$p_{X_j}(x_j) = \prod_{i \in \mathrm{ne}(X_j)} m_{f_i \to X_j}(x_j) \,, \tag{3}$$

$$m_{f_i \to X_j}(x_j) = \sum_{\{\boldsymbol{x}_{\mathrm{ne}(f_i) \setminus \{j\}}\}} \left[f_i(\boldsymbol{x}_{\mathrm{ne}(f_i)}) \cdot \prod_{k \in \mathrm{ne}(f_i) \setminus \{j\}} m_{X_k \to f_i}(x_k) \right] \,, \tag{4}$$

$$m_{X_j \to f_i}(x_j) = \prod_{k \in \mathrm{ne}(X_j) \setminus \{i\}} m_{f_k \to X_j}(x_j) \,, \tag{5}$$

Note that by virtue of (3) and (5) we have for any f_i and $X_j, j \in \mathrm{ne}\,(f_i)$,

$$p_{X_j}(x_j) = m_{f_i \to X_j}(x_j) \cdot m_{X_j \to f_i}(x_j) \,. \tag{6}$$

Normalization constant Often, we require a normalized joint probability $\widetilde{p}(x_1, \ldots, x_n) = \frac{1}{Z} \cdot p(x_1, \ldots, x_n)$ rather than a non-normalized joint probability $p(x_1, \ldots, x_n)$ where the normalization constant Z defined by

$$Z = \sum_{\{x_1\}} \cdots \sum_{\{x_n\}} p(x_1, \ldots, x_n) = \sum_{\{x_1\}} \cdots \sum_{\{x_n\}} \prod_{i=1}^{m} f_i(\boldsymbol{x}_{\mathrm{ne}(f_i)}) \,. \tag{7}$$

Note that by definition for any variable X_j, the normalization is also obtained from (2) by summing over all values of X_j. Thus, for every variable X_j the normalization constant is also given by

$$Z = \sum_{\{x_j\}} p_{X_j}(x_j) \,. \tag{8}$$

Looking at (6) and (4), we see that (8) can also be written as

$$Z = \sum_{\{x_j\}} m_{f_i \to X_j}(x_j) \cdot m_{X_j \to f_i}(x_j) = \sum_{\{\boldsymbol{x}_{\mathrm{ne}(f_i)}\}} \left[f_i(\boldsymbol{x}_{\mathrm{ne}(f_i)}) \cdot \prod_{k \in \mathrm{ne}(f_i)} m_{X_k \to f_i}(x_k) \right] \,, \tag{9}$$

where we used (4) for $m_{f_i \to X_j}(x_j)$. In graph language, (8) follows from making the variable node X_j the root of the factor tree and (9) follows from making the factor node f_i the root of the factor tree (and the summing over each of variables in the branches individually).

Normalization Constants in Message Passing

In addition to the n ways to compute the normalization constant using (8) or the m ways to compute the normalization constant using (9), we can also compute the normalization constant by message passing. Here we will present a result that allows us to compute the normalization constant by message passing in a distributed manner using arbitraty normalization of the messages. Thus, for every factor f_i and for every variable X_j, we introduce two scaled messages by virtue of

$$\widetilde{m}_{f_i \to X_j}(x_j) = \alpha_{i,j} \cdot m_{f_i \to X_j}(x_j)\,, \tag{10}$$

$$\widetilde{m}_{X_j \to f_i}(x_j) = \beta_{j,i} \cdot m_{X_j \to f_i}(x_j)\,.\,. \tag{11}$$

We are now in a position to state the main theorem for computing the normalization constant by message passing.

Theorem 1. *Given a factor tree, the normalization constant Z in terms of scaled messages* (10) *and* (11) *is given by*

$$Z = \left(\prod_{i=1}^{m} Z_{f_i}\right) \cdot \left(\prod_{j=1}^{n} Z_{X_j}\right)\,, \tag{12}$$

where the constants Z_{f_i} for all factors f_i and Z_{X_j} for all variables X_j are given by

$$Z_{f_i} = \frac{\sum_{\{\boldsymbol{x}_{\mathrm{ne}(f_i)}\}} \left[f_i(\boldsymbol{x}_{\mathrm{ne}(f_i)}) \cdot \prod_{k \in \mathrm{ne}(f_i)} \widetilde{m}_{X_k \to f_i}(x_k)\right]}{\sum_{\{\boldsymbol{x}_{\mathrm{ne}(f_i)}\}} \left[\prod_{k \in \mathrm{ne}(f_i)} \widetilde{m}_{f_i \to X_k}(x_k) \cdot \widetilde{m}_{X_k \to f_i}(x_k)\right]}\,, \tag{13}$$

$$Z_{X_j} = \sum_{\{x_j\}} \prod_{i \in \mathrm{ne}(X_j)} \widetilde{m}_{f_i \to X_j}(x_j)\,. \tag{14}$$

Proof. We will start by simplifying (13) in terms of the $\alpha_{i,k}, k \in \mathrm{ne}(f_i)$ and Z. More specifically, we see that

$$\begin{aligned}
Z_{f_i} &= \frac{\sum_{\{\boldsymbol{x}_{\mathrm{ne}(f_i)}\}} \left[f_i(\boldsymbol{x}_{\mathrm{ne}(f_i)}) \cdot \prod_{k \in \mathrm{ne}(f_i)} \beta_{k,i} \cdot m_{X_k \to f_i}(x_k)\right]}{\sum_{\{\boldsymbol{x}_{\mathrm{ne}(f_i)}\}} \left[\prod_{k \in \mathrm{ne}(f_i)} \alpha_{i,k} \cdot m_{f_i \to X_k}(x_k) \cdot \beta_{k,i} \cdot m_{X_k \to f_i}(x_k)\right]} \\
&= \frac{\left(\prod_{k \in \mathrm{ne}(f_i)} \beta_{k,i}\right) \cdot \left(\sum_{\{\boldsymbol{x}_{\mathrm{ne}(f_i)}\}} \left[f_i(\boldsymbol{x}_{\mathrm{ne}(f_i)}) \cdot \prod_{k \in \mathrm{ne}(f_i)} m_{X_k \to f_i}(x_k)\right]\right)}{\left(\prod_{k \in \mathrm{ne}(f_i)} \alpha_{i,k}\right) \cdot \left(\prod_{k \in \mathrm{ne}(f_i)} \beta_{k,i}\right) \cdot \left(\sum_{\{\boldsymbol{x}_{\mathrm{ne}(f_i)}\}} \left[\prod_{k \in \mathrm{ne}(f_i)} m_{f_i \to X_k}(x_k) \cdot m_{X_k \to f_i}(x_k)\right]\right)} \\
&= \frac{\sum_{\{\boldsymbol{x}_{\mathrm{ne}(f_i)}\}} \left[f_i(\boldsymbol{x}_{\mathrm{ne}(f_i)}) \cdot \prod_{k \in \mathrm{ne}(f_i)} m_{X_k \to f_i}(x_k)\right]}{\left(\prod_{k \in \mathrm{ne}(f_i)} \alpha_{i,k}\right) \cdot \left(\prod_{k \in \mathrm{ne}(f_i)} \left[\sum_{\{x_k\}} m_{f_i \to X_k}(x_k) \cdot m_{X_k \to f_i}(x_k)\right]\right)} \\
&= \frac{Z}{\left(\prod_{k \in \mathrm{ne}(f_i)} \alpha_{i,k}\right) \cdot Z^{|\mathrm{ne}(f_i)|}}\,,
\end{aligned}$$

where we used (10) and (11) in the first line, factored out all terms $\beta_{k,i}$ and $\alpha_{k,i}$ with $k \in \mathrm{ne}(f_i)$ in the second line, swapped the summation over all $\{\boldsymbol{x}_{\mathrm{ne}(f_i)}\}$ with the product as each factor only depends on $x_k, k \in \mathrm{ne}(f_i)$ in the third line, and used (9) in the numerator and (8) together with (6) in the denominator in the last line.

Similarly, we will simplify (14) in terms of the $\alpha_{i,j}, i \in \mathrm{ne}(X_j)$ and Z as follows:

$$\begin{aligned}
Z_{X_j} &= \sum_{\{x_j\}} \prod_{i \in \mathrm{ne}(X_j)} \alpha_{i,j} \cdot m_{f_i \to X_j}(x_j) \\
&= \left(\prod_{i \in \mathrm{ne}(X_j)} \alpha_{i,j}\right) \cdot \left(\sum_{\{x_j\}} \prod_{i \in \mathrm{ne}(X_j)} m_{f_i \to X_j}(x_j)\right) \\
&= \left(\prod_{i \in \mathrm{ne}(X_j)} \alpha_{i,j}\right) \cdot Z\,,
\end{aligned}$$

where we used (10) in the first line, factored out all terms $\alpha_{i,j}$ with $i \in \mathrm{ne}\left(X_j\right)$ in the second line, and used (8) together with (3) in the last line.

Now, putting this together we see that

$$\begin{aligned}\left(\prod_{i=1}^{m} Z_{f_i}\right) \cdot \left(\prod_{j=1}^{n} Z_{X_j}\right) &= \left(\prod_{i=1}^{m} \frac{Z}{\left(\prod_{k\in\mathrm{ne}(f_i)} \alpha_{i,k}\right) \cdot Z^{|\mathrm{ne}(f_i)|}}\right) \cdot \left(\prod_{j=1}^{n} \left(\prod_{i\in\mathrm{ne}\left(X_j\right)} \alpha_{i,j}\right) \cdot Z\right) \\ &= \frac{Z^{n+m}}{\prod_{i=1}^{m} \prod_{k\in\mathrm{ne}(f_i)} Z^{|\mathrm{ne}(f_i)|}} \\ &= \frac{Z^{n+m}}{Z^{n+m-1}} = Z\,,\end{aligned}$$

where we used the fact that

$$\prod_{i=1}^{m} \prod_{k\in\mathrm{ne}(f_i)} \alpha_{i,k} = \prod_{j=1}^{n} \prod_{i\in\mathrm{ne}\left(X_j\right)} \alpha_{i,j}$$

in the second line because the two products enumerate all edges of the factor graph, and used the fact that the factor graph is a tree and therefore for the total number of edges (enumerated by $\{(i,j)|i \in \{1,\ldots,m\}, j \in \mathrm{ne}\left(f_i\right)\}$) is $n + m - 1$ in the last line. $\square$